Firebase Cloud Messaging (FCM) is a cross-platform messaging solution that lets you reliably send messages at no cost.1 Here's a breakdown of how FCM push notifications work:

**Core Components:**

1. **Client App:** The application installed on the user's device (Android, iOS, or Web) that is registered to receive FCM messages.
2. **App Server (or Trusted Environment):** Your backend server (or a cloud-based solution like Cloud Functions for Firebase) where you build and send message requests.2 This is where your business logic resides, determining when and what messages to send.
3. **FCM Backend:** Google's cloud service that acts as a mediator between your app server and the client apps.3 It handles message routing, delivery, and various platform-specific considerations.4
4. **Platform-specific Transport Layer:** This includes services like Android Transport Layer (ATL) for Android, Apple Push Notification service (APNs) for iOS, and the Web Push protocol for web apps.5 These are responsible for the actual delivery of messages to the respective devices.

**How the Process Works (Lifecycle Flow):**

1. **Device Registration & Token Generation:**
   * When your app is installed and launched on a user's device, the FCM SDK within the app automatically registers with the FCM backend.6
   * Upon successful registration, FCM generates a unique **FCM Registration Token** (also known as a device token) for that specific app instance on that device.7
   * This token acts as the address for sending messages to that particular device.
   * Your client app typically sends this FCM token to your app server, where it's stored (e.g., in a database) for future use.8
2. **Sending a Message from Your App Server:**
   * When you want to send a push notification (e.g., a new message, an update, a promotional offer), your app server composes a message payload.9
   * This payload can contain two main types of messages:
     + **Notification messages:** These are "display messages" handled automatically by the FCM SDK.10 They have predefined user-visible keys (title, body, icon) and can include an optional data payload.11 When the app is in the background, FCM displays them in the notification tray. In the foreground, the app's code determines the behavior.12
     + **Data messages:** These are entirely handled by your client app's code.13 They contain only custom key-value pairs and are delivered silently to the app, allowing you to process the message logic yourself (e.g., sync data in the background).14
     + You can also send messages with both notification and data payloads.15
   * Your app server then sends this message request, along with the target FCM tokens (or topics for group messaging), to the FCM backend using the Firebase Admin SDK or the FCM HTTP v1 API.16
3. **FCM Backend Processing:**
   * The FCM backend receives your message request.17
   * It performs actions like generating a message ID, distributing messages to devices subscribed to specific topics (if applicable), and preparing the message for delivery.18
   * It then forwards the message to the appropriate platform-specific transport layer (ATL for Android, APNs for iOS, Web Push for web).19
4. **Delivery to the Device:**
   * The platform-specific transport layer takes over. If the device is online, the message is sent directly to the device.
   * These transport layers are highly optimized to ensure efficient and reliable delivery, even handling scenarios like devices being in "Doze mode" (Android) or managing persistent connections.20
   * If a device is offline, FCM can queue messages and deliver them when the device comes back online (up to a certain time-to-live, typically 28 days).21
5. **Receiving and Handling on the Client Device:**
   * The FCM SDK on the client device receives the message.22
   * **For Notification Messages:**
     + If the app is in the **background or killed**, the FCM SDK automatically displays the notification in the device's notification tray.
     + If the app is in the **foreground**, the onMessageReceived() callback method in your app is triggered, and you can programmatically decide how to handle and display the notification (or not display it at all).23
   * **For Data Messages:**
     + Regardless of the app's state (foreground, background, or killed), data messages are always handled by the onMessageReceived() method (or a similar callback on iOS/web).24 Your app's code then processes the custom data within the message.25

**Key Concepts:**

* **FCM Token:** The unique identifier for a specific app instance on a specific device, crucial for targeting messages.26
* **Notification Messages vs. Data Messages:** Understanding the difference allows you to control how messages are handled by the client app (automatically displayed vs. app-controlled).
* **Topics:** A way to send messages to multiple devices that have subscribed to a particular topic, useful for broadcasting messages to segments of your users.27
* **Delivery Options:** FCM offers various delivery options like "collapsible" messages (only the latest message is delivered if multiple are sent before the device comes online) and different priority levels (normal, high) for time-sensitive content.28

In essence, FCM acts as a robust and scalable intermediary, simplifying the complex process of delivering push notifications to your users across various platforms.29