

# PROPS - NAVIGATION

<https://reactnavigation.org/docs/getting-started/>

# Tái sử dụng component

- File AppBar.js:

```
export default function AppBar() {  
  return (  
    <View style={styles.header}>  
      <TouchableOpacity>  
        <Image style={styles.header__icon} source={require('../assets/asset_contact/menu.png')} />  
      </TouchableOpacity>  
      <Text style={styles.header__title}>Contact</Text>  
      <TouchableOpacity>  
        <Image style={styles.header__icon} source={require('../assets/asset_contact/search.png')} />  
      </TouchableOpacity>  
    </View>  
  )  
}  
  
const styles = StyleSheet.create({ .....  
  header: {  
    padding: 10,  
    backgroundColor: '#f0f0f0',  
    marginBottom: 10,  
  },  
  header__icon: {  
    width: 30,  
    height: 30,  
    margin: 5,  
  },  
  header__title: {  
    color: 'black',  
    margin: 5,  
  },  
});
```

# Tái sử dụng component

- File DemoScrollView.js:

```
import { View, Text, TouchableOpacity, Image, StyleSheet } from 'react-native'  
import React from 'react'  
import AppBar from './AppBar'
```

```
export default function DemoScrollView() {  
  
  return (  
    <View style={styles.container}>  
      <AppBar/>  
    </View>  
  )  
}  
  
const styles = StyleSheet.create({
```

# Props

- Props thực chất là viết tắt của Properties
- Props của component chỉ nhận các giá trị được truyền vào và không thể bị chỉnh sửa
- Ví dụ với component Image có thuộc tính source
- Như ví dụ trên: AppBar là 1 custom component. Ta có thể tạo thêm props cho nó bằng cách khai báo 1 tham số (kiểu object, thường có tên là props):

```
export default function AppBar({props}) {
```
- Sau đó có thể quy định thêm các key cho tham số này như:

```
<Text style={styles.header__title}>{props.title}</Text>
```
- Khi sử dụng thì truyền giá trị cho thuộc tính:

```
<AppBar title='New Feed' />
```

# Props

```
import { View, Text, TouchableOpacity, Image, StyleSheet } from 'react-native'
import React from 'react'

export default function AppBar(props) {
  return (
    <View style={styles.header}>
      <TouchableOpacity>
        <Image style={styles.header__icon} source={props.lefticon} />

        </TouchableOpacity>
        <Text style={styles.header__title}>{props.title}</Text>
        <TouchableOpacity>
          <Image style={styles.header__icon} source={props.righticon} />
        </TouchableOpacity>
      </View>
    )
  }
```

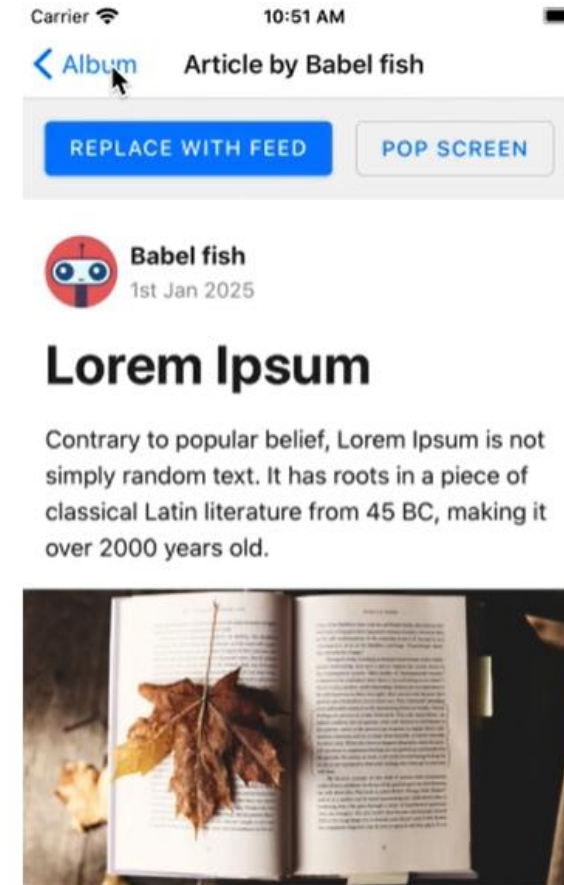
# Props

```
import { View, Text } from 'react-native'
import React from 'react'
import AppBar from './AppBar'

export default function UsingAppBar1() {
  return (
    <View>
      <AppBar title='App Food'
        lefticon = {require('../assets/Emoji_Icons/angry.png')}
        righticon = {require('../assets/Emoji_Icons/haha.png')}
      ></AppBar>
      <Text>UsingAppBar1</Text>
    </View>
  )
}
```

