

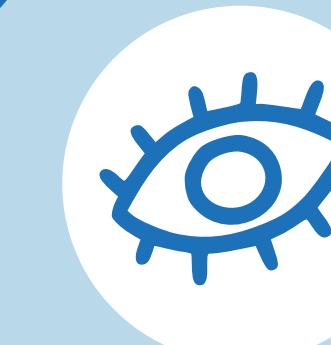
CLOUD CAMPUS

# REACT NATIVE

*SAIKOU BALDÉ CONSULTANT DEV FULLSTACK*



# OBJECTIF



MAÎTRISER L'ENVIRONNEMENT  
DE DEV REACT NATIVE



ZOOM FONCTIONNALITÉES FRONT



LA PRATIQUE DU DEV REACT NATIVE



PUBLICATION SUR LES PLATEFORMES

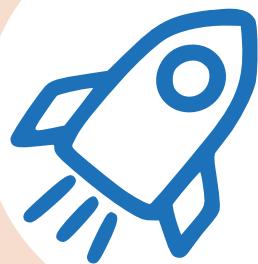
# LES GRANDES LIGNES



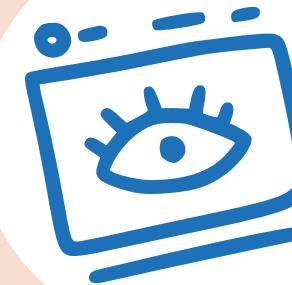
ENV DEV POUR DÉMARRER UN PROJET



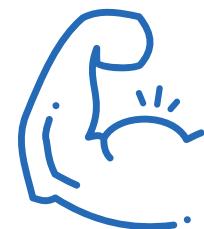
LES PLUGINS TIERS DANS REACT NATIVE



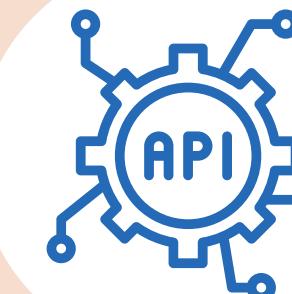
HELLO WORLD APP ET LE DEVICE



UTILISATION DE REDUX DANS REACT NATIVE



ZOOM SUR LES FONCTIONNALITÉS



OPTIMISATION ET MISE EN PROD

CAS PRATIQUES - FIL ROUGE PROJET

# REACT NATIVE

## MISE EN PLACE D'UN ENVIRONNEMENT DE DEV

**React Native** permet aux développeurs maîtrisant React de créer des applications natives. Dans le même temps, les développeurs android et ios natifs peuvent utiliser React Native pour obtenir la parité entre les plates-formes natives en écrivant une seule fois des fonctionnalités communes.

*La meilleure façon d'utiliser React Native est de l'utiliser avec un Framework du style Expo même si l'on peut l'utiliser sans*



<https://docs.expo.dev/get-started/introduction/>

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

<https://reactnative.dev/docs/set-up-your-environment>

<https://reactnative.dev/docs/getting-started-without-a-framework>

# NPX ET CREATE EXPO APP

## MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

**EXPO**

*Expo est un framework qui facilite le développement d'applications Android et iOS. EXPO fournit un routage basé sur des fichiers, une bibliothèque standard de modules natifs et bien plus encore. Expo est open source avec une communauté active sur GitHub et Discord.*



*Beaucoup d'applications célèbres ont été développées avec Expo, parmi lesquelles*

- **Facebook** : l'application mobile de Facebook
- **Uber Eats** : l'application de livraison de repas d'Uber
- **AirBnB** : l'application mobile d'AirBnB
- **TikTok** : l'application mobile de TikTok

# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

## Prérequis

- **Node.js** : Assurez-vous que Node.js est installé sur votre machine. Vous pouvez le télécharger depuis nodejs.org.
- **npm** : npm est inclus avec Node.js, donc si vous avez installé Node.js, vous devez déjà avoir npm.
- **Expo CLI** : Vous aurez besoin d'Expo CLI pour créer et gérer vos projets Expo
- **macOS, Windows (Powershell et WSL 2) et Linux** sont pris en charge par Expo pour la création d'appli React Native.

```
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ node -v
v18.19.1
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npm -v
9.2.0
○ balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ █

● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npx -v
9.2.0
```

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

# NPX ET CREATE EXPO APP

## MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

### Création de Projet

Nous vous recommandons de commencer par le projet par défaut créé par `create-expo-app`. Le projet par défaut inclut un exemple de code pour vous aider à démarrer.

Pour créer un nouveau projet, exécutez la commande suivante :

```
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npx create-expo-app@latest
Need to install the following packages:
  create-expo-app@2.3.5
Ok to proceed? (y) y
✓ What is your app named? ... cloud-capus-react-native
✓ Downloaded and extracted project files.
> npm install
```

```
added 1545 packages, and audited 1546 packages in 52s
147 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
Your project is ready!
To run your project, navigate to the directory and run one of the following npm commands.
- cd cloud-capus-react-native
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web
```

**NB :** Le modèle de projet par défaut inclut la bibliothèque `Expo Router` installée pour créer des applications multi-écrans avec `Expo`. Vous pouvez choisir un modèle différent en utilisant l'option `--template`. Par exemple, en utilisant cette option, vous pouvez choisir le modèle vierge "BLANK" pour créer un projet avec le minimum de bibliothèques requises installées et sans navigation configurée.



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

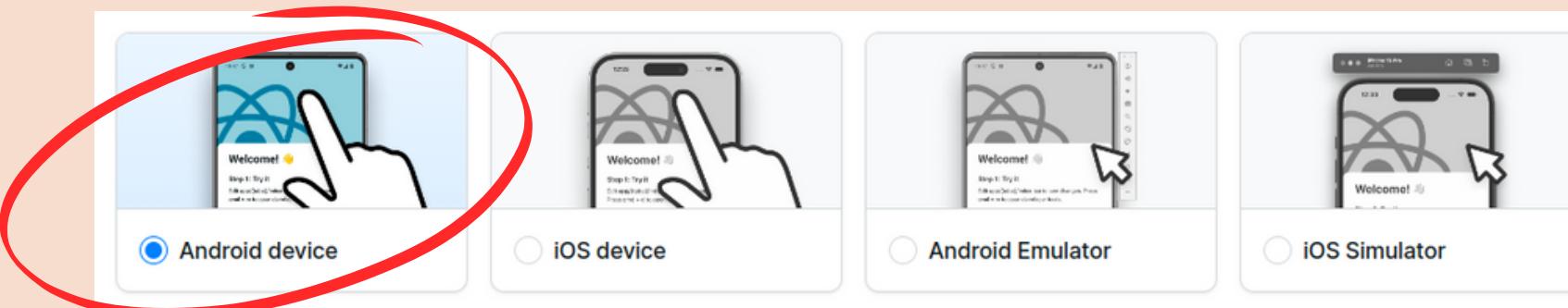
# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

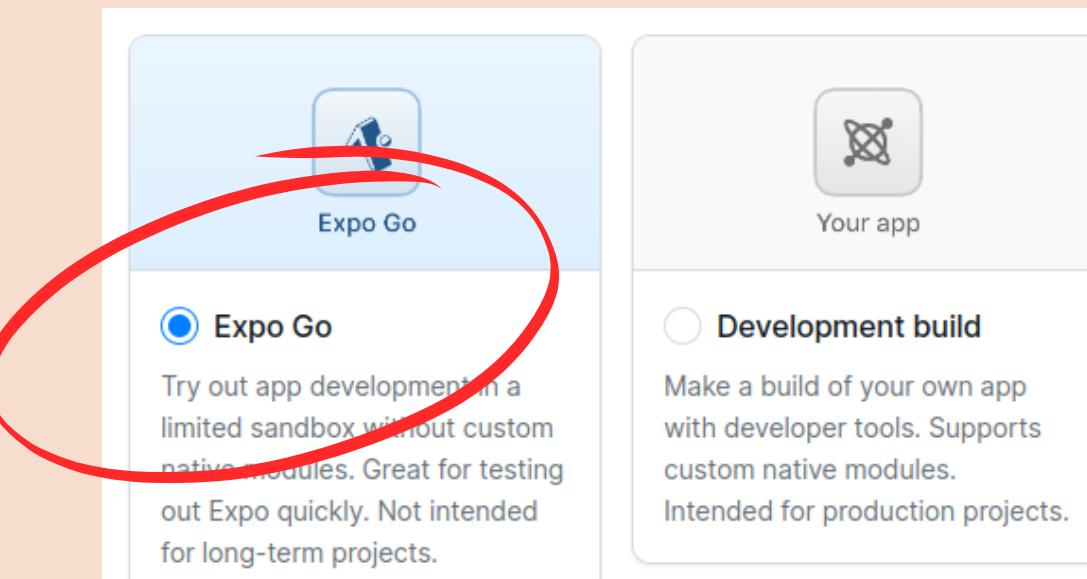
## Configurez votre environnement

Configurons ensemble un environnement de développement local pour exécuter votre projet sur Android et iOS.

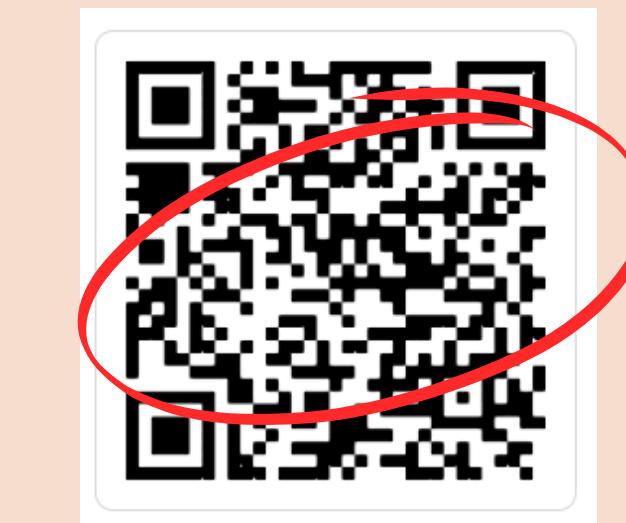
### 1 Dans quel environnement faites vous du dev ?



### 2 Quel outils souhaitez vous utiliser pour le dev ?



### 3 Configurer un appareil Android avec Expo Go



Scannez le code QR pour télécharger l'application depuis le Google Play Store, ou visitez la page Expo Go sur le Google Play Store.

# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI (METRO BUNDLER)

## Commencez à développer

Démarrer un serveur de développement :

Pour démarrer le serveur de développement, exécutez la commande suivante :

```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/cloud-capus-react-native$ npx expo start
Starting project at /home/balde/www/react_native/cloud-capus-react-native
Starting Metro Bundler
QR code to scan: http://10.16.185.152:8081
Press 'q' to quit.

Metro waiting on exp://10.16.185.152:8081
Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

Web is waiting on http://localhost:8081
Using Expo Go
Press s | switch to development build

Press a | open Android
Press w | open web

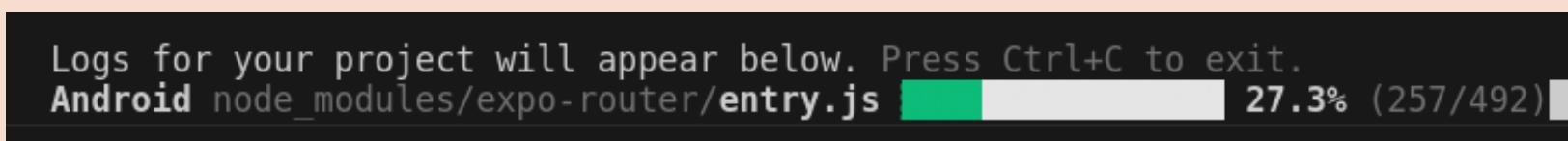
Press j | open debugger
Press r | reload app
Press m | toggle menu
Press o | open project code in your editor
Press ? | show all commands

Logs for your project will appear below. Press Ctrl+C to exit.
```

Ouvrez l'application sur votre appareil ""Mobile

Après avoir exécuté la commande ci-dessus, vous verrez un code QR dans votre terminal. Scannez ce code QR pour ouvrir l'application sur votre appareil.

Si vous utilisez un émulateur Android ou un simulateur iOS, vous pouvez appuyer respectivement sur a ou i pour ouvrir l'application.



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

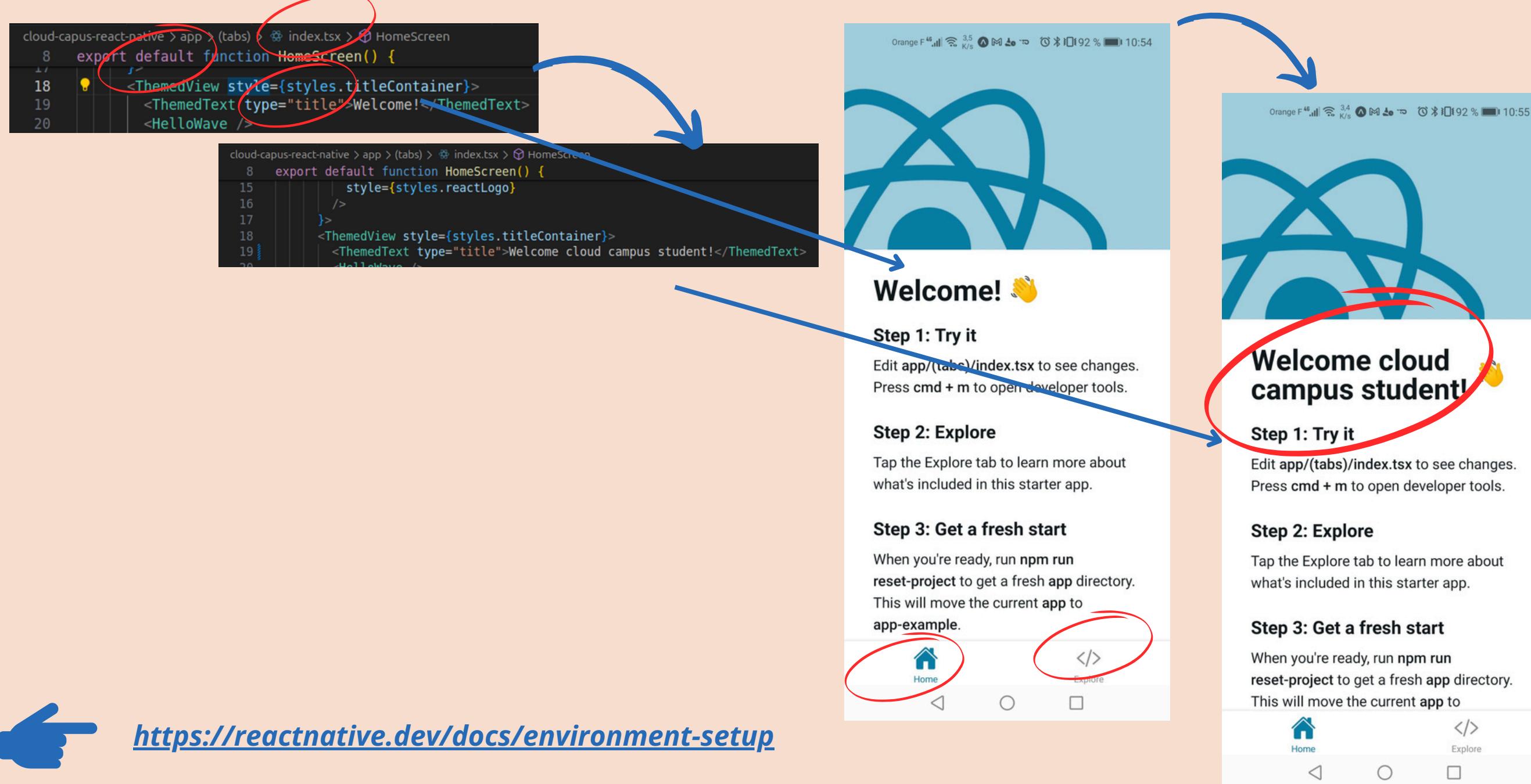
# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

## Commencez à développer

*Effectuez votre premier changement*

Ouvrez le fichier app/(tabs)/index.tsx dans votre éditeur de code et apportez une modification.

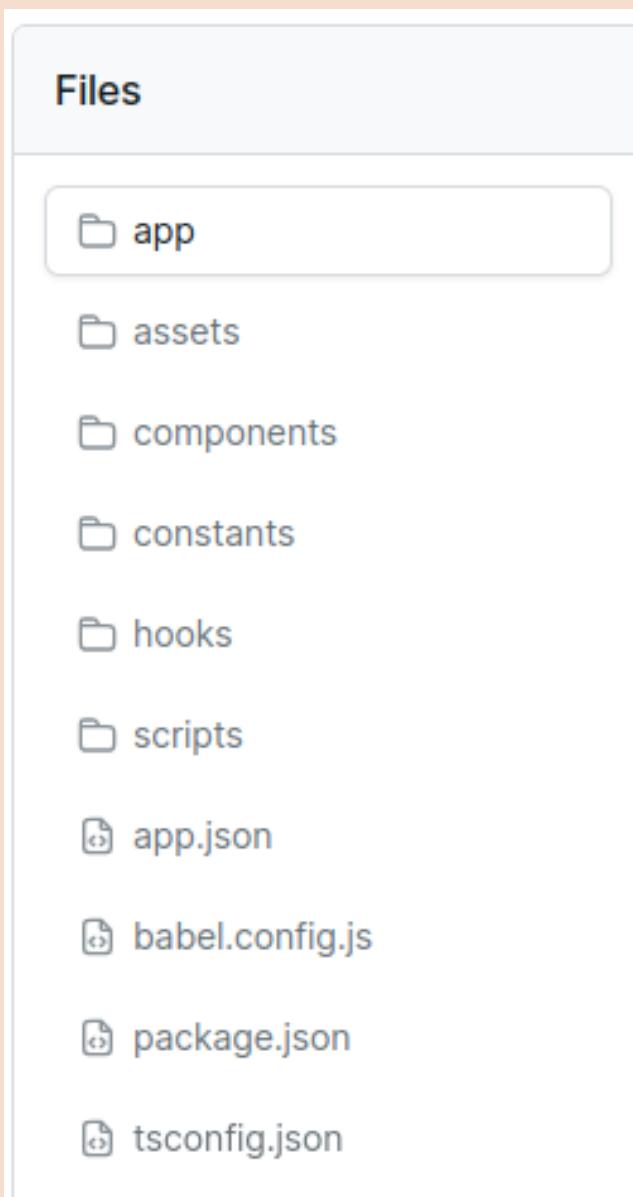


# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

## Structure du fichier

Ci-dessous, vous pouvez vous familiariser avec la structure des fichiers du projet par défaut :



**app** contient la navigation de l'application, basée sur des fichiers.

L'application dispose de deux itinéraires définis par deux fichiers : **app/(tabs)/index.tsx et app/(tabs)/explore.tsx**. Le fichier de mise en page dans **app/(tabs)/\_layout.tsx** configure le navigateur d'onglets.

**assets** contient **adaptive-icon.png** utilisé pour Android et un **icon.png** utilisé pour iOS comme icônes d'application. Il contient également **splash.png** qui est une image pour l'écran de démarrage du projet et un **favicon.png** si l'application s'exécute dans un navigateur.

**components** contient des composants React Native, comme **ThemedText.tsx**, qui crée des éléments de texte qui utilisent le jeu de couleurs de l'application en modes clair et sombre.

**constants** Contient **Colors.ts**, qui contient une liste de valeurs de couleur dans toute l'application.

**hooks** contient des React Hooks, qui permettent de partager un comportement commun entre les composants. Par exemple, **useThemeColor()** est un hook qui renvoie une couleur basée sur le thème actuel.

**scripts** contient **reset-project.js**, qui peut être exécuté avec **npx run reset-project**. Ce script déplacera le répertoire de l'application vers **app-example** et créera un nouveau répertoire d'application avec un fichier **index.tsx**.

**app.json** contient les options de configuration du projet et est appelé configuration de l'application. Ces options modifient le comportement de votre projet lors du développement, de la création, de la soumission et de la mise à jour de votre application.

**babel.config.js** Applique le préréglage **babel-preset-expo** qui étend le préréglage React Native par défaut et ajoute la prise en charge des décorateurs, des packages Web d'arborescence et le chargement d'icônes de police avec des dépendances natives facultatives si elles sont installées.

**package.json** Le fichier **package.json** contient les dépendances, les scripts et les métadonnées du projet. Chaque fois qu'une nouvelle dépendance ajoutée à votre projet, elle sera ajoutée à ce fichier.

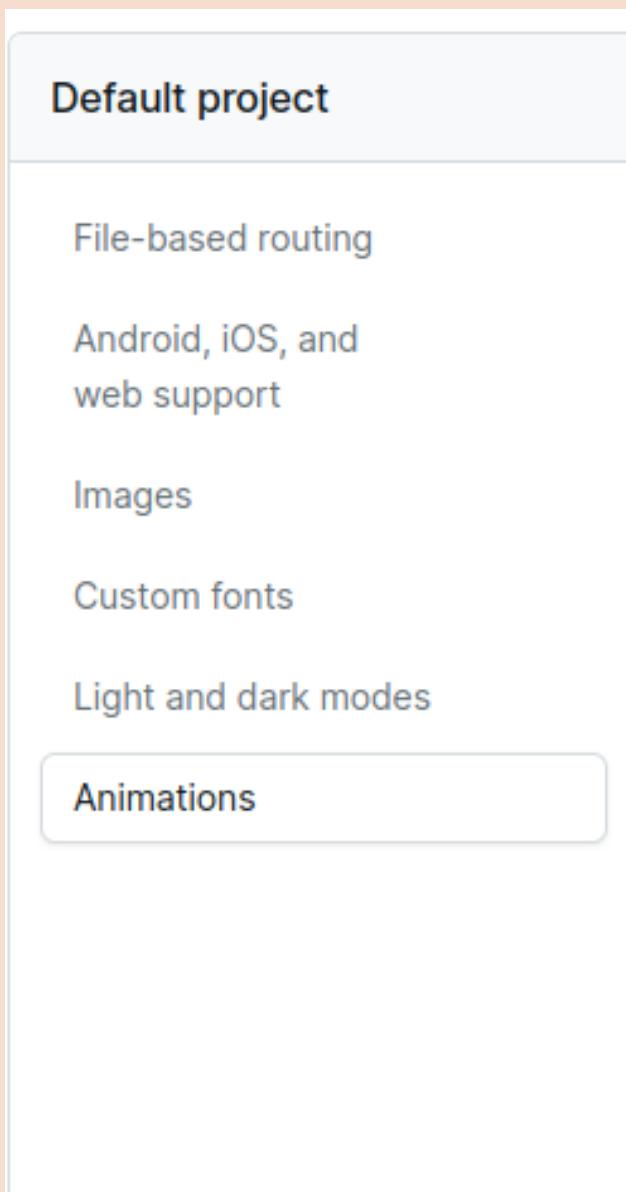
**tsconfig.json** contient les règles que **TypeScript** utilisera pour appliquer la sécurité des types tout au long du projet.

# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

## Fonctionnalités du template

Le modèle de projet par défaut présente les fonctionnalités suivantes :



**File-based routing** L'application dispose de **deux écrans** : `app/(tabs)/index.tsx` et `app/(tabs)/explore.tsx`. Le fichier de mise en page dans `app/(tabs)/_layout.tsx` configure le navigateur d'onglets.

**Android, iOS, and web support** Vous pouvez ouvrir ce projet sur Android, iOS et le Web. Pour ouvrir la version Web, appuyez sur `w` dans le terminal après avoir exécuté le projet.

**Images** Pour les images statiques, vous pouvez utiliser les suffixes `@2x` et `@3x` pour fournir des images pour différentes densités d'écran.

**Custom fonts** Ouvrez `app/_layout.tsx` pour voir comment charger des polices personnalisées.

**Light and dark modes** Ce modèle prend en charge les modes clair et sombre. Le hook `useColorScheme()` vous permet d'inspecter le jeu de couleurs actuel de l'utilisateur, afin que vous puissiez ajuster les couleurs de l'interface utilisateur en conséquence. Apprendre encore plus

**Animations** Le composant `composants/HelloWave.tsx` utilise l'API `Animated` pour créer une animation de main agitante. Sur iOS, le composant `ParallaxScrollView.tsx` crée un effet de parallaxe pour l'image d'en-tête.

# NPX ET CREATE EXPO APP

MISE EN PLACE D'UN ENVIRONNEMENT DE DEV - EXPO CLI

## Réinitialisez votre projet

*Vous pouvez supprimer le code passe-partout et repartir à zéro avec un nouveau projet. Exécutez la commande suivante pour réinitialiser votre projet : Terminal*

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/cloud-capus-react-native$ tree -L 1
.
├── app
├── app.json
├── assets
├── babel.config.js
├── components
├── constants
├── expo-env.d.ts
├── hooks
└── node_modules
    ├── package.json
    └── package-lock.json
    README.md
    scripts
    tsconfig.json

8 directories, 7 files
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/cloud-capus-react-native$ npm run reset-project
> cloud-capus-react-native@1.0.0 reset-project
> node ./scripts/reset-project.js

/app moved to /app-example.
New /app directory created.
app/index.tsx created.
app/_layout.tsx created.
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/cloud-capus-react-native$ tree -L 1
.
├── app
├── app-example
├── app.json
├── assets
├── babel.config.js
├── components
├── constants
├── expo-env.d.ts
├── hooks
└── node_modules
    ├── package.json
    └── package-lock.json
    README.md
    scripts
    tsconfig.json

9 directories, 7 files
```

```
cloud-capus-react-native > {} package.json > ...
1  [
2   "name": "cloud-capus-react-native",
3   "main": "expo-router/entry",
4   "version": "1.0.0",
5   "scripts": {
6     "start": "expo start",
7     "reset-project": "node ./scripts/reset-project.js",
8     "android": "expo start --android",
9     "ios": "expo start --ios",
10    "web": "expo start --web",
11    "test": "jest --watchAll",
12    "lint": "expo lint"
13  },]
```

*Il ne nous reste plus qu'à créer et développer des appli après avoir mis en place l'environnement de dev*

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

# NPX ET CREATE EXPO APP

## UTILISATION DE EXPO CLI DANS REACT NATIVE - AWESOMEPROJECT

```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX507ZC4:~/www/react_native$ npx create-expo-app AwesomeProject
Need to install the following packages:
  create-expo-app@2.3.5
Ok to proceed? (y) y
✓ Downloaded and extracted project files.
> npm install
npm WARN deprecated @babel/plugin-proposal-numeric-separator@7.18.6 · This proposal has been merged to the
```

```
added 1524 packages, and audited 1525 packages in 1m

138 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

✓ Your project is ready!

To run your project, navigate to the directory and run one of the following npm commands.

- cd AwesomeProject
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web
```

# NPX ET CREATE EXPO APP

## UTILISATION DE EXPO CLI DANS REACT NATIVE - AWESOMEPROJECT

### 1 Start a development server

To start the development server, run the following command:

Terminal

Copy

```
- npx expo start
```

```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/AwesomeProject$ npx expo start
Starting project at /home/balde/www/react_native/AwesomeProject
Starting Metro Bundler
The following packages should be updated for best compatibility with the installed expo version:
  expo@51.0.2 - expected version: ~51.0.4
  expo-font@12.0.4 - expected version: ~12.0.5
  expo-router@3.5.11 - expected version: ~3.5.12
Your project may not work correctly until you install the expected versions of the packages.

QR code
Metro waiting on exp://10.16.185.152:8081
Scan the QR code above with Expo Go (Android) or the Camera app (iOS)
Web is waiting on http://localhost:8081
```

### 2 Open the app on your device

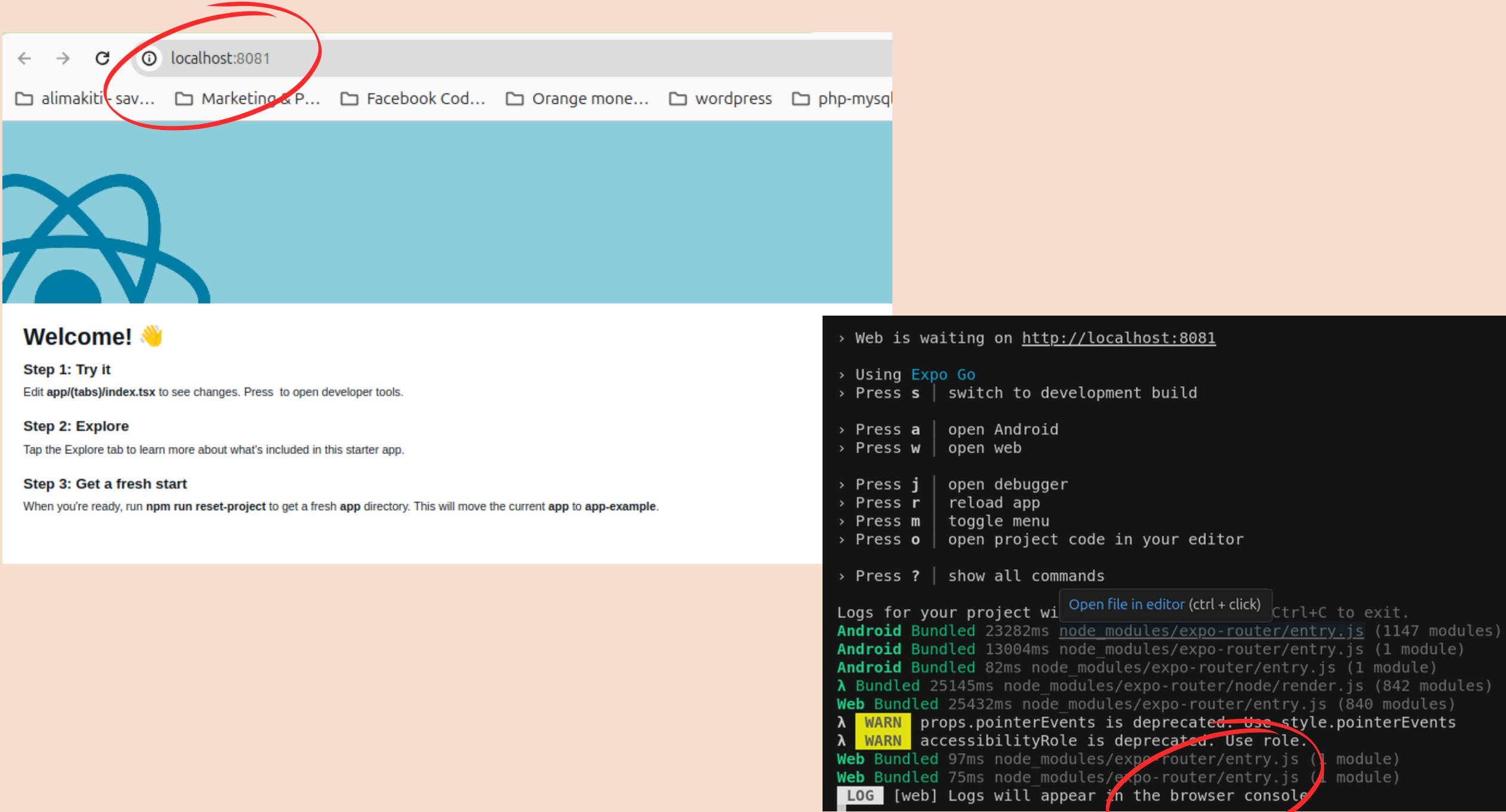
After running the command above, you will see a QR code in your terminal. Scan this QR code to open the app on your device.

If you're using an Android Emulator or iOS Simulator, you can press **a** or **i** respectively to open the app.

Having problems?

# NPX ET CREATE EXPO APP

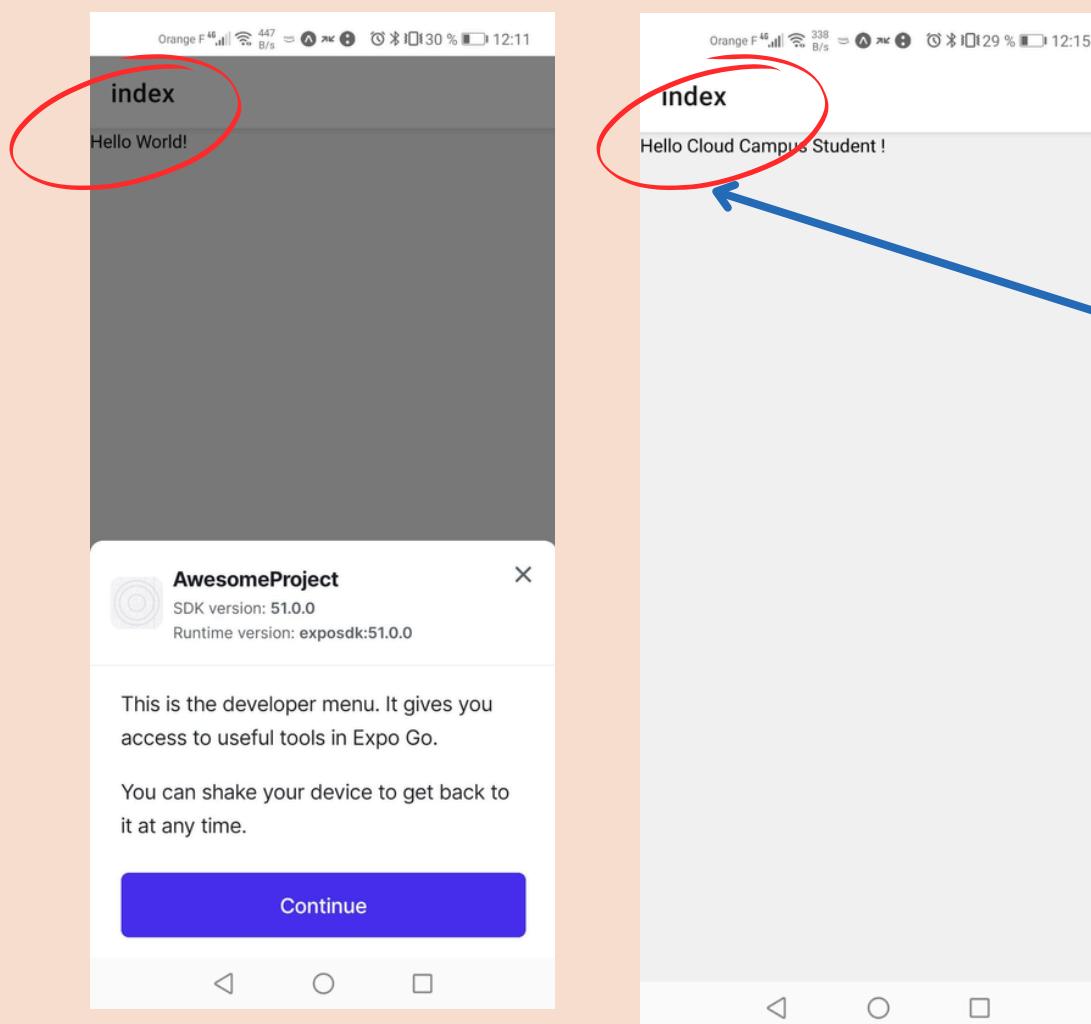
UTILISATION DE EXPO CLI DANS REACT NATIVE - AWESOMEPROJECT



# NPX ET CREATE EXPO APP

## UTILISATION DE EXPO CLI DANS REACT NATIVE - AWESOMEPROJECT

```
batde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/AwesomeProject$ ls
app           app.json   babel.config.js  constants    hooks       package.json  README.md  tsconfig.json
app-example   assets     components      expo-env.d.ts  node_modules  package-lock.json scripts
batde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/AwesomeProject$
```



The diagram illustrates the relationship between code changes and their immediate effect on the app. It features four panels: a terminal window at the top showing project files; a developer menu on the bottom-left; a main app screen with the text 'Hello Cloud Campus Student !' on the bottom-right; and a code editor in the center. A blue arrow points from the code editor towards the developer menu, signifying that modifying the code leads to an instant update in the app's developer interface.

```
home > balde > www > react_native > AwesomeProject > app > index.tsx > Index
1 import {ScrollView, Text, View} from "react-native";
2
3 export default function Index() {
4   return (
5     <ScrollView contentInsetAdjustmentBehavior="automatic">
6       <View>
7         <Text>Hello World!</Text>
8       </View>
9     </ScrollView>
10   );
11 }
```

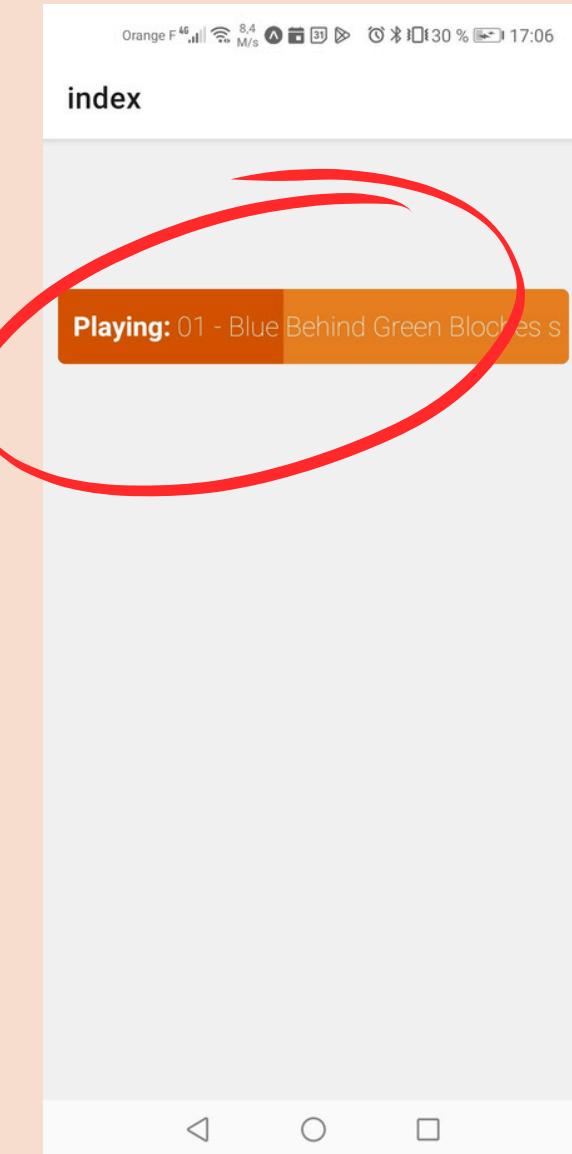
```
home > balde > www > react_native > AwesomeProject > app > index.tsx > Index
1 import {ScrollView, Text, View} from "react-native";
2
3 export default function Index() {
4   return (
5     <ScrollView contentInsetAdjustmentBehavior="automatic">
6       <View>
7         <Text>Hello Cloud Campus Student !</Text>
8       </View>
9     </ScrollView>
10   );
11 }
```

# NPX ET CREATE EXPO APP

UTILISATION DE EXPO CLI DANS REACT NATIVE - AWESOMEPROJECT

```
home > balde > www > react_native > AwesomeProject > app > index.tsx > styles
1 import React from 'react';
2 import {StyleSheet, Text, View} from "react-native";
3
4 export default function Index() {
5   const name = "01 - Blue Behind Green Bloches s";
6   const styles = StyleSheet.create({
7     container: {
8       margin: 10,
9       marginTop: 100,
10      backgroundColor: "#e67e22",
11      borderRadius: 5,
12    },
13    innerContainer: {
14      backgroundColor: "#d35400",
15      height: 50,
16      width: 150,
17      borderTopLeftRadius: 5,
18      borderBottomLeftRadius: 5,
19    },
20    title: {
21      fontSize: 18,
22      fontWeight: '200',
23      color: '#fff',
24      position: 'absolute',
25      backgroundColor: 'transparent',
26      top: 12,
27      left: 10,
28    },
29    subtitle: {
30      fontWeight: 'bold',
31    },
32  });
33}
```

```
home > balde > www > react_native > AwesomeProject > app > index.tsx > styles
4 export default function Index() {
6   const styles = StyleSheet.create({
31   },
32   );
33
34   return (
35     <View style={styles.container}>
36       <View style={styles.innerContainer} />
37       <Text style={styles.title}>
38         <Text style={styles.subtitle}>Playing:</Text> {name}
39       </Text>
40     </View>
41   );
42 }
```



# NPX ET CREATE EXPO APP

## TEMPLATE BLANK (TYPESCRIPT - EXPO CLI - REACT NATIVE)

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npx create-expo-app --template
✓ Choose a template: > Blank (TypeScript)
✓ What is your app named? ... expo-template-app
✓ Downloaded and extracted project files.
> npm install
```

```
added 1192 packages, and audited 1193 packages in 36s

131 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

✓ Your project is ready!

To run your project, navigate to the directory and run one of the following npm commands.

- cd expo-template-app
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development
- npm run web
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ cd expo-template-app/
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ ls
  app.json  App.tsx  assets  babel.config.js  node_modules  package.json  package-lock.json  tsconfig.json
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npm start
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npm start
  > expo-template-app@1.0.0 start
  > expo start

Starting project at "/home/balde/www/react_native/expo-template-app"
Starting Metro Bundler...
The following packages should be updated for best compatibility with the installed expo version:
  typescript@5.4.5 - expected version: ~5.3.3
Your project may not work correctly until you install the expected versions of the packages.
```

```
> Using Expo Go
Press s | switch to development build

> Press a | open Android
> Press w | open web

> Press j | open debugger
> Press r | reload app
> Press m | toggle menu
> Press o | open project code in your editor

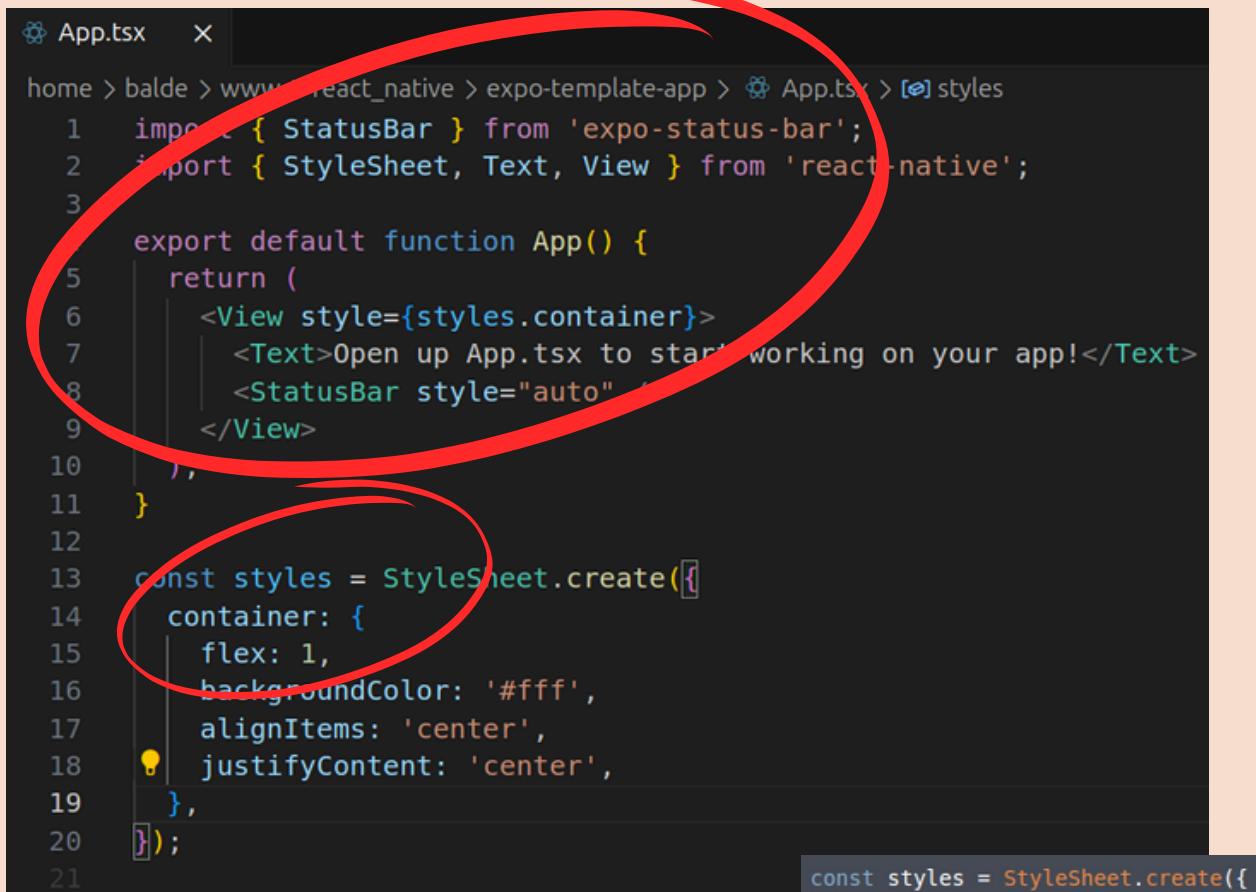
> Press ? | show all commands
```



```
> Metro waiting on exp://10.16.185.152:8081
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)
```

# NPX ET CREATE EXPO APP

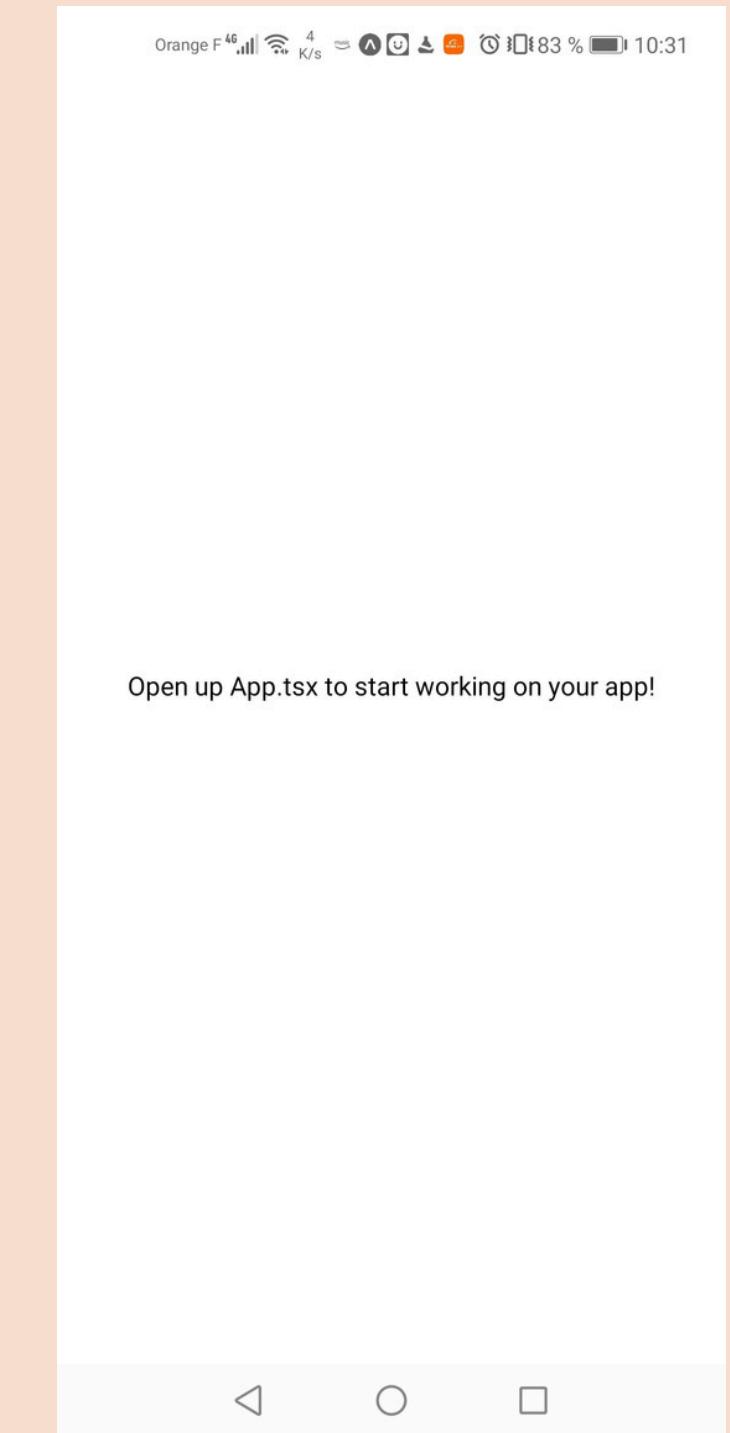
TEMPLATE BLANK (TYPESCRIPT - EXPO CLI - REACT NATIVE)



```
App.tsx  X

home > balde > www > react_native > expo-template-app > App.tsx > styles
1 import { StatusBar } from 'expo-status-bar';
2 import { StyleSheet, Text, View } from 'react-native';
3
4 export default function App() {
5   return (
6     <View style={styles.container}>
7       <Text>Open up App.tsx to start working on your app!</Text>
8       <StatusBar style="auto" />
9     </View>
10   ),
11 }
12
13 const styles = StyleSheet.create([
14   container: {
15     flex: 1,
16     backgroundColor: '#fff',
17     alignItems: 'center',
18     justifyContent: 'center',
19   },
20 ]);
21
```

const styles = StyleSheet.create({
 container: {
 flex: 1,
 backgroundColor: "#fff",
 alignItems: "center",
 justifyContent: "center",
 },
 text: { fontWeight: "bold" },
});



# NPX ET CREATE EXPO APP

## L'OUTILS SNACK EXPO - DÉVELOPPER DANS UN ENV CLOUD

**Expo Snack** est un outil en ligne très pratique pour développer et tester des applications React Native directement dans votre navigateur sans avoir besoin de configurer un environnement de développement complet.

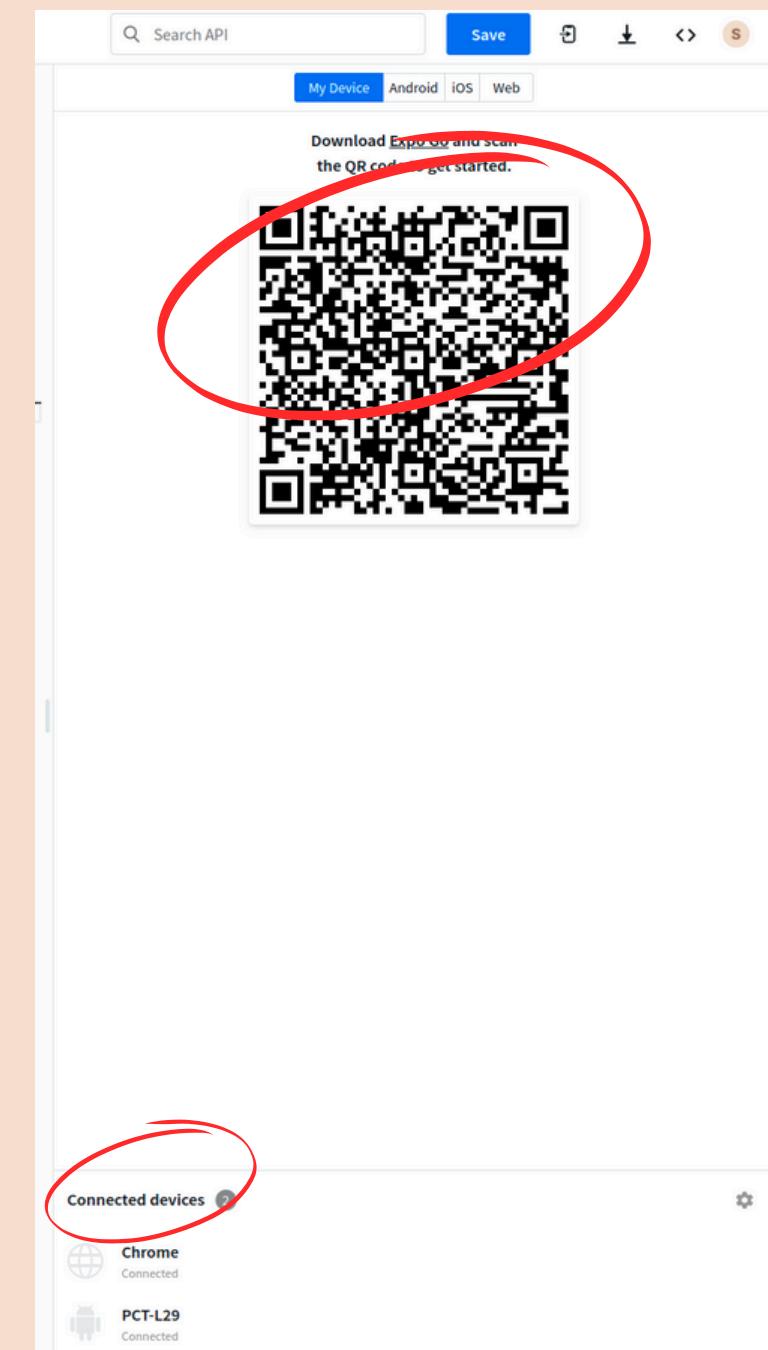
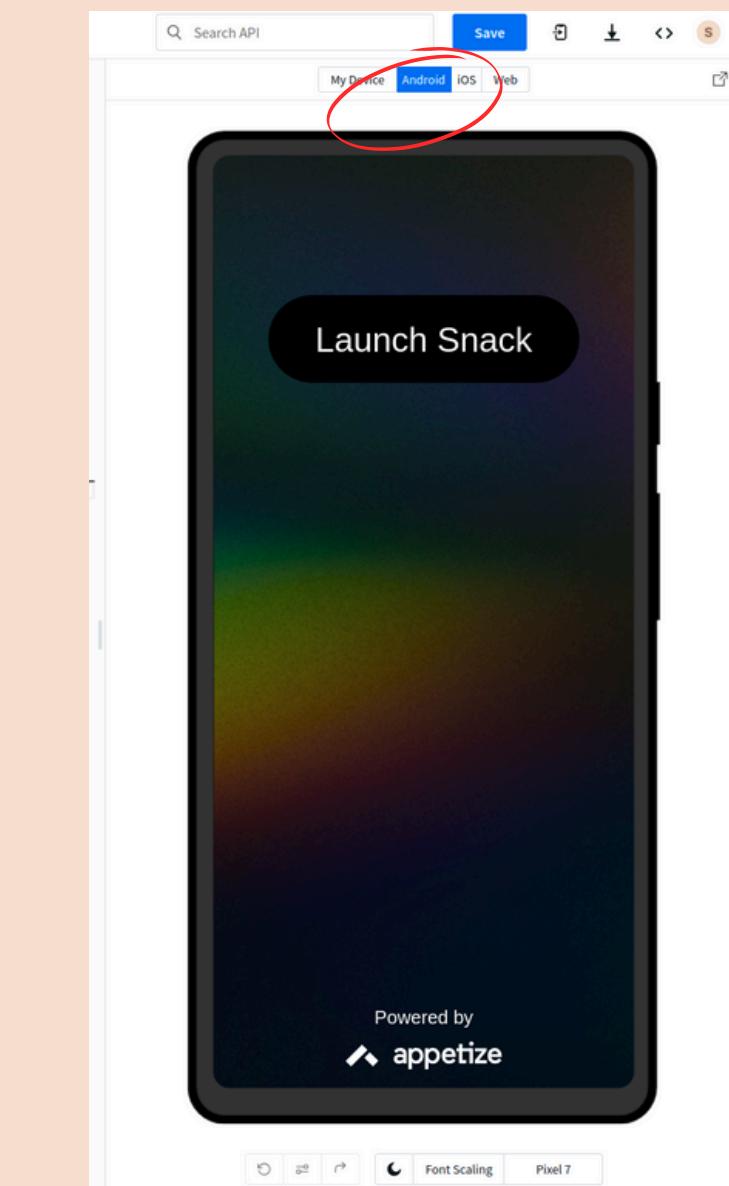
The screenshot shows the Expo Snack editor interface. At the top, there's a title bar with the project name "laughing violet charros" and a note "Not saved yet.". Below it is a sidebar with "Open files" (App.js) and "Project" sections, and icons for "Assets", "components", "App.js", "package.json", and "README.md". The main area contains the following code:

```
import { Text, SafeAreaView, StyleSheet } from 'react-native';
// You can import supported modules from npm
import { Card } from 'react-native-paper';
// or any files within the Snack
import AssetExample from './components/AssetExample';

export default function App() {
  return (
    <SafeAreaView style={styles.container}>
      <Text style={styles.paragraph}>
        Change code in the editor and watch it change on your phone! Save to get a shareable url.
      </Text>
      <Card>
        <AssetExample />
      </Card>
    </SafeAreaView>
  );
}

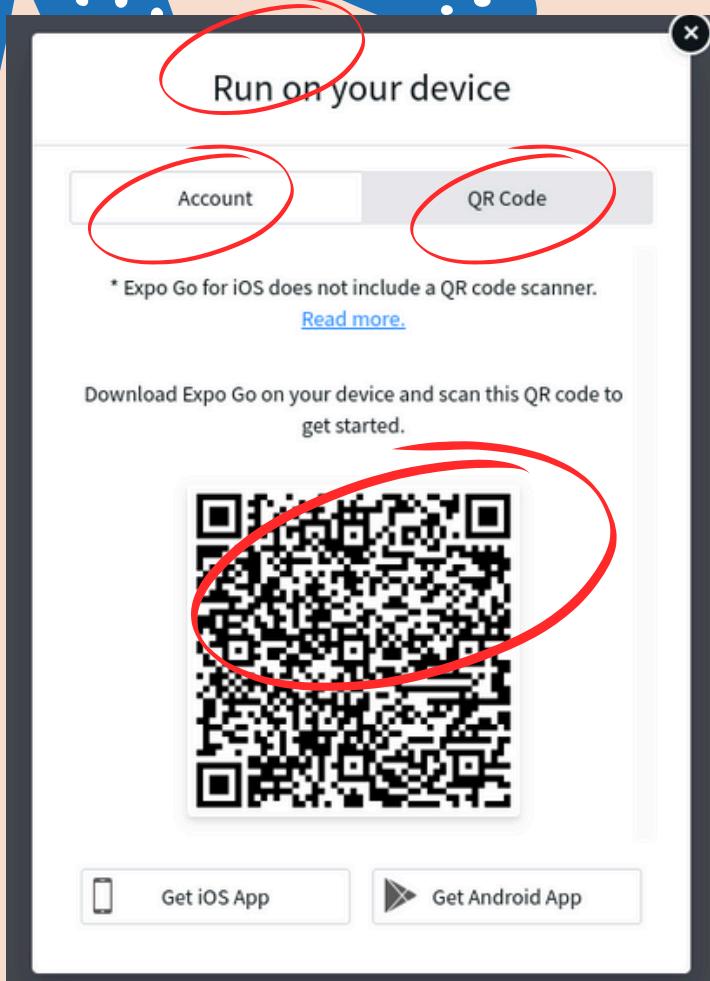
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    backgroundColor: '#ecf0f1',
    padding: 8,
  },
  paragraph: {
    margin: 24,
    fontSize: 18,
    fontWeight: 'bold',
    textAlign: 'center',
  },
});
```

At the bottom, there's a message: "Change code in the editor and watch it change on your phone! Save to get a shareable url." and a note: "Local files and assets can be imported by dragging and dropping them into the editor". A small icon of a Rubik's cube is also present.

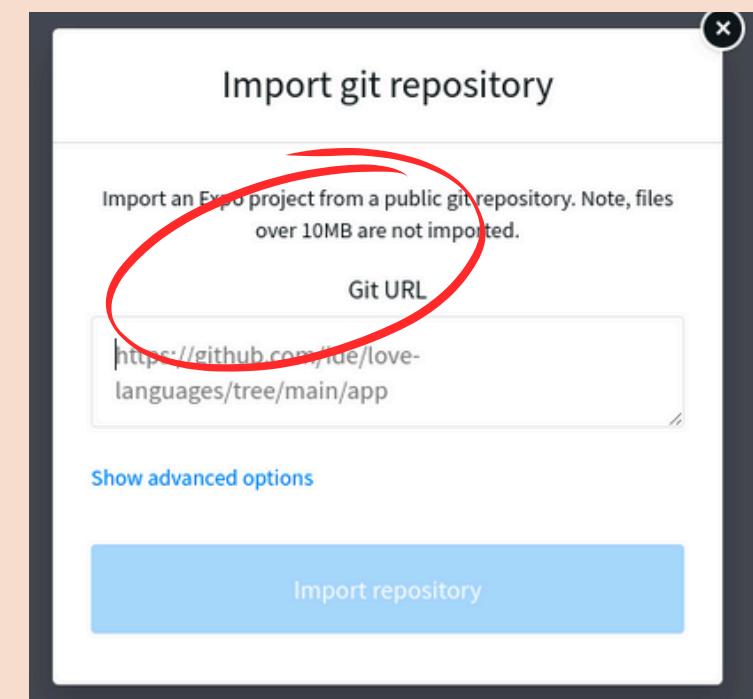
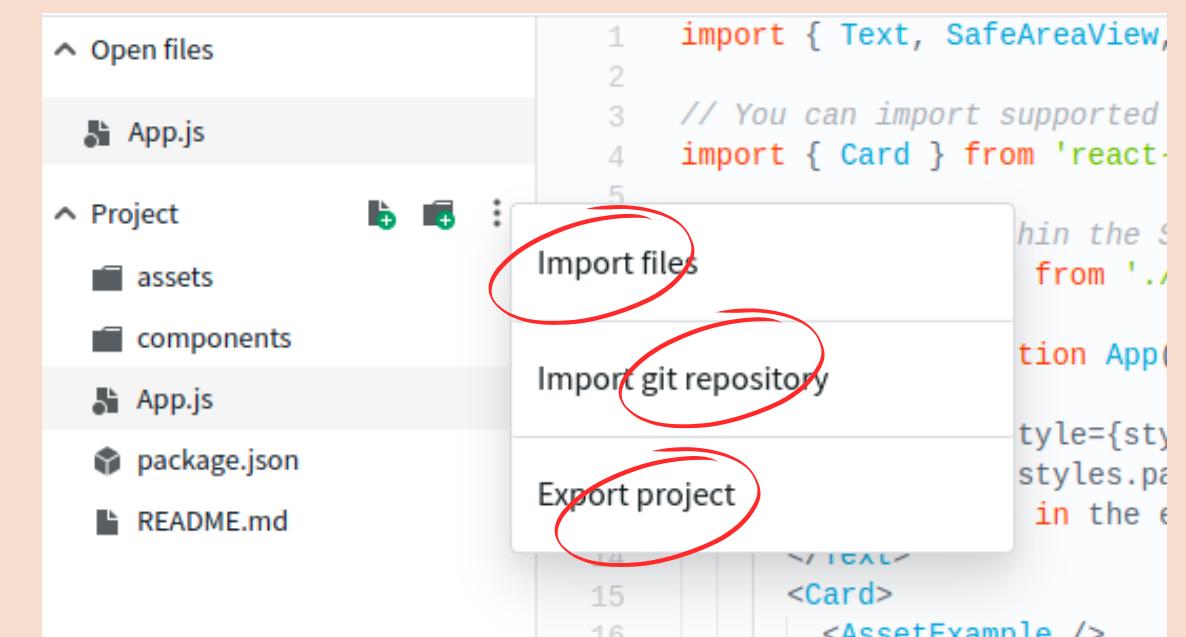


# NPX ET CREATE EXPO APP

## L'OUTILS SNACK EXPO



The screenshot shows the Snack Expo web editor. At the top, there are buttons for "Save", "Share", "Download", and "Edit". Below is a preview window titled "Embed Preview" with tabs for "My Device", "Android", "iOS", and "Web" (which is selected). The preview shows a component named "laughing violet churros". The code editor contains a snippet of React Native code. To the right, there's a sidebar with instructions: "Change code in the editor and watch it change on your phone saikou ! Save to get a shareable url." and "Local files and assets can be imported by dragging and dropping them into the editor". At the bottom, there's an "Embed Code" section with a copy-to-clipboard button.



# NPX ET CREATE EXPO APP

## L'OUTILS SNACK EXPO

```
test-module-externe
Last saved half a minute ago.

Open files
App.js
package.json

Project
```

```
dependencies: {
  "react-native-paper": "4.9.2",
  "@expo/vector-icons": "^14.0.0"
  "axios": "^0.24.0"
}
```

```
test-module-externe
All changes saved less than 5 seconds ago.

Open files
App.js
package.json

Project
```

```
import React, { useEffect, useState } from 'react';
import { Text, View } from 'react-native';
import axios from 'axios';

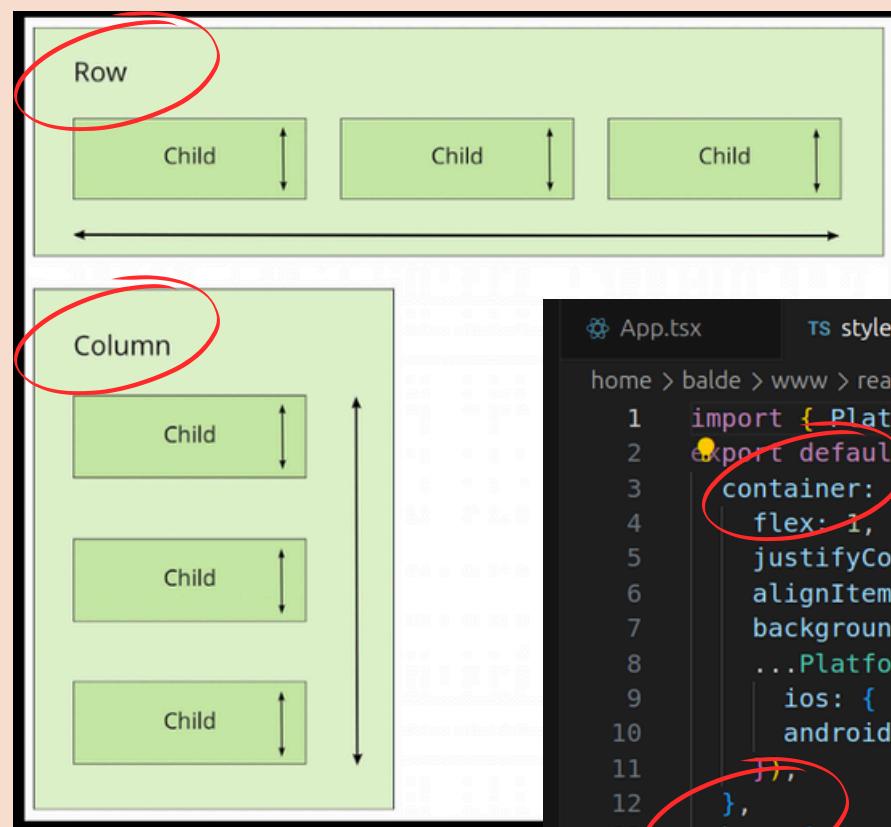
export default function App() {
  const [data, setData] = useState(null);

  useEffect(() => {
    axios.get('https://jsonplaceholder.typicode.com/posts/1')
      .then(response => {
        setData(response.data);
      })
      .catch(error => {
        console.error(error);
      });
  }, []);

  return [
    <View style={{ padding: 20 }}>
      {data ? (
        <Text>{data.body}</Text>
        // <Text>{data.body}</Text>
      ) : (
        <Text>Loading...</Text>
      )}
    </View>
  ];
}
```

# NPX ET CREATE EXPO APP

## LE STYLE (SIMPLEMENT) DANS REACT NATIVE

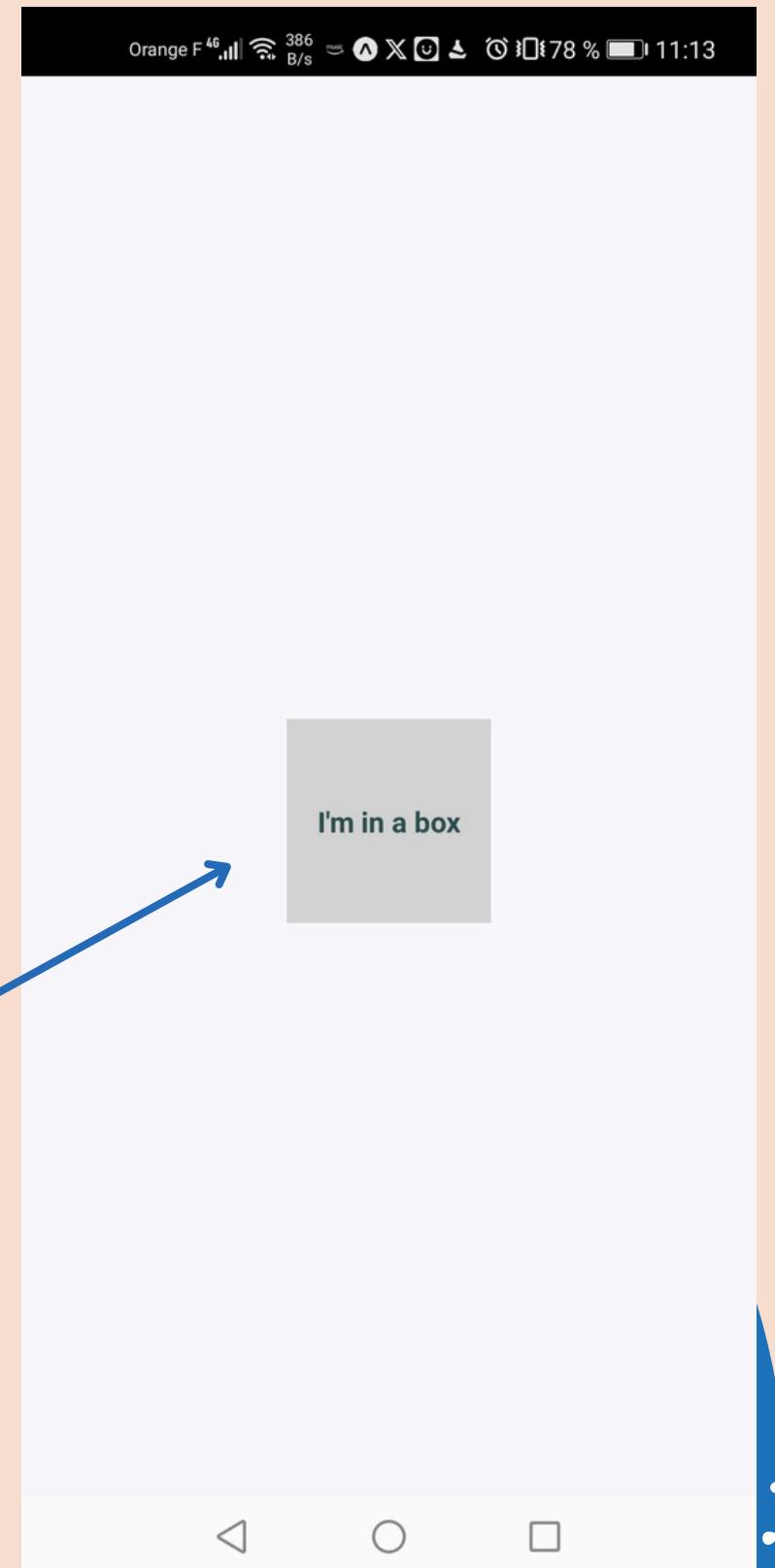


App.tsx

```
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2 export default StyleSheet.create({
3   container: {
4     flex: 1,
5     justifyContent: "center",
6     alignItems: "center",
7     backgroundColor: "ghostwhite",
8     ...Platform.select({
9       ios: { paddingTop: 20 },
10      android: { paddingTop: StatusBar.currentHeight },
11    }),
12  },
13  box: {
14    width: 100,
15    height: 100,
16    justifyContent: "center",
17    alignItems: "center",
18    backgroundColor: "lightgray",
19  },
20  boxText: {
21    color: "darkslategray",
22    fontWeight: "bold",
23  },
24});
```

styles.ts

```
import React from "react";
import { Text, View } from "react-native";
import styles from "./styles";
export default function App() {
  return (
    <View style={styles.container}>
      <View style={styles.box}>
        <Text style={styles.boxText}>I'm in a box</Text>
      </View>
    </View>
  );
}
```



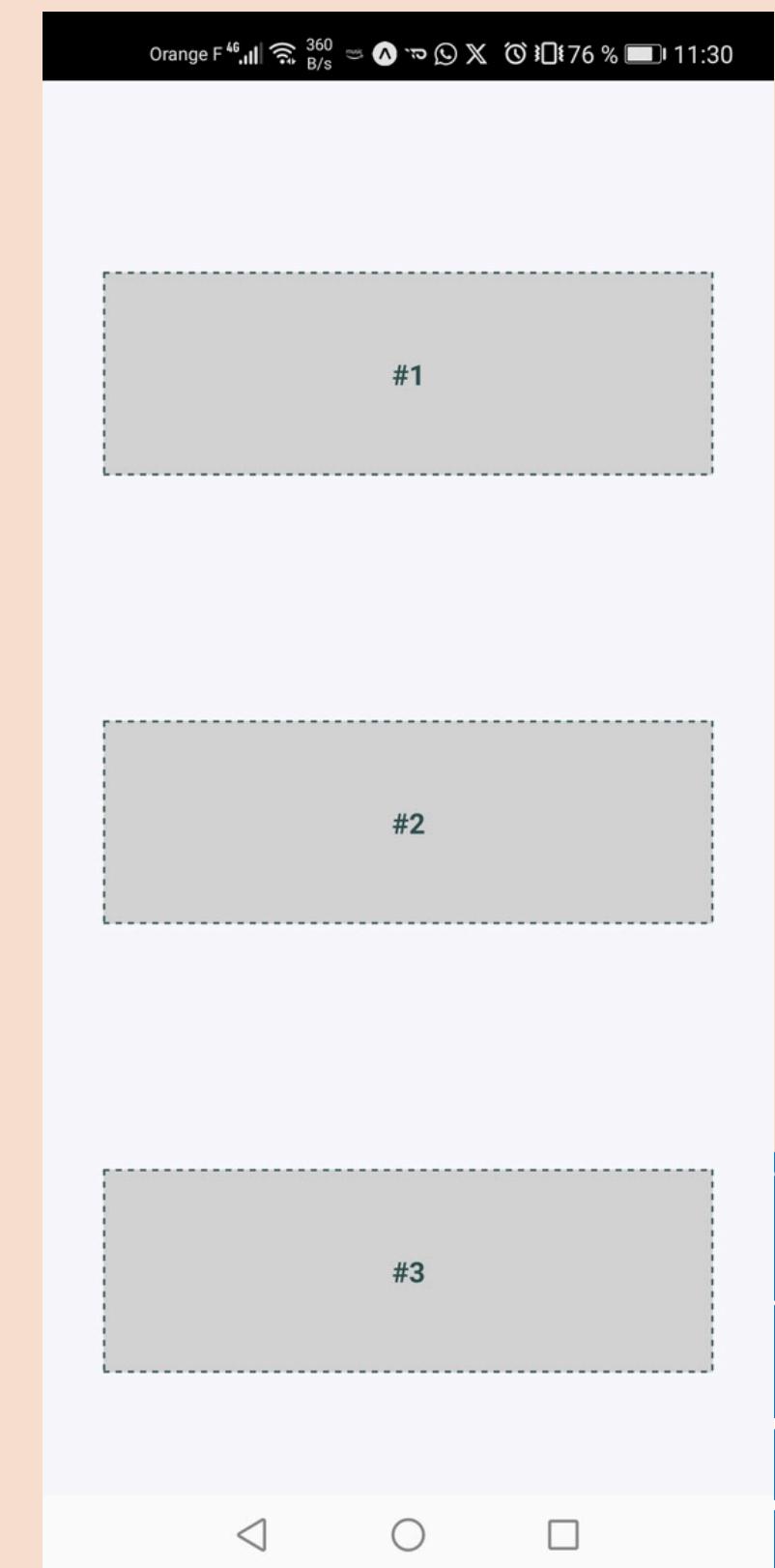
# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### MISE EN PAGE SIMPLE À TROIS COLONNES

```
home > balde > www > react_native > expo-template-app > App.tsx > App
1 import React from "react";
2 import { Text, View } from "react-native";
3 import styles from "./styles";
4 export default function App() {
5   return [
6     <View style={styles.container}>
7       <View style={styles.box}>
8         <Text style={styles.boxText}>#1</Text>
9       </View>
10      <View style={styles.box}>
11        <Text style={styles.boxText}>#2</Text>
12      </View>
13      <View style={styles.box}>
14        <Text style={styles.boxText}>#3</Text>
15      </View>
16    </View>
17  ];
18}
```

```
home > balde > www > react_native > expo-template-app > styles.ts > default > box
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2 export default StyleSheet.create({
3   container: {
4     flex: 1,
5     flexDirection: "column",
6     alignItems: "center",
7     justifyContent: "space-around",
8     backgroundColor: "ghostwhite",
9     ...Platform.select({
10       ios: { paddingTop: 20 },
11       android: { paddingTop: StatusBar.currentHeight }
12     })
13   },
14   box: {
15     width: 300,
16     height: 100,
17     justifyContent: "center",
18     alignItems: "center",
19     backgroundColor: "lightgray",
20     borderwidth: 1,
21     borderStyle: "dashed",
22     borderColor: "darkslategray"
23   },
24   boxText: {
25     color: "darkslategray",
26     fontWeight: "bold"
27   }
28});
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

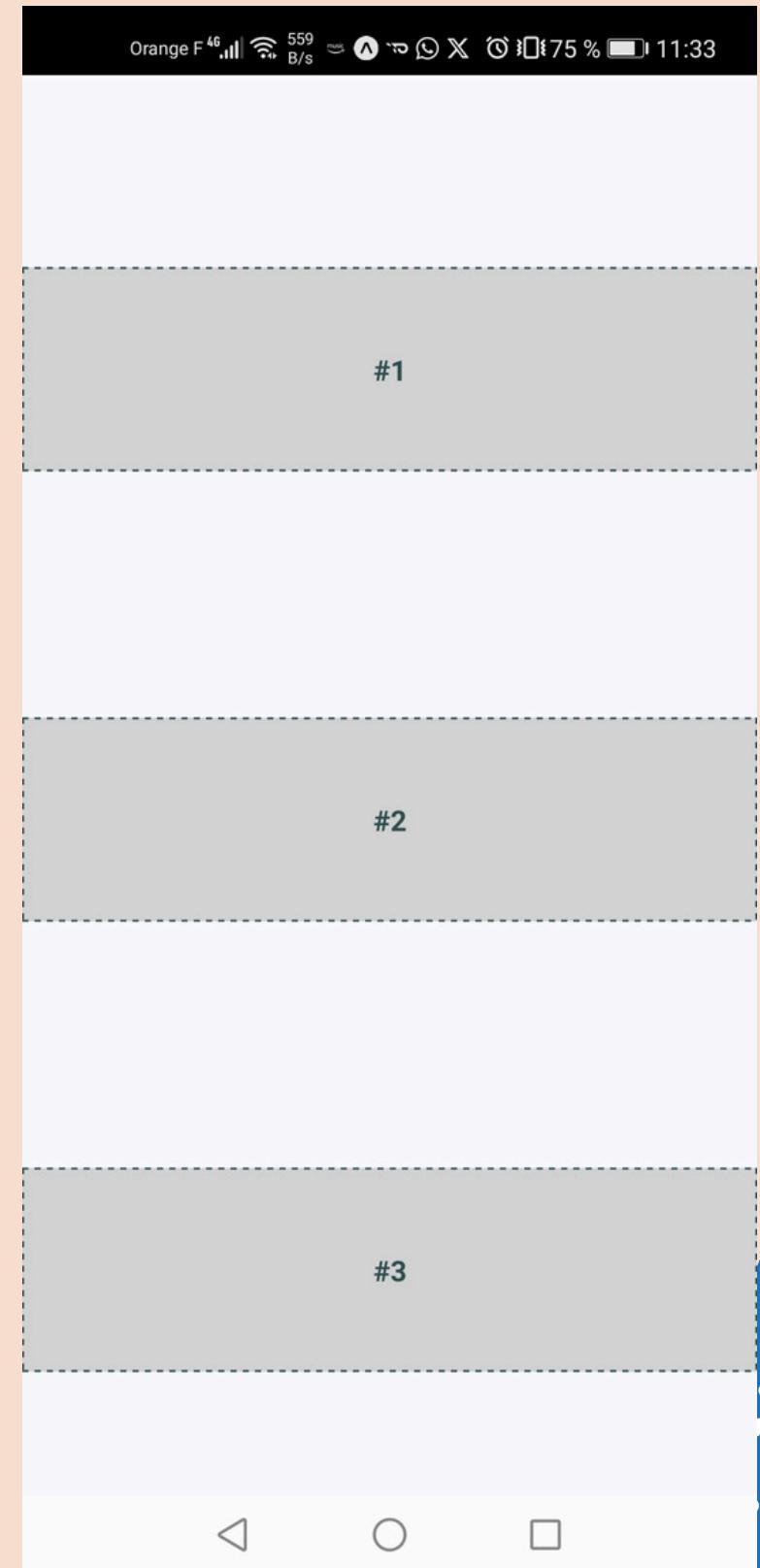
AMELIORER LA MISE EN PAGE SIMPLE À TROIS COLONNES

