

1. Logic Followed

app.json Configuration:

The `app.json` file is used in the Expo project to define configuration settings for the app. In this context, we are setting up the necessary configurations for integrating Facebook Login with Face ID authentication:

- **Expo Configuration:** Specifies basic metadata such as app name, version, and platform support.
- **iOS Configuration:**
 - **Bundle Identifier:** Unique identifier for iOS apps.
 - **Facebook App ID and Display Name:** Essential for Facebook Login integration, providing the app with identification and display details required by Facebook SDK.

App.js Implementation:

The `App.js` file is the main entry point of the React Native application. It handles the implementation of Facebook Login and Face ID authentication using Expo SDKs:

- **useState and useEffect Hooks:** Manage component state and handle side effects like fetching data and performing biometric authentication.
- **Facebook.useAuthRequest:** Provides hooks to initiate Facebook authentication flow and handle authentication responses.
- **LocalAuthentication.authenticateAsync:** Utilizes Expo's local authentication API to verify the user's identity using biometrics (Face ID in this case).
- **AsyncStorage:** Stores user data locally on the device for persistent login sessions.

2. Tools/SDKs Used

Tools:

- **Expo CLI:** Facilitates rapid development, testing, and deployment of React Native apps without the need for native build tools.
- **Expo:** Simplifies development, testing, and deployment of React Native apps by providing a managed workflow and various tools.
- **React:** A JavaScript library for building user interfaces. It is the foundation of React Native and provides the component-based architecture used in the app.
- **React Native:** A framework for building native apps using React. It enables writing the app in JavaScript while delivering a native app experience.
- **Node.js and npm:** Required for installing dependencies and managing packages.
- **Facebook for Developers:** Platform used to create and configure the Facebook App for integrating Facebook Login.

SDKs:

- **expo-auth-session:** Simplifies the implementation of OAuth-based authentication flows, including Facebook Login.
- **expo-local-authentication:** Provides support for biometric authentication methods such as Face ID on iOS.
- **@react-native-async-storage/async-storage:** Enables secure storage of key-value pairs locally on the device.

3. Limitations of the Solution

- **Dependency on Facebook SDK:** Users must have a Facebook account and grant necessary permissions for the app to access their profile information.
- **Platform-Specific Features:** Face ID authentication is limited to iOS devices with Face ID capability.

4. Areas of Improvement

- **Enhanced User Experience:** Improve UI/UX design for a more intuitive and seamless login experience.
- **Error Handling:** Implement robust error handling mechanisms to manage various edge cases during the authentication process.
- **Performance Optimization:** Optimize data fetching and storage operations to enhance app responsiveness and efficiency.
- **Security Enhancements:** Enhance data security by implementing encryption methods for storing sensitive user information locally.

5. Data Privacy Declarations

- **Data Collection:** The app collects user's name and email from their Facebook profile upon authentication for login purposes.
- **Data Usage:** Collected data is used solely for authentication within the app and is not shared with any third parties.
- **Storage:** User data is stored locally on the device using AsyncStorage, ensuring data privacy and compliance with applicable data protection regulations.
- **Permissions:** Users are required to grant necessary permissions for the app to access their Facebook profile information. The app adheres to Facebook's data policies regarding the use and handling of user data.