Name:Qadir Nawaz Khan

Reg No:Sp22-Bse-005

Submitted To:Sir Kamran

Assignment :4

Date:24/12/2024

Mobile Application Development

Assignmnet No :4

App.js

```
import React, { useEffect } from 'react';
import { StyleSheet, View, Text, Button, ActivityIndicator } from 'react-native';
import * as Location from 'expo-location';
import AsyncStorage from '@react-native-async-storage/async-storage';
import { useDispatch, useSelector } from 'react-redux';
import { setLocation, setError, setLoading } from './redux/locationslice';
import { Provider } from 'react-redux';
import store from './redux/store';

const App = () => {
  const dispatch = useDispatch();
  const { location, errorMsg, loading } = useSelector((state) => state.location);

  const fetchLocation = async () => {
    dispatch(setLoading(true));
    try {
      let { status } = await Location.requestForegroundPermissionsAsync();
      if (status !== 'granted') {
        dispatch(setError('Permission to access location was denied'));
        return;
      }

      let currentLocation = await Location.getCurrentPositionAsync({});
      const coords = {
        latitude: currentLocation.coords.latitude,
        longitude: currentLocation.coords.longitude,
      };

      await AsyncStorage.setItem('userLocation', JSON.stringify(coords));
      dispatch(setLocation(coords));
    } catch (error) {
      dispatch(setError('Failed to fetch location'));
    }
  };

  useEffect(() => {
    fetchLocation();
  }, []);

  useEffect(() => {
    const loadLocation = async () => {
      const storedLocation = await AsyncStorage.getItem('userLocation');
      if (storedLocation) {
        dispatch(setLocation(JSON.parse(storedLocation)));
      }
    };
    loadLocation();
  }, []);
  return (
    <View style={styles.container}>
      <Text style={styles.header}>User Location</Text>
      {loading ? (
        <ActivityIndicator size="large" color="#1ecfd8" />
      ) : errorMsg ? (
        <Text style={styles.error}>{errorMsg}</Text>
      ) : (
        location && (
          <View style={styles.locationContainer}>
            <Text style={styles.text}>Latitude: {location.latitude}</Text>
            <Text style={styles.text}>Longitude: {location.longitude}</Text>
          </View>
        )
      )}
      <Button title="Refresh Location" onPress={fetchLocation} />
    </View>
  );
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    backgroundColor:'grey',
    alignItems: 'center',
    padding: 20,
  },
  header: {
    fontSize: 24,
    fontWeight: 'bold',
    marginBottom: 20,
  },
  error: {
    color: 'red',
    fontSize: 16,
    marginBottom: 20,
  },
  locationContainer: {
    marginVertical: 20,
  },
  text: {
    fontSize: 16,
  },
});

const AppWrapper = () => (
  <Provider store={store}>
    <App />
  </Provider>
);

export default AppWrapper;
```

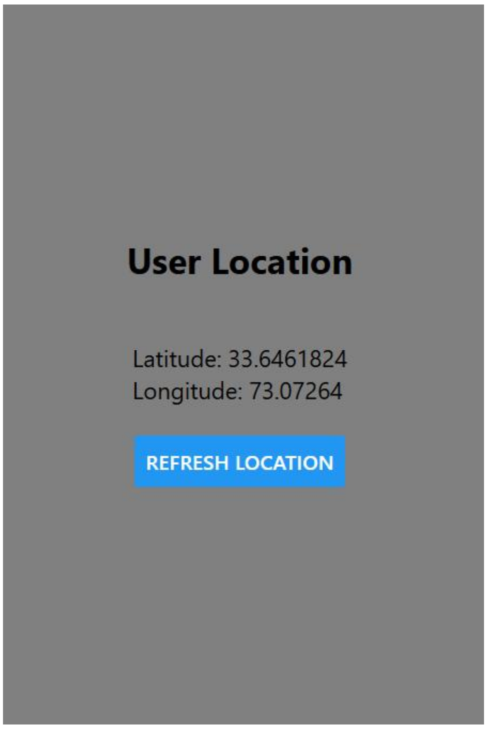
Locationslice.js

```
import { createSlice } from '@reduxjs/toolkit';

const locationSlice = createSlice({
  name: 'location',
  initialState: {
    location: null,
    errorMsg: null,
    loading: true,
  },
  reducers: {
    setLocation: (state, action) => {
      state.location = action.payload;
      state.errorMsg = null;
      state.loading = false;
    },
    setError: (state, action) => {
      state.errorMsg = action.payload;
      state.loading = false;
    },
    setLoading: (state, action) => {
      state.loading = action.payload;
    },
  },
});

export const { setLocation, setError, setLoading } = locationSlice.actions;
export default locationSlice.reducer;
```