```
home > balde > www > react_native > expo-template-app > ts styles.ts > ..  
1 import { Platform, StyleSheet, StatusBar } from "react-native";  
2 export default StyleSheet.create({  
3   container: {  
4     flex: 1,  
5     flexDirection: "column",  
6     backgroundColor: "ghostwhite",  
7     justifyContent: "space-around",  
8     ...Platform.select({  
9       ios: { paddingTop: 20 },  
10      android: { paddingTop: StatusBar.currentHeight },  
11    }),  
12  },  
13  box: {  
14    height: 100,  
15    justifyContent: "center",  
16    alignSelf: "stretch",  
17    alignItems: "center",  
18    backgroundColor: "lightgray",  
19    borderWidth: 1,  
20    borderStyle: "dashed",  
21    borderColor: "darkslategray",  
22  },  
23  boxText: {  
24    color: "darkslategray",  
25    fontWeight: "bold",  
26  },  
27}  
28};
```

Le changement clé ici est la propriété **alignSelf**. Cela indique aux éléments avec le style box de modifier leur largeur ou leur hauteur (en fonction du flexDirection de leur conteneur) pour remplir l'espace. De plus, le style de boîte ne définit plus de propriété de largeur car celle-ci sera désormais calculée à la volée.

**justifyContent** contrôle l'alignement des éléments le long de l'axe principal (horizontal ou vertical selon flexDirection).

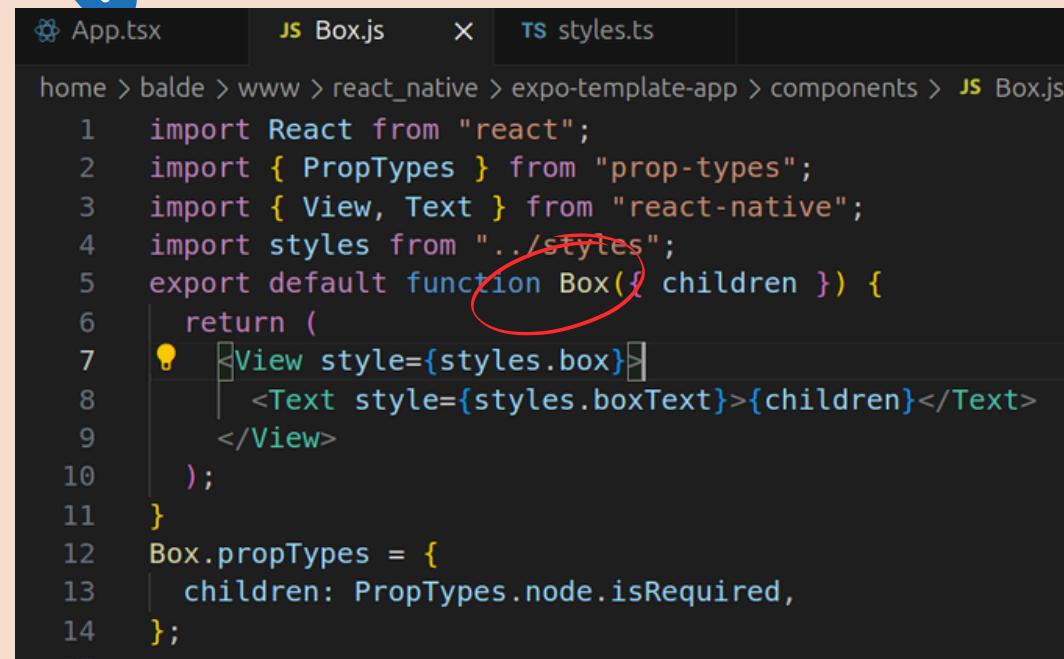
**alignItems** contrôle l'alignement des éléments le long de l'axe secondaire (perpendiculaire à l'axe principal).



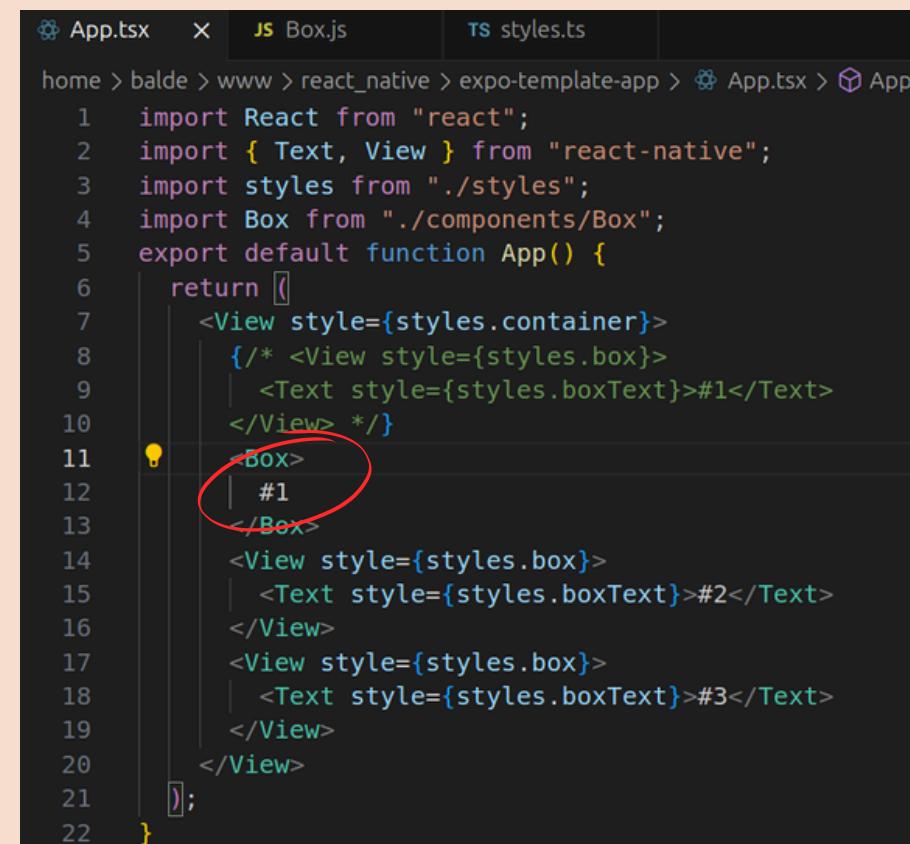
# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

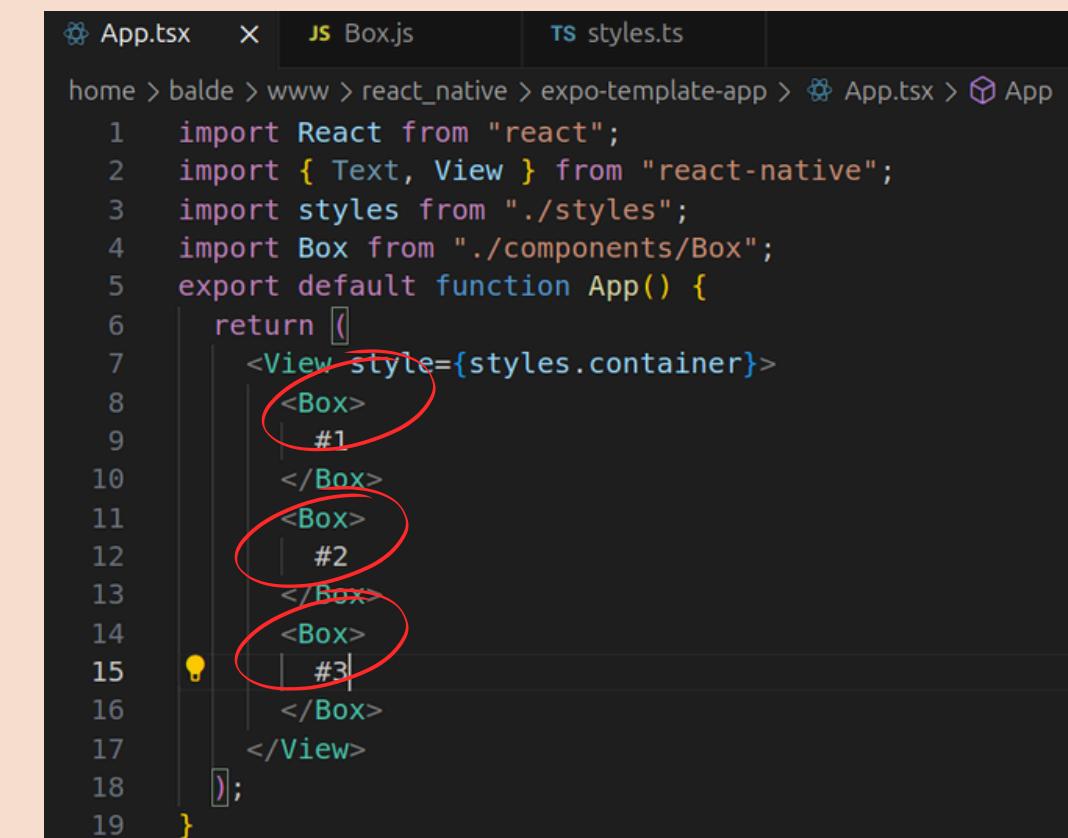
MISE EN PAGE SIMPLE À TROIS COLONNES => OPTIMISATION DU CODE



```
App.tsx JS Box.js X ts styles.ts
home > balde > www > react_native > expo-template-app > components > JS Box.js
1 import React from "react";
2 import { PropTypes } from "prop-types";
3 import { View, Text } from "react-native";
4 import styles from "../styles";
5 export default function Box({ children }) {
6   return (
7     <View style={styles.box}>
8       <Text style={styles.boxText}>{children}</Text>
9     </View>
10  );
11}
Box.propTypes = {
12  children: PropTypes.node.isRequired,
13};
14;
```



```
App.tsx X JS Box.js ts styles.ts
home > balde > www > react_native > expo-template-app > App.tsx > App
1 import React from "react";
2 import { Text, View } from "react-native";
3 import styles from "./styles";
4 import Box from "./components/Box";
5 export default function App() {
6   return [
7     <View style={styles.container}>
8       /* <View style={styles.box}>
9         <Text style={styles.boxText}>#1</Text>
10        </View> */
11      <Box>
12        #1
13      </Box>
14      <View style={styles.box}>
15        <Text style={styles.boxText}>#2</Text>
16      </View>
17      <View style={styles.box}>
18        <Text style={styles.boxText}>#3</Text>
19      </View>
20    </View>
21  ];
22}
```



```
App.tsx X JS Box.js ts styles.ts
home > balde > www > react_native > expo-template-app > App.tsx > App
1 import React from "react";
2 import { Text, View } from "react-native";
3 import styles from "./styles";
4 import Box from "./components/Box";
5 export default function App() {
6   return [
7     <View style={styles.container}>
8       <Box>
9         #1
10      </Box>
11      <Box>
12        #2
13      </Box>
14      <Box>
15        #3
16      </Box>
17    </View>
18  ];
19}
```

# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

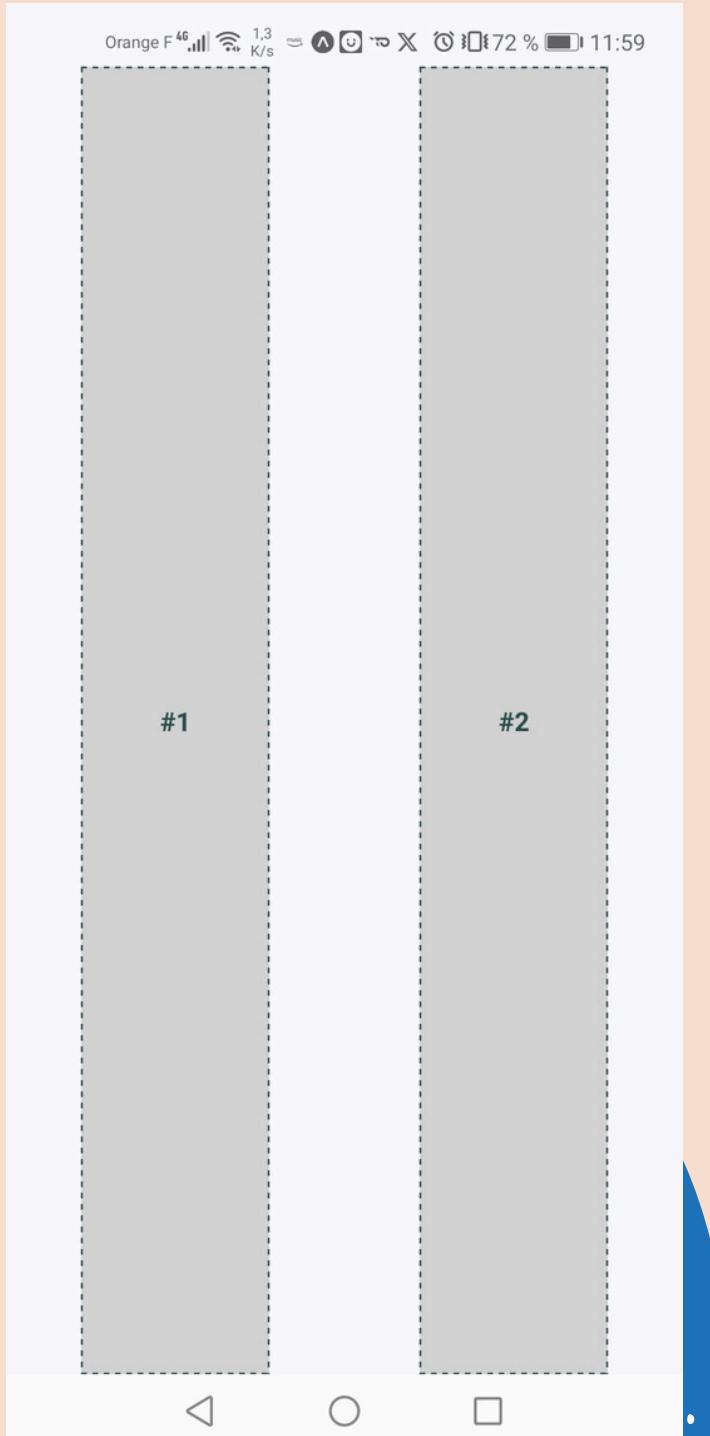
### LIGNES FLEXIBLES : ROWS

```
App.tsx    ts styles.ts  package.json
home > balde > www > react_native > expo-template-app > ts styles.ts > ...
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2 export default StyleSheet.create({
3   container: {
4     flex: 1,
5     flexDirection: "column",
6     backgroundColor: "ghostwhite",
7     justifyContent: "space-around",
8     ...Platform.select({
9       ios: { paddingTop: 20 },
10      android: { paddingTop: StatusBar.currentHeight },
11    }),
12  },
13  box: {
14   height: 100,
15   justifyContent: "center",
16   alignSelf: "stretch",
17   alignItems: "center",
18   backgroundColor: "lightgray",
19   borderwidth: 1,
20   borderStyle: "dashed",
21   borderColor: "darkslategray",
22 },
23  boxText: {
24   color: "darkslategray",
25   fontWeight: "bold",
26 },
27 });
28 })
```

```
ts styles.ts
home > balde > www > react_native > expo-template-app > ts styles.ts > ...
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2 export default StyleSheet.create({
3   container: {
4     flex: 1,
5     flexDirection: "row",
6     backgroundColor: "ghostwhite",
7     alignItems: "center",
8     justifyContent: "space-around",
9     ...Platform.select({
10      ios: { paddingTop: 20 },
11      android: { paddingTop: StatusBar.currentHeight },
12    }),
13  },
14  box: [
15   width: 100,
16   justifyContent: "center",
17   alignSelf: "stretch",
18   alignItems: "center",
19   backgroundColor: "lightgray",
20   borderwidth: 1,
21   borderStyle: "dashed",
22   borderColor: "darkslategray",
23 ],
24  boxText: {
25   color: "darkslategray",
26   fontWeight: "bold",
27 },
28 })
```

```
App.tsx
1 import React from "react";
2 import { View, StatusBar } from "react-native";
3 import styles from "./styles";
4 import Box from "./components/Box";
5 const boxes = new Array(10).fill(null).map((v, i) => i + 1);
6 export default function App() {
7   return (
8     <View style={styles.container}>
9       <StatusBar hidden={false} />
10      {boxes.map((i) => (
11        <Box key={i}>#{i}</Box>
12      ))}
13     </View>
14   );
15 }
```

```
App.tsx
1 import React from "react";
2 import { Text, View } from "react-native";
3 import styles from "./styles";
4 import Box from "./components/Box";
5 export default function App() {
6   return (
7     <View style={styles.container}>
8       <Box>
9         #1
10       </Box>
11       <Box>
12         #2
13       </Box>
14     </View>
15   );
16 }
```

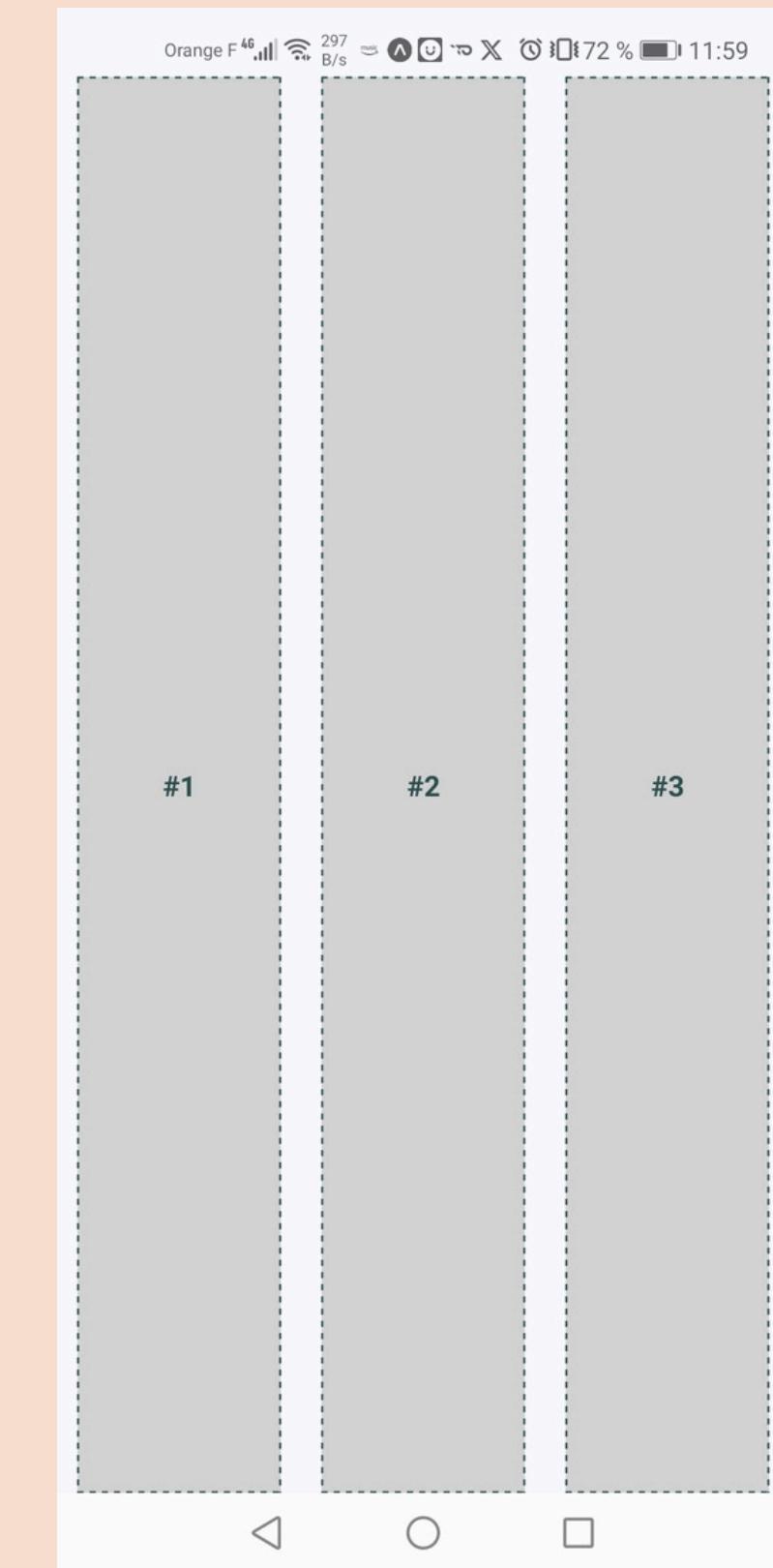


# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

LIGNES FLEXIBLES : ROWS

```
home > balde > www > react_native > expo-template-app > App.tsx >
1  import React from "react";
2  import { Text, View } from "react-native";
3  import styles from "./styles";
4  import Box from "./components/Box";
5  export default function App() {
6    return [
7      <View style={styles.container}>
8        <Box>
9          #1
10         </Box>
11        <Box>
12          #2
13         </Box>
14        <Box>
15          #3
16         </Box>
17       </View>
18    ];
19 }
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### GRILLES FLEXIBLES

```
home > balde > www > react_native > expo-template-app > ts styles.ts > [o] default > box
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2 export default StyleSheet.create({
3   container: {
4     flex: 1,
5     flexDirection: "row",
6     backgroundColor: "ghostwhite",
7     alignItems: "center",
8     justifyContent: "space-around",
9     ...Platform.select({
10       ios: { paddingTop: 20 },
11       android: { paddingTop: StatusBar.currentHeight },
12     }),
13   },
14   box: [
15     width: 100,
16     justifyContent: "center",
17     alignSelf: "stretch",
18     alignItems: "center",
19     backgroundColor: "lightgray",
20     borderWidth: 1,
21     borderStyle: "dashed",
22     borderColor: "darkslategray",
23   ],
24   boxText: {
25     color: "darkslategray",
26     fontWeight: "bold",
27   },
28 });

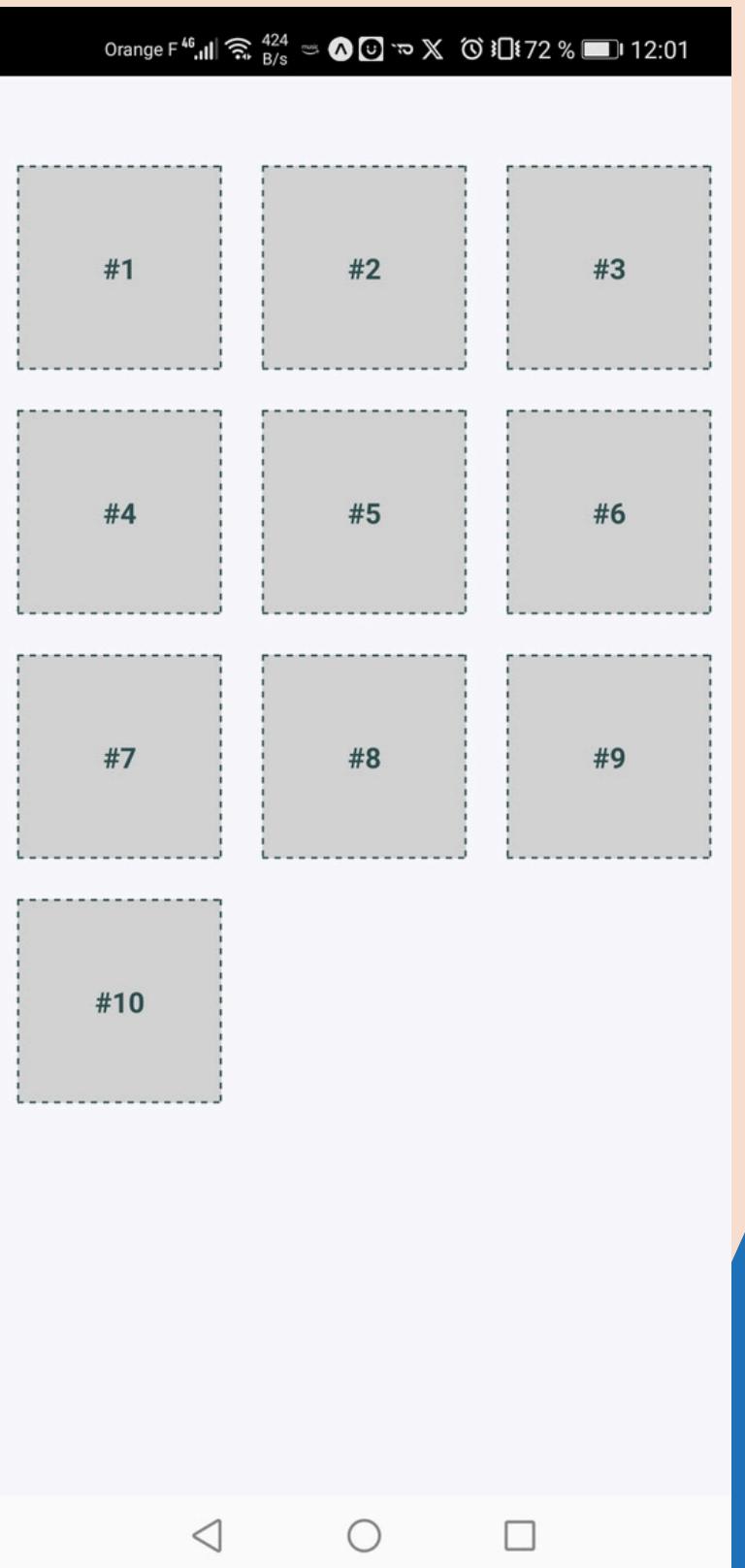
#1
```

```
home > balde > www > react_native > expo-template-app > ts styles.ts > [o] default
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     flexDirection: "row",
7     flexWrap: "wrap",
8     backgroundColor: "ghostwhite",
9     alignItems: "center",
10    ...Platform.select({
11      ios: { paddingTop: 40 },
12      android: { paddingTop: StatusBar.currentHeight },
13    }),
14  },
15
16  box: {
17    height: 100,
18    width: 100,
19    justifyContent: "center",
20    alignItems: "center",
21    backgroundColor: "lightgray",
22    borderWidth: 1,
23    borderStyle: "dashed",
24    borderColor: "darkslategray",
25    margin: 10,
26  },
27
28  boxText: {
29    color: "darkslategray",
30    fontWeight: "bold",
31  },
32 });

#2 #3
```

```
home > balde > www > react_native > expo-template-app > App.tsx > ...
1 import React from "react";
2 import { View, StatusBar } from "react-native";
3 import styles from "./styles";
4 import Box from "./components/Box";
5 const boxes = new Array(10).fill(null).map((v, i) => i + 1);
6 export default function App() {
7   return (
8     <View style={styles.container}>
9       <StatusBar hidden={false} />
10      {boxes.map((i) => (
11        <Box key={i}>#{i}</Box>
12      ))}
13     </View>
14   );
15 }

#4 #5 #6
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

LIGNES ET COLONNES FLEXIBLES - LE MIXTE

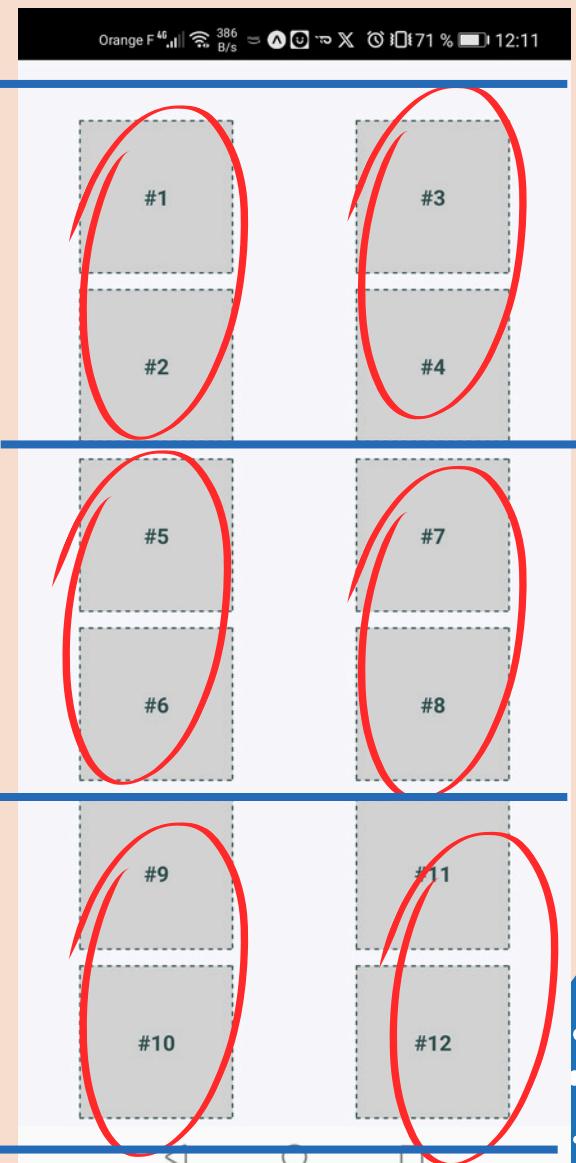
```
ts styles.ts
home > balde > www > react_native > expo-template-app > ts styles.ts > default > container > <unknown>
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     flexDirection: "column",
7     backgroundColor: "ghostwhite",
8     alignItems: "center",
9     justifyContent: "space-around",
10    ...Platform.select({
11      ios: { paddingTop: 40 },
12      android: { paddingTop: StatusBar.currentHeight },
13    }),
14  },
15
16  box: {
17    height: 100,
18    width: 100,
19    justifyContent: "center",
20    alignItems: "center",
21    borderwidth: 1,
22    borderStyle: "dashed",
23    borderColor: "darkslategray",
24    backgroundColor: "lightgray",
25  },
26
27  boxText: {
28    color: "darkslategray",
29    fontWeight: "bold",
30  },
31
32  row: {
33    flex: 1,
34    flexDirection: "row",
35    justifyContent: "space-around",
36    alignSelf: "stretch",
37  },
38
39  column: {
40    flex: 1,
41    flexDirection: "column",
42    alignItems: "center",
43    justifyContent: "space-around",
44    alignSelf: "stretch",
45  },
46});
```

```
home > balde > www > react_native > expo-template-app > components > Row.js
1 import React from "react";
2 import PropTypes from "prop-types";
3 import { View } from "react-native";
4 import styles from "../styles";
5
6 type Props = {
7   children: React.ReactNode;
8 };
9
10 export default function Row({ children }: Props) {
11   return <View style={styles.row}>{children}</View>;
12 }
13
14 Row.propTypes = {
15   children: PropTypes.node.isRequired,
16 };

home > balde > www > react_native > expo-template-app > components > Column.js
1 import React from "react";
2 import PropTypes from "prop-types";
3 import { View } from "react-native";
4 import styles from "../styles";
5
6 type Props = {
7   children: React.ReactNode;
8 };
9
10 export default function Column({ children }: Props) {
11   return <View style={styles.column}>{children}</View>;
12 }
13
14 Column.propTypes = {
15   children: PropTypes.node.isRequired,
16 };
```

```
home > balde > www > react_native > expo-template-app > components > Box.js
1 import React from "react";
2 import { PropTypes } from "prop-types";
3 import { View, Text } from "react-native";
4 import styles from "../styles";
5
6 export default function Box({ children }) {
7   return (
8     <View style={styles.box}>
9       <Text style={styles.boxText}>{children}</Text>
10    </View>
11  );
12
13 Box.propTypes = {
14   children: PropTypes.node.isRequired,
15 };
```

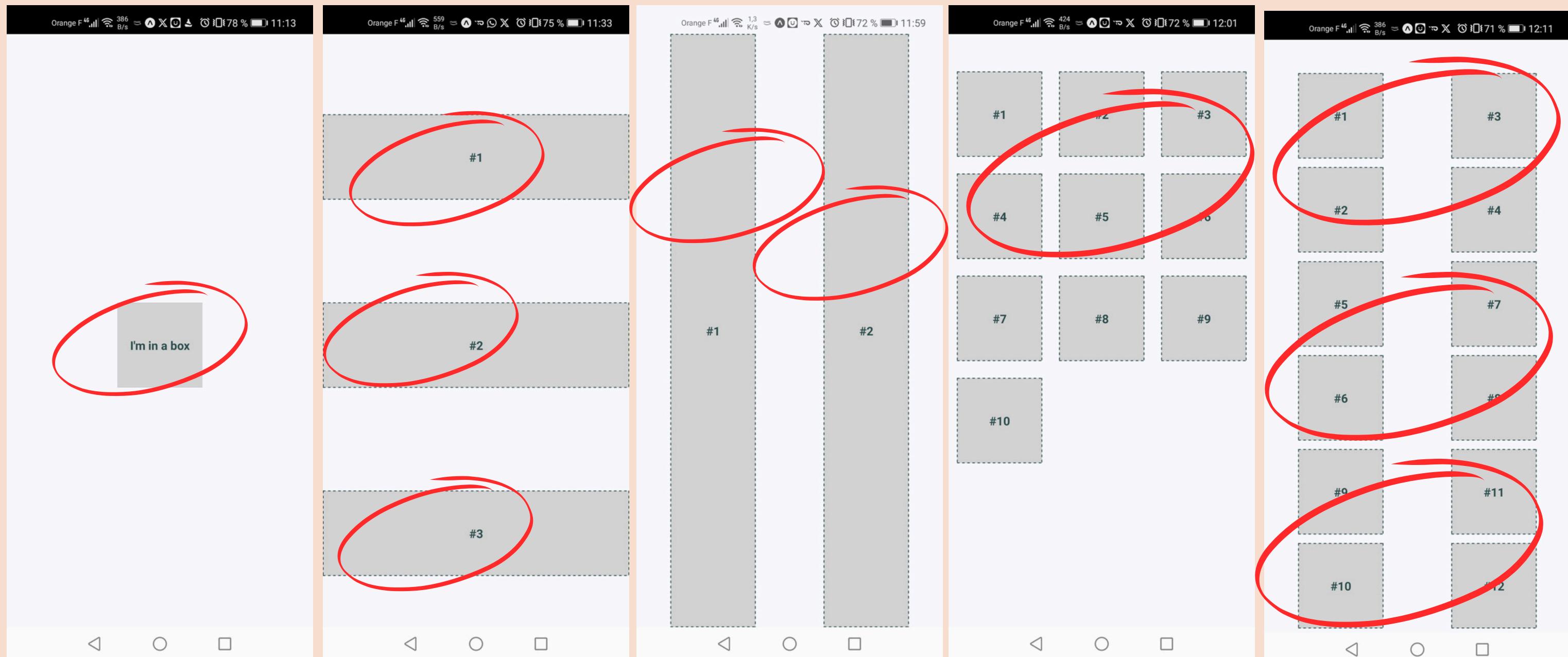
```
home > balde > www > react_native > expo-template-app > App.tsx
1 import React from "react";
2 import { StatusBar } from "react-native";
3 import styles from "./styles";
4 import Row from "./components/Row";
5 import Column from "./components/Column";
6 import Box from "./components/Box";
7
8 export default function App() {
9   return [
10     <View style={styles.container}>
11       <StatusBar hidden={false} />
12       <Row>
13         <Column>
14           <Box>#1</Box>
15           <Box>#2</Box>
16         </Column>
17         <Column>
18           <Box>#3</Box>
19           <Box>#4</Box>
20         </Column>
21       </Row>
22       <Row>
23         <Column>
24           <Box>#5</Box>
25           <Box>#6</Box>
26         </Column>
27         <Column>
28           <Box>#7</Box>
29           <Box>#8</Box>
30         </Column>
31       </Row>
32       <Row>
33         <Column>
34           <Box>#9</Box>
35           <Box>#10</Box>
36         </Column>
37         <Column>
38           <Box>#11</Box>
39           <Box>#12</Box>
40         </Column>
41       </Row>
42     </View>
43   ];
44 }
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

LAYOUT PRÉSENTATION SYNTHÈSE



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### NAVIGATION ENTRE LES ÉCRANS

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npm install @react-navigation/native  
up to date, audited 1211 packages in 3s  
132 packages are looking for funding  
  run `npm fund` for details  
found 0 vulnerabilities
```

react-navigation/native est un module clé de la bibliothèque React Navigation, qui est largement utilisée pour la navigation dans les applications React Native. Il fournit les primitives nécessaires pour créer et gérer la navigation dans une application mobile créée avec React Native

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npx expo install react-native-screens react-native-safe-area-context  
> Installing 2 SDK 51.0.0 compatible native modules using npm  
> npm install  
up to date, audited 1211 packages in 3s  
132 packages are looking for funding  
  run `npm fund` for details  
found 0 vulnerabilities
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npm install @react-navigation/native-stack  
up to date, audited 1211 packages in 3s  
132 packages are looking for funding  
  run `npm fund` for details  
found 0 vulnerabilities
```

```
home > balde > www > react_native > expo-template-app > TS router.ts > [e] RootStackParamList  
1  export type RootStackParamList = {  
2    Home: undefined;  
3    Settings: undefined;  
4  };|
```

dépendances à react-navigation/native et react-navigation/native-stack

react-navigation/native-stack est un module qui fournit un navigateur basé sur des piles pour React Navigation en utilisant les composants natifs pour une performance améliorée. Il est basé sur le package react-native-screens pour utiliser les transitions et les animations de navigation natives.

# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### NAVIGATION ENTRE LES ÉCRANS

```
home > balde > www > react_native > expo-template-app > components > Settings.tsx > ...  
1 import { View, Text, Button, StatusBar } from "react-native";  
2 import styles from "./styles";  
3  
4 import { NativeStackScreenProps } from "@react-navigation/native-stack";  
5 import { RootStackParamList } from "../router";  
6  
7 type Props = NativeStackScreenProps<RootStackParamList>;  
8 export default function Settings({ navigation }: Props) {  
9     return (  
10         <View style={styles.container}>  
11             <StatusBar barStyle="dark-content" />  
12             <Text>Settings Screen</Text>  
13             <Button title="Home" onPress={() => navigation.navigate("Home")}></Button>  
14         </View>  
15     );  
16 }
```

Permet de définir les types pour les propriétés de navigation et les routes, facilitant ainsi la gestion des navigateurs de type stack natif dans React Native

```
home > balde > www > react_native > expo-template-app > components > Home.tsx > ...  
1 import React from "react";  
2 import { View, Text, Button, StatusBar } from "react-native";  
3 import styles from "./styles";  
4 import { NativeStackScreenProps } from "@react-navigation/native-stack";  
5 import { RootStackParamList } from "../router";  
6  
7 type Props = NativeStackScreenProps<RootStackParamList>;  
8 export default function Home({ navigation }: Props) {  
9     return (  
10         <View style={styles.container}>  
11             <StatusBar barStyle="dark-content" />  
12             <Text>Home Screen</Text>  
13             <Button  
14                 title="Settings"  
15                 onPress={() => navigation.navigate("Settings")}></Button>  
16         </View>  
17     );  
18 }
```

gère l'état de la navigation de votre application et doit envelopper vos navigateurs.

```
home > balde > www > react_native > expo-template-app > router.ts > ...  
1 export type RootStackParamList = {  
2     Home: undefined;  
3     Settings: undefined;  
4 };
```

```
home > balde > www > react_native > expo-template-app > ...  
1 import * as React from "react";  
2 import { NavigationContainer } from "@react-navigation/native";  
3 import { createNativeStackNavigator } from "@react-navigation/native-stack";  
4 import Home from "./components/Home";  
5 import Settings from "./components/Settings";  
6 import { RootStackParamList } from "./router";  
7  
8 const Stack = createNativeStackNavigator<RootStackParamList>();  
9 export default function App() {  
10     return (  
11         <NavigationContainer>  
12             <Stack.Navigator>  
13                 <Stack.Screen name="Home" component={Home} />  
14                 <Stack.Screen name="Settings" component={Settings} />  
15             </Stack.Navigator>  
16         </NavigationContainer>  
17     );  
18 }
```

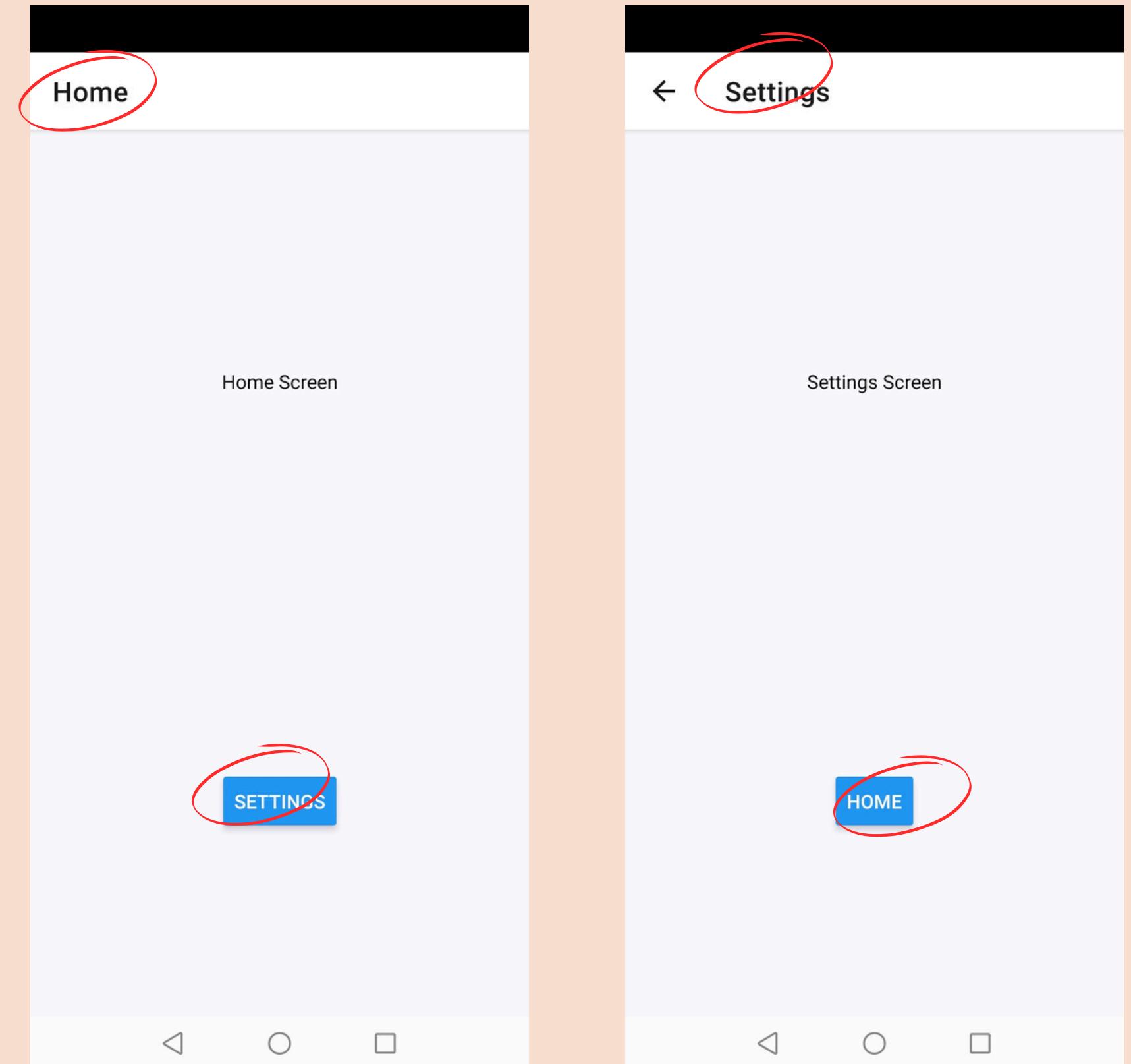
fonction utilisée dans React Navigation pour créer un stack navigator natif. Ce stack navigator est optimisé pour fonctionner de manière performante et intégrée dans des environnements natifs comme iOS et Android.

# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

```
home > balde > www > react_native > expo-template-app > ts styles.ts > default > container
1 import { Platform, StyleSheet, StatusBar } from "react-native";
2
3 export default StyleSheet.create({
4   container: [
5     flex: 1,
6     flexDirection: "column",
7     backgroundColor: "ghostwhite",
8     alignItems: "center",
9     justifyContent: "space-around",
10    ...Platform.select({
11      ios: { paddingTop: 40 },
12      android: { paddingTop: StatusBar.currentHeight },
13    }),
14  ],
15
16   box: {
17     height: 100,
18     width: 100,
19     justifyContent: "center",
20     alignItems: "center",
21     borderwidth: 1,
22     borderStyle: "dashed",
23     borderColor: "darkslategray",
24     backgroundColor: "lightgray",
25   },
26 })
```

### NAVIGATION ENTRE LES ÉCRANS



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### LES PARAMÈTRES DE ROUTE

```
App.tsx      Home.tsx      Details.tsx      TS router.ts X
home > balde > www > react_native > expo-template-app > TS router.ts > ...
1  export type RootStackParamList = {
2    Home: undefined;
3    Details: { title: string };
4  };
5
```

Ce code définit un alias de type nommé `Props` qui spécifie le type de propriétés accessibles à un composant React dans un écran de navigation Native Stack nommé "Details" et "Home".

```
App.tsx      Home.tsx X      Details.tsx      TS router.ts
home > balde > www > react_native > expo-template-app > components > Home.tsx > Home
1  import React from "react";
2  import { View, Text, Button, StatusBar } from "react-native";
3  import styles from "../styles";
4  import { NativeStackScreenProps } from "@react-navigation/native-stack";
5  import { RootStackParamList } from "../router";

6
7  type Props = NativeStackScreenProps<RootStackParamList, "Home">;
8  export default function Home({ navigation }: Props) {
9    return (
10      <View style={styles.container}>
11        <StatusBar barStyle="dark-content" />
12        <Text>Home Screen</Text>
13        <Button
14          title="First Item"
15          onPress={() => navigation.navigate("Details", { title: "First Item" })}
16        />
17        <Button
18          title="Second Item"
19          onPress={() => navigation.navigate("Details", { title: "Second Item" })}
20        />
21        <Button
22          title="Third Item"
23          onPress={() => navigation.navigate("Details", { title: "Third Item" })}
24        />
25      </View>
26    );
27 }
```

Nom du composant

les paramètres routes

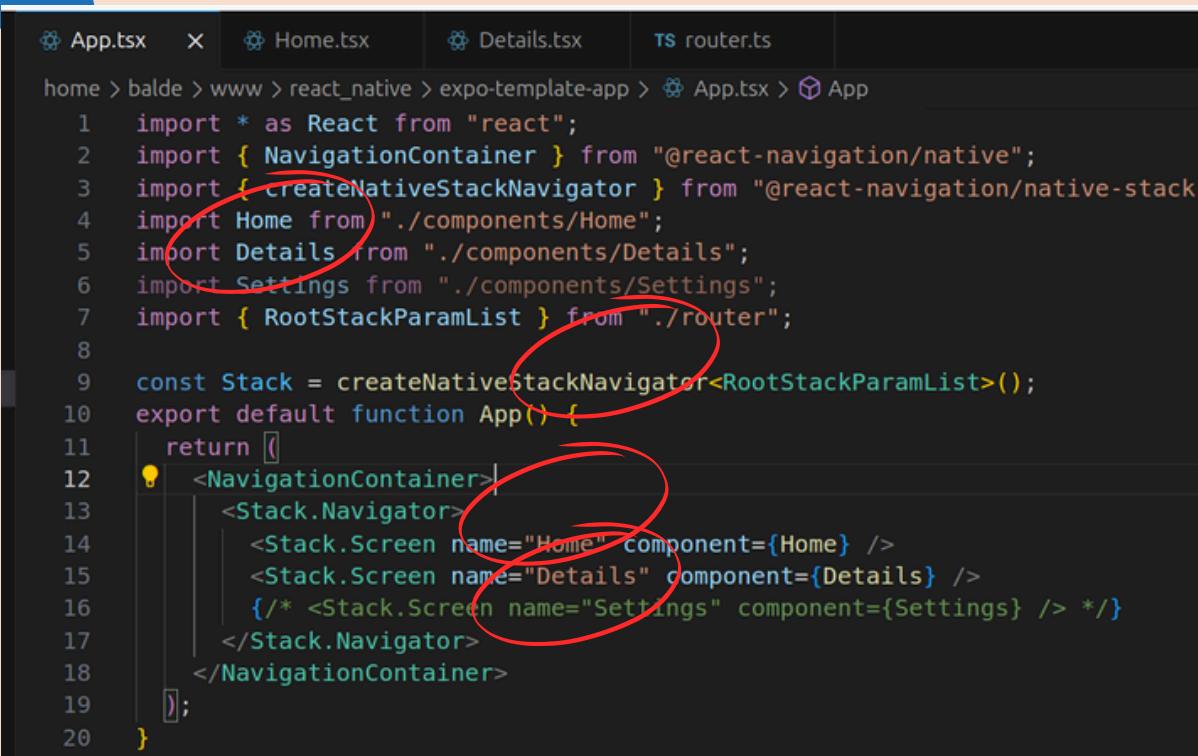
```
App.tsx      Home.tsx      Details.tsx X      TS router.ts
home > balde > www > react_native > expo-template-app > components > Details.tsx > default
1  import { View, Text, StatusBar } from "react-native";
2  import styles from "../styles";
3
4  import { NativeStackScreenProps } from "@react-navigation/native-stack";
5  import { RootStackParamList } from "../router";
6
7  type Props = NativeStackScreenProps<RootStackParamList, "Details">;
8  export default function ({ route }: Props) {
9    const { title } = route.params;
10   return (
11     <View style={styles.container}>
12       <StatusBar barStyle="dark-content" />
13       <Text>{title}</Text>
14     </View>
15   );
16 }
```

Nom du composant

# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### LES PARAMÈTRES DE ROUTE



```
App.tsx  X  Home.tsx  Details.tsx  ts router.ts

home > balde > www > react_native > expo-template-app > App.tsx > App

1 import * as React from "react";
2 import { NavigationContainer } from "@react-navigation/native";
3 import { createNativeStackNavigator } from "@react-navigation/native-stack";
4 import Home from "./components/Home";
5 import Details from "./components/Details";
6 import Settings from "./components/Settings";
7 import { RootStackParamList } from "./router";
8
9 const Stack = createNativeStackNavigator<RootStackParamList>();
10 export default function App() {
11   return [
12     <NavigationContainer>
13       <Stack.Navigator>
14         <Stack.Screen name="Home" component={Home} />
15         <Stack.Screen name="Details" component={Details} />
16         {/* <Stack.Screen name="Settings" component={Settings} /> */}
17       </Stack.Navigator>
18     </NavigationContainer>
19   ];
20 }
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### UTILISATION DE LA BARRE DE NAVIGATION

```
home > balde > www > react_native > expo-template-app > ts router.ts > RoutesParams
1 export type RoutesParams = {
2   Home: undefined;
3   Details: { title: string; stock: number; content: string };
4 };

home > balde > www > react_native > expo-template-app > components > Details.tsx > default > React.use
1 import React from "react";
2 import { View, Text, StatusBar } from "react-native";
3 import styles from "../styles";
4
5 import { NativeStackScreenProps } from "@react-navigation/native-stack";
6 import { RoutesParams } from "../router";
7
8 type Props = NativeStackScreenProps<RoutesParams, "Details">;
9 export default function ({ route, navigation }: Props) {
10   const { content, title, stock } = route.params;
11   React.useEffect(() => [
12     navigation.setOptions({ title }),
13   ], [title]);
14   return (
15     <View style={styles.container}>
16       <StatusBar barStyle="dark-content" />
17       <Text>{content} and stock is {stock}</Text>
18     </View>
19   );
20 }

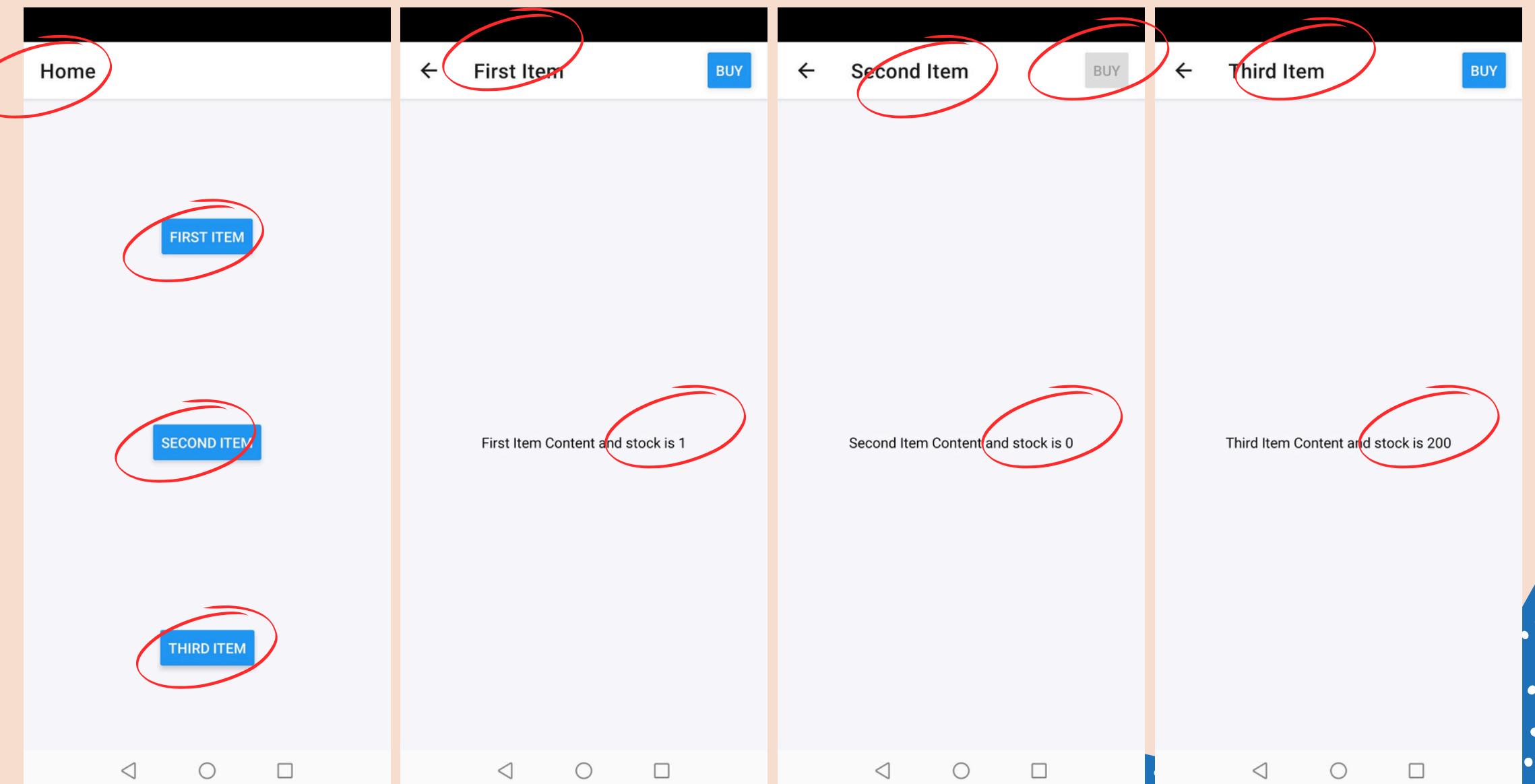
home > balde > www > react_native > expo-template-app > components > Home.tsx > Home > <func
1 import React from "react";
2 import { View, Text, Button, StatusBar } from "react-native";
3 import styles from "../styles";
4 import { NativeStackScreenProps } from "@react-navigation/native-stack";
5 import { RoutesParams } from "../router";
6
7 type Props = NativeStackScreenProps<RoutesParams, "Home">;
8 export default function Home({ navigation }: Props) {
9   return (
10     <View style={styles.container}>
11       <StatusBar barStyle="dark-content" />
12       <Button
13         title="First Item"
14         onPress={() =>
15           navigation.navigate("Details", {
16             title: "First Item",
17             content: "First Item Content",
18             stock: 1,
19           })
20         }
21       >
22         <Button
23           title="Second Item"
24           onPress={() =>
25             navigation.navigate("Details", {
26               title: "Second Item",
27               content: "Second Item Content",
28               stock: 0,
29             })
30           }
31         >
32           <Button
33             title="Third Item"
34             onPress={() =>
35               navigation.navigate("Details", {
36                 title: "Third Item",
37                 content: "Third Item Content",
38                 stock: 200,
39               })
40             }
41           >
42         </View>
43       </Button>
44     </View>
45   );
46 }
```

# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### UTILISATION DE LA BARRE DE NAVIGATION

```
home > balde > www > react_native > expo-template-app > App.tsx > App > <function> > headerRight
1 import * as React from "react";
2 import { NavigationContainer } from "@react-navigation/native";
3 import { createNativeStackNavigator } from "@react-navigation/native-stack";
4 import Home from "./components/Home";
5 import Details from "./components/Details";
// import Settings from "./components/Settings";
7 import { RoutesParams } from "./router";
8 import { Button } from "react-native";
9
10 const Stack = createNativeStackNavigator<RoutesParams>();
11 export default function App() {
12   return (
13     <NavigationContainer>
14       <Stack.Navigator>
15         <Stack.Screen name="Home" component={Home} />
16         <Stack.Screen name="Details" component={Details} options={({ route }) => ({
17           headerRight: () => {
18             return (
19               <Button
20                 title="Buy"
21                 onPress={() => {}}
22                 disabled={route.params.stock === 0}
23               />
24             );
25           },
26         })}
27       />
28       /* <Stack.Screen name="Settings" component={Settings} /> */
29     </Stack.Navigator>
30   </NavigationContainer>
31 );
32 }
```



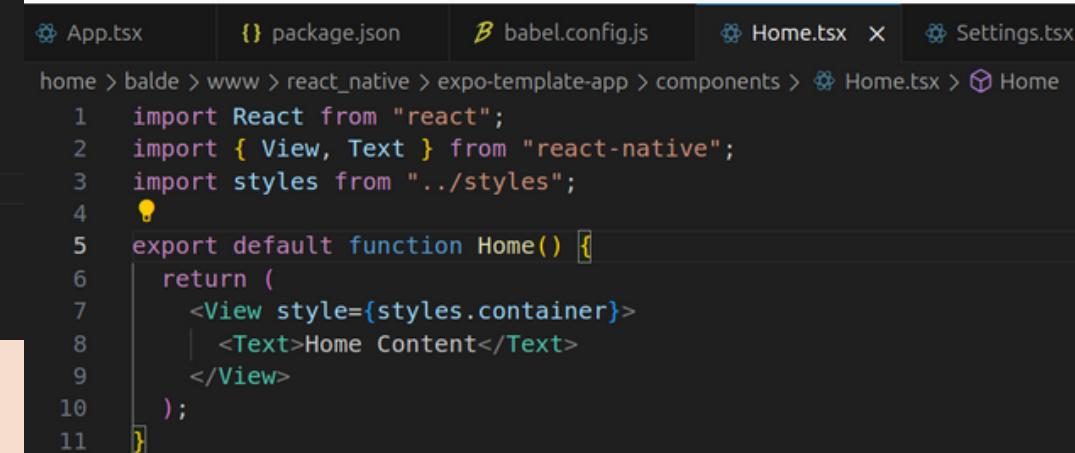
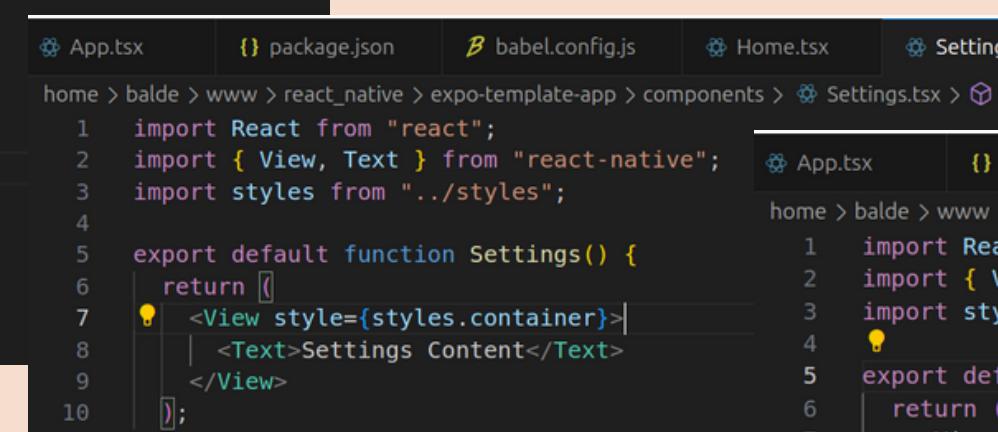
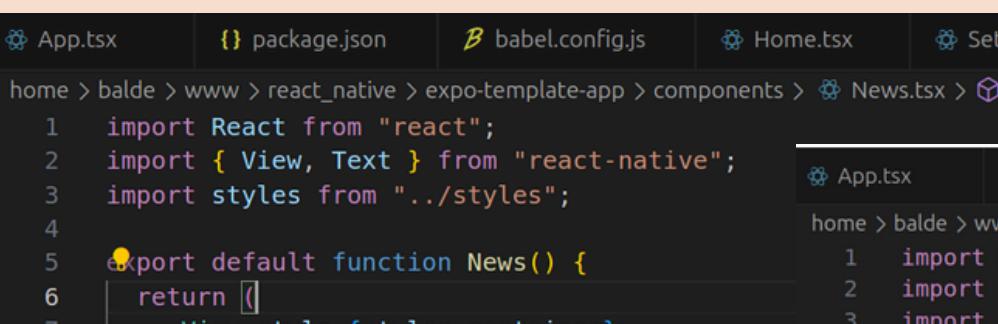
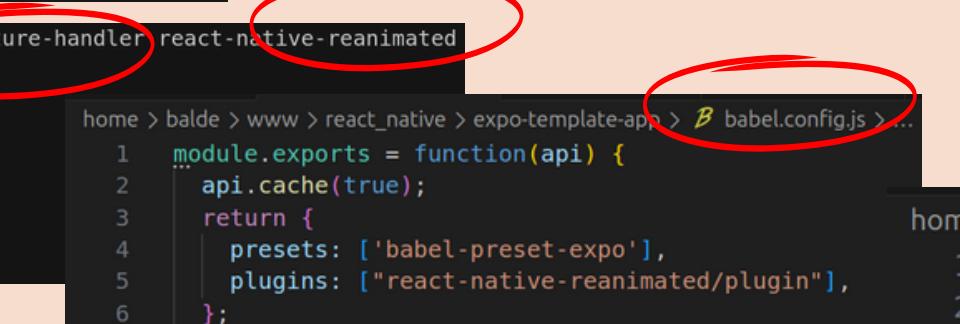
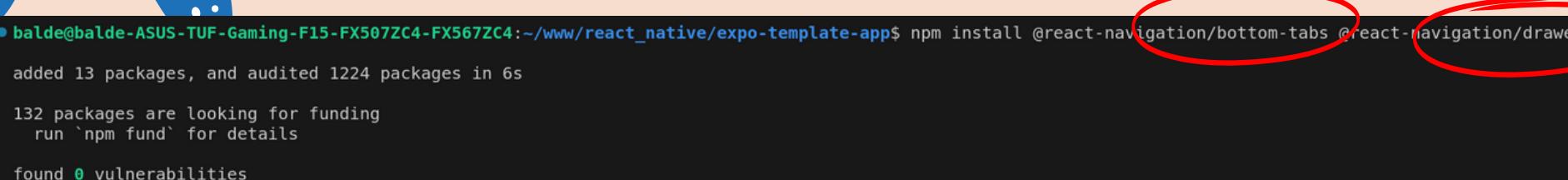
# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

NAVIGATION EN UTILISANT LES ONGLETS ET LES TABS

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npm install @react-navigation/bottom-tabs @react-navigation/drawer
added 13 packages, and audited 1224 packages in 6s
132 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities

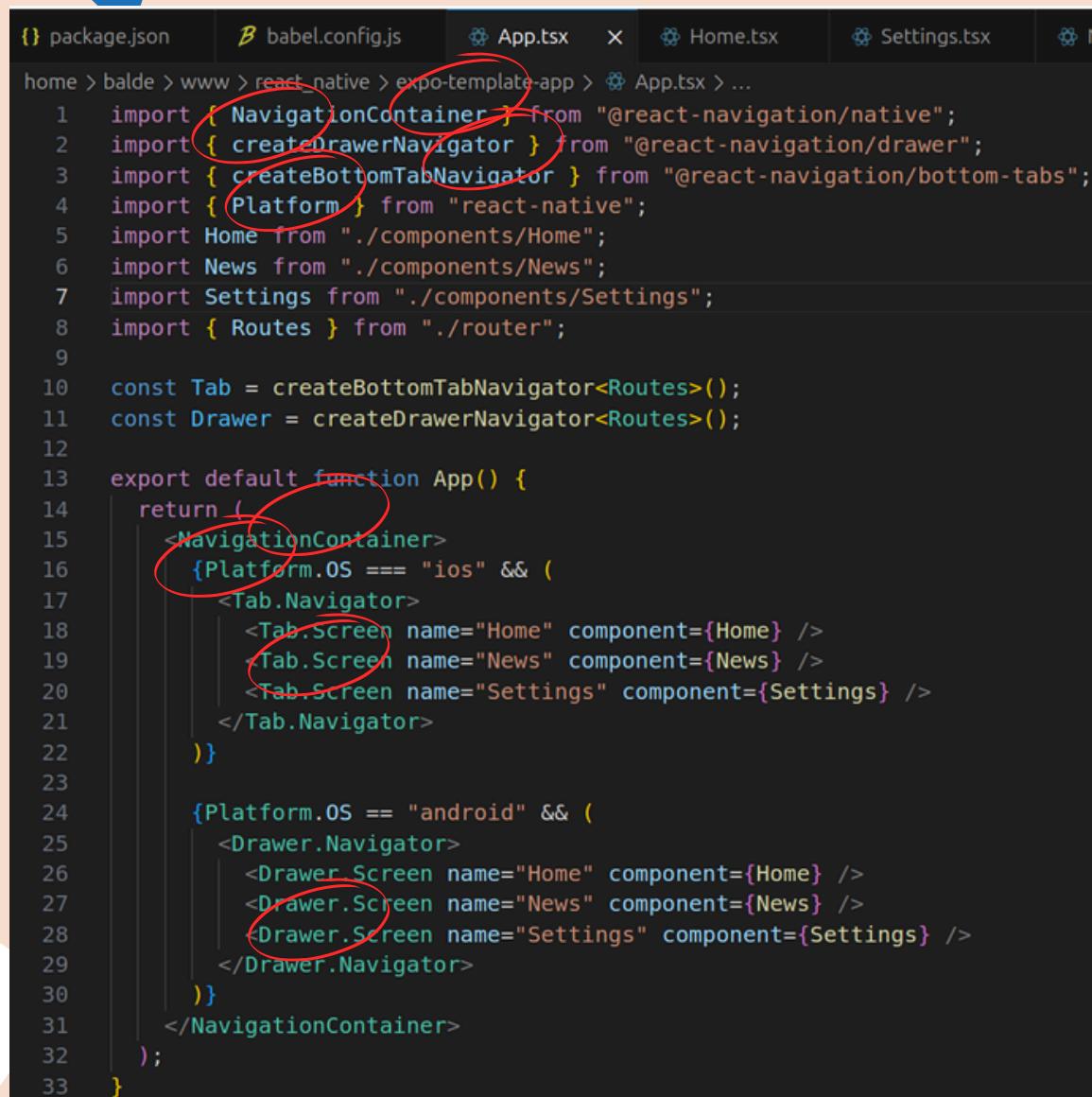
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/expo-template-app$ npx expo install react-native-gesture-handler react-native-reanimated
> Installing 2 SDK 51.0.0 compatible native modules using npm
> npm install
changed 1 package, and audited 1224 packages in 4s
132 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```



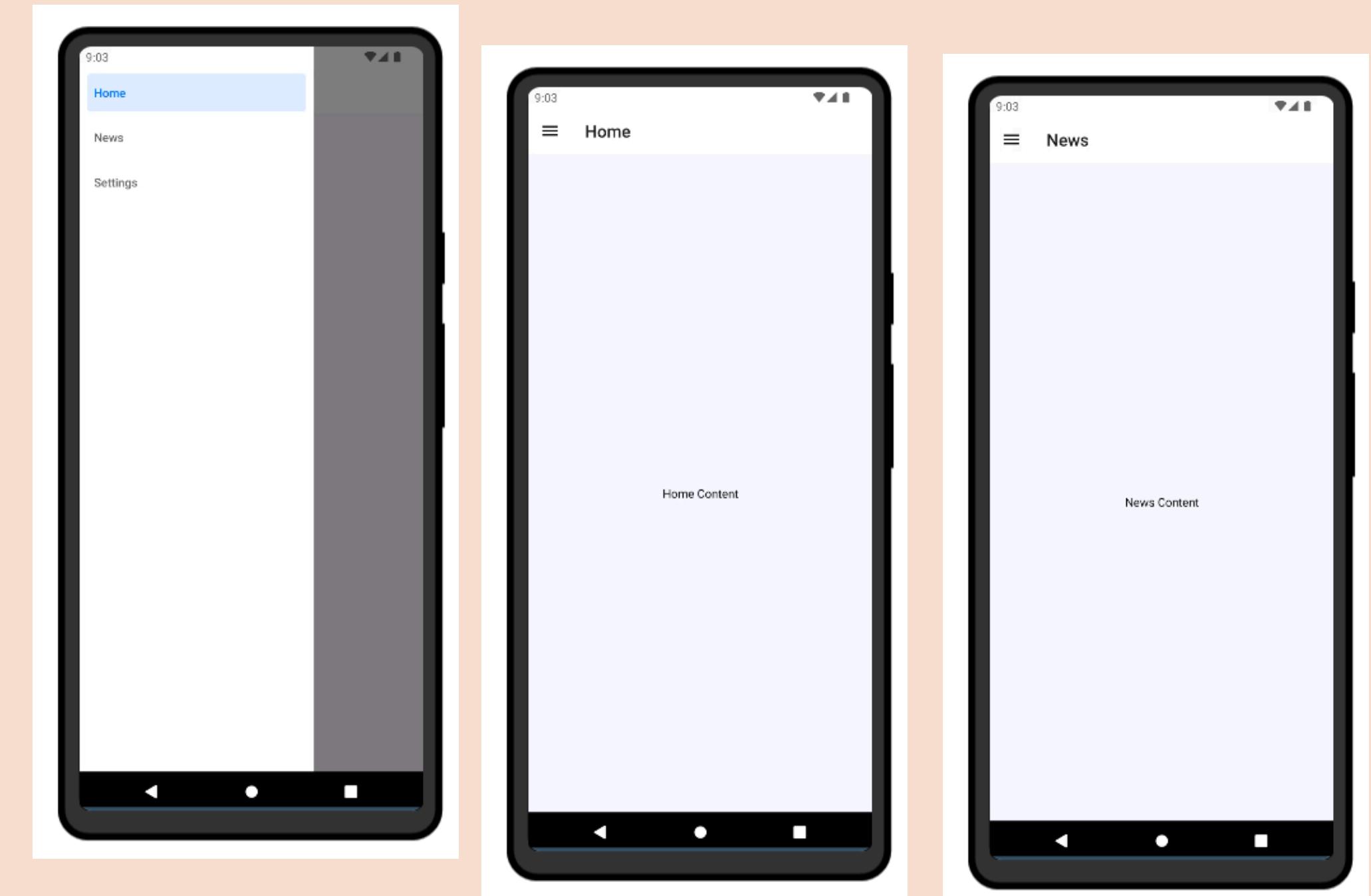
# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

NAVIGATION EN UTILISANT LES ONGLETS ET LES TABS



```
{} package.json    B babel.config.js    App.tsx    X    Home.tsx    Settings.tsx    Nev
home > balde > www > react-native > expo-template-app > App.tsx > ...
1 import { NavigationContainer } from "@react-navigation/native";
2 import { createDrawerNavigator } from "@react-navigation/drawer";
3 import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
4 import { Platform } from "react-native";
5 import Home from "./components/Home";
6 import News from "./components/News";
7 import Settings from "./components/Settings";
8 import { Routes } from "./router";
9
10 const Tab = createBottomTabNavigator<Routes>();
11 const Drawer = createDrawerNavigator<Routes>();
12
13 export default function App() {
14   return (
15     <NavigationContainer>
16       {Platform.OS === "ios" &&
17       <Tab.Navigator>
18         <Tab.Screen name="Home" component={Home} />
19         <Tab.Screen name="News" component={News} />
20         <Tab.Screen name="Settings" component={Settings} />
21       </Tab.Navigator>
22     )
23
24   {Platform.OS === "android" &&
25     <Drawer.Navigator>
26       <Drawer.Screen name="Home" component={Home} />
27       <Drawer.Screen name="News" component={News} />
28       <Drawer.Screen name="Settings" component={Settings} />
29     </Drawer.Navigator>
30   )
31   </NavigationContainer>
32 }
33 }
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

NAVIGATION EN UTILISANT LES ONGLETS ET LES TABS

```
home > balde > www > react_native > expo-template-app > App.tsx > App
1 import { NavigationContainer } from "@react-navigation/native";
2 import { createDrawerNavigator } from "@react-navigation/drawer";
3 import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
4 import { Platform } from "react-native";
5 import Home from "./components/Home";
6 import News from "./components/News";
7 import Settings from "./components/Settings";
8 import { Routes } from "./router";
9
10 const Tab = createBottomTabNavigator<Routes>();
11 const Drawer = createDrawerNavigator<Routes>();
12
13 export default function App() {
14   return (
15     <NavigationContainer>
16       {Platform.OS === "android" && [
17         <Tab.Navigator>
18           <Tab.Screen name="Home" component={Home} />
19           <Tab.Screen name="News" component={News} />
20           <Tab.Screen name="Settings" component={Settings} />
21         </Tab.Navigator>
22       ]}
23
24       {Platform.OS == "ios" && [
25         <Drawer.Navigator>
26           <Drawer.Screen name="Home" component={Home} />
27           <Drawer.Screen name="News" component={News} />
28           <Drawer.Screen name="Settings" component={Settings} />
29         </Drawer.Navigator>
30       ]}
31     </NavigationContainer>
32   );
33 }
```



# NPX ET CREATE EXPO APP

## FLEXBOX & LAYOUT "MISE EN PAGE"

### NAVIGATION EN UTILISANT LES LIENS

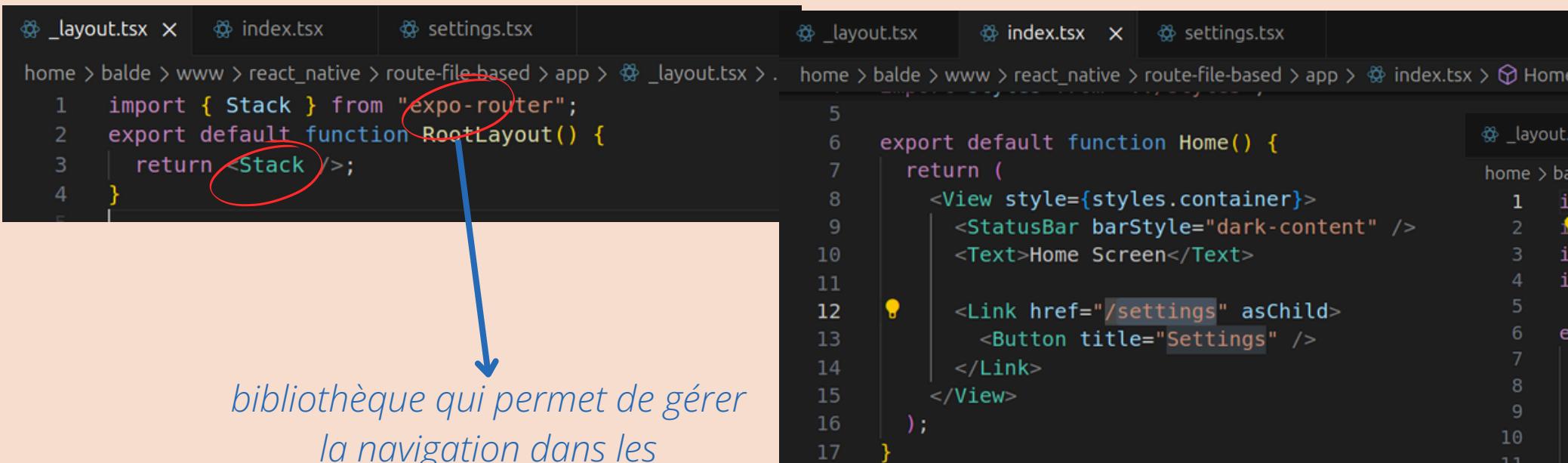
```
npm i -g @expo/react-native-template-examples
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npx create-expo-app route-file-based -template
? Choose a template: > - Use arrow-keys. Return to submit.
  Default
  Blank
  Blank (TypeScript)
  Navigation (TypeScript) - File-based routing with TypeScript enabled
  Blank (Bare)
```

Your project is ready!

To run your project, navigate to the directory and run one of the following npm commands.

