

a) Teljes kód

```
import { React, useState } from 'react';
import { StatusBar } from 'expo-status-bar';
import { StyleSheet, Text, View, TextInput, Pressable, TouchableOpacity,
Alert} from 'react-native';
import { SafeAreaView, SafeAreaProvider } from 'react-native-safe-area-
context';
import { useFonts, Montserrat_400Regular } from '@expo-google-
fonts/montserrat'; //egy nekem tetszo google font importalasa

export default function App() {
  const [fontsLoaded] = useFonts({Montserrat_400Regular});
  const [age, setAge] = useState('');
  const [height, setHeight] = useState('');
  const [weight, setWeight] = useState('');
  const [gender, setGender] = useState('férfi');
  const [heightUnit, setHeightUnit] = useState('cm');
  const [weightUnit, setWeightUnit] = useState('kg');
  const [bmi, setBmi] = useState('');
  const [bmiCategory, setBmiCategory] = useState(null);

  const [ferfiErtekek, setFerfiErtekek] = useState(
    [
      { ertekek: 16.0, kategoria: 'súlyos soványság', ertelmezes: 'Sürgös orvosi
beavatkozást igényel' },
      { ertekek: 16.9, kategoria: 'mérsékelt soványság', ertelmezes: 'Jelentős
súlygyarapodás javasolt' },
      { ertekek: 18.4, kategoria: 'enyhe soványság', ertelmezes: 'Súlygyarapodás
javasolt' },
      { ertekek: 24.9, kategoria: 'normál testsúly', ertelmezes: 'Egészséges
tartomány' },
      { ertekek: 29.9, kategoria: 'túlsúly', ertelmezes: 'Mérsékelt fogyás
javasolt' },
      { ertekek: 34.9, kategoria: 'I. fokú elhízás', ertelmezes: 'Fogyás és
életmódváltás javasolt' },
      { ertekek: 39.9, kategoria: 'II. fokú elhízás', ertelmezes: 'Jelentős fogyás
szükséges' },
      { ertekek: 40, kategoria: 'III. fokú elhízás', ertelmezes: 'Orvosi
felügyelet mellett fogyás szükséges' }
    ]
  );

  const [noiErtekek, setNoiErtekek] = useState(
    [
      { ertekek: 15.0, kategoria: 'súlyos soványság', ertelmezes: 'Sürgös orvosi
beavatkozást igényel' },
```

```

    { ertekek: 15.9, kategoria: 'mérsékelt soványság', ertelmezes: 'Jelentős
súlygyarapodás javasolt' },
    { ertekek: 18.4, kategoria: 'enyhe soványság', ertelmezes: 'Súlygyarapodás
javasolt' },
    { ertekek: 23.9, kategoria: 'normál testsúly', ertelmezes: 'Egészséges
tartomány' },
    { ertekek: 28.9, kategoria: 'túlsúly', ertelmezes: 'Mérsékelt fogyás
javasolt' },
    { ertekek: 33.9, kategoria: 'I. fokú elhízás', ertelmezes: 'Fogyás és
életmódváltás javasolt' },
    { ertekek: 38.9, kategoria: 'II. fokú elhízás', ertelmezes: 'Jelentős fogyás
szükséges' },
    { ertekek: 39, kategoria: 'III. fokú elhízás', ertelmezes: 'Orvosi
felügyelet mellett fogyás szükséges' }
]);

const getBmiCategory = (bmiValue, gender) =>{
    const categories = gender === 'férfi' ? ferfiErtekek : noiErtekek;

    if (bmiValue >= categories[categories.length - 1].ertekek) {
        const lastCategory = categories[categories.length - 1];
        return {
            Bmierterkek: lastCategory.ertekek,
            kategoria: lastCategory.kategoria,
            ertelmezes: lastCategory.ertelmezes
        };
    }

    for(let i=0; i< categories.length - 1; i++)
    {
        if(bmiValue <= categories[i].ertekek)
        {
            return{
                Bmierterkek: categories[i].ertekek,
                kategoria: categories[i].kategoria,
                ertelmezes: categories[i].ertelmezes
            };
        }
    }
};

const calculateBmi = () =>{
    if (height && weight) {

        /*min 50 cm max cm, ezeket leellenorzi inchben is valamint min 3,4 kg
max kg ezeket ellenorzi fontban is
        minimum ertekek az ujszoletett csecsemoke, maximum ertekek pedig ritkan
elofordulo adatok*/}
    }

