****

Lab Assignment 3

Submitted By

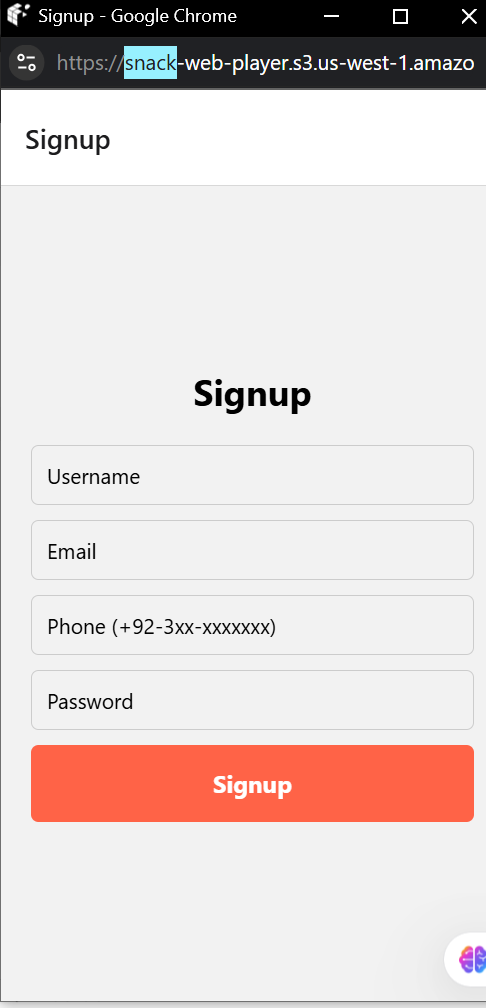
Tayyab Ali

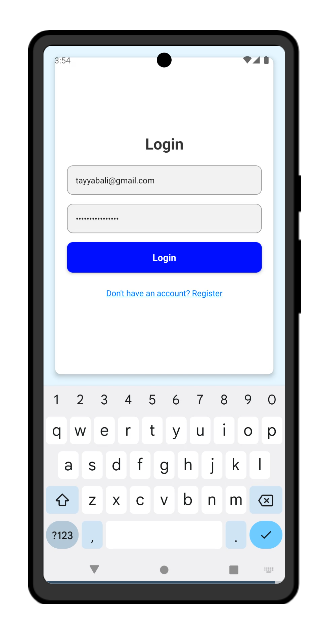
SP22-BSE-061

Submitted To

Sir Muhammad Kamran

Subject



****

**CODE**

import React from 'react';

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

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

import SignupScreen from './screens/SignupScreen';

import LoginScreen from './screens/LoginScreen';

import ProfileScreen from './screens/ProfileScreen';

const Stack = createStackNavigator();

export default function App() {

  return (

    <NavigationContainer>

      <Stack.Navigator initialRouteName="Signup">

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

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

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

      </Stack.Navigator>

    </NavigationContainer>

  );

}

**Loginscreen.js**

import React, { useState } from 'react';

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

const LoginScreen = ({ navigation }) => {

  const [email, setEmail] = useState('');

  const [password, setPassword] = useState('');

  const validateLogin = () => {

    const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

    if (!emailRegex.test(email)) {

      Alert.alert('Invalid Email', 'Please enter a valid email address.');

      return false;

    }

    if (password.length < 6) {

      Alert.alert('Invalid Password', 'Password must be at least 6 characters long.');

      return false;

    }

    return true;

  };

  const handleLogin = () => {

    if (validateLogin()) {

      Alert.alert('Success', 'Login successful!');

      navigation.navigate('Profile');

    }

  };

  return (

    <View style={styles.container}>

      <Text style={styles.title}>Login</Text>

      <TextInput

        style={styles.input}

        placeholder="Email"

        value={email}

        onChangeText={setEmail}

        keyboardType="email-address"

      />

      <TextInput

        style={styles.input}

        placeholder="Password"

        value={password}

        onChangeText={setPassword}

        secureTextEntry

      />

      <TouchableOpacity style={styles.button} onPress={handleLogin}>

        <Text style={styles.buttonText}>Login</Text>

      </TouchableOpacity>

    </View>

  );