- cd route-file-based
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web

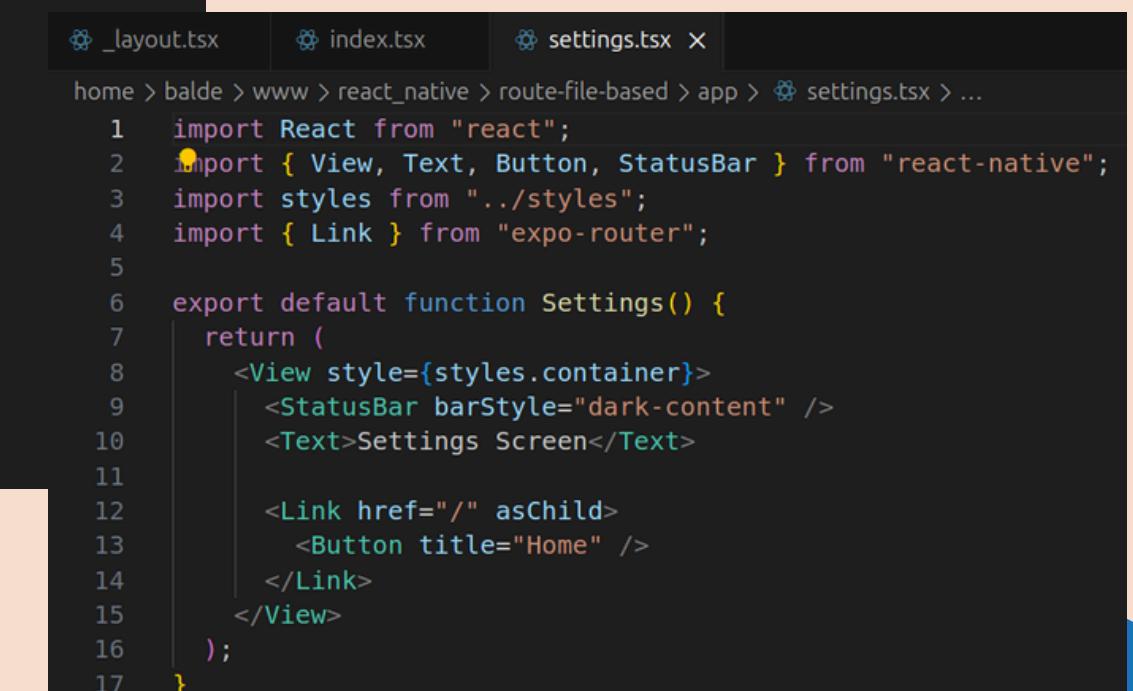
```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/route-file-based$ ls app
+html.tsx _layout.tsx modal.tsx +not-found.tsx '(tabs)'
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/route-file-based$
```



```
home > balde > www > react_native > route-file-based > app > _layout.tsx > .
1 import { Stack } from "expo-router";
2 export default function RootLayout() {
3   return <Stack>;
4 }
```

```
home > balde > www > react_native > route-file-based > app > index.tsx > Home
5
6 export default function Home() {
7   return (
8     <View style={styles.container}>
9       <StatusBar barStyle="dark-content" />
10      <Text>Home Screen</Text>
11
12      <Link href="/settings" asChild>
13        <Button title="Settings" />
14      </Link>
15    </View>
16  );
17}
```

bibliothèque qui permet de gérer la navigation dans les applications Expo de manière simple et efficace en utilisant un système basé sur les routes de fichiers.



```
home > balde > www > react_native > route-file-based > app > settings.tsx > ...
1 import React from "react";
2 import { View, Text, Button, StatusBar } from "react-native";
3 import styles from "../styles";
4 import { Link } from "expo-router";
5
6 export default function Settings() {
7   return (
8     <View style={styles.container}>
9       <StatusBar barStyle="dark-content" />
10      <Text>Settings Screen</Text>
11
12      <Link href="/" asChild>
13        <Button title="Home" />
14      </Link>
15    </View>
16  );
17}
```

# NPX ET CREATE EXPO APP

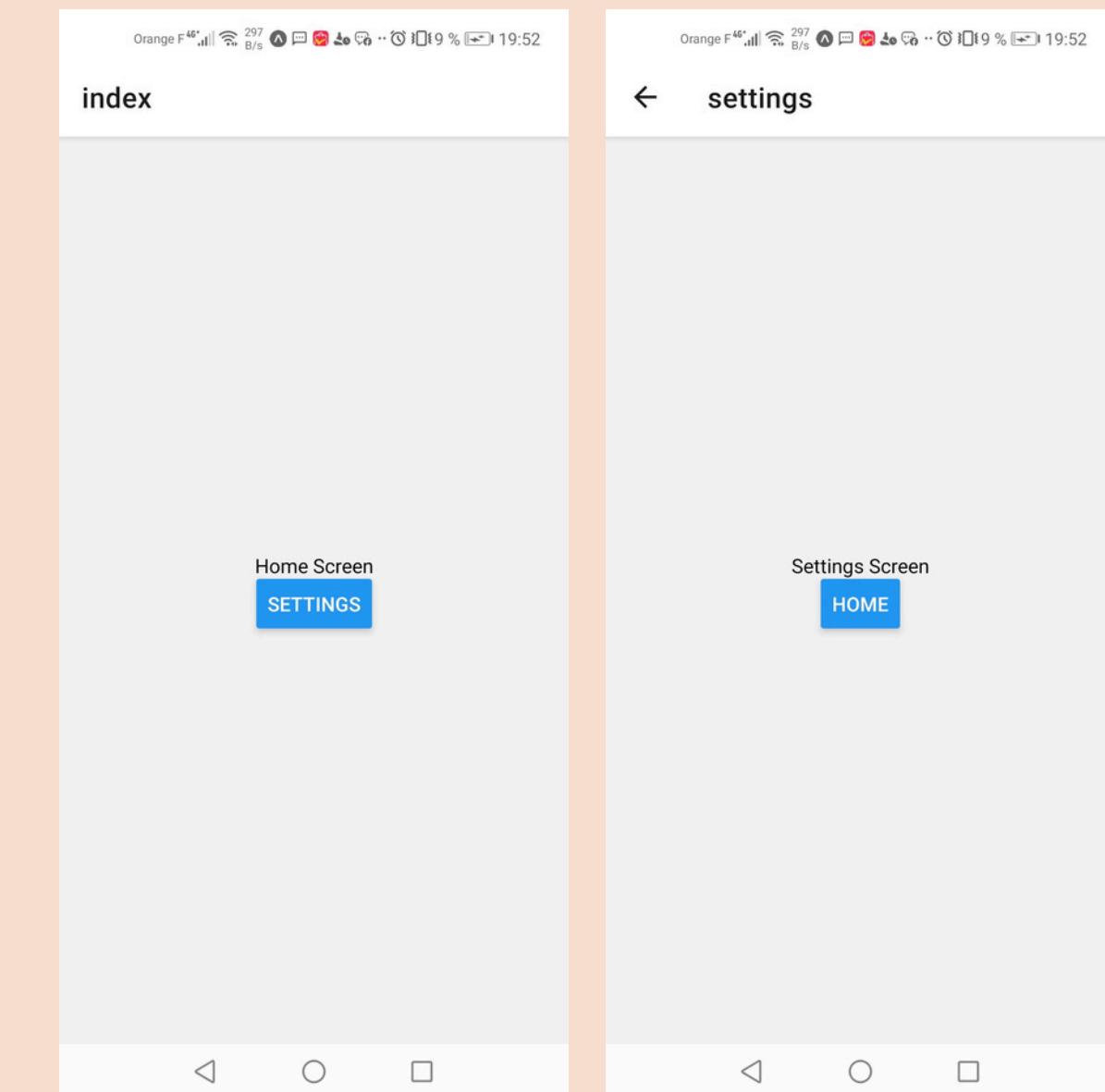
## FLEXBOX & LAYOUT "MISE EN PAGE"

NAVIGATION EN UTILISANT LES LIENS

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/route-file-based$ npm start
> route-file-based@1.0.0 start
> expo start

Starting project at /home/balde/www/react_native/route-file-based
Starting Metro Bundler
QR code to scan: 
> Metro waiting on exp://10.16.185.152:8081
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)
```



# NPX ET CREATE EXPO APP

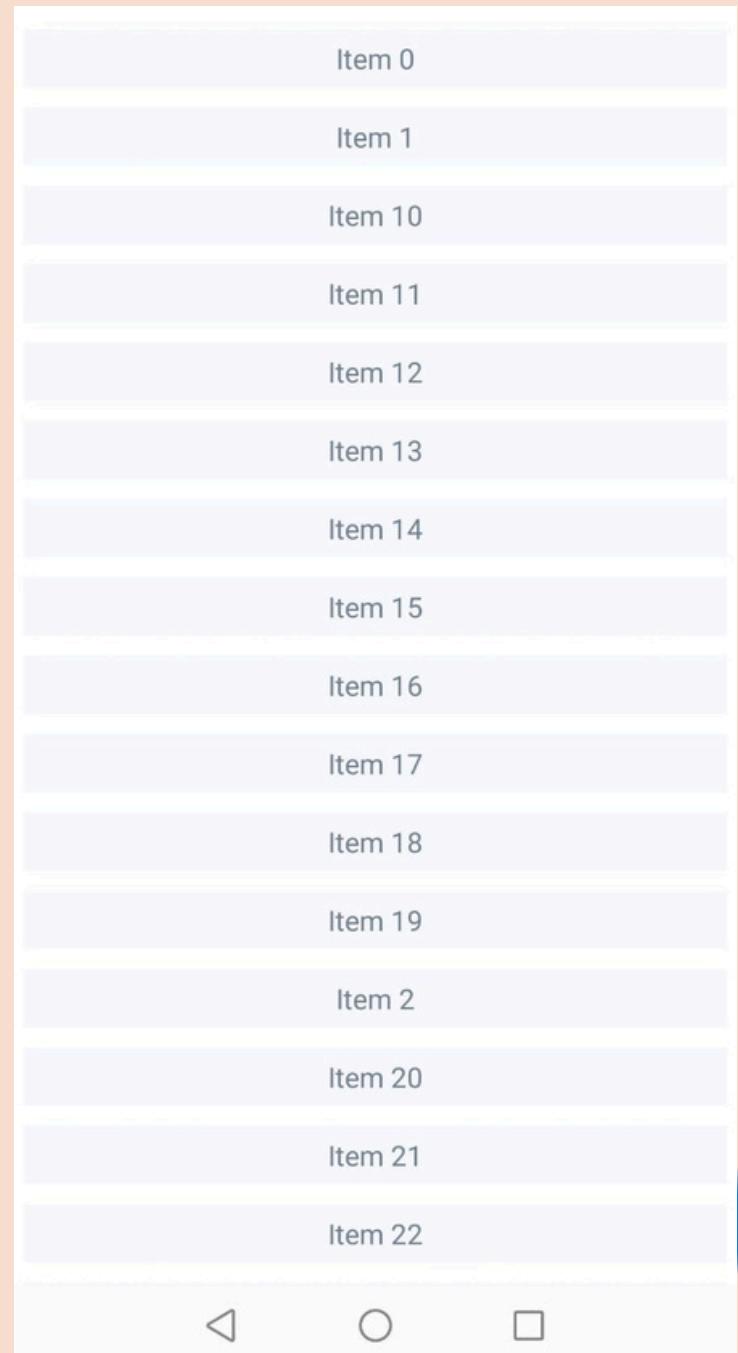
## RENDU DES LISTES D'ÉLÉMENTS

### RENDU DES COLLECTIONS DE DONNÉES

```
App.tsx
1 import { Text, View, FlatList } from "react-native";
2 import styles from "./styles";
3
4 const data = new Array(100)
5   .fill(null)
6   .map((v, i) => ({ key: i.toString(), value: `Item ${i}` }));
7
8 export default function App() {
9   return (
10     <View style={styles.container}>
11       <FlatList
12         data={data}
13         renderItem={({ item }) => <Text style={styles.item}>{item.value}</Text>}
14       />
15     </View>
16   );
17 }
```

```
styles.ts
1 import { StyleSheet } from "react-native";
2 export default StyleSheet.create({
3   container: {
4     flex: 1,
5     flexDirection: "column",
6     paddingTop: 40,
7   },
8   item: {
9     margin: 5,
10    padding: 5,
11    color: "slategrey",
12    backgroundColor: "ghostwhite",
13    textAlign: "center",
14  },
15});
```

permet de rendre des listes défilables de données de manière performante



# NPX ET CREATE EXPO APP

## RENDEZ DES LISTES D'ÉLÉMENTS



```
home > balde > www > react_native > expo-template-app > components > ListContainer.tsx > ListContainer
1 import React, { useState, useMemo } from "react";
2 import List from "./List";
3
4 function mapItems(items: string[]) {
5   return items.map((value, i) => ({ key: i.toString(), value }));
6 }
7
8 const array = new Array(100).fill(null).map((v, i) => `Item ${i}`);
9
10 function filterAndSort(text: string, asc: boolean): string[] {
11   return array
12     .filter(i => text.length === 0 || i.includes(text))
13     .sort(
14       asc
15         ? (a, b) => (a > b ? 1 : a < b ? -1 : 0)
16         : (a, b) => (b > a ? 1 : b < a ? -1 : 0)
17     );
18 }
19
20 export default function ListContainer() {
21   const [asc, setAsc] = useState(true);
22   const [filter, setFilter] = useState("");
23
24   const data = useMemo(() => {
25     return filterAndSort(filter, asc);
26   }, [filter, asc]);
27
28   return (
29     <List
30       data={mapItems(data)}
31       asc={asc}
32       onFilter={(text) => {
33         setFilter(text);
34       }}
35       onSort={() => {
36         setAsc(!asc);
37       }}
38     />
39   );
40 }
```

### TRIER ET FILTRER LES LISTES

```
home > balde > www > react_native > expo-template-app > components > List.tsx > List
1 import React from "react";
2 import { Text, FlatList } from "react-native";
3 import styles from "../styles";
4 import ListControls from "./ListControls";
5
6 type Props = {
7   data: { key: string; value: string }[];
8   onFilter: (text: string) => void;
9   onSort: () => void;
10  asc: boolean;
11 }
12
13 export default function List({ data, onFilter, onSort, asc }: Props) {
14   return (
15     <FlatList
16       data={data}
17       ListHeaderComponent={<ListControls {...{ onFilter, onSort, asc }} />}
18       renderItem={({ item }) => <Text style={styles.item}>{item.value}</Text>}
19     />
20   );
21 }
```

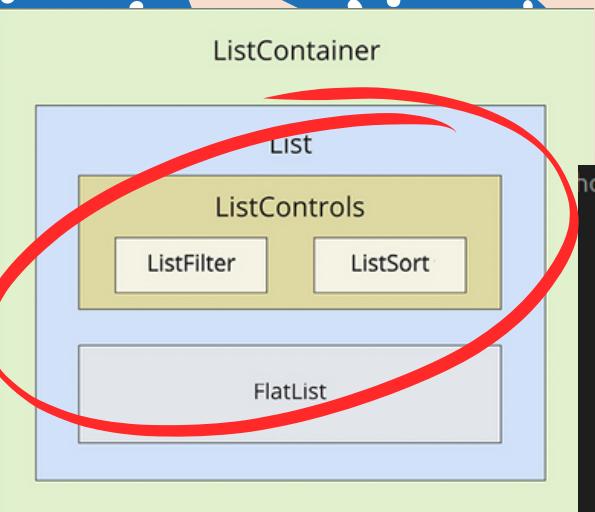
```
home > balde > www > react_native > expo-template-app > components > ListControls.tsx ...
1 import React from "react";
2 import { View } from "react-native";
3 import styles from "../styles";
4 import ListFilter from "./ListFilter";
5 import ListSort from "./ListSort";
6
7 type Props = {
8   onFilter: (text: string) => void;
9   onSort: () => void;
10  asc: boolean;
11 }
12
13 export default function ListControls({ onFilter, onSort, asc }: Props) {
14   return (
15     <View style={styles.controls}>
16       <ListFilter onFilter={onFilter} />
17       <ListSort onSort={onSort} asc={asc} />
18     </View>
19   );
20 }
```

```
home > balde > www > react_native > expo-template-app > ts styles.ts
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     flexDirection: "column",
7     paddingTop: 40,
8   },
9   item: {
10     margin: 5,
11     padding: 5,
12     color: "slategrey",
13     backgroundColor: "ghostwhite",
14     textAlign: "center",
15   },
16   filter: {
17     height: 40,
18     width: 200,
19   },
20   controls: {
21     flex: 1,
22     flexDirection: "row",
23     justifyContent: "space-between",
24     alignItems: "center",
25     padding: 10,
26     backgroundColor: "white",
27   },
28 })
```

# NPX ET CREATE EXPO APP

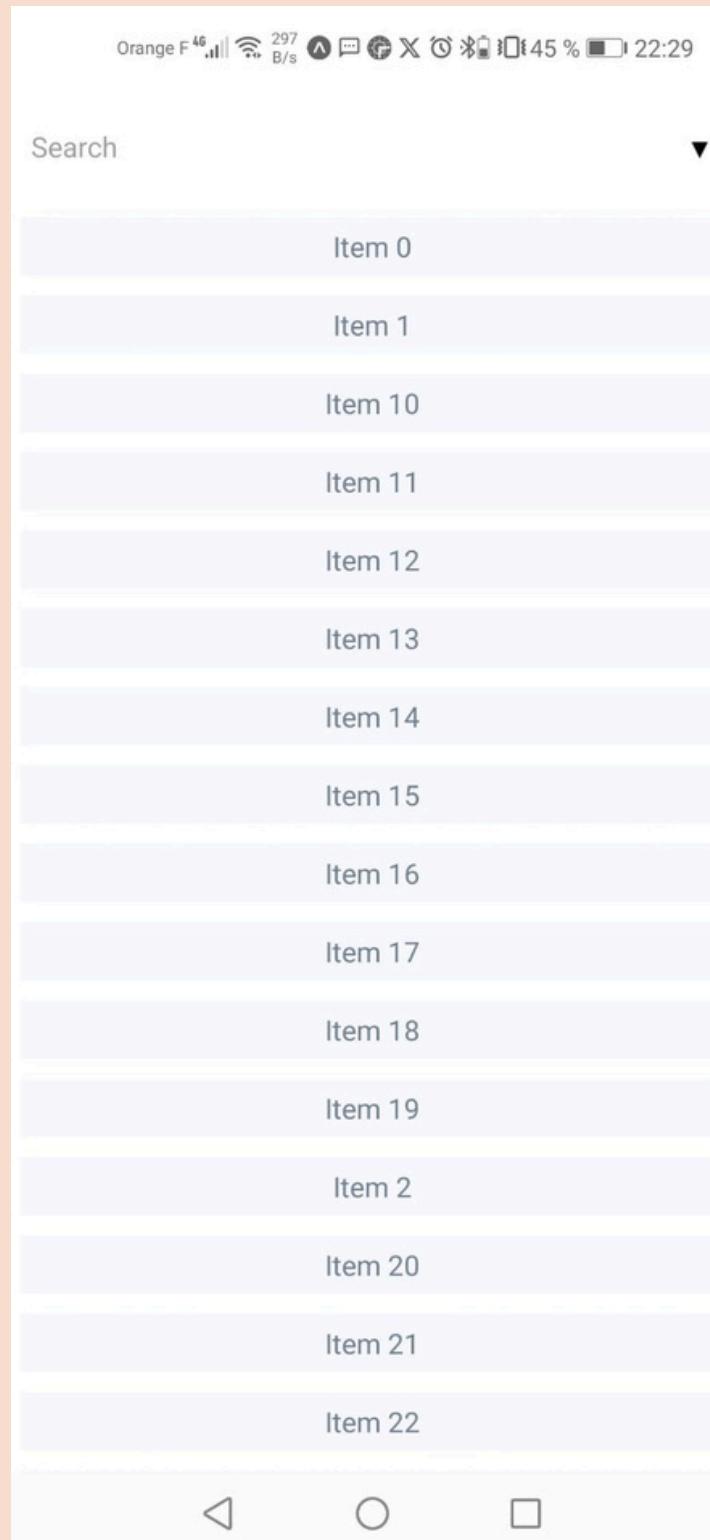
## RENDEU DES LISTES D'ÉLÉMENTS

### TRIER ET FILTRER LES LISTES



```
home > balde > www > react_native > expo-template-app > components > ListFilter.tsx > L
1 import React from "react";
2 import { View, TextInput } from "react-native";
3 import styles from "../styles";
4
5 type Props = {
6   onFilter: (text: string) => void;
7 };
8
9 export default function ListFilter({ onFilter }: Props) {
10   return (
11     <View>
12       <TextInput
13         autoFocus
14         placeholder="Search"
15         style={styles.filter}
16         onChangeText={onFilter}
17       />
18     </View>
19   );
20 }
```

```
home > balde > www > react_native > expo-template-app > components > ListSort.tsx > P
1 import React from "react";
2 import { Text } from "react-native";
3
4 const arrows = new Map([
5   [true, "▼"],
6   [false, "▲"],
7 ]);
8
9 type Props = {
10   onSort: () => void;
11   asc: boolean;
12 };
13
14 export default function ListSort({ onSort, asc }: Props) {
15   return <Text onPress={onSort}>{arrows.get(asc)}</Text>;
16 }
```



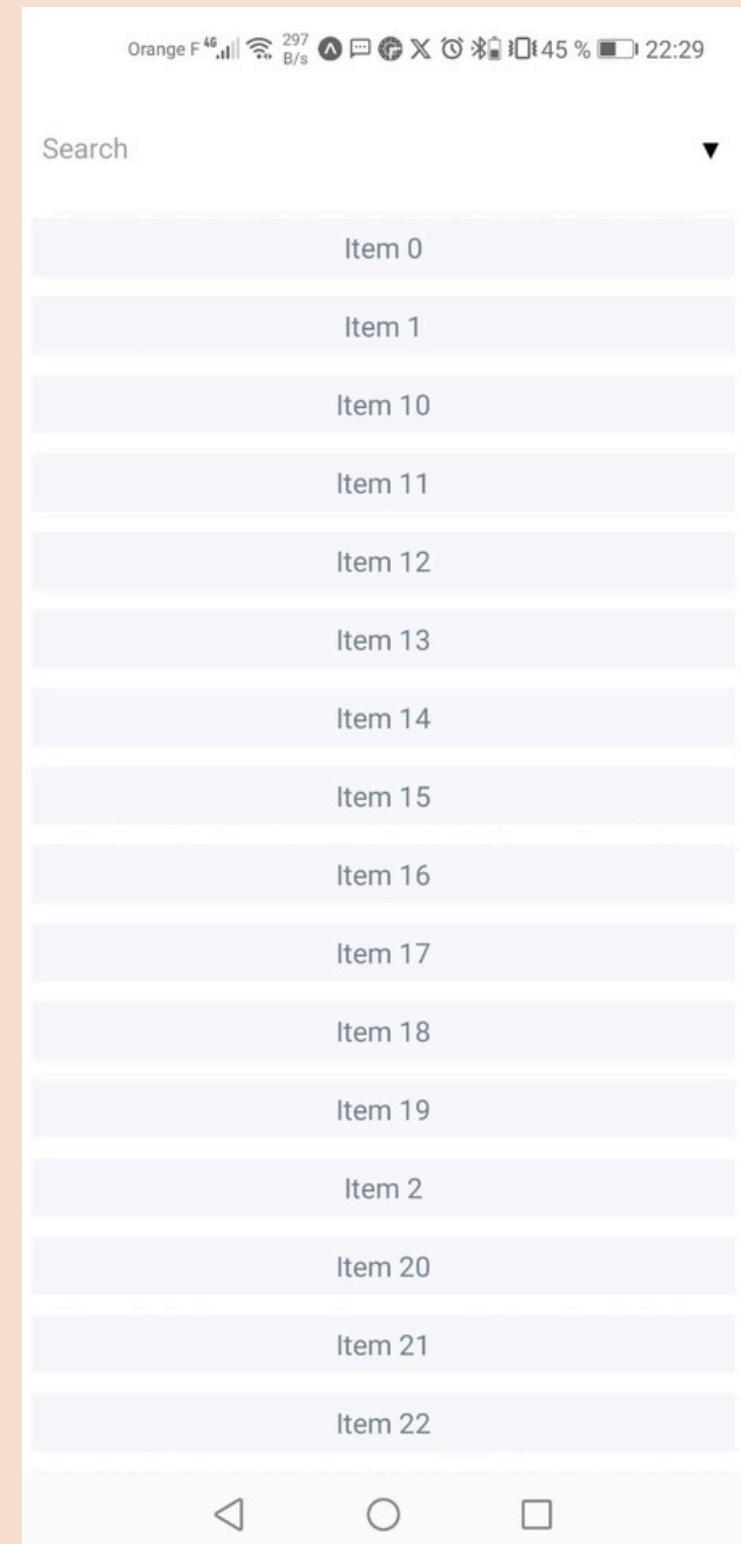
# NPX ET CREATE EXPO APP

## RENDEZ DES LISTES D'ÉLÉMENTS

### RÉCUPÉRATION DES DONNÉES DE LA LISTE VIA API

```
home > balde > www > react_native > expo-template-app > ts api.ts > filterAndSort
1 const items = new Array(100).fill(null).map((v, i) => `Item ${i}`);
2
3 function filterAndSort(data: string[], text: string, asc: boolean) {
4   return data
5     .filter((i) => text.length === 0 || i.includes(text))
6     .sort(
7       asc
8         ? (a, b) => (b > a ? -1 : a === b ? 0 : 1)
9         : (a, b) => [a > b ? -1 : a === b ? 0 : 1]
10    );
11 }
12
13 export function fetchItems(
14   filter: string,
15   asc: boolean
16 ): Promise<{ json: () => Promise<{ items: string[] }> }> {
17   return new Promise((resolve) => {
18     resolve({
19       json: () =>
20         Promise.resolve({
21           items: filterAndSort(items, filter, asc),
22         }),
23     });
24   });
25 }
```

```
home > balde > www > react_native > expo-template-app > components > ListContainer.tsx > ListContainer
1 import React, { useState, useEffect } from "react";
2 import { fetchItems } from "../api";
3 import List from "./List";
4
5 type MappedList = { key: string; value: string }[];
6
7 function mapItems(items: string[]): MappedList {
8   return items.map((value, i) => ({ key: i.toString(), value }));
9 }
10
11 export default function ListContainer() {
12   const [asc, setAsc] = useState(true);
13   const [filter, setFilter] = useState("");
14   const [data, setData] = useState<MappedList>([]);
15
16   useEffect(() => {
17     fetchItems(filter, asc)
18       .then((resp) => resp.json())
19       .then(({ items }) => {
20         setData(mapItems(items));
21       });
22   }, []);
23
24   return (
25     <List
26       data={data}
27       asc={asc}
28       onFilter={(text) => {
29         fetchItems(text, asc)
30           .then((resp) => resp.json())
31           .then(({ items }) => {
32             setFilter(text);
33             setData(mapItems(items));
34           });
35       }}
36       onSort={() => {
37         fetchItems(filter, !asc)
38           .then((resp) => resp.json())
39           .then(({ items }) => {
40             setAsc(!asc);
41             setData(mapItems(items));
42           });
43       }}
44     />
45   );
46 }
```



# NPX ET CREATE EXPO APP

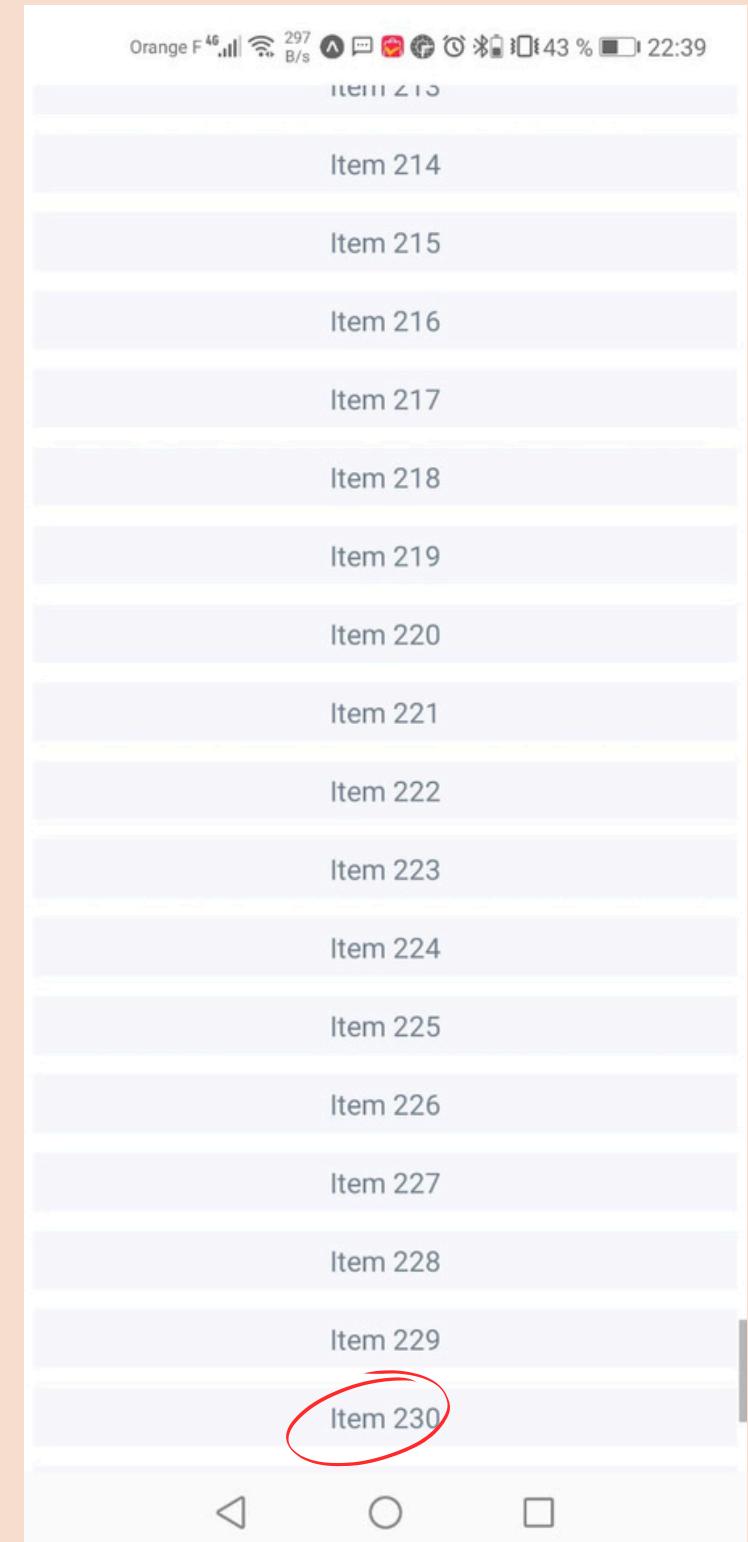
## RENDU DES LISTES D'ÉLÉMENTS

### CHARGEMENT DIFFÉRÉ "LOADING" DE LA LISTE

```
home > balde > www > react_native > expo-template-app > ts api.ts > fetchItems
1 function* genItems() {
2   let cnt = 0;
3
4   while (true) {
5     yield `Item ${cnt++}`;
6   }
7
8
9 let items = genItems();
10
11 export function fetchItems({ refresh }: { refresh?: boolean }) {
12   if (refresh) {
13     items = genItems();
14   }
15
16   return Promise.resolve({
17     json: () =>
18       Promise.resolve({
19         items: new Array(30).fill(null).map(() => items.next().value as string),
20       }),
21     });
22 }
```

```
home > balde > www > react_native > expo-template-app > components > List.tsx > List
1 import React from "react";
2 import { Text, FlatList } from "react-native";
3 import styles from "../styles";
4
5 type Props = {
6   data: { key: string; value: string }[];
7   fetchItems: () => Promise<void>;
8 };
9
10 export default function List({ data, fetchItems, ... }: Props) {
11   return (
12     <FlatList
13       data={data}
14       renderItem={({ item }) => <Text style={styles.item}>{item.value}</Text>}
15       onEndReached={fetchItems}
16     />
17   );
18 }
19
20 }
```

```
home > balde > www > react_native > expo-template-app > components > ListContainer.tsx > ListContainer
1 import React, { useState, useEffect } from "react";
2 import * as api from "../api";
3 import List from "./List";
4
5 export default function ListContainer() {
6   const [data, setData] = useState<{ key: string; value: string }[]>([]);
7   const [isRefreshing, setIsRefreshing] = useState(false);
8
9   function fetchItems() {
10     return api
11       .fetchItems({})
12       .then((resp) => resp.json())
13       .then(({ items }) => {
14         setData([
15           ...data,
16           ...items.map((value) => ({
17             key: value,
18             value,
19           })),
20         ]);
21       });
22 }
23
24 useEffect(() => {
25   fetchItems();
26 }, []);
27
28 return (
29   <List data={data} fetchItems={fetchItems}
30   />
31 );
32 }
```



# NPX ET CREATE EXPO APP

## RENDEZ DES LISTES D'ÉLÉMENTS

IMPLEMENTATION DE "PULL TO REFRESH" => TIRER POUR ACTUALISER

```
home > balde > www > react_native > expo-template-app > ts api.ts > fetchItems
1  function* genItems() {
2    let cnt = 0;
3
4    while (true) {
5      yield `Item ${cnt++}`;
6    }
7
8    let items = genItems();
9
10   export function fetchItems({ refresh }: { refresh?: boolean }) {
11     if (refresh) {
12       items = genItems();
13     }
14
15     return Promise.resolve({
16       json: () =>
17         Promise.resolve({
18           items: new Array(30).fill(null).map(() => items.next().value as string),
19         }),
20     });
21   }
22 }

home > balde > www > react_native > expo-template-app > components > List.tsx > List
1  import React from "react";
2  import { Text, FlatList } from "react-native";
3  import styles from "../styles";
4
5  type Props = {
6    data: { key: string; value: string }[];
7    fetchItems: () => Promise<void>;
8    refreshItems: () => Promise<void>;
9    isRefreshing: boolean;
10  };
11
12  export default function List({ data, fetchItems, refreshItems, isRefreshing, ... }: Props) {
13    return (
14      <FlatList
15        data={data}
16        renderItem={({ item }) => <Text style={styles.item}>{item.value}</Text>}
17        onEndReached={fetchItems}
18        onRefresh={refreshItems}
19        refreshing={isRefreshing}
20      />
21    );
22  }
23
```

```
home > balde > www > react_native > expo-template-app > components > ListContainer.tsx > ListContainer
1  import React, { useState, useEffect } from "react";
2  import * as api from "../api";
3  import List from "./List";
4
5  export default function ListContainer() {
6    const [data, setData] = useState<{ key: string; value: string }[]>([]);
7    const [isRefreshing, setIsRefreshing] = useState(false);
8
9    function ...fetchItems() {
10      return api
11        .fetchItems({})
12        .then((resp) => resp.json())
13        .then(({ items }) => {
14          setData([
15            ...data,
16            ...items.map((value) => ({
17              key: value,
18              value,
19            })),
20          ]);
21        });
22    }
23
24    function ...refreshItems() {
25      setIsRefreshing(true);
26      return api
27        .fetchItems({ refresh: true })
28        .then((resp) => resp.json())
29        .then(({ items }) => {
30          setData(
31            items.map((value) => ({
32              key: value,
33              value,
34            }))
35          );
36        })
37        .finally(() => {
38          setIsRefreshing(false);
39        });
40    }
41
42    useEffect(() => {
43      fetchItems();
44    }, []);
45
46    return (
47      <List data={data} fetchItems={fetchItems}
48        refreshItems={refreshItems}
49        isRefreshing={isRefreshing}
50      />
51    );
52 }
```

Comme nous avons fourni les props `onRefresh` et `refreshing`, notre composant `FlatList` active automatiquement le geste tirer pour actualiser. Le gestionnaire `onRefresh` sera appelé lorsque vous extrayez la liste et la propriété d'actualisation permettra au spinner de chargement de refléter l'état de chargement.

```
home > balde > www > react_native > expo-template-app > components > ListContainer.tsx > ListContainer
5  export default function ListContainer() {
6
7    function ...refreshItems() {
8      setIsRefreshing(true);
9      return api
10        .fetchItems({ refresh: true })
11        .then((resp) => resp.json())
12        .then(({ items }) => {
13          setData(
14            items.map((value) => ({
15              key: value,
16              value,
17            }))
18          );
19        })
20        .finally(() => {
21          setIsRefreshing(false);
22        });
23    }
24
25    useEffect(() => {
26      fetchItems();
27    }, []);
28
29    return (
30      <List data={data} fetchItems={fetchItems}
31        refreshItems={refreshItems}
32        isRefreshing={isRefreshing}
33      />
34    );
35  }
```

Pour appliquer les props définis dans le composant `List`, implémentons la fonction `refreshItems` avec l'état `isRefreshing` dans le composant `ListContainer`. Dans `refreshItems`, ainsi que dans la méthode `fetchItems`, nous obtenons les éléments de la liste mais les enregistrons en tant que nouvelle liste.

# NPX ET CREATE EXPO APP

## LA GEOLOCALISATION - CARTE GOOGLE MAPS

### UTILISATION DE L'API DE GÉOLOCALISATION

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npx create-expo-app geo-and-maps --template
✓ Choose a template: > Blank (TypeScript)
✓ Downloaded and extracted project files.
> npm install
```

To run your project, navigate to the directory and run one of the following npm commands.

```
- cd geo-and-maps
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ cd geo-and-maps/
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/geo-and-maps$ npx expo install expo-location
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

added 1 package, and audited 1194 packages in 4s

131 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

*expo-location est un module Expo qui fournit un accès facile à la localisation de l'appareil sur les appareils mobiles. Il permet à une application Expo d'obtenir des informations sur la position géographique de l'appareil, telles que la latitude, la longitude, la vitesse, l'altitude, etc. Cela permet de créer des applications qui nécessitent des fonctionnalités basées sur la localisation, comme la cartographie, la navigation, le suivi, etc.*



#### Location

A library that provides access to reading geolocation information, polling current location or subscribing location update events from the device.

Expo Documentation

<https://docs.expo.dev/versions/latest/sdk/location/>

```
geo-and-maps > {} package.json > ...
1  {
2    "name": "geo-and-maps",
3    "version": "1.0.0",
4    "main": "expo/AppEntry.js",
5    "scripts": {
6      "start": "expo start",
7      "android": "expo start --android",
8      "ios": "expo start --ios",
9      "web": "expo start --web"
10     },
11     "dependencies": {
12       "expo": "~51.0.8",
13       "expo-status-bar": "~1.12.1",
14       "react": "18.2.0",
15       "react-native": "0.74.1",
16       "expo-location": "~17.0.1"
17     },
18     "devDependencies": {
19       "@babel/core": "^7.20.0",
20       "@types/react": "~18.2.45",
21       "typescript": "^5.1.3"
22     },
23     "private": true
24   }
25 }
```

```
geo-and-maps > {} app.json > {} expo
1  {
2    "expo": [
3      {
4        "name": "geo-and-maps",
5        "slug": "geo-and-maps",
6        "version": "1.0.0",
7        "orientation": "portrait",
8        "icon": "./assets/icon.png",
9        "userInterfaceStyle": "light",
10       "splash": {
11         "image": "./assets/splash.png",
12         "resizeMode": "contain",
13         "backgroundColor": "#ffffff"
14       },
15       "ios": {
16         "supportsTablet": true
17       },
18       "android": {
19         "adaptiveIcon": {
20           "foregroundImage": "./assets/adaptive-icon.png",
21           "backgroundColor": "#ffffff"
22         }
23       },
24       "web": {
25         "favicon": "./assets/favicon.png"
26       },
27       "plugins": [
28         {
29           "expo-location": {
30             "locationAlwaysAndWhenInUsePermission": "Allow $(PRODUCT_NAME) to use your location."
31           }
32         }
33       ]
34     }
35   }
```

# NPX ET CREATE EXPO APP

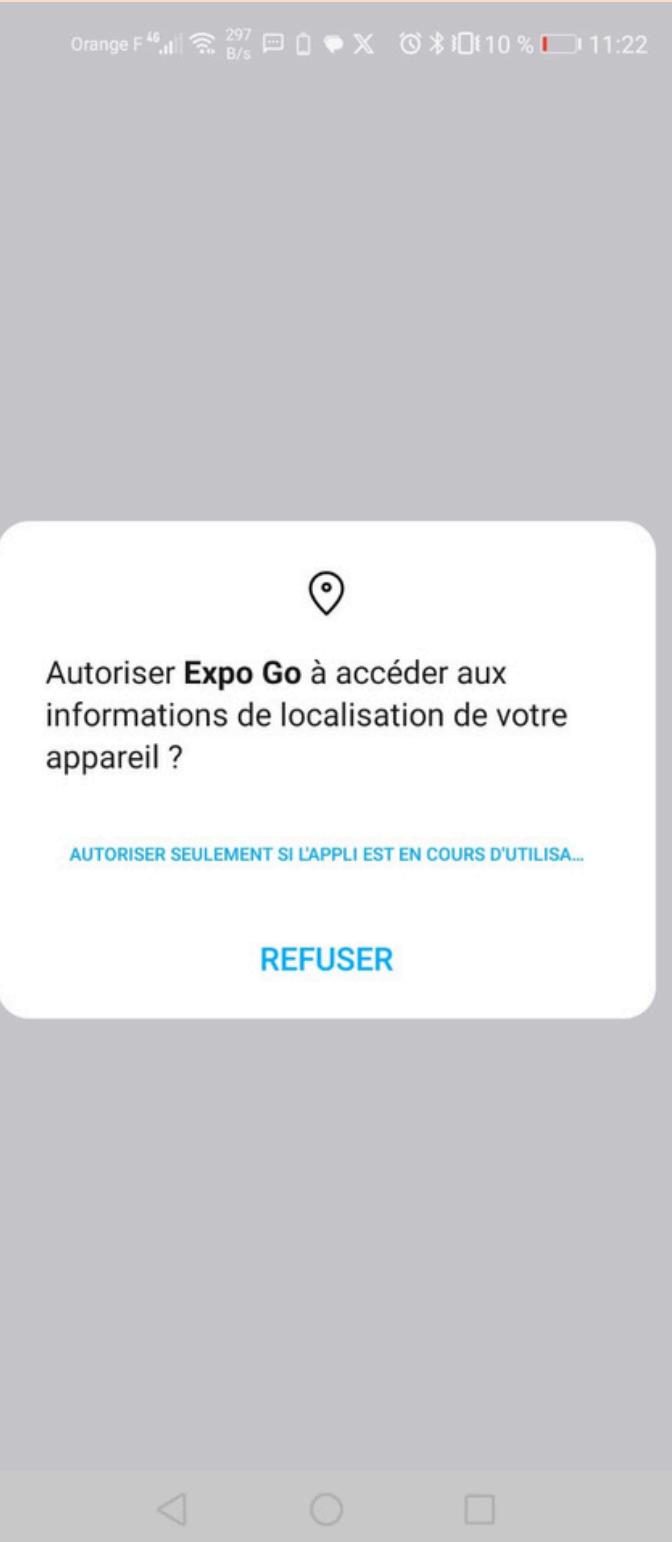
## LA GEOLOCALISATION - CARTE GOOGLE MAPS

### UTILISATION DE L'API DE GÉOLOCALISATION

```
geo-and-maps > App.tsx > WhereAmI > useEffect() callback > setPosition > then() callback
1 import React, { useState, useEffect } from "react";
2 import { Text, View } from "react-native";
3 import * as Location from "expo-location";
4 import styles from "./styles";
5
6 const API_KEY = "";
7 const URL = `https://maps.googleapis.com/maps/api/geocode/json?key=${API_KEY}&latlng=`;
8
9 export default function WhereAmI() {
10   const [address, setAddress] = useState("loading...");
11   const [longitude, setLongitude] = useState<number | undefined>();
12   const [latitude, setLatitude] = useState<number | undefined>();
13
14   useEffect(() => {
15     function setPosition({
16       coords: { latitude, longitude },
17     }: Location.LocationObject) {
18       setLongitude(longitude);
19       setLatitude(latitude);
20
21       fetch(`${URL}${latitude},${longitude}`)
22         .then((resp) => resp.json())
23         .then(({ results }) => {
24           if (results.length > 0) {
25             setAddress(results[0].formatted_address);
26           }
27         })
28         .catch((error) => {
29           console.log(error.message);
30         });
31   }
32
33   let watcher: Location.LocationSubscription;
34
35   (async () => {
36     let { status } = await Location.requestForegroundPermissionsAsync();
37     if (status !== "granted") {
38       return;
39     }
40
41     let location = await Location.getCurrentPositionAsync({});
42     setPosition(location);
43
44     watcher = await Location.watchPositionAsync(
45       { accuracy: Location.LocationAccuracy.Highest },
46       setPosition
47     );
48   })();
49
50   return () => {
51     watcher?.remove();
52   };
53 }, [ ]);
54
55   return (
56     <View style={styles.container}>
57       <Text style={styles.label}>Address: {address}</Text>
58       <Text style={styles.label}>Latitude: {latitude}</Text>
59       <Text style={styles.label}>Longitude: {longitude}</Text>
60     </View>
61   );
62 }
```

```
geo-and-maps > App.tsx > WhereAmI > useEffect() callback > setPosition > then() callback
9   export default function WhereAmI() {
14     useEffect(() => {
15       (async () => {
16         watcher = await Location.watchPositionAsync(
17           { accuracy: Location.LocationAccuracy.Highest },
18           setPosition
19         );
20       })();
21
22       return () => {
23         watcher?.remove();
24       };
25     }, [ ]);
26
27   return (
28     <View style={styles.container}>
29       <Text style={styles.label}>Address: {address}</Text>
30       <Text style={styles.label}>Latitude: {latitude}</Text>
31       <Text style={styles.label}>Longitude: {longitude}</Text>
32     </View>
33   );
34 }
```

```
geo-and-maps > ts styles.ts > default > label
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10  label: {
11    textAlign: "center",
12    margin: 10,
13  },
14  address: {
15    fontWeight: "bold",
16  },
17});
```



Orange F 4G | 297 B/s 10% 11:22

Orange F 4G | 3,5 K/s 10% 11:22

Address: loading...

Latitude: 47.8304714

Longitude: -0.3256963

# NPX ET CREATE EXPO APP

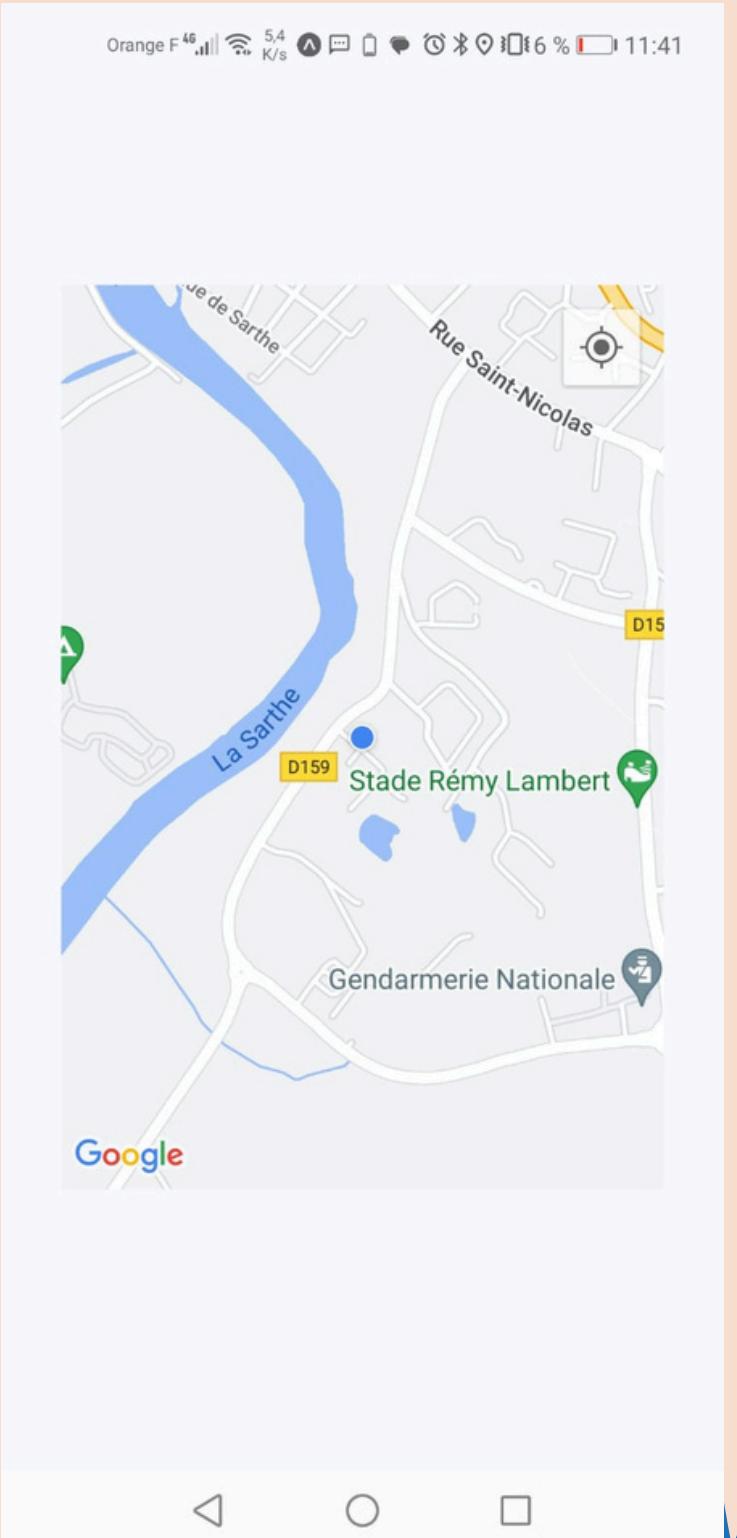
## LA GEOLOCALISATION - CARTE GOOGLE MAPS

### RENDU DE LA CARTE

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/geo-and-maps$ npm i react-native-maps
added 2 packages, and audited 1196 packages in 4s
131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
geo-and-maps > App.tsx > default
1 import React from "react";
2 import { View, StatusBar } from "react-native";
3 import MapView from "react-native-maps";
4 import styles from "./styles";
5
6 StatusBar.setBarStyle("dark-content");
7
8 export default () => (
9   <View style={styles.container}>
10    <MapView style={styles.mapView} showsUserLocation followsUserLocation />
11  </View>
12);
```

```
geo-and-maps > styles.ts > default > mapView
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   mapView: [
12     alignSelf: "stretch",
13     height: 450,
14     margin: 30,
15   ],
16});
```



# NPX ET CREATE EXPO APP

## LA GEOLOCALISATION - CARTE GOOGLE MAPS

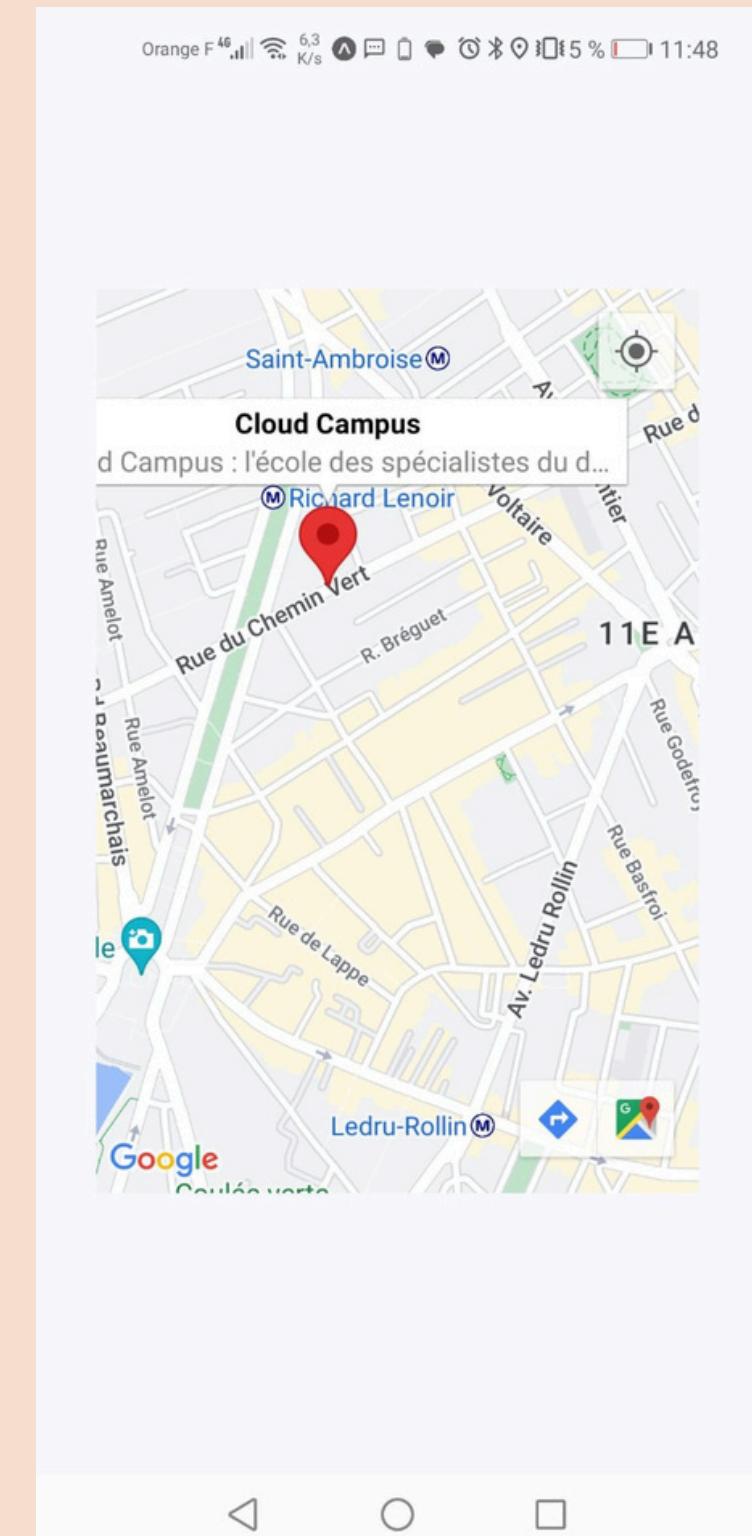
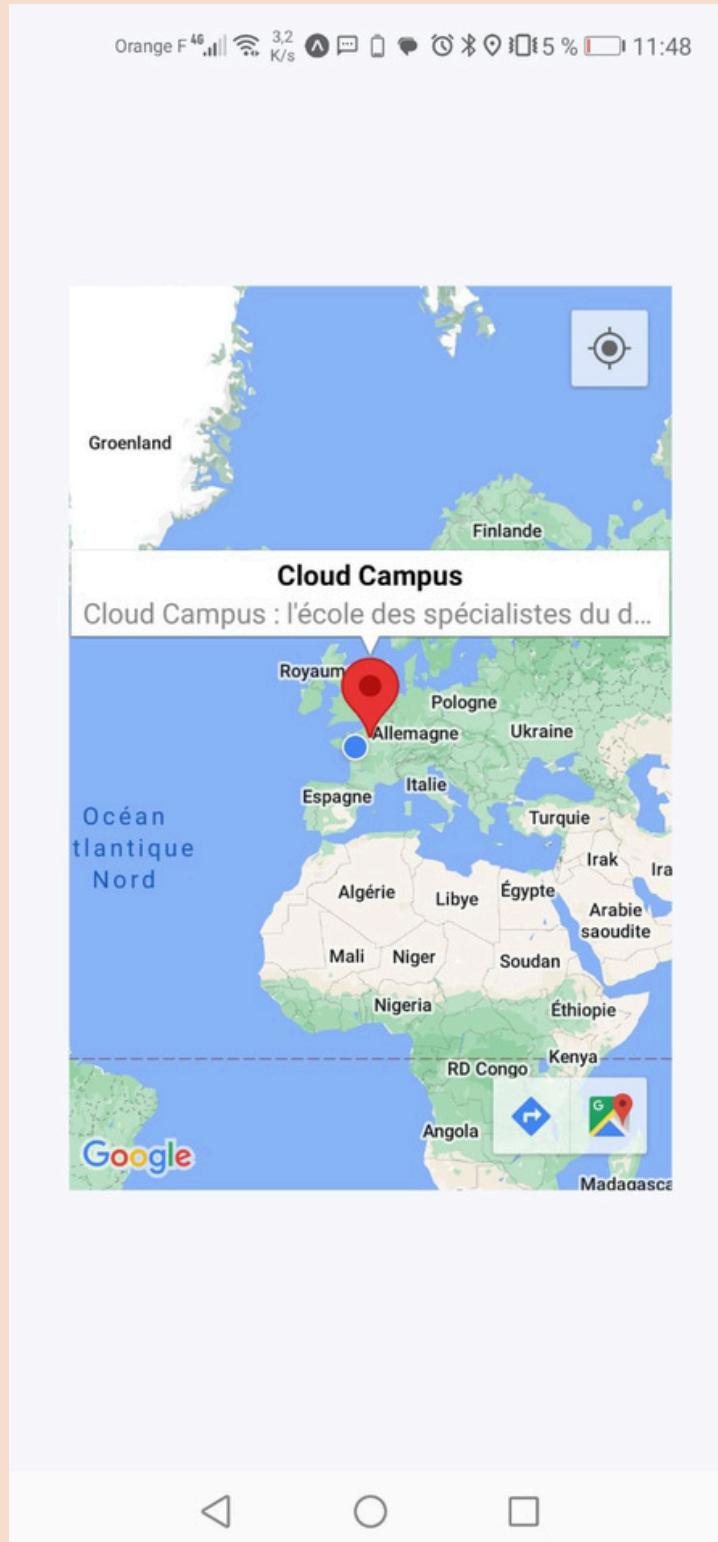
### ANNOTER DES POINTS D'INTÉRÊT

```
geo-and-maps > App.tsx > ...
1 import React from "react";
2 import { View, StatusBar } from "react-native";
3 import MapView, {Marker} from "react-native-maps";
4 import styles from "./styles";
5
6 StatusBar.setBarStyle("dark-content");
7
8 export default () =>
9   <View style={styles.container}>
10    <MapView style={styles.mapView} showsUserLocation followsUserLocation >
11
12      <Marker
13        title="Cloud Campus"
14        description="Cloud Campus : l'école des spécialistes du développement web"
15        coordinate={{
16          latitude: 48.8585802,
17          longitude: 2.3730884,
18        }}>
19      />
20    </MapView>
21  </View>
22 );
```

```
geo-and-maps > ts styles.ts > default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   mapView: {
12     alignSelf: "stretch",
13     height: 450,
14     margin: 30,
15   },
16 });

```



# NPX ET CREATE EXPO APP

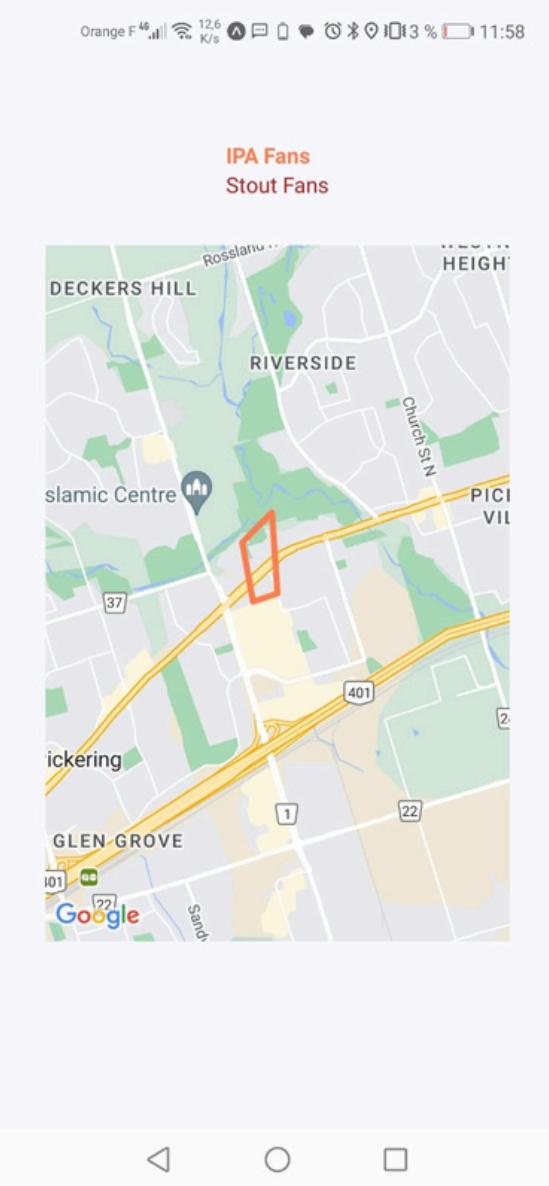
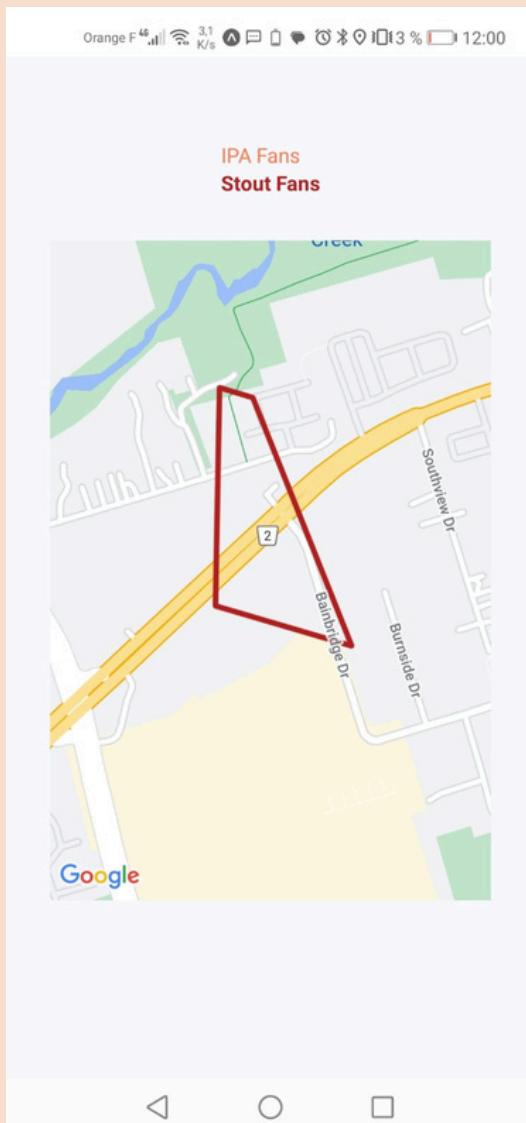
## LA GEOLOCALISATION - CARTE GOOGLE MAPS

### TRAÇAGE DES SUPERPOSITIONS

```
geo-and-maps > App.tsx > PlottingOverlays > onClickStout
1 import React, { useState } from "react";
2 import { View, Text, StatusBar } from "react-native";
3 import MapView, { Polygon } from "react-native-maps";
4 import styles from "./styles";
5
6 StatusBar.setBarStyle("dark-content");
7
8 type Overlay = {
9   coordinates: { latitude: number; longitude: number }[];
10  strokeColor: string;
11  strokeWidth: number;
12};
13
14 const ipaRegion: Overlay = {
15   coordinates: [
16     { latitude: 43.8486744, longitude: -79.0695283 },
17     { latitude: 43.8537168, longitude: -79.0700046 },
18     { latitude: 43.8518394, longitude: -79.0725697 },
19     { latitude: 43.8481651, longitude: -79.0716377 },
20     { latitude: 43.8486744, longitude: -79.0695283 },
21   ],
22   strokeColor: "coral",
23   strokeWidth: 4,
24 };
25
26 const stoutRegion: Overlay = {
27   coordinates: [
28     { latitude: 43.8486744, longitude: -79.0693283 },
29     { latitude: 43.8517168, longitude: -79.0710046 },
30     { latitude: 43.8518394, longitude: -79.0715697 },
31     { latitude: 43.8491651, longitude: -79.0716377 },
32     { latitude: 43.8486744, longitude: -79.0693283 },
33   ],
34   strokeColor: "firebrick",
35   strokeWidth: 4,
36 };
37
38 export default function PlottingOverlays() {
39   const [ipaStyles, setIpStyles] = useState<any>([
40     styles.ipaText,
41     styles.boldText,
42   ]);
43   const [stoutStyles, setStoutStyles] = useState<any>([styles.stoutText]);
44   const [overlays, setOverlays] = useState<Overlay[]>([ipaRegion]);
45
46   function onClickIpa() {
47     setIpStyles([...ipaStyles, styles.boldText]);
48     setStoutStyles([stoutStyles[0]]);
49     setOverlays([ipaRegion]);
50   }
51
52   function onClickStout() {
53     setStoutStyles([...stoutStyles, styles.boldText]);
54     setIpStyles([ipaStyles[0]]);
55     setOverlays([stoutRegion]);
56   }
57 }
```

```
geo-and-maps > App.tsx > PlottingOverlays > onClickStout
38 export default function PlottingOverlays() {
39   function onClickStout() {
40     setStoutStyles([...stoutStyles, styles.boldText]);
41     setIpStyles([ipaStyles[0]]);
42     setOverlays([stoutRegion]);
43   }
44
45   return (
46     <View style={styles.container}>
47       <View>
48         <Text style={ipaStyles} onPress={onClickIpa}>
49           IPA Fans
50         </Text>
51         <Text style={stoutStyles} onPress={onClickStout}>
52           Stout Fans
53         </Text>
54       </View>
55       <MapView
56         style={styles.mapView}
57         showsPointsOfInterest={false}
58         initialRegion={{
59           latitude: 43.8486744,
60           longitude: -79.0695283,
61           latitudeDelta: 0.002,
62           longitudeDelta: 0.04,
63         }}
64
65         >
66           {overlays.map((v, i) => (
67             <Polygon
68               key={i}
69               coordinates={v.coordinates}
70               strokeColor={v.strokeColor}
71               strokeWidth={v.strokeWidth}
72             />
73           ))}
74         </MapView>
75       </View>
76     );
77   }
78 }
```

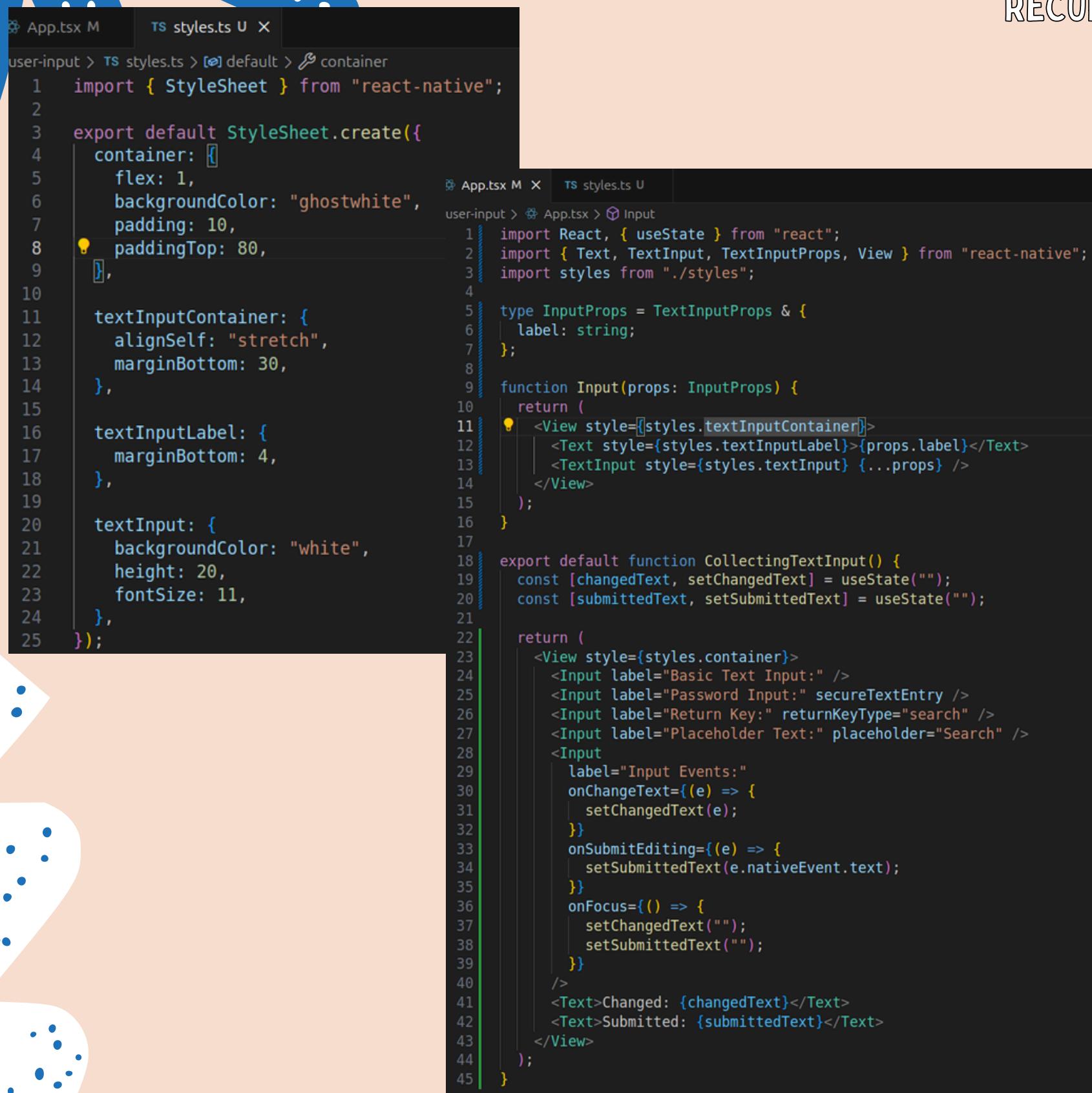
```
geo-and-maps > ts styles.ts > default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   mapView: {
12     alignSelf: "stretch",
13     height: 450,
14     margin: 30,
15   },
16
17   ipaText: {
18     color: "coral",
19   },
20
21   stoutText: {
22     color: "firebrick",
23   },
24
25   boldText: {
26     fontWeight: "bold",
27   },
28});
```



# NPX ET CREATE EXPO APP

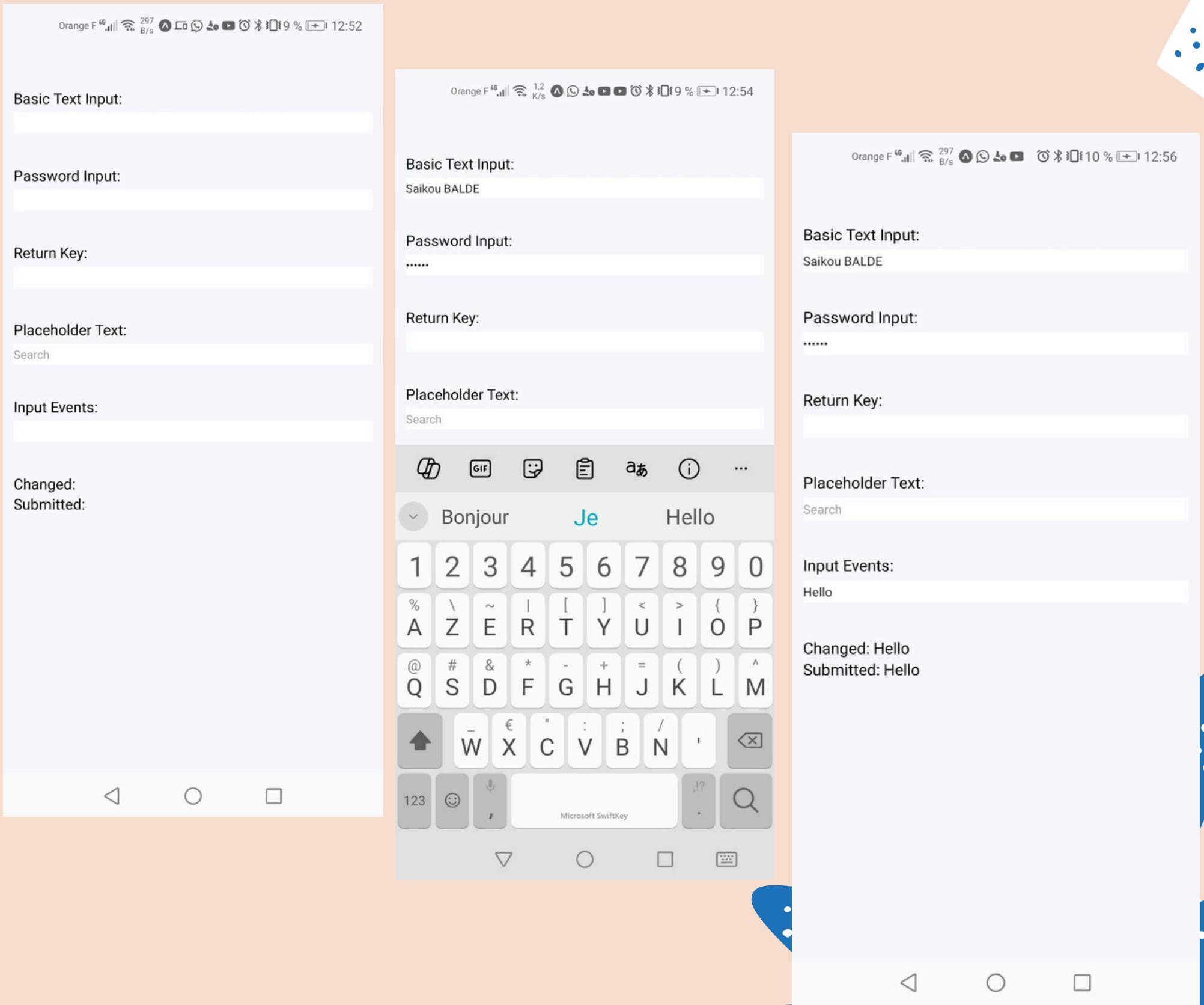
## MANIPULATION DES INPUTS

### RÉCUPÉRATION DE DONNÉES



```
App.tsx M TS styles.ts U X
user-input > TS styles.ts > [o] default > </> container
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: [
5     flex: 1,
6     backgroundColor: "ghostwhite",
7     padding: 10,
8     paddingTop: 80,
9   ],
10
11   textInputContainer: {
12     alignSelf: "stretch",
13     marginBottom: 30,
14   },
15
16   textInputLabel: {
17     marginBottom: 4,
18   },
19
20   textInput: {
21     backgroundColor: "white",
22     height: 20,
23     fontSize: 11,
24   },
25 });
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45

App.tsx M TS styles.ts U
user-input > App.tsx > </> Input
1 import React, { useState } from "react";
2 import { Text, TextInput, TextInputProps, View } from "react-native";
3 import styles from "./styles";
4
5 type InputProps = TextInputProps & {
6   label: string;
7 };
8
9 function Input(props: InputProps) {
10   return (
11     <View style={styles.textInputContainer}>
12       <Text style={styles.textInputLabel}>{props.label}</Text>
13       <TextInput style={styles.textInput} {...props} />
14     </View>
15   );
16 }
17
18 export default function CollectingTextInput() {
19   const [changedText, setChangedText] = useState("");
20   const [submittedText, setSubmittedText] = useState("");
21
22   return (
23     <View style={styles.container}>
24       <Input label="Basic Text Input:" />
25       <Input label="Password Input:" secureTextEntry />
26       <Input label="Return Key:" returnKeyType="search" />
27       <Input label="Placeholder Text:" placeholder="Search" />
28       <Input
29         label="Input Events:"
30         onChangeText={(e) => {
31           setChangedText(e);
32         }}
33         onSubmitEditing={(e) => {
34           setSubmittedText(e.nativeEvent.text);
35         }}
36         onFocus={() => {
37           setChangedText("");
38           setSubmittedText("");
39         }}
40       />
41       <Text>Changed: {changedText}</Text>
42       <Text>Submitted: {submittedText}</Text>
43     </View>
44   );
45 }
```



# NPX ET CREATE EXPO APP

## MANIPULATION DES INPUTS

### SÉLECTION PARMI UNE LISTE D'OPTIONS

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/user-input$ npx expo install @react-native-picker/picker
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

added 1 package, and audited 1194 packages in 4s

131 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
user-input > ts styles.ts > (e) default > ⚡ pickersBlock
  1 import { StyleSheet } from "react-native";
  2
  3 export default StyleSheet.create({
  4   container: {
  5     flex: 1,
  6     flexDirection: "column",
  7     backgroundColor: "ghostwhite",
  8     justifyContent: "center",
  9   },
 10
 11   pickersBlock: [
 12     flex: 2,
 13     flexDirection: "row",
 14     justifyContent: "space-around",
 15     alignItems: "center",
 16   ],
 17
 18   pickerHeight: {
 19     height: 250,
 20   },
 21
 22   pickerContainer: {
 23     flex: 1,
 24     flexDirection: "column",
 25     alignItems: "center",
 26     backgroundColor: "white",
 27     padding: 6,
 28     height: 240,
 29   },
 30
 31   pickerLabel: {
 32     fontSize: 14,
 33     fontWeight: "bold",
 34   },
 35
 36   picker: {
 37     width: 150,
 38     backgroundColor: "white",
 39   },
 40
 41   selection: {
 42     flex: 1,
 43     textAlign: "center",
 44   },
 45 })
```

```
user-input > ⚡ Select.android.tsx > ⚡ Select > ⚡ props.items.map() callback
1  import React from "react";
2  import { View, Text } from "react-native";
3  import { Picker } from "@react-native-picker/picker";
4  import styles from "./styles";
5  import { SelectProps } from "./SelectProps";
6
7  export default function Select(props: SelectProps) {
8    return (
9      <View>
10        <Text style={styles.pickerLabel}>{props.label}</Text>
11        <Picker style={styles.picker} {...props}>
12          {props.items.map((i) => (
13            <Picker.Item key={[i.label]} {...i} />
14          )));
15        </Picker>
16      </View>
17    );
18  }
```

```
user-input > ⚙ Select.ios.tsx > ⚙ Select
1 import React from "react";
2 import { View, Text } from "react-native";
3 import { Picker } from "@react-native-picker/picker";
4 import styles from "./styles";
5 import { SelectProps } from "./SelectProps";
6
7 export default function Select(props: SelectProps) {
8     return (
9         <View style={styles.pickerHeight}>
10            <View style={styles.pickerContainer}>
11                <Text style={styles.pickerLabel}>{props.label}</Text>
12                <Picker style={styles.picker} {...props}>
13                    {props.items.map((i) => (
14                        <Picker.Item key={i.label} {...i} />
15                    ))}
16                </Picker>
17            </View>
18        </View>
19    );
20 }
```

