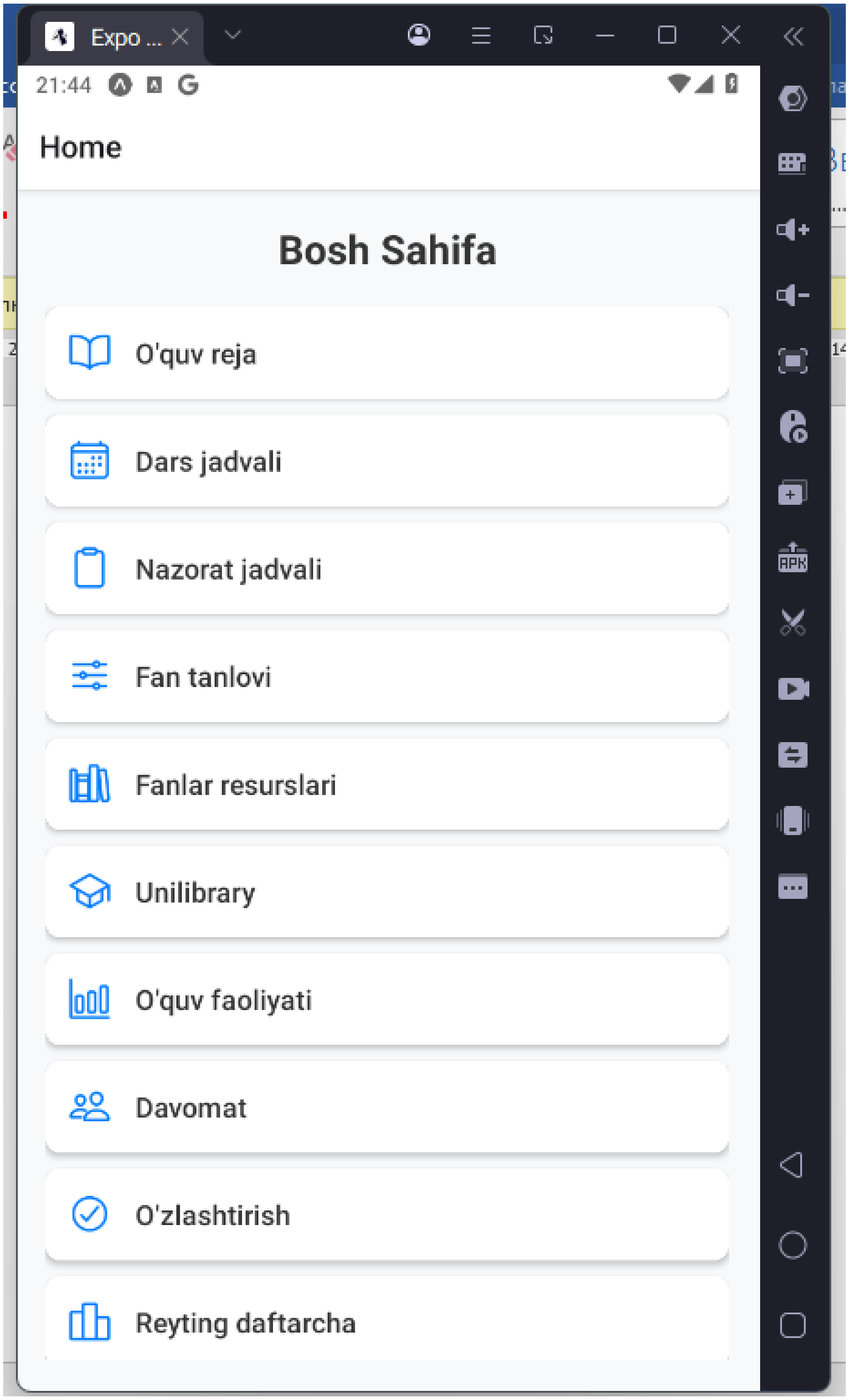
Bosh sahifa



21:44

Home

Bosh Sahifa

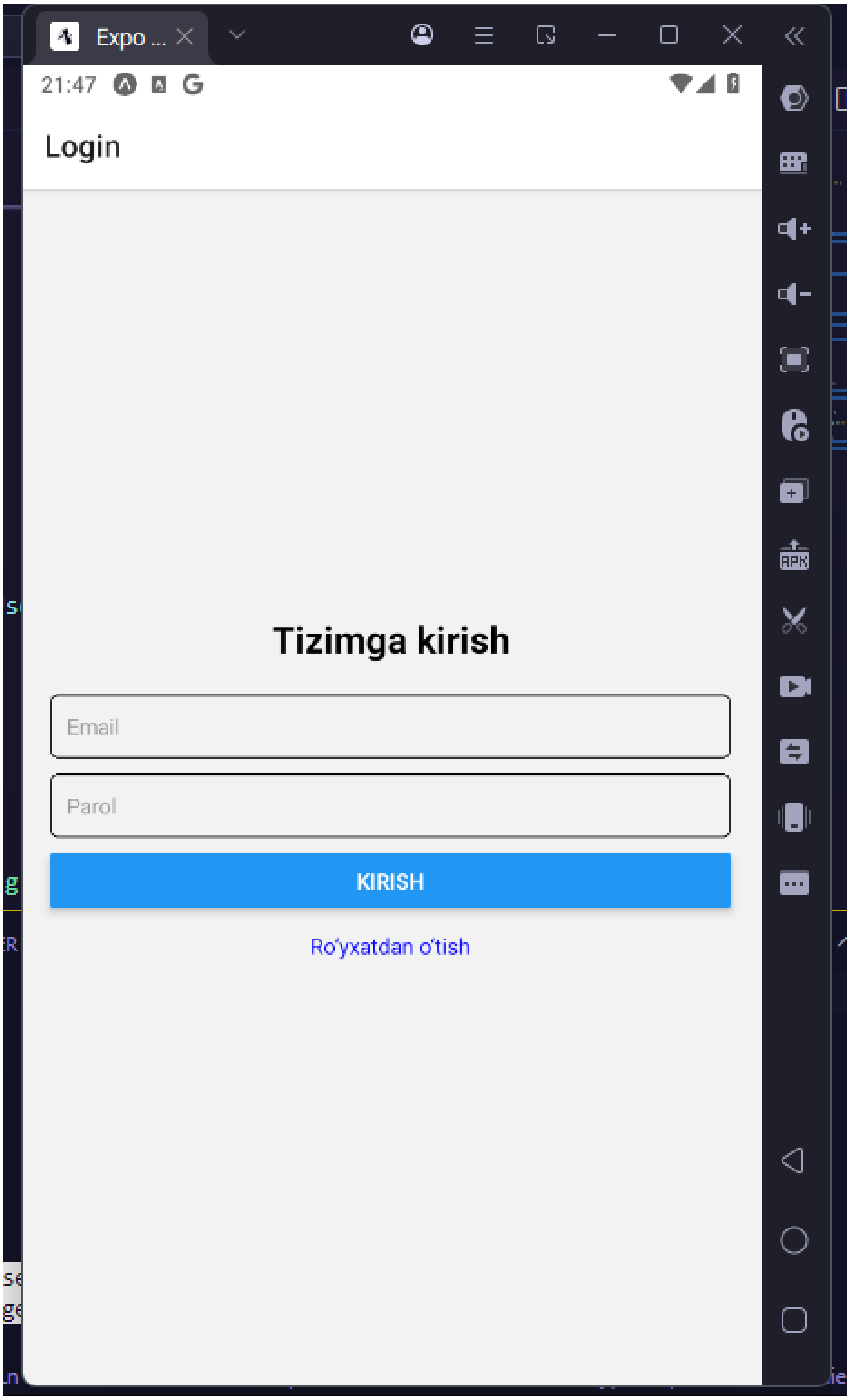
- O'quv reja
- Dars jadvali
- Nazorat jadvali
- Fan tanlovi
- Fanlar resurslari
- Unilibrary
- O'quv faoliyati
- Davomat
- O'zlashtirish
- Reyting daftarcha