};

const styles = StyleSheet.create({

  container: { flex: 1, justifyContent: 'center', padding: 20 },

  title: { fontSize: 24, fontWeight: 'bold', textAlign: 'center', marginBottom: 20 },

  input: { borderWidth: 1, borderColor: '#ccc', padding: 10, marginBottom: 10, borderRadius: 5 },

  button: { backgroundColor: '#ff6347', padding: 15, borderRadius: 5, alignItems: 'center' },

  buttonText: { color: '#fff', fontWeight: 'bold', fontSize: 16 },

});

export default LoginScreen;

**ProfileScreen.js**

import React from 'react';

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

const ProfileScreen = () => {

  const recentOrders = [

    {

      id: '1',

      items: 'Beef Burger, Ice Cream, Berries',

      date: '2 | Thu 10:00',

      price: '$38.15',

      image: require(), *// Replace with your image path*

    },

    {

      id: '2',

      items: 'Beef Burger, Pepperoni Moo, Cheese Bust',

      date: '15 | Fri 18:00',

      price: '$38.15',

      image: require(), *// Replace with your image path*

    },

    {

      id: '3',

      items: 'Beef Burger, Ketchup, French Fries',

      date: '7 | Mon 10:00',

      price: '$30.50',

      image: require(), *// Replace with your image path*

    },

  ];

  const renderOrderItem = ({ item }) => (

    <View style={styles.orderCard}>

      <Image source={item.image} style={styles.orderImage} />

      <View style={styles.orderDetails}>

        <Text style={styles.orderItems}>{item.items}</Text>

        <Text style={styles.orderDate}>{item.date}</Text>

      </View>

      <Text style={styles.orderPrice}>{item.price}</Text>

    </View>

  );

  return (

    <View style={styles.container}>

      {*/\* Header Section \*/*}

      <View style={styles.header}>

        <Text style={styles.headerText}>My Profile</Text>

        <TouchableOpacity>

          <Text style={styles.settingsIcon}>⚙️</Text>

        </TouchableOpacity>

      </View>

      {*/\* Profile Section \*/*}

      <View style={styles.profileCard}>

        <Image

          source={{ uri: 'https://via.placeholder.com/100' }} *// Replace with your image URL*

          style={styles.profileImage}

        />

        <Text style={styles.profileName}>Kari Rasmussen</Text>

        <View style={styles.statsRow}>

          <View style={styles.statItem}>

            <Text style={styles.statValue}>14.850</Text>

            <Text style={styles.statLabel}>Product Sells</Text>

          </View>

          <View style={styles.statItem}>

            <Text style={styles.statValue}>30bil+</Text>

            <Text style={styles.statLabel}>Earning</Text>

          </View>

        </View>

      </View>

      {*/\* Recent Orders Section \*/*}

      <View style={styles.recentOrdersHeader}>

        <Text style={styles.recentOrdersTitle}>Recent Orders</Text>

        <TouchableOpacity>

          <Text style={styles.seeAllText}>See All</Text>

        </TouchableOpacity>

      </View>

      <FlatList

        data={recentOrders}

        renderItem={renderOrderItem}

        keyExtractor={(item) => item.id}

        contentContainerStyle={styles.ordersList}

      />

    </View>

  );

};