```
user-input > App.tsx > SelectingOptions
1 import React, { useState } from "react";
2 import { View, Text } from "react-native";
3 import styles from "./styles";
4 import Select from "./Select.android";
5
6 const sizes = [
7   { label: "", value: null },
8   { label: "S", value: "S" },
9   { label: "M", value: "M" },
10  { label: "L", value: "L" },
11  { label: "XL", value: "XL" },
12];
13
14 const garments = [
15  { label: "", value: null, sizes: ["S", "M", "L", "XL"] },
16  { label: "Socks", value: 1, sizes: ["S", "L"] },
17  { label: "Shirt", value: 2, sizes: ["M", "XL"] },
18  { label: "Pants", value: 3, sizes: ["S", "L"] },
19  { label: "Hat", value: 4, sizes: ["M", "XL"] },
20];
21
22 export default function SelectingOptions() {
23  const [availableGarments, setAvailableGarments] = useState<typeof garments>(
24    []
25  );
26  const [selectedSize, setSelectedSize] = useState<string | null>(null);
27  const [selectedGarment, setSelectedGarment] = useState<number | null>(null);
28
29  return (
30    <View style={styles.container}>
31      <View style={styles.pickersBlock}>
32        <Select
33          label="Size"
34          items={sizes}
35          selectedValue={selectedSize}
36          onValueChange={(size: string) => {
37            setSelectedSize(size);
38            setSelectedGarment(null);
39            setAvailableGarments(
40              garments.filter((i) => i.sizes.includes(size))
41            );
42          }}
43        />
44        <Select
45          label="Garment"
46          items={availableGarments}
47          selectedValue={selectedGarment}
48          onValueChange={(garment: number) => {
```

```
user-input > App.tsx > SelectingOptions
22 export default function SelectingOptions() {
42     }
43     </>
44     <Select
45         label="Garment"
46         items={availableGarments}
47         selectedValue={selectedGarment}
48         onChange={(garment: number) => {
49             setSelectedGarment(garment);
50         }}
51     >
52     </View>
53     <Text style={styles.selection}>
54         {selectedSize &&
55             selectedGarment &&
56             `${selectedSize} ${
57                 garments.find((i) => i.value === selectedGarment)?.label
58             }`}
59     </Text>
60     </View>
61 );
62 }
```

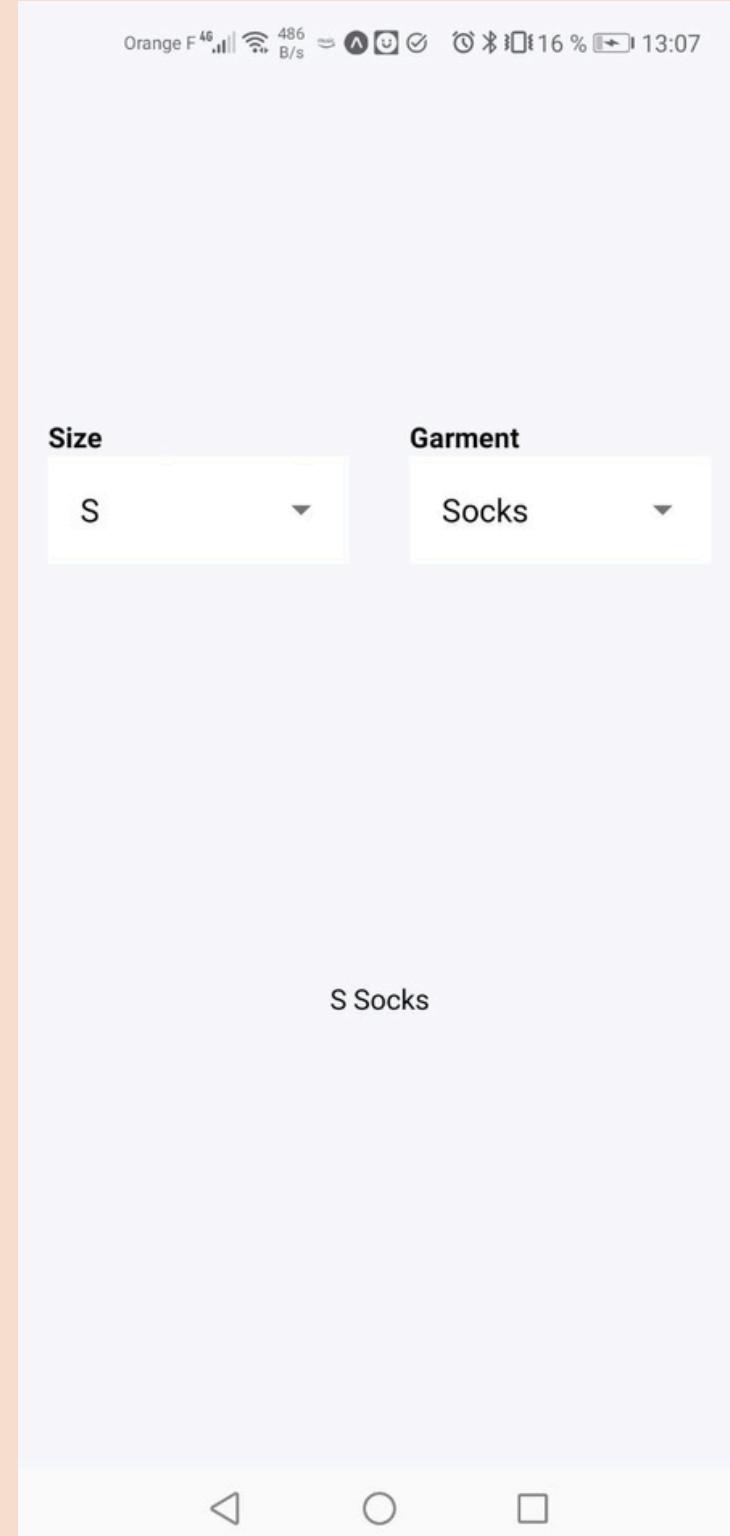
# NPX ET CREATE EXPO APP

## MANIPULATION DES INPUTS

### SÉLECTION PARMI UNE LISTE D'OPTIONS

```
user-input > App.tsx > SelectingOptions
1 import React, { useState } from "react";
2 import { View, Text } from "react-native";
3 import styles from "./styles";
4 import Select from "./Select.android";
5
6 const sizes = [
7   { label: "", value: null },
8   { label: "S", value: "S" },
9   { label: "M", value: "M" },
10  { label: "L", value: "L" },
11  { label: "XL", value: "XL" },
12];
13
14 const garments = [
15  { label: "", value: null, sizes: ["S", "M", "L", "XL"] },
16  { label: "Socks", value: 1, sizes: ["S", "L"] },
17  { label: "Shirt", value: 2, sizes: ["M", "XL"] },
18  { label: "Pants", value: 3, sizes: ["S", "L"] },
19  { label: "Hat", value: 4, sizes: ["M", "XL"] },
20];
21
22 export default function SelectingOptions() {
23  const [availableGarments, setAvailableGarments] = useState<typeof garments>(
24    []
25  );
26  const [selectedSize, setSelectedSize] = useState<string | null>(null);
27  const [selectedGarment, setSelectedGarment] = useState<number | null>(null);
28
29  return [
30    <View style={styles.container}>
31      <View style={styles.pickersBlock}>
32        <Select
33          label="Size"
34          items={sizes}
35          selectedValue={selectedSize}
36          onValueChange={(size: string) => {
37            setSelectedSize(size);
38            setSelectedGarment(null);
39            setAvailableGarments(
40              garments.filter((i) => i.sizes.includes(size))
41            );
42          }}
43        />
44        <Select
45          label="Garment"
46          items={availableGarments}
47          selectedValue={selectedGarment}
48          onValueChange={(garment: number) => {
49            setSelectedGarment(garment);
50          }}
51        />
52      </View>
53      <Text style={styles.selection}>
54        {selectedSize &&
55        selectedGarment &&
56        `${selectedSize} ${{
57          garments.find((i) => i.value === selectedGarment)?.label
58        }}`}
59      </Text>
60    </View>
61  ];
62}
```

```
user-input > App.tsx > SelectingOptions
22  export default function SelectingOptions() {
23    <View style={styles.container}>
24      <View style={styles.pickersBlock}>
25        <Select
26          label="Size"
27          items={sizes}
28          selectedValue={selectedSize}
29          onValueChange={(size: string) => {
30            setSelectedSize(size);
31            setSelectedGarment(null);
32            setAvailableGarments(
33              garments.filter((i) => i.sizes.includes(size))
34            );
35          }}
36        />
37        <Select
38          label="Garment"
39          items={availableGarments}
40          selectedValue={selectedGarment}
41          onValueChange={(garment: number) => {
42            setSelectedGarment(garment);
43          }}
44        />
45      </View>
46      <Text style={styles.selection}>
47        {selectedSize &&
48        selectedGarment &&
49        `${selectedSize} ${{
50          garments.find((i) => i.value === selectedGarment)?.label
51        }}`}
52      </Text>
53    </View>
54  };
55}
```



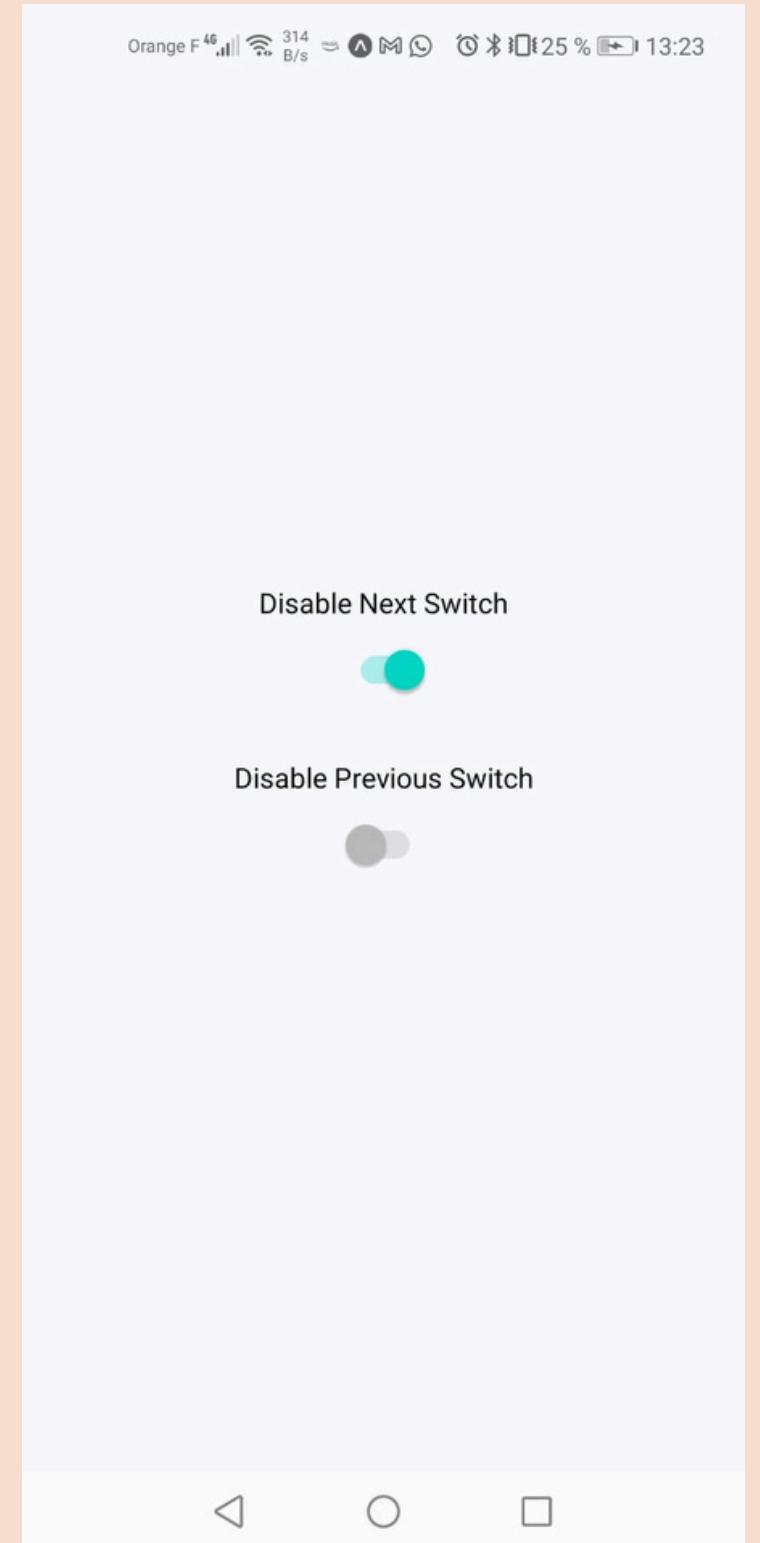
# NPX ET CREATE EXPO APP

## MANIPULATION DES INPUTS

### BASCULER ENTRE ACTIVÉ ET DÉSACTIVÉ

```
user-input > ts styles.ts > [o] default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create([
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   customSwitch: {
12     alignItems: "center",
13     margin: 10,
14   },
15 ]); user-input > S Switch.tsx > ⚡ CustomSwitch
1 import React from "react";
2 import { View, Text, Switch, SwitchProps } from "react-native";
3 import styles from "./styles";
4
5 type CustomSwitchProps = SwitchProps & {
6   label: string;
7 };
8
9 export default function CustomSwitch(props: CustomSwitchProps) {
10   return (
11     <View style={styles.customSwitch}>
12       <Text>{props.label}</Text>
13       <Switch {...props} />
14     </View>
15   );
16 }
```

```
user-input > S App.tsx > ⚡ TogglingOnAndOff
1 import React, { useState } from "react";
2 import { View } from "react-native";
3 import styles from "./styles";
4 import Switch from "./Switch";
5
6 export default function TogglingOnAndOff() {
7   const [first, setFirst] = useState(false);
8   const [second, setSecond] = useState(false);
9
10  return (
11    <View style={styles.container}>
12      <Switch
13        label="Disable Next Switch"
14        value={first}
15        disabled={second}
16        onValueChange={setFirst}
17      />
18      <Switch
19        label="Disable Previous Switch"
20        value={second}
21        disabled={first}
22        onValueChange={setSecond}
23      />
24    </View>
25  );
26 }
```



# NPX ET CREATE EXPO APP

## MANIPULATION DES INPUTS

### COLLECTE DE LA DATE/HEURE

```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/user-input$ npx expo install @react-native-community/datetimepicker
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

added 1 package, and audited 1195 packages in 4s

131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
user-input > ts styles.ts > default > datePickerLabel
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   datePickerContainer: {
12     width: 260,
13     margin: 20,
14   },
15
16   datePickerLabel: {
17     fontSize: 18,
18     fontWeight: "bold",
19   },
20});
```

```
user-input > ts TimePickerProps.ts > TimePickerProps
1 export type TimePickerProps = {
2   label: string;
3   value: Date;
4   onChange: (date: Date) => void;
5 };
```

```
user-input > ts DatePickerProps.ts > DatePickerProps
1 export type DatePickerProps = {
2   label: string;
3   value: Date;
4   onChange: (date: Date) => void;
5 };
```

```
user-input > TimePicker.android.tsx > TimePicker
1 import React from "react";
2 import { Text, View } from "react-native";
3 import { DateTimePickerAndroid } from "@react-native-community/datetimepicker";
4 import styles from "./styles";
5 import { TimePickerProps } from "./TimePickerProps";
6
7 export default function TimePicker({
8   label,
9   value,
10  onChange,
11  }: TimePickerProps) {
12  return [
13    <View style={styles.datePickerContainer}>
14      <Text style={styles.datePickerLabel}>{label}</Text>
15      <Text
16        onPress={() =>
17          DateTimePickerAndroid.open({
18            value: value,
19            mode: "time",
20            is24Hour: true,
21            onChange: (event, date) => {
22              if (event.type === "set" && date) {
23                onChange(date);
24              }
25            },
26          })
27        >
28          {value.toLocaleTimeString()}
29        </Text>
30      </View>
31    ];
32}
```

```
user-input > DatePicker.android.tsx > DatePicker > <function>
1 import React from "react";
2 import { Text, View } from "react-native";
3 import { DateTimePickerAndroid } from "@react-native-community/datetimepicker";
4 import styles from "./styles";
5 import { DatePickerProps } from "./DatePickerProps";
6
7 export default function DatePicker({
8   label,
9   value,
10  onChange,
11  }: DatePickerProps) {
12  return (
13    <View style={styles.datePickerContainer}>
14      <Text style={styles.datePickerLabel}>{label}</Text>
15      <Text
16        onPress={() => {
17          DateTimePickerAndroid.open({
18            value: value,
19            mode: "date",
20            onChange: (event, date) => {
21              if (event.type === "set" && date) {
22                onChange(date);
23              }
24            },
25          });
26        >
27          {value.toLocaleDateString()}
28        </Text>
29      </View>
30    );
31}
```

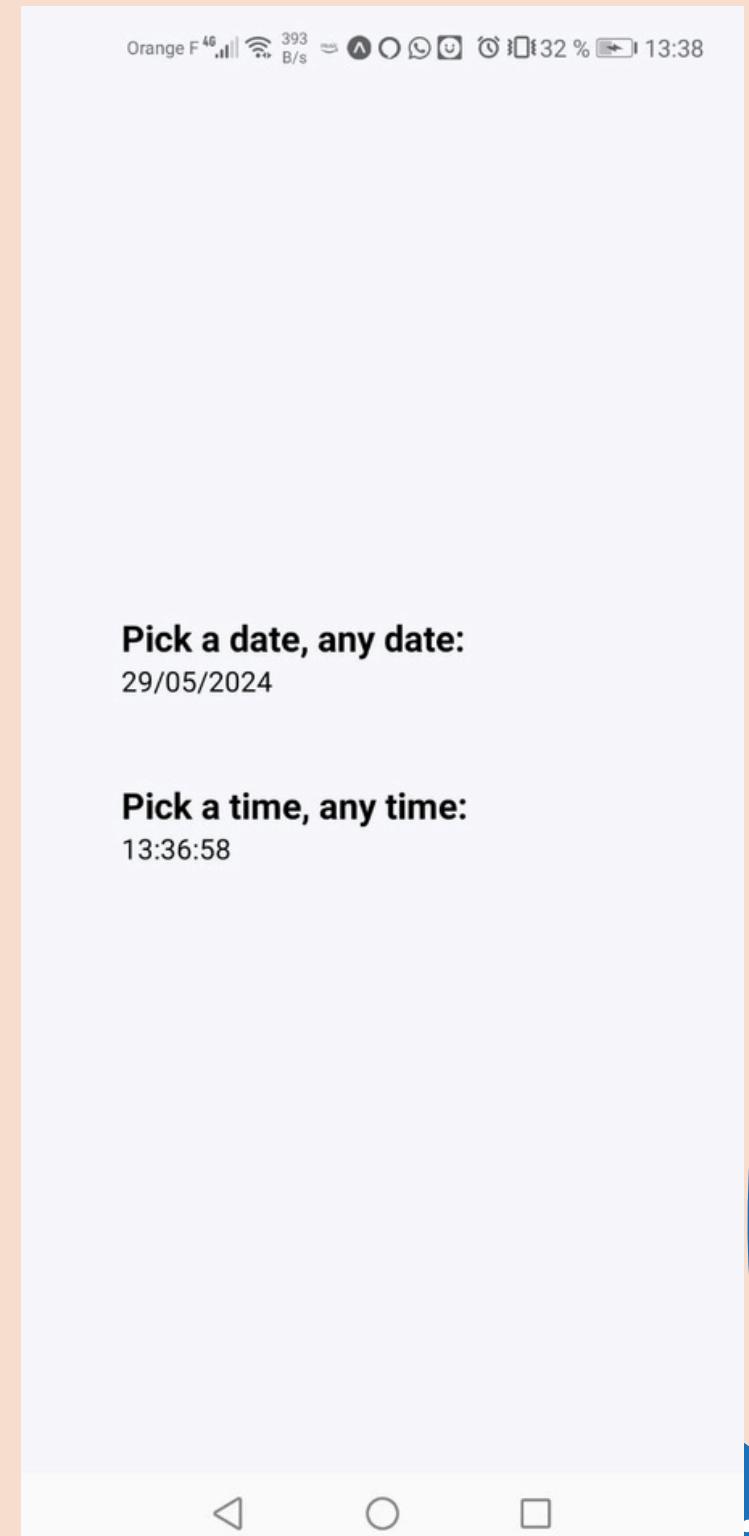
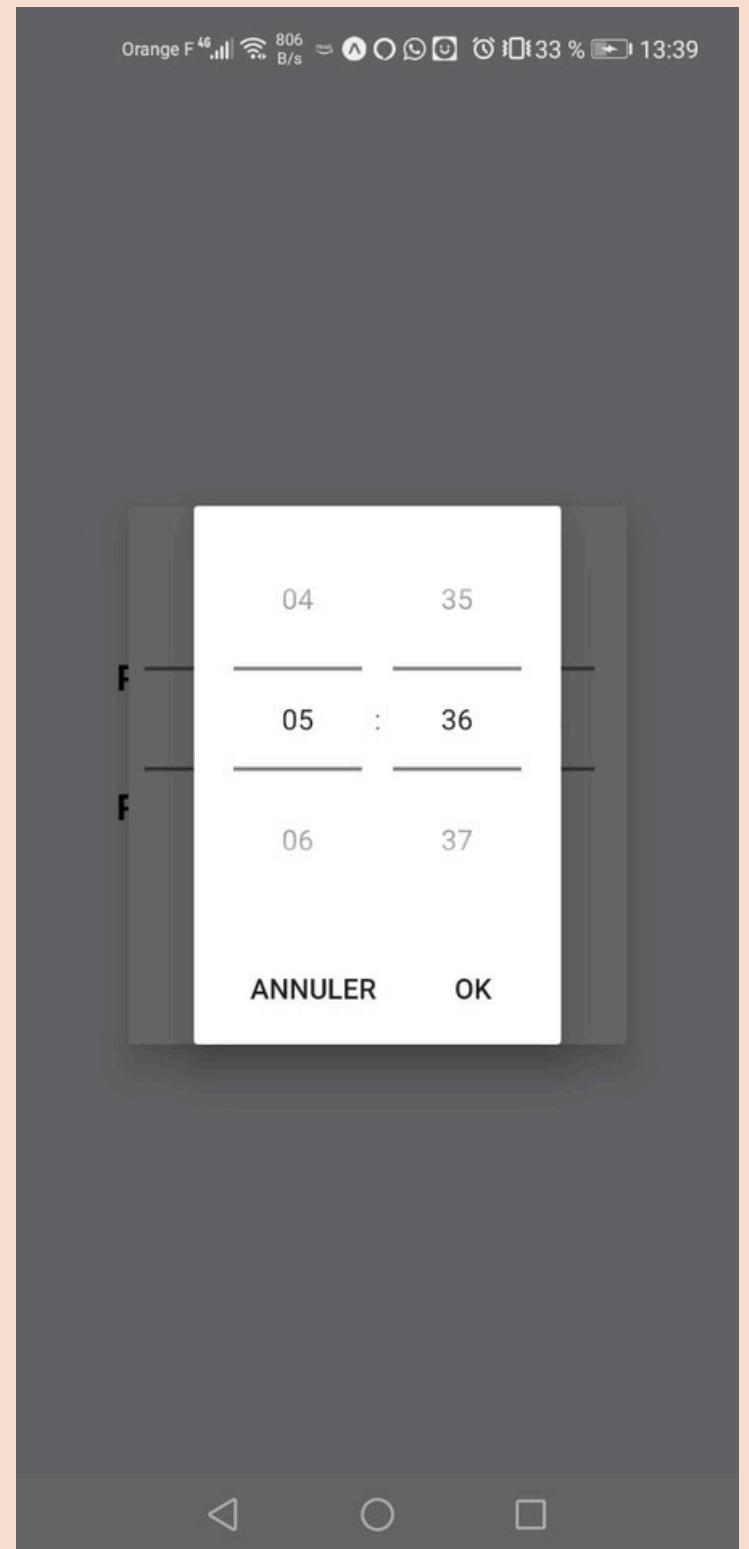
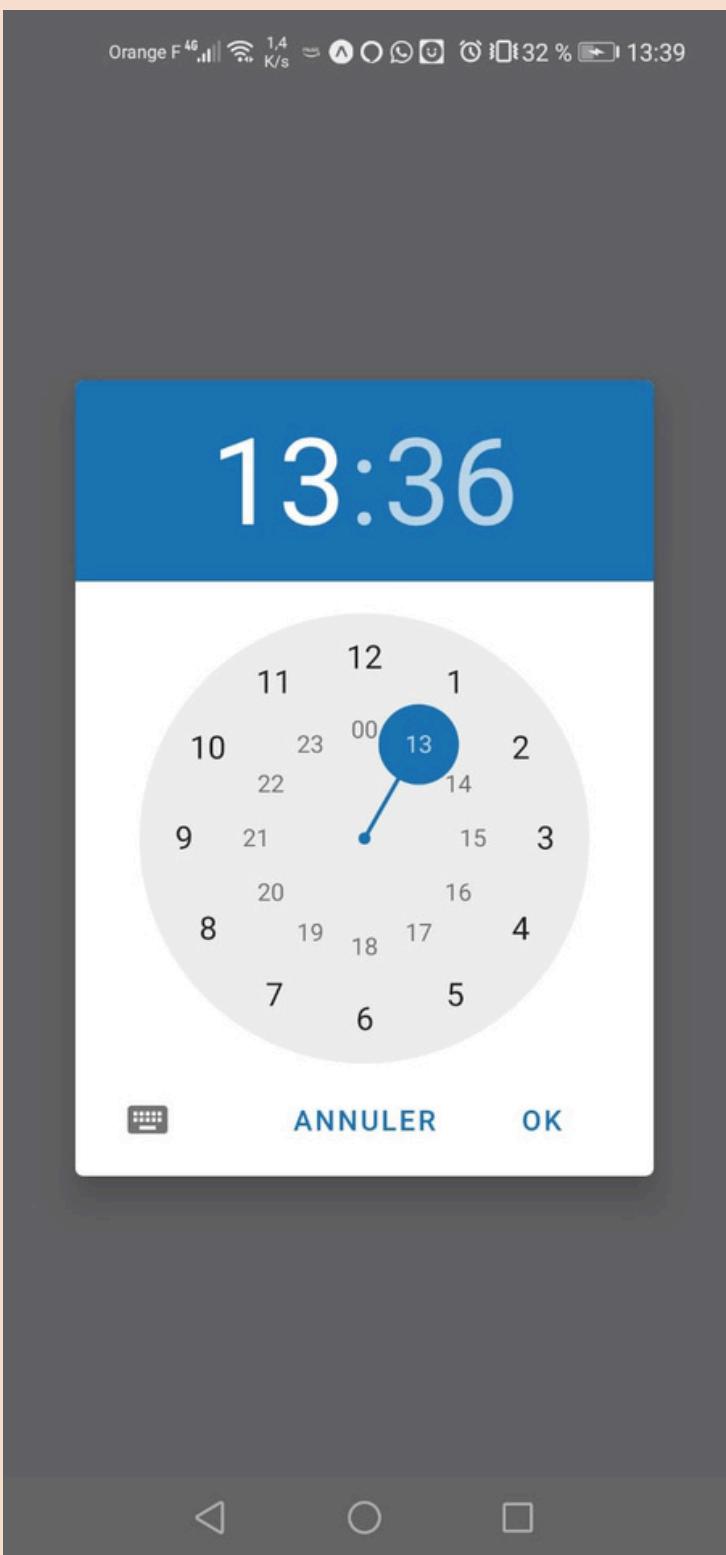
```
user-input > App.tsx > CollectingDateTimeInput
1 import React, { useState } from "react";
2 import { View } from "react-native";
3 import DatePicker from "./DatePicker";
4 import TimePicker from "./TimePicker";
5 import styles from "./styles";
6
7 export default function CollectingDateTimeInput() {
8  const [date, setDate] = useState(new Date());
9  const [time, setTime] = useState(new Date());
10
11  return (
12    <View style={styles.container}>
13      <DatePicker
14        label="Pick a date, any date:"
15        value={date}
16        onChange={setDate}
17      />
18      <TimePicker
19        label="Pick a time, any time:"
20        value={time}
21        onChange={setTime}
22      />
23    </View>
24  );
25}
```

# NPX ET CREATE EXPO APP

## MANIPULATION DES INPUTS

### COLLECTE DE LA DATE/HEURE

```
user-input > App.tsx > ...  
  
1 | import React, { useState } from "react";
2 | import { View } from "react-native";
3 | import DatePicker from "./DatePicker.android";
4 | import TimePicker from "./TimePicker.android";
5 | import styles from "./styles";
6 |
7 | export default function CollectingDateTimeInput() {
8 |   const [date, setDate] = useState(new Date());
9 |   const [time, setTime] = useState(new Date());
10 |
11 |   return (
12 |     <View style={styles.container}>
13 |       <DatePicker
14 |         label="Pick a date, any date:"
15 |         value={date}
16 |         onChange={setDate}
17 |       />
18 |       <TimePicker
19 |         label="Pick a time, any time:"
20 |         value={time}
21 |         onChange={setTime}
22 |       />
23 |     </View>
24 |   );
25 | }
```



**Pick a date, any date:**  
29/05/2024

**Pick a time, any time:**  
13:36:58

# NPX ET CREATE EXPO APP

## RÉPONDRE AUX GESTES DES UTILISATEURS

### FAIRE DÉFILER AVEC LES DOIGTS

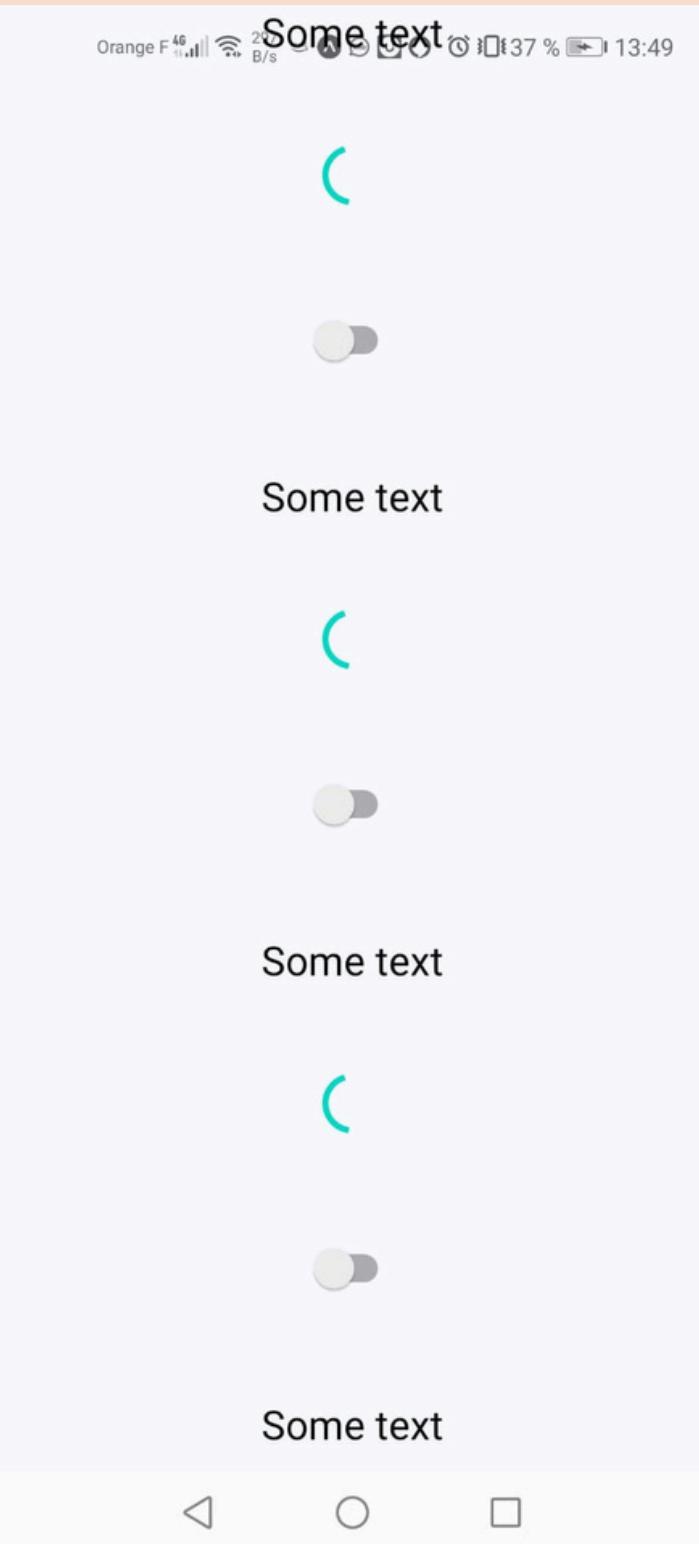


Your project is ready!

To run your project, navigate to the directory and run one of the following npm commands.

```
- cd user-gesture  
- npm run android  
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac  
- npm run web
```

```
user-gesture > ts styles.ts > default > container  
1 import { StyleSheet } from "react-native";  
2  
3 export default StyleSheet.create({  
4   container: {  
5     flex: 1,  
6     justifyContent: "center",  
7     alignItems: "center",  
8     backgroundColor: "ghostwhite",  
9   },  
10  
11   text: {  
12     fontSize: 20,  
13     textAlign: "center",  
14   },  
15  
16   scroll: {  
17     height: 1,  
18     alignSelf: "stretch",  
19   },  
20  
21   scrollItem: {  
22     margin: 20,  
23     alignSelf: "center",  
24   },  
25 });  
  
user-gesture > App.tsx > ...  
1 import React from "react";  
2 import {  
3   Text,  
4   ScrollView,  
5   ActivityIndicator,  
6   Switch,  
7   View,  
8 } from "react-native";  
9  
9 import styles from "./styles";  
10  
12 export default function App() {  
13   return (  
14     <View style={styles.container}>  
15       <ScrollView style={styles.scroll}>  
16         {new Array(20).fill(null).map((v, i) => (  
17           <View key={i}>  
18             <Text style={[styles.scrollItem, styles.text]}>Some text</Text>  
19             <ActivityIndicator style={styles.scrollItem} size="large" />  
20             <Switch style={styles.scrollItem} />  
21           </View>  
22         ))}  
23       </ScrollView>  
24     </View>  
25   );  
26 }
```



# NPX ET CREATE EXPO APP

## RÉPONDRE AUX GESTES DES UTILISATEURS

### ÉVÈNEMENT TOUCH TACTILES

```
user-gesture > ts styles.ts > default > container

1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: [
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   ],
10
11   button: {
12     padding: 10,
13     margin: 5,
14     backgroundColor: "azure",
15     borderWidth: 1,
16     borderRadius: 4,
17     borderColor: "slategrey",
18   },
19
20   buttonText: {
21     color: "slategrey",
22   },
23 });


```

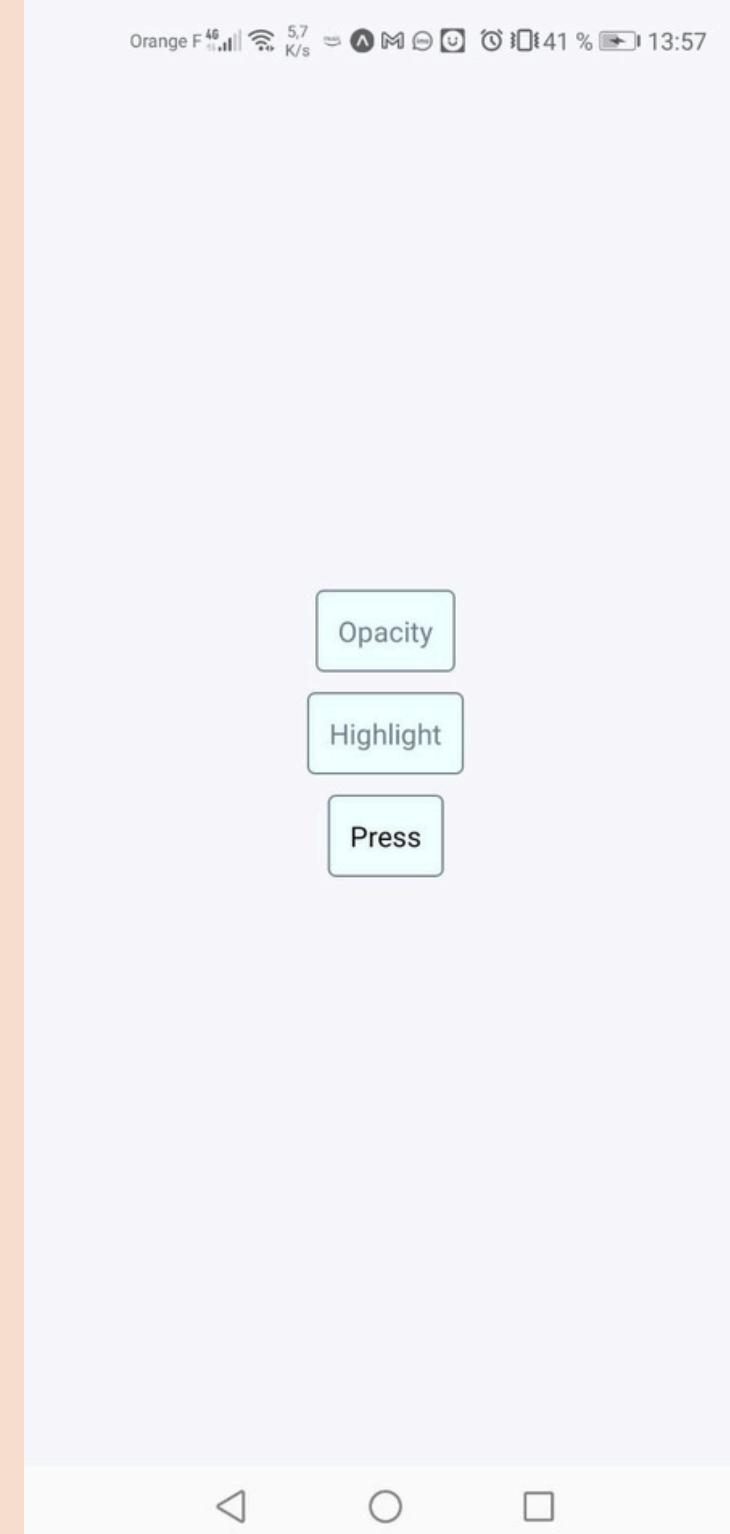
```
user-gesture > ts PressableButton.tsx > PressableButton

1 import React, { useState } from "react";
2 import { Pressable, Text } from "react-native";
3 import styles from "./styles";
4
5 const PressableButton = () => [
6   const [text, setText] = useState("Not Pressed");
7
8   return (
9     <Pressable
10       onPress={() => setText("Pressed")}
11       onPressOut={() => setText("Press")}
12       onLongPress={() => {
13         setText("Long Pressed");
14       }}
15       delayLongPress={500}
16       style={({ pressed }) => [
17         {
18           opacity: pressed ? 0.5 : 1,
19         },
20         styles.button,
21       ]}
22     >
23     | <Text>{text}</Text>
24     </Pressable>
25   );
26 ]
27
28 export default PressableButton;
```

```
user-gesture > ts App.tsx > App

1 import React from "react";
2 import { View } from "react-native";
3 import styles from "./styles";
4 import { OpacityButton, HighlightButton } from "./Button";
5 import PressableButton from "./PressableButton";
6
7 export default function App() {
8   return (
9     <View style={styles.container}>
10      | <OpacityButton onPress={() => {}} label="Opacity" />
11      | <HighlightButton onPress={() => {}} label="Highlight" />
12      | <PressableButton />
13    </View>
14  );
15 }
```

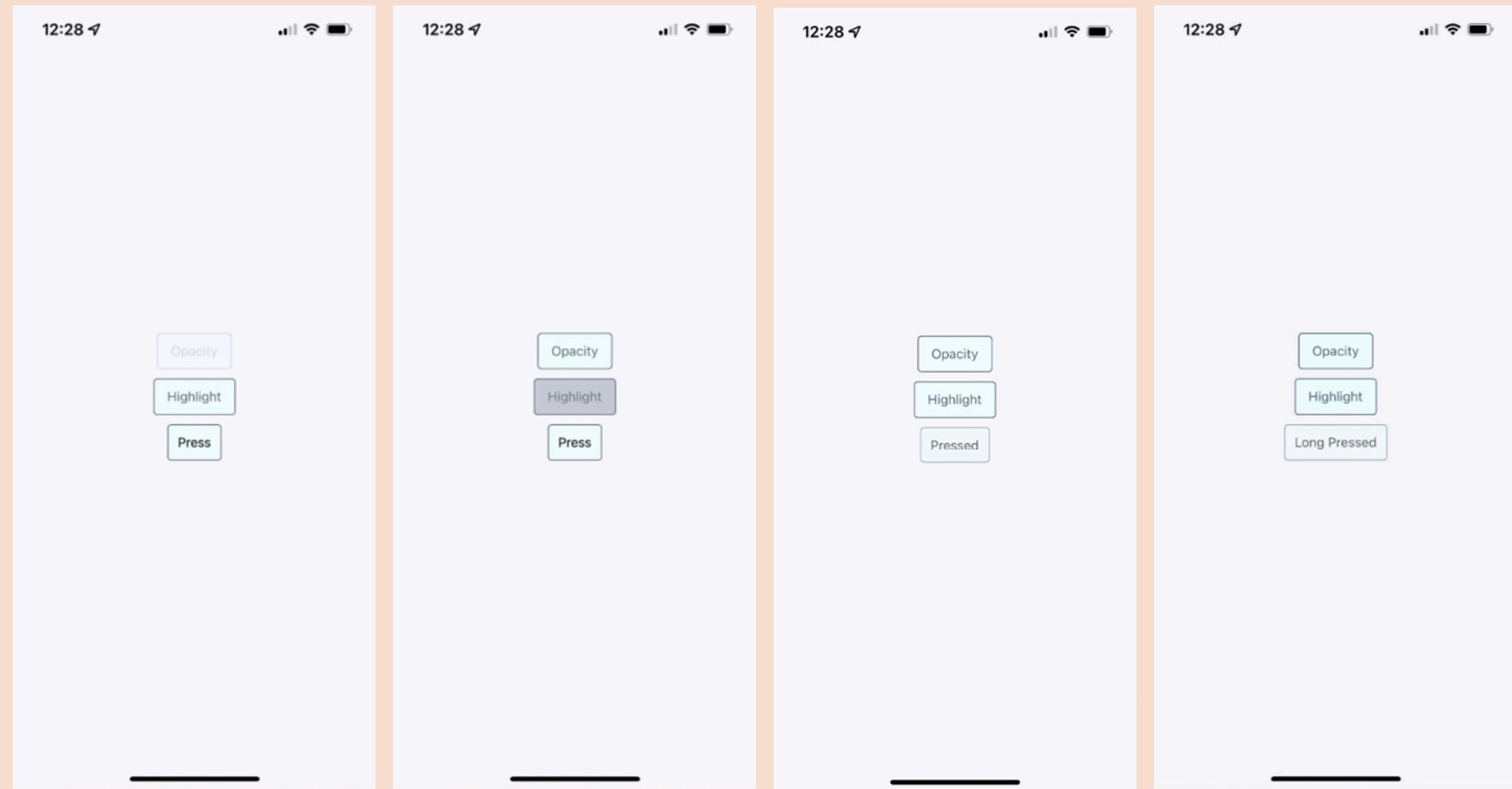
```
user-gesture > ts Button.tsx > ...
1 import React from "react";
2 import { Text, TouchableOpacity, TouchableHighlight } from "react-native";
3 import styles from "./styles";
4
5 type ButtonProps = {
6   label: string;
7   onPress: () => void;
8 };
9
10 export const OpacityButton = ({ label, onPress }: ButtonProps) => {
11   return (
12     <TouchableOpacity
13       style={styles.button}
14       onPress={onPress}
15       activeOpacity={0.5}
16     >
17     | <Text style={styles.buttonText}>{label}</Text>
18     </TouchableOpacity>
19   );
20 }
21
22 export const HighlightButton = ({ label, onPress }: ButtonProps) => {
23   return (
24     <TouchableHighlight
25       style={styles.button}
26       underlayColor="rgba(112,128,144,0.3)"
27       onPress={onPress}
28     >
29     | <Text style={styles.buttonText}>{label}</Text>
30     </TouchableHighlight>
31   );
32 }
```



# NPX ET CREATE EXPO APP

## RÉPONDRE AUX GESTES DES UTILISATEURS

### ÉVÈNEMENT TOUCH TACTILES



# NPX ET CREATE EXPO APP

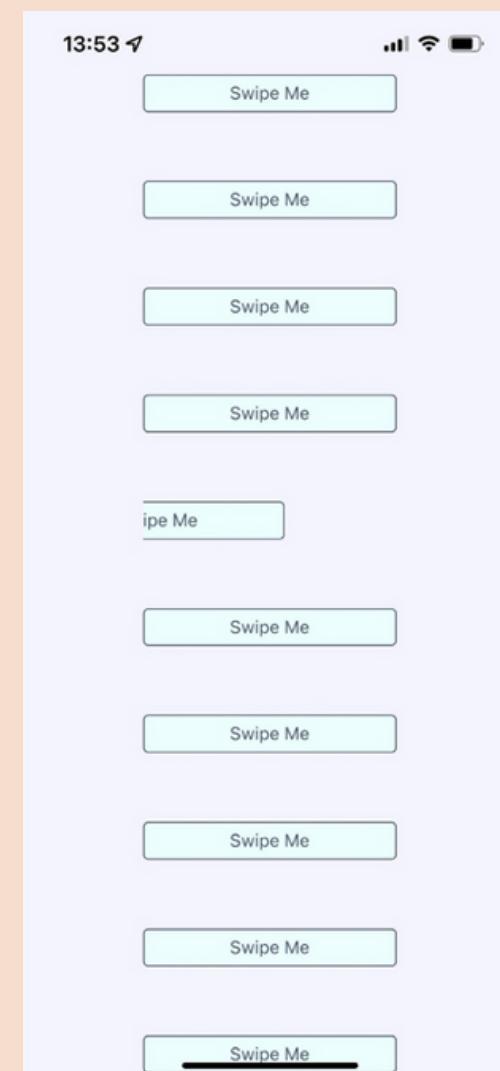
## RÉPONDRE AUX GESTES DES UTILISATEURS

UTILISATION DE COMPOSANTS GLISSANTS ET ANNULABLES

```
user-gesture > ts styles.ts > [o] default > ↵ swipeBlank
  3  export default StyleSheet.create({
  4    container: {
  5      flex: 1,
  6      justifyContent: "space-between",
  7      alignItems: "center",
  8      backgroundColor: "ghostwhite",
  9    },
 10
 11   swipeContainer: {
 12     flex: 1,
 13     flexDirection: "row",
 14     width: 200,
 15     height: 30,
 16     marginTop: 50,
 17   },
 18
 19   swipeItem: {
 20     height: 30,
 21     backgroundColor: "azure",
 22     justifyContent: "center",
 23     borderwidth: 1,
 24     borderRadius: 4,
 25     borderColor: "slategrey",
 26   },
 27
 28   swipeItemText: {
 29     textAlign: "center",
 30     color: "slategrey",
 31   },
 32
 33   swipeBlank: [
 34     {
 35       height: 30,
 36     },
 37   ];
 38 }
```

```
user-gesture > ↵ Swipeable.tsx > ...
  1  import React from "react";
  2  import [
  3    View,
  4    ScrollView,
  5    Text,
  6    NativeSyntheticEvent,
  7    NativeScrollEvent,
  8  ] from "react-native";
  9  import styles from "./styles";
 10
 11 type SwipeableProps = {
 12   name: string;
 13   width: number;
 14   onSwipe: () => void;
 15 };
 16
 17 export default function Swipeable({ name, width, onSwipe }: SwipeableProps) {
 18   function onScroll(e: NativeSyntheticEvent<NativeScrollEvent>) {
 19     console.log(e.nativeEvent.contentOffset.x);
 20     e.nativeEvent.contentOffset.x >= width && onSwipe();
 21   }
 22
 23   return (
 24     <View style={styles.swipeContainer}>
 25       <ScrollView
 26         horizontal
 27         snapToInterval={width}
 28         showsHorizontalScrollIndicator={false}
 29         scrollEventThrottle={10}
 30         onScroll={onScroll}
 31       >
 32         <View style={[styles.swipeItem, { width }]}>
 33           <Text style={styles.swipeItemText}>{name}</Text>
 34         </View>
 35
 36         <View style={[styles.swipeBlank, { width }]} />
 37       </ScrollView>
 38     </View>
 39   );
 40 }
```

```
user-gesture > ↵ App.tsx > ↵ SwipableAndCancellable
  1  import React, { useState } from "react";
  2  import { View } from "react-native";
  3  import styles from "./styles";
  4  import Swipeable from "./Swipeable";
  5
  6  type SwipeableProps = {
  7    name: string;
  8    width: number;
  9    onSwipe: () => void;
 10  };
 11
 12 export default function SwipableAndCancellable() {
 13   const [items, setItems] = useState(
 14     new Array(10).fill(null).map((v, id) => ({ id, name: "Swipe Me" }))
 15   );
 16
 17   function onSwipe(id: number) {
 18     return () => {
 19       setItems(items.filter((item) => item.id !== id));
 20     };
 21   }
 22
 23   return (
 24     <View style={[styles.container]}>
 25       {items.map((item) => (
 26         <Swipeable
 27           key={item.id}
 28           onSwipe={onSwipe(item.id)}
 29           name={item.name}
 30           width={200}
 31         >
 32           <View style={[styles.swipeItem, { width }]}>
 33             <Text style={styles.swipeItemText}>{item.name}</Text>
 34           </View>
 35         </Swipeable>
 36       ))
 37     </View>
 38   );
 39 }
```

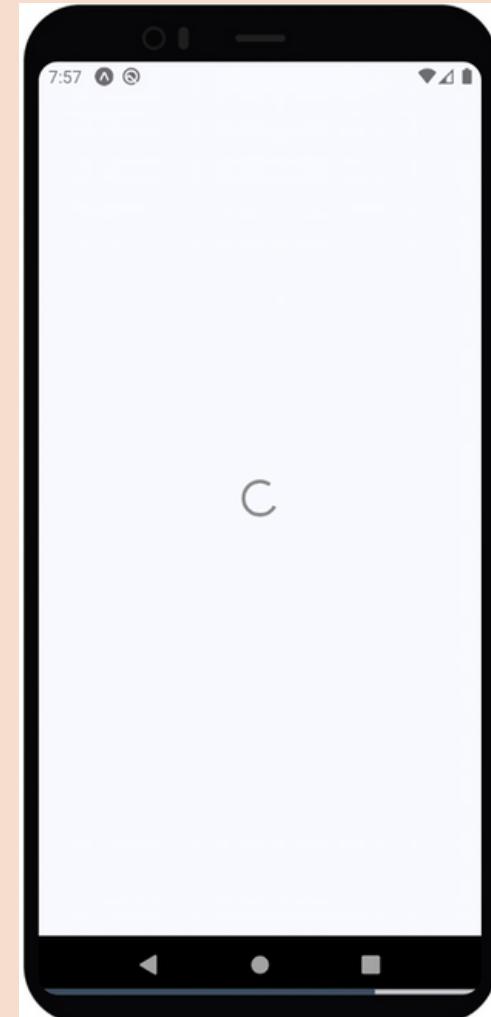
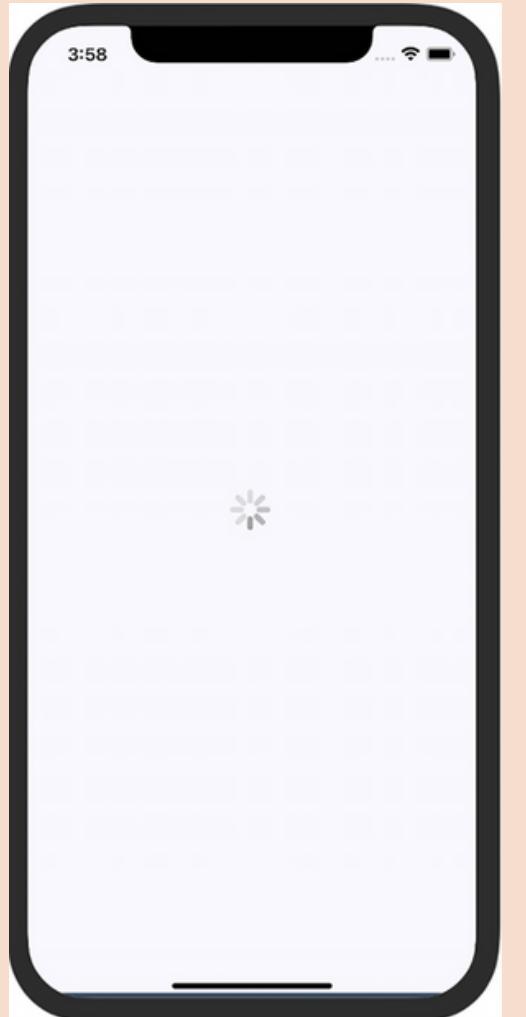


# NPX ET CREATE EXPO APP

## LA BARRE DE PROGRESSION

### INDIQUER LES PROGRÈS

```
progress-bar > ts styles.ts > [o] default
1  import { StyleSheet } from "react-native";
2
3  export default StyleSheet.create({
4    container: {
5      flex: 1,
6      justifyContent: "center",
7      alignItems: "center",
8      backgroundColor: "ghostwhite",
9    },
10 });
progress-bar > App.tsx > ...
1  import React from "react";
2  import { View, ActivityIndicator } from "react-native";
3  import styles from "./styles";
4  export default function App() {
5    return (
6      <View style={styles.container}>
7        <ActivityIndicator size="large" />
8      </View>
9    );
10 }
```



# NPX ET CREATE EXPO APP

## LA BARRE DE PROGRESSION

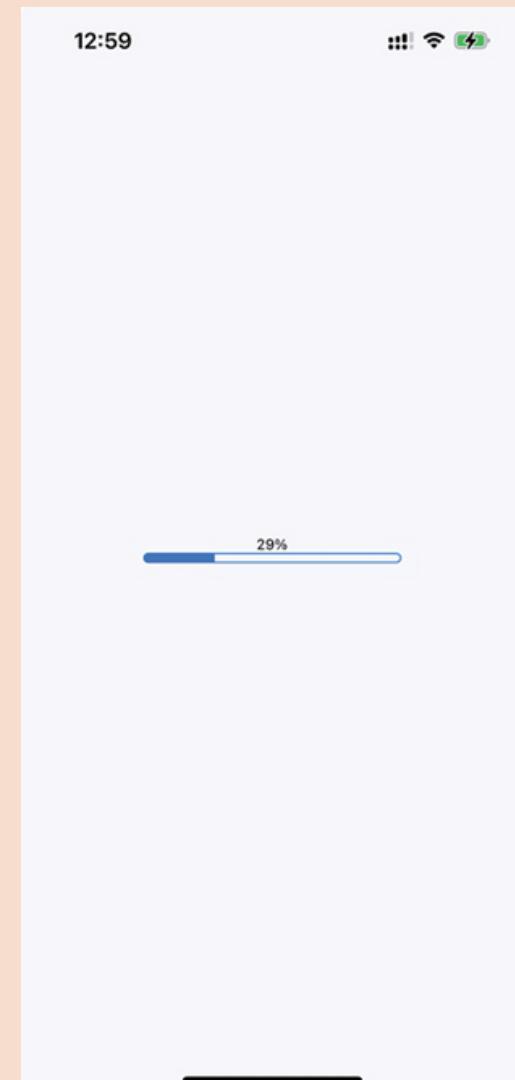
### MESURER L'EVOLUTION DE LA BARRE DE PROGRESSION

```
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/progress-bar$ npm i react-native-progress
added 14 packages, and audited 1207 packages in 7s
139 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
progress-bar > ts styles.ts > default
1  import { StyleSheet } from "react-native";
2
3  export default StyleSheet.create({
4    container: {
5      flex: 1,
6      justifyContent: "center",
7      alignItems: "center",
8      backgroundColor: "ghostwhite",
9    },
10
11   progress: {
12     width: 200,
13   },
14
15   progressText: {
16     fontSize: 11,
17     textAlign: "center",
18   },
19 });

progress-bar > ProgressBar.tsx > ProgressBar
1  import React from "react";
2  import { View, Text } from "react-native";
3  import styles from "./styles";
4  import * as Progress from "react-native-progress";
5
6  type ProgressBarProps = {
7    progress: number;
8  };
9
10
11  export default function ProgressBar({ progress }: ProgressBarProps) {
12    return (
13      <View style={styles.progress}>
14        <Text style={styles.progressText}>{Math.round(progress * 100)}%</Text>
15        <Progress.Bar width={200} useNativeDriver={true} progress={progress} />
16      </View>
17    );
18}
```

```
progress-bar > App.tsx > MeasuringProgress > useEffect() callback > updateProgress()
1  import React, { useState, useEffect } from "react";
2  import { View } from "react-native";
3  import styles from "./styles";
4  import ProgressBar from "./ProgressBar";
5
6  export default function MeasuringProgress() {
7    const [progress, setProgress] = useState(0);
8
9    useEffect(() => {
10      let timeoutRef: NodeJS.Timeout | null = null;
11
12      function updateProgress() {
13        setProgress((currentProgress) => {
14          if (currentProgress < 1) {
15            return currentProgress + 0.01;
16          } else {
17            return 0;
18          }
19        });
20
21        timeoutRef = setTimeout(updateProgress, 100);
22      }
23
24      updateProgress();
25
26      return () => {
27        timeoutRef && clearTimeout(timeoutRef);
28      };
29    }, []);
30
31    return (
32      <View style={styles.container}>
33        <ProgressBar progress={progress} />
34      </View>
35    );
36}
```



# NPX ET CREATE EXPO APP

## LA BARRE DE PROGRESSION

### BARRE DE PROGRESSION PAR ÉTAPE

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/progress-bar$ npm i @react-navigation/native @react-navigation/native-stack
added 18 packages, and audited 1225 packages in 7s
140 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
progress-bar > styles.ts > default > container
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: [
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   ],
10
11   nav: {
12     backgroundColor: "lightgray",
13   },
14
15   title: {
16     flex: 1,
17     fontSize: 16,
18     fontWeight: "bold",
19   },
20
21   content: {
22     fontSize: 23,
23     fontWeight: "bold",
24   },
25
26   progress: {
27     justifyContent: "center",
28     alignItems: "center",
29     padding: 1,
30   },
31 });

```

```
progress-bar > First.tsx > First
1 import React from "react";
2 import { View, Text } from "react-native";
3 import styles from "./styles";
4
5 export default function First() {
6   return (
7     <View style={styles.container}>
8       <Text style={styles.content}>First Content</Text>
9     </View>
10   );
11 }
```

```
progress-bar > Second.tsx > Second
1 import React from "react";
2 import { View, Text } from "react-native";
3 import styles from "./styles";
4 import ProgressBar from "./ProgressBar";
5
6 export default function Second() {
7   return [
8     <View style={styles.container}>
9       <Text style={styles.content}>Second Content</Text>
10     </View>
11   ];
12 }
```

```
progress-bar > Third.tsx > Third
1 import React from "react";
2 import { View, Text } from "react-native";
3 import styles from "./styles";
4 import ProgressBar from "./ProgressBar";
5
6 export default function Third() {
7   return (
8     <View style={styles.container}>
9       <Text style={styles.content}>Third Content</Text>
10     </View>
11   );
12 }
```

```
progress-bar > router.ts > Routes
1 export type Routes = {
2   First: { progress: number };
3   Second: { progress: number };
4   Third: { progress: number };
5   Fourth: { progress: number };
6 };

```

```
progress-bar > App.tsx > routes
1 import React from "react";
2 import { View, Text, Button } from "react-native";
3 import { NavigationContainer } from "@react-navigation/native";
4 import { createStackNavigator } from "@react-navigation/native-stack";
5 import First from "./First";
6 import Second from "./Second";
7 import Third from "./Third";
8 import Fourth from "./Fourth";
9 import ProgressBar from "./ProgressBar";
10 import styles from "./styles";
11 import { Routes } from "./router";
12
13 const Stack = createStackNavigator<Routes>();
14
15 const routes = [
16   { name: "First", component: First },
17   { name: "Second", component: Second },
18   { name: "Third", component: Third },
19   { name: "Fourth", component: Fourth },
20 ] as const;
21
22 function App() {
23   return (
24     <NavigationContainer>
25       <Stack.Navigator
26         screenOptions={({ route, navigation }) => {
27           const currentRouteIndex = routes
28             .map((r) => r.name)
29             .indexOf(route.name);
30
31           const prevRouteName = routes[currentRouteIndex - 1]?.name;
32           const nextRouteName = routes[currentRouteIndex + 1]?.name;
33
34           return {
35             headerBackVisible: false,
36             headerTitleAlign: "center",
37             headerTitle: () => (
38               <View style={styles.progress}>
39                 <Text style={styles.title}>{route.name}</Text>
40                 <ProgressBar progress={route.params.progress} />
41               </View>
42             ),
43             headerLeft: () => (
44               <Button
45                 title="Prev"
46                 disabled={currentRouteIndex === 0}
47                 onPress={() => navigation.navigate(prevRouteName)}
48               />
49             ),
50             headerRight: () => (
51               <Button
52                 title="Next"
53                 disabled={currentRouteIndex === 3}
54                 onPress={() => navigation.navigate(nextRouteName)}
55               />
56             );
57           );
58         }
59       >
60         {routes.map((routeProps, index) => (
61           <Stack.Screen
62             key={routeProps.name}
63             {...routeProps}
64             initialParams={{ progress: (index + 1) / routes.length }}
65           />
66         )));
67       </Stack.Navigator>
68     </NavigationContainer>
69   );
70 }
71
72 export default App;
```

# NPX ET CREATE EXPO APP

## LA BARRE DE PROGRESSION



# NPX ET CREATE EXPO APP

## AFFICHAGE DES ÉCRANS MODAUX

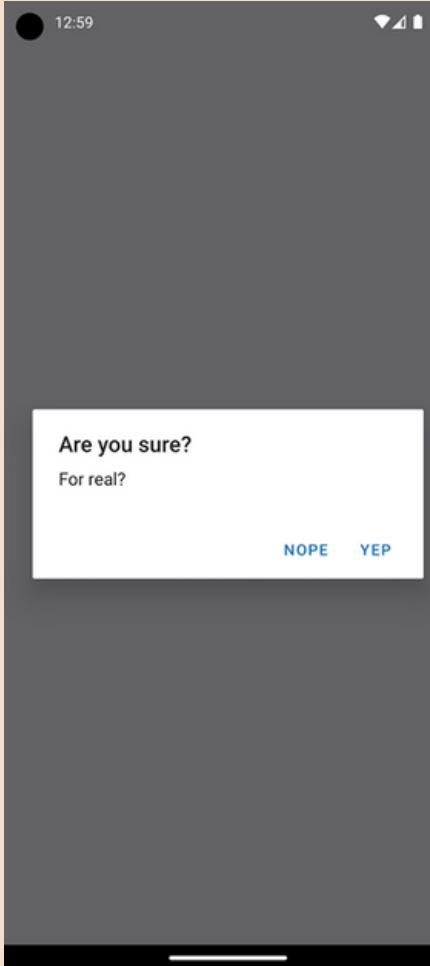
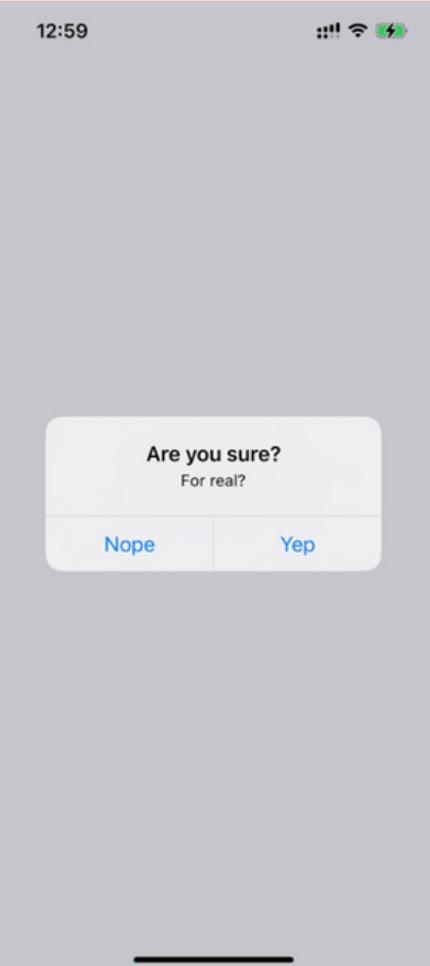
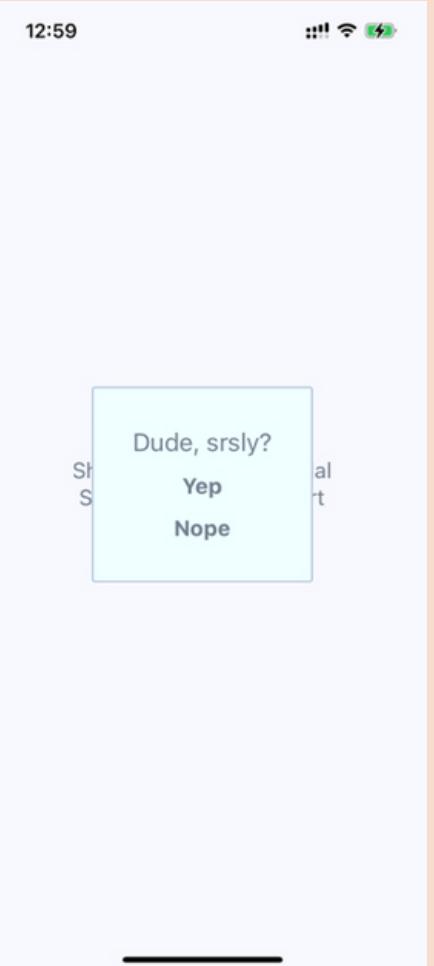
OBTENIR LA CONFIRMATION DE L'UTILISATEUR

```
✓ Your project is ready!
To run your project, navigate to the directory and run one of the following npm commands.
- cd modal-screen
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web
```

```
App.tsx M ConfirmationModal.tsx U TS styles.ts U
modal-screen > TS styles.ts > (default) > modalInner
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   text: {
12     color: "slategrey",
13   },
14
15   modalContainer: {
16     flex: 1,
17     justifyContent: "center",
18     alignItems: "center",
19   },
20
21   modalInner: {
22     backgroundColor: "azure",
23     padding: 20,
24     borderWidth: 1,
25     borderColor: "lightsteelblue",
26     borderRadius: 2,
27     alignItems: "center",
28   },
29
30   modalText: {
31     fontSize: 16,
32     margin: 5,
33     color: "slategrey",
34   },
35
36   modalButton: {
37     fontWeight: "bold",
38     margin: 5,
39     color: "slategrey",
40   },
41});
```

```
App.tsx M ConfirmationModal.tsx U TS styles.ts U
modal-screen > TS App > App > toggleAlert
1 import React, { useState } from "react";
2 import { View, Text, Alert } from "react-native";
3 import ConfirmationModal from "./ConfirmationModal";
4
5 import styles from "./styles";
6
7 export default function App() {
8   const [modalVisible, setModalVisible] = useState(false);
9
10   function toggleModal() {
11     setModalVisible(!modalVisible);
12   }
13
14   function toggleAlert() {
15     Alert.alert("Are you sure?", "For real?", [
16       { text: " Nope", onPress: () => {} },
17       { text: " Yep", onPress: () => {} },
18     ]);
19   }
20
21   return (
22     <View style={styles.container}>
23       <ConfirmationModal
24         animationType="fade"
25         visible={modalVisible}
26         onPressConfirm={toggleModal}
27         onPressCancel={toggleModal}
28       />
29       <Text style={styles.text} onPress={toggleModal}>
30         Show Confirmation Modal
31       </Text>
32       <Text style={styles.text} onPress={toggleAlert}>
33         Show Confirmation Alert
34       </Text>
35     </View>
36   );
37 }
```

```
App.tsx M ConfirmationModal.tsx U TS styles.ts U
modal-screen > ConfirmationModal.tsx > ConfirmationModal
1 import React from "react";
2 import { View, Text, Modal, ModalProps } from "react-native";
3 import styles from "./styles";
4
5 type Props = ModalProps & {
6   onPressConfirm: () => void;
7   onPressCancel: () => void;
8 };
9
10 export default function ConfirmationModal({onPressConfirm, onPressCancel, ...modalProps}: Props) {
11   return (
12     <Modal transparent onRequestClose={() => {}}>{...modalProps}</Modal>
13     <View style={styles.modalContainer}>
14       <View style={styles.modalInner}>
15         <Text style={styles.modalText}>Dude, srsly?</Text>
16         <Text style={styles.modalButton}>Yep</Text>
17         <Text style={styles.modalButton}> Nope</Text>
18       </View>
19     </View>
20   );
21 }
```



# NPX ET CREATE EXPO APP

## AFFICHAGE DES ÉCRANS MODAUX

### CONFIRMATION D'ERREUR

The code editor shows three files:

- App.tsx**: Contains the main application structure with imports for React, useState, View, Text, Modal, and ModalProps from react-native, and styles from ./styles.
- ErrorModal.tsx**: A modal component with styles for container, modalContainer, modalInner, modalInnerError, modalText, modalTextError, modalButton, and modalButtonError.
- styles.ts**: A file containing CSS-like styles for various components.

```
App.tsx
import React from "react";
import { View, Text, Modal, ModalProps } from "react-native";
import styles from "./styles";

export default StyleSheet.create([
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
    backgroundColor: "ghostwhite",
  },
  text: {
    color: "slategrey",
  },
  modalContainer: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  modalInner: {
    backgroundColor: "azure",
    padding: 20,
    borderWidth: 1,
    borderColor: "lightsteelblue",
    borderRadius: 2,
    alignItems: "center",
  },
  modalInnerError: {
    backgroundColor: "lightcoral",
    borderColor: "darkred",
  },
  modalText: {
    fontSize: 16,
    margin: 5,
    color: "slategrey",
  },
  modalTextError: {
    fontSize: 18,
    color: "darkred",
  },
  modalButton: {
    fontWeight: "bold",
    margin: 5,
    color: "slategrey",
  },
  modalButtonError: {
    color: "black",
  },
])

ErrorModal.tsx
import React from "react";
import { View, Text, Modal, ModalProps } from "react-native";
import styles from "./styles";

const innerViewStyle = [styles.modalInner, styles.modalInnerError];
const textStyle = [styles.modalText, styles.modalTextError];
const buttonStyle = [styles.modalButton, styles.modalButtonError];

type Props = ModalProps & {
  onPressConfirm: () => void;
  onPressCancel: () => void;
};

export default function ErrorModal({
  onPressConfirm,
  onPressCancel,
  ...modalProps
}: Props) {
  return (
    <Modal transparent onRequestClose={() => {}}>{...modalProps}</Modal>
    <View style={styles.modalContainer}>
      <View style={innerViewStyle}>
        <Text style={textStyle}>Epic fail!</Text>
        <Text style={buttonStyle} onPress={onPressConfirm}>
          Fix it
        </Text>
        <Text style={buttonStyle} onPress={onPressCancel}>
          Ignore it
        </Text>
      </View>
    </View>
  );
}

styles.ts
export const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: "center",
    alignItems: "center",
  },
  modalInner: {
    backgroundColor: "#fff",
    padding: 20,
    borderWidth: 1,
    borderColor: "#ccc",
    borderRadius: 2,
    alignItems: "center",
  },
  modalInnerError: {
    backgroundColor: "#f00",
    borderColor: "#800000",
  },
  modalText: {
    color: "#000",
    margin: 5,
  },
  modalTextError: {
    color: "#c00",
    margin: 5,
  },
  modalButton: {
    color: "#000",
    margin: 5,
  },
  modalButtonError: {
    color: "#000",
  },
});
```

The code editor shows two files:

- ErrorModal.tsx**: Contains the implementation of the ErrorModal component, which displays an error message and two buttons: "Fix it" and "Ignore it".
- App.tsx**: Contains the main application logic, including useState, a modalVisible state, and functions to toggle the modal and alert.

```
ErrorModal.tsx
import React from "react";
import { View, Text, Modal, ModalProps } from "react-native";
import styles from "./styles";

const innerViewStyle = [styles.modalInner, styles.modalInnerError];
const textStyle = [styles.modalText, styles.modalTextError];
const buttonStyle = [styles.modalButton, styles.modalButtonError];

type Props = ModalProps & {
  onPressConfirm: () => void;
  onPressCancel: () => void;
};

export default function ErrorModal({
  onPressConfirm,
  onPressCancel,
  ...modalProps
}: Props) {
  return (
    <Modal transparent onRequestClose={() => {}}>{...modalProps}</Modal>
    <View style={styles.modalContainer}>
      <View style={innerViewStyle}>
        <Text style={textStyle}>Epic fail!</Text>
        <Text style={buttonStyle} onPress={onPressConfirm}>
          Fix it
        </Text>
        <Text style={buttonStyle} onPress={onPressCancel}>
          Ignore it
        </Text>
      </View>
    </View>
  );
}

App.tsx
import React, { useState } from "react";
import { View, Text, Alert } from "react-native";
import styles from "./styles";
import ErrorModal from "./ErrorModal";

export default function App() {
  const [modalVisible, setModalVisible] = useState(false);

  function toggleModal() {
    setModalVisible(!modalVisible);
  }

  function toggleAlert() {
    Alert.alert("", "Failed to do the thing...", [
      {
        text: "Dismiss",
      },
    ]);
  }

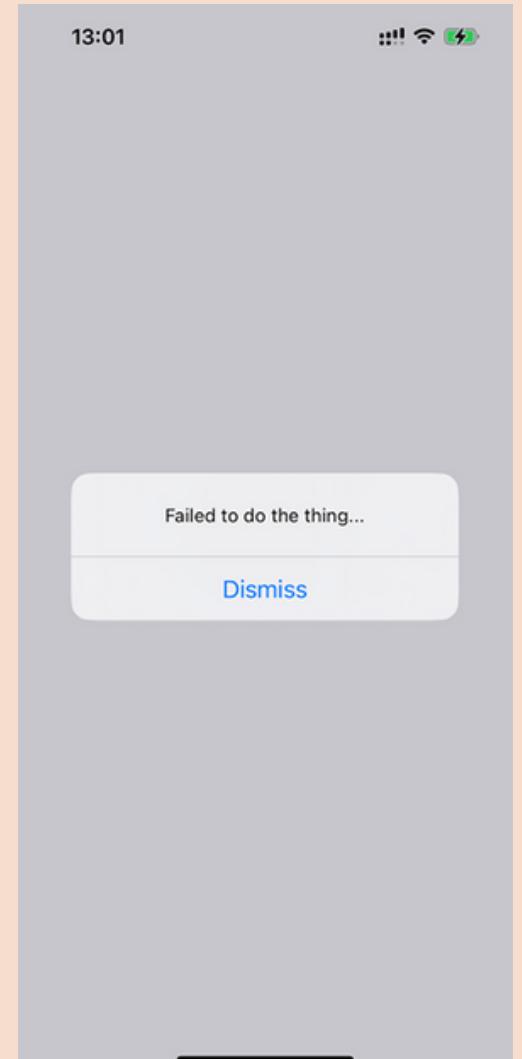
  return (
    <View style={styles.container}>
      <ErrorModal
        animationType="fade"
        visible={modalVisible}
        onPressConfirm={toggleModal}
        onPressCancel={toggleModal}
      />

      <Text style={styles.text} onPress={toggleModal}>
        Show Error Modal
      </Text>
      <Text style={styles.text} onPress={toggleAlert}>
        Show Error Alert
      </Text>
    </View>
  );
}
```

The mobile device screen shows a modal dialog with the following content:

- Epic fail!** (Red background)
- Fix it**
- Ignore it**

At the bottom right of the modal, there is a red button labeled "Epic fail!" and three buttons: "Fix it", "Ignore it", and "Dismiss".



