

React Native

React Native Styling



React Native Styling

- ☐ Layout with FlexBox
- ☐ Navigate Screen

Objektif sesi

- Memahami dan mengimplementasikan Flexbox
- Memahami dan mengimplementasikan Screen Navigation

Flexbox

Flexbox merupakan model layout yang kuat dan responsif yang memungkinkan pengembang untuk merancang layout yang dinamis dan mudah disesuaikan.

- Flexbox secara otomatis menangani distribusi ruang di dalam kontainer, membuatnya mudah untuk mengatur elemen-elemen dalam baris atau kolom sesuai dengan kebutuhan desain.
- Flexbox menyediakan properti seperti **justifyContent** dan **alignItems** untuk mengontrol perataan elemen dalam kontainer. Ini memungkinkan penataan vertikal dan horizontal yang fleksibel.

Using Flexbox

Membuat Layout menggunakan Flexbox

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

const FlexboxExample = () => {
  return (
    <View style={styles.container}>
      <View style={styles.header}>
        <Text style={styles.headerText}>Header</Text>
      </View>

      <View style={styles.content}>
        <View style={styles.sidebar}>
          <Text style={styles.sidebarText}>Sidebar</Text>
        </View>

        <View style={styles.mainContent}>
          <Text>Main Content</Text>
          <Text>Main Content</Text>
          <Text>Main Content</Text>
        </View>
      </View>

      <View style={styles.footer}>
        <Text style={styles.footerText}>Footer</Text>
      </View>
    </View>
  );
};
```

Using Flexbox

Membuat styles untuk header

```
container: {  
  flex: 1,  
  flexDirection: 'column', // Default is column  
},  
header: {  
  flex: 1,  
  backgroundColor: 'skyblue',  
  justifyContent: 'center',  
  alignItems: 'center',  
},  
headerText: {  
  fontSize: 18,  
  fontWeight: 'bold',  
},
```

Using Flexbox

Membuat styles untuk content

```
content: {  
  flex: 2,  
  flexDirection: 'row',  
},  
sidebar: {  
  flex: 1,  
  backgroundColor: 'lightgreen',  
  justifyContent: 'center',  
  alignItems: 'center',  
},  
sidebarText: {  
  fontSize: 16,  
},  
mainContent: {  
  flex: 3,  
  backgroundColor: 'lightcoral',  
  justifyContent: 'center',  
  alignItems: 'center',  
},
```

Using Flexbox

Membuat styles untuk footer

```
footer: {  
  flex: 1,  
  backgroundColor: 'lightgray',  
  justifyContent: 'center',  
  alignItems: 'center',  
},  
footerText: {  
  fontSize: 18,  
},  
));
```


Using Flexbox

Penjelasan singkat tentang struktur layout ini:

Komponen utama (**container**) menggunakan **flex: 1** untuk mengisi seluruh ruang layar.

header, sidebar, mainContent, dan footer menggunakan **flex** untuk mendefinisikan seberapa besar ruang yang mereka ambil dalam layout.

Masing-masing komponen dalam **content** menggunakan **flex** untuk mengatur proporsi ruang mereka di dalam baris.


justifyContent dan **alignItems** digunakan untuk menentukan cara komponen mengisi ruang yang tersedia.

Navigation antar Page

Untuk membuat navigasi antar halaman dalam sebuah aplikasi React Native, Anda dapat menggunakan library navigasi seperti **react-navigation**. Berikut adalah contoh sederhana menggunakan **react-navigation di App.js**:

Instalasi library navigasi:

bash

 Copy code

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

Navigation antar Page

Buat komponen-komponen yang akan digunakan:

Membuat HomeScreen.js

jsx

Copy code

```
// components/HomeScreen.js
import React from 'react';
import { View, Text, Button } from 'react-native';

const HomeScreen = ({ navigation }) => {
  return (
    <View style={{ flex: 1, justifyContent: 'center', alignItems: 'center' }}>
      <Text>Home Screen</Text>
      <Button
        title="Go to Details"
        onPress={() => navigation.navigate('Details')}
      />
    </View>
  );
};

export default HomeScreen;
```

Navigation antar Page

Membuat DetailScreen.js

jsx

Copy code

```
// components/DetailsScreen.js
import React from 'react';
import { View, Text } from 'react-native';

const DetailsScreen = () => {
  return (
    <View style={{ flex: 1, justifyContent: 'center', alignItems: 'center' }}>
      <Text>Details Screen</Text>
    </View>
  );
};

export default DetailsScreen;
```

Navigation antar Page

Membuat konfigurasi navigasi pada **App.js**

Penggunaan ***createStackNavigator*** untuk membuat tumpukan navigasi.

Stack.Screen digunakan untuk menentukan setiap layar dalam tumpukan.

Saat tombol **"Go to Details"** ditekan di **HomeScreen**, navigasi akan beralih ke **DetailsScreen**.

jsx

Copy code

```
// App.js
import React from 'react';
import { NavigationContainer } from '@react-navigation/native';
import { createStackNavigator } from '@react-navigation/stack';
import HomeScreen from './components/HomeScreen';
import DetailsScreen from './components/DetailsScreen';

const Stack = createStackNavigator();

const App = () => {
  return (
    <NavigationContainer>
      <Stack.Navigator initialRouteName="Home">
        <Stack.Screen name="Home" component={HomeScreen} />
        <Stack.Screen name="Details" component={DetailsScreen} />
      </Stack.Navigator>
    </NavigationContainer>
  );
};

export default App;
```




Referensi Flexbox



Thank you