Code

```
import React from "react";
import {
  View,
  Text,
  FlatList,
  TouchableOpacity,
  StyleSheet,
} from "react-native";
import { Ionicons } from "@expo/vector-icons"; // Ikonkalar uchun

type Sections = {
  id: string;
  title: string;
  icon: string;
};

// Bo‘limlar ro‘yxati
const sections: Sections[] = [
  { id: "1", title: "O'quv reja", icon: "book-outline" },
  { id: "2", title: "Dars jadvali", icon: "calendar-outline" },
  { id: "3", title: "Nazorat jadvali", icon: "clipboard-outline" },
  { id: "4", title: "Fan tanlovi", icon: "options-outline" },
  { id: "5", title: "Fanlar resurslari", icon: "library-outline" },
  { id: "6", title: "Unilibrary", icon: "school-outline" },
  { id: "7", title: "O'quv faoliyati", icon: "bar-chart-outline" },
  { id: "8", title: "Davomat", icon: "people-outline" },
  { id: "9", title: "O'zlashtirish", icon: "checkmark-circle-outline" },
  { id: "10", title: "Reyting daftarcha", icon: "podium-outline" },
  { id: "11", title: "Imtihonlar", icon: "document-text-outline" },
  { id: "12", title: "Bitiruv ishi", icon: "medal-outline" },
];

// Har bir element uchun Card
const ListItem = ({ item }: { item: { title: string; icon: string } }) => {
  return (
    <TouchableOpacity
      style={styles.card}
      onPress={() => alert(`${item.title} bo‘limi tanlandi!`)}
    >
      <Ionicons
```

```
        name={
          item.icon as
            | "book-outline"
            | "calendar-outline"
            | "clipboard-outline"
            | "options-outline"
            | "library-outline"
            | "school-outline"
            | "bar-chart-outline"
            | "people-outline"
            | "checkmark-circle-outline"
        }
        size={28}
        color="#007bff"
        style={styles.icon}
      />
      <Text style={styles.title}>{item.title}</Text>
    </TouchableOpacity>
  );
};

const HomeScreen = () => {
  return (
    <View style={styles.container}>
      <Text style={styles.header}>Bosh Sahifa</Text>
      <FlatList
        data={sections}
        keyExtractor={(item) => item.id}
        renderItem={({ item }) => <ListItem item={item} />}
        showsVerticalScrollIndicator={false}
      />
    </View>
  );
};

// 🌟 Stil
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: "#f8f9fa",
    padding: 20,
  },
  header: {
    fontSize: 26,
    fontWeight: "bold",
    textAlign: "center",
    marginBottom: 20,
    color: "#333",
  },
  card: {
    flexDirection: "row",
    alignItems: "center",
    backgroundColor: "#fff",
    padding: 15,
    marginBottom: 10,
    borderRadius: 10,
    shadowColor: "#000",
    shadowOffset: { width: 0, height: 2 },
    shadowOpacity: 0.1,
    shadowRadius: 4,
    elevation: 3,
  },
  icon: {
    marginRight: 15,
  },
  title: {
    fontSize: 18,
    fontWeight: "500",
    color: "#333",
  },
});

export default HomeScreen;
```