# Điều hướng (Navigate)

- Để điều hướng: sử dụng thư viện react-navigation
- Có 3 dạng navigate:
  1. Stack Navigator: chuyển tiếp giữa các màn hình theo cách tổ chức của stack (đặt 1 màn hình mới lên top của stack)

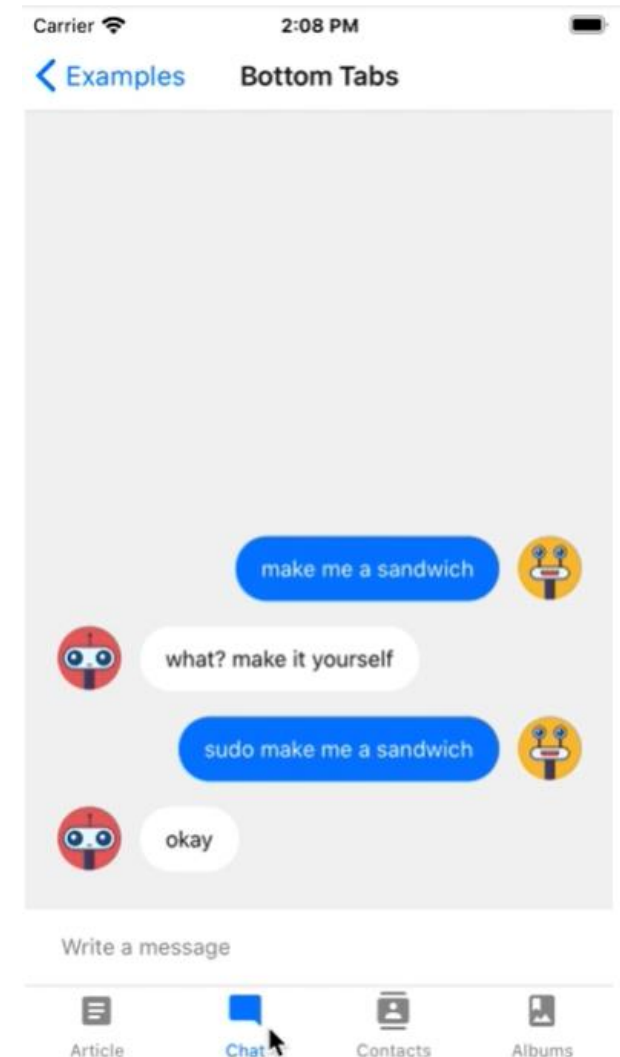


Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked

# Điều hướng (Navigate)

## 2. Bottom Tabs

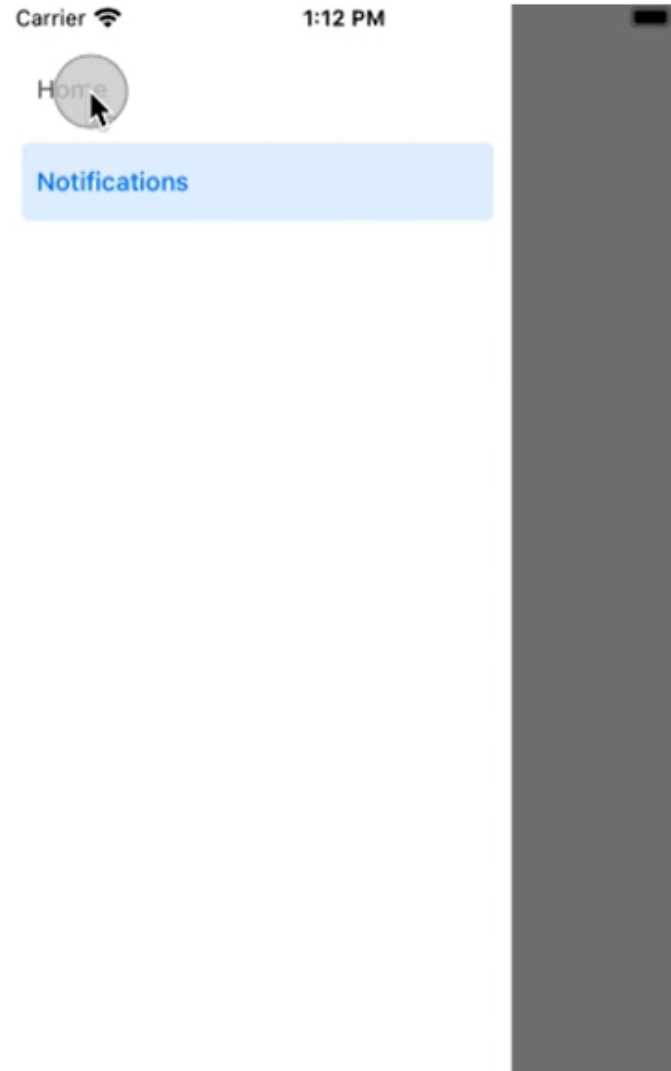
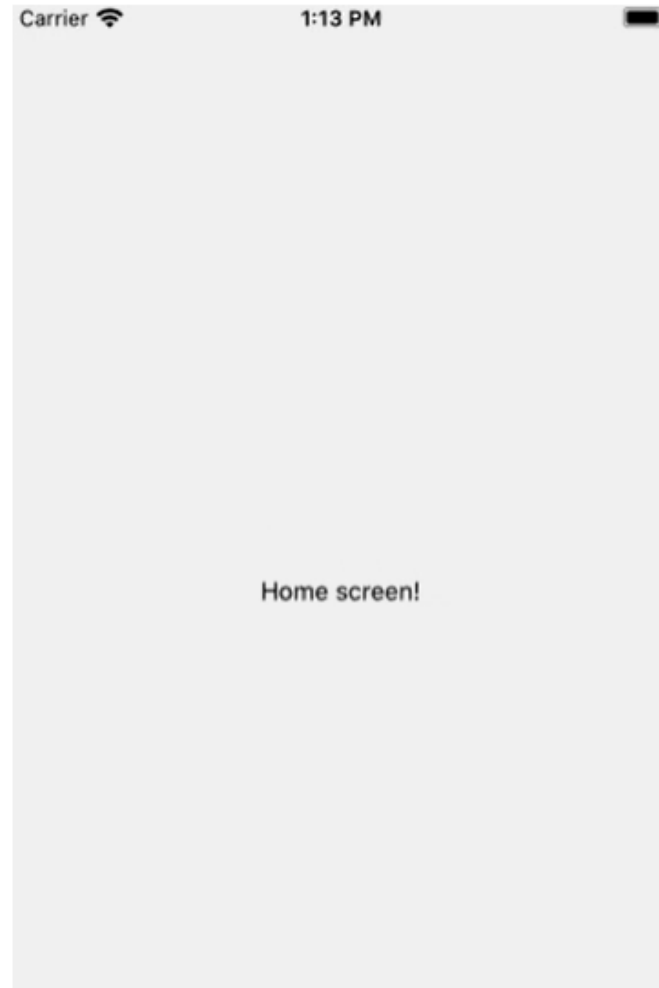
Navigator: thanh tab bar ở cuối màn hình





# Điều hướng (Navigate)

3. Drawer Navigator:  
thanh menu bên trái



# Cài đặt react-navigation

- Cài đặt:

```
npm install @react-navigation/native
```

- Đối với Expo, cài thêm:

```
npx expo install react-native-screens react-native-safe-area-context
```

- Import:

```
import { NavigationContainer } from '@react-navigation/native'
```

- Sử dụng `<NavigationContainer>` component để bao toàn bộ app và đăng ký tất cả các màn hình cần điều hướng. Mỗi app chỉ có duy nhất 1 `<NavigationContainer>`

```
export default function App() {  
  return (  
    <NavigationContainer>  
      {/* Đăng ký tất cả các màn hình cần điều hướng */}  
    </NavigationContainer>  
  );  
}
```

# Stack Navigation

- Bước 1: Cài đặt thư viện :

```
npm install @react-navigation/native-stack
```

- Bước 2: Import thư viện:

```
import { createNativeStackNavigator } from '@react-navigation/native-stack'
```

- Bước 3: Sử dụng <NavigationContainer>

- Bước 4: Tạo đối tượng StackNavigator: bên ngoài hàm render, nhập:

```
const Stack = createNativeStackNavigator()
```

- Hàm createNativeStackNavigator() trả về 1 đối tượng có 2 thuộc tính:

- Screen: dùng để khai báo các màn hình. Screen có 2 thuộc tính: name: tên, component: tên file của page
- Navigator: chứa tất cả các Screen