```

```

        if((height >= 50 && height <= 225) && ((height * 0.39) >= 19.7 &&
(height * 0.39) <= 88.6) && (weight >= 3.4 && weight <= 150) && ((weight *
2.20) >= 7.5 && (weight * 2.20) <= 330.75)){
            let heightValue = parseFloat(height);
            let weightValue = parseFloat(weight);

            if(heightUnit === 'foot'){
                heightValue *= 30.48; //váltás centibe, így megadhatja a magasságot
labban, a súlyat kiloban, mindegy, mivel minden vissza van alakítva meterbe és
kiloba
            }
            if(weightUnit === 'lb'){
                weightValue *= 0.45;
            }

            const heightInMeters = heightValue / 100;
            const bmiValue = (weightValue / (heightInMeters *
heightInMeters)).toFixed(2);
            setBmi(bmiValue);

            const category = getBmiCategory(parseFloat(bmiValue), gender);
            setBmiCategory(category);
        } else{
            Alert.alert('Hiba', 'Abnormális értékek!');
        }
    } else {
        Alert.alert('Hiba', 'Kérlek add meg a magasságot és a súlyt!');
    }
};

const deleteData = () => {
    setAge('');
    setHeight('');
    setWeight('');
    setBmi('');
    setBmiCategory(null);
};

const toggleHeightUnit = () => {
    setHeightUnit(heightUnit === 'cm' ? 'foot' : 'cm');
    setHeight(''); // Értékek törlése, ha váltás történik
};

const toggleWeightUnit = () => {
    setWeightUnit(weightUnit === 'kg' ? 'lb' : 'kg');
    setWeight('');
};

if(!fontsLoaded){

```

```

    return null;
}

return (
  <SafeAreaView>
    <SafeAreaView style={styles.container} edges={['top', 'bottom']}>
      <View>
        <Text style={styles.text, {fontSize: 36}}>BMI kalkulátor</Text>

        <Text style={styles.text}>Korod</Text>
        <View style={{flexDirection: 'row', paddingBottom: 20}}>
          <TextInput style={styles.textInput, {paddingLeft: 10}}
            onChangeText={setAge}
            value={age}
            placeholder="Írd ide a korod..."
            keyboardType="numeric"/>
          <Text style={styles.units}>év</Text>
        </View>

        <Text style={styles.text}>Magasságod</Text>
        <View style={{flexDirection: 'row', paddingBottom: 20}}>
          <TextInput style={styles.textInput, {paddingLeft: 10}}
            onChangeText={setHeight}
            value={height}
            placeholder="Írd ide a magasságod..."
            keyboardType="numeric"/>
          <Pressable
            onPress={toggleHeightUnit}
            style={({pressed}) => [
              styles.units,
              {
                backgroundColor: pressed ? '#F4CE14' : '#D7DDF7',
                borderLeftWidth: 1,
              },
            ]>
            <Text style={{
              fontFamily: 'Montserrat_400Regular',
              color: '#000'
            }}>{heightUnit}</Text>
          </Pressable>
        </View>

        <Text style={styles.text}>Súlyod</Text>
        <View style={{flexDirection: 'row', paddingBottom: 40}}>
          <TextInput style={styles.textInput, {paddingLeft: 10}}
            onChangeText={setWeight}
            value={weight}
            placeholder="Írd ide a súlyod..."
            keyboardType="numeric"/>

```

```

        <Pressable
          onPress={toggleWeightUnit}
          style={({pressed}) => [
            styles.units,
            {
              backgroundColor: pressed ? '#F4CE14' : '#D7DDF7',
              borderLeftWidth: 1,
            },
          ]}>
        <Text style={{
          fontFamily: 'Montserrat_400Regular',
          color: '#000',
        }}>{weightUnit}</Text>
      </Pressable>
    </View>

    <Text style={[[styles.text, {textAlign: 'center', fontSize:
22}]]}>Nemed</Text>
    <View style={{flexDirection: 'row', paddingBottom: 20,
justifyContent: 'space-between'}}>
      <Pressable
        onPress={() => setGender('férfi')}
        style={({pressed}) => [
          styles.buttons,
          {
            backgroundColor: pressed ? '#F4Ce14' : '#0B31EB',
          },
        ]}>
        {({pressed}) => (
          <Text style={{ color: pressed ? '#000' : '#f5f7f8' }}>{pressed ?
'Male' : 'Férfi'}</Text> //ha nincs lenyomva a gomb akkor a szoveg szine
f5f7f8 erteku
        )}
      </Pressable>

      <Pressable
        onPress={() => setGender('nő')}
        style={({pressed}) => [
          styles.buttons,
          {
            backgroundColor: pressed ? '#F4CE14' : '#0B31EB',
          },
        ]}>
        {({pressed}) => (
          <Text style={{ color: pressed ? '#000' : '#f5f7f8' }}>{pressed ?
'Female' : 'Nő'}</Text>
        )}
      </Pressable>
    </View>

