

# Let's Get Fit Together

Login

Register



## Fitness App

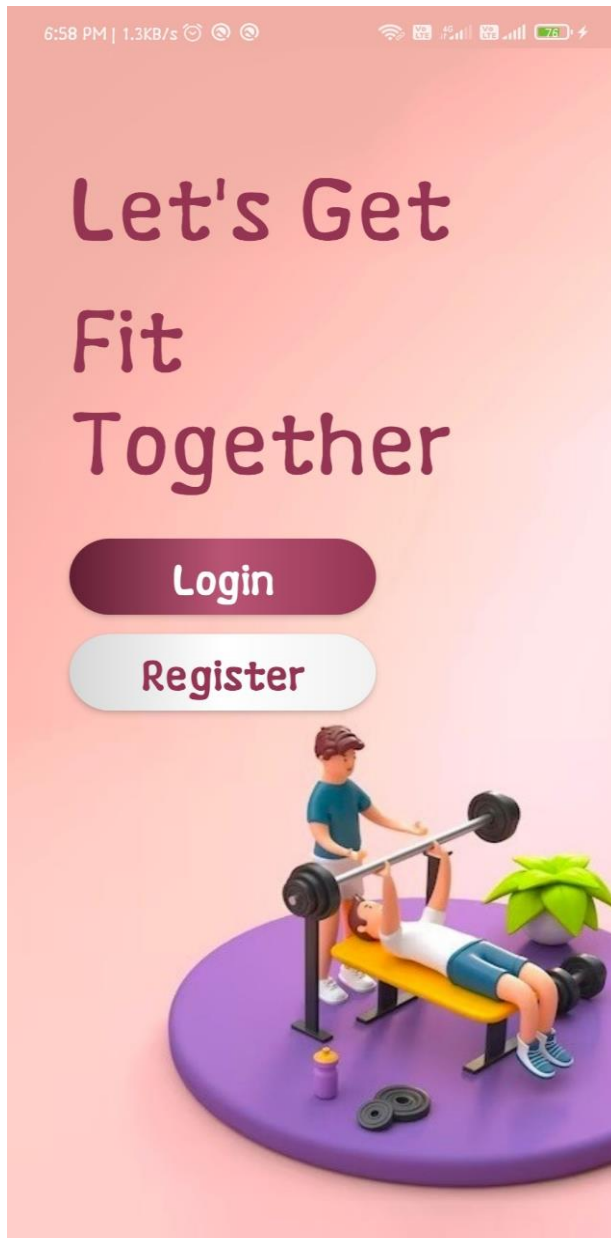
RSL SOLUTION

Saurabh Potdukhe | React Native Project | 05 October 2023

# App.js

```
18
19 function App() {
20   return (
21     <>
22       <StatusBar backgroundColor='#f0aea5' />
23       <NavigationContainer>
24         <Stack.Navigator initialRouteName='Home' screenOptions={{ headerShown: false }}>
25           <Stack.Screen name="Home" component={Home} />
26           <Stack.Screen name="Signup" component={Signup} />
27           <Stack.Screen name="Login" component={Login} />
28           <Stack.Screen name="Settings" component={Settings} />
29           <Stack.Screen name="BMICalculator" component={BMICalculator} />
30           <Stack.Screen name="GenderScreen" component={GenderScreen} />
31           <Stack.Screen name="HomeScreen" component={HomeScreen} />
32           <Stack.Screen name="ModeScreen" component={ModeScreen} />
33           <Stack.Screen name="WeekScreen" component={WeekScreen} />
34           <Stack.Screen name="ExerciseScreen" component={ExerciseScreen} />
35           <Stack.Screen name="ExerciseDetail" component={ExerciseDetail} />
36         </Stack.Navigator>
37       </NavigationContainer>
38     </>
39   );
40 }
41
42
43 export default App;
44
```

# Home Screen



```
6
7 const Home = (props) => {
8   return (
9     <Background>
10      <View style={{ marginHorizontal: 40, marginVertical: 60 }}>
11        <Text style={{ color: purple, fontSize: 60 }}>Let's Get</Text>
12        <Text style={{ color: purple, fontSize: 60 }}>Fit</Text>
13        <Text style={{ color: purple, fontSize: 60, marginBottom: 25 }}>Together</Text>
14        <Btn textColor={white}
15          btnLabel='Login'
16          Press={() => props.navigation.navigate("Login")} />
17        <Btn textColor={purple}
18          btnLabel='Register'
19          Press={() => props.navigation.navigate("Signup")} />
20      </View>
21    </Background>
22  );
23 }
24
25 export default Home;
```

# Login Screen

6:58 PM | 1.4KB/s

# Login

## Welcome Back

Login to your account

Email

Password

Forgot Password ?

