Firebase Cloud Messaging (FCM) Implementation Guide

1. Add Dependencies

Add to `pubspec.yaml`:

```
dependencies:
  firebase_messaging: ^14.7.10
```

2. Android Setup

Add to `android/app/src/main/AndroidManifest.xml` inside `<application>`:

```
<meta-data
    android:name="com.google.firebase.messaging.default_notification_channel_id"
    android:value="high_importance_channel" />
```

3. iOS Setup

Add to `ios/Runner/Info.plist`:

```
<key>FirebaseAppDelegateProxyEnabled</key>
<false/>
```

4. Implementation Steps

A. Create Notification Service

Create `lib/core/services/notification_service.dart`:

```
import 'package:firebase_messaging/firebase_messaging.dart';
import 'package:flutter/material.dart';

class NotificationService {
  final FirebaseMessaging _firebaseMessaging = FirebaseMessaging.instance;

  Future<void> initialize() async {
    // Request permission for iOS
    NotificationSettings settings = await _firebaseMessaging.requestPermission(
      alert: true,
      badge: true,
      sound: true,
    );

    // Get FCM token
    String? token = await _firebaseMessaging.getToken();
    debugPrint('FCM Token: $token');

    // Handle foreground messages
    FirebaseMessaging.onMessage.listen((RemoteMessage message) {
      debugPrint('Got a message whilst in the foreground!');
      debugPrint('Message data: ${message.data}');

      if (message.notification != null) {
        debugPrint('Message also contained a notification: ${message.notification}');
      }
    });

    // Handle background messages
    FirebaseMessaging.onBackgroundMessage(_firebaseBackgroundHandler);

    // Handle notification taps
    FirebaseMessaging.onMessageOpenedApp.listen((RemoteMessage message) {
      debugPrint('Message opened from background: ${message.data}');
    });
  }

  Future<void> subscribeToTopic(String topic) async {
    await _firebaseMessaging.subscribeToTopic(topic);
  }

  Future<void> unsubscribeFromTopic(String topic) async {
    await _firebaseMessaging.unsubscribeFromTopic(topic);
  }
}

@pragma('vm:entry-point')
Future<void> _firebaseBackgroundHandler(RemoteMessage message) async {
  debugPrint('Handling a background message: ${message.messageId}');
}
```

B. Add to Dependency Injection

In `lib/core/services/get_it_services.dart`:

```
import 'package:costly/core/services/notification_service.dart';

void setupGetIt() {
  // Add this line in the services section
  getIt.registerLazySingleton<NotificationService>(
    () => NotificationService(),
    );
  }
```

C. Initialize in Main

In `lib/main.dart`:

```
void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  // ... other initialization code ...

  setupGetIt();

  // Initialize notification service
  await getIt<NotificationService>().initialize();

  runApp(const MyApp());
}
```

5. Testing

1. Run `flutter pub get`
2. Get FCM token from logs
3. Use Firebase Console or Postman to send test notifications

6. Common Issues

- Android: Ensure Google Services JSON is properly configured
- iOS: Check provisioning profiles and capabilities
- Background messages: Verify proper channel setup
- Token issues: Check Firebase project configuration

7. Additional Resources

- Firebase Cloud Messaging Documentation
- Flutter FCM Plugin Documentation