# NPX ET CREATE EXPO APP

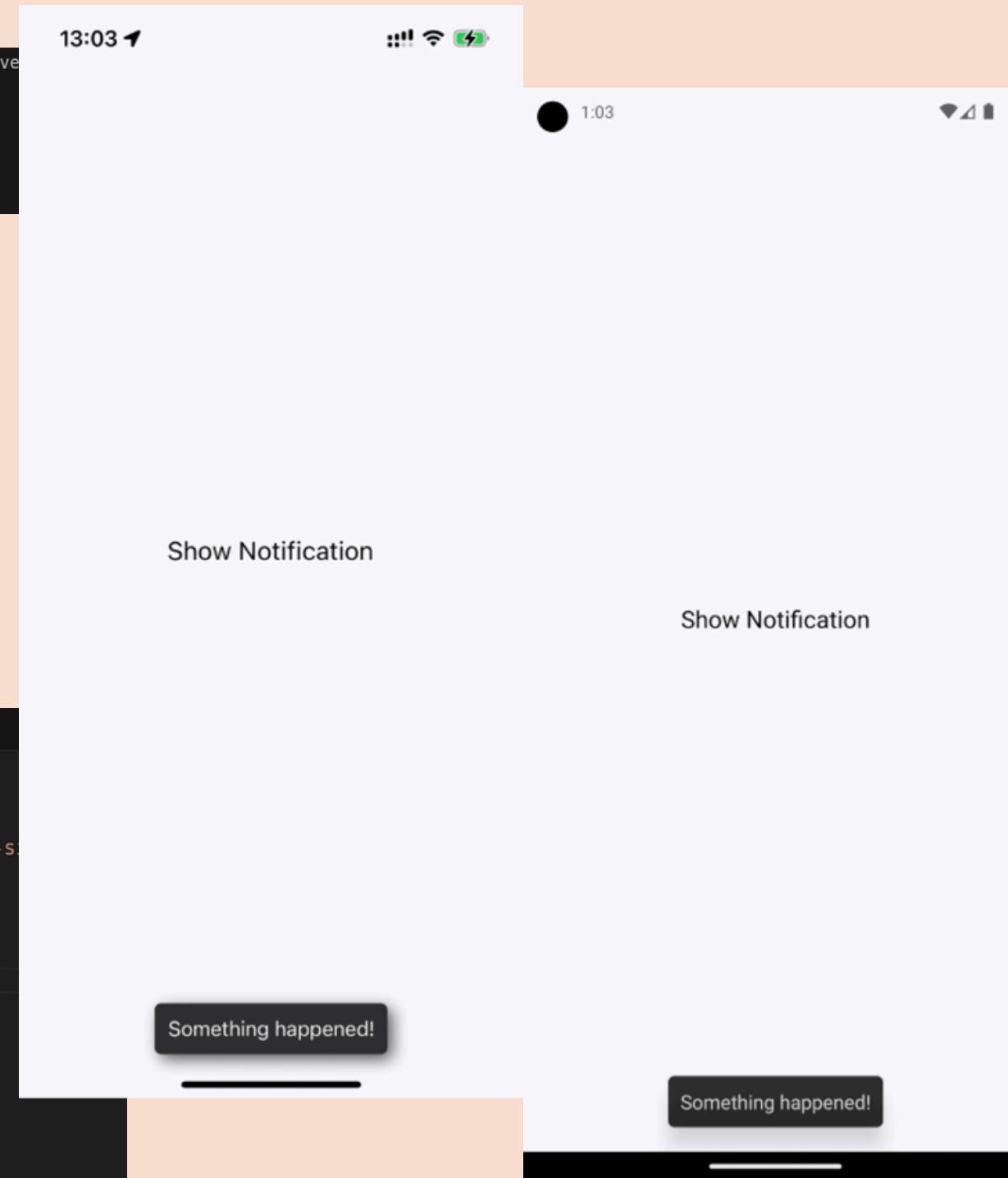
## AFFICHAGE DES ÉCRANS MODAUX

### NOTIFICATIONS PASSIVES

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/modal-screen$ npm i react-native-root-siblings react-native-root-toast
added 5 packages, and audited 1198 packages in 8s
131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
App.tsx M | TS styles.ts U X
modal-screen > TS styles.ts > default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create([
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   notificationContainer: {
12     flex: 1,
13     justifyContent: "center",
14     alignItems: "center",
15   },
16
17   notificationInner: {
18     backgroundColor: "azure",
19     padding: 20,
20     borderWidth: 1,
21     borderColor: "lightsteelblue",
22     borderRadius: 2,
23     alignItems: "center",
24   },
25 ]);

App.tsx M X | TS styles.ts U
modal-screen > App.tsx > PassiveNotifications
1 import React from "react";
2 import { Text, View } from "react-native";
3 import { RootSiblingParent } from "react-native-root-siblings";
4 import styles from "./styles";
5 import Toast from "react-native-root-toast";
6
7 export default function PassiveNotifications() {
8   return [
9     <RootSiblingParent>
10       <View style={styles.container}>
11         <Text
12           onPress={() => {
13             Toast.show("Something happened!", {
14               duration: Toast.durations.LONG,
15             });
16           }}>
17           >
18             Show Notification
19           </Text>
20         </View>
21       </RootSiblingParent>
22     ];
23 }
```



# NPX ET CREATE EXPO APP

## AFFICHAGE DES ÉCRANS MODAUX

### MODAUX D'ACTIVITÉ

```
App.tsx 1,M Activity.tsx U ts styles.ts U X
modal-screen > ts styles.ts > [o] default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   modalContainer: {
12     flex: 1,
13     justifyContent: "center",
14     alignItems: "center",
15     backgroundColor: "rgba(0, 0, 0, 0.2)",
16   },
17 });

```

```
App.tsx 1,M Activity.tsx U ts styles.ts U
modal-screen > Activity.tsx > ⓘ Activity
1 import React from "react";
2 import { View, Modal, ActivityIndicator } from "react-native";
3 import styles from "./styles";
4
5 type ActivityProps = {
6   visible: boolean;
7   size?: "small" | "large";
8 };
9
10 export default function Activity({ visible, size = "large" }: ActivityProps) {
11   return (
12     <Modal visible={visible} transparent>
13       <View style={styles.modalContainer}>
14         <ActivityIndicator size={size} />
15       </View>
16     </Modal>
17   );
18 }
```

```
App.tsx M X Activity.tsx U ts styles.ts U
modal-screen > App.tsx > ⓘ App > ⓘ onPress
1 import React, { useState } from "react";
2 import { Text, View } from "react-native";
3 import styles from "./styles";
4 import Activity from "./Activity";
5
6 export default function App() {
7   const [fetching, setFetching] = useState(false);
8   const [promise, setPromise] = useState(Promise.resolve());
9
10   function onPress() {
11     setPromise(
12       new Promise((resolve) => setTimeout(resolve, 3000)).then(() => {
13         setFetching(false);
14       })
15     );
16     setFetching(true);
17   }
18
19   return (
20     <View style={styles.container}>
21       <Activity visible={fetching} />
22       <Text onPress={onPress}>Fetch Stuff...</Text>
23     </View>
24   );
25 }
```



# NPX ET CREATE EXPO APP

## UTILISER DES ANIMATIONS

UTILISER LES ANIMATIONS REACT NATIVE RÉANIMÉ

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/modal-screen$ npx expo install react-native-reanimated
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

added 1 package, and audited 1199 packages in 5s

131 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
modal-screen > ⚡ babel.config.js > ...
1 module.exports = function(api) {
2   api.cache(true);
3   return {
4     presets: ['babel-preset-expo'],
5     plugins: ['react-native-reanimated/plugin'],
6   };
7 }
8 |
```

# NPX ET CREATE EXPO APP

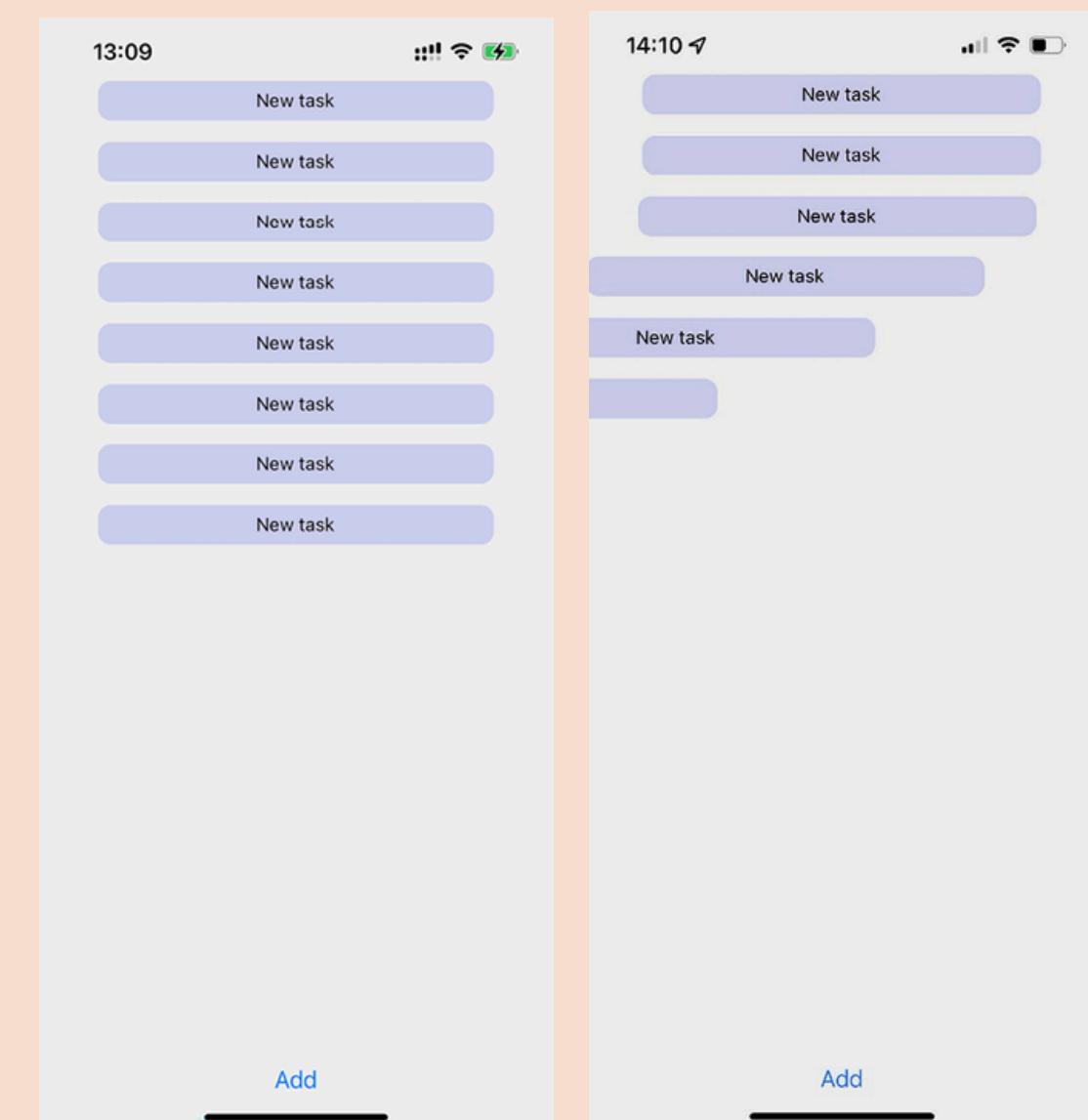
## UTILISER DES ANIMATIONS

### ANIMATION DES COMPOSANTS DE MISE EN PAGE

```
modal-screen > ts styles.ts > styles > container
1 import { StyleSheet } from "react-native";
2
3 export const styles = StyleSheet.create({
4   container: [
5     paddingTop: 40,
6     paddingBottom: 20,
7     flexDirection: "column",
8     alignItems: "center",
9     flex: 1,
10    backgroundColor: "#eee",
11  ],
12
13  todoItem: {
14    width: 300,
15    height: 30,
16    backgroundColor: "#ccceee",
17    borderRadius: 10,
18    alignItems: "center",
19    justifyContent: "center",
20    marginVertical: 8,
21  },
22});
```

```
modal-screen > TodoItem.tsx > ...
1 import { Text, TouchableOpacity } from "react-native";
2 import Animated, { SlideInLeft, SlideOutRight } from "react-native-reanimated";
3 import { styles } from "./styles";
4
5 export type Todo = {
6   id: string;
7   title: string;
8 };
9
10 type TodoItemProps = Todo & {
11   onPress: (id: string) => void;
12 };
13
14 export const TodoItem = ({ id, title, onPress }: TodoItemProps) => {
15   return (
16     <Animated.View entering={SlideInLeft} exiting={SlideOutRight}>
17       <TouchableOpacity onPress={() => onPress(id)} style={styles.todoItem}>
18         <Text>{title}</Text>
19       </TouchableOpacity>
20     </Animated.View>
21   );
22};
```

```
modal-screen > App.tsx > App > addTask
1 import { useState } from "react";
2 import { Button, View } from "react-native";
3 import { TodoItem, Todo } from "./TodoItem";
4 import { styles } from "./styles";
5
6 export default function App() {
7   const [todoList, setTodoList] = useState<Todo[]>([]);
8
9   const addTask = () => {
10     setTodoList([
11       ...todoList,
12       { id: String(new Date().getTime()), title: "New task" },
13     ]);
14   };
15
16   const deleteTask = (id: string) => {
17     setTodoList(todoList.filter(todo => todo.id !== id));
18   };
19
20   return (
21     <View style={styles.container}>
22       <View style={{ flex: 1 }}>
23         {todoList.map(({ id, title }) => (
24           <TodoItem key={id} id={id} title={title} onPress={deleteTask} />
25         ))}
26       </View>
27       <Button onPress={addTask} title="Add" />
28     </View>
29   );
30 }
```



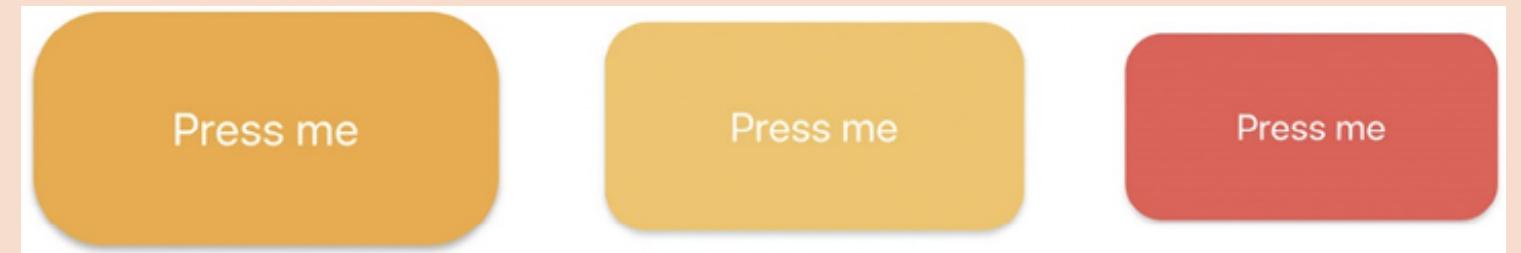
# NPX ET CREATE EXPO APP

## UTILISER DES ANIMATIONS

### ANIMATION DES STYLES DE COMPOSANTS

```
modal-screen > ts styles.ts > [o] styles > ↵ buttonContainer
1 import { StyleSheet } from "react-native";
2
3 const styles = StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     height: "100%",
9   },
10  buttonContainer: [
11    height: 100,
12    width: 200,
13    backgroundColor: "orange",
14    shadowColor: "#000",
15    shadowOffset: {
16      width: 0,
17      height: 2,
18    },
19    shadowOpacity: 0.3,
20    shadowRadius: 1.4,
21  ],
22  button: {
23    flex: 1,
24    justifyContent: "center",
25    alignItems: "center",
26  },
27  buttonText: {
28    color: "#fff",
29    fontSize: 20,
30  },
31});
32
33 export default styles;
```

```
modal-screen > ↵ App.tsx > [o] App
12 const App = () => {
13   const animatedStyles = useAnimatedStyle(() => {
14     const [radius, setRadius] = useState(20);
15     const [color, setColor] = useState("white");
16     const [scale, setScale] = useState(1);
17     const [opacity, setOpacity] = useState(0.7);
18
19     const onPressIn = () => {
20       setRadius(20);
21       setScale(0.9);
22       setOpacity(0.7);
23     };
24
25     const onLongPress = () => {
26       setRadius(30);
27       setScale(0.8);
28       setColor("black");
29     };
30
31     const onPressOut = () => {
32       setRadius(20);
33       setScale(1);
34       setColor("white");
35     };
36
37     return (
38       <View style={styles.container}>
39         <Animated.View style={[styles.buttonContainer, animatedStyles]}>
40           <Pressable
41             onPressIn={onPressIn}
42             onPressOut={onPressOut}
43             onLongPress={onLongPress}
44             style={styles.button}
45           >
46             <Text style={styles.buttonText}>Press me</Text>
47           </Pressable>
48         </Animated.View>
49       </View>
50     );
51   });
52
53   return (
54     <View style={styles.container}>
55       <Text style={styles.buttonText}>Press me</Text>
56     </View>
57   );
58 }
59
60 </View>
61 </App>
62 );
63 </>;
64
65 | export default App;
```

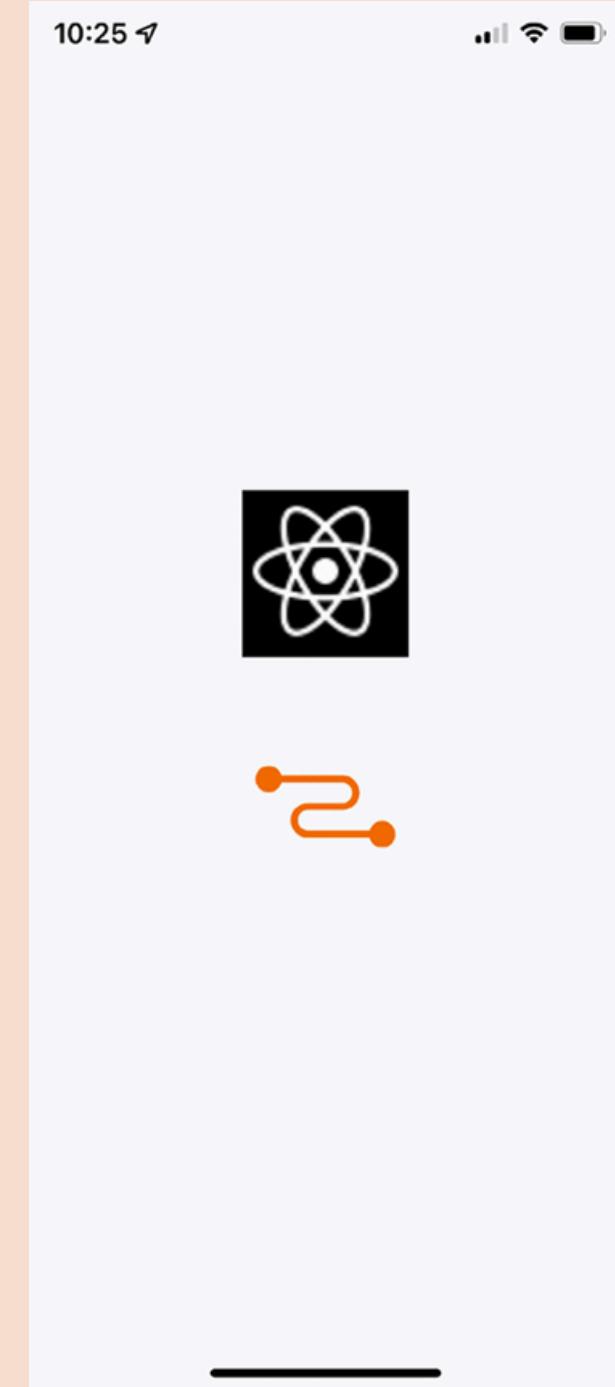


# NPX ET CREATE EXPO APP

## Contrôle de l'affichage des images

Chargement d'images

```
image-controle > ts styles.ts > default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create([
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   image: {
12     width: 100,
13     height: 100,
14     margin: 20,
15   },
16 ]);
image-controle > App.tsx > App
1 import React from "react";
2 import { View, Image } from "react-native";
3 import styles from "./styles";
4
5 const reactLogo = "https://reactnative.dev/docs/assets/favicon.png";
6 const relayLogo = require("./assets/relay.png");
7
8 export default function App() {
9   return (
10     <View style={styles.container}>
11       <Image style={styles.image} source={{ uri: reactLogo }} />
12       <Image style={styles.image} source={relayLogo} />
13     </View>
14   );
15 }
```



# NPX ET CREATE EXPO APP

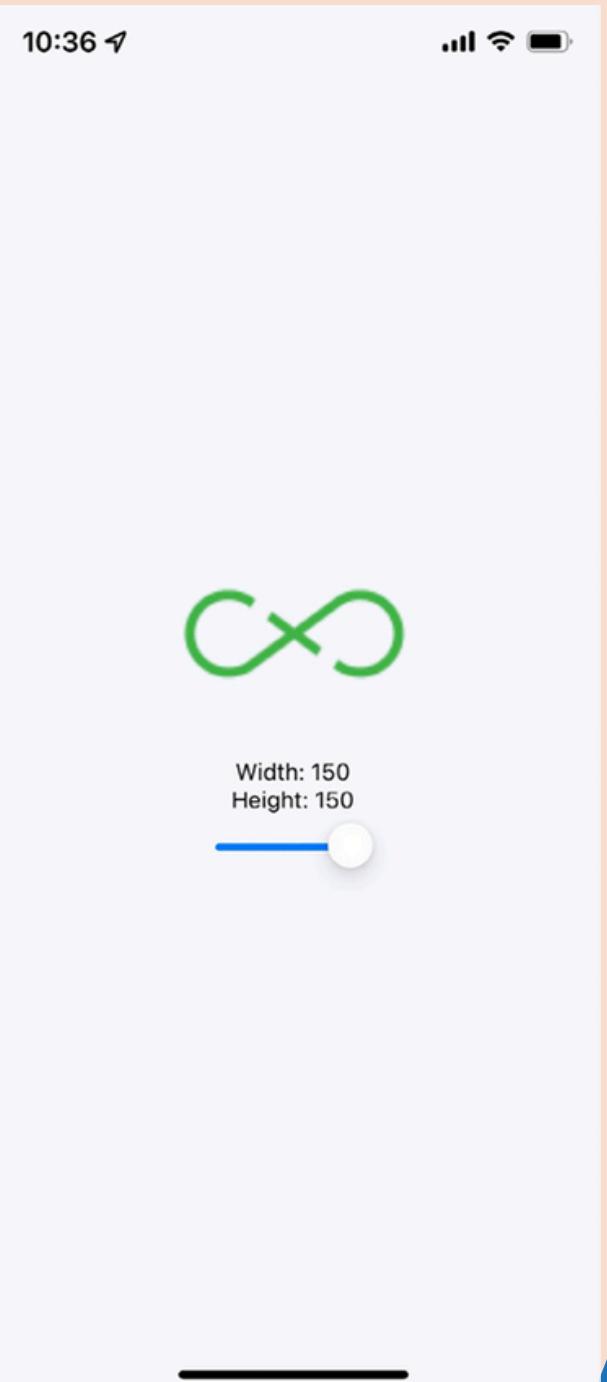
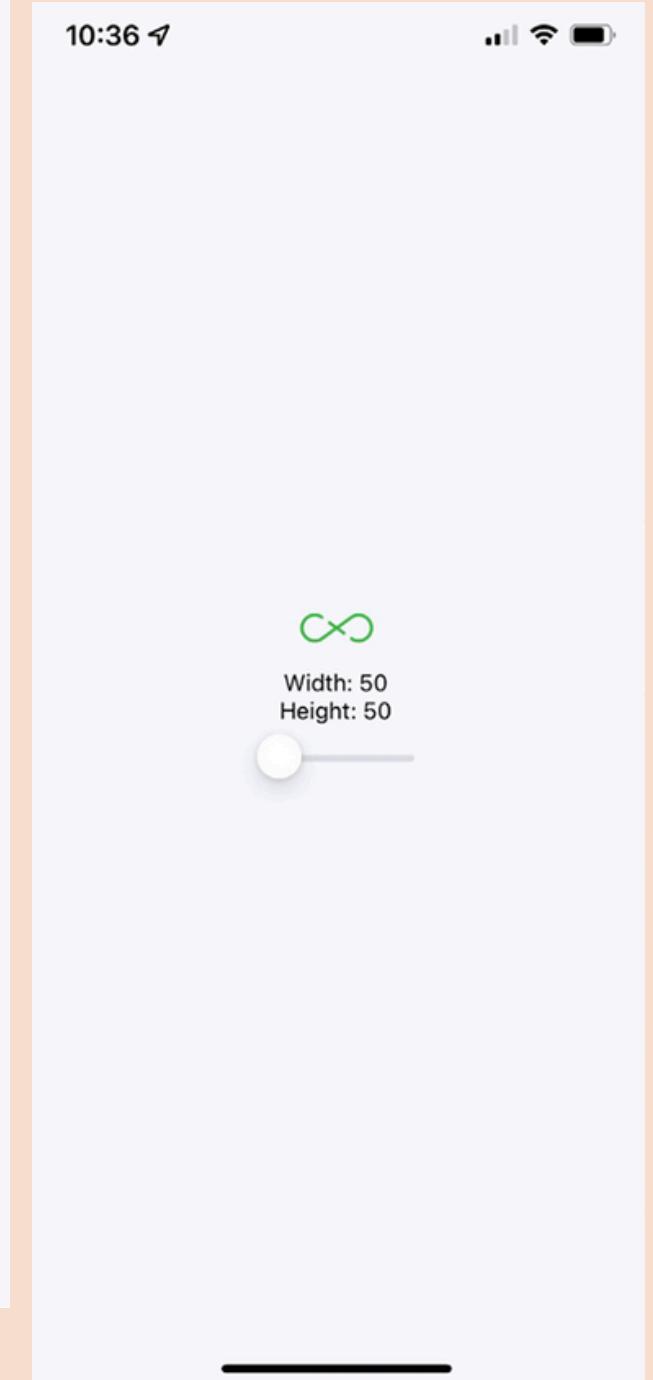
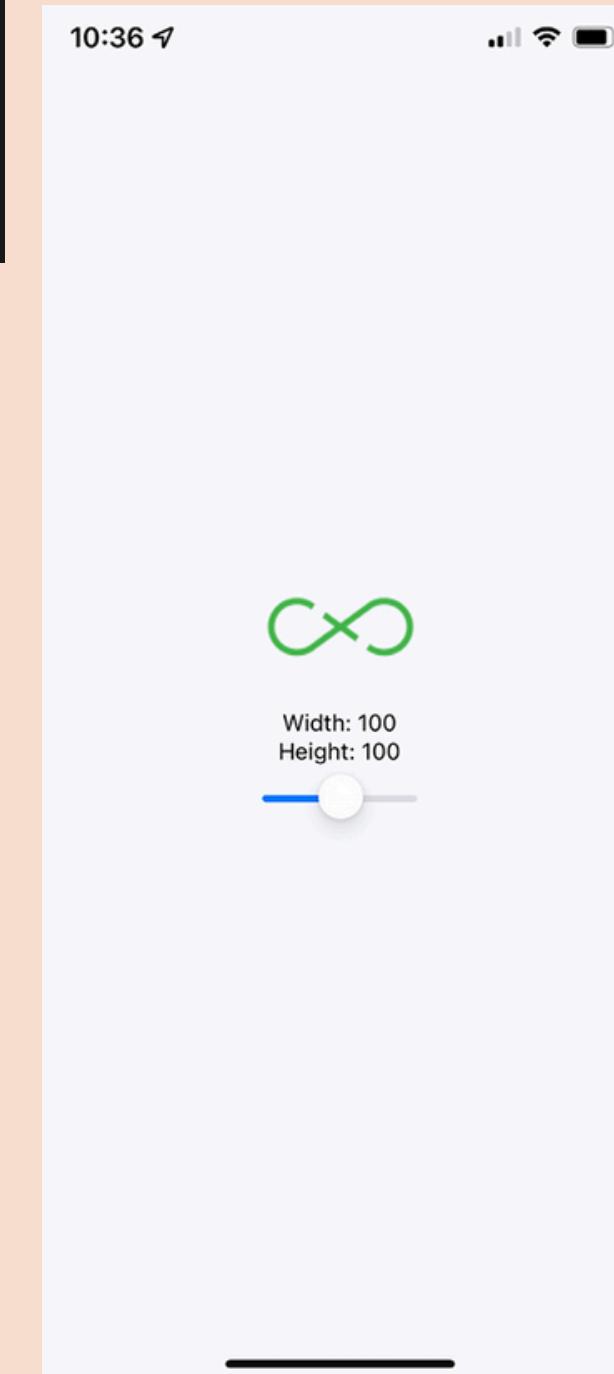
## CONTRÔLE DE L'AFFICHAGE DES IMAGES

### REDIMENSIONNER LES IMAGES

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/image-controle$ npm i @react-native-community/slider
added 1 package, and audited 1194 packages in 5s
131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
○ balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/image-controle$
```

```
App.tsx styles.ts X
image-controle > styles.ts > default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create([
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   slider: {
12     width: 100,
13   },
14 ]);
```

```
App.tsx X styles.ts
image-controle > App.tsx > App
1 import React, { useState } from "react";
2 import { View, Text, Image } from "react-native";
3 import Slider from "@react-native-community/slider";
4 import styles from "./styles";
5
6 export default function App() {
7   const source = require("./assets/flux.png");
8   const [width, setWidth] = useState(100);
9   const [height, setHeight] = useState(100);
10
11   return (
12     <View style={styles.container}>
13       <Image source={source} style={{ width, height }} />
14       <Text>Width: {width}</Text>
15       <Text>Height: {height}</Text>
16       <Slider
17         style={styles.slider}
18         minimumValue={50}
19         maximumValue={150}
20         value={width}
21         onValueChange={(value) => {
22           setWidth(value);
23           setHeight(value);
24         }}
25       />
26     </View>
27   );
28 }
```



# NPX ET CREATE EXPO APP

## CONTRÔLE DE L'AFFICHAGE DES IMAGES

### CHARGEMENT D'IMAGE EN DIFFÉRÉ

```
App.tsx | Button.tsx | LazyImage.tsx | ts styles.ts
image-controle > ts styles.ts > default > button
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9   },
10
11   button: [
12     padding: 10,
13     margin: 5,
14     backgroundColor: "azure",
15     borderWidth: 1,
16     borderRadius: 4,
17     borderColor: "slategrey",
18   ],
19
20   buttonText: {
21     color: "slategrey",
22   },
23 });

App.tsx | Button.tsx | LazyImage.tsx | ts styles.ts
image-controle > LazyImage.tsx > LazyImage
1 import React, { useState } from "react";
2 import {
3   View,
4   Image,
5   ImageProps,
6   ImageStyle,
7   StyleProp,
8   ImageSourcePropTypes,
9 } from "react-native";
10
11 const placeholder = require("./assets/placeholder.png");
12
13 type PlaceholderProps = {
14   loaded: boolean;
15   style: StyleProp<ImageStyle>;
16 };
17
18 function Placeholder({ loaded, style }: PlaceholderProps) {
19   if (loaded) {
20     return null;
21   } else {
22     return <Image style={style} source={placeholder} />;
23   }
24 }
25
26 type Props = {
27   style: StyleProp<ImageStyle>;
28   resizeMode: ImageProps["resizeMode"];
29   source: ImageSourcePropTypes | null;
30 };
31
32 export default function LazyImage({ style, resizeMode, source }: Props) {
33   const [loaded, setLoaded] = useState(false);
34
35   return (
36     <View style={style}>
37       {!source ? [
38         <Image
39           source={source}
40           resizeMode={resizeMode}
41           style={style}
42           onLoad={() => {
43             setLoaded(true);
44           }}
45         />
46         : (
47           <Placeholder loaded={loaded} style={style} />
48         )
49       ]}>
50     </View>
51   );
}

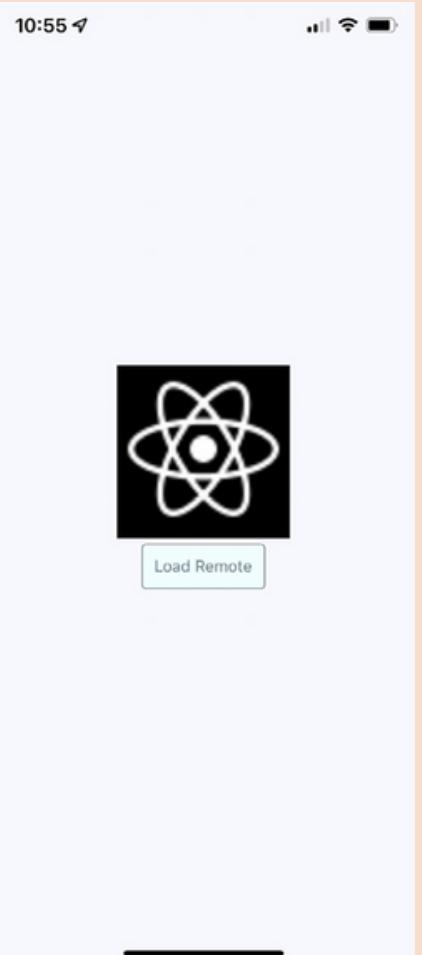
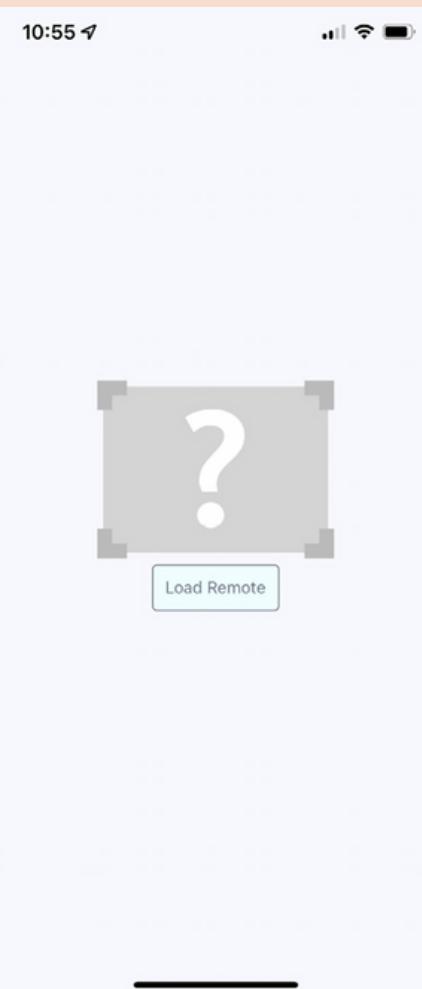
LazyImage.tsx | ts styles.ts
image-controle > ts styles.ts
10:55 ↗
```

```
App.tsx | Button.tsx | LazyImage.tsx | ts styles.ts
image-controle > LazyImage.tsx > LazyImage
1 import React from "react";
2 import { Text, TouchableOpacity } from "react-native";
3 import styles from "./styles";
4
5 type Props = {
6   label: string;
7   onPress: () => void;
8 };
9
10 export default function Button({ label, onPress }: Props) {
11   return (
12     <TouchableOpacity style={styles.button} onPress={onPress}>
13       <Text style={styles.buttonText}>{label}</Text>
14     </TouchableOpacity>
15   );
}

LazyImage.tsx | ts styles.ts
image-controle > ts styles.ts
10:55 ↗
```

```
App.tsx | Button.tsx | LazyImage.tsx | ts styles.ts
image-controle > App.tsx > LazyLoading
1 import React, { useState } from "react";
2 import { ImageSourcePropTypes, View } from "react-native";
3 import styles from "./styles";
4 import LazyImage from "./LazyImage";
5 import Button from "./Button";
6
7 const remote = "https://reactnative.dev/docs/assets/favicon.png";
8
9 export default function LazyLoading() {
10   const [source, setSource] = useState<ImageSourcePropTypes | null>(null);
11
12   return [
13     <View style={styles.container}>
14       <LazyImage
15         style={{ width: 200, height: 150 }}
16         resizeMode="contain"
17         source={source}
18       />
19       <Button
20         label="Load Remote"
21         onPress={() => {
22           setSource({ uri: remote });
23         }}
24       />
25     </View>
26   ];
}

LazyImage.tsx | ts styles.ts
image-controle > ts styles.ts
10:55 ↗
```



# NPX ET CREATE EXPO APP

## CONTRÔLE DE L'AFFICHAGE DES IMAGES

### ICÔNES DE RENDU

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/image-controle$ npm i @react-native-picker/picker
added 1 package, and audited 1195 packages in 5s
131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/image-controle$ npx expo install @expo/vector-icons
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

up to date, audited 1195 packages in 3s
131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

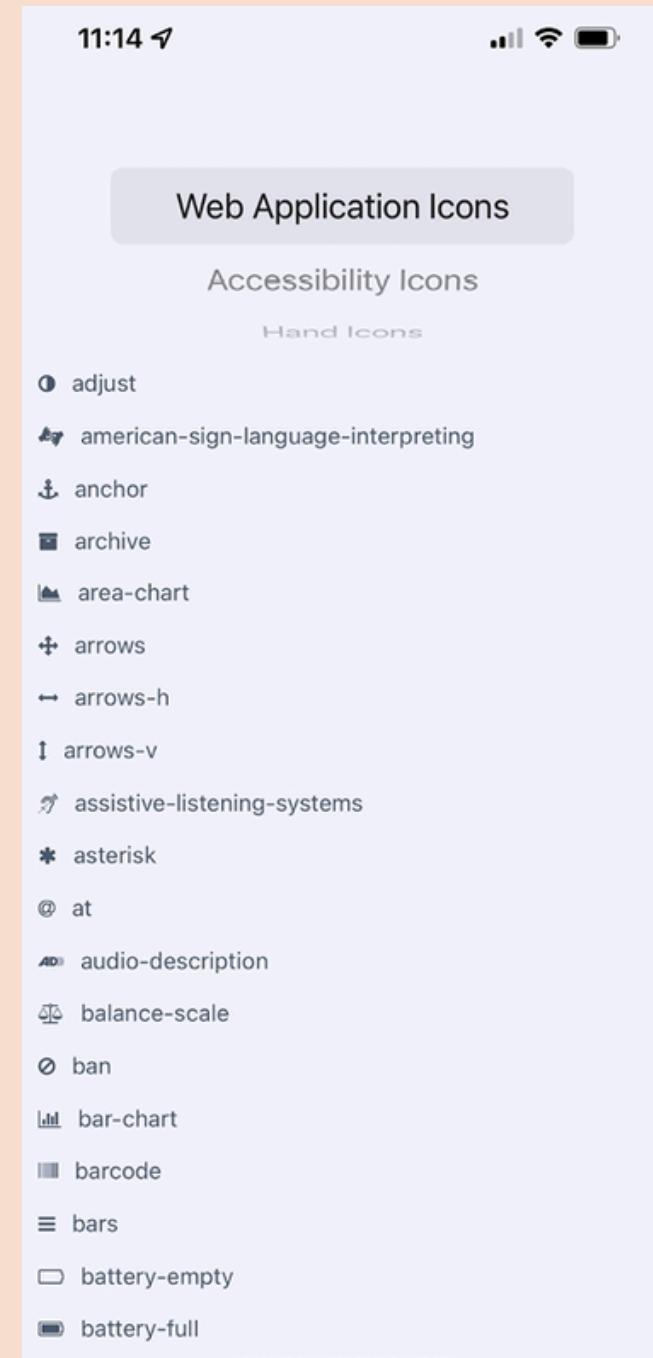
```
App.tsx 1  ts icon-names.ts  ts styles.ts
image-controle > ts styles.ts > [o] default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create([
4   container: {
5     flex: 1,
6     alignItems: "center",
7     backgroundColor: "ghostwhite",
8   },
9
10  picker: {
11    height: 200,
12    width: 300,
13    marginTop: 20,
14  },
15
16  icons: {
17    alignSelf: "stretch",
18  },
19
20  item: {
21    flex: 1,
22    flexDirection: "row",
23    alignItems: "center",
24  },
25
26  itemIcon: {
27    padding: 10,
28    color: "slategrey",
29  },
30
31  itemText: {
32    color: "slategrey",
33  },
34]);
```

```
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
```

```
ally: [
  "american-sign-language-interpreting",
  "assistive-listening-systems",
  "audio-description",
  "blind",
  "braille",
  "cc",
  "deaf",
  "archive",
  "area-chart",
  "arrows",
  "arrows-h",
  "arrows-v",
  "assistive-listening-systems",
  "asterisk",
  "at",
  "audio-description",
  "balance-scale",
  "ban",
  "bar-chart",
  "barcode",
  "bars",
  "battery-empty",
  "battery-full",
  "battery-half",
  "battery-quarter",
  "battery-three-quarters",
  "bed",
  "beer",
  "bell",
],
ally: [
  "american-sign-language-interpreting",
  "assistive-listening-systems",
  "audio-description",
  "blind",
  "braille",
  "cc",
  "deaf",
]
```

```
export const iconNames = {
  web_app_icons: [
    "adjust",
    "american-sign-language-interpreting",
    "anchor",
    "archive",
    "area-chart",
    "arrows",
    "arrows-h",
    "arrows-v",
    "assistive-listening-systems",
    "asterisk",
    "at",
    "audio-description",
    "balance-scale",
    "ban",
    "bar-chart",
    "barcode",
    "bars",
    "battery-empty",
    "battery-full"
  ],
  hand: [
    "hand-lizard-o",
    "hand-o-down",
    "hand-o-left",
    "hand-o-right",
    "hand-o-up",
    "hand-paper-o",
    "hand-peace-o",
    "hand-pointer-o",
    "hand-rock-o",
    "hand-scissors-o",
    "hand-spock-o",
    "thumbs-down",
    "thumbs-o-down",
    "thumbs-o-up",
    "thumbs-up"
  ]
} as const;
```

```
export type IconsType = keyof typeof iconNames;
export type IconName = (typeof iconNames)[IconsType][number];
```



# NPX ET CREATE EXPO APP

## SE DÉCONNECTER

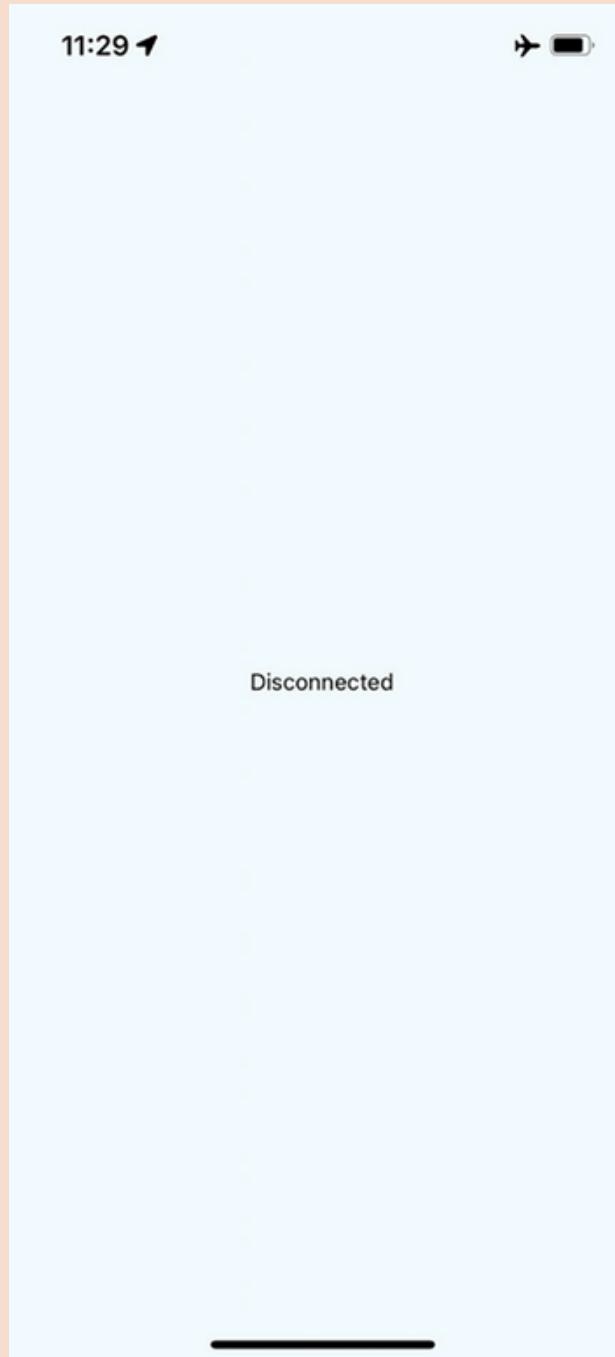
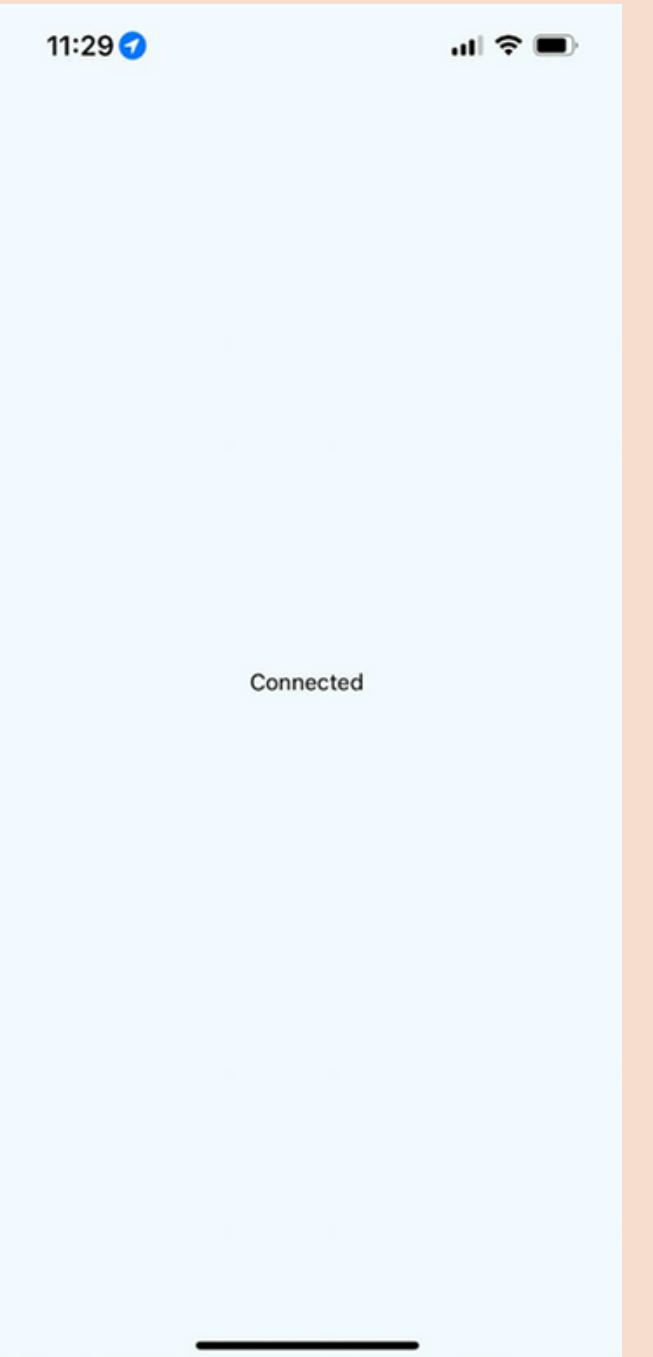
### DÉTECTION DE L'ÉTAT DU RÉSEAU

```
✓ Your project is ready!
To run your project, navigate to the directory and run one of the following npm commands.
- cd offline
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web

● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/offline$ npm i @react-native-community/netinfo
added 1 package, and audited 1194 packages in 6s
131 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

```
App.tsx M TS styles.ts U
offline > TS styles.ts > [o] default > welcome
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "#F5FCFF",
9   },
10  welcome: [
11    {
12      fontSize: 20,
13      textAlign: "center",
14      margin: 10,
15    },
16    {
17      instructions: {
18        textAlign: "center",
19        color: "#333333",
20        marginBottom: 5,
21      },
22    };
23  ];
24});
```

```
App.tsx M TS styles.ts U
offline > App.tsx > ...
1 import React, { useState, useEffect } from "react";
2 import { Text, View } from "react-native";
3 import NetInfo, { NetInfoState } from "@react-native-community/netinfo";
4 import styles from "./styles";
5
6 const connectedMap = {
7   none: "Disconnected",
8   unknown: "Disconnected",
9   cellular: "Connected",
10  wifi: "Connected",
11  bluetooth: "Connected",
12  ethernet: "Connected",
13  wimax: "Connected",
14  vpn: "Connected",
15  other: "Connected",
16} as const;
17
18 export default function App() {
19  const [connected, setConnected] = useState("");
20
21  useEffect(() => {
22    function onNetworkChange(connection: NetInfoState) {
23      const type = connection.type;
24      setConnected(connectedMap[type]);
25    }
26
27    const unsubscribe = NetInfo.addEventListener(onNetworkChange);
28
29    return () => {
30      unsubscribe();
31    };
32  }, []);
33
34  return (
35    <View style={styles.container}>
36      <Text>{connected}</Text>
37    </View>
38  );
39}
```



# NPX ET CREATE EXPO APP

## SE DÉCONNECTER

### STOCKAGE DES DONNÉES D'APPLICATION

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/offline$ npx expo install @react-native-async-storage/async-storage
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

added 3 packages, and audited 1197 packages in 5s

131 packages are looking for funding
  run `npm fund` for details

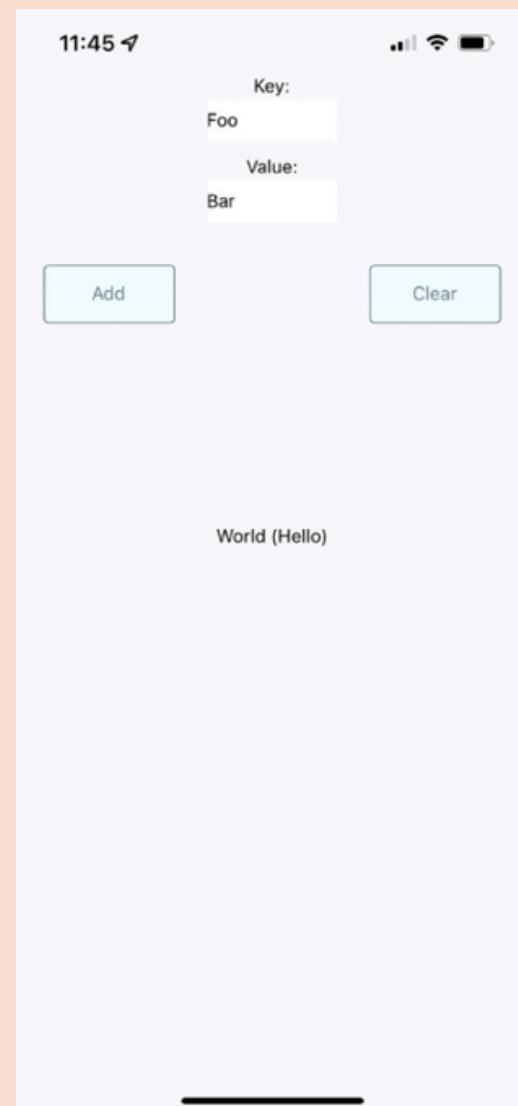
found 0 vulnerabilities
```

```
App.tsx M | Button.tsx U | TS styles.ts U
offline > TS styles.ts > default > button
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "ghostwhite",
9     paddingTop: 50,
10    },
11
12   input: {
13     height: 35,
14     backgroundColor: "white",
15     marginBottom: 10,
16     width: 100,
17   },
18
19   list: {
20     height: 450,
21   },
22
23   controls: {
24     flex: 1,
25     flexDirection: "row",
26     alignSelf: "stretch",
27     justifyContent: "space-between",
28     padding: 20,
29   },
30
31   button: [
32     height: 45,
33     width: 100,
34     padding: 10,
35     margin: 1,
36     backgroundColor: "azure",
37     borderWidth: 1,
38     borderRadius: 4,
39     borderColor: "slategrey",
40     justifyContent: "center",
41     alignItems: "center",
42   ],
43
44   buttonText: {
45     color: "slategrey",
46   },
47 });

Button.tsx U
offline > App.tsx > App
1 import React, { useState, useEffect } from "react";
2 import { Text, TextInput, View, FlatList } from "react-native";
3 import AsyncStorage from "@react-native-async-storage/async-storage";
4 import styles from "./styles";
5 import { Button } from "./Button";
6 import { KeyValuePair } from "@react-native-async-storage/async-storage/lib/typescript/types";
7
8 export default function App() {
9   const [key, setKey] = useState("");
10  const [value, setValue] = useState("");
11  const [source, setSource] = useState<KeyValuePair[]>([{}]);
12
13  function setItem() {
14    return AsyncStorage.setItem(key, value)
15      .then(() => {
16        setKey("");
17        setValue("");
18      })
19      .then(loadItems);
20  }
21
22  function clearItems() {
23    return AsyncStorage.clear().then(loadItems);
24  }
25
26  async function loadItems() {
27    const keys = await AsyncStorage.getAllKeys();
28    const values = await AsyncStorage.multiGet(keys);
29    setSource([...values]);
30  }
31
32  useEffect(() => {
33    loadItems();
34  }, []);
35
36  return (
37    <View style={styles.container}>
38      <Text>Key:</Text>
39      <TextInput
40        style={styles.input}
41        value={key}
42        onChangeText={(v) => {
43          setKey(v);
44        }}
45      />
46      <Text>Value:</Text>
47      <TextInput
48        style={styles.input}
49        value={value}
50        onChangeText={(v) => {
51          setValue(v);
52        }}
53      />
54      <View style={styles.controls}>
55        <Button label="Add" onPress={setItem} />
56        <Button label="Clear" onPress={clearItems} />
57      </View>
58      <View style={styles.list}>
59        <FlatList
60          data={source.map(([key, value]) => ({
61            key: key.toString(),
62            value,
63          }))};
64          renderItem={({ item: { value, key } }) => (
65            <Text>
66              | {value} ({key})
67            </Text>
68          )}
69        </FlatList>
70      </View>
71    </View>
72  );
73}
```

```
App.tsx M | Button.tsx U | TS styles.ts U
offline > Button.tsx > Button
1 import React from "react";
2 import { Text, TouchableOpacity } from "react-native";
3 import styles from "./styles";
4
5 type ButtonProps = {
6   label: string;
7   onPress: () => void;
8 };
9
10 export const Button = ({ label, onPress }: ButtonProps) => {
11   return (
12     <TouchableOpacity
13       style={styles.button}
14       onPress={onPress}
15       activeOpacity={0.5}
16     >
17       <Text style={styles.buttonText}>{label}</Text>
18     </TouchableOpacity>
19   );
20};
```

```
App.tsx M | Button.tsx U | TS styles.ts U
offline > App.tsx > App
8 export default function App() {
44   </View>
45   <Text>Value:</Text>
46   <TextInput
47     style={styles.input}
48     value={value}
49     onChangeText={(v) => {
50       setValue(v);
51     }}
52   />
53   <View style={styles.controls}>
54     <Button label="Add" onPress={setItem} />
55     <Button label="Clear" onPress={clearItems} />
56   </View>
57   <View style={styles.list}>
58     <FlatList
59       data={source.map(([key, value]) => ({
60         key: key.toString(),
61         value,
62       }))};
63       renderItem={({ item: { value, key } }) => (
64         <Text>
65           | {value} ({key})
66         </Text>
67       )}
68     </FlatList>
69   </View>
70 }
71
72 }
```



# NPX ET CREATE EXPO APP

## SE DÉCONNECTER

### SYNCHRONISATION DES DONNÉES D'APPLICATION

```
App.tsx M | TS store.ts U | TS styles.ts X
offline > ts styles.ts > [!] default
1 import { StyleSheet } from "react-native";
2
3 export default StyleSheet.create({
4   container: {
5     flex: 1,
6     justifyContent: "center",
7     alignItems: "center",
8     backgroundColor: "#F5FCFF",
9   },
10  welcome: {
11    fontSize: 20,
12    textAlign: "center",
13    margin: 10,
14  },
15  instructions: {
16    textAlign: "center",
17    color: "#333333",
18    marginBottom: 5,
19  },
20});
```

```
App.tsx M | TS store.ts U X | TS styles.ts U
offline > ts store.ts > [!] get > [!] <function>
1 import AsyncStorage from "@react-native-async-storage/async-storage";
2 import NetInfo from "@react-native-community/netinfo";
3
4 const fakeNetworkData = {
5   first: false,
6   second: false,
7   third: false,
8 };
9
10 export type Key = keyof typeof fakeNetworkData;
11
12 let connected = false;
13 const unsynced: string[] = [];
14
15 export function set(key: Key, value: boolean) {
16   return new Promise((resolve, reject) => {
17     if (connected) {
18       fakeNetworkData[key] = value;
19       resolve(true);
20     } else {
21       AsyncStorage.setItem(key, value.toString()).then(
22         () => {
23           unsynced.push(key);
24           resolve(false);
25         },
26         (err) => reject(err)
27       );
28     }
29   });
30 }
31
32 export function get(key?: Key): Promise<boolean | typeof fakeNetworkData> {
33   return new Promise((resolve, reject) => {
34     if (connected) {
35       resolve(key ? fakeNetworkData[key] : fakeNetworkData);
36     } else if (key) {
37       AsyncStorage.getItem(key)
38         .then((item) => resolve(item === "true"))
39         .catch((err) => reject(err));
40     } else {
41     });
42   });
43 }
```

```
App.tsx M | TS store.ts U X | TS styles.ts U
offline > ts store.ts > [!] get > [!] <function>
32  export function get(key?: Key): Promise<boolean | typeof fakeNetworkData> {
33    return new Promise((resolve, reject) => {
34      .then((item) => resolve(item === true))
35      .catch((err) => reject(err));
36    } else {
37      AsyncStorage.getAllKeys()
38        .then((keys) =>
39          AsyncStorage.multiGet(keys).then((items) =>
40            resolve(Object.fromEntries(items) as any)
41          )
42        )
43        .catch((err) => reject(err));
44    });
45  }
46
47  NetInfo.fetch().then(
48    (connection) => {
49      connected = ["wifi", "unknown"].includes(connection.type);
50    },
51    () => {
52      connected = false;
53    }
54  );
55
56  NetInfo.addEventListener((connection) => {
57    connected = ["wifi", "unknown"].includes(connection.type);
58
59    if (connected && unsynced.length) {
60      AsyncStorage.multiGet(unsynced).then((items) => {
61        items.forEach(([key, val]) => set(key as Key, val === "true"));
62        unsynced.length = 0;
63      });
64    }
65  });
66 }
```

```
App.tsx M X | TS store.ts U | TS styles.ts U
offline > [!] App.tsx > [!] App
1 import React, { useState, useEffect } from "react";
2 import { Text, View, Switch } from "react-native";
3 import NetInfo from "@react-native-community/netinfo";
4 import styles from "./styles";
5 import { set, get, Key } from "./store";
6
7 export default function App() {
8   const [message, setMessage] = useState<string | null>(null);
9   const [first, setFirst] = useState(false);
10  const [second, setSecond] = useState(false);
11  const [third, setThird] = useState(false);
12  const setters = new Map([
13    ["first", setFirst],
14    ["second", setSecond],
15    ["third", setThird],
16  ]);
17
18  function save(key: Key) {
19    return (value: boolean) => {
20      set(key, value).then(
21        (connected) => {
22          setters.get(key)?.(value);
23          setMessage(connected ? null : "Saved Offline");
24        },
25        (err) => {
26          setMessage(err);
27        }
28      );
29    };
30  }
31
32  useEffect(() => {
33    NetInfo.fetch().then(() =>
34      get().then(
35        (items) => {
36          for (let [key, value] of Object.entries(items)) {
37            setters.get(key)?.(value);
38          }
39        },
40        (err) => {
41          setMessage(err);
42        }
43      );
44    );
45  }, []);
46
47  return (
48    <View style={styles.container}>
49      <Text>{message}</Text>
50      <View>
51        <Text>First</Text>
52        <Switch value={first} onValueChange={save("first")}>/>
53      </View>
54      <View>
55        <Text>Second</Text>
56        <Switch value={second} onValueChange={save("second")}>/>
57      </View>
58      <View>
59        <Text>Third</Text>
60        <Switch value={third} onValueChange={save("third")}>/>
61      </View>
62    </View>
63  );
64}
```

```
App.tsx M X | TS store.ts U | TS styles.ts U
offline > [!] App.tsx > [!] App
1 import React, { useState, useEffect } from "react";
2 import { Text, View, Switch } from "react-native";
3 import NetInfo from "@react-native-community/netinfo";
4 import styles from "./styles";
5 import { set, get, Key } from "./store";
6
7 export default function App() {
8   const [message, setMessage] = useState<string | null>(null);
9   const [first, setFirst] = useState(false);
10  const [second, setSecond] = useState(false);
11  const [third, setThird] = useState(false);
12  const setters = new Map([
13    ["first", setFirst],
14    ["second", setSecond],
15    ["third", setThird],
16  ]);
17
18  function save(key: Key) {
19    return (value: boolean) => {
20      set(key, value).then(
21        (connected) => {
22          setters.get(key)?.(value);
23          setMessage(connected ? null : "Saved Offline");
24        },
25        (err) => {
26          setMessage(err);
27        }
28      );
29    };
30  }
31
32  useEffect(() => {
33    NetInfo.fetch().then(() =>
34      get().then(
35        (items) => {
36          for (let [key, value] of Object.entries(items)) {
37            setters.get(key)?.(value);
38          }
39        },
40        (err) => {
41          setMessage(err);
42        }
43      );
44    );
45  }, []);
46
47  return (
48    <View style={styles.container}>
49      <Text>{message}</Text>
50      <View>
51        <Text>First</Text>
52        <Switch value={first} onValueChange={save("first")}>/>
53      </View>
54      <View>
55        <Text>Second</Text>
56        <Switch value={second} onValueChange={save("second")}>/>
57      </View>
58      <View>
59        <Text>Third</Text>
60        <Switch value={third} onValueChange={save("third")}>/>
61      </View>
62    </View>
63  );
64}
```



# LA GESTION DES ETATS - REACT NATIVE

## STATE "ETATS" ET PROPS "PROPRIÉTÉS"

QU'EST-CE QUE L'ÉTAT ET EN QUOI EST-IL DIFFÉRENT DES PROPS ?

```
import React from "react";
import { View, Text, Pressable } from "react-native";
export const ManagedButton = () => {
  return (
    <View>
      <Text>this text will display the current status</Text>
      <Pressable onPress="">
        <Text>Press here to check/uncheck</Text>
      </Pressable>
    </View>
  );
};
```

```
import React, { useState } from "react";
import { View, Text, Pressable } from "react-native";
export const ManagedButton = () => {
  const [checkedState, setCheckedState] = useState("unchecked");
  return (
    <View>
      <Text>this text will display the current status, which is: {checkedState}</Text>
      <Pressable onPress="">
        <Text>Press here to check/uncheck</Text>
      </Pressable>
    </View>
  );
};
```

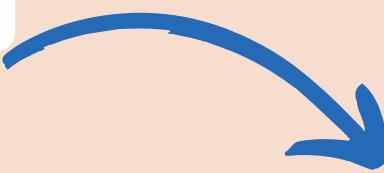
# LA GESTION DES ETATS - REACT NATIVE

## STATE "ETATS" ET PROPS "PROPRIÉTÉS"

QU'EST-CE QUE L'ÉTAT ET EN QUOI EST-IL DIFFÉRENT DES PROPS ?

```
import React, { useState } from "react";
import { View, Text, Pressable } from "react-native";
const ManagedText = ({checkedState}) => {
  return (
    <Text>this text will display the current status, which is: {checkedState}</Text>
  );
};

export const ParentComponent = () => {
  const [checkedState, setCheckedState] = useState("unchecked");
  return (
    <View>
      <ManagedText checkedState={checkedState} />
      <Pressable onPress={() => setCheckedState("checked")}>
        <Text>Press here to check/uncheck</Text>
      </Pressable>
    </View>
  );
};
```



```
const ManagedText = (fancyComponentStuff) => {
  return (
    <Text>this text will display the current status, which is:{fancyComponentStuff}</Text>
  );
};

export const ParentComponent = () => {
  const [checkedState, setCheckedState] = useState("unchecked");
  return (
    <View>
      <ManagedText fancyComponentStuff={checkedState} />
      <Pressable onPress={setCheckedState("checked")}>
        <Text>Press here to check/uncheck</Text>
      </Pressable>
    </View>
  );
};
```

La chose la plus importante à retenir à propos des propos est la suivante : **les props sont immuables (ou en lecture seule)**.

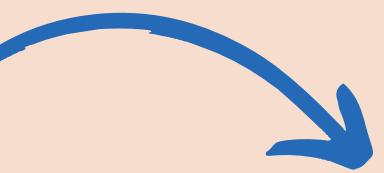
# LA GESTION DES ETATS - REACT NATIVE

## LES HOOKS

### USESTATE ET USEEFFECT

```
const LikesParentComponent = () => {
  const getCounterNumberFromApi = someFunctionRetrievingDataFromAPI();
  const [counterNumber, setCounterNumber] = useState(getCounterNumberFromApi)
  return (
    <LikesComponent counterNumber={counterNumber} />
  );
};

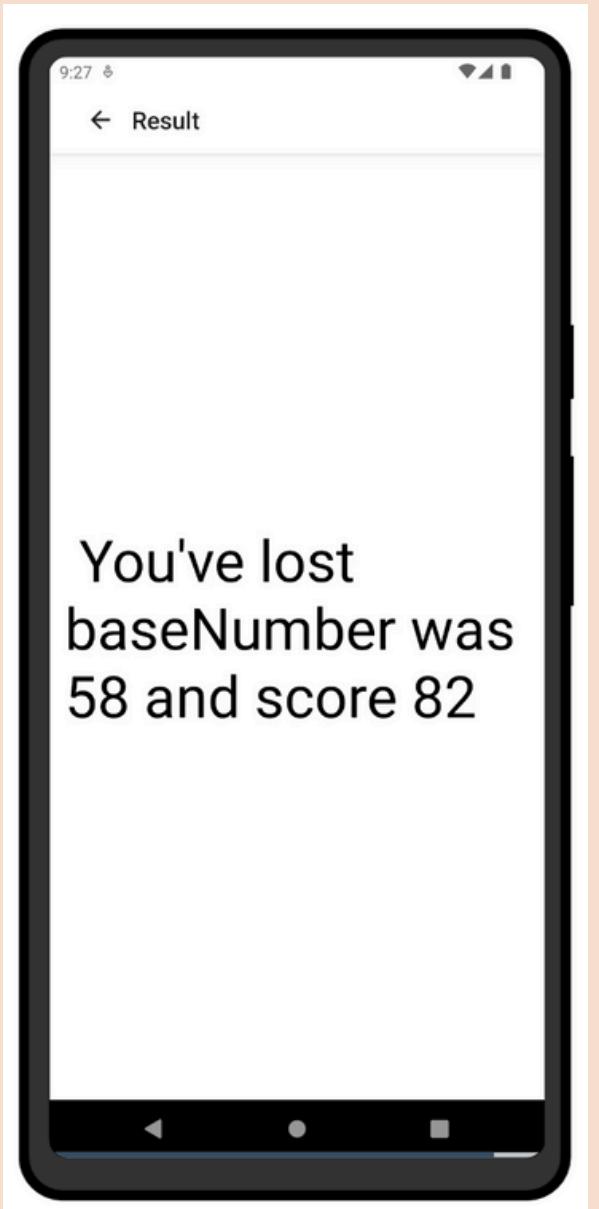
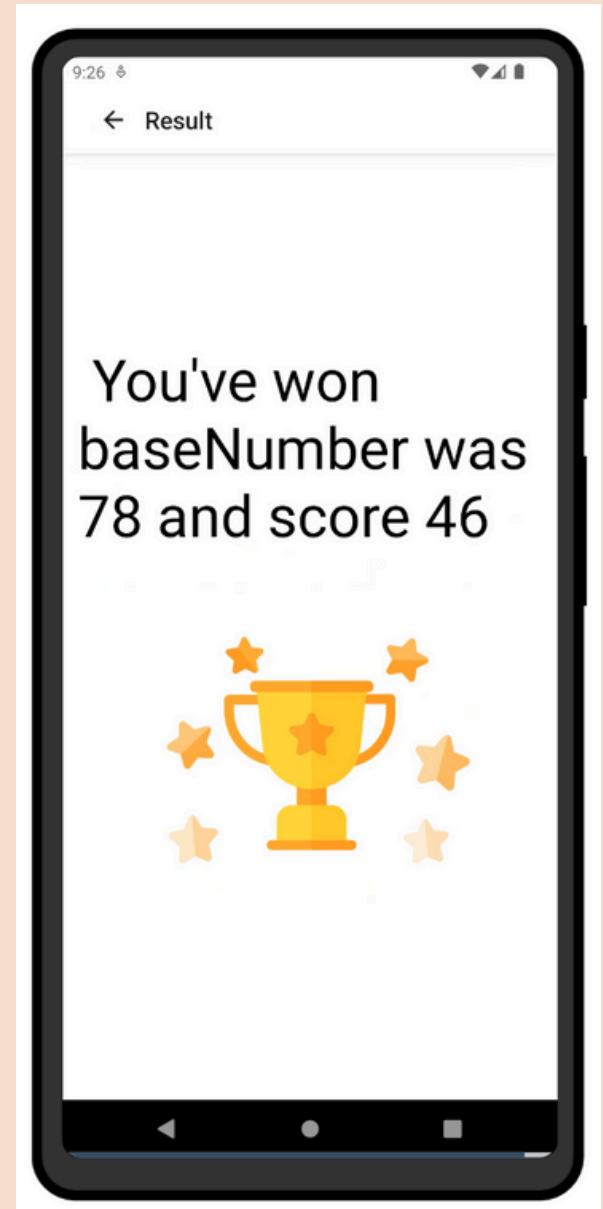
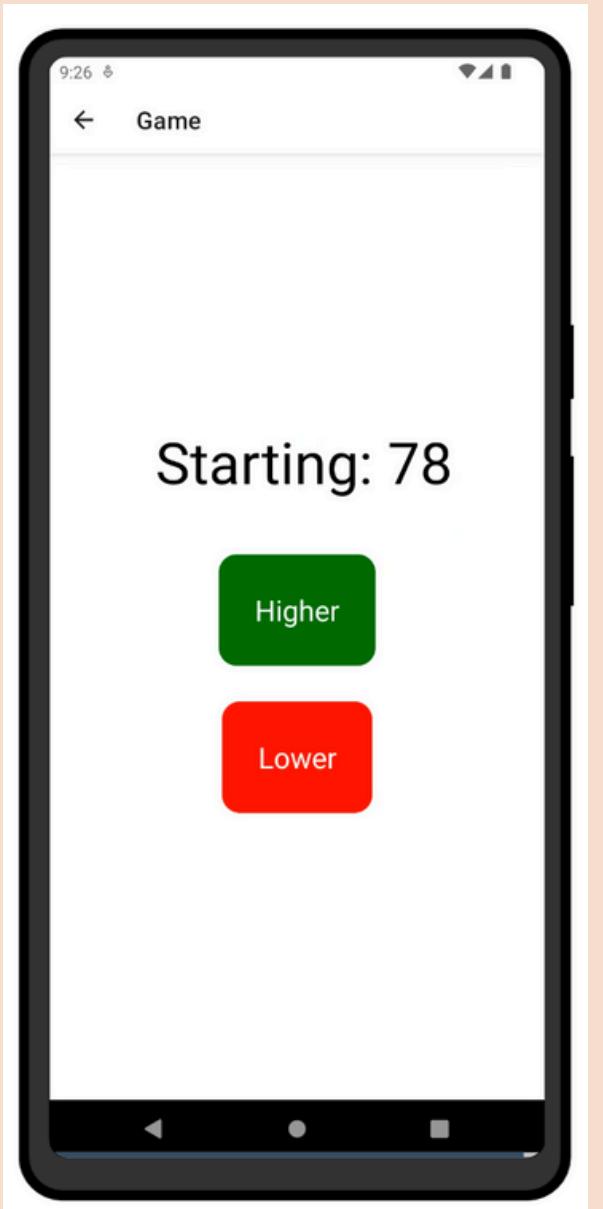
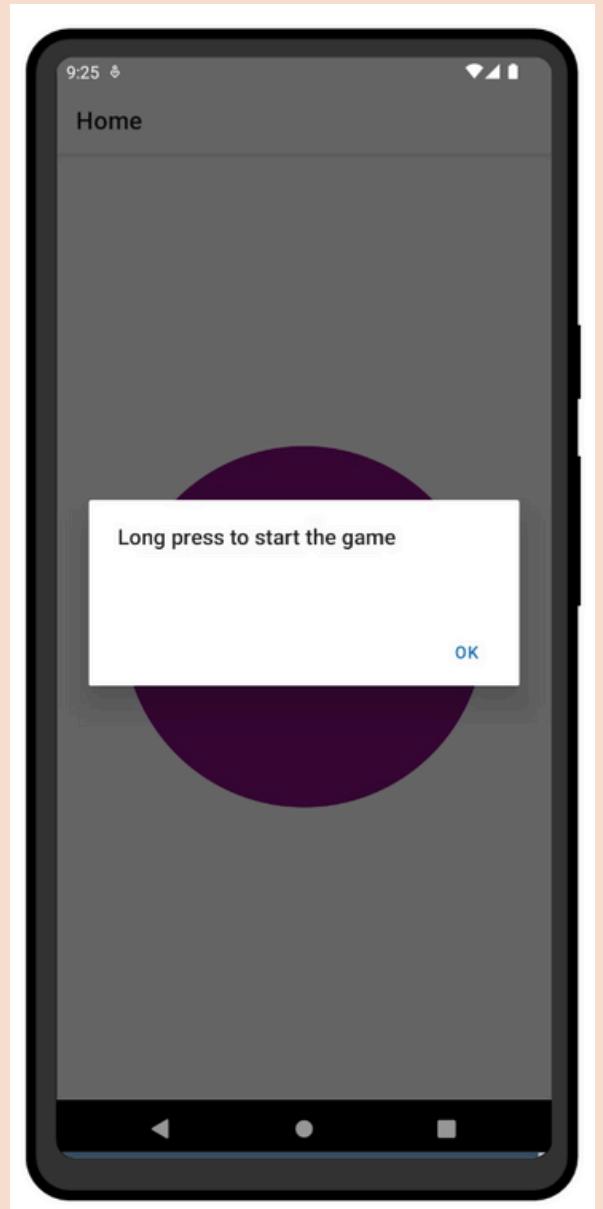
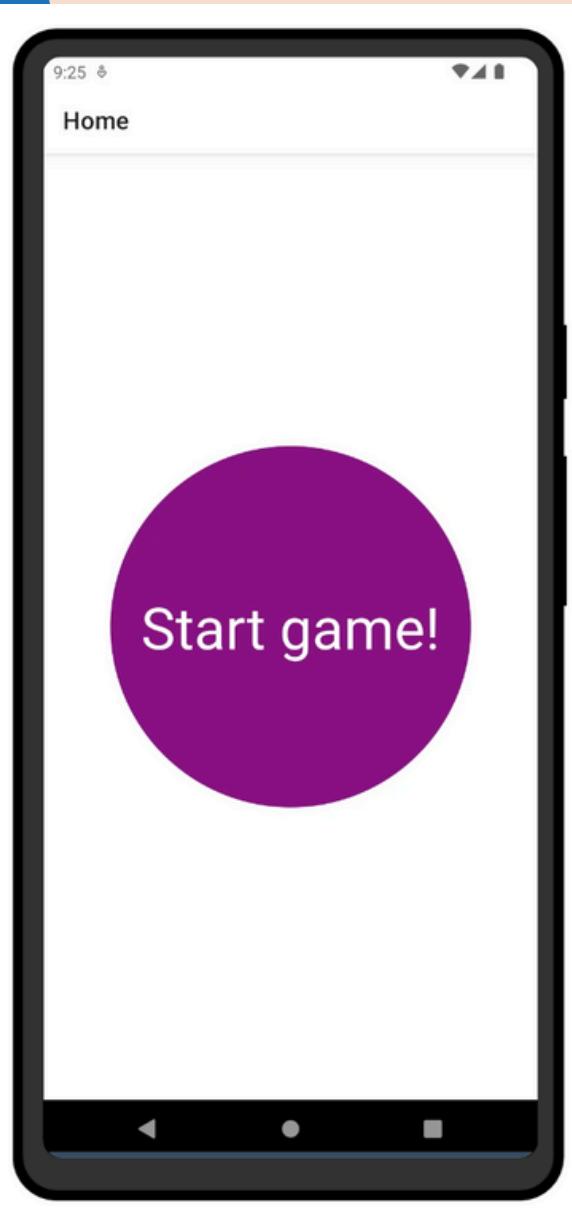
const LikesComponent = (counterNumber) => {
  const [likeState, setLikedState] = useState ("haven't yet liked");
  return (
    <View>
      <Text>you {likeState} this post</Text>
      <Pressable onPress={setLikedState("liked")}>
        <Text>Press here to check/uncheck</Text>
      </Pressable>
      <Text>{counterNumber} other people liked this
    </View>
  );
};
```



```
const LikesComponent = (counterNumber, setCountNumber) => {
  const [likeState, setLikedState] = useState ("haven't yet liked");
  useEffect(() => {
    if (likeState === "liked") {
      setCounterNumber(counterNumber++)
    } else {
      setCounterNumber(counterNumber-1)
    }
  }, [likeState])
  return (
    <View>
      <Text>you {likeState} this post</Text>
      <Pressable onPress={setLikedState("liked")}>
        <Text>Press here to check/uncheck</Text>
      </Pressable>
      <Text>{counterNumber} other people liked this
    </View>
  );
};
```

# CAS D'APPLICATION - JEUX

## PRÉSENTATION DU PROJET



# CAS D'APPLICATION - JEUX

## MISE EN PLACE DU PROJET



**animated game app** ⓘ  
Last saved about 6 hours ago. [See previous saves.](#)

App.js  
babel.config.js

```
1 module.exports = function(api) {
2   api.cache(true);
3   return {
4     presets: ['babel-preset-expo'],
5   };
6 }
```

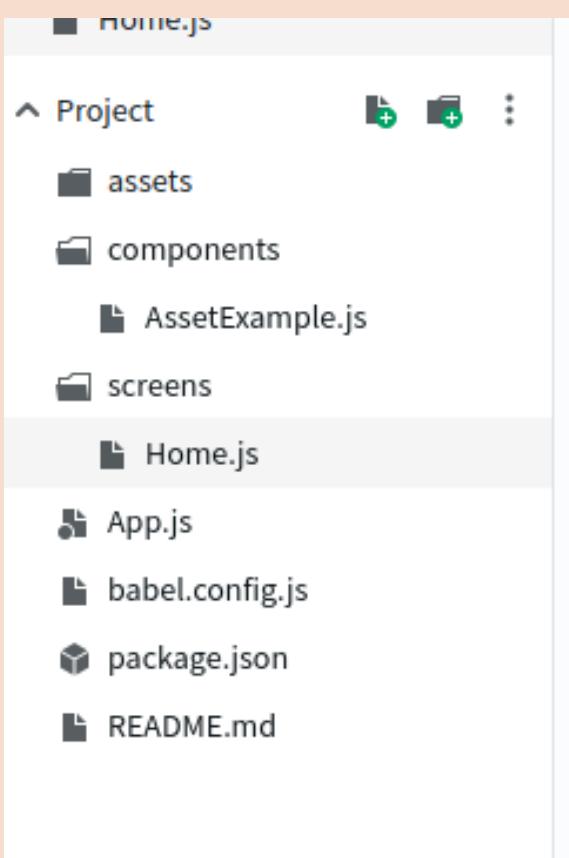
```
16 days ago. ✓
1
2 "dependencies": {
3   "react-native-paper": "4.9.2",
4   "@expo/vector-icons": "^14.0.2",
5   "@react-navigation/elements": "1.3.3",
6   "@react-navigation/native": "6.0.10",
7   "@react-navigation/native-stack": "6.6.1",
8   "expo-status-bar": "~1.12.1",
9   "lottie-react-native": "6.7.0",
10  "react-native-gesture-handler": "~2.16.1",
11  "react-native-safe-area-context": "4.10.5",
12  "react-native-screens": "3.31.1",
13  "@dotlottie/react-player": "1.6.1",
14  "@lottiefiles/react-lottie-player": "3.5.3"
15 }
16 }
```

```
16 days ago. ✓
1 import { StatusBar } from 'expo-status-bar';
2 import React from 'react';
3 import { StyleSheet, Text, View, SafeAreaView } from 'react-native';
4 import { NavigationContainer } from '@react-navigation/native';
5 import { createNativeStackNavigator } from '@react-navigation/native-stack';
6 const Stack = createNativeStackNavigator();

7
8
9 export default function App() {
10   return [
11     <SafeAreaView>
12       <Text>
13         Change code in the editor and watch it change on your phone! Save to get a shareable url.
14       </Text>
15     </SafeAreaView>
16   ];
17 }
```

# CAS D'APPLICATION - JEUX

ÉCRAN HOME



