Flutter Transaction SMS Filter App - Complete Setup Guide

Prerequisites

1. **Install Flutter SDK**
   - Download Flutter from flutter.dev
   - Add Flutter to your system PATH
   - Run `flutter doctor` to verify installation
2. **Install Android Studio or VS Code**
   - Android Studio (recommended for beginners)
   - VS Code with Flutter extension
3. **Set up Android Device/Emulator**
   - Physical Android device with USB debugging enabled
   - OR Android emulator through Android Studio

Step-by-Step Setup

Step 1: Create Flutter Project

```bash
flutter create transaction_sms_app
cd transaction_sms_app
```

Step 2: Update pubspec.yaml

Replace the dependencies section in `pubspec.yaml`:

```yaml
dependencies:
  flutter:
    sdk: flutter
  permission_handler: ^10.4.3
  telephony: ^0.2.0
  cupertino_icons: ^1.0.2

dev_dependencies:
  flutter_test:
    sdk: flutter
  flutter_lints: ^2.0.0
```

Step 3: Update Android Permissions

Add these permissions to `android/app/src/main/AndroidManifest.xml` inside the `<manifest>` tag:

```xml
<uses-permission android:name="android.permission.READ_SMS" />
<uses-permission android:name="android.permission.RECEIVE_SMS" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
```

Step 4: Replace main.dart

Replace the content of `lib/main.dart` with the provided Flutter code.

Step 5: Install Dependencies

```bash
flutter pub get
```

Running the App

For Physical Android Device:

1. **Enable Developer Options:**
   - Go to Settings > About Phone
   - Tap "Build Number" 7 times
   - Go back to Settings > Developer Options
   - Enable "USB Debugging"
2. **Connect Device:**
   - Connect phone via USB
   - Allow USB debugging when prompted
   - Run: `flutter devices` to verify connection
3. **Run the App:**

   ```bash
   flutter run
   ```

For Android Emulator:

1. **Create Emulator in Android Studio:**
    - Open Android Studio
    - Tools > AVD Manager
    - Create Virtual Device
    - Choose a device definition and system image
    - Start the emulator
2. **Run the App:**

    bash

    ```
    flutter run
    ```

App Features

What the App Does:

- **Scans SMS messages** for transaction-related content
- **Filters messages** from known banks and payment gateways
- **Displays filtered messages** in a clean, organized list
- **Shows message details** when tapped
- **Refreshes automatically** with pull-to-refresh

Transaction Detection Logic:

The app identifies transaction messages by:

- **Sender verification**: Checks against known bank/payment gateway codes
- **Keyword matching**: Looks for words like "debited", "credited", "payment", etc.
- **Currency patterns**: Detects currency symbols (₹, $) and amounts

Troubleshooting

Common Issues:

1. **Permission Denied Error:**
   - Ensure SMS permission is granted in phone settings
   - Go to Settings > Apps > Your App > Permissions > Enable SMS
2. **App Not Installing:**
   - Check if USB debugging is enabled
   - Try `flutter clean` then `flutter pub get`
3. **No Messages Showing:**
   - Verify you have transaction SMS in your inbox
   - Check if app has SMS reading permission
   - Try refreshing the app
4. **Build Errors:**
   - Run `flutter doctor` to check for issues
   - Ensure all dependencies are properly installed
   - Check Android SDK is up to date

Testing the App:

1. **Grant Permissions:**
   - When first opened, grant SMS reading permission
   - If denied, go to phone Settings > Apps > Permissions
2. **Verify Functionality:**
   - Check if your bank SMS appear in the app
   - Test with different transaction messages
   - Use pull-to-refresh to update messages

Security & Privacy Notes

⚠️ **Important Considerations:**

- This app reads SMS messages from your device
- Only use on your own device
- Be cautious about sharing or distributing the app
- SMS access is a sensitive permission
- Consider adding additional security measures for production use

Customization Options

Adding More Banks/Gateways:

Modify the `bankSenders` list in the code:

```dart
final List<String> bankSenders = [
  'SBIINB', 'HDFCBK', 'ICICIB', 'KOTAKBK', 'AXISBK',
  'PAYTM', 'GPAY', 'PHONEPE', 'AMAZONPAY', 'FREECHARGE',
  // Add your bank codes here
  'YOURBANK', 'CUSTOMCODE'
];
```

Adding Keywords:

Modify the `transactionKeywords` list:

```dart
final List<String> transactionKeywords = [
  'debited', 'credited', 'transaction', 'payment', 'transfer',
  // Add custom keywords
  'purchase', 'refund', 'cashback'
];
```

Building Release APK

To create an installable APK:

```bash
flutter build apk --release
```

The APK will be created in `build/app/outputs/flutter-apk/app-release.apk`

Next Steps

- Test thoroughly with your transaction messages
- Consider adding data export functionality
- Implement message categorization (debit/credit)
- Add search and filter capabilities
- Consider adding backup/restore features

Need Help?

- Check Flutter documentation: docs.flutter.dev
- Android permissions guide: developer.android.com
- For SMS handling: pub.dev/packages/telephony