Login

Don't have an account ? [Signup](#)

```
9 import { connectAuthEmulator, signInWithEmailAndPassword } from 'firebase/auth';
10 import { auth } from '../config/firebase';
11
12 const LoginScreen = (props) => {
13   const navigation = useNavigation();
14   const [email, setEmail] = useState('');
15   const [password, setPassword] = useState('');
16   const [validationError, setValidationError] = useState('');
17
18   const handleSubmit = async() =>{
19     setValidationError('');
20     if(email && password) {
21       try{
22         await signInWithEmailAndPassword(auth, email, password);
23         alert('Logged In');
24       }catch(err){
25         setValidationError('Login error : ',err.message);
26       }
27     }
28     else {
29       setValidationError('Please fill out all fields.');
```

```
return (
  <Background>
    <View style={styles.container}>
      <Text style={styles.title}>Login</Text>
      <View style={styles.formContainer}>
        <Text style={styles.welcomeText}>Welcome Back</Text>
        <Text style={styles.loginDescription}>Login to your account</Text>
        <Field placeholder="Email" keyboardType="email-address" value={email} onChangeText={value => setEmail}>
        <Field placeholder="Password" secureTextEntry={true} value={password} onChangeText={value=>setPassword}>
        <TouchableOpacity style={styles.forgotPassword}
          onPress={() => { alert('Forgot Password?'); }} >
          <Text style={styles.forgotPasswordText}>Forgot Password ?</Text>
        </TouchableOpacity>
        <Btn textColor={white} bgColor={purple} btnLabel="Login"
          Press={() => { alert('Logged In');
            props.navigation.navigate('GenderScreen'); }}
        </Btn>
      </View>
      <View style={styles.signupContainer}>
        <Text style={styles.signupText}>Don't have an account ?</Text>
        <TouchableOpacity onPress={() => props.navigation.navigate('Signup')}>
          <Text style={styles.signupLink}>Signup</Text>
        </TouchableOpacity>
      </View>
    </View>
  </Background>
);
```

# Sign-up Screen

6:58 PM | 1.4KB/s

## Register

Create a new account

First Name

Last Name

Email Address

Contact Number

Password

Confirm Password

**Submit**

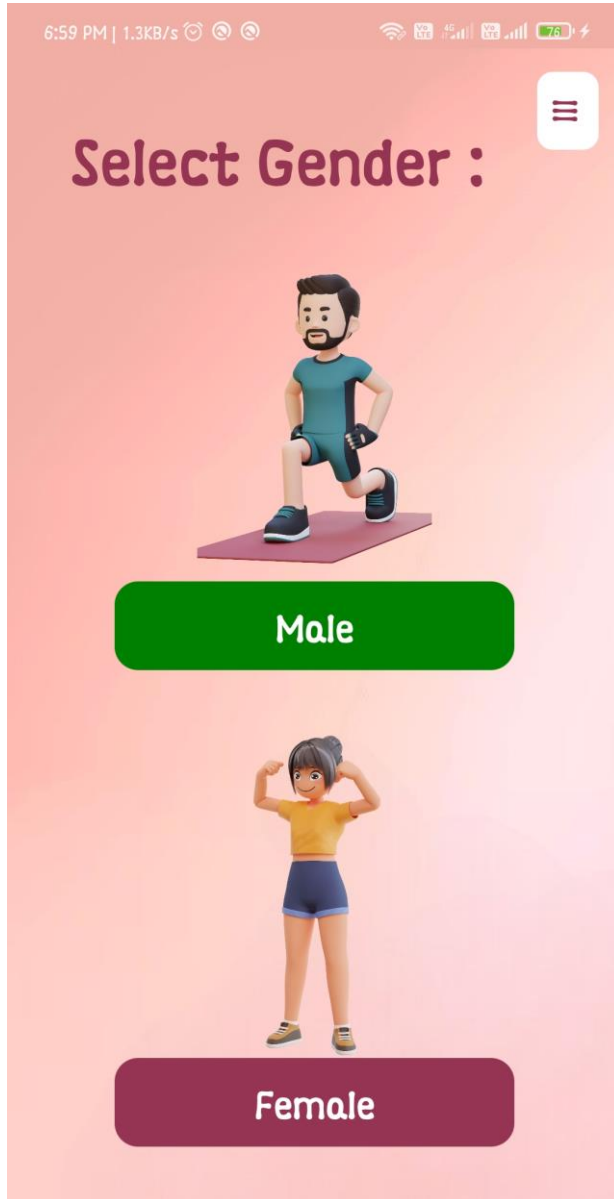
Already have an account ? [Login](#)

```
const SignUpScreen = (props) => {
  const navigation = useNavigation();
  const [firstName, setFirstName] = useState('');
  const [lastName, setLastName] = useState('');
  const [email, setEmail] = useState('');
  const [contactNumber, setContactNumber] = useState('');
  const [password, setPassword] = useState('');
  const [confirmPassword, setConfirmPassword] = useState('');
  const [validationError, setValidationError] = useState('');

  const handleSubmit = async () => {
    setValidationError('');
    const passwordRegex = /^(?=.*[A-Za-z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}$/;
    if (!firstName || !lastName || !email || !contactNumber || !password || !confirmPassword) {
      setValidationError('Please fill out all fields.');
```

```
<Background>
  <ScrollView style={styles.scrollView}>
    <View style={styles.container}>
      <Text style={styles.registerText}>Register</Text>
      <View style={styles.formContainer}>
        <Text style={styles.formHeaderText}>Create a new account</Text>
        <Field placeholder="First Name" value={firstName} onChangeText={setFirstName} />
        <Field placeholder="Last Name" value={lastName} onChangeText={setLastName} />
        <Field placeholder="Email Address" keyboardType="email-address" value={email} onChangeText={setEmail} />
        <Field placeholder="Contact Number" keyboardType="number" value={contactNumber} onChangeText={setContactNumber} />
        <Field placeholder="Password" secureTextEntry={true} value={password} onChangeText={setPassword} />
        <Field placeholder="Confirm Password" secureTextEntry={true} value={confirmPassword} onChangeText={setConfirmPassword} />
        <Text style={styles.errorText}>{validationError}</Text> : null
      <Text></Text>
      <Btn
        textColor={white}
        bgColor={purple}
        btnLabel="Submit"
        Press={handleSignup}
      />
      <View style={styles.loginContainer}>
        <Text style={styles.loginText}>Already have an account ?</Text>
        <TouchableOpacity onPress={() => props.navigation.navigate("Login")}>
          <Text style={styles.loginLink}>Login</Text>
        </TouchableOpacity>
      </View>
    </View>
  </ScrollView>
</Background>
```

