Design a React Native application that serves as a BMI (Body Mass Index) calculator. The app should include a background image, stack navigation for multiple screens, and a modal or alert box to display the calculated BMI result.

* Describe the initial steps to set up a new React Native project for the BMI calculator. Explain how to configure stack navigation to accommodate multiple screens.
* Outline the components and layout required for the app's user interface, including input fields for height and weight. Discuss how to design a visually appealing input form for capturing user data.
* Describe the logic for calculating the BMI based on user input (BMI formula: weight (kg) / (height (m) \* height (m))). Explain how to display the calculated BMI result on a separate screen or in a modal/alert box.
* Explain how to set a background image for the React Native application. Describe styling considerations to ensure the background image fits the screen and complements the overall design.

Step - 1: Run npx create-expo-app bmiApp

Step-2: Run cd bmiApp

Step-3: Run npm install @react-navigation/native @react-navigation/native-stack in the terminal

Step-4: Since it is an expo managed project, run npx expo install react-native-screens react-native-safe-area-context in the terminal

**App.js**

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

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

import BMIForm from './screens/BMIForm';

import Result from './screens/Result';

const Stack = createNativeStackNavigator();

export default function App() {

return (

<NavigationContainer>

<Stack.Navigator initialRouteName="SForm">

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

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

</Stack.Navigator>

</NavigationContainer>

);

}

**BMIForm.js**

import { useState } from 'react';

import { View, StyleSheet, TextInput, Button, ImageBackground} from 'react-native';

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

const image = { uri: 'https://cdn.pixabay.com/photo/2023/03/21/15/11/ai-generated-7867492\_1280.jpg' };

export default function BMIForm() {

const navigation = useNavigation();

const [weight, setWeight] = useState('');

const [height, setHeight] = useState('');

const handleSubmit = () => {

// Navigate to the Result and pass the form data as params

navigation.navigate('Result', { weight, height });

};

return (

<View style={styles.container}>

<ImageBackground source={image} style={styles.image} imageStyle={{ opacity: 0.4 }}>

<View style={styles.subcontainer}>

<TextInput

value={weight}

onChangeText={(text) => setWeight(text)}

placeholder="Enter weight"

style = {styles.input}

/>

<TextInput

value={height}

onChangeText={(text) => setHeight(text)}

placeholder="Enter height"

style = {styles.input}

/>

<Button

title="Submit"

onPress={handleSubmit}

style = {styles.btn}

/>

</View>

</ImageBackground>

</View>

);

}

const styles = StyleSheet.create({

container: {

flex: 1,

backgroundColor: '#000',

alignItems: 'center',

justifyContent: 'center',

},

subcontainer:{

backgroundColor: 'white',

padding: 30,

borderRadius: 5,

},

image: {

flex: 1,

justifyContent: 'center',

resizeMode: 'cover',

alignSelf: 'stretch',

alignItems: 'center',

},

input: {

padding: 20,

width: 250,

}

});

**Result.js**

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

import { useRoute } from '@react-navigation/native';

export default function Result() {

const route = useRoute();

const { weight, height } = route.params;

// Calculate BMI

const weightKg = parseFloat(weight); // Convert weight to a numeric value (assuming it's in kilograms)

const heightMeters = parseFloat(height) / 100; // Convert height to meters

const bmi = (weightKg / (heightMeters \* heightMeters)).toFixed(2);

return (

<View style={styles.container}>

<Text style={styles.display}>Your BMI is: {bmi} </Text>

</View>

);

}

const styles = StyleSheet.create({

container: {

flex: 1,

backgroundColor: '#000',

alignItems: 'center',

justifyContent: 'center',

},

display: {

color: 'white',

}

});