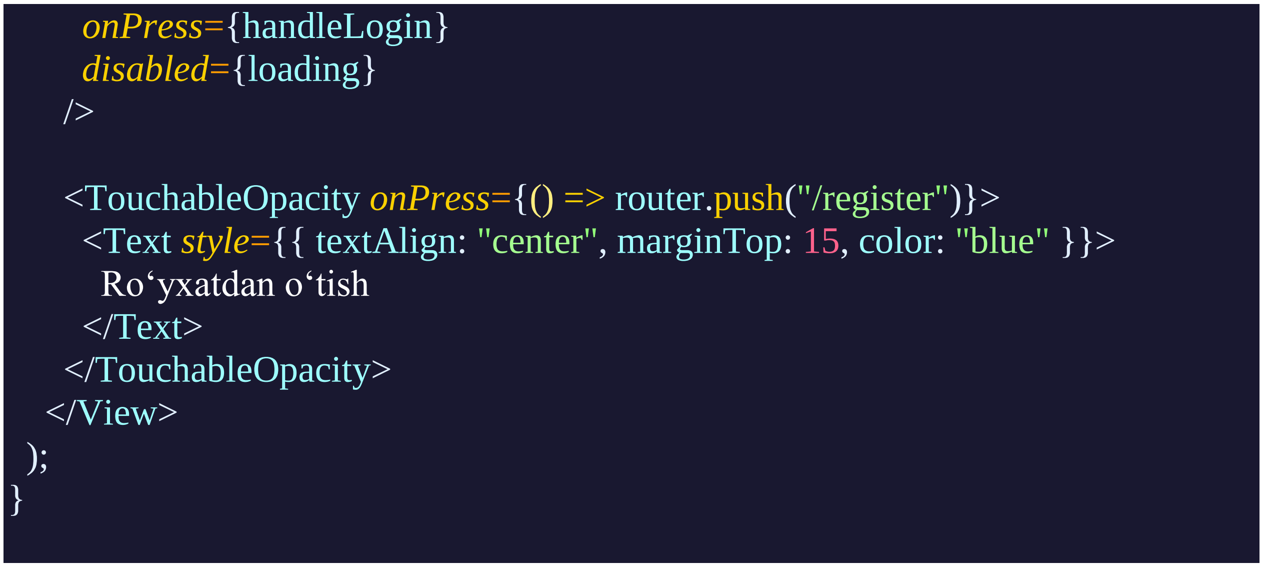
Login page


Expo ...
21:47
Login
Tizimga kirish
Email
Parol
KIRISH
Ro‘yxatdan o‘tish

Code

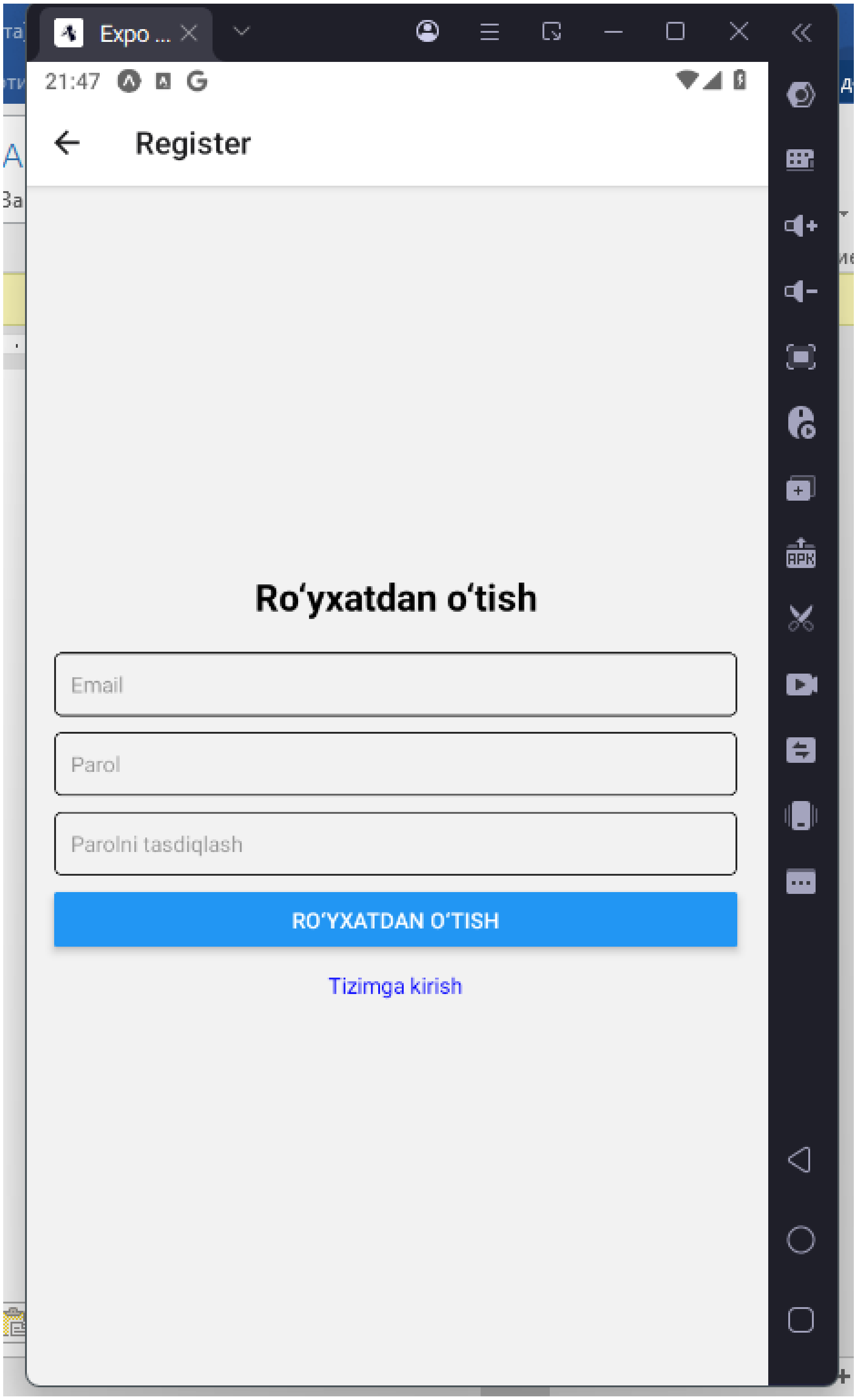
```
import { useRouter } from "expo-router";
import { useState } from "react";
import { signInWithEmailAndPassword } from "firebase/auth";
import { auth } from "../../config/firebase";
import {
  View,
  TextInput,
  Button,
  Text,
  Alert,
  TouchableOpacity,
} from "react-native";
import AsyncStorageUserFunctions from "@/utils/AsyncStorage";

export default function LoginScreen() {
  const router = useRouter();
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [loading, setLoading] = useState(false);

  const handleLogin = async () => {
    if (!email || !password) {
      Alert.alert("Xatolik", "Iltimos, email va parolni kiriting.");
      return;
    }

    setLoading(true);
    try {
      await signInWithEmailAndPassword(auth, email, password);
      AsyncStorageUserFunctions.saveUser({ email, password });
      router.replace("/(home)");
    } catch (error: any) {
      Alert.alert("Kirishda xatolik:", error.message);
    } finally {
      setLoading(false);
    }
  };

  return (
    <View style={{ flex: 1, justifyContent: "center", padding: 20 }}>
```

```
<Text
  style={{
    fontSize: 24,
    fontWeight: "bold",
    textAlign: "center",
    marginBottom: 20,
  }}
>
  Tizimga kirish
</Text>

<TextInput
  placeholder="Email"
  value={email}
  onChangeText={setEmail}
  keyboardType="email-address"
  autoCapitalize="none"
  style={{
    borderWidth: 1,
    padding: 10,
    marginBottom: 10,
    borderRadius: 5,
  }}
/>

<TextInput
  placeholder="Parol"
  value={password}
  onChangeText={setPassword}
  secureTextEntry
  style={{
    borderWidth: 1,
    padding: 10,
    marginBottom: 10,
    borderRadius: 5,
  }}
/>

<Button
  title={loading ? "Kirish..." : "Kirish"}
```



```
        onPress={handleLogin}
        disabled={loading}
      />

      <TouchableOpacity onPress={() => router.push("/register")}>
        <Text style={{ textAlign: "center", marginTop: 15, color: "blue" }}>
          Ro‘yxatdan o‘tish
        </Text>
      </TouchableOpacity>
    </View>
  );
}
```