Store.js

```
import { configureStore } from '@reduxjs/toolkit';
import locationReducer from './locationslice';
```

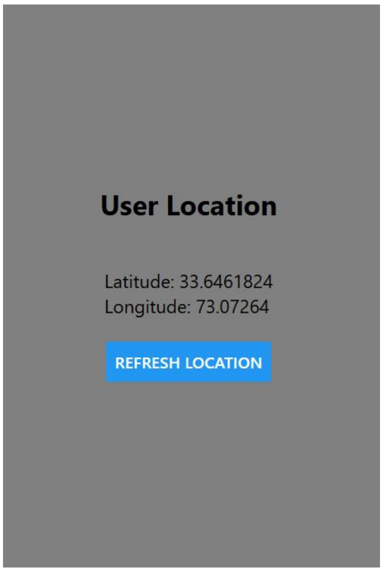
```
const store = configureStore({
  reducer: {
    location: locationReducer,
  },
});

export default store;
```



User Location

Latitude: 33.6461824
Longitude: 73.07264

Live Location

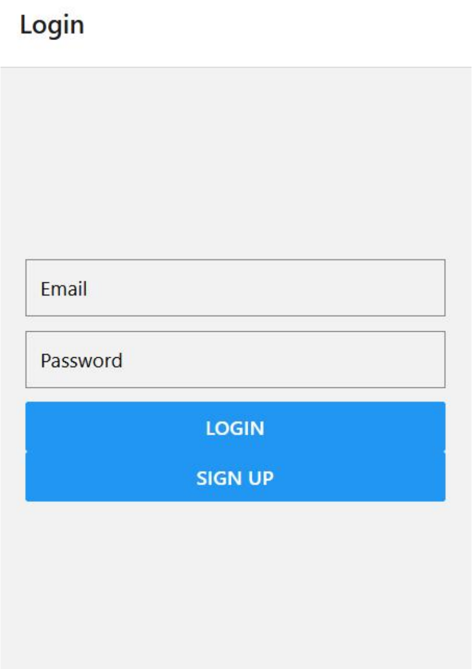


User Location

Latitude: 33.6461824
Longitude: 73.07264

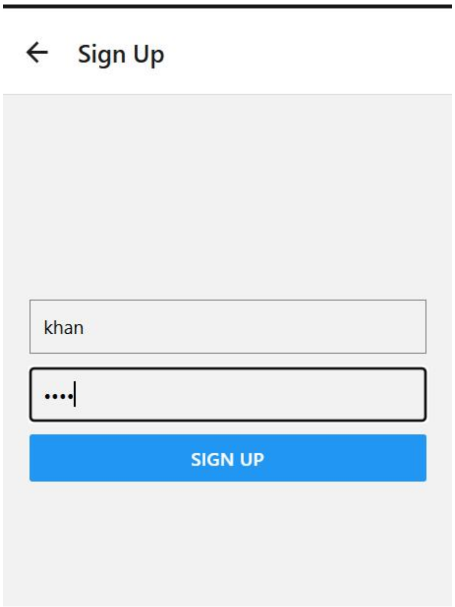
LoginScreen.js

```
import React, { useState } from 'react';
import { View, TextInput, Button, Text, StyleSheet } from 'react-native';
import { auth } from './firebaseConfig';
import { signInWithEmailAndPassword } from 'firebase/auth';
import * as Location from 'expo-location';
import AsyncStorage from '@react-native-async-storage/async-storage';

const LoginScreen = ({ navigation }) => {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [error, setError] = useState('');

  const handleLogin = async () => {
    try {

      await signInWithEmailAndPassword(auth, email, password);

      // Request location permissions
      let { status } = await Location.requestForegroundPermissionsAsync();
      if (status !== 'granted') {
        setError('Permission to access location was denied');
        return;
      }


      const location = await Location.getCurrentPositionAsync({});
      const userLocation = {
        latitude: location.coords.latitude,
        longitude: location.coords.longitude,
      };

      // Store location in AsyncStorage
      await AsyncStorage.setItem('userLocation', JSON.stringify(userLocation));

      // Navigate to Profile screen
      navigation.navigate('Profile');
    } catch (err) {
      setError(err.message);
    }
  };

  return (
    <View style={styles.container}>
      <TextInput
        style={styles.input}
        placeholder="Email"
        value={email}
        onChangeText={setEmail}
      />
      <TextInput
        style={styles.input}
        placeholder="Password"
        secureTextEntry
        value={password}
        onChangeText={setPassword}
      />
      {error && <Text style={styles.error}>{error}</Text>}
      <Button title="Login" onPress={handleLogin} />
    </View>
  );
};

const styles = StyleSheet.create({
  container: { padding: 20, flex: 1, justifyContent: 'center' },
  input: { height: 40, borderColor: 'gray', borderWidth: 1, marginBottom: 10, paddingLeft: 10 },
  error: { color: 'red', marginBottom: 10 },
});

export default LoginScreen;
```