```
1 import React from 'react';
2 import { StyleSheet, Text, View, StatusBar } from 'react-native';
3 export default function Home() {
4   return (
5     <View style={styles.container}>
6       <StatusBar barStyle="dark-content" />
7       <Text> home screen</Text>
8     </View>
9   );
10 }
// export default function Home() {
11 //   return (
12 //     <View style={styles.container}>
13 //       <StatusBar barStyle="dark-content" />
14 //       <Text> home screen</Text>
15 //     </View>
16 //   );
17 //}
18 const styles = StyleSheet.create({
19   container: {
20     flex: 1,
21     backgroundColor: '#fff',
22     alignItems: 'center',
23     justifyContent: 'center',
24   },
25 });
26 }
```

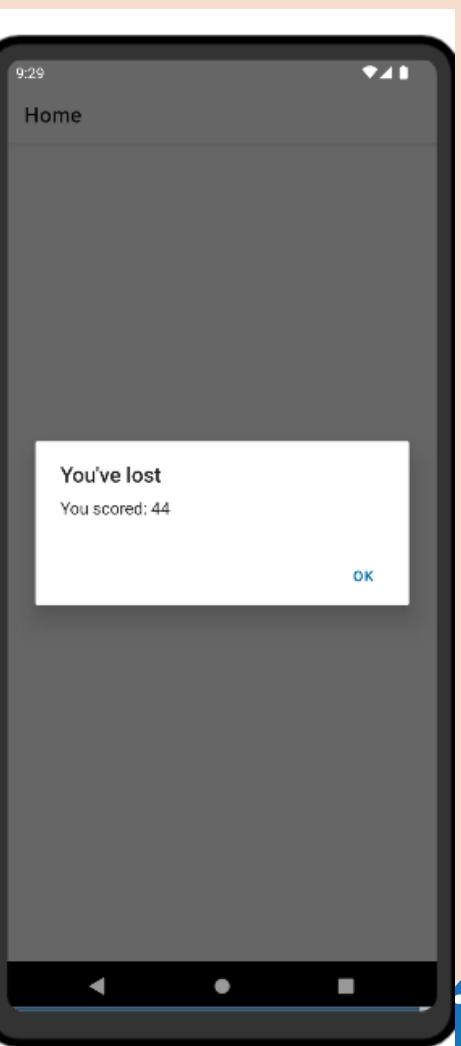
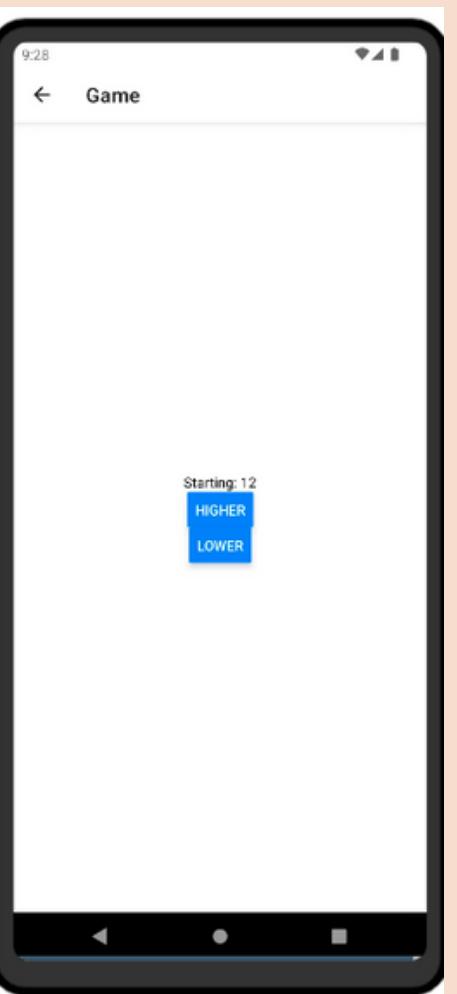
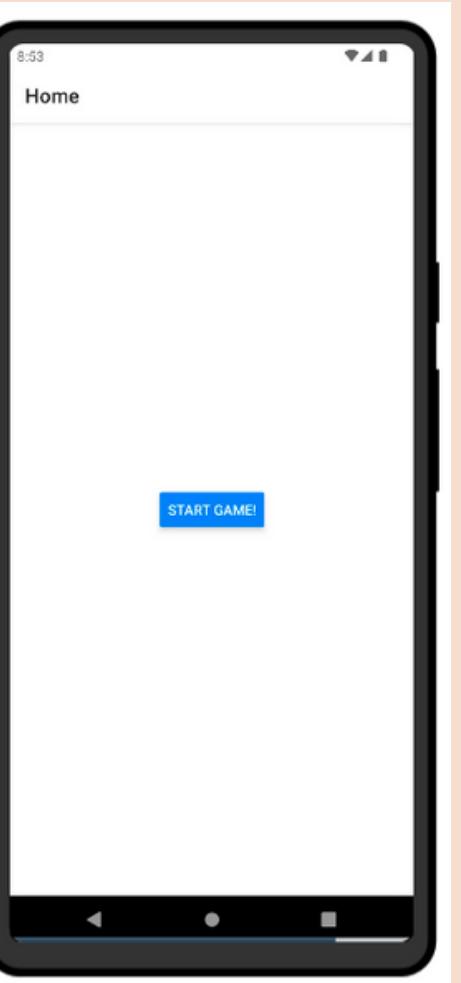


# CAS D'APPLICATION - JEUX

```
1 import { StatusBar } from 'expo-status-bar';
2 import React from 'react';
3 import { StyleSheet } from 'react-native';
4 import { NavigationContainer } from '@react-navigation/native';
5 import { createNativeStackNavigator } from '@react-navigation/native-stack';
6 const Stack = createNativeStackNavigator();
7 import Home from './screens/Home';
8 import Game from './screens/Game';
9
10
11 export default function App() {
12   return [
13     <NavigationContainer>
14       <StatusBar style='auto' />
15       <Stack.Navigator initialRouteName='Home'>
16         <Stack.Screen name='Home' component={Home} />
17         <Stack.Screen name='Game' component={Game} />
18       </Stack.Navigator>
19     ]];
20 }
21 }
```

```
1 import React from 'react';
2 import { StyleSheet, Button, View, StatusBar } from 'react-native';
3 import { useNavigation } from '@react-navigation/native';
4
5 export default function Home() {
6   const navigation = useNavigation();
7
8   return (
9     <View style={styles.container}>
10      <StatusBar barStyle="dark-content" />
11      <Button onPress={() => navigation.navigate(
12        'Game'
13      )} title='Start game!' />
14    </View>
15  );
16
17 const styles = StyleSheet.create({
18   container: {
19     flex: 1,
20     backgroundColor: '#fff',
21     alignItems: 'center',
22     justifyContent: 'center',
23   },
24 });
25
26
27
28
29
30
31
32
33
34
35
```

```
1 import React, { useEffect, useState } from 'react';
2 import { StyleSheet, Text, View, StatusBar, Button, Alert } from 'react-native';
3 import { useNavigation } from '@react-navigation/native';
4 export default function Game() {
5
6   const [choice, setChoice] = useState('');
7   const baseNumber = Math.floor(Math.random() * 100);
8   const score = Math.floor(Math.random() * 100);
9
10  const navigation = useNavigation();
11  useEffect(() => {
12
13    if (choice) {
14      console.log(` base Number = ${baseNumber} et score = ${score}`);
15      const winner =
16        (choice === 'higher' && score > baseNumber) ||
17        (choice === 'lower' && baseNumber > score);
18      Alert.alert(`You've ${winner ? 'won' : 'lost'}`,
19        `You scored: ${score}`);
20      navigation.goBack();
21    }
22  }, [baseNumber, score, choice]);
23
24  return (
25    <View style={styles.container}>
26      <StatusBar barStyle="dark-content" />
27      <Text>Starting: {baseNumber}</Text>
28      <Button onPress={() => setChoice('higher')} title='Higher' />
29      <Button onPress={() => setChoice('lower')} title='Lower' />
30    </View>
31  );
32
33
34 const styles = StyleSheet.create({
35   container: {
```



# CAS D'APPLICATION - JEUX

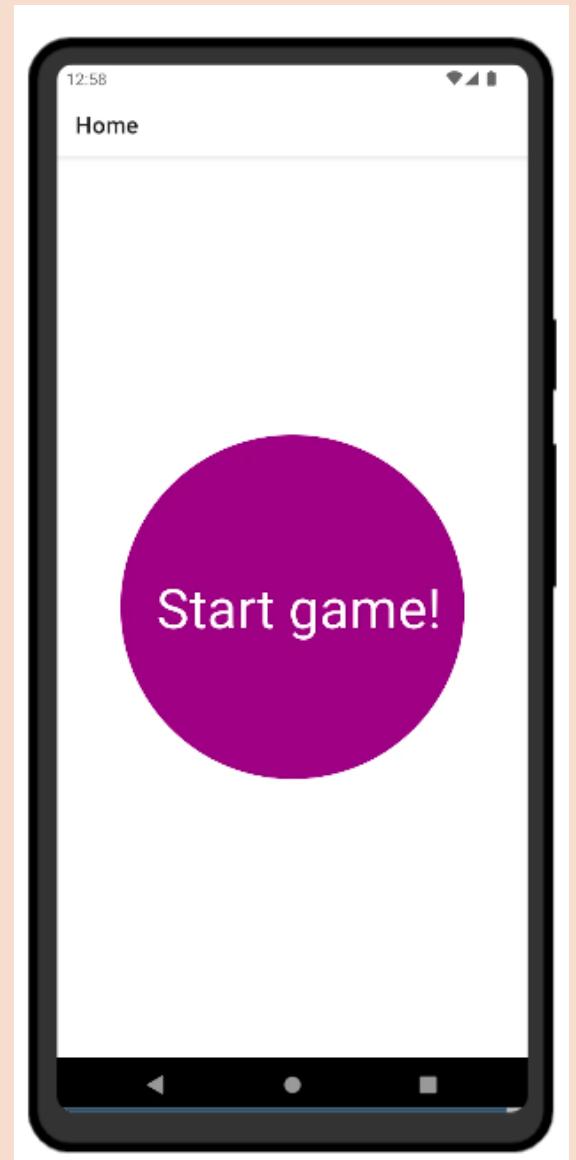
STYLE HOME

```
import React from 'react';
import { StyleSheet, Button, View, StatusBar, TouchableHighlight, Text } from 'react-native';
import { useNavigation } from '@react-navigation/native';

export default function Home() {
  const navigation = useNavigation();

  return (
    <View style={styles.container}>
      <StatusBar barStyle="dark-content" />
      <TouchableHighlight onPress={() => navigation.navigate('Game')} style={styles.button} >
        <Text style={styles.buttonText}> Start game! </Text>
      </TouchableHighlight>
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#fff',
    alignItems: 'center',
    justifyContent: 'center',
  },
  button: {
    width: 300,
    height: 300,
    display: 'flex',
    alignItems: 'center',
    justifyContent: 'space-around',
    borderRadius: 150,
    backgroundColor: 'purple',
  },
  buttonText: {
    color: 'white',
    fontSize: 48,
  },
});
```



# CAS D'APPLICATION - JEUX

## STYLE GAME

```
import React, { useEffect, useState } from 'react';
import { StyleSheet, Text, View, StatusBar, TouchableHighlight, Alert } from 'react-native';
import { useNavigation } from '@react-navigation/native';
export default function Game() {

  const [choice, setChoice] = useState('');

  const baseNumber = Math.floor(Math.random() * 100);
  const score = Math.floor(Math.random() * 100);

  console.log(`beginning base Number = ${baseNumber} et score = ${score}`)

  const navigation = useNavigation();
  useEffect(() => {

    if (choice) {
      console.log(`in useeffect base Number = ${baseNumber} et score = ${score}`)
      const winner =
        (choice === 'higher' && score > baseNumber) ||
        (choice === 'lower' && baseNumber > score);
      Alert.alert(`You've ${winner ? 'won' : 'lost'}`, `You scored: ${score}`);
    }

    navigation.goBack();
  }, [baseNumber, score, choice])

  return (
    <View style={styles.container}>
      <StatusBar barStyle="dark-content" />
      <Text style={styles.baseNumber}>
        Starting: {baseNumber}</Text>
      <TouchableHighlight onPress={() =>
        setChoice('higher')}>
        <Text style={styles.buttonText}>Higher</Text>
      </TouchableHighlight>
      <TouchableHighlight onPress={() =>
        setChoice('lower')}>
        <Text style={styles.buttonText}>Lower</Text>
      </TouchableHighlight>
    </View>
  );
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#fff',
    alignItems: 'center',
    justifyContent: 'center'
  },
  baseNumber: {
    fontSize: 48,
    marginBottom: 30,
  },
  button: {
    display: 'flex',
    alignItems: 'center',
    justifyContent: 'space-around',
    borderRadius: 15,
    padding: 30,
    marginVertical: 15,
  },
  buttonRed: {
    backgroundColor: 'red',
  },
  buttonGreen: {
    backgroundColor: 'green',
  },
  buttonText: {
    color: 'white',
    fontSize: 24,
  },
});
```



Starting: 57

Higher

Lower

# CAS D'APPLICATION - JEUX

## LES ANIMATIONS

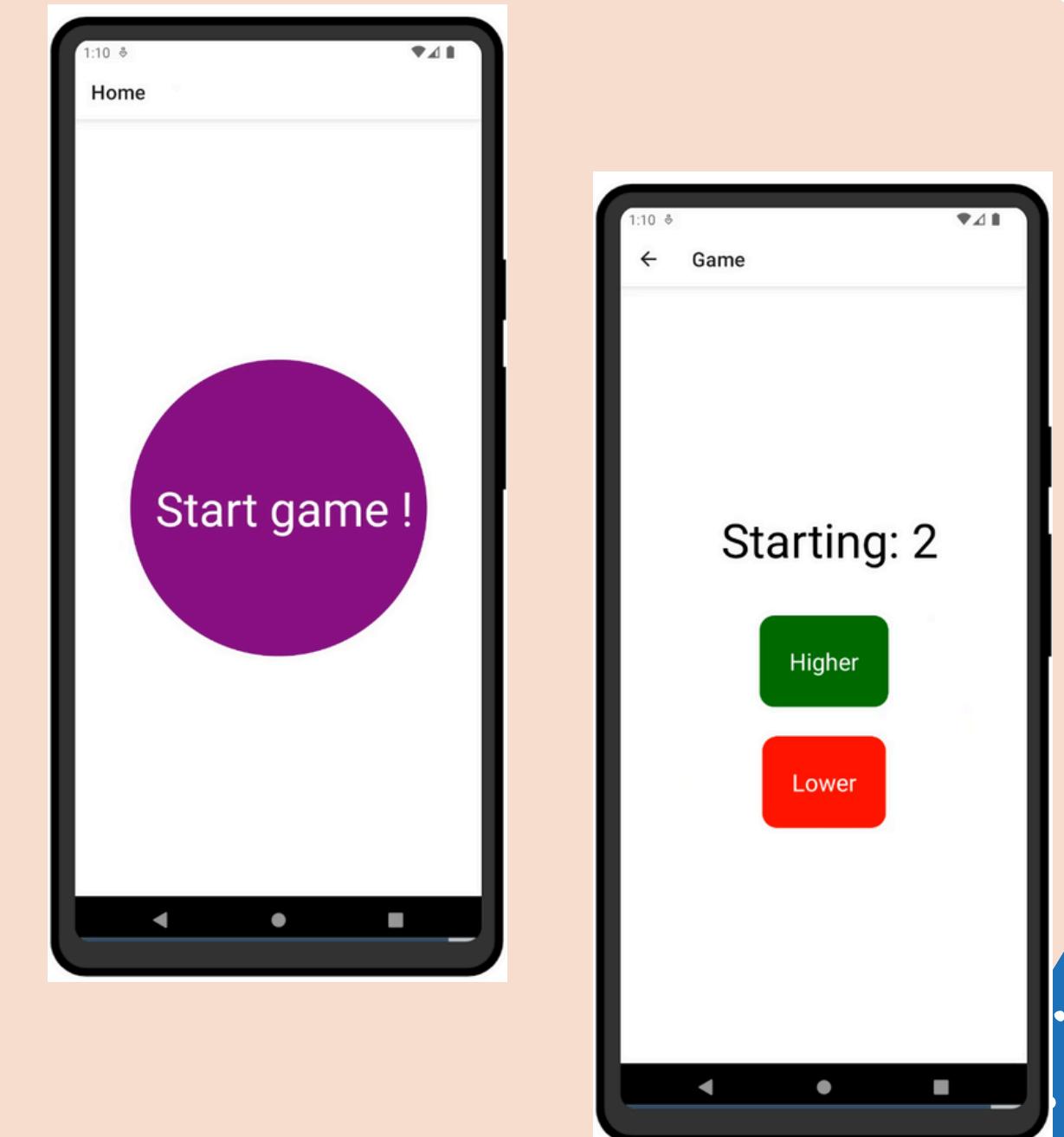
The image shows a code editor interface with two tabs open. The left tab is 'AnimatedButton.js' and the right tab is 'Game.js'. Both tabs show code in a syntax-highlighted editor.

```
AnimatedButton.js:
```

```
1 import React from 'react';
2 import { StyleSheet, Text, TouchableHighlight }
3   from 'react-native';
4 export default function AnimatedButton({ action,
5   onPress }) {
6   return (
7     <TouchableHighlight
8       onPress={onPress}
9       underlayColor="lightblue"
10      style={[
11        styles.button,
12        action === 'higher' ? styles.buttonGreen :
13          styles.buttonRed,
14      ]}
15     >
16       <Text style={styles.buttonText}>{action}</Text>
17     </TouchableHighlight>
18   );
19 }
20 const styles = StyleSheet.create({
21   button: [
22     display: 'flex',
23     alignItems: 'center',
24     justifyContent: 'space-around',
25     borderRadius: 15,
26     padding: 30,
27     marginVertical: 15,
28   ],
29   buttonRed: {
30     backgroundColor: 'red',
31   },
32   buttonGreen: {
33     backgroundColor: 'green',
34   },
35   buttonText: {
36     color: 'white',
37     fontSize: 24,
38     textTransform: 'capitalize',
39   },
40 });
41 
```

```
Game.js:
```

```
1 import React, { useEffect, useState } from 'react';
2 import { StyleSheet, Text, View, StatusBar, Alert } from 'react-native';
3 import { useNavigation } from '@react-navigation/native';
4 import AnimatedButton from '../components/AnimatedButton';
5
6 export default function Game() {
7   const [choice, setChoice] = useState('');
8
9   const baseNumber = Math.floor(Math.random() * 100);
10  const score = Math.floor(Math.random() * 100);
11
12  console.log(`beginning base Number = ${baseNumber} et score = ${score}`);
13
14  const navigation = useNavigation();
15  useEffect(() => {
16
17    if (choice) {
18      console.log(`in useeffect base Number = ${baseNumber} et score = ${score}`);
19      const winner =
20        (choice === 'higher' && score > baseNumber) ||
21        (choice === 'lower' && baseNumber > score);
22      Alert.alert(`You've ${winner ? 'won' : 'lost'}`, `You scored: ${score}`);
23
24      navigation.goBack();
25    }
26  }, [baseNumber, score, choice]);
27
28  return (
29    <View style={styles.container}>
30      <StatusBar barStyle="dark-content" />
31      <Text style={styles.baseNumber}> Starting: {baseNumber}</Text>
32      <AnimatedButton action='higher' onPress={() => setChoice('higher')} />
33      <AnimatedButton action='lower' onPress={() => setChoice('lower')} />
34    </View>
35  );
36
37  const styles = StyleSheet.create({
38    container: {
39      flex: 1,
40      backgroundColor: '#fff',
41      alignItems: 'center',
42      justifyContent: 'center'
43    },
44    baseNumber: {
45      fontSize: 48,
46      marginBottom: 30,
47    }
48 });
49 
```

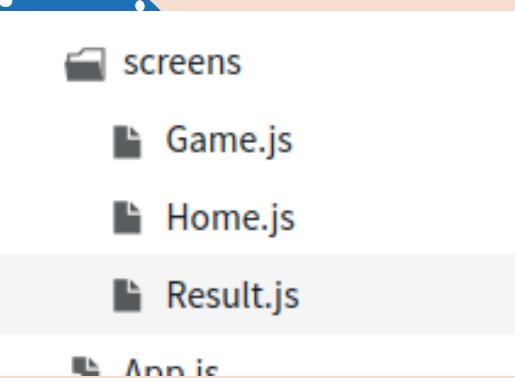


# CAS D'APPLICATION - JEUX

```
1 import React, {useRef} from 'react';
2 import { StyleSheet, Text, TouchableOpacity, Animated, View } from 'react-native';
3
4 export default function AnimatedButton({ action, onPress }) {
5   return (
6     <TouchableOpacity onPress={onPress}>
7       <View style={[ styles.button, action === 'higher'
8         ? styles.buttonGreen : styles.buttonRed ]}>
9         <Text style={styles.buttonText}>{action}</Text>
10      </View>
11    </TouchableOpacity>
12  );
13}
14
15 const styles = StyleSheet.create({
16   button: {
17     display: 'flex',
18     alignItems: 'center',
19     justifyContent: 'space-around',
20     borderRadius: 15,
21     padding: 30,
22     marginVertical: 15,
23   },
24   buttonRed: [
25     backgroundColor: 'red',
26   ],
27   buttonGreen: [
28     backgroundColor: 'green',
29   ],
30   buttonText: {
31     color: 'white',
32     fontSize: 24,
33     textTransform: 'capitalize',
34   },
35 });
36
```

```
See previous saves. ✓
1 import React, {useRef} from 'react';
2 import { StyleSheet, Text, TouchableOpacity, Animated, View } from 'react-native';
3
4 export default function AnimatedButton({ action, onPress }) {
5
6   const opacity = useRef(new Animated.Value(1));
7
8   return (
9     <TouchableOpacity onPress={() => {
10       Animated.timing(opacity.current, {
11         toValue: 0.2,
12         duration: 800,
13         useNativeDriver: true,
14       }).start(() => onPress());
15     }}
16   >
17     <Animated.View
18       style={[
19         styles.button,
20         action === 'higher' ? styles.buttonGreen :
21           styles.buttonRed,
22         { opacity: opacity.current },
23       ]}
24     >
25       <Text style={styles.buttonText}>{action}</Text>
26     </Animated.View>
27   </TouchableOpacity>
28 );
29
30 const styles = StyleSheet.create([
31   button: {
32     display: 'flex',
33     alignItems: 'center',
34     justifyContent: 'space-around',
35     borderRadius: 15,
36     padding: 30,
37     marginVertical: 15,
38   },
39   buttonRed: [
40     backgroundColor: 'red',
41   ],
42   buttonGreen: [
43     backgroundColor: 'green',
44   ],
45   buttonText: {
46     color: 'white',
47     fontSize: 24,
48     textTransform: 'capitalize',
49   },
50 ]);
51
52
53
54
```

# CAS D'APPLICATION - JEUX

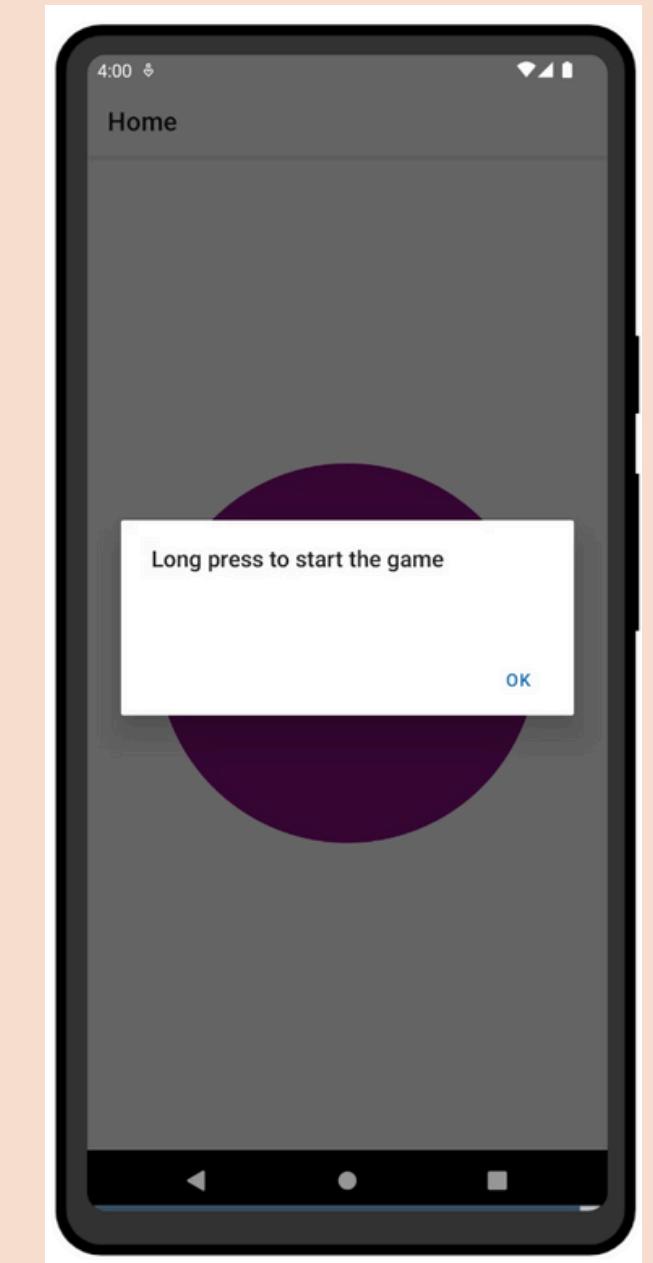


```
us saves. ✓  
1 import React from 'react';  
2 import { StyleSheet, Text, View } from 'react-native';  
3 export default function Result() {  
4   return (  
5     <View style={styles.container}>  
6       <Text>Result screen </Text>  
7     </View>  
8   );  
9 }  
10 const styles = StyleSheet.create({  
11   container: {  
12     flex: 1,  
13     backgroundColor: '#fff',  
14     alignItems: 'center',  
15     justifyContent: 'center',  
16   },  
17 });
```

```
1 import { StatusBar } from 'expo-status-bar';  
2 import React from 'react';  
3 import { StyleSheet } from 'react-native';  
4 import { NavigationContainer } from '@react-navigation/native';  
5 import { createNativeStackNavigator } from '@react-navigation/native-stack';  
6 import { HeaderBackButton } from '@react-navigation/elements';  
7 const Stack = createNativeStackNavigator();  
8 import Home from './screens/Home';  
9 import Game from './screens/Game';  
10 import Result from './screens/Result';  
11  
12  
13 export default function App() {  
14   return (  
15     <NavigationContainer>  
16       <StatusBar style='auto' />  
17       <Stack.Navigator initialRouteName='Home'>  
18         <Stack.Screen name='Home' component={Home} />  
19         <Stack.Screen name='Game' component={Game} />  
20         <Stack.Screen  
21           name='Result'  
22           component={Result}  
23           options={({ navigation }) => ({  
24             headerLeft: (props) => (  
25               <HeaderBackButton  
26                 {...props}  
27                 label='Home'  
28                 onPress={() =>  
29                   navigation.navigate('Home')  
30                 }  
31               ),  
32             })}  
33           />  
34         </Stack.Navigator>  
35       </NavigationContainer>  
36     );  
37   }  
38  
39 }
```

# CAS D'APPLICATION - JEUX

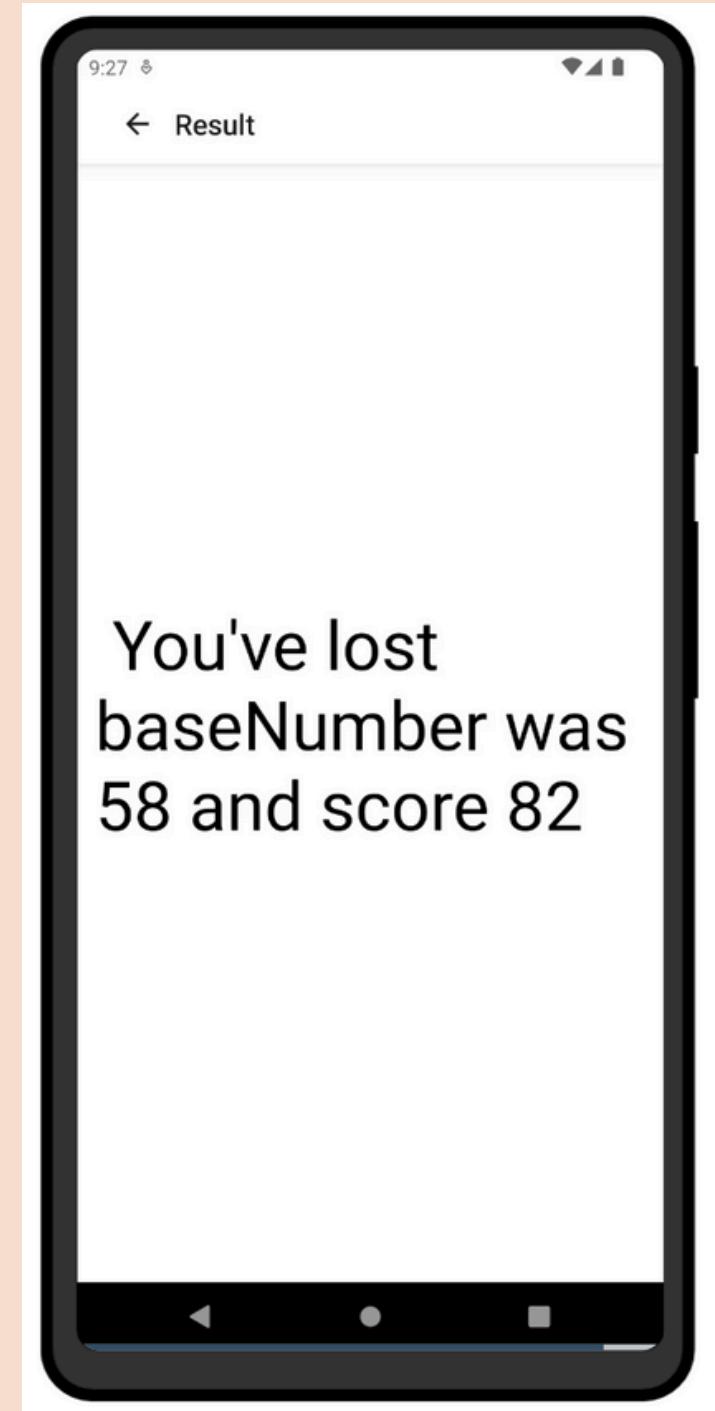
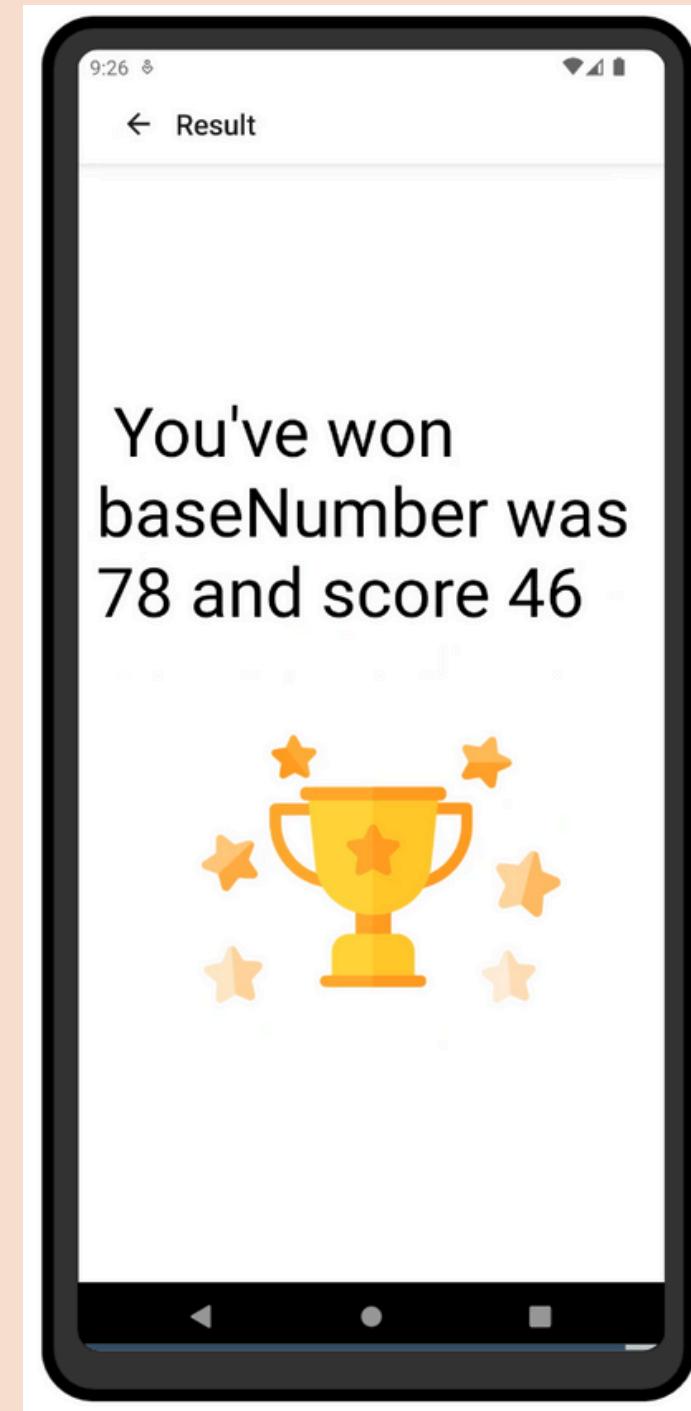
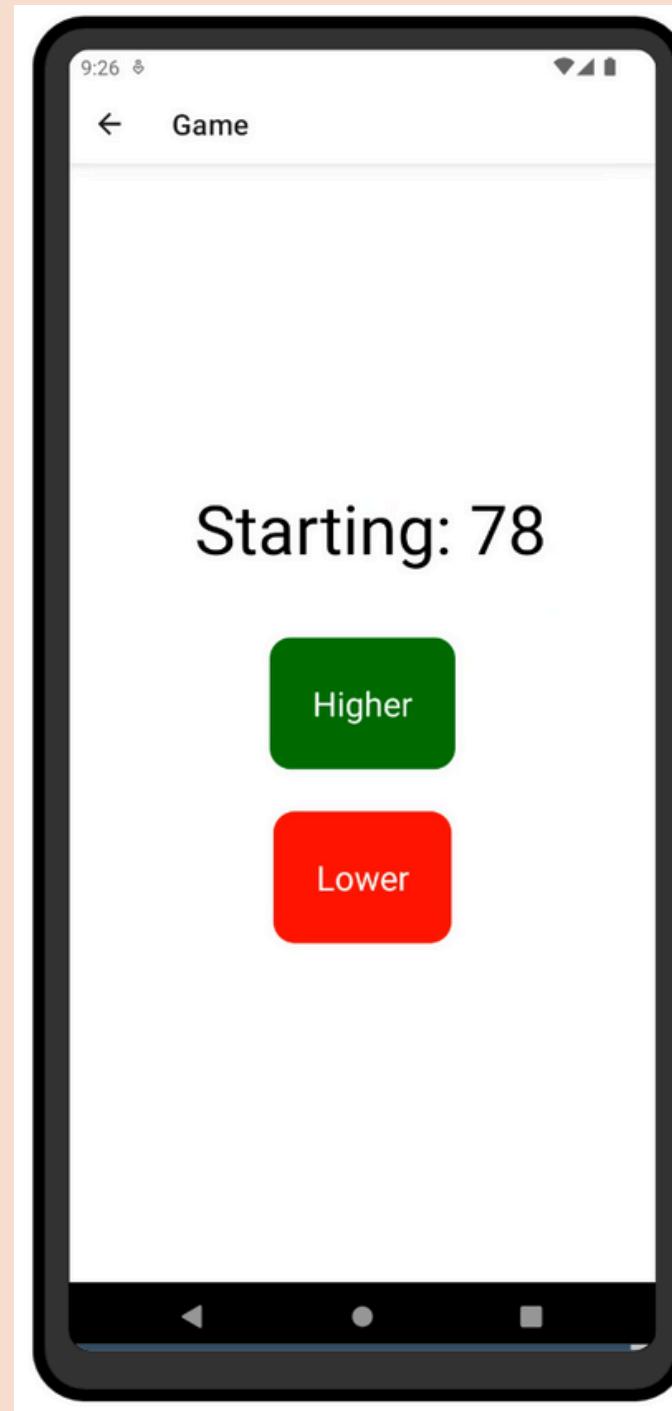
```
1 import React from 'react';
2 import { StyleSheet, Text, View, Alert } from 'react-native';
3 import { useNavigation } from '@react-navigation/native';
4 import {
5   GestureHandlerRootView,
6   LongPressGestureHandler,
7   TapGestureHandler,
8   State,
9 } from 'react-native-gesture-handler';
10
11 export default function Home() {
12   const navigation = useNavigation();
13
14   function onLongPress(e) {
15     if (e.nativeEvent.state === State.ACTIVE) {
16       navigation.navigate('Game');
17     }
18   }
19
20   function onTap(e) {
21     if (e.nativeEvent.state === State.ACTIVE) {
22       Alert.alert('Long press to start the game');
23     }
24   }
25
26   return [
27     <GestureHandlerRootView style={{ flex: 1 }}>
28       <View style={styles.container}>
29         <TapGestureHandler onHandlerStateChange={onTap}>
30           <LongPressGestureHandler
31             onHandlerStateChange={onLongPress}
32             minDurationMs={600}
33           >
34             <View style={styles.button}>
35               <Text style={styles.buttonText}>Start game!</Text>
36             </View>
37           </LongPressGestureHandler>
38         </TapGestureHandler>
39       </View>
40     </GestureHandlerRootView>
41   ];
42 }
43
44 const styles = StyleSheet.create({
45   container: {
46     flex: 1,
47     backgroundColor: '#fff',
48     alignItems: 'center',
49     justifyContent: 'center',
50   },
51   button: {
52     width: 300,
53     height: 300,
54     display: 'flex',
55     alignItems: 'center',
56     justifyContent: 'space-around',
57     borderRadius: 150,
58     backgroundColor: 'purple',
```



# CAS D'APPLICATION - JEUX

```
1 import React from 'react';
2 import { StyleSheet, Text, View } from 'react-native';
3 import LottieView from 'lottie-react-native';
4 import { useRoute } from '@react-navigation/native';
5 export default function Result() {
6   const route = useRoute();
7   const { winner, baseNumber, score } = route.params;
8   return (
9     <View style={styles.container}>
10       <Text style={styles.message}> You've {winner ? `won baseNumber was ${baseNumber} and score ${score}` : `lost baseNumber was ${baseNumber} and score ${score}`} </Text>
11       {winner && (
12         <LottieView
13           autoPlay
14           style={{
15             width: 300,
16             height: 300,
17           }}
18           source={require('../assets/winner.json')}
19         />
20       )}
21     </View>
22   );
23 }
24 const styles = StyleSheet.create({
25   container: {
26     flex: 1,
27     backgroundColor: '#fff',
28     alignItems: 'center',
29     justifyContent: 'center',
30   },
31   message: {
32     fontSize: 48,
33   }
34 });
```

# CAS D'APPLICATION - JEUX

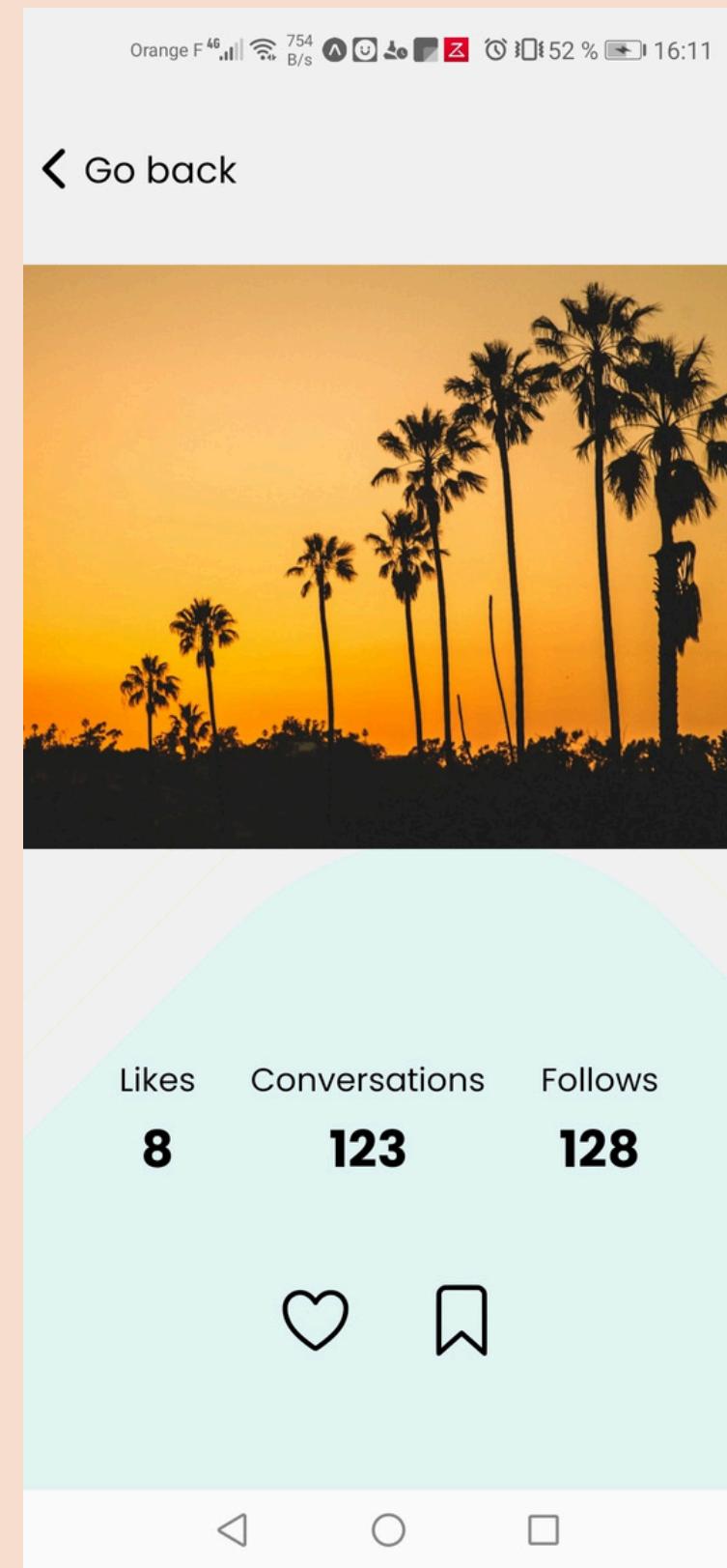
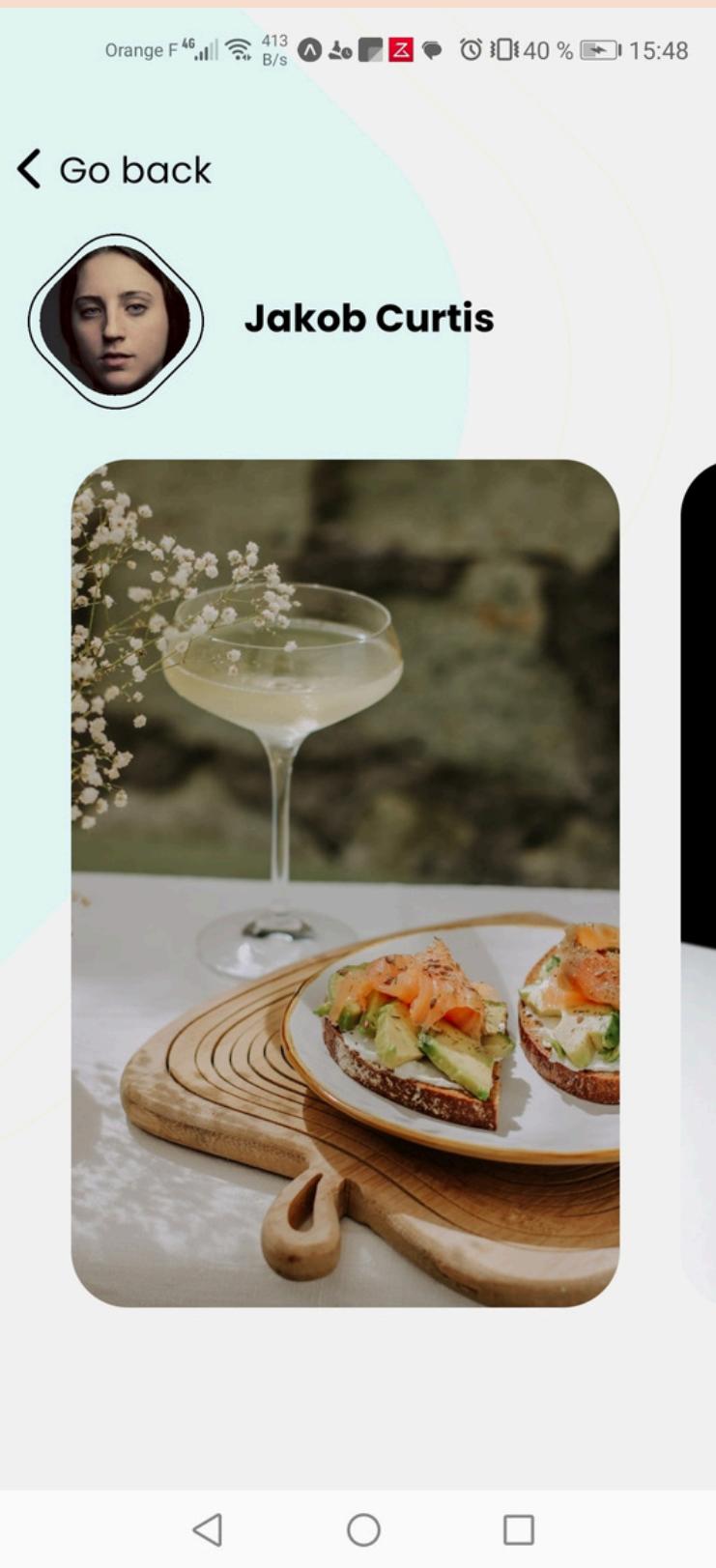
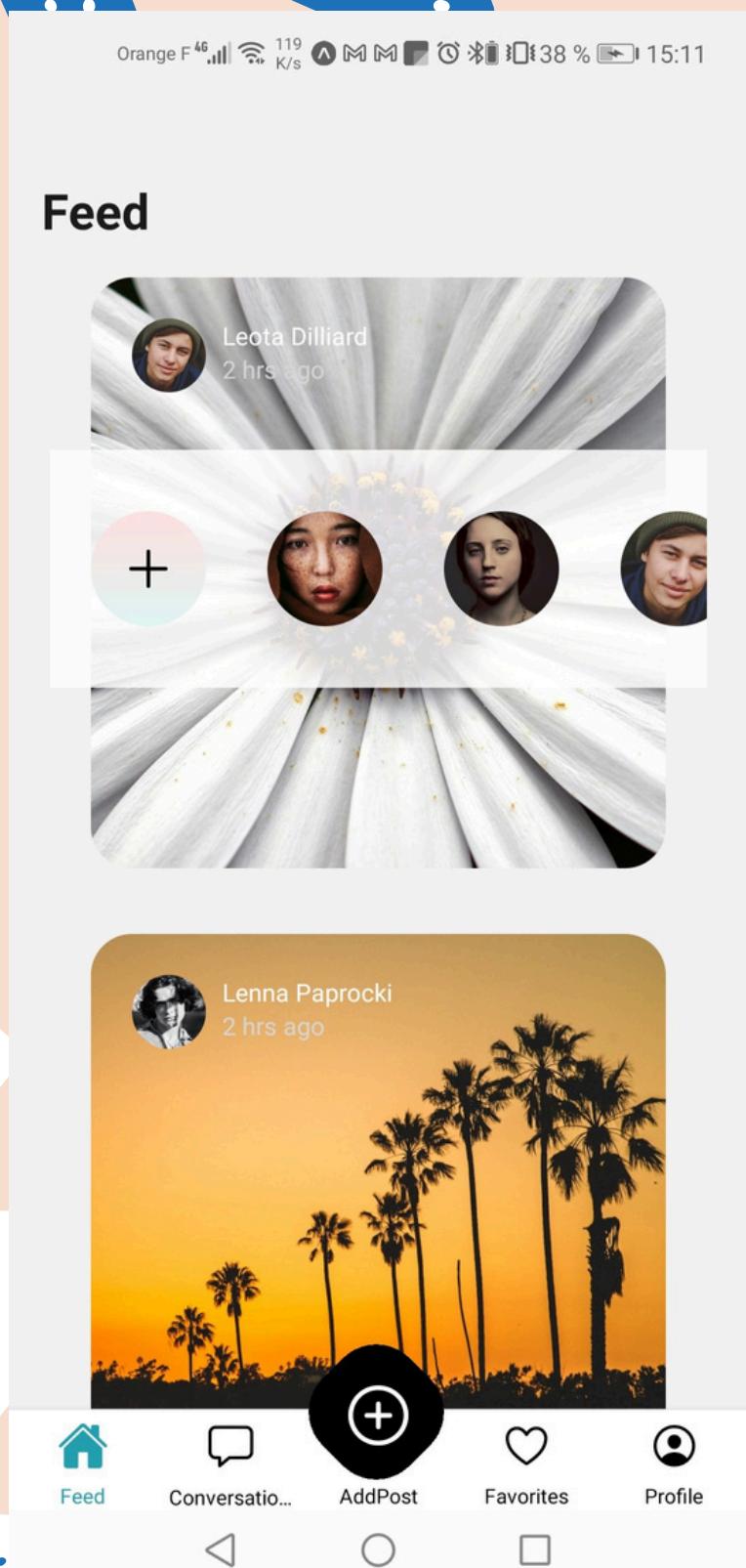


# CAS D'APPLICATION - JEUX

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET EN FIL ROUGE

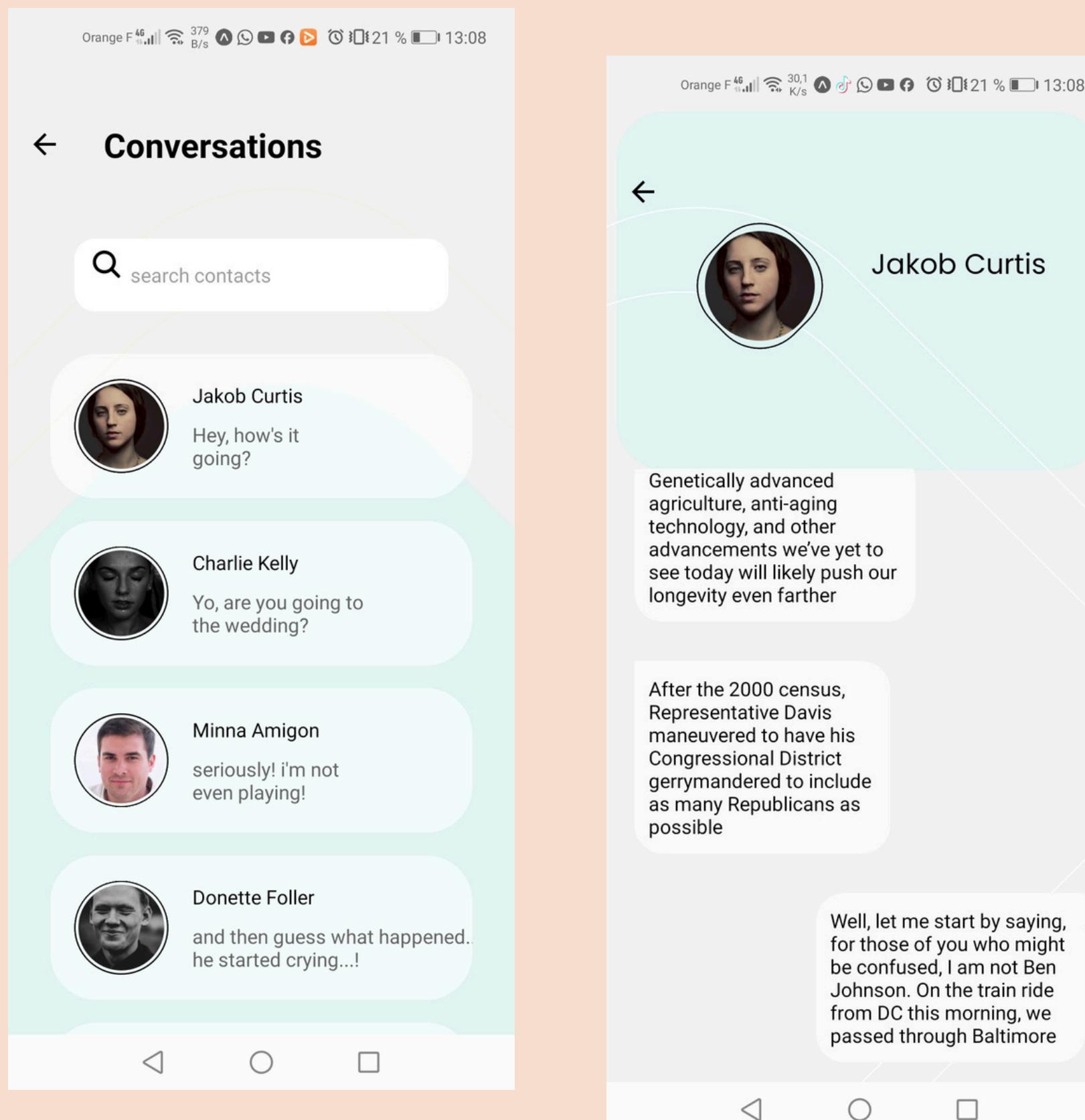
### PRÉSENTATION (FEED) ET CRÉATION



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET EN FIL ROUGE

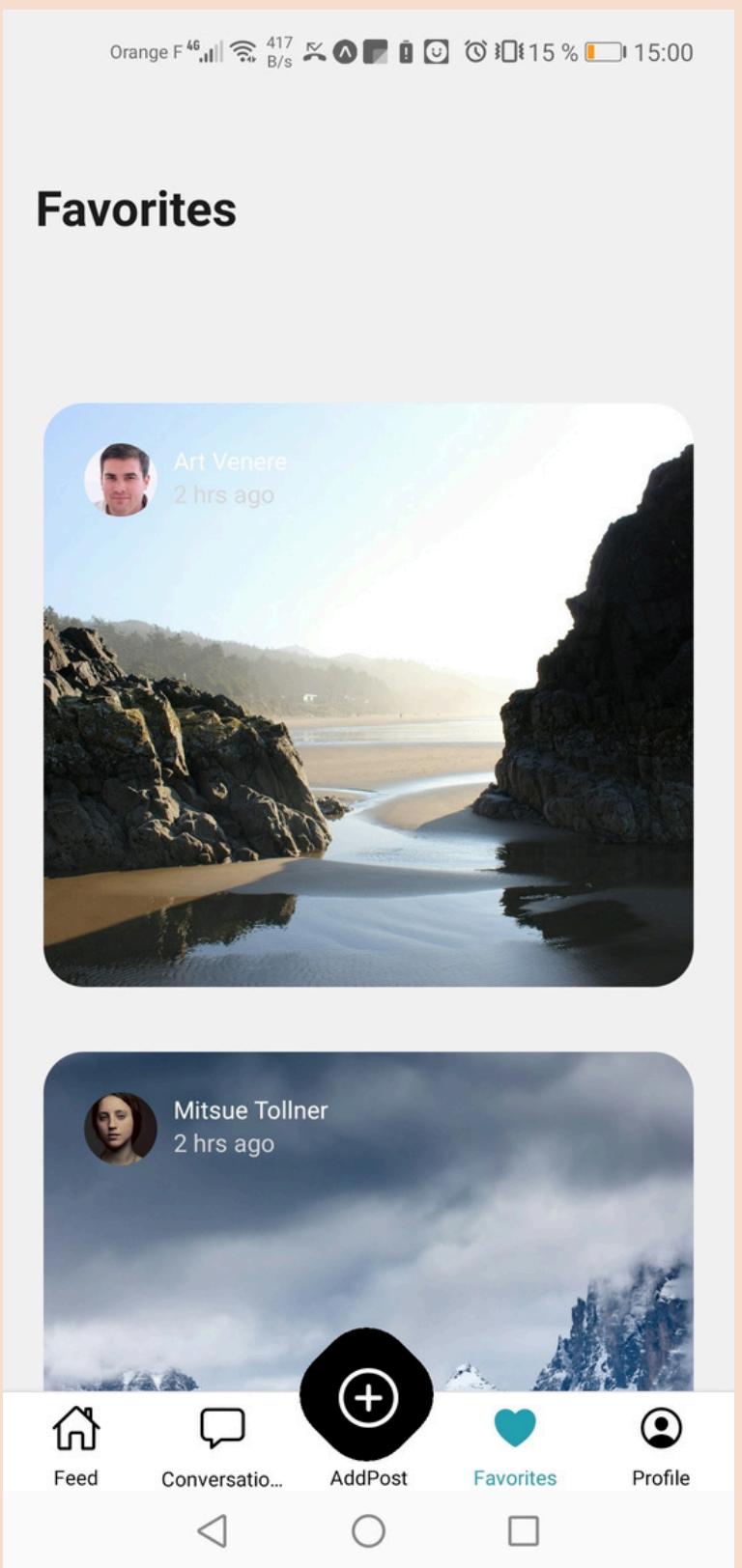
PRESENTATION (CONVERSATION) ET CRÉATION



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET EN FIL ROUGE

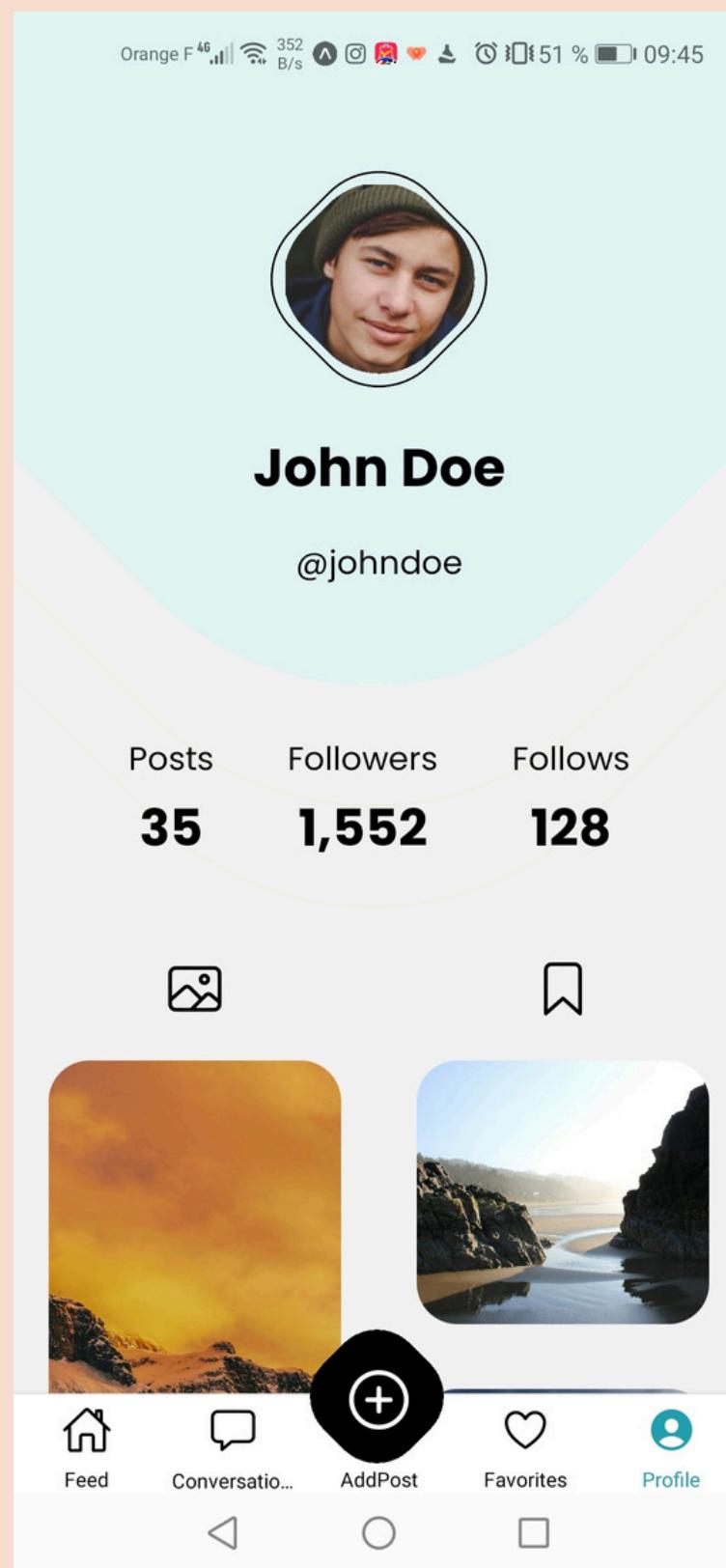
### PRESENTATION (FAVORITE) ET CRÉATION



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET EN FIL ROUGE

### PRESENTATION (PROFILE) ET CRÉATION



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET EN FIL ROUGE

### CRÉATION

```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ npx create-expo-app@latest --template
✓ Choose a template: > Blank
✓ What is your app named? ... book-app
✓ Downloaded and extracted project files.
> npm install

      added 1188 packages, and audited 1189 packages in 31s

        131 packages are looking for funding
          run `npm fund` for details

        found 0 vulnerabilities

    ✓ Your project is ready!

To run your project, navigate to the directory and run one of the following npm commands.

- cd book-app
- npm run android
- npm run ios # you need to use macOS to build the iOS project - use the Expo app if you need to do iOS development without a Mac
- npm run web
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native$ cd book-app/
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ ls
App.js  app.json  assets  babel.config.js  node_modules  package.json  package-lock.json
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### LES MODULES EXTERNES

```
npm i @react-navigation/native
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npm i @react-navigation/native
  added 12 packages, and audited 1201 packages in 5s
  132 packages are looking for funding
    run `npm fund` for details
  found 0 vulnerabilities
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npx expo install react-native-screens react-native-safe-area-context
  > Installing 2 SDK 51.0.0 compatible native modules using npm
  > npm install
  added 4 packages, and audited 1205 packages in 4s
  132 packages are looking for funding
    run `npm fund` for details
  found 0 vulnerabilities
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npm i @react-navigation/stack
  added 12 packages, and audited 1217 packages in 5s
  132 packages are looking for funding
    run `npm fund` for details
  found 0 vulnerabilities
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npx expo install react-native-gesture-handler
  > Installing 1 SDK 51.0.0 compatible native module using npm
  > npm install
  up to date, audited 1217 packages in 4s
  132 packages are looking for funding
    run `npm fund` for details
  found 0 vulnerabilities
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npm i @react-navigation/bottom-tabs
  added 1 package, and audited 1218 packages in 3s
  132 packages are looking for funding
    run `npm fund` for details
  found 0 vulnerabilities
○ balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ █
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### LES ECRANS DE BASE

```
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ mkdir -p src/surfaces
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ touch src/surfaces/Login.js
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ touch src/surfaces/Feed.js
● balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$
```

```
JS Login.js U X JS Feed.js U JS App.js M {} package.json M
book-app > src > surfaces > JS Login.js > [e] Login
1 // ./src/surfaces/Login.js
2 import React from "react";
3 import { View, Text } from "react-native";
4 export const Login = () => {
5   return (
6     <View>
7       |   <Text>this will be the login screen</Text>
8     </View>
9   );
10};
```

```
JS Login.js U JS Feed.js U X JS App.js M {} package.json M
book-app > src > surfaces > JS Feed.js > [e] Feed
1 // ./src/surfaces/Feed.js
2 import React from "react";
3 import { View, Text } from "react-native";
4 export const Feed = () => {
5   return (
6     <View>
7       |   <Text>this will be the feed screen</Text>
8     </View>
9   );
10};
```

```
JS Login.js U JS Feed.js U JS App.js M X {} package.json M
book-app > JS App.js > [e] Home
1 // ./App.js
2 import 'react-native-gesture-handler';
3 import React, { useState } from "react";
4 import { NavigationContainer } from "@react-navigation/native";
5 import { createStackNavigator } from "@react-navigation/stack";
6 import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
7 import { Login } from "./src/surfaces/Login";
8 import { Feed } from "./src/surfaces/Feed";
9 const Stack = createStackNavigator();
10 const Tab = createBottomTabNavigator();
11 function Home() {
12   return (
13     <Tab.Navigator>
14       <Tab.Screen name="Feed" component={Feed} />
15     </Tab.Navigator>
16   );
17 }
18 export default function App() {
19   const [userLoggedIn, setIsUserLoggedIn] = useState(true);
20   return (
21     <NavigationContainer>
22       <Stack.Navigator>
23         {!userLoggedIn ? (
24           <Stack.Screen name="Login" component={Login} />
25         ) : (
26           <Stack.Screen
27             name="Home"
28             component={Home}
29             options={{ headerShown: false }}
30           />
31         )
32       </Stack.Navigator>
33     </NavigationContainer>
34   );
35 }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

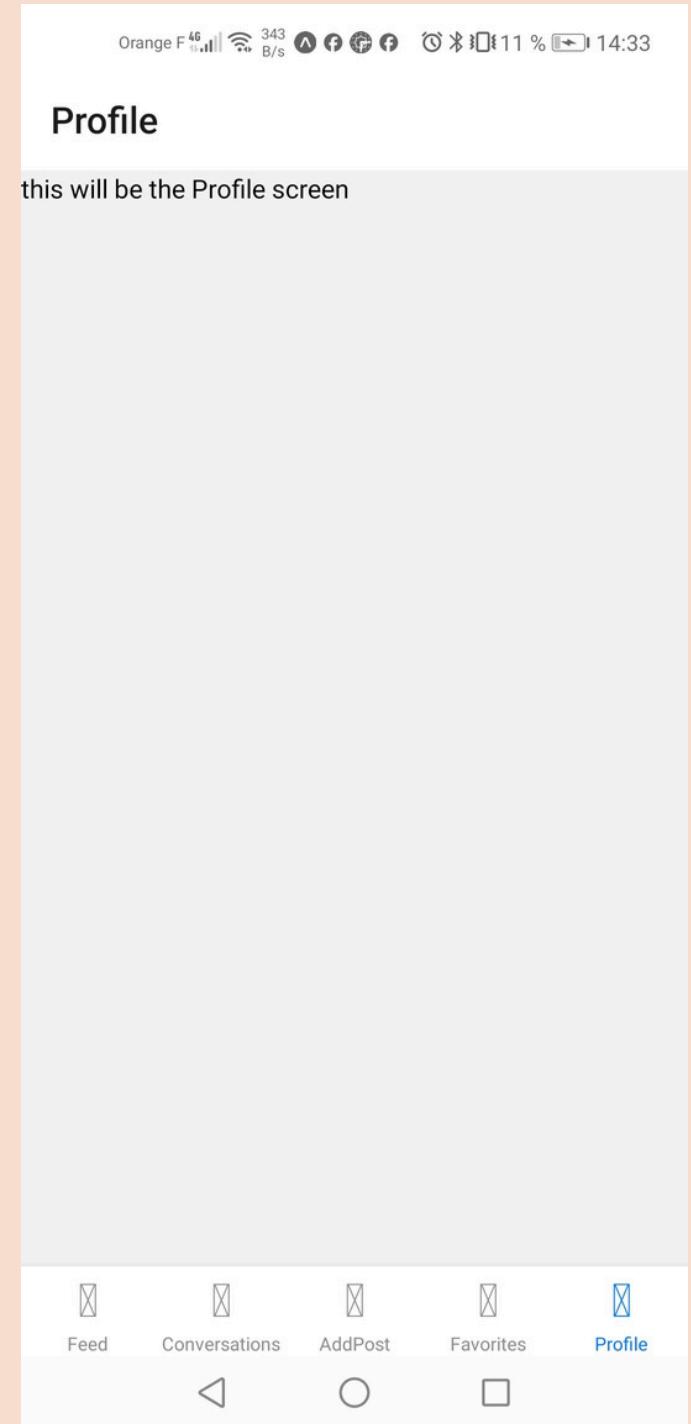
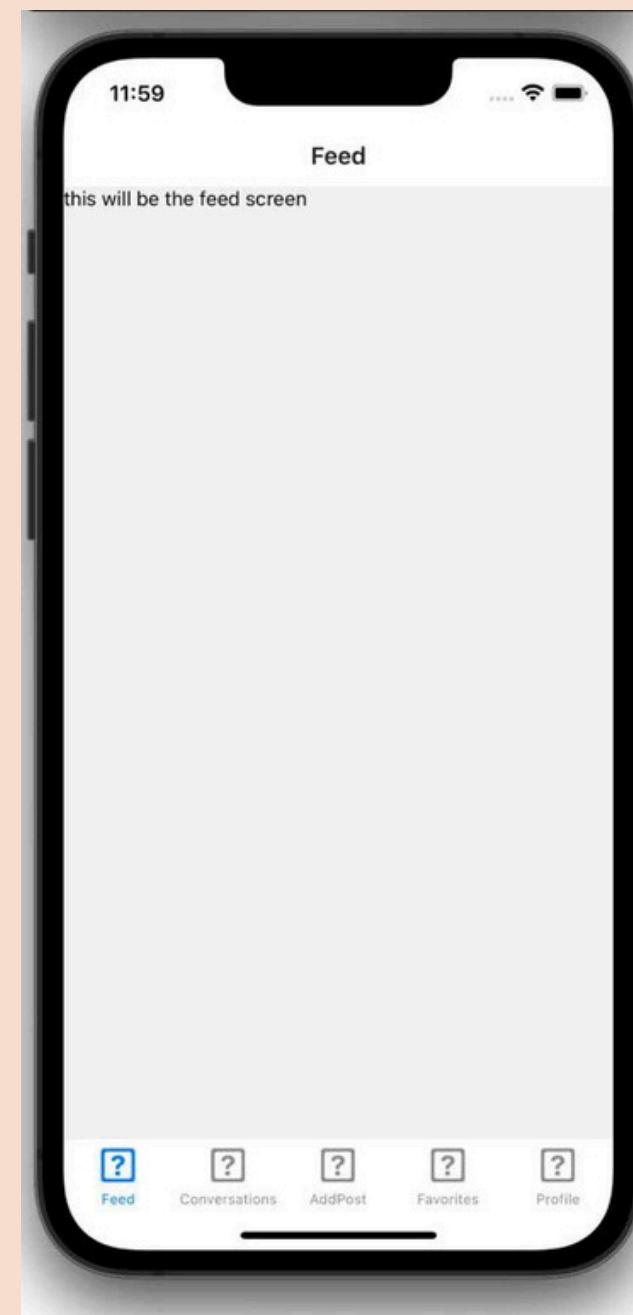
### LES ECRANS DE BASE

```
JS App.js M JS Login.js U JS Feed.js U JS Conversations.js U
book-app > src > surfaces > JS Conversations.js > [o] Conversations
1 import React from "react";
2 import { View, Text } from "react-native";
3 export const Conversations = () => {
4   return (
5     <View>
6       | <Text>this will be the chat screen</Text>
7     </View>
8   );
9 };

JS App.js M JS Login.js U JS Feed.js U JS Conversations.js U
book-app > src > surfaces > JS AddPost.js > ...
1 import React from "react";
2 import { View, Text } from "react-native";
3 export const AddPost = () => {
4   return (
5     <View>
6       | <Text>this will be the AddPost screen</Text>
7     </View>
8   );
9 };

JS App.js M JS Login.js U JS Feed.js U JS Conversations.js U
book-app > src > surfaces > JS Favorites.js > [o] Favorites
1 import React from "react";
2 import { View, Text } from "react-native";
3 export const Favorites = () => {
4   return [
5     <View>
6       | <Text>this will be the Favorites screen</Text>
7     </View>
8   ];
9 };

JS App.js M JS Login.js U JS Feed.js U JS Conversations.js U
book-app > src > surfaces > JS Profile.js > [o] Profile
1 import React from "react";
2 import { View, Text } from "react-native";
3 export const Profile = () => {
4   return [
5     <View>
6       | <Text>this will be the Profile screen</Text>
7     </View>
8   ];
9 };
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### LES ECRANS DE BASE

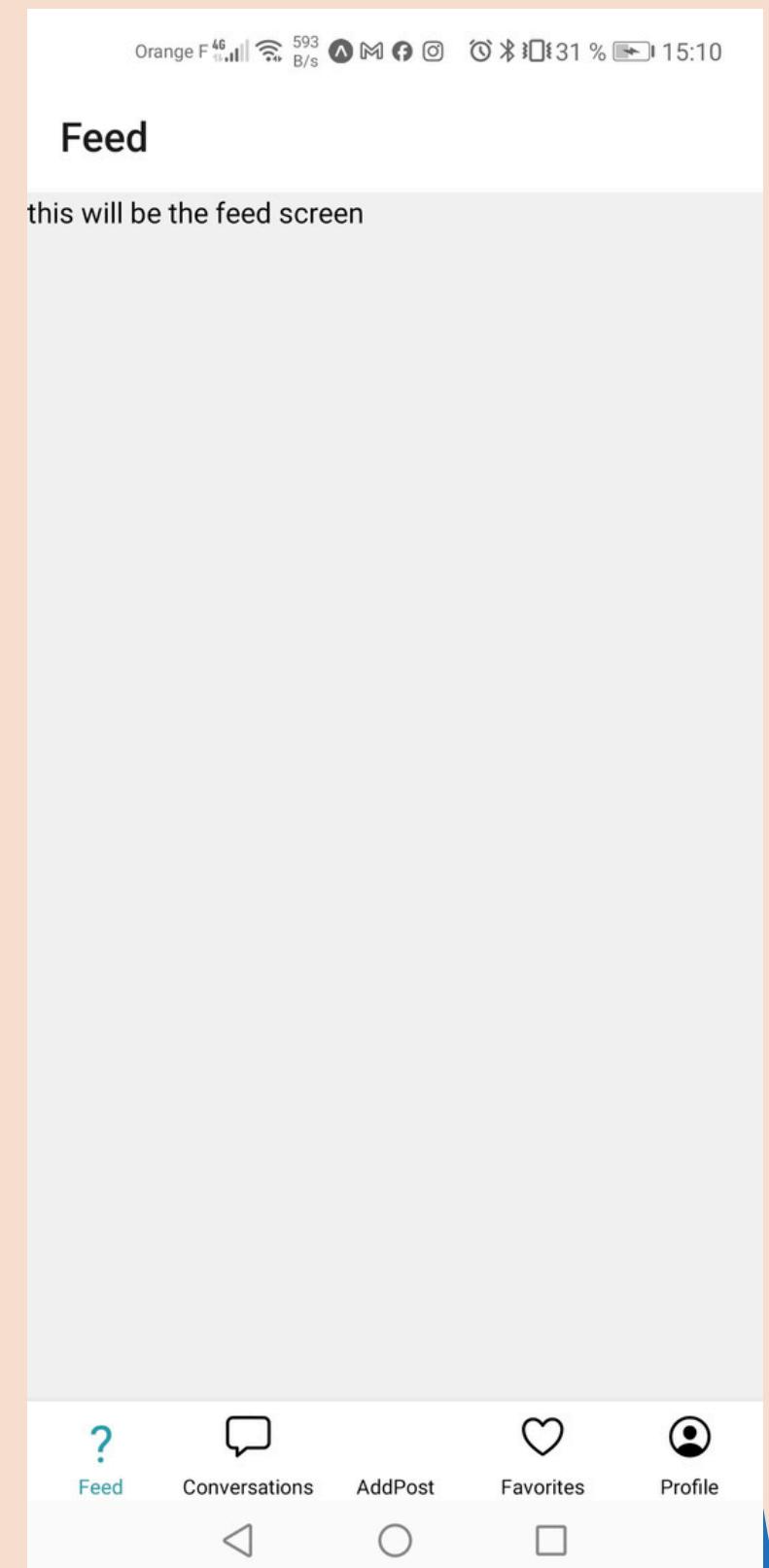
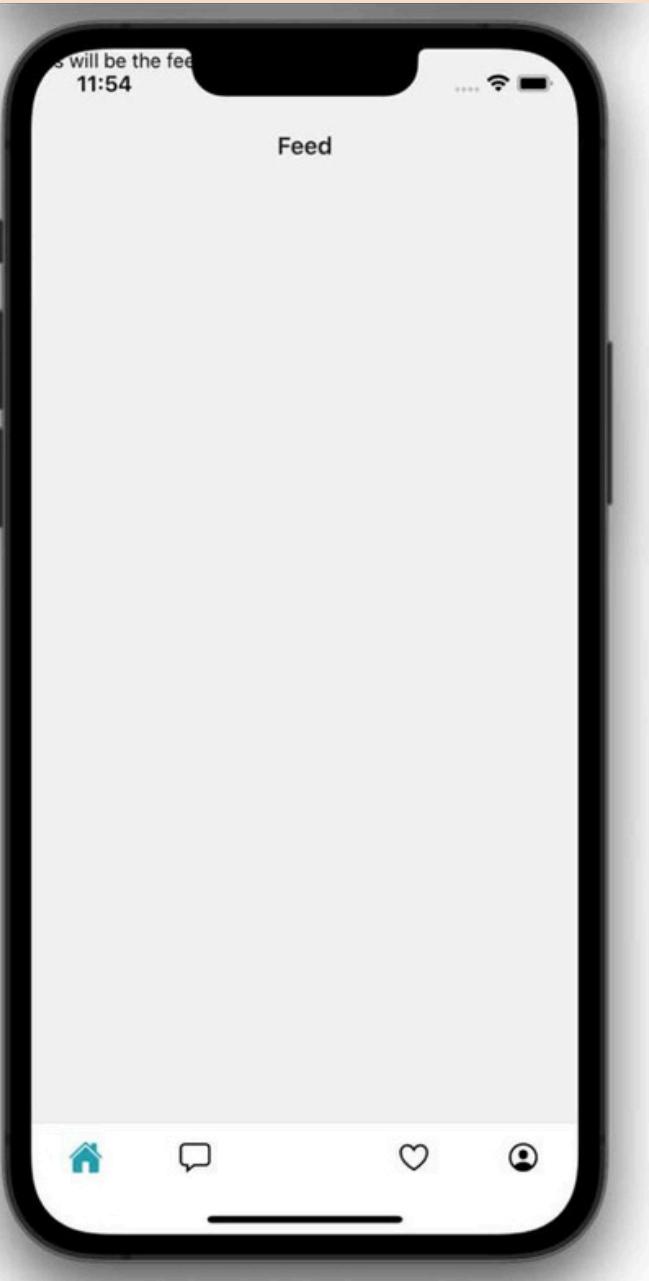
```
balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npx expo install @expo/vector-icons
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

up to date, audited 1218 packages in 5s

132 packages are looking
run `npm fund` for
found 0 vulnerabilities

const Stack = createStackNavigator();
const Tab = createBottomTabNavigator();
function Home() {
  return (
    <Tab.Navigator>
      <Tab.Screen name='Feed' component={Feed} />
      <Tab.Screen name='Conversations' component={Conversations} />
      <Tab.Screen name='AddPost' component={AddPost} />
      <Tab.Screen name='Favorites' component={Favorites} />
      <Tab.Screen name='Profile' component={Profile} />
    </Tab.Navigator>
  );
}

default function App() {
  return (
    <Stack.Navigator screenOptions={{headerShown: false}}>
      <Stack.Screen name='Home' component={Home} />
    </Stack.Navigator>
  );
}
```



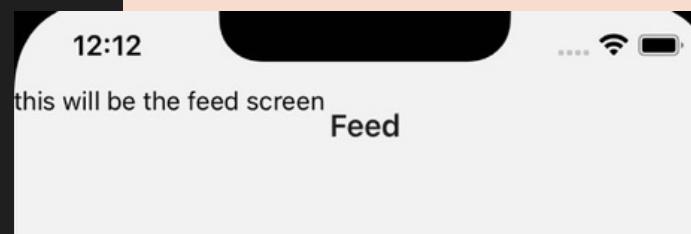
# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### SAFEAREAPROVIDER