Register page



Ro‘yxatdan o‘tish

Code

```
import { useState } from "react";
import { useRouter } from "expo-router";
import {
  View,
  TextInput,
  Button,
  Text,
  Alert,
  TouchableOpacity,
} from "react-native";
import { createUserWithEmailAndPassword } from "firebase/auth";
import { auth } from "../../config/firebase";
import AsyncStorageUserFunctions from "@/utils/AsyncStorage";

export default function RegisterScreen() {
  const router = useRouter();
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [confirmPassword, setConfirmPassword] = useState("");
  const [loading, setLoading] = useState(false);

  const handleRegister = async () => {
    if (!email || !password || !confirmPassword) {
      Alert.alert("Xatolik", "Iltimos, barcha maydonlarni to‘ldiring.");
      return;
    }

    if (password !== confirmPassword) {
      Alert.alert("Xatolik", "Parollar mos kelmadi.");
      return;
    }

    setLoading(true);
    try {
      await createUserWithEmailAndPassword(auth, email, password);
      AsyncStorageUserFunctions.saveUser({ email, password });
      router.replace("/(home)");
    } catch (error: any) {
      Alert.alert("Ro‘yxatdan o‘tishda xatolik:", error.message);
    } finally {
```

```
    setLoading(false);
  }
};

return (
  <View style={{ flex: 1, justifyContent: "center", padding: 20 }}>
    <Text
      style={{
        fontSize: 24,
        fontWeight: "bold",
        textAlign: "center",
        marginBottom: 20,
      }}
    >
      Ro‘yxatdan o‘tish
    </Text>

    <TextInput
      placeholder="Email"
      value={email}
      onChangeText={setEmail}
      keyboardType="email-address"
      autoCapitalize="none"
      style={{
        borderWidth: 1,
        padding: 10,
        marginBottom: 10,
        borderRadius: 5,
      }}
    />

    <TextInput
      placeholder="Parol"
      value={password}
      onChangeText={setPassword}
      secureTextEntry
      style={{
        borderWidth: 1,
        padding: 10,
        marginBottom: 10,
          borderRadius: 5,
        }}
      />

      <TextInput
        placeholder="Parolni tasdiqlash"
        value={confirmPassword}
        onChangeText={setConfirmPassword}
        secureTextEntry
        style={{
          borderWidth: 1,
          padding: 10,
          marginBottom: 10,
          borderRadius: 5,
        }}
      />

      <Button
        title={loading ? "Ro‘yxatdan o‘tish..." : "Ro‘yxatdan o‘tish"}
        onPress={handleRegister}
        disabled={loading}
      />

      <TouchableOpacity onPress={() => router.push("/(auth)")}>
        <Text style={{ textAlign: "center", marginTop: 15, color: "blue" }}>
          Tizimga kirish
        </Text>
      </TouchableOpacity>
    </View>
  );
}
```