# Gender Screen



```
const handleGenderSelection = (selectedGender) => {  
  setGender(selectedGender);  
  props.navigation.navigate("HomeScreen", { selectedGender: selectedGender });  
};
```

```
<Text style={style.title}> Select Gender : </Text>  
<ScrollView style={style.scrollStyle}  
  showsVerticalScrollIndicator={false} >  
  { genderOptions.map((option) => (  
    <View key={option} style={{ alignItems: 'center', marginBottom: 20 }}>  
      <Image  
        style={{ width: 150, height: 210, }}  
        source={ option === 'Male'  
          ? require('../images/male_character.png')  
          : require('../images/female_character.png')  
        }  
      />  
      <BtnMenu  
        bgColor={option === gender ? 'green' : 'purple'}  
        textColor="white"  
        btnLabel={option}  
        Press={() => {handleGenderSelection(option);}}  
      />  
    </View>  
  ) ) }  
  <View style={{ marginBottom: 60 }}></View>  
</ScrollView>
```



# Exercise Type Screen



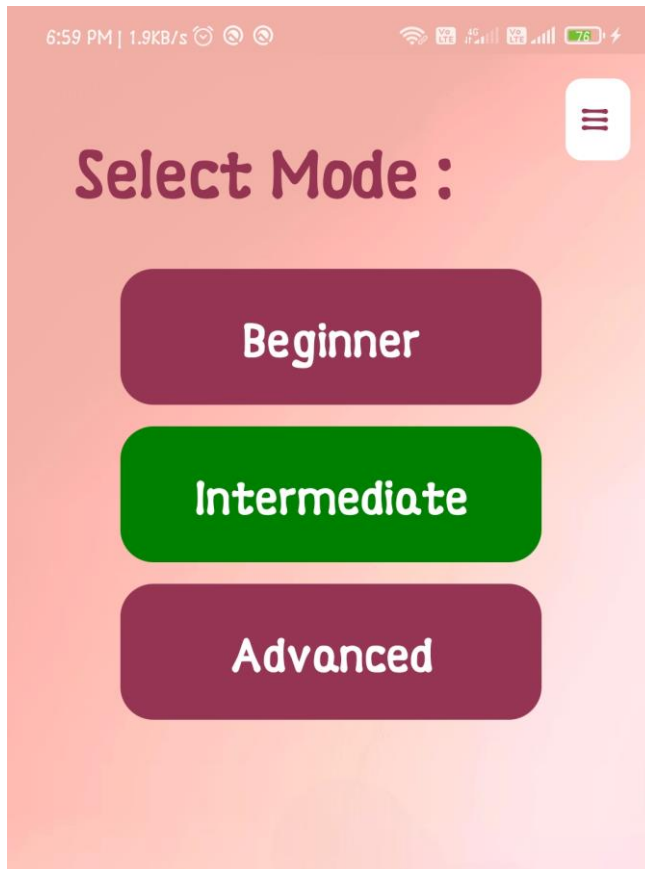
```
const [menuVisible, setMenuVisible] = useState(false);
const [selectedExerciseType, setSelectedExerciseType] = useState('');
const exerciseTypeOptions = [
  'Gym Exercise',
  'Dumbbells Exercise',
  'Exercise without Equipments',
];
const selectedGender = props.route.params.selectedGender;

const handleExerciseTypeSelection = (exerciseType) => {
  setSelectedExerciseType(exerciseType);
  props.navigation.navigate("ModeScreen", {
    selectedGender: selectedGender,
    exerciseType: exerciseType,
  });
};
```

```
<Text style={style.title}>
  Select Exercise Type :
</Text>

<ScrollView style={style.scrollStyle}
  showsVerticalScrollIndicator={false} >
  {exerciseTypeOptions.map((option) => (
    <BtnMenu
      key={option}
      bgColor={option === selectedExerciseType ? 'green' : purple}
      textColor="white"
      btnLabel={option}
      Press={() => handleExerciseTypeSelection(option)}
    />
  ))}
</ScrollView>
```

