📘 Assignment: Biometric Authentication Flow in Flutter

🎯 Objective:

Build a Flutter app that demonstrates biometric authentication using **Flutter packages** with BLoC pattern. The app should support both **Face ID Android(Compulsory) and (iOS)(Optional)** and **Fingerprint (Android & iOS)** authentication methods and navigate to a **Dashboard screen** upon successful authentication.

Please do the assignment by Monday(14th-July)

The assignment is given to understand how the developer implements functionality with **quality and clarity**. The focus is on:

- **Code structure**
- **Architecture decisions**
- **Error handling**
- **Best practices**

📋 Requirements:

1. UI Design: Authentication Screen

Create a visually appealing authentication screen with:

- **Title**: "Secure Login"
- **Two buttons**:
    - `Unlock with Face ID`
    - `Unlock with Biometric Fingerprint`
- Use icons (`Icons.face`, `Icons.fingerprint`) appropriately
- Ensure UI is responsive and styled for both Android and iOS

2. Biometric Authentication Logic

Use `Flutter package` available or any updated alternative to:

- Check device support for biometric authentication
- Handle Face ID on iOS and Fingerprint on supported platforms
- Show appropriate error or fallback messages if not supported continue as guest

3.Flutter BLoC Implementation

- Define Events: `FaceIdRequested`, `FingerprintRequested or other`
- Define States: `AuthInitial`, `AuthLoading`, `AuthSuccess`, `AuthFailure or other`
- Implement full BLoC pattern with separation of concerns

4. Navigation

- After successful authentication, navigate to a `DashboardScreen` that simply shows:

-
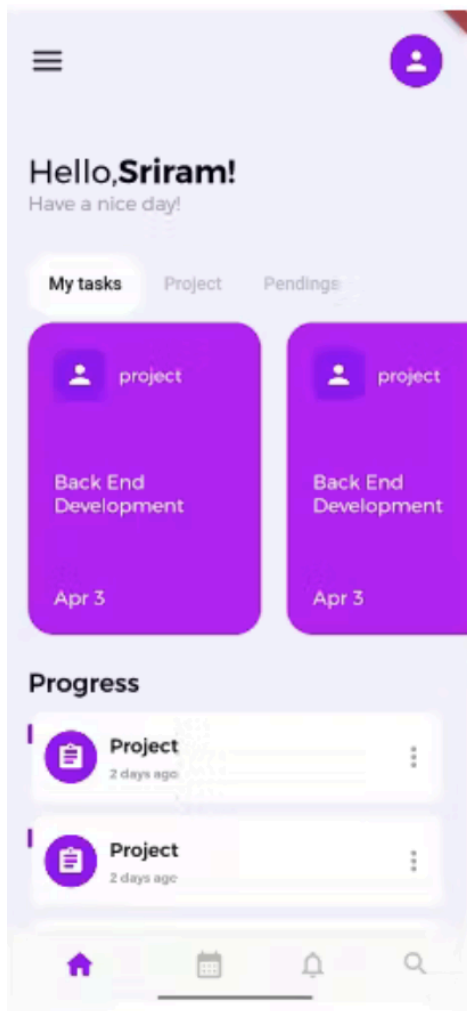
5. Project Structure
Follow a modular architecture:
css
CopyEdit
```
lib/
├── blocs/
```

```
├── screens/
├── widgets/
├── services/
└── main.dart
```

-

6. Extra Credit (Optional)

- Use animations for button taps and transitions
- Reusability of components
- The Assignment Submission time will be considered
- The Design should be Responsive and supported for mobile and tablets

---

✅ Evaluation Criteria:

- Correct use of BLoC
- Clean and scalable Flutter architecture
- Code readability and best practices
- Platform compatibility (FaceID on iOS, fingerprint on Android)
- UI/UX design quality