const styles = StyleSheet.create({

  container: {

    flex: 1,

    backgroundColor: '#f9f9f9',

    padding: 20,

  },

  header: {

    flexDirection: 'row',

    justifyContent: 'space-between',

    alignItems: 'center',

    marginBottom: 20,

  },

  headerText: {

    fontSize: 24,

    fontWeight: 'bold',

  },

  settingsIcon: {

    fontSize: 20,

  },

  profileCard: {

    backgroundColor: '#fef7eb',

    borderRadius: 15,

    padding: 20,

    alignItems: 'center',

    marginBottom: 20,

  },

  profileImage: {

    width: 80,

    height: 80,

    borderRadius: 40,

    marginBottom: 10,

  },

  profileName: {

    fontSize: 18,

    fontWeight: 'bold',

    marginBottom: 20,

  },

  statsRow: {

    flexDirection: 'row',

    justifyContent: 'space-around',

    width: '100%',

  },

  statItem: {

    alignItems: 'center',

  },

  statValue: {

    fontSize: 16,

    fontWeight: 'bold',

  },

  statLabel: {

    fontSize: 12,

    color: '#666',

  },

  recentOrdersHeader: {

    flexDirection: 'row',

    justifyContent: 'space-between',

    alignItems: 'center',

    marginBottom: 10,

  },

  recentOrdersTitle: {

    fontSize: 18,

    fontWeight: 'bold',

  },

  seeAllText: {

    fontSize: 14,

    color: '#ff6347',

  },

  ordersList: {

    paddingBottom: 20,

  },

  orderCard: {

    flexDirection: 'row',

    alignItems: 'center',

    backgroundColor: '#fff',

    borderRadius: 10,

    padding: 15,

    marginBottom: 15,

    elevation: 2,

  },

  orderImage: {

    width: 50,

    height: 50,

    borderRadius: 25,

    marginRight: 10,

  },

  orderDetails: {

    flex: 1,

  },

  orderItems: {

    fontSize: 14,

    fontWeight: 'bold',

  },

  orderDate: {

    fontSize: 12,

    color: '#666',

  },

  orderPrice: {

    fontSize: 14,

    fontWeight: 'bold',

  },

});

export default ProfileScreen;

**SignUpScreen.js**

import React, { useState } from 'react';

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

const SignupScreen = ({ navigation }) => {

  const [username, setUsername] = useState('');

  const [email, setEmail] = useState('');

  const [phone, setPhone] = useState('');

  const [password, setPassword] = useState('');

  const validateSignup = () => {

    const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

    const phoneRegex = /^\+92-3\d{2}-\d{7}$/;

    const usernameRegex = /^[a-zA-Z]+$/;

    if (!usernameRegex.test(username)) {

      Alert.alert('Invalid Username', 'Username must contain only alphabets.');

      return false;

    }

    if (!emailRegex.test(email)) {

      Alert.alert('Invalid Email', 'Please enter a valid email address.');

      return false;

    }

    if (!phoneRegex.test(phone)) {

      Alert.alert('Invalid Phone', 'Phone must match +92-3xx-xxxxxxx format.');

      return false;

    }

    if (password.length < 6) {

      Alert.alert('Weak Password', 'Password must be at least 6 characters long.');

      return false;

    }

    return true;

  };

  const handleSignup = () => {

    if (validateSignup()) {

      Alert.alert('Success', 'Signup successful!');

      navigation.navigate('Login');

    }

  };

  return (

    <View style={styles.container}>

      <Text style={styles.title}>Signup</Text>

      <TextInput

        style={styles.input}

        placeholder="Username"

        value={username}

        onChangeText={setUsername}

      />

      <TextInput

        style={styles.input}

        placeholder="Email"

        value={email}

        onChangeText={setEmail}

        keyboardType="email-address"

      />

      <TextInput

        style={styles.input}

        placeholder="Phone (+92-3xx-xxxxxxx)"

        value={phone}

        onChangeText={setPhone}

        keyboardType="phone-pad"

      />

      <TextInput

        style={styles.input}

        placeholder="Password"

        value={password}

        onChangeText={setPassword}

        secureTextEntry

      />

      <TouchableOpacity style={styles.button} onPress={handleSignup}>

        <Text style={styles.buttonText}>Signup</Text>

      </TouchableOpacity>

    </View>

  );

};

const styles = StyleSheet.create({

  container: { flex: 1, justifyContent: 'center', padding: 20 },

  title: { fontSize: 24, fontWeight: 'bold', textAlign: 'center', marginBottom: 20 },

  input: { borderWidth: 1, borderColor: '#ccc', padding: 10, marginBottom: 10, borderRadius: 5 },

  button: { backgroundColor: '#ff6347', padding: 15, borderRadius: 5, alignItems: 'center' },

  buttonText: { color: '#fff', fontWeight: 'bold', fontSize: 16 },

});

export default SignupScreen;