**BÀI TẬP 01: CÀI ĐẶT MÔI TRƯỜNG**

**Họ tên:** Nguyễn Minh Tú

**MSSV:** 20110741

1. **Khởi tạo project**

Để khởi tạo project mới ta thực hiện câu lệnh sau:

npx create-expo-app <ten-project>

*Ví dụ:*

npx create-expo-app BT01\_CaiDatMoiTruong

Sau khi quá trình khởi tạo hoàn tất thì ta tiến hành dùng lệnh cd để đổi sang thư mục của Project và chạy project với câu lệnh npm start.

*Ví dụ:*

A screenshot of a computer

Description automatically generated

1. **Cài đặt thư viên React Navigation**

Để có thể thực hiện chuyển giữa hai màn hình ta cần cài đặt thư viện React Navigation.

Docs: <https://reactnavigation.org/docs/7.x/getting-started>

Câu lệnh cài đặt:

npm install @react-navigation/native@next

Ngoài ra khi dùng với Expo ta cần cài đặt thêm thư viện sau:

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

1. **Tạo 2 trang Intro và Home**

Thực hiện khởi tạo thư mục **components** (nơi chứa mọi screen và component cần dùng cho app). Tiếp theo thực hiện khởi tạo 2 thực mục con là **Intro** và **Home**. Mỗi thư mục chứa file **index.js** (là nơi chứa code hiển thị cho mỗi màn hình).

A screenshot of a computer

Description automatically generated

Code trang index.js (thư mục **Home**):

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

import React from "react";

const Home = () => {

  return (

    <View style={styles.container}>

      <Text>This is home screen</Text>

    </View>

  );

};

export default Home;

const styles = StyleSheet.create({

  container: {

    flex: 1,

    justifyContent: "center",

    alignItems: "center",

  },

});

Code trang index.js (thư mục Intro):

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

import React, { useEffect } from "react";

const Intro = () => {

  return (

    <View style={styles.container}>

      <Text>Họ tên: Nguyễn Minh Tú</Text>

      <Text>MSSV: 20110741</Text>

    </View>

  );

};

export default Intro;

const styles = StyleSheet.create({

  container: {

    flex: 1,

    justifyContent: "center",

    alignItems: "center",

  },

});

1. **Thực hiện khởi tạo Stack Navigation**

Khởi tạo Stack Navigation để có thể thực hiện điều hướng giữa 2 màn hình Intro và Home:

Docs: <https://reactnavigation.org/docs/7.x/hello-react-navigation>

Cài đặt bằng câu lệnh:

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

Sau đó thực hiện thay đổi code trang App.js

import { NavigationContainer } from "@react-navigation/native";

import { createNativeStackNavigator } from "@react-navigation/native-stack";

import { StatusBar } from "expo-status-bar";

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

import Home from "./components/Home";

import Intro from "./components/Intro";

const Stack = createNativeStackNavigator();

export default function App() {

  return (

    <View style={{ flex: 1 }}>

      <NavigationContainer>

        <Stack.Navigator

          initialRouteName="Intro"

          screenOptions={{

            headerShown: false,

          }}

        >

          <Stack.Screen name="Home" component={Home} />

          <Stack.Screen name="Intro" component={Intro} />

        </Stack.Navigator>

      </NavigationContainer>

    </View>

  );

}

const styles = StyleSheet.create({

  container: {

    flex: 1,

    backgroundColor: "red",

    alignItems: "center",

    justifyContent: "center",

  },

});

Trong đoạn code trên ta đã định nghĩa một Stack Navigator với 2 Screen là **Home** và **Intro**. Trong đó Intro sẽ được hiển thị đầu tiên mỗi khi khởi chạy ứng dụng bằng cách chỉ định thông qua prop initialRouteName. Ngoài ra để ẩn header của Stack Navigatior ta dùng thuộc tính screenOptions={{headerShown: false }}

1. **Chỉnh sửa code trang Intro để chuyển sang trang Home tự động**

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

import React, { useEffect } from "react";

import { useNavigation } from "@react-navigation/native";

const Intro = () => {

  const navigation = useNavigation();

  useEffect(() => {

    const id = setTimeout(() => {

      navigation.reset({

        index: 0,

        routes: [{ name: "Home" }],

      });

// navigation.navigate("Home");

    }, 10000);

    return () => {

      clearTimeout(id);

    };

  }, []);

  return (

    <View style={styles.container}>

      <Text>Họ tên: Nguyễn Minh Tú</Text>

      <Text>MSSV: 20110741</Text>

    </View>

  );

};

export default Intro;

const styles = StyleSheet.create({

  container: {

    flex: 1,

    justifyContent: "center",

    alignItems: "center",

  },

});