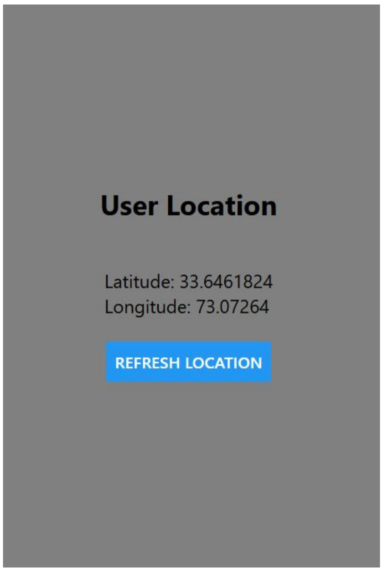
SignupScreen.js

```
import React, { useState } from 'react';
import { View, TextInput, Button, Text, StyleSheet } from 'react-native';
import { auth } from './firebaseConfig';
import { createUserWithEmailAndPassword } from 'firebase/auth';
import * as Location from 'expo-location';
import AsyncStorage from '@react-native-async-storage/async-storage';

const SignupScreen = ({ navigation }) => {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [error, setError] = useState('');

  const handleSignup = async () => {
    try {

      const userCredential = await createUserWithEmailAndPassword(auth, email,
password);

      let { status } = await Location.requestForegroundPermissionsAsync();
      if (status !== 'granted') {
        setError('Permission to access location was denied');
        return;
      }

      const location = await Location.getCurrentPositionAsync({});
      const userLocation = {
        latitude: location.coords.latitude,
        longitude: location.coords.longitude,
      };

          await AsyncStorage.setItem('userLocation', JSON.stringify(userLocation));
      await AsyncStorage.setItem('userEmail', email);


      navigation.navigate('Profile');
    } catch (err) {
      setError(err.message);
    }
  };

  return (
    <View style={styles.container}>
      <TextInput
        style={styles.input}
        placeholder="Email"
        value={email}
        onChangeText={setEmail}
      />
      <TextInput
        style={styles.input}
        placeholder="Password"
        secureTextEntry
        value={password}
        onChangeText={setPassword}
      />
      {error && <Text style={styles.error}>{error}</Text>}
      <Button title="Sign Up" onPress={handleSignup} />
    </View>
  );
};

const styles = StyleSheet.create({
  container: { padding: 20, flex: 1, justifyContent: 'center' },
  input: { height: 40, borderColor: 'gray', borderWidth: 1, marginBottom: 10,
paddingLeft: 10 },
  error: { color: 'red', marginBottom: 10 },
});

export default SignupScreen;
```



ProfileScreen.js

```
import React, { useEffect, useState } from 'react';
import { View, Text, StyleSheet } from 'react-native';
import AsyncStorage from '@react-native-async-storage/async-storage';

const ProfileScreen = () => {
  const [userLocation, setUserLocation] = useState(null);
  const [userEmail, setUserEmail] = useState(null);

  useEffect(() => {
    const loadUserData = async () => {
      const location = await AsyncStorage.getItem('userLocation');
      const email = await AsyncStorage.getItem('userEmail');
      if (location) {
        setUserLocation(JSON.parse(location));
      }
      if (email) {
        setUserEmail(email);
      }
    };
    loadUserData();
  }, []);

  return (
    <View style={styles.container}>
      {userLocation && userEmail ? (
        <>
          <Text style={styles.text}>Email: {userEmail}</Text>
          <Text style={styles.text}>Latitude: {userLocation.latitude}</Text>
          <Text style={styles.text}>Longitude: {userLocation.longitude}</Text>
        </>
      ) : (
        <Text style={styles.text}>No user data found.</Text>
      )}
    </View>
  );
};

const styles = StyleSheet.create({
  container: { padding: 20, flex: 1, justifyContent: 'center' },
  text: { fontSize: 18, marginBottom: 10 },
});

export default ProfileScreen;
```



User Location

Latitude: 33.6461824
Longitude: 73.07264

Thanks