```
import { Ionicons } from '@expo/vector-icons';
import { SafeAreaProvider } from 'react-native-safe-area-context';
//import Ionicons from 'react-native-vector-icons/Ionicons';

const Stack = createStackNavigator();
export default function App() {
  const [userLoggedIn, setIsUserLoggedIn] = useState(true);
  return (
    <SafeAreaProvider>
      <NavigationContainer>
        <Stack.Navigator>
          {!userLoggedIn ? (
            <Stack.Screen name="Login" component={Login} />
          ) : (
            <Stack.Screen
              name="Home"
              component={Home}
              options={{ headerShown: false }}
            />
          )}
        </Stack.Navigator>
      </NavigationContainer>
    </SafeAreaProvider>
  );
}
```

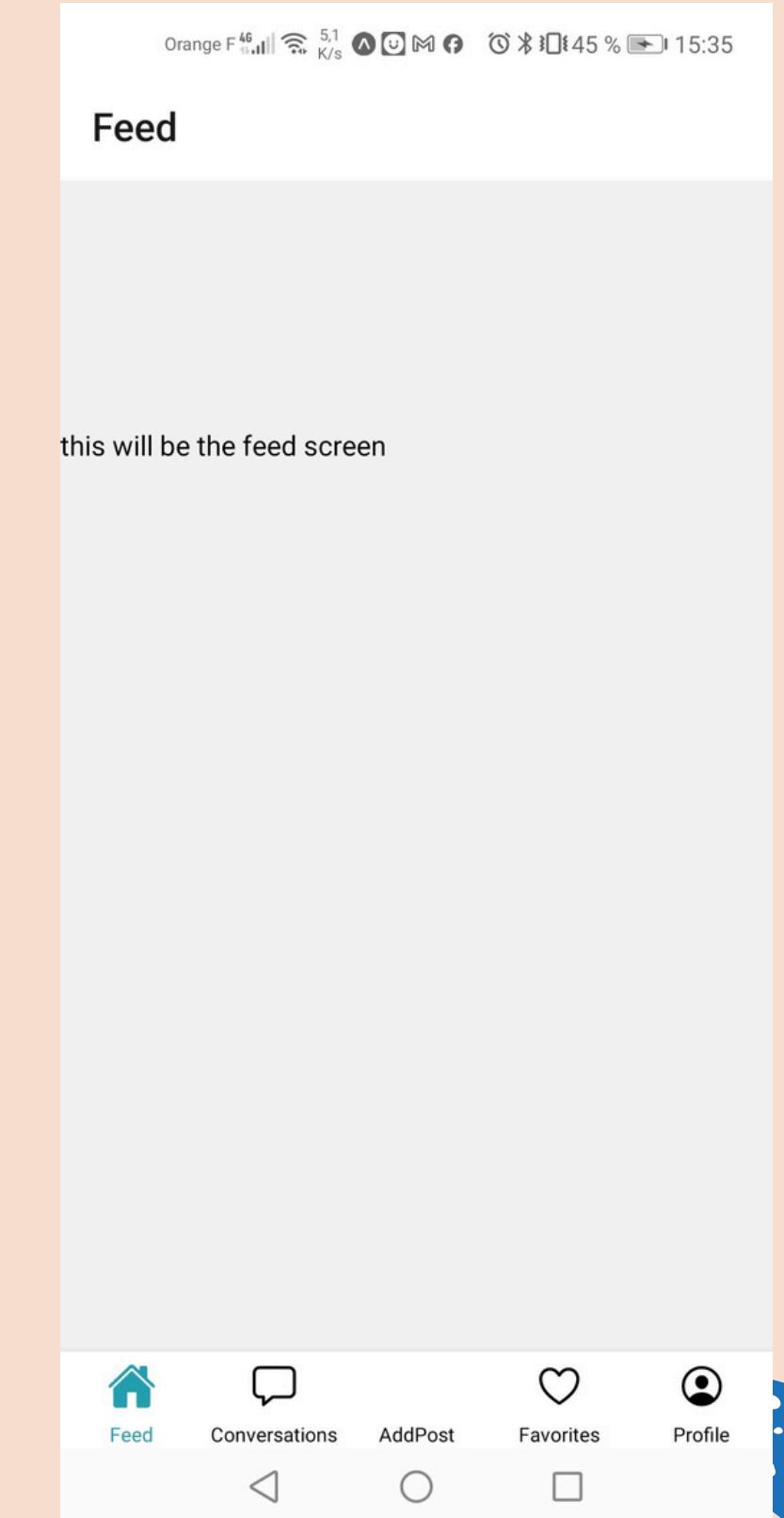


# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### USEHEADERHEIGHT()

```
import React from "react";
import { SafeAreaView } from "react-native-safe-area-context";
import { View, Text } from "react-native";
import { useHeaderHeight } from "@react-navigation/elements";
export const Feed = () => {
  const headerHeight = useHeaderHeight();
  return (
    <SafeAreaView style={{ flex: 1, paddingTop: headerHeight }}>
      <View>
        <Text>this will be the feed screen</Text>
      </View>
    </SafeAreaView>
  );
};
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### GOOGLE FONTS

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npx expo install expo-font
> Installing 1 SDK 51.0.0 compatible native module using npm
> npm install

up to date, audited 1223 packages in 6s

132 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
  > Added config plugin: expo-font
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npx expo install @expo-google-fonts/poppins
> Installing 1 other package using npm
> npm install --save @expo-google-fonts/poppins

added 1 package, and audited 1224 packages in 8s

132 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
  > Added config plugin: expo-font
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npx expo install expo-app-loading
> Installing 1 other package using npm
> npm install --save expo-app-loading
npm WARN deprecated glob@7.1.6: Glob versions prior to v9 are no longer supported

added 47 packages, and audited 1271 packages in 13s

132 packages are looking for funding
  run `npm fund` for details
```

```
import AppLoading from "expo-app-loading";
import [
  useFonts,
  Poppins_400Regular,
  Poppins_700Bold
] from "@expo-google-fonts/poppins";
```

```
export default function App() {
  const [userLoggedIn, setIsUserLoggedIn] = useState(true);
  let [fontsLoaded] = useFonts({
    Poppins_400Regular,
    Poppins_700Bold
  });

  if (!fontsLoaded) {
    return <AppLoading />;
  }

  return (
    <SafeAreaProvider>
```

```
    );
}

export default function App() {
  const [userLoggedIn, setIsUserLoggedIn] = useState(true);
  let [fontsLoaded] = useFonts({
    Poppins_400Regular,
    Poppins_700Bold
  });

  // if (!fontsLoaded) {
  //   return <AppLoading />;
  // }

  if (!fontsLoaded) {
    return (
      <View>
        <Text>Loading fonts...</Text>
      </View>
    )
  }
```

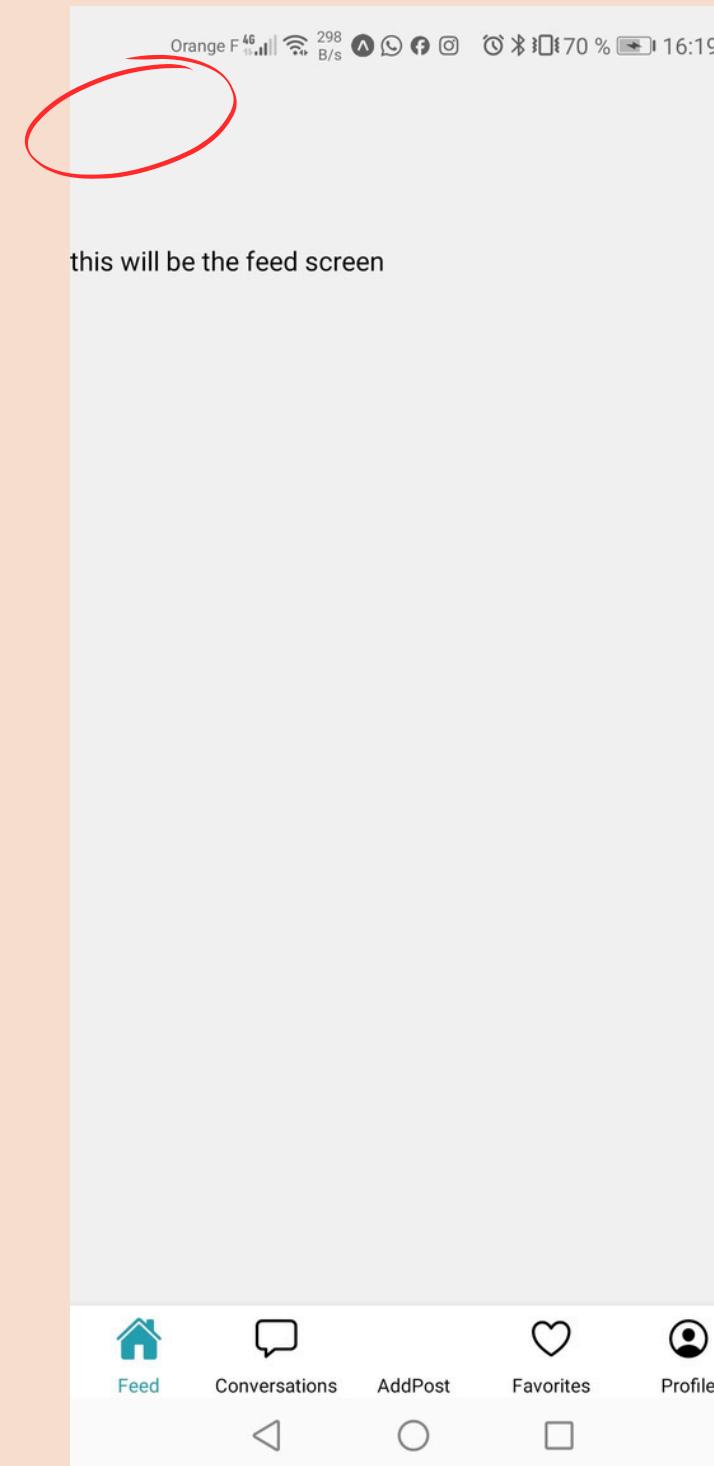
<Text style={{ fontFamily: "Poppins\_400Regular" }}>

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### GOOGLE FONTS

```
const Stack = createStackNavigator();
const Tab = createBottomTabNavigator();
function Home() {
  return (
    <Tab.Navigator screenOptions={({ route }) => ({
      tabBarIcon: ({ focused, color, size }) => {
        let iconName;
        if (route.name === "Feed") {
          iconName = focused ? "home" : "home-outline";
        } else if (route.name === "Conversations") {
          iconName = focused ? "chatbox" : "chatbox-outline";
        } else if (route.name === "Favorites") {
          iconName = focused ? "heart" : "heart-outline";
        } else if (route.name === "Profile") {
          iconName = focused ? "person-circle" : "person-circle-outline";
        }
        return <Ionicons name={iconName} size={size} color={color} />;
      },
      tabBarActiveTintColor: "#25A0B0",
      tabBarInactiveTintColor: "#000000",
      //tabBarShowLabel: false,
      headerTransparent: true,
      headerTitleAlign: "right",
      headerTitleStyle: {
        paddingTop: 140,
        paddingBottom: 40,
        textAlign: "left",
        fontWeight: "bold",
      },
    })}
  >
```



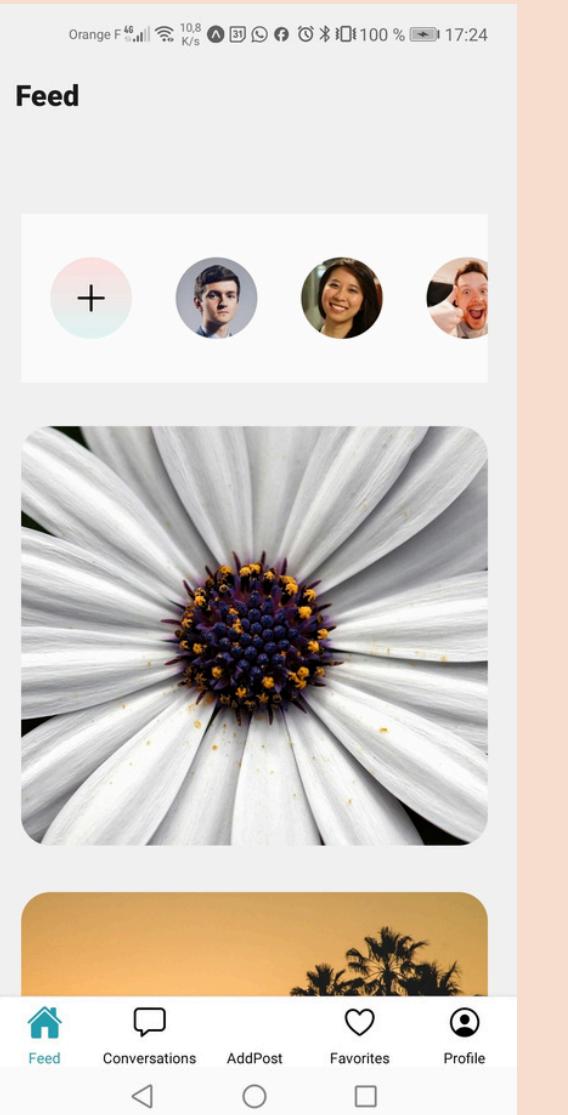
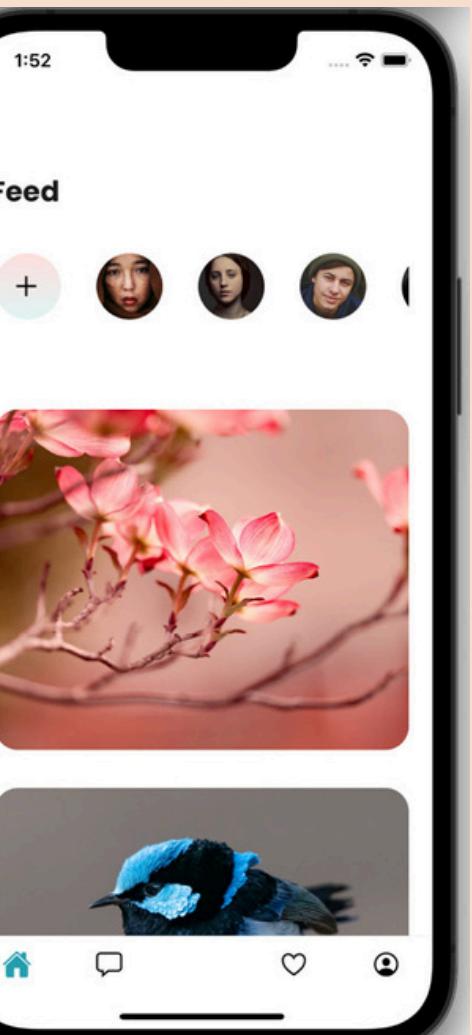
# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

CRÉATION ET STYLE DE COMPOSANTS : FEED - LISTOFAVATARDS - LISTOFCARDS - LISTHEADERCOMPONENTS

book-app > src > surfaces > **JS** Feed.js > Feed

```
1 import React from "react";
2 import { SafeAreaView } from "react-native-safe-area-context";
3 import { View, Text } from "react-native";
4 import { useHeaderHeight } from "@react-navigation/elements";
5 import { ListOfAvatars } from "../components/ListOfAvatars";
6 import { ListOfCards } from "../components/ListOfCards";
7 export const Feed = () => {
8   const headerHeight = useHeaderHeight();
9   return (
10     <SafeAreaView
11       style={{ flex: 1, paddingTop: headerHeight + 20, paddingHorizontal: 20 }}
12     >
13       <View>
14         |   <ListOfAvatars />
15         |   <ListOfCards />
16       </View>
17     </SafeAreaView>
18   );
19 }
```



src > components > **JS** ListOfAvatars.js > ListOfAvatars > renderItem

```
1 import React from "react";
2 import { View, FlatList, Pressable, Image } from "react-native";
3 import { ListHeaderComponent } from "./ListHeaderComponent";
4
5 export const ListOfAvatars = () => {
6   const renderItem = ({ item }) => {
7     return (
8       <Pressable onPress={() => console.log("hello")}>
9         <Image
10           style={{ height: 56, width: 56, borderRadius: 28, marginRight: 30 }}
11           source={{
12             uri: item.url,
13           }}
14         />
15       </Pressable>
16     );
17   };
18
19 const arrayOfAvatars = [
20   { id: 1, url: "https://randomuser.me/api/portraits/men/1.jpg" },
21   { id: 2, url: "https://randomuser.me/api/portraits/women/2.jpg" },
22   { id: 3, url: "https://randomuser.me/api/portraits/men/3.jpg" },
23   { id: 4, url: "https://randomuser.me/api/portraits/women/4.jpg" },
24   { id: 5, url: "https://randomuser.me/api/portraits/men/5.jpg" },
25   { id: 6, url: "https://randomuser.me/api/portraits/women/6.jpg" },
26   { id: 7, url: "https://randomuser.me/api/portraits/men/7.jpg" },
27   { id: 8, url: "https://randomuser.me/api/portraits/women/8.jpg" },
28   { id: 9, url: "https://randomuser.me/api/portraits/men/9.jpg" },
29   { id: 10, url: "https://randomuser.me/api/portraits/women/10.jpg" }
30 ];
31
32 return (
33   <View style={{
34     zIndex: 100,
35     paddingVertical: 30,
36     paddingLeft: 20,
37     backgroundColor: "rgba(255,255,255, 0.85)",
38   }}>
39     <FlatList
40       data={arrayOfAvatars}
41       renderItem={renderItem}
42       keyExtractor={(item) => item.id}
43       horizontal
44       ListHeaderComponent=<ListHeaderComponent />
45       showsHorizontalScrollIndicator={false}
46       snapToInterval={86}
47       decelerationRate='fast'
48     />
49   </View>
50 );
51 }
```

```
20   };
21   const arrayOfImages = [
22     {
23       "itemId": 101,
24       "authorId": 11,
25       "timeStamp": "2 hrs ago",
26       "url": "https://images.unsplash.com/photo-1653546226145-91aa8ce0cc96?ixlib=rb-1.2.1&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1376&q=80",
27     },
28   ],
29 }
```

src > components > **JS** ListOfCards.js > ListOfCards

```
1 import { Image, View } from "react-native";
2 import { FlatList } from "react-native-gesture-handler";
3
4 // src/components/ListOfCards.js
5 export const ListOfCards = () => {
6   const renderItem = ({ item }) => {
7     return (
8       <Image
9         style={{
10           width: "100%",
11           height: 288,
12           borderRadius: 20,
13           marginBottom: 32,
14         }}
15         source={{
16           uri: item.url,
17         }}
18       />
19     );
20   };
21   const arrayOfImages = [
22 ];
23
24 return (
25   <View style={{ paddingVertical: 30 }}>
26     <FlatList
27       data={arrayOfImages}
28       renderItem={renderItem}
29       keyExtractor={(item) => item.id}
30       showsVerticalScrollIndicator={false}
31     />
32   </View>
33 );
34 }
```

```
1 import React from "react";
2 import { View, Pressable } from "react-native";
3 import Ionicons from "@expo/vector-icons/Ionicons";
4 import { LinearGradient } from "expo-linear-gradient";
5
6 export const ListHeaderComponent = () => {
7   return (
8     <Pressable onPress={() => console.log("pressed the add button")}>
9       <View
10         style={{
11           width: 56,
12           height: 56,
13           marginRight: 30,
14         }}>
15         <LinearGradient
16           colors={[ "#FFE1E0", "#E1F6F4" ]}>
17           <View style={{
18             width: "100%",
19             height: "100%",
20             justifyContent: "center",
21             alignItems: "center",
22           }}>
23             <Ionicons name="add" size={30} color="#000000" />
24           </View>
25         </LinearGradient>
26       </View>
27     </Pressable>
28   );
29 };
30 }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (APP)

```
JS App.js > ⚡ App
1 import 'react-native-gesture-handler';
2 import React, { useState } from "react";
3 import { NavigationContainer } from "@react-navigation/native";
4 import { createStackNavigator } from "@react-navigation/stack";
5 import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
6 import { Login } from "./src/surfaces/Login";
7 import { SafeAreaProvider } from 'react-native-safe-area-context';
8 import {
9   useFonts,
10  Poppins_400Regular,
11  Poppins_700Bold
12 } from "@expo-google-fonts/poppins";
13 import { View, Text } from 'react-native';
14 import { ConversationsNavigation } from './src/surfaces/ConversationsNavigation';
15 import { Home } from './src/surfaces/Home';
16
17 const Stack = createStackNavigator();
18 const Tab = createBottomTabNavigator();
19
20 export default function App() {
21   const [userLoggedIn, setIsUserLoggedIn] = useState(true);
22   let [fontsLoaded] = useFonts({
23     Poppins_400Regular,
24     Poppins_700Bold
25   });
26
27   if (!fontsLoaded) {
28     return (
29       <View>
30         | <Text>Loading fonts...</Text>
31       </View>
32     )
33   }
34
35   return (
36     <SafeAreaProvider>
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4:~/www/react_native/book-app$ npm i react-native-gesture-handler
up to date, audited 1272 packages in 11s
132 packages are looking for funding
  run `npm fund` for details
```

```
JS App.js > ⚡ App
21  export default function App() {
22    return (
23      <View>
24        | <Text>Loading fonts...</Text>
25      </View>
26    )
27
28    return (
29      <SafeAreaProvider>
30        <NavigationContainer>
31          <Stack.Navigator>
32            {!userLoggedIn ? (
33              <Stack.Screen name="Login" component={Login} />
34            ) : (
35              <Stack.Screen
36                name="Home"
37                component={Home}
38                options={{ headerShown: false }}
39              />
40              <Stack.Screen
41                name='ConversationsNav'
42                component={ConversationsNavigation}
43                options={{ headerShown: false }}
44              />
45            )
46          </Stack.Navigator>
47        </NavigationContainer>
48      </SafeAreaProvider>
49    );
50  }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (HOME)

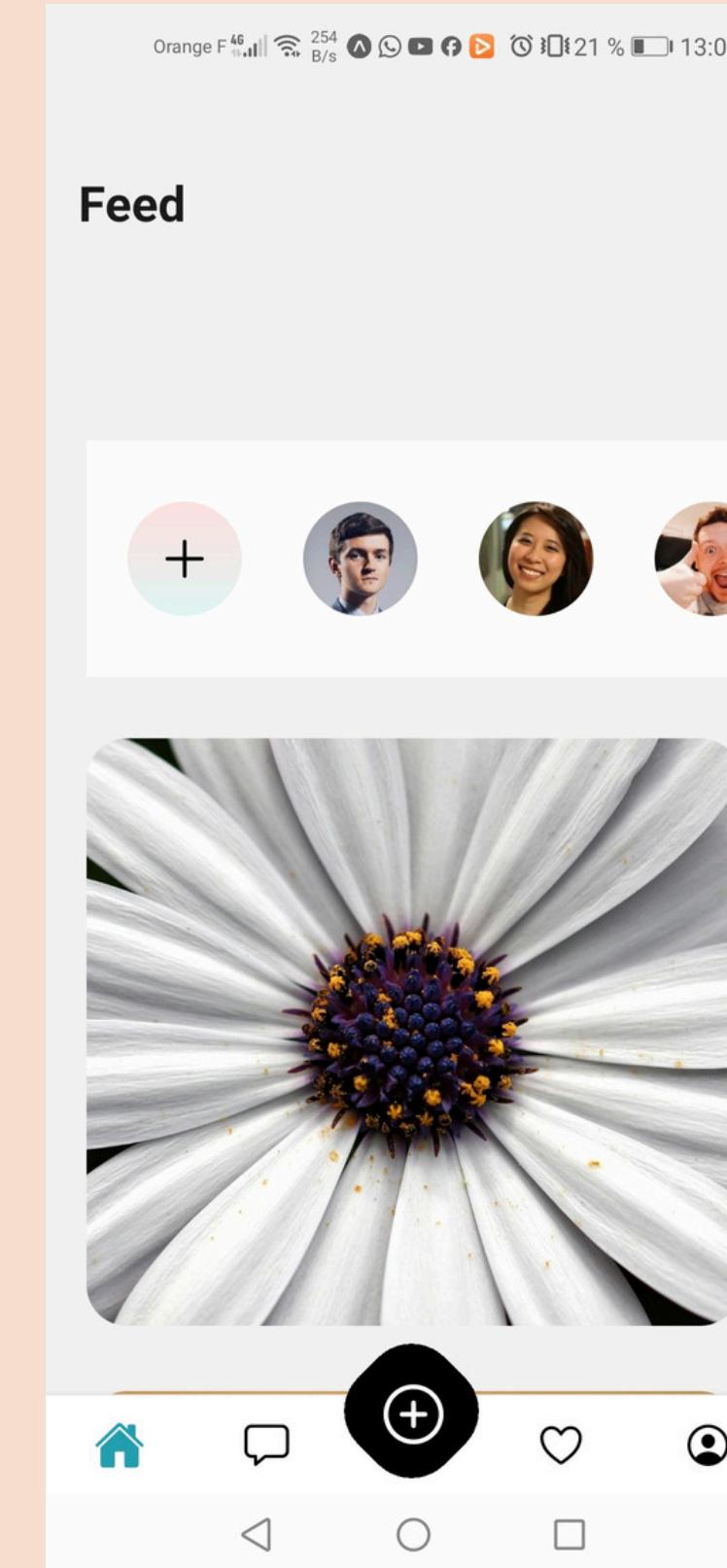
```
src > surfaces > JS Home.js > ...
1 import "react-native-gesture-handler";
2 import React from "react";
3 import { createBottomTabNavigator } from "@react-navigation/bottom-tabs";
4 import { Feed } from "./Feed";
5 import { Profile } from "./Profile";
6 import { Favorites } from "./Favorites";
7 import { AddPost } from "./AddPost";
8 import Ionicons from "@expo/vector-icons/Ionicons";
9 import { colors } from "../styles/colors";
10 import { View } from "react-native";
11
12 const Tab = createBottomTabNavigator();
13
14 const ConversationsBase = () => <View style={{ flex: 1 }} />;
15
16 export const Home = () => {
17   return (
18     <Tab.Navigator
19       screenOptions={({ route }) => ({
20         tabBarIcon: ({ focused, color, size }) => {
21           let iconName;
22
23           if (route.name === "Feed") {
24             iconName = focused ? "home" : "home-outline";
25           } else if (route.name === "Favorites") {
26             iconName = focused ? "heart" : "heart-outline";
27           } else if (route.name === "Profile") {
28             iconName = focused ? "person-circle" : "person-circle-outline";
29           }
30
31           return <Ionicons name={iconName} size={size} color={color} />;
32         },
33         tabBarActiveTintColor: colors.accentStroke,
34         tabBarInactiveTintColor: colors.black,
35       })
36     >
37       <Tab.Screen name='Feed' component={Feed} />
38       <Tab.Screen
39         name='ConversationsMain'
40         component={ConversationsBase}
41         options={{
42           tabBarIcon: ({ size }) => (
43             <Ionicons name='chatbox-outline' color="#000000" size={size} />
44           ),
45         }}
46       >
47         <Tab.Screen name='AddPost' component={AddPost} />
48         <Tab.Screen
49           name='Favorites'
50           component={Favorites}
51           options={{
52             tabBarIcon: ({ size }) => (
53               <Ionicons name='add-circle-outline' color="#ffffff" size={size} />
54             ),
55           }}
56         >
57           <Tab.Screen name='Profile' component={Profile} />
58           <Tab.Screen
59             name='FavoritesNav'
60             component={Favorites}
61             options={{
62               headerShown: false
63             }}
64           >
65             <View
66               style={{
67                 height: 160,
68                 width: 160,
69                 position: "absolute",
70                 left: -80,
71                 top: -80,
72                 border: "2px solid #000000",
73                 borderRadius: 50,
74                 overflow: "hidden"
75               }}
76             >
77               <Image
78                 source={require("../assets/images/icon-profile.png")}
79                 style={{ width: "100%", height: "100%" }}
80               >
81             </View>
82           >
83         </Tab.Screen>
84       </Tab.Screen>
85     </Tab.Navigator>
86   );
87 }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (CONVERSATIONNAVIGATION)

```
src > surfaces > JS ConversationsNavigation.js > headerTitleStyle
1 import { Conversations } from "./Conversations";
2 import { Messages } from "./Messages";
3 import { createStackNavigator } from "@react-navigation/stack";
4
5 const Stack = createStackNavigator();
6
7 export const ConversationsNavigation = () => {
8   //const [conversationId, setConversationId] = useState(null);
9   return (
10
11   <Stack.Navigator
12     screenOptions={{
13       headerBackTitleVisible: false,
14       headerTintColor: "#000000",
15       headerTransparent: true,
16       headerTitleAlign: "left",
17       headerStyle: {
18         height: 160,
19       },
20       headerTitleStyle: [
21         {
22           textAlign: "left",
23           fontWeight: "bold",
24           fontFamily: "Poppins_700Bold",
25           fontSize: 24,
26         },
27     }
28   >
29     <Stack.Screen name='Conversations' component={Conversations} />
30     <Stack.Screen
31       name='Messages'
32       component={Messages}
33       options={({ route }) => ({
34         title: route.params.name,
35         headerTitleStyle: {
36           textAlign: "center",
37           fontFamily: "Poppins_400Regular",
38           fontSize: 20,
39           position: "absolute",
40           top: 100,
41           left: 120,
42         },
43       })}
44   </Stack.Navigator>
45
46 );
47 }
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (CONVERSATIONS)

```
src > surfaces > JS Conversations.js > Conversations
1 import React from "react";
2 import { View, TextInput, Pressable } from "react-native";
3 import { useHeaderHeight } from "@react-navigation/elements";
4 import { SafeAreaView } from "react-native-safe-area-context";
5 import Ionicons from "@expo/vector-icons/Ionicons";
6 import { ListOfConvos } from "../components/ListOfConvos";
7
8 export const Conversations = ({ navigation }) => {
9   const headerHeight = useHeaderHeight();
10  const [text, onChangeText] = React.useState("");
11
12  return (
13    <SafeAreaView style={{ flex: 1, paddingTop: headerHeight - 30 }}>
14      <View
15        style={{
16          width: 650,
17          height: 570,
18          borderRadius: 155,
19          borderWidth: 1,
20          borderColor: "#EEF2E2",
21          position: "absolute",
22          top: 210,
23          left: -160,
24          transform: [{ rotate: "-45deg" }],
25        }}
26      />
27      <View
28        style={{
29          width: 650,
30          height: 570,
31          borderRadius: 155,
32          borderWidth: 1,
33          borderColor: "#EEF2E2",
34          position: "absolute",
35          top: 280,
36          left: -160,
37          transform: [{ rotate: "-45deg" }],
38        }}
39      />
40      <View
41        style={{
42          width: 650,
43          height: 570,
44          borderRadius: 155,
45          position: "absolute",
46        }}
47    </SafeAreaView>
```

```
src > surfaces > JS Conversations.js > Conversations
8 export const Conversations = ({ navigation }) => {
44   borderRadius: 155,
45   position: "absolute",
46   top: 350,
47   left: -160,
48   backgroundColor: "#E1F6F4",
49   transform: [{ rotate: "-45deg" }],
50 }
51 </>
52 <View style={{ marginHorizontal: 30, position: "relative" }}>
53   <View>
54     <TextInput
55       style={{
56         fontSize: 14,
57         paddingVertical: 12,
58         paddingLeft: 40,
59         marginHorizontal: 17,
60         borderRadius: 15,
61         backgroundColor: "#ffffff",
62         shadowColor: "#000000",
63         shadowOffset: { width: 0, height: 4 },
64         shadowOpacity: 0.1,
65         shadowRadius: 9,
66       }}
67       onChangeText={onChangeText}
68       value={text}
69       placeholder='search contacts'
70     />
71     <Ionicons
72       name='search'
73       size={24}
74       color='#000000'
75       style={{ position: "absolute", left: 28, top: 6 }}
76     />
77   </View>
78   <ListOfConvos navigation={navigation} />
79 </View>
80 </SafeAreaView>
81 };
82 }
```

```
src > styles > JS colors.js > ...
1 export const colors = {
2   black: "#000000",
3   grey: "#4E4E4E",
4   lightGrey: "#C4C4C4",
5   white: "#FFFFFF",
6   accentStroke: "#25A0B0",
7   mainTheme: "#E1F6F4",
8   pink: "#F5C8C6",
9   accentLines: "#EEF2E2",
10 };
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

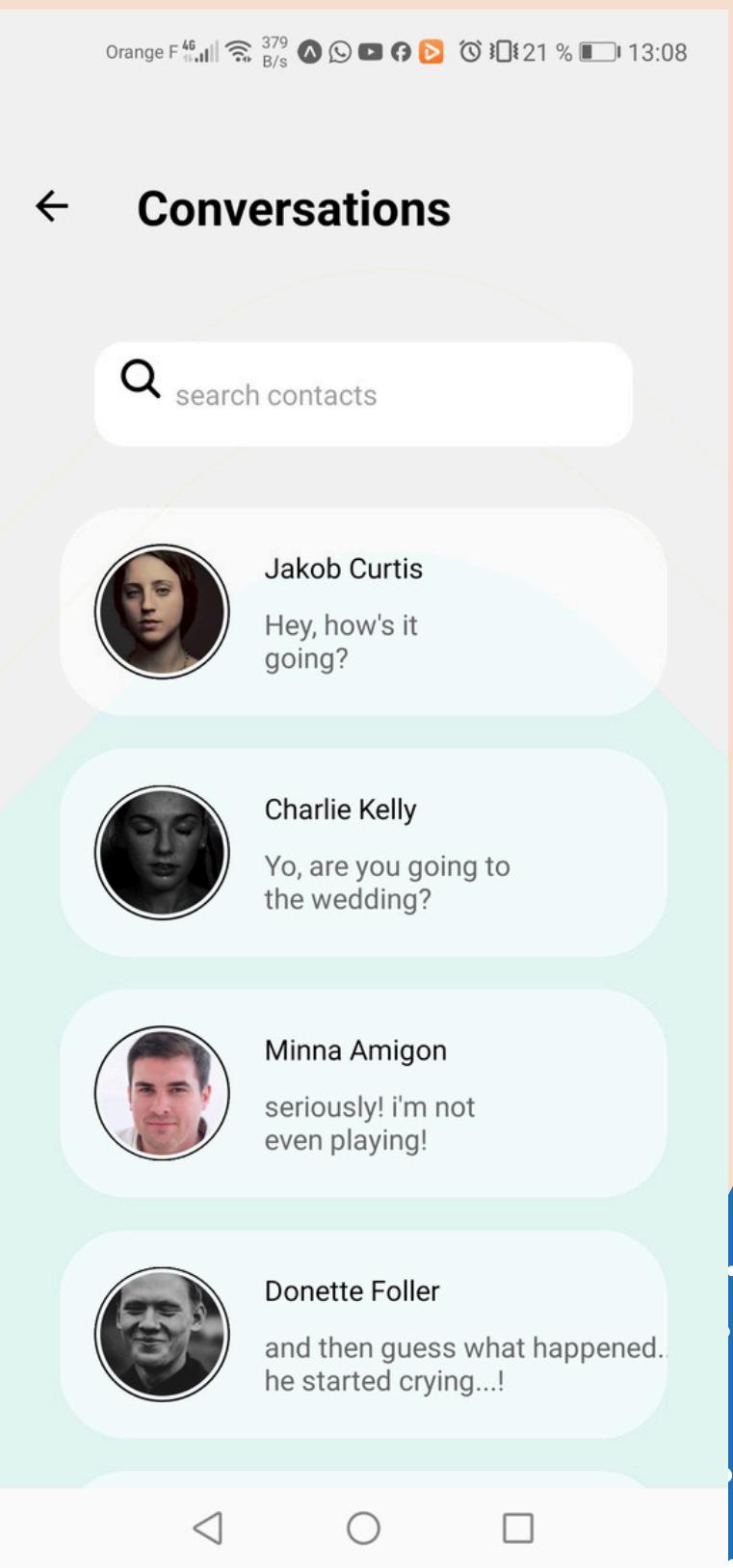
### CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (CONVERSATIONS)

```
src > components > JS ListOfConvos.js > ...
1 import React, { useState, useEffect } from "react";
2 import { View, FlatList, Pressable, Image, Text } from "react-native";
3
4 export const ListOfConvos = ({ navigation }) => {
5
6   const userList = [...]
7
8   const conversationsList = [...]
9
10  const renderItem = ({ item }) => {
11
12    const currentUser = userList.filter(
13      (user) => user.id === item.userId
14    );
15
16    return (
17      <Pressable
18        onPress={() => navigation.navigate("Messages", { name:currentUser[0].name, avatar:currentUser[0].url })}
19        style={{
20          height: 103,
21          backgroundColor: "rgba(255,255,255,0.6)",
22          borderRadius: 33,
23          marginBottom: 16,
24          justifyContent: "flex-start",
25          alignItems: "center",
26          flexDirection: "row",
27        }}
28      >
29        <View>
30          <Image
31            style={{
32              width: 67,
33              height: 67,
34              borderRadius: 35,
35              borderColor: "#000000",
36              borderWidth: 1,
37              marginHorizontal: 17,
38            }}
39          >
40            <Image
41              style={{
42                width: 61,
43                height: 61,
44                borderRadius: 35,
45                marginTop: 2,
46                marginLeft: 2,
47              }}
48              source={{
49                uri: currentUser[0].url,
50              }}
51            />
52          </Image>
53        </View>
54      </View>
55    );
56  };
57
58  return (
59    <FlatList
60      data={conversationsList}
61      renderItem={renderItem}
62      keyExtractor={(item) => item.id}
63      showsVerticalScrollIndicator={false}
64      snapToInterval={119}
65      decelerationRate='fast'
66      ListHeaderComponent=<View style={{ height: 30 }} />
67    );
68  );
69}
```

```
6 const userList = [
7   {
8     "id": 1,
9     "name": "Malena Tudi",
10    "url": "https://raw.githubusercontent.com/PacktPublishing/Simplifying-State-Management-in-React-Native/cd04c474053275d4e22a8173695a2b972d012567/chapter-4/assets/avatars/1.png"
11  },
12]
```

```
src > components > JS ListOfConvos.js > JS ListOfConvos > JS renderItem
4 export const ListOfConvos = ({ navigation }) => {
5
6   const renderItem = ({ item }) => {
7
8     <Image
9       style={{
10         width: 61,
11         height: 61,
12         borderRadius: 35,
13         marginTop: 2,
14         marginLeft: 2,
15       }}
16       source={{
17         uri: currentUser[0].url,
18       }}
19     />
20   </Image>
21   <View>
22     <Text style={{ fontSize: 14, paddingBottom: 9 }}>
23       {currentUser[0].name}
24     </Text>
25     <Text style={{ color: "#656565", width: "65%" }}>
26       {item.text}
27     </Text>
28   </View>
29   </Pressable>
30 };
31
32 return (
33   <View
34     style={{
35       paddingTop: 30,
36       marginTop: -30,
37       marginBottom: 50,
38     }}
39   >
40     <FlatList
41       data={conversationsList}
42       renderItem={renderItem}
43       keyExtractor={(item) => item.id}
44       showsVerticalScrollIndicator={false}
45       snapToInterval={119}
46       decelerationRate='fast'
47       ListHeaderComponent=<View style={{ height: 30 }} />
48     </FlatList>
49   </View>
50 );
51
```

```
109 const conversationsList = [
110   {
111     "id": 1,
112     "userId": 2,
113     "text": "Hey, how's it going?"
114   },
115   {
116     "id": 2,
```



# LA GESTION DES ETATS - REACT NATIVE

## CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (MESSAGES)

```
src > surfaces > JS Messages.js > [x] Messages
1 import React from "react";
2 import { View, Text, Image } from "react-native";
3 import { useHeaderHeight } from "@react-navigation/elements";
4 import { SafeAreaView } from "react-native-safe-area-context";
5 import { ListOfMessages } from "../components/ListOfMessages";
6
7
8 export const Messages = ({ route }) => {
9   const headerHeight = useHeaderHeight();
10  console.log("log route in message surface ", JSON.stringify(route))
11
12  return (
13    
14      <>
15        <View
16          style={{
17            position: "absolute",
18            top: 40,
19            left: "2%",
20            backgroundColor: "#E1F6F4",
21            width: "96%",
22            height: 255,
23            borderRadius: 34,
24          }}
25        />
26        <View
27          style={{
28            width: 650,
29            height: 570,
30            borderRadius: 155,
31            borderWidth: 1,
32            borderColor: "#ffffff",
33            position: "absolute",
34            top: 250,
35            left: -300,
36            transform: [{ rotate: "-45deg" }],
37          }}
38        />
39        <View
40          style={{
41            width: 650
42          }}
43      </>
44    
45  )
46}
47
48
```

```
surfaces > JS Messages.js > [愍] Messages
export const Messages = ({ route }) => {
  <SafeAreaView>
    <View style={{ width: 650, height: 570, borderStyle: "solid", border: "1px solid black", borderRadius: 15, position: "absolute", top: 190, left: -290, transform: [{ rotate: "-45deg" }], }}>
      <View style={{ position: "absolute", top: 125, left: 70, }}>
        <View style={{ width: 84, height: 84, borderStyle: "solid", border: "1px solid black", borderRadius: 35, border: "1px solid black", position: "absolute", top: -3, left: -3, }}>
          <Image style={{ height: 78, width: 78, borderRadius: 40 }} source={{ uri: route.params.avatar, }} />
        </View>
      </View>
      <ListOfMessages conversationId="2" />
    </View>
  </SafeAreaView>
};

```

```
components > JS ListOfMessages.js > [x] styles > ↗ text

import { View, FlatList, Text, StyleSheet } from "react-native";

export const ListOfMessages = () => {
  const messages = [...];

  const renderItem = ({ item }) => {
    return (
      <View
        style={[
          styles.text,
          item.type === "from" ? styles.textTo : styles.textFrom,
        ]}
      >
        <Text style={{}}>{item.text}</Text>
      </View>
    );
  };

  return (
    <View
      style={{
        paddingHorizontal: 20,
      }}
    >
      <FlatList
        data={messages.messages}
        renderItem={renderItem}
        keyExtractor={(item) => item.id}
        showsVerticalScrollIndicator={false}
        inverted
      />
    </View>
  );
};

const styles = StyleSheet.create({
  text: [
    backgroundColor: "#FAFAFA",
    borderBottomRightRadius: 20,
    borderBottomLeftRadius: 20,
    fontSize: 14,
    padding: 10,
    maxWidth: "65%",
    marginVertical: 14,
  ],
  textFrom: {
    borderTopLeftRadius: 20,
    alignSelf: "flex-end",
  },
  textTo: {
    borderTopRightRadius: 20,
    alignSelf: "flex-start",
  },
});
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (MESSAGES)

```
src > components > JS ListOfMessages.js > messages > "messages"

1 import { View, FlatList, Text, StyleSheet } from "react-native";
2
3 export const ListOfMessages = () => {
4   const messages = [
5     {
6       "id": 2,
7       "messages": [
8         {
9           "id": 1,
10          "type": "to",
11          "text": "Well, let me start by saying, for those of you who might be confused, I am not Ben Johnson. On the train ride from DC this morning, we passed through Baltimore"
12        },
13        {
14          "id": 2,
15          "type": "from",
16          "text": "After the 2000 census, Representative Davis maneuvered to have his Congressional District gerrymandered to include as many Republicans as possible"
17        },
18        {
19          "id": 3,
20          "type": "from",
21          "text": "Genetically advanced agriculture, anti-aging technology, and other advancements we've yet to see today will likely push our longevity even farther"
22        },
23        {
24          "id": 4,
25          "type": "to",
26          "text": "It shouldn't be surprising, then, that many of the biggest proponents of banning performance enhancing drugs in sports are also suspect of some The Rosenkranz Foundation"
27        }
28      ]
29    }
30  }

const renderItem = ({ item }) => {
```



# LA GESTION DES ETATS - REACT NATIVE

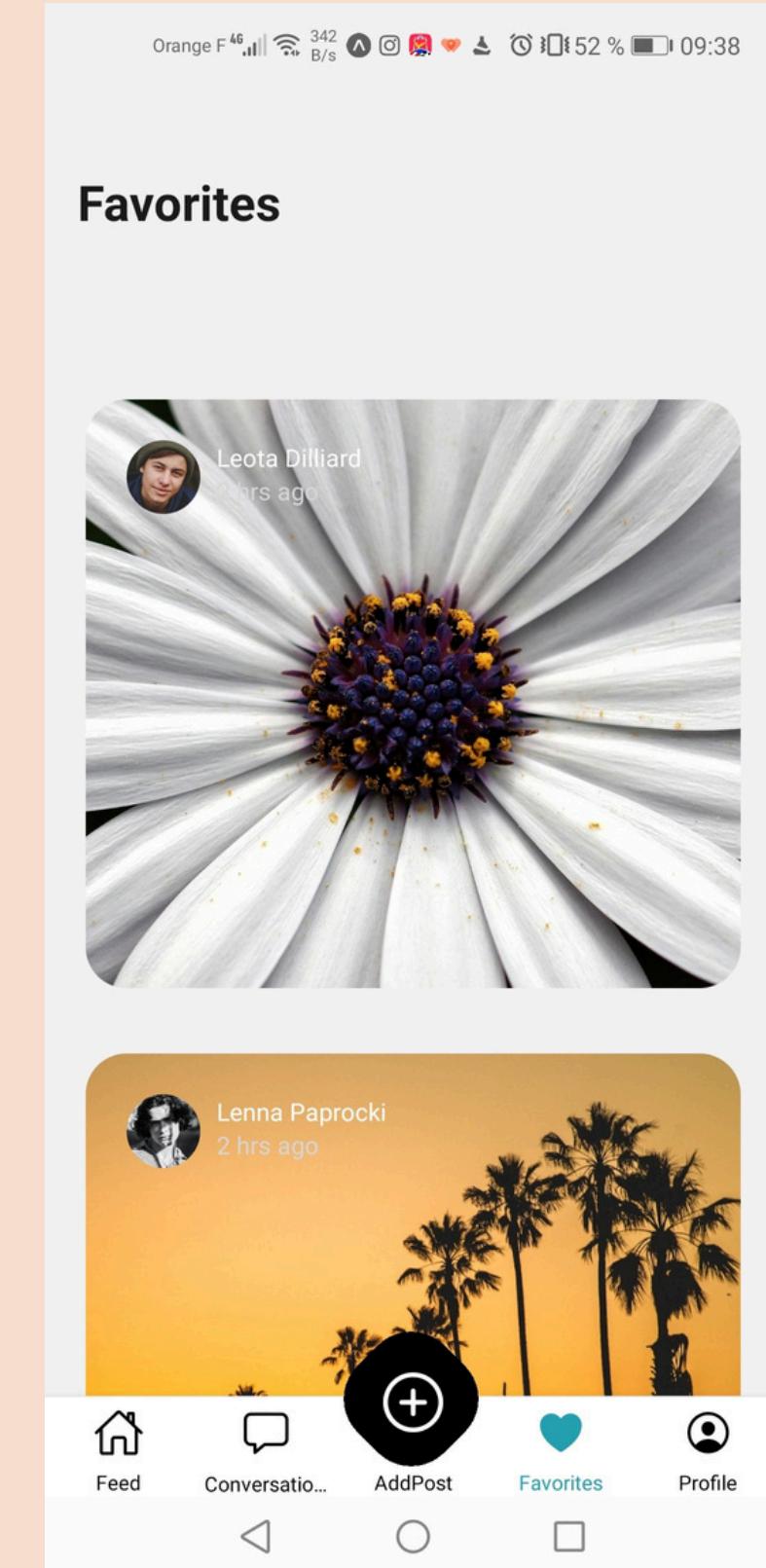
## LE PROJET

CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (FAVORITES)

```
src > surfaces > JS Favorites.js > ...
1 import React from "react";
2 import { useHeaderHeight } from "@react-navigation/elements";
3 import { SafeAreaView } from "react-native-safe-area-context";
4 import { ListOfFavorites } from "../components/ListOfFavorites";
5
6 export const Favorites = () => {
7   const headerHeight = useHeaderHeight();
8
9   return (
10     <SafeAreaView style={{ flex: 1, paddingTop: headerHeight }}>
11       <ListOfFavorites />
12     </SafeAreaView>
13   );
14 }

src > components > JS ListOfFavorites.js > [o] ListOfFavorites
1 import React, { useState, useEffect } from "react";
2 import { View, FlatList } from "react-native";
3 import { Card } from "../components/Card";
4 import AppLoading from "expo-app-loading";
5 import { requestBase } from "../utils/constants";
6
7 export const ListOfFavorites = () => {
8   const [cardList, setCardList] = useState(null);
9
10  useEffect(() => {
11    fetchCardData();
12  }, []);
13
14  if (!cardList) {
15    return <AppLoading />;
16  }
17  const renderItem = ({ item }) => {
18    return <Card item={item} />;
19};
20  return (
21    <View
22      style={{
23        paddingHorizontal: 20,
24      }>
25      <FlatList
26        data={cardList.reverse()}
27        renderItem={renderItem}
28        keyExtractor={(item) => item.itemId}
29        showsVerticalScrollIndicator={false}
30        snapToInterval={312}
31        decelerationRate='fast'
32      />
33    </View>
34  );
35}

41 }
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (PROFILE)

```
src > surfaces > JS Profile.js > Profile
1 import React from "react";
2 import { View, Text, Image } from "react-native";
3 import { SafeAreaView } from "react-native-safe-area-context";
4 import { ProfileStatistics } from "../components/ProfileStatistics";
5 import { AddedImages } from "../components/AddedImages";
6 import { FavoritedImages } from "../components/FavoritedImages";
7 import Ionicons from "@expo/vector-icons/Ionicons";
8
9 export const Profile = () => {
10   return (
11     <SafeAreaView style={{ flex: 1 }}>
12       <View
13         style={[
14           width: 650,
15           height: 570,
16           borderRadius: 155,
17           borderwidth: 1,
18           bordercolor: "#EEF2E2",
19           position: "absolute",
20           top: -210,
21           left: -120,
22           transform: [{ rotate: "-45deg" }],
23         ]}
24       />
25       <View
26         style={[
27           width: 650,
28           height: 570,
29           borderRadius: 155,
30           borderwidth: 1,
31           bordercolor: "#EEF2E2",
32           position: "absolute",
33           top: -260,
34           left: -120,
35           transform: [{ rotate: "-45deg" }],
36         ]}
37       />
38       <View
39         style={[
40           width: 650,
41           height: 570,
42           bordercolor: "#E1F6F4",
43           borderwidth: 1,
44           borderradius: 155,
45           position: "absolute",
46           top: -320,
47           left: -120,
48           backgroundColor: "#E1F6F4",
49           transform: [{ rotate: "-45deg" }],
50         ]}
51       />
52       <View style={{ paddingTop: 50 }}>
53         <View
54           style={[
55             width: 96,
56             height: 96,
57             borderRadius: 35,
58             bordercolor: "#000000",
59             borderwidth: 1,
60             transform: [{ rotate: "-45deg" }],
61             alignSelf: "center",
62           ]}
63         </View>
64         <View
65           style={[
66             overflow: "hidden",
67             alignSelf: "center",
68             transform: [{ rotate: "-45deg" }],
69             borderRadius: 35,
70             height: 87,
71             width: 87,
72             marginTop: -92,
73           ]}
74         >
75           <Image
76             style={[
77               height: 105,
78               width: 105,
79               transform: [{ rotate: "45deg" }],
80               alignSelf: "center",
81             ]}
82         </Image>
83       </View>
84     </View>
85   </SafeAreaView>
86 }
```

```
src > surfaces > JS Profile.js > Profile
9 export const Profile = () => {
10   return (
11     <SafeAreaView style={{ flex: 1 }}>
12       <View
13         style={[
14           width: 650,
15           height: 570,
16           borderRadius: 155,
17           position: "absolute",
18           top: -320,
19           left: -120,
20           backgroundColor: "#E1F6F4",
21           transform: [{ rotate: "-45deg" }],
22         ]}
23       />
24       <View style={{ paddingTop: 50 }}>
25         <View
26           style={[
27             width: 96,
28             height: 96,
29             borderRadius: 35,
30             bordercolor: "#000000",
31             borderwidth: 1,
32             transform: [{ rotate: "-45deg" }],
33             alignSelf: "center",
34           ]}
35         </View>
36         <View
37           style={[
38             overflow: "hidden",
39             alignSelf: "center",
40             transform: [{ rotate: "-45deg" }],
41             borderRadius: 35,
42             height: 87,
43             width: 87,
44             marginTop: -92,
45           ]}
46         >
47           <Image
48             style={[
49               height: 105,
50               width: 105,
51               transform: [{ rotate: "45deg" }],
52               alignSelf: "center",
53             ]}
54         </Image>
55       </View>
56     </View>
57   </SafeAreaView>
58 }
```

```
src > surfaces > JS Profile.js > Profile
9 export const Profile = () => {
10   return (
11     <SafeAreaView style={{ flex: 1 }}>
12       <View
13         style={[
14           width: 650,
15           height: 570,
16           borderRadius: 155,
17           position: "absolute",
18           top: -320,
19           left: -120,
20           backgroundColor: "#E1F6F4",
21           transform: [{ rotate: "-45deg" }],
22         ]}
23       />
24       <View style={{ paddingTop: 50 }}>
25         <View
26           style={[
27             width: 96,
28             height: 96,
29             borderRadius: 35,
30             bordercolor: "#000000",
31             borderwidth: 1,
32             transform: [{ rotate: "-45deg" }],
33             alignSelf: "center",
34           ]}
35         </View>
36         <Text
37           style={[
38             fontFamily: "Poppins_700Bold",
39             fontSize: 26,
40             alignSelf: "center",
41             marginTop: 30,
42           ]}
43         >
44           John Doe
45         </Text>
46         <Text
47           style={[
48             fontFamily: "Poppins_400Regular",
49             fontSize: 16,
50             alignSelf: "center",
51             marginTop: 10,
52           ]}
53         >
54           @johndoe
55         </Text>
56       </View>
57       <ProfileStatistics />
58       <View
59         style={[
60           marginTop: 40,
61           flexDirection: "row",
62           width: "100%",
63           justifyContent: "space-around",
64         ]}
65       >
66         <View style={{ alignItems: "center" }}>
67           <Ionicons name='image-outline' color='#000000' size={30} />
68           <AddedImages />
69         </View>
70         <View style={{ alignItems: "center" }}>
71           <Ionicons name='bookmark-outline' color='#000000' size={30} />
72           <FavoritedImages />
73         </View>
74       </View>
75     </SafeAreaView>
76   );
77 }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### CRÉATION ET STYLE DE COMPOSANTS : FEED - CONVERSATIONS - MESSAGES (PROFILE)

```
src > components > JS ProfileStatistics.js > ProfileStatistics
1 import React from "react";
2 import { View, Text, StyleSheet } from "react-native";
3
4 export const ProfileStatistics = () => {
5   return (
6     <View
7       style={{
8         paddingTop: 70,
9         paddingHorizontal: 20,
10        flexDirection: "row",
11        width: "100%",
12        justifyContent: "space-evenly",
13      }}
14    >
15      <View style={styles.general}>
16        <Text style={styles.headerText}>Posts</Text>
17        <Text style={styles.stats}>35</Text>
18      </View>
19      <View style={styles.general}>
20        <Text style={styles.headerText}>Followers</Text>
21        <Text style={styles.stats}>1,552</Text>
22      </View>
23      <View style={styles.general}>
24        <Text style={styles.headerText}>Follows</Text>
25        <Text style={styles.stats}>128</Text>
26      </View>
27    </View>
28  );
29}
30
31 const styles = StyleSheet.create({
32   general: {
33     alignItems: "center",
34   },
35   headerText: {
36     fontSize: 16,
37     fontFamily: "Poppins_400Regular",
38   },
39   stats: {
40     fontFamily: "Poppins_700Bold",
41     fontSize: 25,
42   },
43 });
44
```

```
src > components > JS AddedImages.js > AddedImages
1 <import React from "react";
2 <import {
3   View,
4   FlatList,
5   Pressable,
6   Image,
7   useWindowDimensions,
8 } from "react-native";
9
10 > const arrayOfImages = [...]
11 Click to expand the range.
12
13 <export const AddedImages = () => {
14   const imageWidth = useWindowDimensions().width * 0.4;
15
16   const renderItem = ({ item }) => {
17     return (
18       <Image
19         style={{
20           width: imageWidth,
21           height: 220,
22           borderRadius: 20,
23           marginBottom: 32,
24           borderColor: "#000000",
25         }}
26         source={{
27           uri: item.url,
28         }}
29       />
30     );
31   };
32   return (
33     <View
34       style={{
35         paddingHorizontal: 20,
36         paddingTop: 20,
37       }}
38     >
39       <FlatList
40         data={arrayOfImages}
41         renderItem={renderItem}
42         keyExtractor={(item) => item.id}
43         showsVerticalScrollIndicator={false}
44         snapToInterval={240}
45         decelerationRate='fast'
46       />
47     </View>
48   );
49 }
50
```

```
src > components > JS FavoritedImages.js > FavoritedImages
1 <import React from "react";
2 <import {
3   View,
4   FlatList,
5   Pressable,
6   Image,
7   useWindowDimensions,
8 } from "react-native";
9
10 > const arrayOfImages = [...]
11
12 <export const FavoritedImages = () => {
13   const imageWidth = useWindowDimensions().width * 0.4;
14
15   const renderItem = ({ item }) => {
16     return (
17       <Image
18         style={{
19           width: imageWidth,
20           height: 130,
21           borderRadius: 20,
22           marginBottom: 32,
23           borderColor: "#000000",
24         }}
25         source={{
26           uri: item.url,
27         }}
28       />
29     );
30   };
31   return (
32     <View
33       style={{
34         paddingHorizontal: 20,
35         paddingTop: 20,
36       }}
37     >
38       <FlatList
39         data={arrayOfImages}
40         renderItem={renderItem}
41         keyExtractor={(item) => item.id}
42         showsVerticalScrollIndicator={false}
43         />
44     </View>
45   );
46 }
47
```



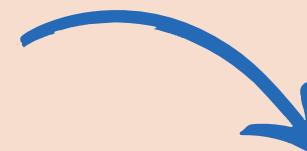
```
const arrayOfImages = [
  {
    id: 1,
    url: "https://images.unsplash.com/photo-1654512799227-94e56fbcd599?ixlib=rb-1.2.1&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1740&q=80",
  },
  {
    id: 2,
    url: "https://images.unsplash.com/photo-1654512799227-94e56fbcd599?ixlib=rb-1.2.1&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1740&q=80",
  },
  {
    id: 3,
    url: "https://images.unsplash.com/photo-1654512799227-94e56fbcd599?ixlib=rb-1.2.1&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1740&q=80",
  },
  {
    id: 4,
    url: "https://images.unsplash.com/photo-1654512799227-94e56fbcd599?ixlib=rb-1.2.1&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1740&q=80",
  },
  {
    id: 5,
    url: "https://images.unsplash.com/photo-1654512799227-94e56fbcd599?ixlib=rb-1.2.1&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1740&q=80",
  },
]
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (PROPS)

```
// src/App.js
export default function App() {
  const [userLoggedIn, setLoggedIn] = useState(true);
  const [userList, setUserList] = useState(null);
  //...
  async function fetchUserData(id) {
    const response = await fetch(requestBase + "/users.json");
    setUserList(await response.json());
  }
  useEffect(() => {
    fetchUserData();
  }, []);
  //...
  if (!userList) {
    return <AppLoading />;
  }
}
```



```
<Stack.Screen name='Home' options={{ headerShown: false }}>
  {(props) => <Home {...props} userList={userList} />}
</Stack.Screen>
```

# LA GESTION DES ÉTATS - REACT NATIVE

## LE PROJET

UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (CONTEXT API)

src > JS context.js > ...

```
1 import React from "react";
2
3 export const UserListContext = React.createContext(null);
```

src > components > JS ListOfAvatars.js > ListOfAvatars > <function>

```
3 import { ListHeaderComponent } from "./ListHeaderComponent";
4 import { UserListContext } from "../context";
5
6 export const ListOfAvatars = () => {
7   const renderItem = ({ item }) => {
8     return (
9       <Pressable onPress={() => console.log("hello")}>
10         <Image
11           style={{ height: 56, width: 56, borderRadius: 28, marginRight: 30 }}
12           source={{ uri: item.url, }}
13         />
14       </Pressable>
15     );
16   };
17
18   return (
19     <UserListContext.Consumer>
20       &{({ userList }) => (
21         <View style={{
22           zIndex: 100,
23           paddingVertical: 30,
24           paddingLeft: 20,
25           backgroundColor: "rgba(255,255,255, 0.85),
26         }}
27       >
28         <FlatList
29           data={userList}
30           renderItem={renderItem}
31           keyExtractor={(item) => item.id}
32           horizontal
33           ListHeaderComponent={<ListHeaderComponent />}
34           showsHorizontalScrollIndicator={false}
35           snapToInterval={86}
36           decelerationRate='fast'
37         />
38       </View>
39     )
20       </UserListContext.Consumer>
41     )
42   );
43 }
```

JS App.js > App

```
17 import { UserListContext } from './src/context';
18 import { requestBase } from "./src/utils/constants";
19
20
21 const Stack = createStackNavigator();
22 const Tab = createBottomTabNavigator();
23
24
25 export default function App() {
26   const [userLoggedIn, setIsUserLoggedIn] = useState(true);
27   const [userList, setUserList] = useState(null);
28
29   async function fetchUserData(id) {
30     const response = await fetch(requestBase + "/users.json");
31     setUserList(await response.json());
32   }
33   useEffect(() => {
34     fetchUserData();
35   }, []);
36 }
```

JS App.js > App

```
24 export default function App() {
25   return (
26     <SafeAreaProvider>
27       <UserListContext.Provider value={{ userList: userList }>
28         <NavigationContainer>
29           <Stack.Navigator>
30             <Stack.Screen
31               name="Home"
32               component={HomeScreen}
33             />
34           </Stack.Navigator>
35         </NavigationContainer>
36       </UserListContext.Provider>
37     </SafeAreaProvider>
38   );
39 }
```

# LA GESTION DES ÉTATS - REACT NATIVE

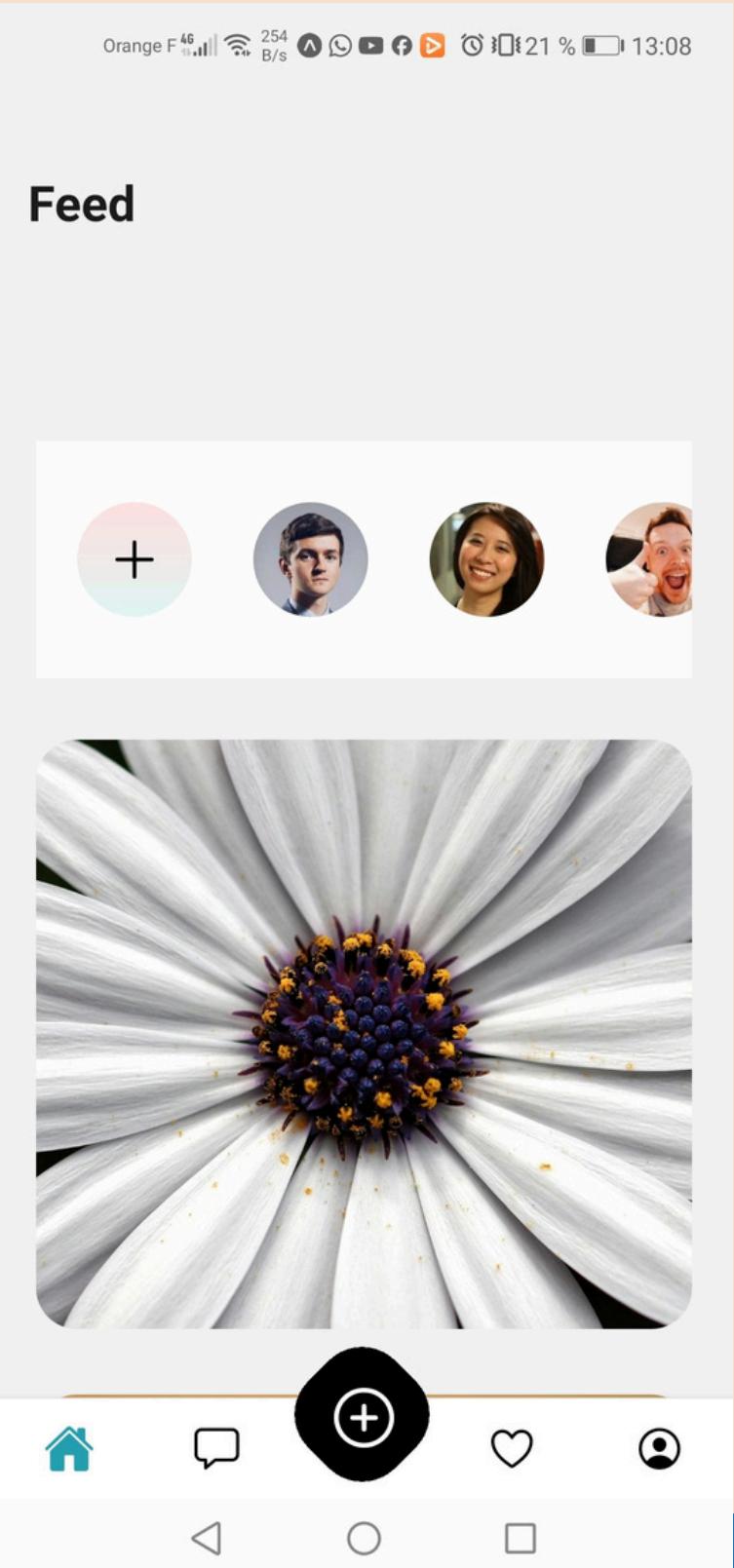
## LE PROJET

UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (CONTEXT API) - LISTOFAVATARS / LISTOFCARDS

```
src > components > JS ListOfAvatars.js > (1) ListOfAvatars > (2) <function>
  3 import { ListHeaderComponent } from "./ListHeaderComponent";
  4 import { UserListContext } from "../context";
  5
  6 export const ListOfAvatars = () => {
  7   const renderItem = ({ item }) => {
  8     return (
  9       <Pressable onPress={() => console.log("hello")}>
 10         <Image
 11           style={{ height: 56, width: 56, borderRadius: 28, marginRight: 30 }}
 12           source={{
 13             uri: item.url,
 14           }}
 15         />
 16       </Pressable>
 17     );
 18   };
 19   return (
 20     <UserListContext.Consumer>
 21       <({ userList }) => (
 22         <View style={{
 23           zIndex: 100,
 24           paddingVertical: 30,
 25           paddingLeft: 20,
 26           backgroundColor: "rgba(255,255,255, 0.85),
 27         }}
 28       >
 29         <FlatList
 30           data={userList}
 31           renderItem={renderItem}
 32           keyExtractor={({item}) => item.id}
 33           horizontal
 34           ListHeaderComponent={<ListHeaderComponent />}
 35           showsHorizontalScrollIndicator={false}
 36           snapToInterval={86}
 37           decelerationRate='fast'
 38         />
 39         <View>
 40       )
 41     </UserListContext.Consumer>
 42   );
 43 }
```

```
src > components > JS ListOfCards.js > ...
  1 import { useEffect, useState } from "react";
  2 import { View, Text } from "react-native";
  3 import { Card } from "../components/Card";
  4 import { FlatList } from "react-native-gesture-handler";
  5 import { requestBase } from "../utils/constants";
  6
  7 export const ListOfCards = () => {
  8
  9   const [cardList, setCardList] = useState(null);
 10   async function fetchCardData() {
 11     try {
 12       console.log(`requestBase ${requestBase}`);
 13       const response = await fetch(requestBase + "/home.json");
 14       const data = await response.json();
 15       console.log(`data json ok => ${JSON.stringify(data)}`);
 16       setCardList(data);
 17     } catch (error) {
 18       console.log(`listcard fetch error ${error.message}`);
 19     }
 20   }
 21   useEffect(() => {
 22     fetchCardData();
 23   }, []);
 24
 25   if (!cardList) {
 26     return (
 27       <View>
 28         <Text>Loading cardList ...</Text>
 29       </View>
 30     );
 31   }
 32   const renderItem = ({ item }) => {
 33     return <Card item={item} />;
 34   };
 35
 36   return (
 37     <View>
 38       <FlatList
 39         data={cardList}
 40         renderItem={renderItem}
 41         keyExtractor={({item}) => item.id}
 42         showsVerticalScrollIndicator={false}
 43       />
 44     </View>
 45   );
 46 }
```

```
src > components > JS Card.js > (1) Card > (2) <function>
  1 import React from "react";
  2 import { View, Text, Image } from "react-native";
  3 import { UserListContext } from "../context";
  4
  5 export const Card = ({ item }) => {
  6   return (
  7     <UserListContext.Consumer>
  8       <({ userList }) => {
  9         const currentUser = userList.filter(
 10           (user) => user.id === item.authorId
 11         );
 12         return (
 13           <View>
 14             <Image
 15               style={{
 16                 width: "100%",
 17                 height: 288,
 18                 borderRadius: 20,
 19                 marginBottom: 32,
 20               }}
 21               source={{
 22                 uri: item.url,
 23               }}
 24             />
 25             <View
 26               style={{
 27                 position: "absolute",
 28                 top: 20,
 29                 left: 20,
 30                 flexDirection: "row",
 31               }}
 32             >
 33               <Image
 34                 style={{
 35                   width: 36,
 36                   height: 36,
 37                   borderRadius: 20,
 38                   marginRight: 8,
 39                 }}
 40                 source={{
 41                   uri: currentUser[0].url,
 42                 }}
 43               />
 44               <View>
 45                 <Text style={{ color: "#ffffff", fontSize: 12 }}>
 46                   {currentUser[0].name}
 47                 </Text>
 48                 <Text style={{ color: "#D8D8D8", fontSize: 12 }}>
 49                   2 hrs ago
 50                 </Text>
 51               </View>
 52             </View>
 53           );
 54         );
 55       }
 56     </UserListContext.Consumer>
 57   );
 58 }
```



# LA GESTION DES ÉTATS - REACT NATIVE

## LE PROJET

### UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (CONTEXT API) - CONVERSATIONS - LISTOFCONVOS

```
src > JS context.js > ...
1 import React from "react";
2
3 export const UserListContext = React.createContext(null);
4 export const ConversationContext = React.createContext(null);
```

```
src > surfaces > JS ConversationsNavigation.js > [e] ConversationsNavigation > ⚡ <function>
1 import { Conversations } from "./Conversations";
2 import { Messages } from "./Messages";
3 import { createStackNavigator } from "@react-navigation/stack";
4 import { ConversationContext } from "../context";
5 import { useState } from "react";
6
7 const Stack = createStackNavigator();
8
9 export const ConversationsNavigation = () => {
10   const [conversationId, setConversationId] = useState(null);
11   return (
12     <ConversationContext.Provider
13       value={{
14         conversationId: conversationId,
15         setConversationId: setConversationId,
16       }}>
17     <Stack.Navigator
18       screenOptions={{
19         headerBackTitleVisible: false,
20         headerTintColor: "#000000",
21         headerTransparent: true,
22         headerTitleAlign: "left",
23         headerStyle: {
24           height: 160,
25         },
26         headerTitleStyle: {
27           textAlign: "left",
28           fontWeight: "bold",
29           fontFamily: "Poppins_700Bold",
30           fontSize: 24,
31         },
32       }}>
33     <Stack.Screen name='Conversations' component={Conversations} />
34     <Stack.Screen
35       name='Messages'
36       component={Messages}
37       options={({ route }) => ({}
38         title: route.params.name,
39         headerTitleStyle: {
40           textAlign: "center",
41           fontFamily: "Poppins_400Regular",
42           fontSize: 20,
43           position: "absolute",
44           top: 100,
45           left: 120,
46         },
47       )}
48     </Stack.Screen>
49   </Stack.Navigator>
50   <ConversationContext.Provider>
51     ;
52   </ConversationContext.Provider>
53 };
54
```

```
src > surfaces > JS Conversations.js > [e] Conversations
1 import React from "react";
2 import { View, TextInput, Pressable } from "react-native";
3 import { useHeaderHeight } from "@react-navigation/elements";
4 import { SafeAreaView } from "react-native-safe-area-context";
5 import Ionicons from "@expo/vector-icons/Ionicons";
6 import { ListOfConvos } from "../components/ListOfConvos";
7
8 export const Conversations = ({ navigation }) => {
9   const headerHeight = useHeaderHeight();
10  const [text, onChangeText] = React.useState("");
11
12  return (
13    <SafeAreaView style={{ flex: 1, paddingTop: headerHeight - 30 }}>
14      <View
15        style={{
16          width: 650,
17          height: 570,
18          borderRadius: 155,
19          borderWidth: 1,
20          borderColor: "#EEF2E2",
21          position: "absolute",
22          top: 210,
23          left: -160,
24          transform: [{ rotate: "-45deg" }],
25        }}
26      >
27        <View
28          style={{
29            width: 650,
30            height: 570,
31            borderRadius: 155,
32            borderWidth: 1,
33            borderColor: "#EEF2E2",
34            position: "absolute",
35            top: 280,
36            left: -160,
37            transform: [{ rotate: "-45deg" }],
38          }}
39        </View>
40        <View
41          style={{
42            width: 650,
43            height: 570,
44            borderRadius: 155,
45            position: "absolute",
46            top: 350,
47            left: -160,
48            backgroundColor: "#E1F6F4",
49          }}
50      </View>
51      <View style={{ marginHorizontal: 30, position: "relative" }}>
52        <View>
53          <TextInput
54            style={{
55              fontSize: 14,
56              paddingVertical: 12,
57              paddingLeft: 40,
58              marginHorizontal: 17,
59              borderRadius: 15,
60              backgroundColor: "#ffffff",
61              shadowColor: "#000000",
62              shadowOffset: { width: 0, height: 4 },
63              shadowOpacity: 0.1,
64              shadowRadius: 9,
65            }}
66            onChangeText={onChangeText}
67            value={text}
68            placeholder='search contacts'
69          />
70          <Ionicons
71            name='search'
72            size={24}
73            color='#000000'
74            style={{ position: "absolute", left: 28, top: 6 }}
75          />
76        </View>
77        <ListOfConvos navigation={navigation} />
78      </View>
79    </SafeAreaView>
80  );
81}
```

```
src > surfaces > JS Conversations.js > [e] Conversations
8 export const Conversations = ({ navigation }) => {
46   top: 350,
47   left: -160,
48   backgroundColor: "#E1F6F4",
49   transform: [{ rotate: "-45deg" }],
50 }
51 </View>
52 <View style={{ marginHorizontal: 30, position: "relative" }}>
53 <View>
54   <TextInput
55     style={{
56       fontSize: 14,
57       paddingVertical: 12,
58       paddingLeft: 40,
59       marginHorizontal: 17,
60       borderRadius: 15,
61       backgroundColor: "#ffffff",
62       shadowColor: "#000000",
63       shadowOffset: { width: 0, height: 4 },
64       shadowOpacity: 0.1,
65       shadowRadius: 9,
66     }}
67   onChangeText={onChangeText}
68   value={text}
69   placeholder='search contacts'
70 </View>
71 <Ionicons
72   name='search'
73   size={24}
74   color='#000000'
75   style={{ position: "absolute", left: 28, top: 6 }}
76 </Ionicons>
77 <ListOfConvos navigation={navigation} />
78 </View>
79 </SafeAreaView>
80 );
81 }
```