- Bước 5: Khai báo Navigator và Screen

- Chú ý: Screen đăng ký đầu tiên đóng vai trò như Homepage

# Stack Navigation

- Ví dụ:

```
import { View, Text } from 'react-native'
import React from 'react'
import { NavigationContainer } from '@react-navigation/native'
import Home from './screens/Home'
import Detail from './screens/Detail'
const Stack = createNativeStackNavigator()

export default function App() {
  return (
    <NavigationContainer>
      <Stack.Navigator>
        <Stack.Screen name='Home' component={Home} options={{title:'Home Page'}}/>
        <Stack.Screen name='Detail' component={Detail} />
      </Stack.Navigator>
    </NavigationContainer>
  )
}
```

# Stack Navigation

- Chuyển trang: Sử dụng props navigation. Có 2 cách:
- Cách 1: Gọi hàm navigate của navigation

```
export default function Home({ navigation }) {  
  return (  
    <View style={styles.container}>  
      <TouchableOpacity style={styles.button}  
        onPress={() => { navigation.navigate('Detail') }}>  
        <Text style={styles.button__text}>Home</Text>  
      </TouchableOpacity>  
    </View>  
  )  
}
```

# Stack Navigation

- Cách 2: Gọi hàm push của navigation

```
export default function Home({ navigation }) {  
  return (  
    <View style={styles.container}>  
      <TouchableOpacity style={styles.button}  
        onPress={() => { navigation.push('Detail') }}>  
        <Text style={styles.button__text}>Home</Text>  
      </TouchableOpacity>  
    </View>  
  )  
}
```

# Stack Navigation

- Truyền dữ liệu (tham số) khi điều hướng:
  - Ở HomePage: gửi trực tiếp giá trị cần truyền vào phương thức navigate hoặc push:  
`onPress={() => { navigation.push('Detail', "I'm from HomePage") }}`
  - Ở DetailPage: sử dụng thuộc tính params của props route để nhận giá trị  
`export default function Detail({route}) {  
 return (  
 <View>  
 <Text>Giá trị từ HomePage: {route.params}</Text>  
 </View>  
 )  
}`

# Stack Navigation

- Truyền dữ liệu (tham số) khi điều hướng:
  - Nếu truyền nhiều tham số: **sử dụng object**
  - Ở HomePage:  
onPress={() => { navigation.push('Detail', {user:'Admin', pass:'12345'}) }}>
  - Ở DetailPage:  
<Text>UserName: {route.params.user}, Password: {route.params.pass}  
</Text>



# Tab Navigation

- Bước 1: Cài đặt thư viện:

```
npm install @react-navigation/bottom-tabs
```

- Bước 2: Import thư viện:

```
import { createBottomTabNavigator } from '@react-navigation/bottom-tabs'
```

- Bước 3: Tạo đối tượng TabNavigator: bên ngoài hàm render, nhập:

```
const Tab = createBottomTabNavigator()
```

- Khai báo Navigator và Screen tương tự như Stack. Mỗi Screen là 1 tab

```
<NavigationContainer>  
  <Tab.Navigator>  
    <Tab.Screen name='Home' component={Home} options={{title:'Home Page'}}  
  />  
    <Tab.Screen name='Detail' component={Detail} />  
  </Tab.Navigator>  
</NavigationContainer>
```

# Tab Navigation

- Thiết lập icon cho tab:

<https://reactnavigation.org/docs/tab-based-navigation>

# Drawer Navigation

- Bước 1: Cài đặt thư viện:  
npm install @react-navigation/drawer
- Với Expo cài thêm:  
npx expo install react-native-gesture-handler react-native-reanimated
- Trong file index.js hoặc App.js: đặt dòng sau vào dòng đầu tiên  
import 'react-native-gesture-handler'
- Cấu hình: Mở file babel.config.js, thêm đoạn sau vào:  
plugins: [  
 'react-native-reanimated/plugin', // This line  
,

```
1 module.exports = function(api) {  
2   api.cache(true);  
3   return {  
4     presets: ['babel-preset-expo'],  
5     plugins: [  
6       'react-native-reanimated/plugin', // This line.  
7     ],  
8   };  
9 };  
10
```

# Drawer Navigation

- File babel.config.js được như sau:

```
1  module.exports = function(api) {  
2    ...  
3    api.cache(true);  
4    return {  
5      presets: ['babel-preset-expo'],  
6      plugins: [  
7        'react-native-reanimated/plugin', // This line.  
8      ],  
9    };  
10 }
```

- Chạy lệnh sau (đối với Expo): expo start -c

# Drawer Navigation

- Khai báo Navigator và Screen tương tự như Stack.

```
<NavigationContainer>  
  <Drawer.Navigator>  
    <Drawer.Screen name='Home' component={Home} options={{title:'Home  
Page'}} />  
    <Drawer.Screen name='Detail' component={Detail} />  
  </Drawer.Navigator>  
</NavigationContainer>
```