HOW TO RUN THE PROJECT

To run a project that implements the described multi-layered authentication and voice-controlled transaction system, you need both hardware and software components configured and functioning together. Here's a breakdown of what's required and how to run the project:

System Requirements

Hardware:

1. Android Device – Must support:
    - Microphone
    - Fingerprint sensor
    - Text-to-speech (TTS)
2. RFID Reader – Connected via:
    - USB OTG or
    - Bluetooth
3. RFID Tags – Pre-registered in your database

Software Requirements

- Android Studio (latest version)
- Java/Kotlin (preferred language for Android)
- Android SDK tools
- RFID reader SDK/API (based on manufacturer)
- Firebase or SQLite – For storing voice PINs, RFID data, fingerprints (if local)
- Permissions Required:
    - RECORD_AUDIO
    - USE_BIOMETRIC
    - INTERNET (for voice recognition APIs)

- USB_PERMISSION or BLUETOOTH (for RFID)

Steps to Run the Project

1. Setup Project in Android Studio

- Clone or import your Android project.
- Add necessary dependencies in build.gradle:
  - Speech recognition
  - Fingerprint authentication
  - Text-to-speech
  - RFID SDK

2. Grant All Permissions

- Ask for runtime permissions in the app.
- Configure AndroidManifest.xml for:

xml

```
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.USE_BIOMETRIC"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.USB_PERMISSION"/>
```

3. Voice-Based PIN Verification

- Use SpeechRecognizer API to capture and convert spoken input to text.
- Compare it with stored PIN:

java

```
if(recognizedText.equals(storedPin)){
   proceedToRFID();
}
```

4. RFID Authentication

- Connect RFID reader (check connection: USB or Bluetooth).

- Use SDK to read UID.
- Compare with database (local or cloud).

5. Fingerprint Authentication

- Use BiometricPrompt for fingerprint verification:

kotlin

```
val biometricPrompt = BiometricPrompt(...)
biometricPrompt.authenticate(...)
```

6. Voice-Based Shop Name Input

- Prompt user using TextToSpeech or UI.
- Use SpeechRecognizer again to capture the shop name.

7. Voice-Based Payment Input

- Same as shop name, but parse spoken numbers:

java

```
Double amount = Double.parseDouble(recognizedText);
```

8. Voice Confirmation

- Use TextToSpeech to confirm:

java

```
tts.speak("Payment of " + amount + " to " + shopName + " successful", ...)
```

Testing Flow

1. Launch app
2. Speak the PIN
3. Tap RFID tag
4. Scan fingerprint
5. Speak shop name
6. Speak amount
7. Receive voice confirmation