# LA GESTION DES ÉTATS - REACT NATIVE

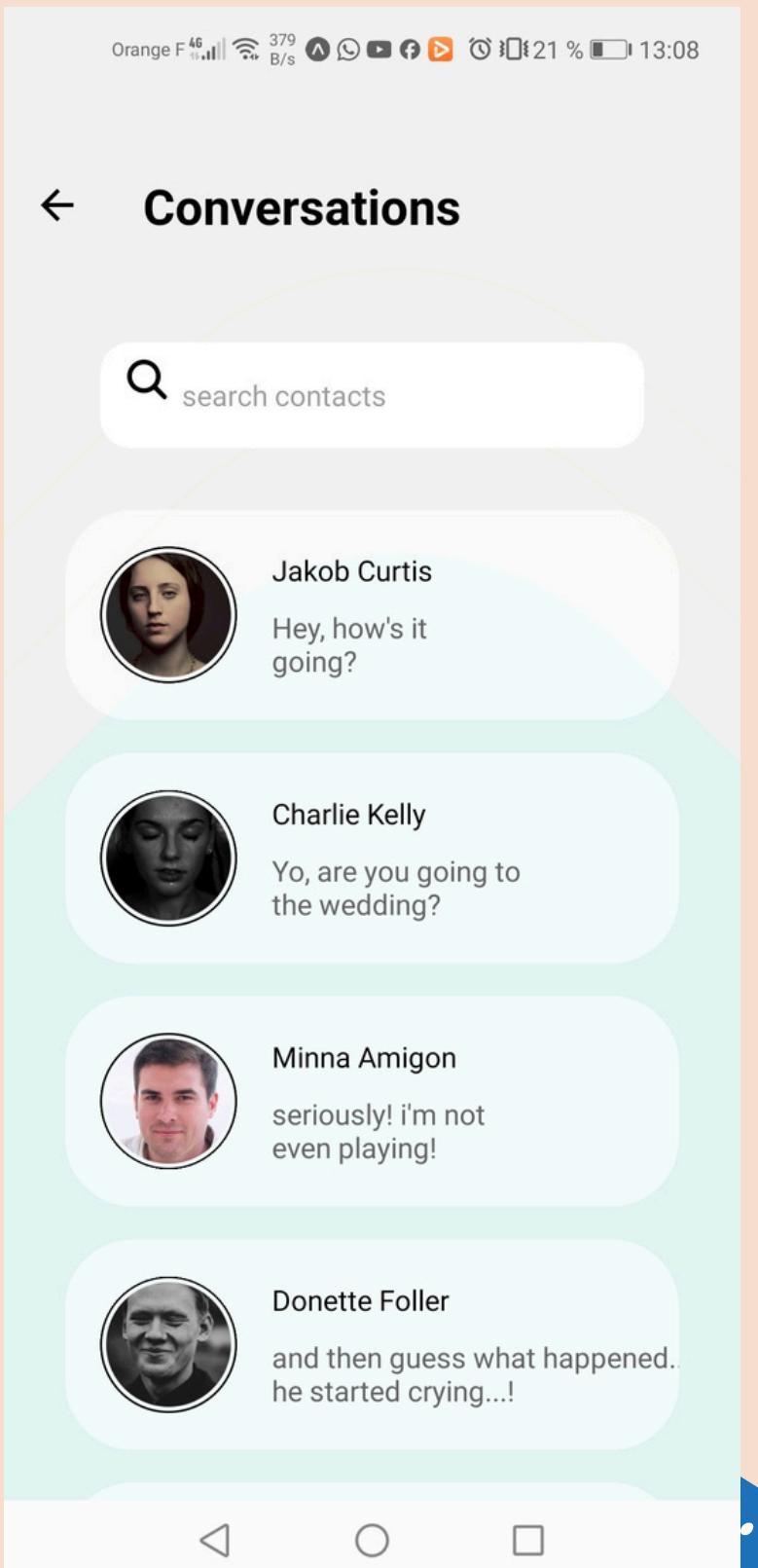
## LE PROJET

UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (CONTEXT API) - CONVERSATIONS - LISTOFCONVOS - CONVERSATIONITEM

```
src > components > js ListOfConvos.js > eo ListOfConvos > eo renderItem
1 import React, { useState, useEffect } from "react";
2 import { View, FlatList } from "react-native";
3 import { ConversationItem } from "./ConversationItem";
4 import { requestBase } from "../utils/constants";
5 import Loading from "./Loading";
6
7 export const ListOfConvos = ({ navigation }) => {
8
9   const [conversationsList, setConversationsList] = useState(null);
10
11  async function fetchConversationData() {
12    const response = await fetch(requestBase + "/conversations.json");
13    setConversationsList(await response.json());
14  }
15
16  useEffect(() => {
17    fetchConversationData();
18  }, []);
19
20  if (!conversationsList) {
21    return <Loading message="Loading conv list"/>
22  }
23
24  const renderItem = ({ item }) => [
25    return <ConversationItem navigation={navigation} item={item} />;
26  ];
27
28  return (
29    <View
30      style={{
31        paddingTop: 30,
32        marginTop: -30,
33        marginBottom: 50,
34      }}
35    >
36      <FlatList
37        data={conversationsList}
38        renderItem={renderItem}
39        keyExtractor={({ item }) => item.id}
40        showsVerticalScrollIndicator={false}
41        snapToInterval={119}
42        decelerationRate='fast'
43        ListHeaderComponent=<View style={{ height: 30 }} />
44      />
45    </View>
46  );
47};
```

```
src > components > js ConversationItem.js > eo ConversationItem
1 import React from "react";
2 import { View, Pressable, Image, Text } from "react-native";
3 import { UserListContext, ConversationContext } from "../context";
4
5 export const ConversationItem = ({ navigation, item }) => [
6
7   const onPressItem = (setConversationId, currentUser) => {
8     setConversationId(item.id);
9     navigation.navigate("Messages", {
10       name: currentUser[0].name,
11       avatar: currentUser[0].url,
12     });
13   };
14
15   return (
16     <ConversationContext.Consumer>
17       &({ setConversationId }) => (
18         <UserListContext.Consumer>
19           &({ userList }) => {
20             const currentUser = userList.filter(
21               (user) => user.id === item.userId
22             );
23             return (
24               <Pressable
25                 onPress={() => onPressItem(setConversationId, currentUser)}
26                 style={{
27                   height: 103,
28                   backgroundColor: "rgba(255,255,255,0.6)",
29                   borderRadius: 33,
30                   marginBottom: 16,
31                   justifyContent: "flex-start",
32                   alignItems: "center",
33                   flexDirection: "row",
34                 }}
35             >
36               <Image
37                 style={{
38                   width: 61,
39                   height: 61,
40                   borderRadius: 35,
41                   margin: 10,
42                   marginHorizontal: 10,
43                 }}
44             </Image>
45             <View>
46               <Text style={{ fontSize: 14, padding: 0 }}>
47                 {currentUser[0].name}
48               </Text>
49               <Text style={{ color: "#656565", width: "65%" }}>
50                 {item.text}
51               </Text>
52             </View>
53           </Pressable>
54         </UserListContext.Consumer>
55       &({}) => (
56         <ConversationContext.Consumer>
57           &({}) => (
58             <Image
59               style={{
60                 width: 61,
61                 height: 61,
62                 borderRadius: 35,
63                 margin: 10,
64                 marginHorizontal: 10,
65                 border: 1px solid #000000,
66                 margin: 10,
67               }}
68             </Image>
69           </ConversationContext.Consumer>
70         &({}) => (
71           <Image
72             style={{
73               width: 61,
74               height: 61,
75               borderRadius: 35,
76               margin: 10,
77             }}
78         </Image>
79       );
80     </ConversationContext.Consumer>
81   );
82 };
83
84 </UserListContext.Consumer>
85 </ConversationContext.Consumer>
86 );
```

```
src > components > js ConversationItem.js > eo ConversationItem > o <function> > o <function>
5 export const ConversationItem = ({ navigation, item }) => {
6   &({ setConversationId }) => (
7     &({ userList }) => {
8       <Image
9         style={{
10           width: 61,
11           height: 61,
12           borderRadius: 35,
13           margin: 10,
14           marginHorizontal: 10,
15           border: 1px solid #000000,
16           margin: 10,
17         }}
18       </Image>
19       <View>
20         <Text style={{ fontSize: 14, padding: 0 }}>
21           {currentUser[0].name}
22         </Text>
23         <Text style={{ color: "#656565", width: "65%" }}>
24           {item.text}
25         </Text>
26       </View>
27     </Image>
28   );
29 };
30
31 </UserListContext.Consumer>
32 </ConversationContext.Consumer>
33 );
```



# LA GESTION DES ÉTATS - REACT NATIVE

## LE PROJET

UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (CONTEXT API) - CONVERSATIONS - MESSAGES - LISTOFMESSAGES

```
src > surfaces > JS Messages.js > @@Messages
1 import React from "react";
2 import { View, Image } from "react-native";
3 import { useHeaderHeight } from "@react-navigation/elements";
4 import { SafeAreaView } from "react-native-safe-area-context";
5 import { ListOfMessages } from "../components/ListOfMessages";
6 import { ConversationContext } from "../context";
7
8 export const Messages = ({ route }) => [
9   const headerHeight = useHeaderHeight();
10
11   return (
12     <SafeAreaView style={{ flex: 1, paddingTop: headerHeight + 100 }}>
13       <ConversationContext.Consumer>
14         {(conversationId) => (
15           <View
16             style={{
17               position: "absolute",
18               top: 40,
19               left: "2%",
20               backgroundColor: "#E1F6F4",
21               width: "96%",
22               height: 255,
23               borderRadius: 34,
24             }}
25           />
26           <View
27             style={{
28               width: 650,
29               height: 570,
30               borderRadius: 155,
31               border: 1px solid #fffffe,
32               position: "absolute",
33               top: 250,
34               left: -300,
35               transform: [{ rotate: "-45deg" }],
36             }}
37           />
38           <View
39             style={{
40               width: 650,
41               height: 570,
42               borderRadius: 155,
43               border: 1px solid #fffffe,
44               position: "absolute",
45               top: 250,
46               left: -300,
47               transform: [{ rotate: "-45deg" }],
48             }}
49           />
50           <View
51             style={{
52               position: "absolute",
53               top: 190,
54               left: -290,
55               transform: [{ rotate: "-45deg" }],
56             }}
57           />
58           <View
59             style={{
60               position: "absolute",
61               top: 125,
62               left: 70,
63             }}
64           />
65           <View
66             style={{
67               width: 84,
68               height: 84,
69               borderRadius: 35,
70               border: 1px solid #000000,
71               position: "absolute",
72               top: -3,
73               left: -3,
74             }}
75           />
76           <Image
77             style={{ height: 78, width: 78, borderRadius: 40 }}
78             source={{
79               uri: route.params.avatar,
80             }}
81           />
82           <ListOfMessages conversationId={conversationId} />
83         )
84       </ConversationContext.Consumer>
85     </SafeAreaView>
86   );
87 ];
```

```
src > surfaces > JS Messages.js > @@Messages >  <function>
8 export const Messages = ({ route }) => [
14   {(conversationId) => (
45     <View
46       style={{
47         border: 1px solid #000000,
48         borderLeftWidth: 1,
49         borderLeftColor: "#fffffe",
50         position: "absolute",
51         top: 190,
52         left: -290,
53         transform: [{ rotate: "-45deg" }],
54       }}
55     />
56     <View
57       style={{
58         position: "absolute",
59         top: 125,
60         left: 70,
61       }}
62     />
63     <View
64       style={{
65         width: 84,
66         height: 84,
67         borderRadius: 35,
68         border: 1px solid #000000,
69         borderLeft: 1px solid #fffffe,
70         transform: [{ rotate: "-45deg" }],
71         position: "absolute",
72         top: -3,
73         left: -3,
74       }}
75     />
76     <Image
77       style={{ height: 78, width: 78, borderRadius: 40 }}
78       source={{
79         uri: route.params.avatar,
80       }}
81     />
82     <ListOfMessages conversationId={conversationId} />
83   )
84 }
85 ];
```

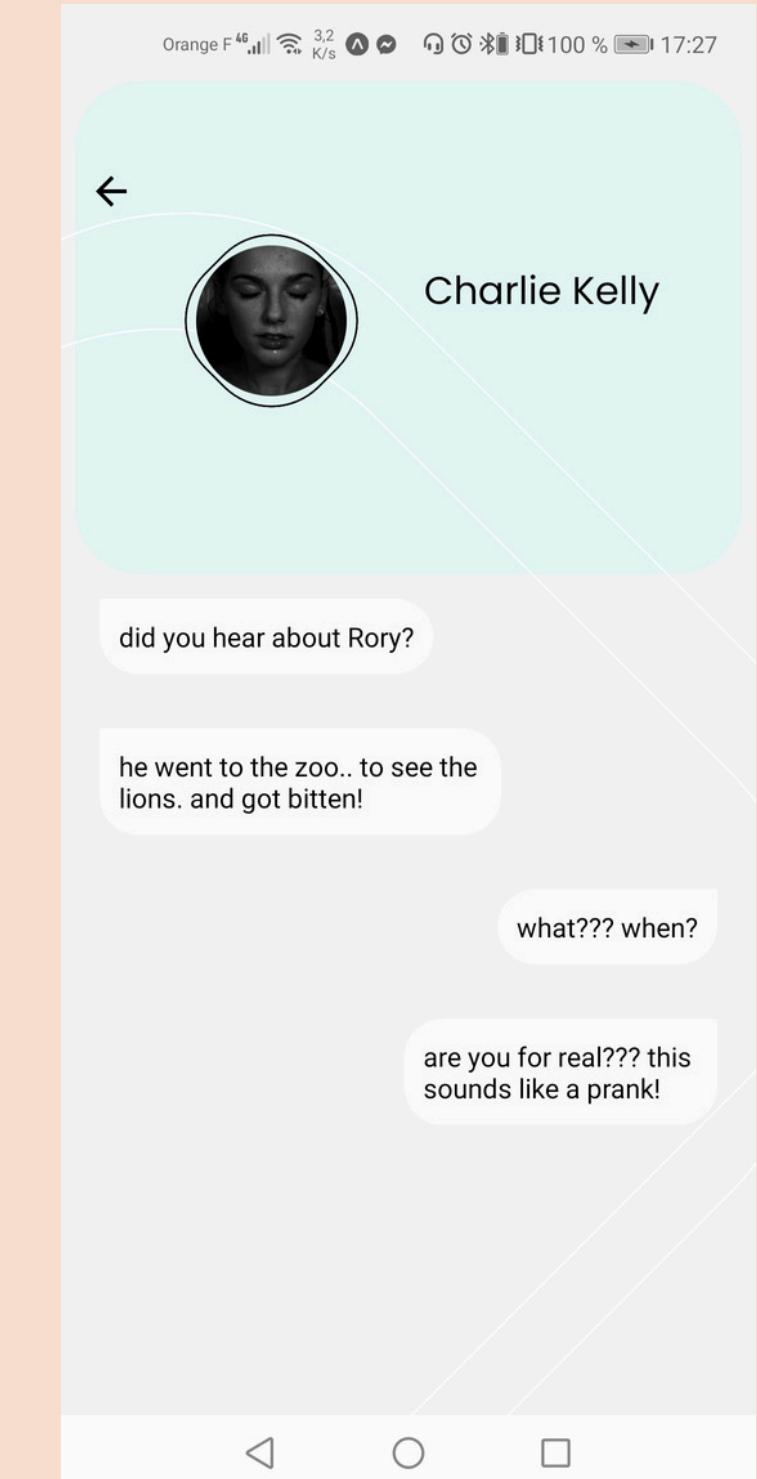
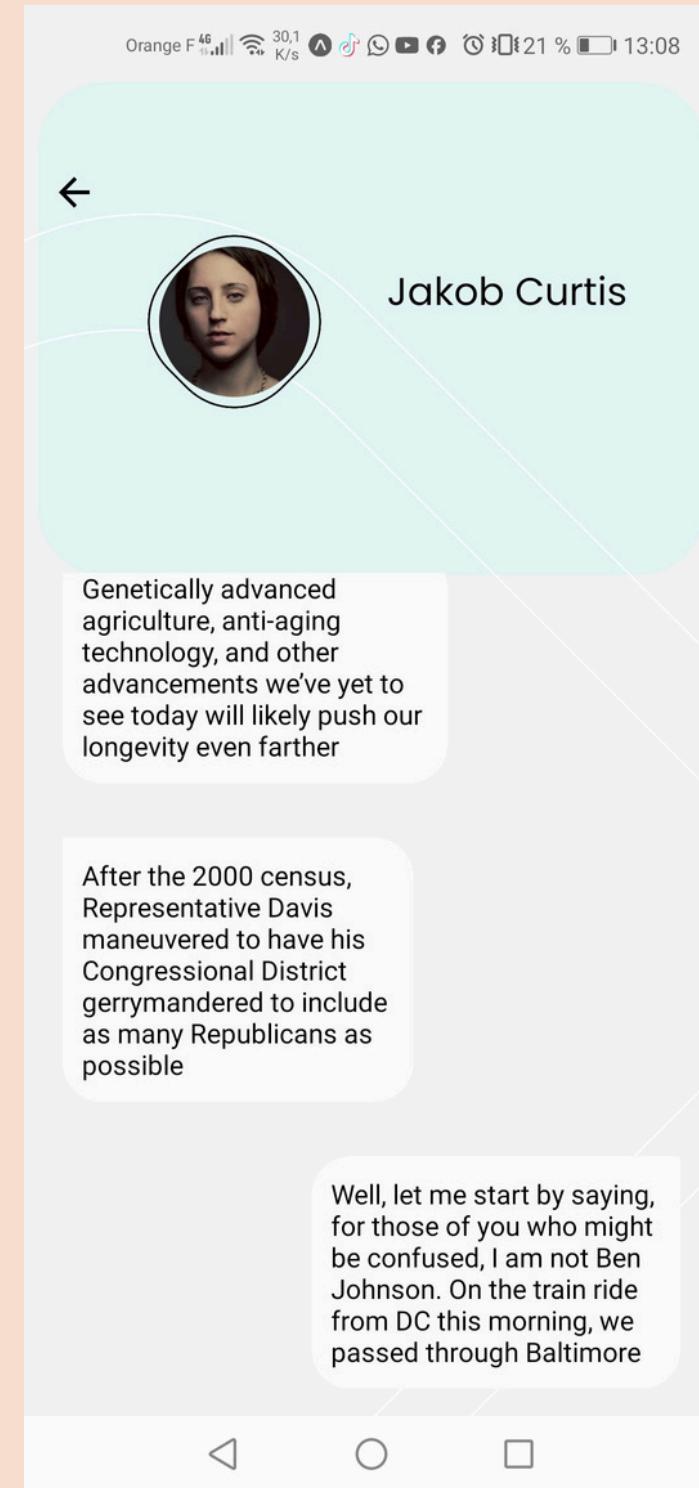
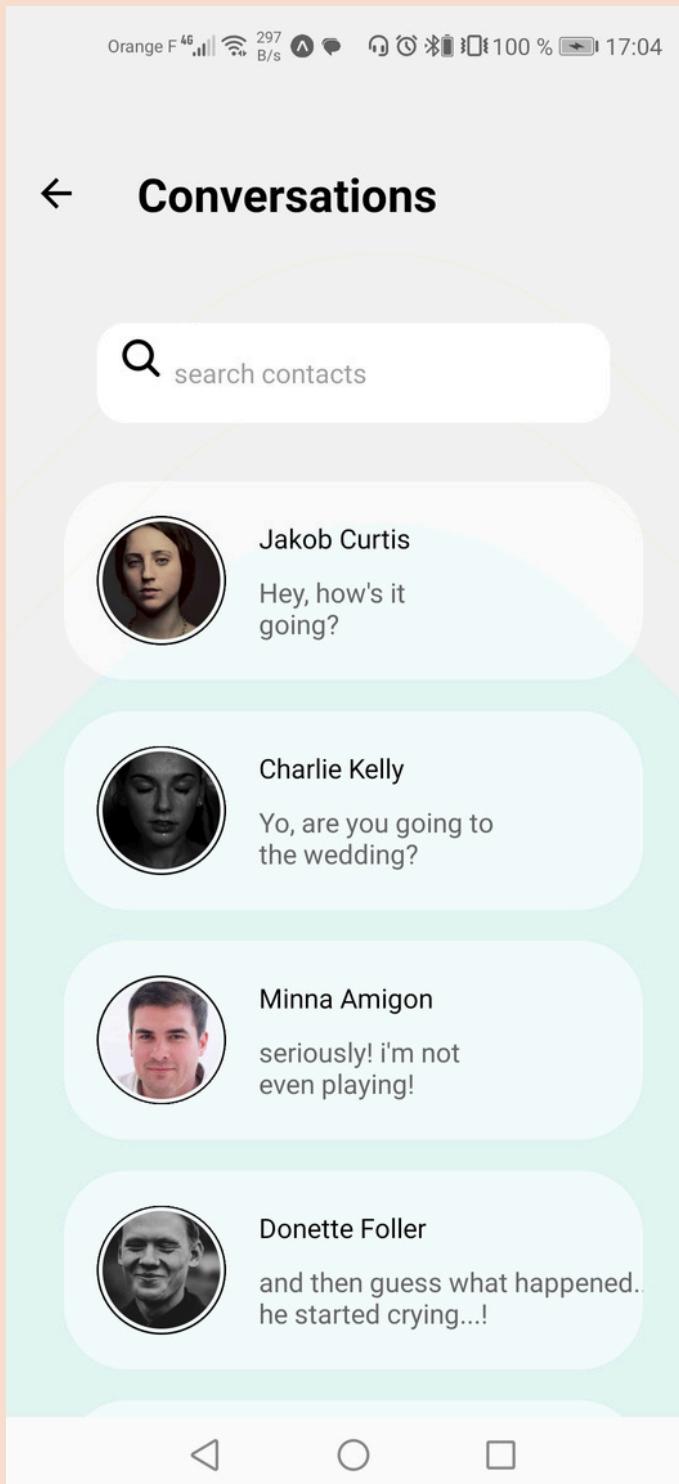
```
src > components > JS ListOfMessages.js > @@ListOfMessages
1 import React, { useState, useEffect } from "react";
2 import { View, FlatList, Text, StyleSheet } from "react-native";
3 import { requestBase } from "../utils/constants";
4 import Loading from "./Loading";
5
6 export const ListOfMessages = ({ conversationId }) => {
7   const [messages, setMessages] = useState(null);
8
9   async function fetchMessages() {
10     const response = await fetch(
11       requestBase + "/messages/" + conversationId + ".json"
12     );
13     setMessages(await response.json());
14   }
15
16   useEffect(() => {
17     fetchMessages();
18   }, []);
19   if (!messages) {
20     return <Loading message="loading message"/>
21   }
22
23   const renderItem = ({ item }) => {
24     return (
25       <View
26         style={[
27           styles.text,
28           item.type === "from" ? styles.textTo : styles.textFrom,
29         ]}
30       >
31         <Text style={{}}>{item.text}</Text>
32       </View>
33     );
34   };
35
36   return (
37     <View
38       style={{
39         padding: 10,
40         paddingVertical: 15,
41         margin: 10,
42         marginVertical: 10,
43         border: 1px solid #000000,
44         borderLeftWidth: 0,
45         borderLeftColor: "#fffffe",
46         borderRightWidth: 0,
47         borderRightColor: "#fffffe",
48         borderRadius: 10,
49         overflow: "hidden",
50       }}
51     >
52       <FlatList
53         data={messages.messages}
54         renderItem={renderItem}
55         keyExtractor={(item) => item.id}
56       >
57     </View>
58   );
59 }
```

```
src > components > JS ListOfMessages.js > ...
6 export const ListOfMessages = ({ conversationId }) => {
42   const messages = useState(null);
43   const renderItem = ({ item }) => {
44     const [text, setText] = useState(item.text);
45     const [textFrom, setTextFrom] = useState(item.type === "from");
46     const [textTo, setTextTo] = useState(item.type === "to");
47     const [isInverted, setIsInverted] = useState(false);
48
49     const styles = StyleSheet.create({
50       text: {
51         backgroundColor: "#FAFAFA",
52         border: 1px solid #000000,
53         borderLeftWidth: 0,
54         borderLeftColor: "#fffffe",
55         borderRightWidth: 0,
56         borderRightColor: "#fffffe",
57         padding: 10,
58         margin: 10,
59         width: "100%",
60         height: 40,
61         borderRadius: 10,
62         overflow: "hidden",
63         position: "relative",
64         alignSelf: "flex-end",
65       },
66       textFrom: {
67         border: 1px solid #000000,
68         borderLeftWidth: 0,
69         borderLeftColor: "#fffffe",
70         borderRightWidth: 0,
71         borderRightColor: "#fffffe",
72         padding: 10,
73         margin: 10,
74         width: "100%",
75         height: 40,
76         borderRadius: 10,
77         overflow: "hidden",
78         position: "relative",
79         alignSelf: "flex-start",
80       },
81       textTo: {
82         border: 1px solid #000000,
83         borderLeftWidth: 0,
84         borderLeftColor: "#fffffe",
85         borderRightWidth: 0,
86         borderRightColor: "#fffffe",
87         padding: 10,
88         margin: 10,
89         width: "100%",
90         height: 40,
91         borderRadius: 10,
92         overflow: "hidden",
93         position: "relative",
94         alignSelf: "flex-end",
95       },
96     });
97
98     return (
99       <View
100         style={{
101           width: "100%",
102           height: 40,
103           border: 1px solid #000000,
104           borderLeftWidth: 0,
105           borderLeftColor: "#fffffe",
106           borderRightWidth: 0,
107           borderRightColor: "#fffffe",
108           padding: 10,
109           margin: 10,
110           borderRadius: 10,
111           overflow: "hidden",
112           position: "relative",
113           alignSelf: "flex-end",
114         }}
115       >
116         <Text style={{}}>{text}</Text>
117       </View>
118     );
119   };
120
121   const styles = StyleSheet.create({
122     text: {
123       backgroundColor: "#FAFAFA",
124       border: 1px solid #000000,
125       borderLeftWidth: 0,
126       borderLeftColor: "#fffffe",
127       borderRightWidth: 0,
128       borderRightColor: "#fffffe",
129       padding: 10,
130       margin: 10,
131       width: "100%",
132       height: 40,
133       borderRadius: 10,
134       overflow: "hidden",
135       position: "relative",
136       alignSelf: "flex-end",
137     },
138     textFrom: {
139       border: 1px solid #000000,
140       borderLeftWidth: 0,
141       borderLeftColor: "#fffffe",
142       borderRightWidth: 0,
143       borderRightColor: "#fffffe",
144       padding: 10,
145       margin: 10,
146       width: "100%",
147       height: 40,
148       borderRadius: 10,
149       overflow: "hidden",
150       position: "relative",
151       alignSelf: "flex-start",
152     },
153     textTo: {
154       border: 1px solid #000000,
155       borderLeftWidth: 0,
156       borderLeftColor: "#fffffe",
157       borderRightWidth: 0,
158       borderRightColor: "#fffffe",
159       padding: 10,
160       margin: 10,
161       width: "100%",
162       height: 40,
163       borderRadius: 10,
164       overflow: "hidden",
165       position: "relative",
166       alignSelf: "flex-end",
167     },
168   });
169
170   return (
171     <View
172       style={{
173         width: "100%",
174         height: 40,
175         border: 1px solid #000000,
176         borderLeftWidth: 0,
177         borderLeftColor: "#fffffe",
178         borderRightWidth: 0,
179         borderRightColor: "#fffffe",
180         padding: 10,
181         margin: 10,
182         borderRadius: 10,
183         overflow: "hidden",
184         position: "relative",
185         alignSelf: "flex-end",
186       }}
187     >
188       <Text style={{}}>{text}</Text>
189     </View>
190   );
191 
```

# LA GESTION DES ÉTATS - REACT NATIVE

## LE PROJET

UTILISATION DES ÉTATS : EXTRAIRE DES DONNÉES POUR L'APPLICATION (CONTEXT API) - CONVERSATIONS - MESSAGES - LISTOFMESSAGES



# LA GESTION DES ETATS - REACT NATIVE

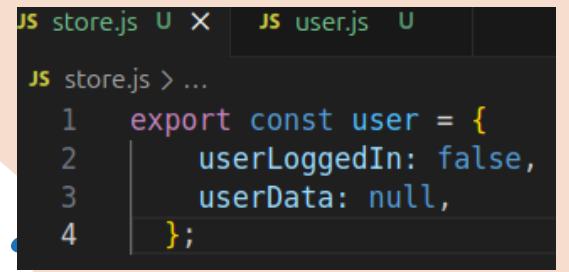
## LE PROJET

### IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : STORE - ACTIONS - REDUCERS

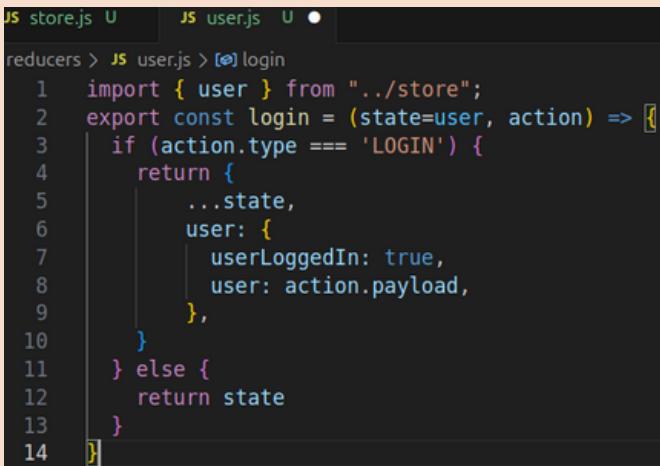
```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npm install @reduxjs/toolkit
added 5 packages, and audited 1277 packages in 4s
133 packages are looking for funding
  run `npm fund` for details
```

```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npm install react-redux
added 3 packages, and audited 1280 packages in 3s
133 packages are looking for funding
  run `npm fund` for details
```

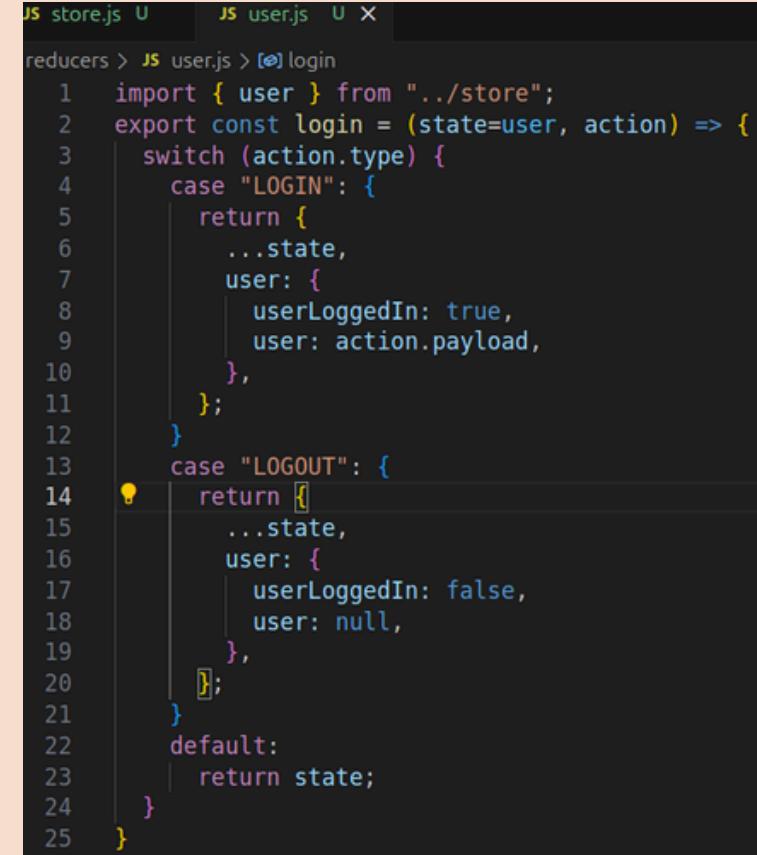
```
• balde@balde-ASUS-TUF-Gaming-F15-FX507ZC4-FX567ZC4:~/www/react_native/book-app$ npm install --save-dev @redux-devtools/core
added 2 packages, and audited 1282 packages in 4s
133 packages are looking for funding
  run `npm fund` for details
```



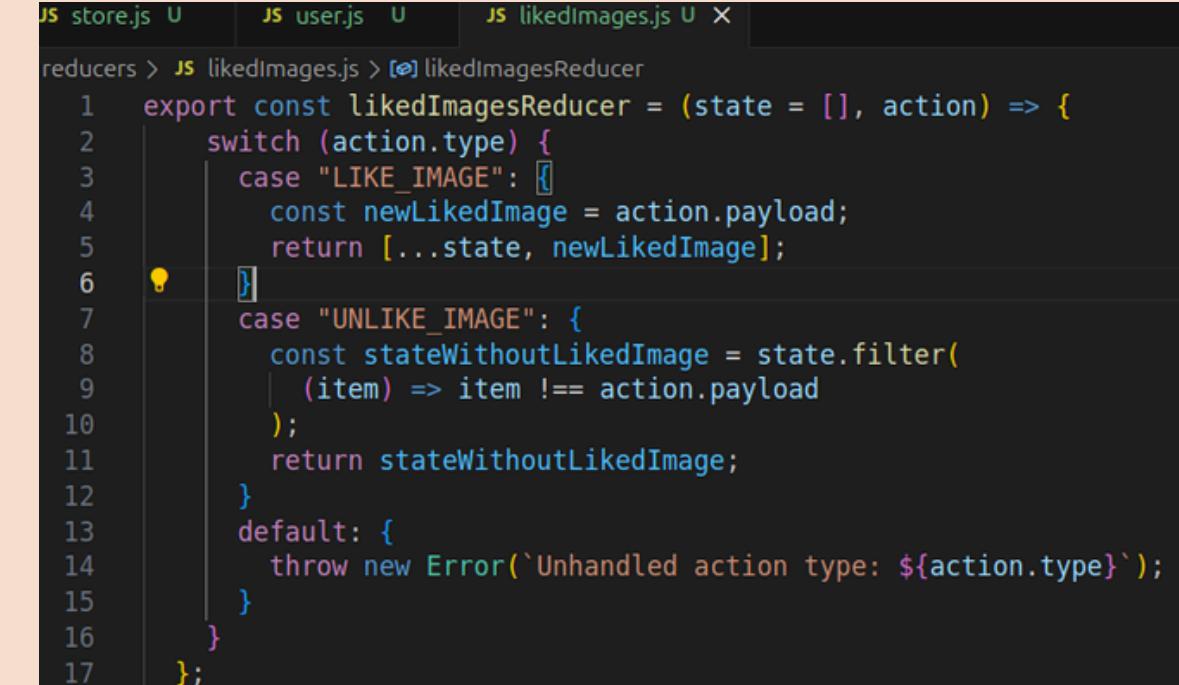
```
JS store.js U X JS user.js U
JS store.js > ...
1 export const user = {
2   userLoggedIn: false,
3   userData: null,
4 }
```



```
JS store.js U JS user.js U
reducers > JS user.js > [o] login
1 import { user } from "../store";
2 export const login = (state=user, action) => [
3   if (action.type === 'LOGIN') {
4     return {
5       ...state,
6       user: {
7         userLoggedIn: true,
8         user: action.payload,
9       },
10    }
11  } else {
12    return state
13  }
14 ]
```



```
JS store.js U JS user.js U X
reducers > JS user.js > [o] login
1 import { user } from "../store";
2 export const login = (state=user, action) => {
3   switch (action.type) {
4     case "LOGIN": {
5       return {
6         ...state,
7         user: {
8           userLoggedIn: true,
9           user: action.payload,
10          },
11        };
12      }
13      case "LOGOUT": {
14        return [
15          ...state,
16          user: {
17            userLoggedIn: false,
18            user: null,
19          },
20        ];
21      }
22      default:
23        return state;
24    }
25 }
```

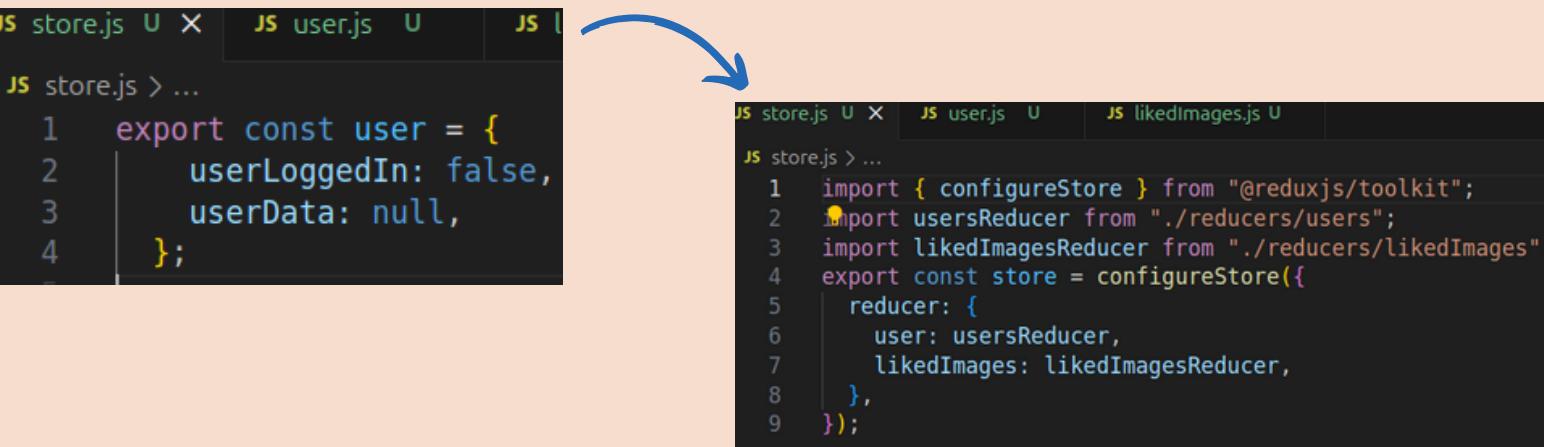


```
JS store.js U JS user.js U JS likedImages.js U X
reducers > JS likedImages.js > [o] likedImagesReducer
1 export const likedImagesReducer = (state = [], action) => {
2   switch (action.type) {
3     case "LIKE_IMAGE": {
4       const newLikedImage = action.payload;
5       return [...state, newLikedImage];
6     }
7     case "UNLIKE_IMAGE": {
8       const stateWithoutLikedImage = state.filter(
9         (item) => item !== action.payload
10       );
11       return stateWithoutLikedImage;
12     }
13     default: {
14       throw new Error(`Unhandled action type: ${action.type}`);
15     }
16   }
17 }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : COMBINAISON DE REDUCERS - TOOLKIT REDUX



```
JS store.js U X JS user.js U JS likedImages.js U
JS store.js > ...
1 export const user = {
2   userLoggedIn: false,
3   userData: null,
4 };

```

```
JS store.js U X JS user.js U JS likedImages.js U
JS store.js > ...
1 import { configureStore } from "@reduxjs/toolkit";
2 import usersReducer from "./reducers/users";
3 import likedImagesReducer from "./reducers/likedImages";
4 export const store = configureStore({
5   reducer: {
6     user: usersReducer,
7     likedImages: likedImagesReducer,
8   },
9 });

```

```
JS store.js U JS App.js M ● JS user.js U JS likedImages.js U
JS App.js > ...
19 import Loading from './src/components>Loading';
20 import { store } from "./store";
21 import { Provider } from "react-redux";
22 
```

```
JS App.js > ⚡ App
27 export default function App() {
28   return (
29     <SafeAreaProvider>
30       <Provider store={store}>
31         <UserListContext.Provider value={{ userList: userList }}>
32           <NavigationContainer>
```

# LA GESTION DES ETATS - REACT NATIVE

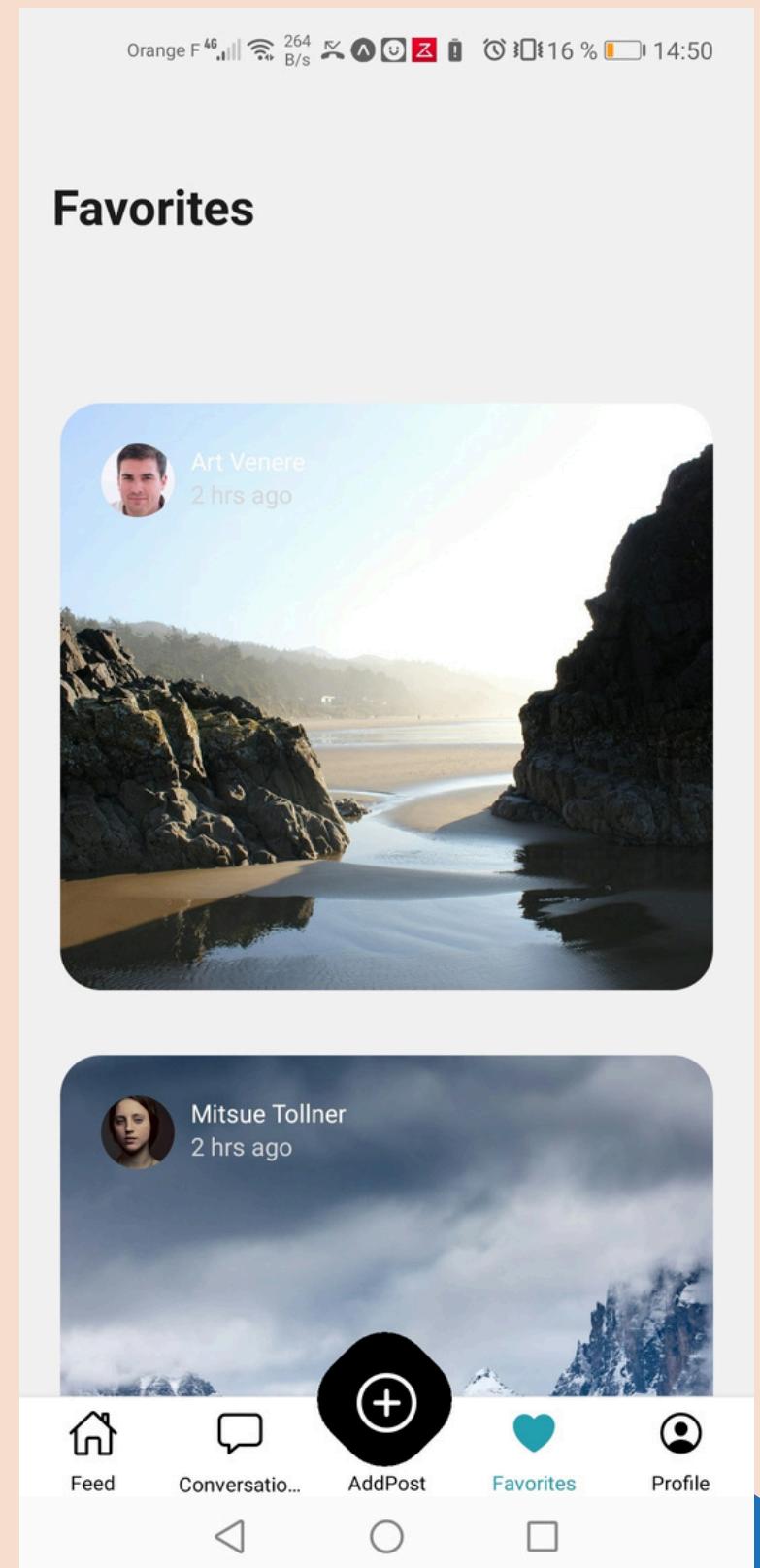
## LE PROJET

### IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : COMBINAISON DE REDUCERS - TOOLKIT REDUX : FAVORITE

```
reducers > js likedImages.js > likedImagesSlice

1 import { createSlice } from "@reduxjs/toolkit";
2 import { fetchLikedImages } from "../asyncFetches";
3
4 export const likedImagesSlice = createSlice({
5   name: "likedImages",
6   initialState: {
7     likedImages: [],
8     loading: true,
9   },
10  reducers: {
11    likeImage: (state, action) => {
12      const newLikedImage = action.payload;
13      return { ...state, likedImages: [...state.likedImages, newLikedImage] };
14    },
15    unLikeImage: (state, action) => {
16      const stateWithoutLikedImage = state.likedImages.filter(
17        (item) => item.itemId !== action.payload.itemId
18      );
19      state.likedImages = stateWithoutLikedImage;
20    },
21  },
22  extraReducers: (builder) => {
23    builder.addCase(fetchLikedImages.pending, (state) => {
24      state.loading = true;
25    });
26    builder.addCase(fetchLikedImages.fulfilled, (state, action) => {
27      state.likedImages = action.payload;
28      state.loading = false;
29    });
30    builder.addCase(fetchLikedImages.rejected, (state) => {
31      state.loading = false;
32    });
33  },
34);
35
36 export const { initLikedImages, likeImage, unLikeImage } =
37   likedImagesSlice.actions;
38
39 export default likedImagesSlice.reducer;
```

```
src > surfaces > js Home.js > ...
1 import { View } from "react-native";
2 import { fetchLikedImages } from "../../asyncFetches";
3 import { useDispatch } from "react-redux";
4
5 const Tab = createBottomTabNavigator();
6
7 const ConversationsBase = () => <View style={{ flex: 1 }} />;
8
9 export const Home = () => {
10   const dispatch = useDispatch();
11
12   useEffect(() => {
13     dispatch(fetchLikedImages());
14   }, []);
15
16   return (
17     <Tab.Navigator>
18       <Tab.Screen name="Home" component={ConversationsBase} />
19     </Tab.Navigator>
20   );
21 }
22
23
24
25
26
27
src > components > js ListOfFavorites.js > ListOfFavorites > useEffect() callback
1 import React, { useState, useEffect } from "react";
2 import { View, FlatList } from "react-native";
3 import { Card } from "../components/Card";
4 import { useSelector } from "react-redux";
5 import Loading from "./Loading";
6
7 export const ListOfFavorites = ({ navigation }) => {
8   const [ likedImages ] = useSelector((state) => state.likedImages);
9   const [ imageList, setImageList ] = useState([]);
10
11   if (!imageList) {
12     return <Loading message="loading image list"/>
13   }
14
15   useEffect(() => {
16     const reversedImages = [...likedImages].reverse();
17     setImageList(reversedImages);
18   }, [likedImages]);
19
20   const renderItem = ({ item }) => {
21     return <Card item={item} navigation={navigation} />;
22   }
23
24   return (
25     <View
26       style={{
27         paddingHorizontal: 20,
28       }}
29     >
30       <FlatList
31         data={imageList}
32         renderItem={renderItem}
33         keyExtractor={(item) => item.itemId}
34         showsVerticalScrollIndicator={false}
35         snapToInterval={312}
36         decelerationRate='fast'
37       />
38     </View>
39   );
}
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : COMBINAISON DE REDUCERS - TOOLKIT REDUX : USERDETAILSMODAL

```
JS App.js > ⚡ App > ⚡ FetchUserData
19 import Loading from './src/components>Loading';
20 import store from "./store";
21 import { Provider } from "react-redux";
22 import { UserDetailsModal } from './src/surfaces/UserDetailsModal';
23

<SafeAreaProvider>
  <Provider store={store}>
    <UserListContext.Provider value={{ userList: userList }}>
      <NavigationContainer>
        <Stack.Navigator>
          <Stack.Group>
            {!userLoggedIn ? (
              <Stack.Screen name='Login' component={Login} />
            ) : (
              <>
                <Stack.Screen
                  name='Home'
                  component={Home}
                  options={{ headerShown: false }}
                />
                <Stack.Screen
                  name='ConversationsNav'
                  component={ConversationsNavigation}
                  options={{ headerShown: false }}
                />
              </>
            )}
          </Stack.Group>
          <Stack.Group screenOptions={{ presentation: "modal" }}>
            <Stack.Screen
              name='UserDetailsModal'
              component={UserDetailsModal}
              options={{ headerShown: false }}
            />
          </Stack.Group>
        </Stack.Navigator>
      </NavigationContainer>
    </UserListContext.Provider>
  </Provider>
</SafeAreaProvider>
```

```
src > surfaces > ⚡ UserDetailsModal.js > ⚡ UserDetailsModal
1 import React from "react";
2 import { View, Text, Pressable, Image } from "react-native";
3 import { SafeAreaView } from "react-native-safe-area-context";
4 import Ionicons from "@expo/vector-icons/Ionicons";
5 import { UserDetailsModalImages } from "../components/UserDetailsModalImages";
6
7 export const UserDetailsModal = ({ navigation, route }) => {
8   return (
9     <SafeAreaView style={{ flex: 1, paddingTop: 30 }}>
10    <View
11      style={{
12        width: 650,
13        height: 570,
14        borderRadius: 155,
15        borderWidth: 1,
16        borderColor: "#EEF2E2",
17        position: "absolute",
18        top: -80,
19        left: -360,
20        transform: [{ rotate: "-45deg" }],
21      }}
22    </>
23    <View
24      style={{
25        width: 650,
26        height: 570,
27        borderRadius: 155,
28        borderWidth: 1,
29        borderColor: "#EEF2E2",
30        position: "absolute",
31        top: -80,
32        left: -410,
33        transform: [{ rotate: "-45deg" }],
34      }}
35    </>
36    <View
37      style={{
38        width: 650,
39        height: 570,
40        borderRadius: 155,
41        position: "absolute",
42        top: -80,
43        left: -460,
44        backgroundColor: "#E1F6F4",
45        transform: [{ rotate: "-45deg" }],
46      }}
47    </>
48    <Pressable
49      onPress={() => navigation.goBack()}
50      style={{ flexDirection: "row" }}
51    >
52      <Ionicons name='chevron-back-outline' size={30} color="#000000" />
53      <Text
54        style={{
55          fontFamily: "Poppins_400Regular",
56          fontSize: 18,
```

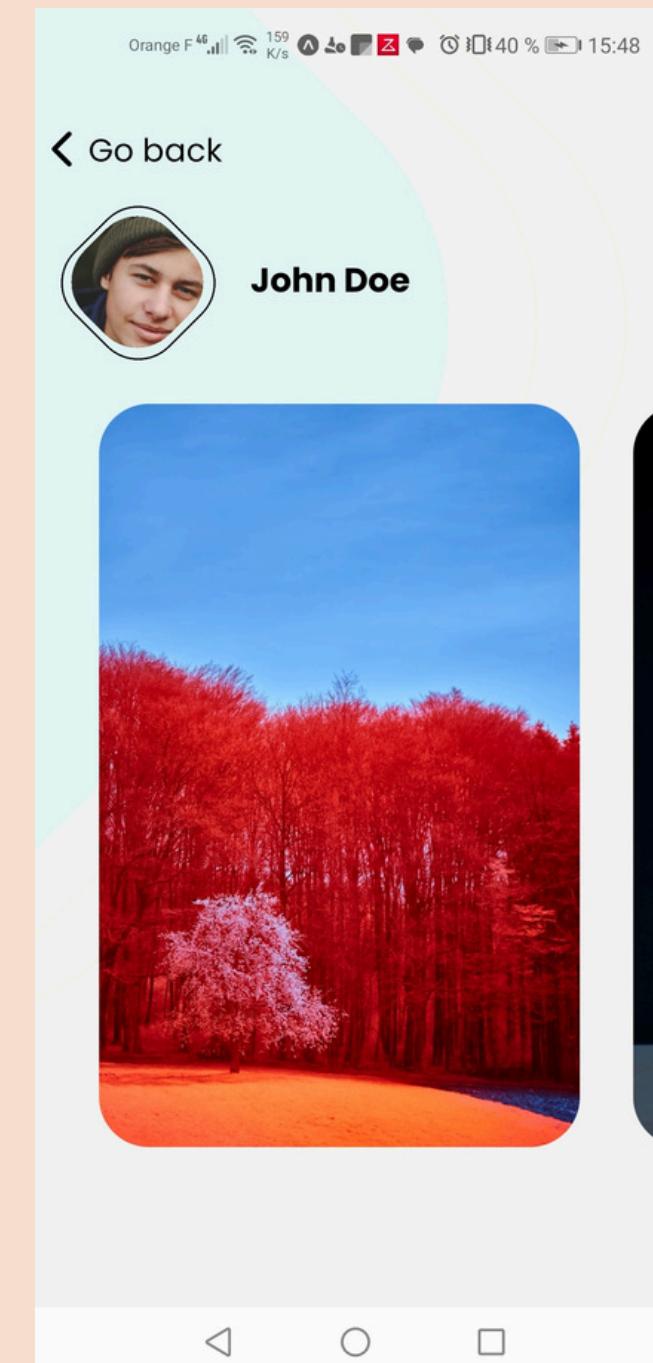
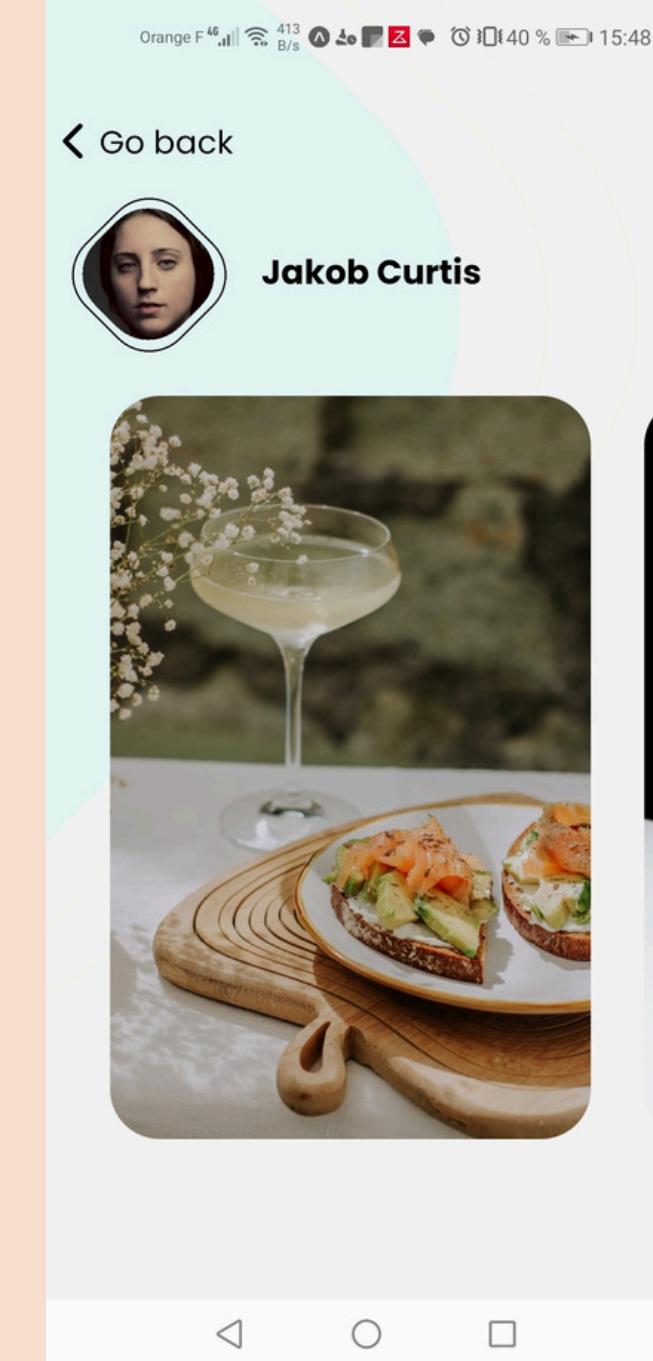
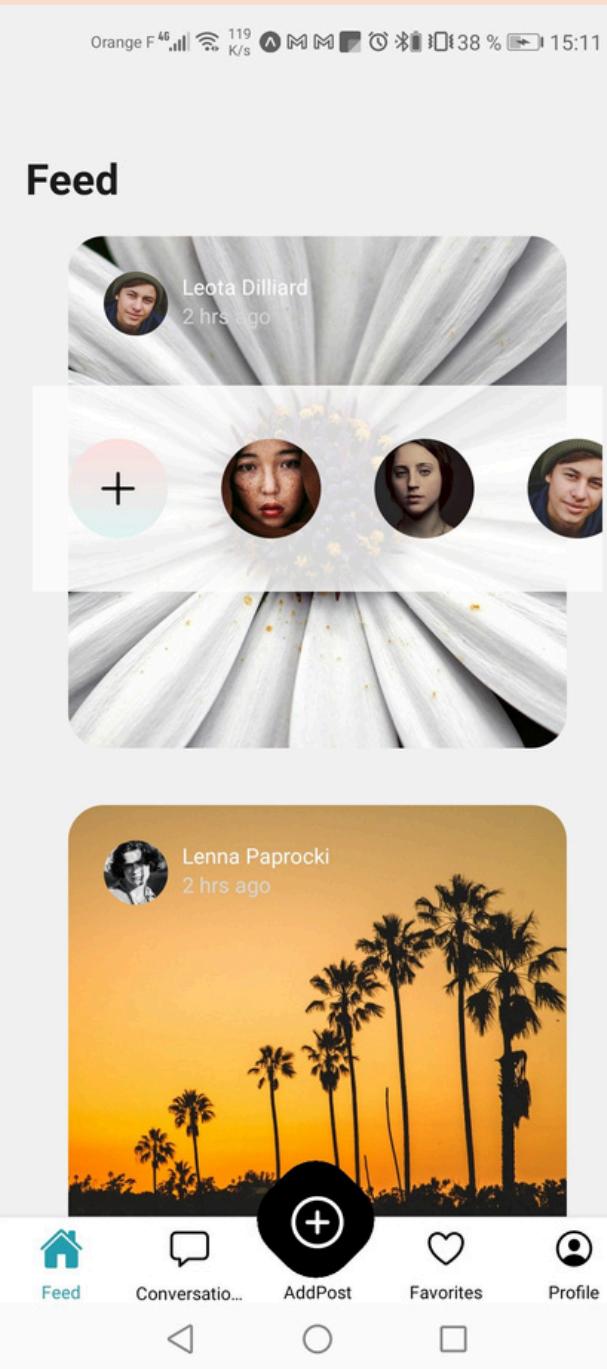
```
src > surfaces > ⚡ UserDetailsModal.js > ⚡ UserDetailsModal
7 export const UserDetailsModal = ({ navigation, route }) => {
5/   marginTop: 3,
58   }
59   >
60   <Text>
61   </Text>
62   </Pressable>
63   <View style={{ paddingTop: 20, marginLeft: 20, flexDirection: "row" }}>
64     <View
65       style={{
66         width: 76,
67         height: 76,
68         borderRadius: 25,
69         border: 1px solid "#000000",
70         transform: [{ rotate: "-45deg" }],
71         alignSelf: "flex-start",
72       }}
73     </>
74     <View
75       style={{
76         overflow: "hidden",
77         alignSelf: "flex-start",
78         transform: [{ rotate: "-45deg" }],
79         borderRadius: 25,
80         height: 67,
81         width: 67,
82         marginLeft: -72,
83         marginTop: 4,
84       }}
85     >
86     <Image
87       style={{
88         height: 95,
89         width: 95,
90         transform: [{ rotate: "45deg" }],
91         alignSelf: "center",
92       }}
93     <source={{ uri: route.params.user.url, }}>
94   </Image>
95   <Text
96     style={{
97       fontFamily: "Poppins_700Bold",
98       fontSize: 19,
99       alignSelf: "center",
100      marginLeft: 30,
101    }}
102    >
103      {route.params.user.name}
104    </Text>
105  </View>
106  <UserDetailsModalImages />
107 </SafeAreaView>
108 .
109 .
110 .
111 .
112 .
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : COMBINAISON DE REDUCERS - TOOLKIT REDUX : USERDETAILSMODAL

```
src > components > JS UserDetailsModalImages.js > UserDetailsModalImages
1 import React, { useState, useEffect } from "react";
2 import { FlatList, Image, View, Text } from "react-native";
3 import { requestBase } from "../utils/constants";
4 import { useWindowDimensions } from "react-native";
5 import Loading from "./Loading";
6
7 export const UserDetailsModalImages = () => {
8   const randomImageSet = Math.round(Math.random() * 4);
9   const [imageList, setImageList] = useState(null);
10  const { height, width } = useWindowDimensions();
11
12  useEffect(() => {
13    const fetchImageData() {
14      const response = await fetch(
15        requestBase + "/userImages/" + randomImageSet + ".json"
16      );
17      setImageList(await response.json());
18    }
19  }, []);
20
21  if (!imageList) {
22    return 
23  }
24  const renderItem = ({ item }) => {
25    return (
26      <Image
27        style={{
28          height: height * 0.6,
29          width: width * 0.75,
30          borderRadius: 28,
31          marginRight: 30,
32        }}
33        source={{
34          uri: item.url,
35        }}
36      />
37    );
38  };
39
40  return (
41    <View style={{ paddingTop: 30 }}>
42      <FlatList
43        data={imageList.listOfItems}
44        renderItem={renderItem}
45        keyExtractor={(item) => item.itemId}
46        horizontal
47        showsHorizontalScrollIndicator={false}
48        snapToInterval={width * 0.825}
49        decelerationRate='fast'
50        ListHeaderComponent=<View style={{ width: width * 0.1 }} />
51      />
52    </View>
53  );
54}
55
56};
```



# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

### IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : COMBINAISON DE REDUCERS - TOOLKIT REDUX : IMAGEDETAILSMODAL

```
src > App.js > ...
21 | import { Provider } from "react-redux";
22 | import { UserDetailsModal } from './src/surfaces/UserDetailsModal';
23 | import { ImageDetailsModal } from './src/surfaces/ImageDetailsModal';

  src > App.js > ...
28 |   export default function App() {
29 |     component={UserDetailsModal}
30 |     options={{ headerShown: false }}
31 |   />
32 |   <Stack.Screen
33 |     name='ImageDetailsModal'
34 |     component={ImageDetailsModal}
35 |     options={{ headerShown: false }}
36 |   />
37 |   </Stack.Group>
```

```
src > components > ListOfCards.js > ...
7 |   export const ListOfCards = ({navigation}) => {
8 |
9 |     const renderItem = ({ item }) => {
10 |       return <Card item={item} navigation={navigation} />;
11 |     };
12 |
13 |     return (
14 |       <Pressable
15 |         onPress={() =>
16 |           navigation.navigate("ImageDetailsModal", { imageItem: item })
17 |         }
18 |       >
19 |         <Image
20 |           style={{
21 |             width: "100%",
22 |             height: 288,
23 |             borderRadius: 20,
24 |             marginBottom: 32,
25 |           }}
26 |           source={{
27 |             uri: item.url,
28 |           }}
29 |         />
30 |         <View
31 |           style={{
32 |             position: "absolute",
33 |             top: 20,
34 |             left: 20,
35 |             flexDirection: "row",
36 |           }
37 |         >
```

```
src > components > Card.js > ...
5 |   export const Card = ({ item, navigation }) => {
6 |     return (
7 |       <UserListContext.Consumer>
8 |         ({ userList }) => {
9 |           const currentUser = userList.filter(
10 |             (user) => user.id === item.authorId
11 |           );
12 |           return (
13 |             <Pressable
14 |               onPress={() =>
15 |                 navigation.navigate("ImageDetailsModal", { imageItem: item })
16 |               }
17 |             >
18 |               <Image
19 |                 style={{
20 |                   width: "100%",
21 |                   height: 288,
22 |                   borderRadius: 20,
23 |                   marginBottom: 32,
24 |                 }}
25 |                 source={{
26 |                   uri: item.url,
27 |                 }}
28 |               />
29 |               <View
30 |                 style={{
31 |                   position: "absolute",
32 |                   top: 20,
33 |                   left: 20,
34 |                   flexDirection: "row",
35 |                 }
36 |               >
```

```
src > ImageDetailsModal.js > ...
1 | import React, { useEffect, useState } from "react";
2 | import { View, Text, Pressable, Image, StyleSheet } from "react-native";
3 | import { SafeAreaView } from "react-native-safe-area-context";
4 | import Ionicons from "@expo/vector-icons/Ionicons";
5 | import { likeImage, unLikeImage } from "../../reducers/likedImages";
6 | import { useDispatch, useSelector } from "react-redux";
7 |
8 | export const ImageDetailsModal = ({ navigation, route }) => {
9 |   const { likedImages } = useSelector((state) => state.likedImages);
10 |   const [isCurrentImageLiked, setIsCurrentImageLiked] = useState(false);
11 |   const dispatch = useDispatch();
12 |
13 |   useEffect(() => {
14 |     const checkIfLiked =
15 |       likedImages?.filter(
16 |         (favoritedImg) => favoritedImg.itemId === route.params.imageItem
17 |       ).length > 0;
18 |     setIsCurrentImageLiked(checkIfLiked);
19 |   }, [likedImages]);
20 |
21 |   return (
22 |     <SafeAreaView style={{ flex: 1, paddingTop: 30 }}>
23 |       <View
24 |         style={[
25 |           width: 650,
26 |           height: 570,
27 |           borderRadius: 155,
28 |           borderWidth: 1,
29 |           borderColor: "#EEF2E2",
30 |           position: "absolute",
31 |           bottom: -260,
32 |           left: -140,
33 |           transform: [{ rotate: "-45deg" }],
34 |         ]}
35 |       >
36 |         <Text> Go back </Text>
37 |         <Pressable
38 |           onPress={() => navigation.goBack()}
39 |           style={{ flexDirection: "row" }}
40 |         >
41 |           <Ionicons name='chevron-back-outline' size={30} color="#000000" />
42 |           <Text
43 |             style={{
44 |               fontFamily: "Poppins_400Regular",
45 |               fontSize: 18,
46 |               marginTop: 3,
47 |             }}
48 |           >
49 |             <Text> Follows </Text>
50 |             <Text> 128 </Text>
51 |           </Text>
52 |         </Pressable>
53 |         <View style={{ paddingTop: 30 }}>
54 |           <Image
55 |             style={{
56 |               width: "100%",
57 |               height: 288,
58 |               marginBottom: 32,
59 |             }}
60 |             source={{
61 |               uri: route.params.imageItem.url,
62 |             }}
63 |           />
64 |           <View
65 |             style={{
66 |               width: 650,
67 |               height: 570,
68 |               borderRadius: 155,
69 |               borderWidth: 1,
70 |               borderColor: "#EEF2E2",
71 |               position: "absolute",
72 |               bottom: -290,
73 |               left: -140,
74 |               transform: [{ rotate: "-45deg" }],
75 |             }}
76 |           >
77 |             <View style={styles.general}>
78 |               <Text style={styles.headerText}>Likes </Text>
79 |               <Text style={styles.stats}>{route.params.imageItem.likes}</Text>
80 |             </View>
81 |             <View style={styles.general}>
82 |               <Text style={styles.headerText}>Conversations </Text>
83 |               <Text style={styles.stats}>{route.params.imageItem.conversations}</Text>
84 |             </View>
85 |             <View style={styles.general}>
86 |               <Text style={styles.headerText}>Follows </Text>
87 |               <Text style={styles.stats}>128 </Text>
88 |             </View>
89 |           </View>
90 |         </SafeAreaView>
91 |   );
92 |
93 |   const styles = StyleSheet.create({
94 |     general: {
95 |       alignItems: "center",
96 |     },
97 |     headerText: {
98 |       fontSize: 16,
99 |       fontFamily: "Poppins_400Regular",
100 |     },
101 |     stats: {
102 |       fontFamily: "Poppins_700Bold",
103 |       fontSize: 25,
104 |     },
105 |   );
106 | }
```

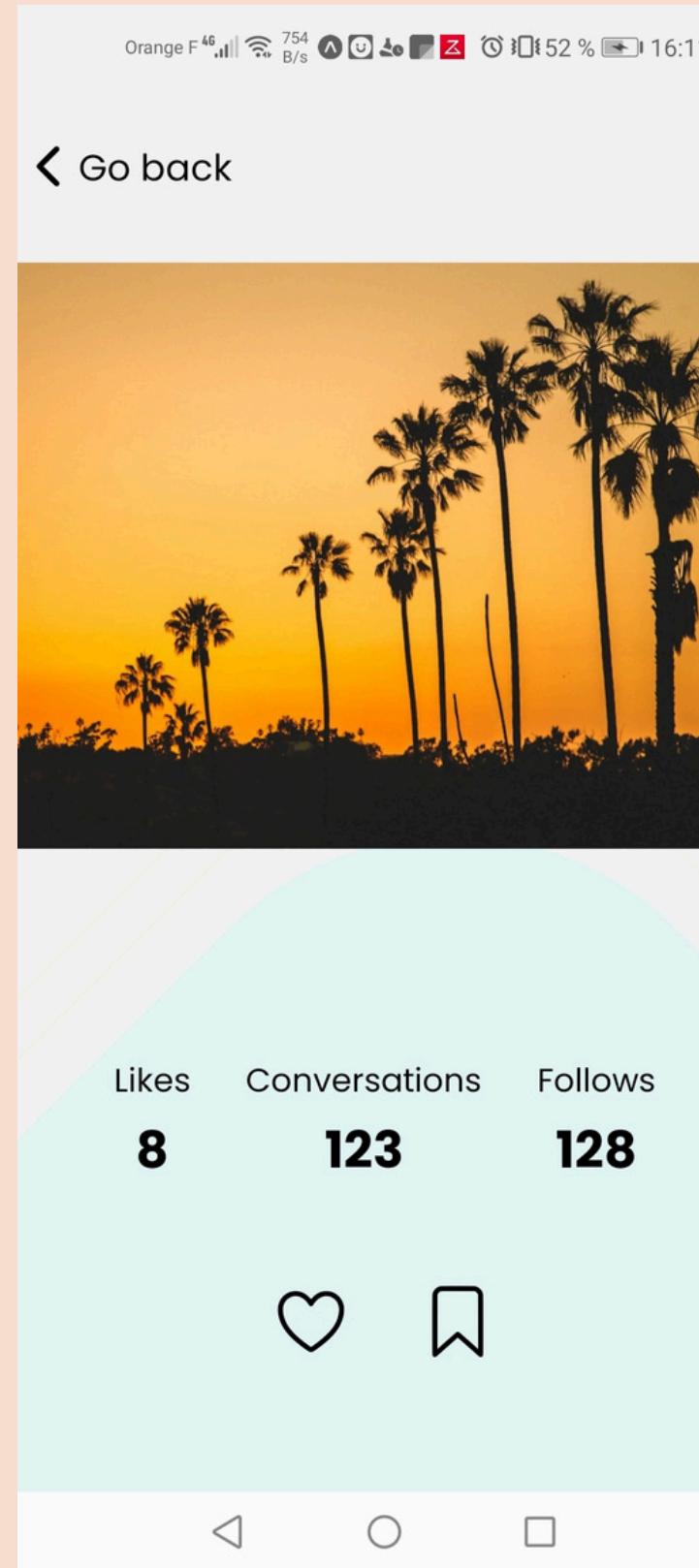
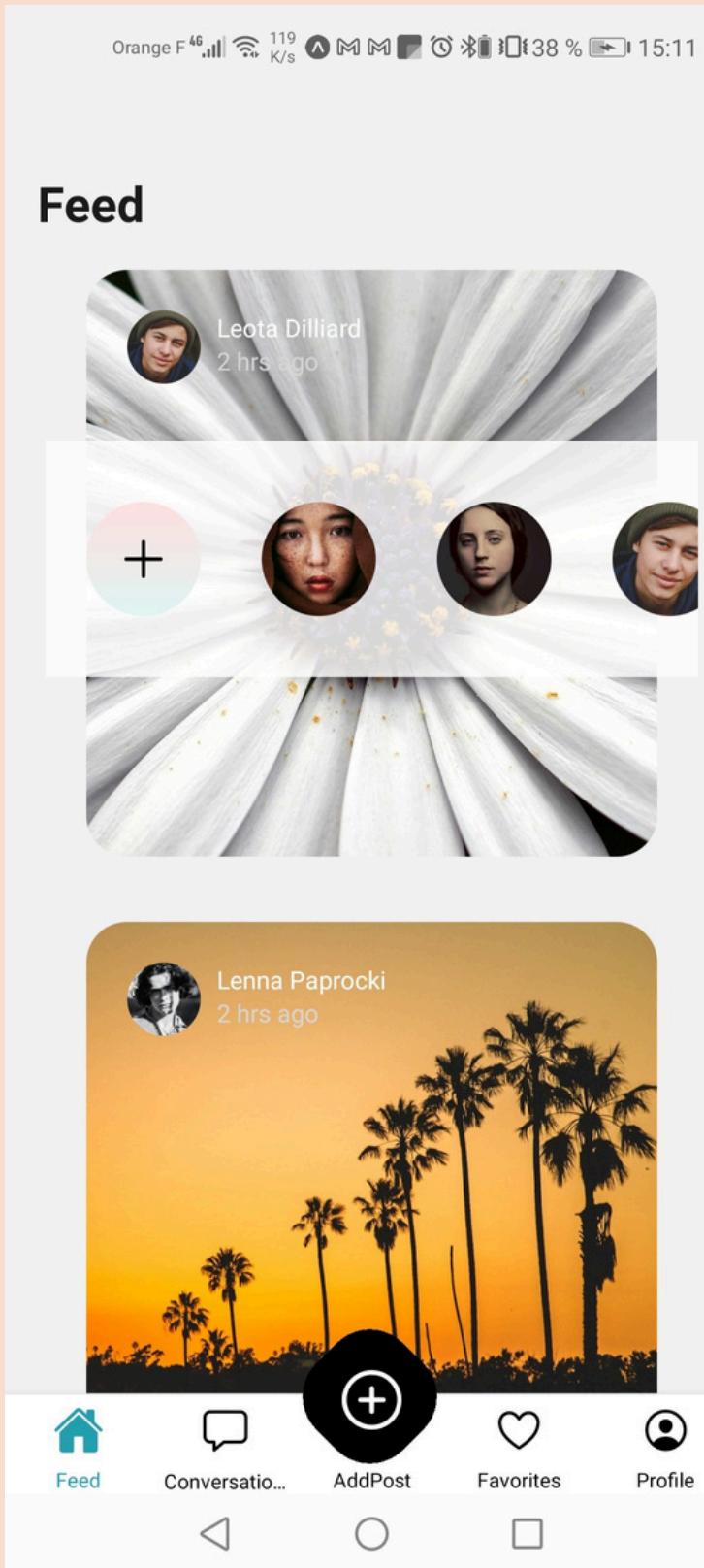
```
src > ImageDetailsModal.js > ...
1 | import React, { useEffect, useState } from "react";
2 | import { View, Text, Pressable, Image, StyleSheet } from "react-native";
3 | import { SafeAreaView } from "react-native-safe-area-context";
4 | import Ionicons from "@expo/vector-icons/Ionicons";
5 | import { likeImage, unLikeImage } from "../../reducers/likedImages";
6 | import { useDispatch, useSelector } from "react-redux";
7 |
8 | export const ImageDetailsModal = ({ navigation, route }) => {
9 |   const { likedImages } = useSelector((state) => state.likedImages);
10 |   const [isCurrentImageLiked, setIsCurrentImageLiked] = useState(false);
11 |   const dispatch = useDispatch();
12 |
13 |   useEffect(() => {
14 |     const checkIfLiked =
15 |       likedImages?.filter(
16 |         (favoritedImg) => favoritedImg.itemId === route.params.imageItem
17 |       ).length > 0;
18 |     setIsCurrentImageLiked(checkIfLiked);
19 |   }, [likedImages]);
20 |
21 |   return (
22 |     <SafeAreaView style={{ flex: 1, paddingTop: 30 }}>
23 |       <View
24 |         style={[
25 |           width: 650,
26 |           height: 570,
27 |           borderRadius: 155,
28 |           borderWidth: 1,
29 |           borderColor: "#EEF2E2",
30 |           position: "absolute",
31 |           bottom: -330,
32 |           left: -140,
33 |           transform: [{ rotate: "-45deg" }],
34 |         ]}
35 |       >
36 |         <Text> Go back </Text>
37 |         <Pressable
38 |           onPress={() => navigation.goBack()}
39 |           style={{ flexDirection: "row" }}
40 |         >
41 |           <Ionicons name='chevron-back-outline' size={30} color="#000000" />
42 |           <Text
43 |             style={{
44 |               fontFamily: "Poppins_400Regular",
45 |               fontSize: 18,
46 |               marginTop: 3,
47 |             }}
48 |           >
49 |             <Text> Follows </Text>
50 |             <Text> 128 </Text>
51 |           </Text>
52 |         </Pressable>
53 |         <View style={{ paddingTop: 30 }}>
54 |           <Image
55 |             style={{
56 |               width: "100%",
57 |               height: 288,
58 |               marginBottom: 32,
59 |             }}
60 |             source={{
61 |               uri: route.params.imageItem.url,
62 |             }}
63 |           />
64 |           <View
65 |             style={{
66 |               width: 650,
67 |               height: 570,
68 |               borderRadius: 155,
69 |               borderWidth: 1,
70 |               borderColor: "#EEF2E2",
71 |               position: "absolute",
72 |               bottom: -260,
73 |               left: -140,
74 |               transform: [{ rotate: "-45deg" }],
75 |             }}
76 |           >
77 |             <View style={styles.general}>
78 |               <Text style={styles.headerText}>Likes </Text>
79 |               <Text style={styles.stats}>{route.params.imageItem.likes}</Text>
80 |             </View>
81 |             <View style={styles.general}>
82 |               <Text style={styles.headerText}>Conversations </Text>
83 |               <Text style={styles.stats}>{route.params.imageItem.conversations}</Text>
84 |             </View>
85 |             <View style={styles.general}>
86 |               <Text style={styles.headerText}>Follows </Text>
87 |               <Text style={styles.stats}>128 </Text>
88 |             </View>
89 |           </View>
90 |         </SafeAreaView>
91 |   );
92 |
93 |   const styles = StyleSheet.create({
94 |     general: {
95 |       alignItems: "center",
96 |     },
97 |     headerText: {
98 |       fontSize: 16,
99 |       fontFamily: "Poppins_400Regular",
100 |     },
101 |     stats: {
102 |       fontFamily: "Poppins_700Bold",
103 |       fontSize: 25,
104 |     },
105 |   );
106 | }
```

```
src > ImageDetailsModal.js > ...
1 | import React, { useEffect, useState } from "react";
2 | import { View, Text, Pressable, Image, StyleSheet } from "react-native";
3 | import { SafeAreaView } from "react-native-safe-area-context";
4 | import Ionicons from "@expo/vector-icons/Ionicons";
5 | import { likeImage, unLikeImage } from "../../reducers/likedImages";
6 | import { useDispatch, useSelector } from "react-redux";
7 |
8 | export const ImageDetailsModal = ({ navigation, route }) => {
9 |   const { likedImages } = useSelector((state) => state.likedImages);
10 |   const [isCurrentImageLiked, setIsCurrentImageLiked] = useState(false);
11 |   const dispatch = useDispatch();
12 |
13 |   useEffect(() => {
14 |     const checkIfLiked =
15 |       likedImages?.filter(
16 |         (favoritedImg) => favoritedImg.itemId === route.params.imageItem
17 |       ).length > 0;
18 |     setIsCurrentImageLiked(checkIfLiked);
19 |   }, [likedImages]);
20 |
21 |   return (
22 |     <SafeAreaView style={{ flex: 1, paddingTop: 30 }}>
23 |       <View
24 |         style={[
25 |           width: 650,
26 |           height: 570,
27 |           borderRadius: 155,
28 |           borderWidth: 1,
29 |           borderColor: "#EEF2E2",
30 |           position: "absolute",
31 |           bottom: -260,
32 |           left: -140,
33 |           transform: [{ rotate: "-45deg" }],
34 |         ]}
35 |       >
36 |         <Text> Go back </Text>
37 |         <Pressable
38 |           onPress={() => navigation.goBack()}
39 |           style={{ flexDirection: "row" }}
40 |         >
41 |           <Ionicons name='chevron-back-outline' size={30} color="#000000" />
42 |           <Text
43 |             style={{
44 |               fontFamily: "Poppins_400Regular",
45 |               fontSize: 18,
46 |               marginTop: 3,
47 |             }}
48 |           >
49 |             <Text> Follows </Text>
50 |             <Text> 128 </Text>
51 |           </Text>
52 |         </Pressable>
53 |         <View style={{ paddingTop: 30 }}>
54 |           <Image
55 |             style={{
56 |               width: "100%",
57 |               height: 288,
58 |               marginBottom: 32,
59 |             }}
60 |             source={{
61 |               uri: route.params.imageItem.url,
62 |             }}
63 |           />
64 |           <View
65 |             style={{
66 |               width: 650,
67 |               height: 570,
68 |               borderRadius: 155,
69 |               borderWidth: 1,
70 |               borderColor: "#EEF2E2",
71 |               position: "absolute",
72 |               bottom: -290,
73 |               left: -140,
74 |               transform: [{ rotate: "-45deg" }],
75 |             }}
76 |           >
77 |             <View style={styles.general}>
78 |               <Text style={styles.headerText}>Likes </Text>
79 |               <Text style={styles.stats}>{route.params.imageItem.likes}</Text>
80 |             </View>
81 |             <View style={styles.general}>
82 |               <Text style={styles.headerText}>Conversations </Text>
83 |               <Text style={styles.stats}>{route.params.imageItem.conversations}</Text>
84 |             </View>
85 |             <View style={styles.general}>
86 |               <Text style={styles.headerText}>Follows </Text>
87 |               <Text style={styles.stats}>128 </Text>
88 |             </View>
89 |           </View>
90 |         </SafeAreaView>
91 |   );
92 |
93 |   const styles = StyleSheet.create({
94 |     general: {
95 |       alignItems: "center",
96 |     },
97 |     headerText: {
98 |       fontSize: 16,
99 |       fontFamily: "Poppins_400Regular",
100 |     },
101 |     stats: {
102 |       fontFamily: "Poppins_700Bold",
103 |       fontSize: 25,
104 |     },
105 |   );
106 | }
```

# LA GESTION DES ETATS - REACT NATIVE

## LE PROJET

IMPLÉMENTATION DE REDUX DANS NOTRE APPLICATION : COMBINAISON DE REDUCERS - TOOLKIT REDUX : IMAGEDETAILSMODAL



MERCI POUR L'ÉCOUTE  
&  
LA PARTICIPATION