# Exercise Mode Screen



```
const [selectedMode, setSelectedMode] = useState('');
const modeOptions = ['Beginner', 'Intermediate', 'Advanced'];

const selectedGender = props.route.params.selectedGender;
const exerciseType = props.route.params.exerciseType;

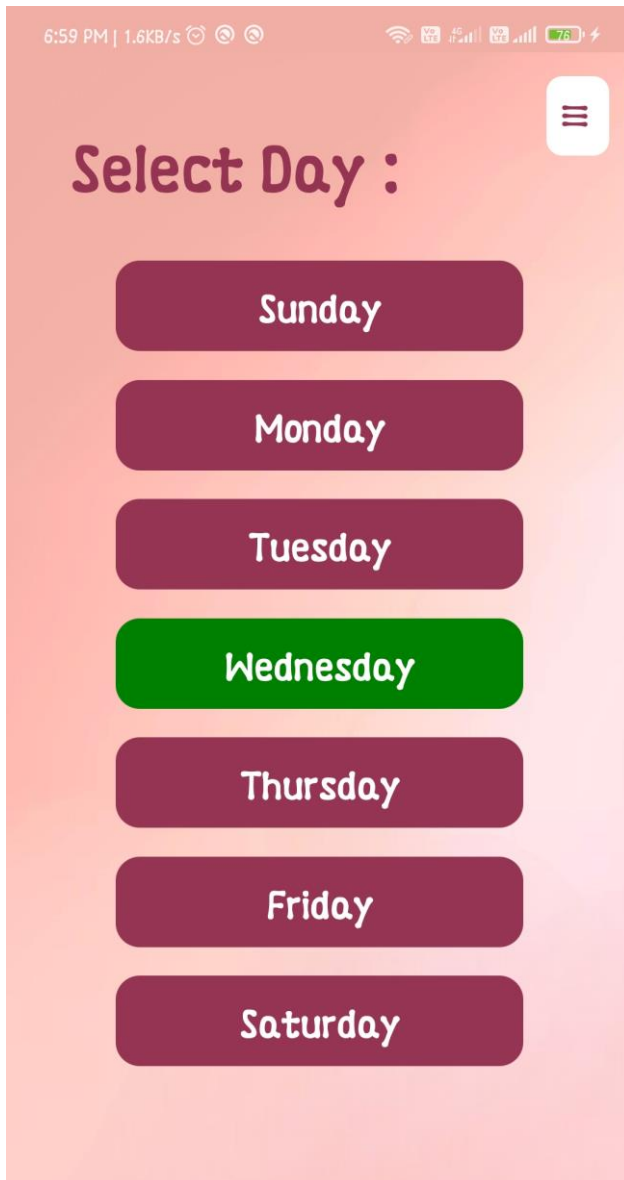
const handleModeSelection = (mode) => {
  setSelectedMode(mode);
  props.navigation.navigate('WeekScreen', {
    selectedGender: selectedGender,
    exerciseType: exerciseType,
    selectedMode: mode,
  });
};
```

```
<Text style={style.title}>
  Select Mode :
</Text>

<ScrollView style={style.scrollStyle}
  showsVerticalScrollIndicator={false} >
  {modeOptions.map((option) => (
    <BtnMenu
      key={option}
      bgColor={option === selectedMode ? 'green' : purple}
      textColor="white"
      btnLabel={option}
      Press={() => handleModeSelection(option)}
    />
  ))}
</ScrollView>
```



# Day of the Week Screen



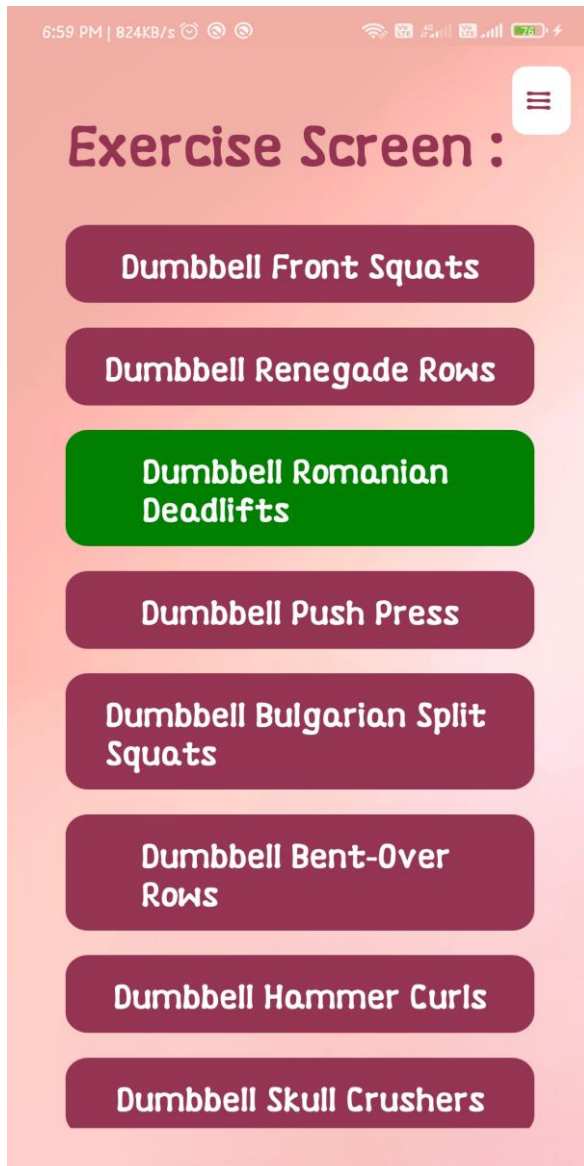
```
const [selectedDay, setSelectedDay] = useState('');
const daysOfWeek = [
  'Sunday',
  'Monday',
  'Tuesday',
  'Wednesday',
  'Thursday',
  'Friday',
  'Saturday',
];

const handleDaySelection = (day) => {
  setSelectedDay(day);
  const { selectedGender, exerciseType, selectedMode } = props.route.params;
  props.navigation.navigate('ExerciseScreen', {
    selectedGender: selectedGender,
    exerciseType: exerciseType,
    selectedMode: selectedMode,
    selectedDay: day,
  });
};
```

```
<Text style={style.title}>
  Select Day :
</Text>

<ScrollView style={style.scrollStyle}
  showsVerticalScrollIndicator={false} >
  {daysOfWeek.map((day) => (
    <BtnMenu
      key={day}
      bgColor={day === selectedDay ? 'green' : purple}
      textColor="white"
      btnLabel={day}
      Press={() => handleDaySelection(day)}
    />
  ))}
  <View style={{ marginBottom: 60 }}></View>
</ScrollView>
```