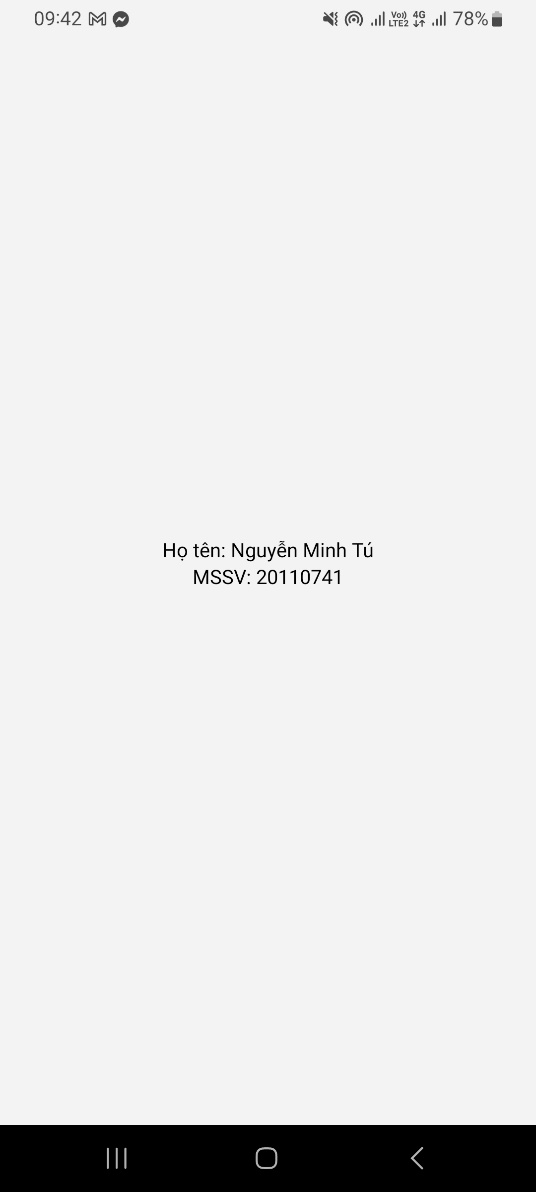
Để chuyển sang một màn hình Home ta cần khởi tạo một đối tượng **navigation** với hook **useNavigation.**

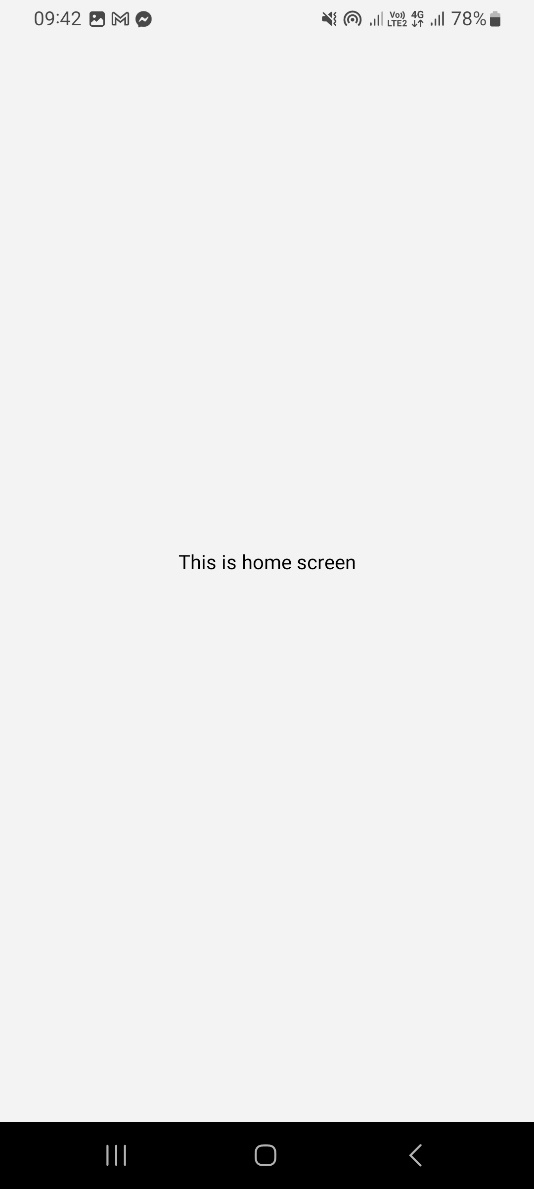
Về cơ bản chuyển sang màn hình khác ta sẽ dùng câu lệnh:

navigation.navigate(“Home”)

Tuy nhiên thông thường các trang **Intro** thường sẽ bị thay thế bởi trang **Home** và không thể quay về trang Intro nên ta cần dùng function reset để thiết lập một route mới hoàn toàn. Và để thực hiện chuyển sang trang **Home** tự động ta cần dùng hàm setTimeOut và clear nó bằng *clean up function* của **useEffect**.

1. **Kết quả:**

****

****