Warisha Shuaib

Sp22-bse-018

Mobile App Development

Assignment 4

24-december-2024

Code:

```
import { View, Text, TextInput, Button, Alert, Image, FlatList, ImageBackground, StyleSheet }
from 'react-native';

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

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

import { Provider, useDispatch, useSelector } from 'react-redux';

import { configureStore, createSlice } from '@reduxjs/toolkit';


const AppStack = createStackNavigator();


const userSlice = createSlice({
 name: 'user',
 initialState: {
  name: '',
  secret: '',
  mail: '',
```

```
      contact: '',
    },
   reducers: {
    setName: (state, action) => {
     state.name = action.payload;
    },
    setSecret: (state, action) => {
     state.secret = action.payload;
    },
    setMail: (state, action) => {
     state.mail = action.payload;
    },
    setContact: (state, action) => {
     state.contact = action.payload;
    },
    clearUser: (state) => {
     state.name = '';
     state.secret = '';
     state.mail = '';
     state.contact = '';
    },
   },
});

const { setName, setSecret, setMail, setContact } = userSlice.actions;

const appStore = configureStore({
```

```
  reducer: {

    user: userSlice.reducer,

  },

});


const Wrapper = ({ children }) => (

  <ImageBackground

    source={{ uri: 'https://i.imgur.com/qhulaCG.png' }}

    style={appStyles.background}

  >

    {children}

  </ImageBackground>

);


function RegisterScreen({ navigation }) {

  const dispatch = useDispatch();

  const { name, secret, mail, contact } = useSelector((state) => state.user);


  const validateAndProceed = () => {

    const namePattern = /^[A-Za-z]+$/;

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

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


    if (secret.length < 8) {

      Alert.alert('Error', 'Password must be at least 8 characters long.');

    } else if (!namePattern.test(name)) {

      Alert.alert('Error', 'Name must contain only alphabets.');
```

```jsx
    } else if (!mailPattern.test(mail)) {

      Alert.alert('Error', 'Invalid email format.');

    } else if (!contactPattern.test(contact)) {

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

    } else {

      navigation.navigate('Login');

    }

  };


  return (

    <Wrapper>

      <View style={appStyles.container}>

        <Text style={appStyles.title}>Register</Text>

        <TextInput

          style={appStyles.input}

          placeholder="Name"

          onChangeText={(text) => dispatch(setName(text))}

          value={name}

        />

        <TextInput

          style={appStyles.input}

          placeholder="Password"

          secureTextEntry

          onChangeText={(text) => dispatch(setSecret(text))}

          value={secret}

        />

        <TextInput
```

```jsx
        style={appStyles.input}

        placeholder="Email"

        keyboardType="email-address"

        onChangeText={(text) => dispatch(setMail(text))}

        value={mail}

      />

      <TextInput

        style={appStyles.input}

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

        keyboardType="phone-pad"

        onChangeText={(text) => dispatch(setContact(text))}

        value={contact}

      />

      <Button title="Register" onPress={validateAndProceed} />

    </View>

  </Wrapper>

 );

}


function LoginScreen({ navigation }) {

 const dispatch = useDispatch();

 const { name, secret } = useSelector((state) => state.user);


 const handleUserLogin = () => {

  if (name && secret) {

   navigation.navigate('Dashboard');

  } else {
```

```
      Alert.alert('Error', 'Please enter both name and password.');

    }

  };


  return (

    <Wrapper>

      <View style={appStyles.container}>

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

        <TextInput

          style={appStyles.input}

          placeholder="Name"

          onChangeText={(text) => dispatch(setName(text))}

          value={name}

        />

        <TextInput

          style={appStyles.input}

          placeholder="Password"

          secureTextEntry

          onChangeText={(text) => dispatch(setSecret(text))}

          value={secret}

        />

        <Button title="Login" onPress={handleUserLogin} />

      </View>

    </Wrapper>

  );

}
```

```
const OrderThumbnail = ({ source }) => (

  <Image source={source} style={appStyles.orderImage} />

);


function DashboardScreen() {

  const profileDetails = {

    displayName: 'Marwa',

    profilePic: 'https://i.imgur.com/56WrMzH.jpeg',

    sales: '14,850',

    revenue: '$50B+',

  };


  const recentOrders = [

    {

      id: '1',

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

      orderDate: '21 Thu 10.00',

      cost: '$38.15',

      images: [

        'https://i.imgur.com/Ntwlus3.png',

        'https://i.imgur.com/7p0d7ZP.png',

        'https://i.imgur.com/dJOohu2.png',

      ],

    },

    {

      id: '2',

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

