



Meeting 7 - Do It Yourself!

Sebelum membuat kondisi untuk kalah, maka kamu harus menyiapkan screen yang akan dituju terlebih dahulu!



Yuk pergi ke halaman berikutnya untuk melihat output dari game over screen nantinya!







Tampilan yang dibuat akan terlihat seperti gambar di samping ya!

Yuk langsung saja menuju halaman berikutnya untuk mengimplementasikan code-nya ya!





Ketik dan lengkapi code berikut pada file GameOverScreen.js kamu ya!



marginTop: 32 }}>
<TouchableOpacity onPress={() => ...}> 5

Play Again

</Text>
</TouchableOpacity>

</View>

<Text style={{ textTransform: 'uppercase',
 fontSize: 24,</pre>

fontWeight: 'bold' }}>

)

};

</View>

export default GameOverScreen;





MINT!

- Image background berasal dari link https://wallpaperaccess.com/full/3147153.jpg
- 2. Atur transparansi dari image background menjadi 0.5
- 3. Penempatan component yang ada di dalam image background akan berada di bagian tengah (horizontal maupun vertikal)
- Gambar game over diambil dari file internal
 .../.../assets/images/game-over.png/
- 5. Saat tombol **PLAY AGAIN** diklik maka akan berpindah ke halaman guess the country.

-- SELAMAT MENCOBA --

Jangan lupa untuk menambahkan GameOverScreen component kamu ke dalam stack navigation ya! Yuk ketik penambahan code pada file MainNavigator.js ya!

```
import GameOverScreen from `../screens/GameOverScreen';

{/* tambahkan di dalam <Stack.Navigator> */}

<Stack.Screen

    name="GameOver"
    component={GameOverScreen}
    options={{ headerShown: false }}

/>
```

Untuk dapat melihat game over screen kamu, jangan lupa ganti initialRouteName="GameOver" ya!

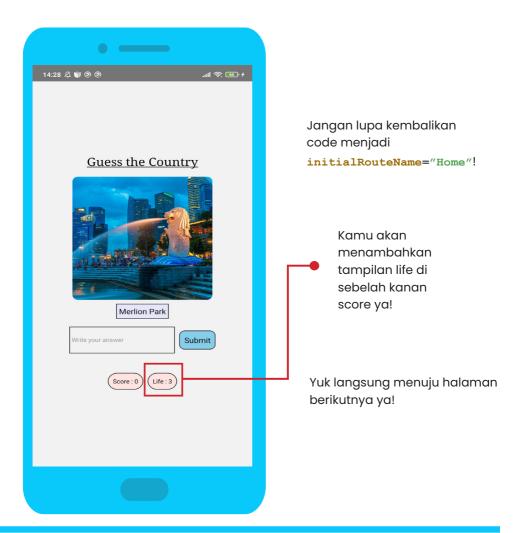




- 1
- Apakah tampilan game over screen kamu sudah seperti gambar pada halaman 2?
- 1

Apakah tombol PLAY AGAIN bekerja dengan benar ketika kamu klik?

Oke, tugas kamu sekarang adalah menambahkan tampilan life dulu seperti gambar berikut ya!







Ketik penambahan code berikut pada file GuessTheCountryScreen.js ya!

```
Codes
//letakkan di bawah state score
const [life, setLife] = useState(3)
{/*letakkan di bawah text component yang menampilkan result*/}
<View
    style={{
        justifyContent: 'center',
        alignItems: 'center',
        padding: 16,
        flexDirection: \...'
    }}>
    <View>
        {/* di sini tampilan score yang sudah ada*/}
    </View>
    {/* TAMBAHKAN CODE DI BAWAH INI */}
    <View
        style={{
            borderWidth: 1,
            padding: 8,
            alignItems: 'center',
            backgroundColor: 'mistyrose',
            borderRadius: 20
        11>
        <Text>Life : {life}</Text>
    </View>
</View>
```

Lengkapi code I untuk menempatkan score dan life ke arah horizontal (baris)

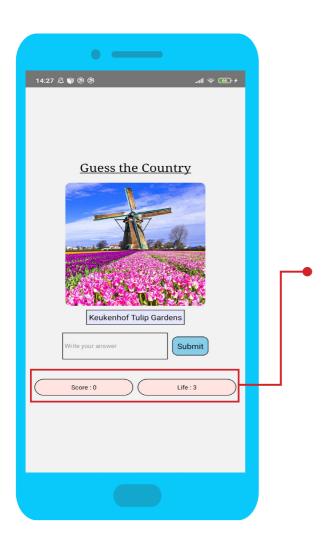






Bukankah tampilan score dan life tidak memiliki jarak?

Kamu dapat memberikan properti marginRight: 8 pada <View> yang membungkus tampilan score kamu ya!



Sekarang berikan flex:1
pada masing-masing

<View> yang
membungkus tampilan
score dan life millikmu ya!

Yuk langsung menuju halaman berikutnya dan ikuti arahannya!





Tahap selanjutnya coba kamu kerjakan mandiri dengan mengikuti langkah-langkah di bawah ini ya!



- 1. Ketika jawaban kamu salah, maka life akan berkurang sebanyak 1
- 2. Buatkan kondisi, dimana jika **life** sudah bernilai **0** maka akan langsung menuju **game over screen** ya! Kamu bisa memanfaatkan **useEffect** hook lho!
- 3. Coba kamu mainkan game dan sengaja membuat life menjadi 0!
- 4. Apakah berhasil menuju ke game over screen?
- 5. Jika jawaban di nomor 4 adalah iya, maka tugas kamu selanjutnya adalah klik button **PLAY AGAIN** ya!
- 6. Apakah life kamu masih 0 ketika mengulang permainan?
- 7. Ingatkah kamu apa yang harus kamu atur saat kamu **focus** lagi ke guess the country screen?
- 8. Atur kembali life menjadi 3 ketika mengulang permainan ya!

-- SELAMAT MENCOBA --