Flutter Push Notifications: Firebase Cloud Messaging (FCM) Basics

Subtitle: Learn how to send and receive push notifications in Flutter apps using Firebase Cloud Messaging.

Website Name: haas.dev
Website Link: https://dev-roast-app.vercel.app

Introduction

Push notifications keep users **engaged** and informed. Flutter, combined with **Firebase Cloud Messaging (FCM)**, makes it easy to send notifications to mobile apps. This guide covers setup, receiving, and handling notifications.

Step 1: Add Firebase Messaging Dependency

```
dependencies:
  firebase_messaging: ^14.7.3
```

Run:

```
flutter pub get
```

Import in Dart:

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

Step 2: Initialize Firebase Messaging

```
FirebaseMessaging messaging = FirebaseMessaging.instance;

// Request permission (iOS)
NotificationSettings settings = await messaging.requestPermission(
  alert: true,
  badge: true,
  sound: true,
);
print('User granted permission: ${settings.authorizationStatus}');
```

Step 3: Get Device Token

```
String? token = await FirebaseMessaging.instance.getToken();
print("Device Token: $token");
```

- Device token identifies the device for push notifications
- Save this token in Firestore if sending targeted notifications

---

Step 4: Listen to Foreground Messages

```
FirebaseMessaging.onMessage.listen((RemoteMessage message) {
  print('Received message: ${message.notification?.title}');
});
```

Exercise: Display a SnackBar when a notification arrives while app is open.

Step 5: Background & Terminated Messages

```
FirebaseMessaging.onBackgroundMessage(_firebaseMessagingBackgroundHandler);

Future<void> _firebaseMessagingBackgroundHandler(RemoteMessage message) async {
  print("Handling a background message: ${message.messageId}");
}
```

- Register the handler in `main()` before `runApp()`
- Handles messages when app is in background or terminated

Step 6: Sending Notifications

- Use Firebase Console → Cloud Messaging → Send test message
- Or use server-side code (Node.js, Python, etc.) with device token

Exercise: Send a test notification to your device using Firebase Console.

Step 7: Mini Project

Build a **Flutter Notification App**:

- Initialize FCM in app
- Display incoming notifications in a SnackBar
- Save device token to Firestore
- Test notifications via Firebase Console

Key Takeaways

- FCM integrates seamlessly with Flutter for push notifications
- Foreground, background, and terminated messages require different handling
- Device tokens are essential for targeted notifications
- Push notifications **increase user engagement and retention**

Ignoring push notifications makes apps **less interactive and engaging**, missing a key feature expected in modern apps.

Visit **haas.dev** for more Flutter guides, tutorials, and complete project resources.

Website Name: haas.dev
Website Link: https://dev-roast-app.vercel.app