# Exercise List Screen



```
import dumbbellExercisesMen from '../exerciseData/dumbbellExercisesMen';
import gymExercisesMen from '../exerciseData/gymExercisesMen';
import noEquipmentExercisesMen from '../exerciseData/noEquipmentExercisesMen';
import dumbbellExercisesWomen from '../exerciseData/dumbbellExercisesWomen';
import gymExercisesWomen from '../exerciseData/gymExercisesWomen';
import noEquipmentExercisesWomen from '../exerciseData/noEquipmentExercisesWomen';
```

```
const [data, setData] = useState('');
const { selectedDay, selectedGender, exerciseType, selectedMode } = props.route.params;
const gender = selectedGender === 'Male' ? 'Men' : 'Women';
const selectedExercise = (exerciseType === 'Gym Exercise') ? 'gymExercises'
: ( exerciseType === 'Dumbbells Exercise' ? 'dumbbellExercises'

const fileName = selectedExercise+gender;
let fileOpt;
if(fileName === 'gymExercisesMen')
  fileOpt = gymExercisesMen;
else if(fileName === 'noEquipmentExercisesMen')
  fileOpt = noEquipmentExercisesMen;
else if(fileName === 'dumbbellExercisesWomen')
  fileOpt = dumbbellExercisesWomen;
else if(fileName === 'gymExercisesWomen')
  fileOpt = gymExercisesWomen;
else if(fileName === 'noEquipmentExercisesWomen')
  fileOpt = noEquipmentExercisesWomen;
else
  fileOpt = dumbbellExercisesMen;

let exercises = fileOpt[selectedMode].exercises;

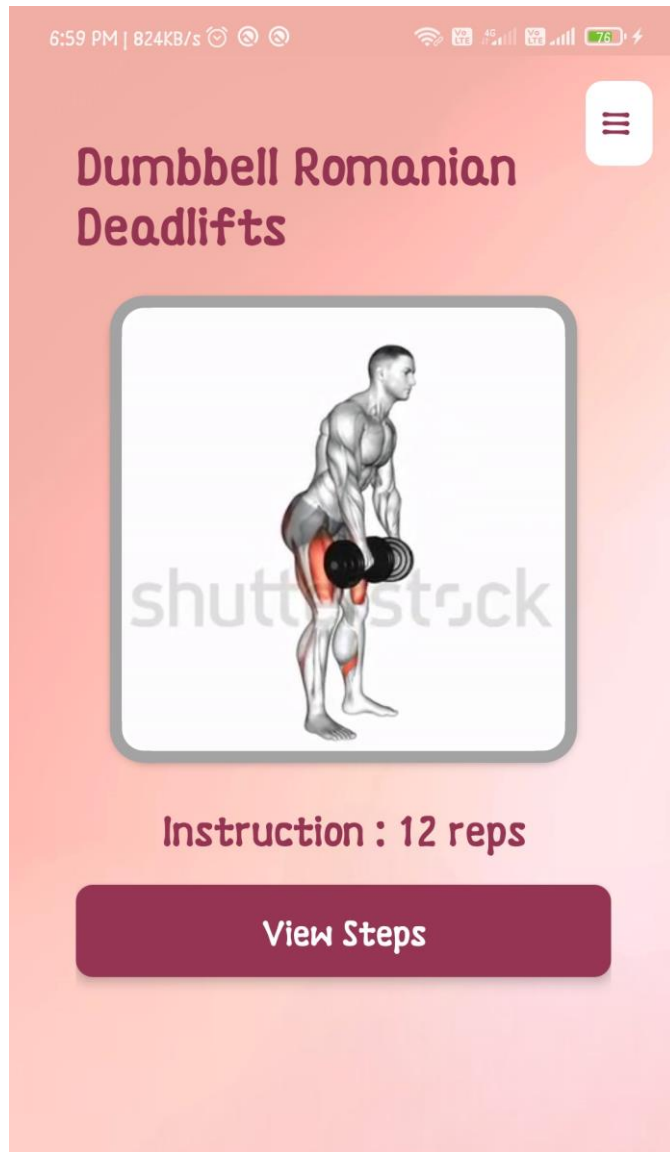
const handleExercisePress = (exercise) => {
  setData(exercise);
  props.navigation.navigate('ExerciseDetail', {
    exercise: exercise,
```

```
<Text style={style.title}>
  Exercise Screen :
</Text>

<ScrollView style={style.scrollStyle}
  showsVerticalScrollIndicator={false} >
  {exercises.map((exercise, index) => (
    <BtnMenu
      key={index}
      bgColor={exercise === data ? 'green' : 'purple'}
      textColor="white"
      btnLabel={exercise.name}
      Press={() => handleExercisePress(exercise)}
    />
  ))}

  <View style={{ marginBottom: 100, }}></View>
</ScrollView>
```