```
      orderDate: '15 Fri 18.00',

      cost: '$38.15',

      images: [

        'https://i.imgur.com/o6cPGYi.png',

        'https://i.imgur.com/9srqTxP.png',

        'https://i.imgur.com/mpnL9Ot.png',

      ],

    },

  ];


  return (

    <Wrapper>

      <View style={appStyles.container}>

        <View style={appStyles.profileContainer}>

          <Text style={appStyles.profileTitle}>User Dashboard</Text>

          <View style={appStyles.profileCard}>

            <Image source={{ uri: profileDetails.profilePic }} style={appStyles.avatar} />

            <Text style={appStyles.name}>{profileDetails.displayName}</Text>

          </View>

        </View>

        <Text style={appStyles.ordersTitle}>Recent Orders</Text>

        <FlatList

          data={recentOrders}

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

          renderItem={({ item }) => (

            <View style={appStyles.orderCard}>

              <View style={appStyles.imageRow}>
```

```jsx
        {item.images.map((img, index) => (

          <OrderThumbnail key={index} source={img} />

         ))}

       </View>

       <Text style={appStyles.orderTitle}>{item.items}</Text>

       <Text style={appStyles.orderDate}>{item.orderDate}</Text>

       <Text style={appStyles.orderCost}>{item.cost}</Text>

      </View>

    )}

   />

  </View>

 </Wrapper>

);
}


export default function RedevelopedApp() {
 return (
  <Provider store={appStore}>
   <NavigationContainer>
    <AppStack.Navigator initialRouteName="Register">
     <AppStack.Screen name="Register" component={RegisterScreen} />
     <AppStack.Screen name="Login" component={LoginScreen} />
     <AppStack.Screen name="Dashboard" component={DashboardScreen} />
    </AppStack.Navigator>
   </NavigationContainer>
  </Provider>
 );
```
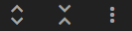
```
}

const appStyles = StyleSheet.create({
  background: {
    flex: 1,
    resizeMode: 'cover',
  },
  container: {
    flex: 1,
    padding: 20,
  },
  title: {
    fontSize: 25,
    textAlign: 'center',
    marginBottom: 20,
  },
  input: {
    borderWidth: 1,
    borderColor: '#ccc',
    borderRadius: 5,
    padding: 10,
    marginBottom: 10,
  },
  profileTitle: {
    fontSize: 18,
    fontWeight: 'bold',
  },
```

```
profileCard: {

 alignItems: 'center',

 padding: 20,

 backgroundColor: '#fff',

 borderRadius: 10,

},

avatar: {

 width: 80,

 height: 80,

 borderRadius: 40,

},

ordersTitle: {

 fontSize: 18,

 fontWeight: 'bold',

 marginBottom: 10,

},

orderCard: {

 backgroundColor: '#fff',

 padding: 10,

 borderRadius: 10,

 marginBottom: 10,

},

imageRow: {

 flexDirection: 'row',

 marginBottom: 10,

},

orderImage: {
```

```
    width: 50,

    height: 50,

    borderRadius: 7,

    marginRight: 3,

  },

  orderTitle: {

   fontSize: 19,

   fontWeight: 'bold',

  },

  orderDate: {

   fontSize: 14,

    color: '#888',

  },

  orderCost: {

   fontSize: 15,

   fontWeight: 'bold',

  },
});
```

Screenshots:

warishaShuaib + 🗑

email: "warishashuaib3747@gmail.com"
name: "Warisha"
password: "testpass1@"
phonenumber: "+92-312-5996481"

Login

warishashuaib3747@gmail.com

•••••••••

Login

Allow **Expo Go** to access this device's location?

WHILE USING THE APP

ONLY THIS TIME

DON'T ALLOW

Lo

warishashuaib3747@gmail.com

●●●●●●●●●●

Login

← Profile

Warisha Shuaib

📦 10,897 Product Sells $ 30k+ Earning

Recent Orders   See All

Beef Burger, Ice Cream, Berries
0.0004962779156327543   **540**

Beef Burger, Pepperoni Moo, Cheese Bust
0.0004962779156327543   **540**

Beef Burger, Ketchup, French Fries
0.0004962779156327543   **540**

Signup

Warisha

warishashuaib3747@gmail.com

+92-312-5996481

••••••••

Signup