```

```

    <View style={{flexDirection: 'row'}}>
      <TouchableOpacity
        onPress={()=>calculateBmi()} activeOpacity={0.8}>
        <Text style={[styles.touchableOpacity]}>BMI kiszámítása</Text>
      </TouchableOpacity>
      <TouchableOpacity
        onPress={()=>deleteData()} activeOpacity={0.8}>
        <Text
style={[styles.touchableOpacity,{marginLeft:19}]}>Törlés</Text>
      </TouchableOpacity>
    </View>

    {bmi && bmiCategory && (
      <TextInput style={styles.outputBox}
        value ={
          gender === 'férfi'
            ? `BMI értéked: ${bmi}\nA ${gender}aknál az adott adatokkal ez
${bmiCategory.kategoria}.\nÉrtelmezés: ${bmiCategory.ertelmezes}`
            : `BMI értéked: ${bmi}\nA ${gender}knél az adott adatokkal ez
${bmiCategory.kategoria}.\nÉrtelmezés: ${bmiCategory.ertelmezes}`
          }
        editable={false} multiline={true}/>
    )}
  </View>
</SafeAreaView>
</SafeAreaProvider>
);
}

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#000',
    alignItems: 'center',
    justifyContent: 'center',
    borderWidth: 8,
  },
  text:{
    fontFamily: 'Montserrat_400Regular',
    fontSize: 18,
    color: '#f5f7f8',
    paddingBottom: 2
  },
  textInput:{
    fontFamily: 'Montserrat_400Regular',
    backgroundColor: '#f5f7f8',
    width: 225,
    height: 45,

```

```
borderTopLeftRadius: 8,
borderBottomLeftRadius: 8
},
units:{
  fontFamily: 'Montserrat_400Regular',
  backgroundColor: '#f5f7f8',
  paddingHorizontal: 10,
  paddingVertical: 11,
  textAlign: 'center',
  alignItems: 'center',
  justifyContent: 'center',
  borderLeftWidth: 1,
  width: 50,
  height: 45,
  borderTopRightRadius: 8,
  borderBottomRightRadius: 8,
},
buttons: {
  fontFamily: 'Montserrat_400Regular',
  width: 70,
  height: 45,
  borderRadius: 8,
  alignItems: 'center',
  justifyContent: 'center'
},
touchableOpacity:{
  fontFamily: 'Montserrat_400Regular',
  width: 'auto',
  maxWidth: 275,
  height: 45,
  padding: 9,
  borderRadius: 8,
  textAlign: 'center',
  backgroundColor: '#f5f7f8',
  fontSize: 20
},
outputBox: {
  fontFamily: 'Montserrat_400Regular',
  height: 'auto',
  maxWidth: 275,
  borderWidth: 1,
  borderRadius: 8,
  padding: 10,
  marginTop: 10,
  backgroundColor: '#f2f2f2',
},
});
```

b) Futó alkalmazás screenshot-ja

The screenshot shows a mobile application interface for a BMI calculator. At the top, the status bar displays the time 14:25 and various connectivity icons. The app title 'BMI kalkulátor' is centered in a large white font. Below it, there are three input sections: 'Korod' (Your age) with a value of 25 and unit 'év' (years); 'Magasságod' (Your height) with a value of 168 and unit 'cm'; and 'Súlyod' (Your weight) with a value of 88 and unit 'kg'. A 'Nemed' (Gender) section has two blue buttons: 'Férfi' (Male) and 'Nő' (Female). Below these are two white buttons: 'BMI kiszámítása' (Calculate BMI) and 'Törlés' (Clear). At the bottom, a white box contains the calculated BMI value (31.18) and a message in Hungarian indicating that the user is in the first degree of obesity and suggesting a diet and lifestyle change.

14:25

BMI kalkulátor

Korod

25 év

Magasságod

168 cm

Súlyod

88 kg

Nemed

Férfi Nő

BMI kiszámítása Törlés

BMI értéked: 31.18  
A férfiaknál az adott adatokkal ez I. fokú elhízás.  
Értelmezés: Fogyás és életmódváltás javasolt