# Exercise Detail Screen



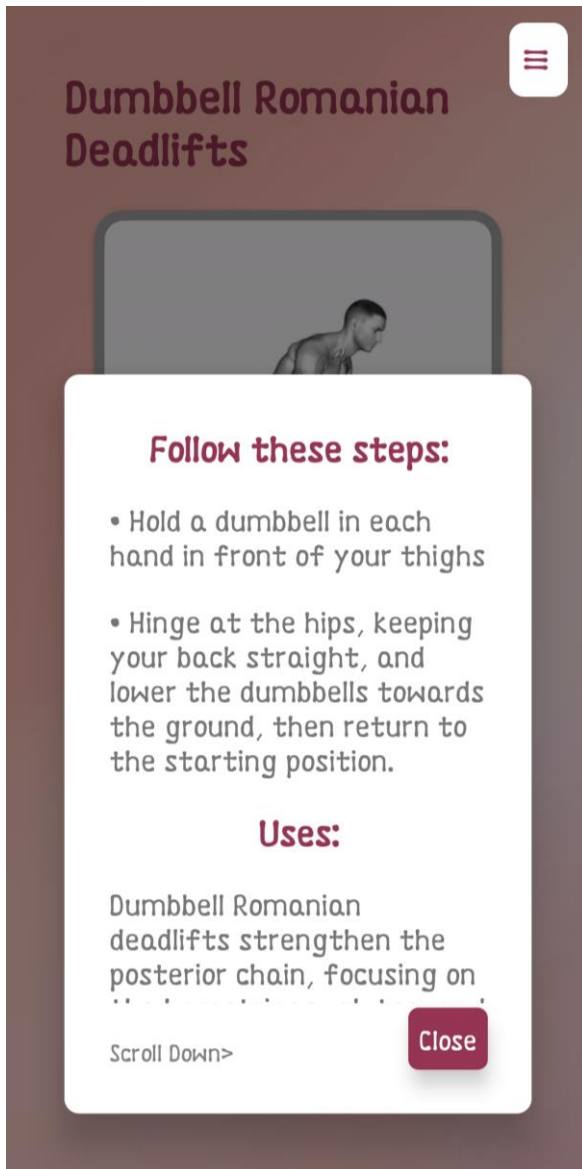
```
// dumbbellExercisesMen (beginner)
const GobletSquats= "../Videos/dumbbellExercisesMen/GobletSquats.mp4";
const DumbbellBenchPress = "../Videos/dumbbellExercisesMen/DumbbellBenchPress.mp4";
const DumbbellBentOverRows = "../Videos/dumbbellExercisesMen/DumbbellBentOverRows.mp4";
const DumbbellLunges = "../Videos/dumbbellExercisesMen/DumbbellLunges.mp4";
const DumbbellShoulderPress = "../Videos/dumbbellExercisesMen/DumbbellShoulderPress.mp4";
const DumbbellRomanianDeadlifts = "../Videos/dumbbellExercisesMen/DumbbellRomanianDeadlifts.mp4";
const DumbbellBicepCurls = "../Videos/dumbbellExercisesMen/DumbbellBicepCurls.mp4";
const TricepDipswithDumbbells = "../Videos/dumbbellExercisesMen/TricepDipswithDumbbells.mp4";
const DumbbellRussianTwists = "../Videos/dumbbellExercisesMen/DumbbellRussianTwists.mp4";
const DumbbellPlankRows = "../Videos/dumbbellExercisesMen/DumbbellPlankRows.mp4";

let videoUri;

switch (pathUrl) {
  // Beginner dumbbell
  case "GobletSquats": videoUri = require(GobletSquats); break;
  case "DumbbellBenchPress": videoUri = require(DumbbellBenchPress); break;
  case "DumbbellBentOverRows": videoUri = require(DumbbellBentOverRows); break;
  case "DumbbellLunges": videoUri = require(DumbbellLunges); break;
  case "DumbbellShoulderPress": videoUri = require(DumbbellShoulderPress); break;
  case "DumbbellRomanianDeadlifts": videoUri = require(DumbbellRomanianDeadlifts); break;
  case "DumbbellBicepCurls": videoUri = require(DumbbellBicepCurls); break;
  case "TricepDipswithDumbbells": videoUri = require(TricepDipswithDumbbells); break;
  case "DumbbellRussianTwists": videoUri = require(DumbbellRussianTwists); break;
  case "DumbbellPlankRows": videoUri = require(DumbbellPlankRows); break;
}
```

```
<Text style={style.subTitle}>
  {exercise.name}
</Text>
<View style={style.videoContainer}>
  <Video
    // source={link2}
    source={videoUri}
    paused={false} // make it start
    repeat={true} // make it a loop
    style={style.video}
    resizeMode= "contain"
  />
</View>
<Text style={style.infoModalTitle}>
  Instruction : {exercise.reps}
</Text>
<TouchableOpacity
  style={style.customButton}
  onPress={toggleInfoModal}
>
  <Text style={style.customButtonText}>View Steps</Text>
</TouchableOpacity>
```

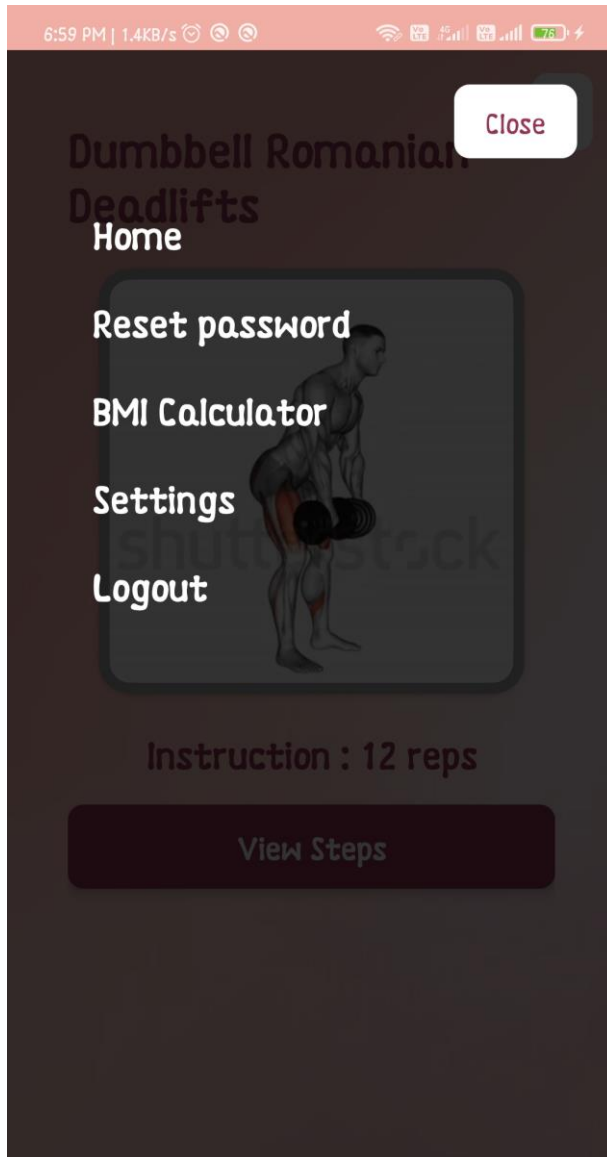
# Exercise Steps & Uses Modal



```
<ScrollView
  showsVerticalScrollIndicator={false}>
  <Text style={style.infoModalTitle}>Follow these steps:</Text>
  <View>
    {exercise.description.split('. ').map((point, index) => (
      <Text key={index} style={styles.infoModalDescription}>
        • {point}
      </Text>
    ))}
  </View>

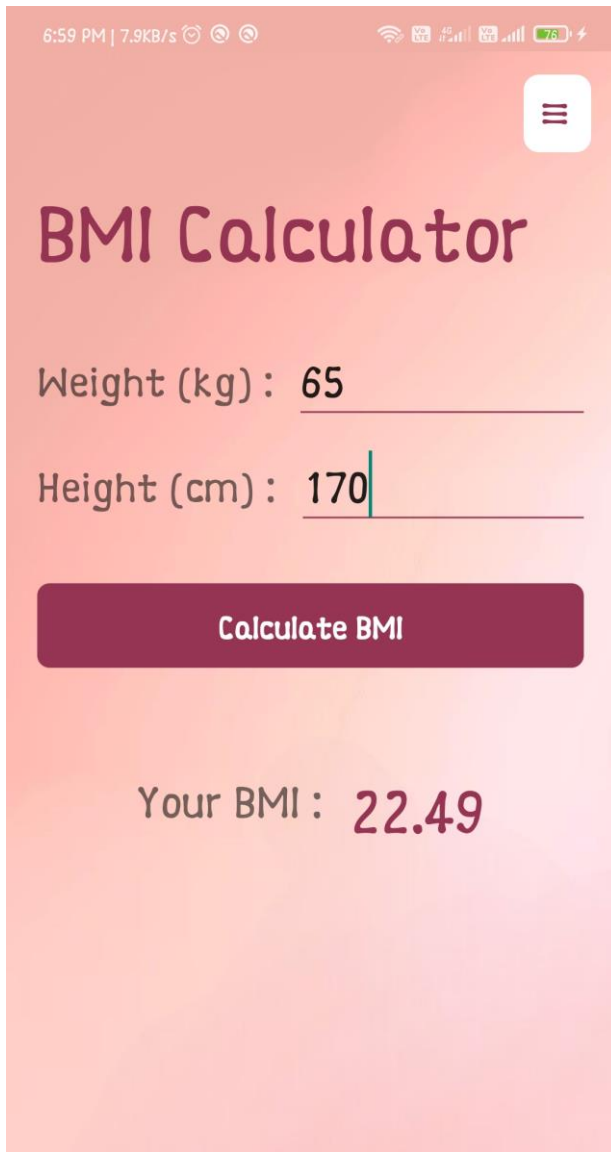
  <Text style={style.infoModalTitle}>Uses:</Text>
  <View>
    <Text style={styles.infoModalDescription}>{exercise.use}</Text>
  </View>
</ScrollView>
```

# Main Menu Modal



```
<Modal
  transparent={true}
  animationType="slide"
  visible={isVisible}
  onRequestClose={toggleMenu}
>
  <View style={styles.menuContainer}>
    <Text style={styles.menuItem} onPress={() => navigation.navigate("GenderScreen")}>Home</Text>
    <Text style={styles.menuItem} onPress={() => alert("Reset password")}>Reset password</Text>
    <Text style={styles.menuItem} onPress={() => navigation.navigate("BMICalculator")}>BMI Calculator</Text>
    <Text style={styles.menuItem} onPress={() => navigation.navigate("Settings")}>Settings</Text>
    <Text style={styles.menuItem} onPress={() => { alert("You are logged out!"); navigation.navigate("Home") }}>Logout</Text>
    <TouchableOpacity style={styles.closeButton} onPress={toggleMenu}>
      <Text style={styles.closeButtonText}>Close</Text>
    </TouchableOpacity>
  </View>
</Modal>
```

# BMI Calculator Screen



The screenshot shows a mobile app interface for a BMI calculator. At the top, there's a status bar with the time 6:59 PM, network speed 7.9KB/s, and battery level 76%. The app title "BMI Calculator" is displayed in a large, dark red font. Below the title, there are two input fields: "Weight (kg): 65" and "Height (cm): 170". A dark red button labeled "Calculate BMI" is positioned below the inputs. At the bottom, the result "Your BMI : 22.49" is shown in a dark red font. A hamburger menu icon is located in the top right corner.

```
<Text style={{ color: purple, fontSize: 50, marginBottom: 25, marginTop: 55 }}>
  BMI Calculator</Text>

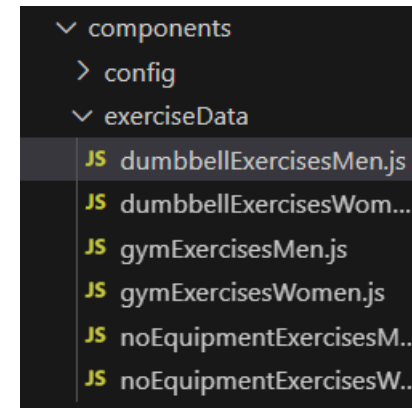
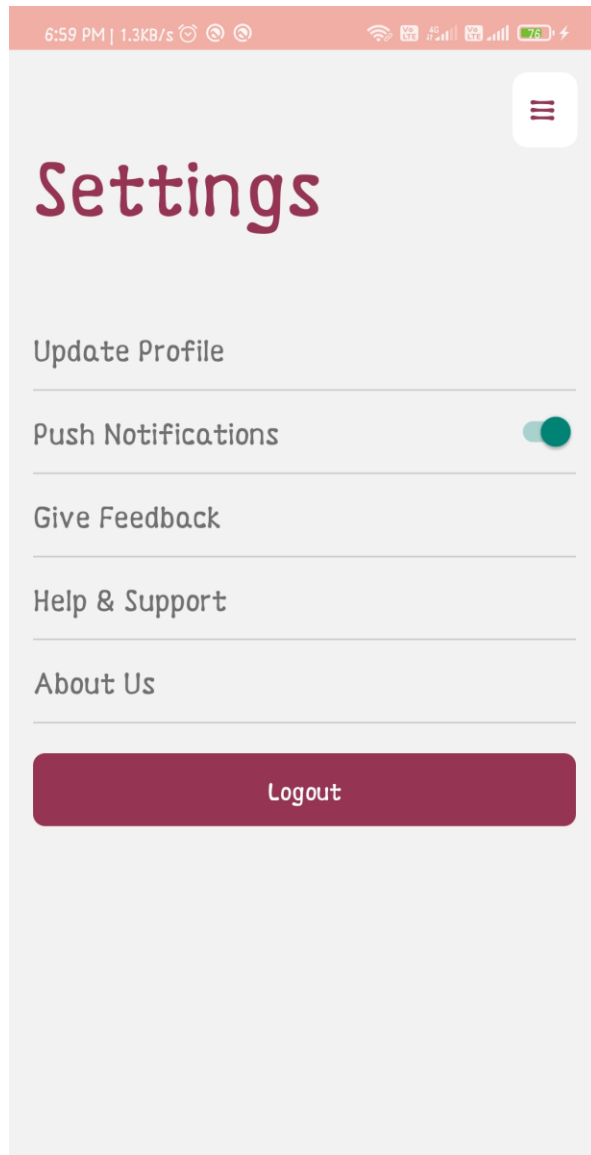
<View style={styles.inputContainer}>
  <Text style={{ fontSize: 26, }}>Weight (kg) : </Text>
  <TextInput
    style={styles.input}
    placeholder="Enter your weight"
    keyboardType="numeric"
    value={weight}
    onChangeText={({text) => setWeight(text)}}
  />
</View>
<View style={styles.inputContainer}>
  <Text style={{ fontSize: 26, }}>Height (cm) : </Text>
  <TextInput
    style={styles.input}
    placeholder="Enter your height"
    keyboardType="numeric"
    value={height}
    onChangeText={({text) => setHeight(text)}}
  />
</View>

<TouchableOpacity style={styles.calculateButton} onPress={calculateBMI}>
  <Text style={styles.calculateButtonText}>Calculate BMI</Text>
</TouchableOpacity>

{bmi !== null && (
  <View style={styles.resultContainer}>
    <Text style={{ fontSize: 28, }}>Your BMI : </Text>
    <Text style={styles.bmiText}> {bmi}</Text>
  </View>
)}
```



# Settings Screen & Exercise Data



```
export default dumbbellExercisesMen = {  
  // ... (beginner level with descriptions)  
  Beginner: {  
    warmUp: "Warm-Up: 5-7 minutes • Perform light cardio (jumping jacks, high knees, jogging in place) t  
    exercises: [  
      {  
        name: "Goblet Squats",  
        reps: "10 reps",  
        description: "Hold a dumbbell close to your chest with both hands. Perform squats, keeping your  
        path: "GobletSquats",  
        use: "Goblet squats are excellent for working your leg muscles, especially your quadriceps and g  
      },  
      {  
        name: "Dumbbell Bench Press",  
        reps: "10 reps",  
        description: "Lie on your back on a bench, holding a dumbbell in each hand at chest level. Push  
        path: "DumbbellBenchPress",  
        use: "Dumbbell bench presses target your chest muscles (pectoralis major) and also engage your t  
      },  
      {  
        name: "Bent-Over Rows",  
        reps: "10 reps",  
        description: "Bend at the hips, keeping your back straight, and hold a dumbbell in each hand. Pu  
        path: "DumbbellBentOverRows",  
        use: "Bent-over rows primarily work your upper back muscles, including the latissimus dorsi, rho  
      },  
    ],  
  },  
}
```

# Thank You