

## LẬP TRÌNH ĐA NỀN TẢNG 2

BÀI 8: GIỚI THIỆU VỀ FCM TRONG FIREBASE  
VÀ TÙY CHỈNH NOTIFICATION

PHẦN 1: GIỚI THIỆU VỀ FCM TRONG  
FIREBASE

☐ Giới thiệu về FCM

☐ Cài đặt thư viện

## **Firestore Cloud Messaging**

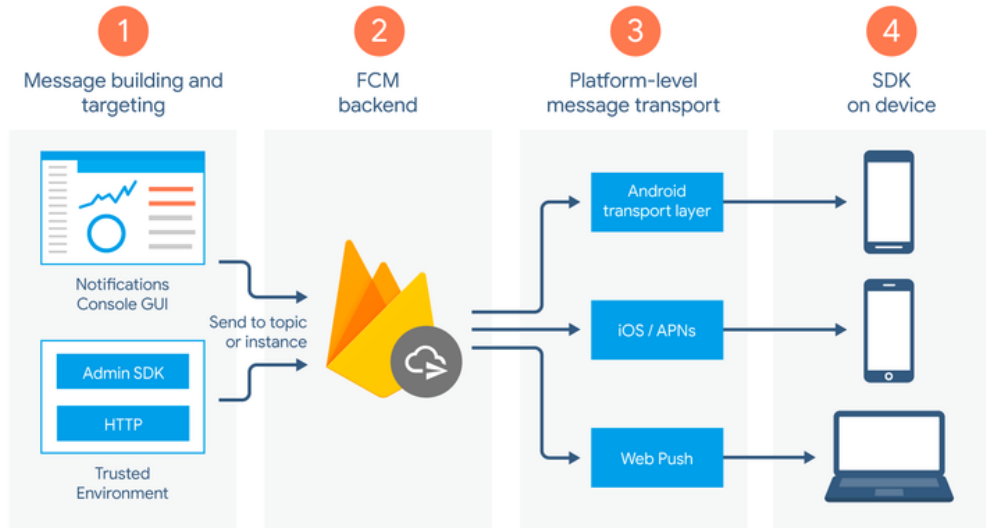
**Firestore Cloud Messaging (FCM)** là một giải pháp nhắn tin đa nền tảng cho phép bạn gửi tin nhắn miễn phí một cách đáng tin cậy.

Sử dụng FCM, bạn có thể thông báo cho ứng dụng khách rằng email mới hoặc dữ liệu khác có sẵn để đồng bộ hóa. Đối với các trường hợp sử dụng như nhắn tin tức thời, một tin nhắn có thể chuyển tải trọng lên đến 4000 byte sang ứng dụng khách.

## □ Tổng quan kiến trúc FCM

FCM dựa trên tập hợp các thành phần sau đây để xây dựng, vận chuyển và nhận thông điệp:

1. Công cụ để soạn hoặc xây dựng yêu cầu tin nhắn. Trình soạn thảo thông báo cung cấp tùy chọn dựa trên GUI để tạo yêu cầu thông báo. Để tự động hóa hoàn toàn và hỗ trợ cho tất cả các loại tin nhắn, bạn phải tạo yêu cầu tin nhắn trong môi trường máy chủ đáng tin cậy hỗ trợ SDK quản trị Firebase hoặc giao thức máy chủ FCM. Môi trường này có thể là **Cloud Functions** cho Firebase, **App Engine** hoặc máy chủ ứng dụng của riêng bạn.



2. FCM backend, (trong số các chức năng khác) chấp nhận các yêu cầu tin nhắn, thực hiện phân xuất tin nhắn thông qua các chủ đề và tạo siêu dữ liệu tin nhắn như ID tin nhắn.

3. Lớp truyền tải cấp nền tảng, định tuyến tin nhắn đến thiết bị được nhắm mục tiêu, xử lý việc gửi tin nhắn và áp dụng cấu hình dành riêng cho nền tảng khi thích hợp. Lớp vận chuyển này bao gồm:

- Lớp truyền tải Android (ATL) dành cho thiết bị Android có dịch vụ của Google Play.
- Dịch vụ Thông báo đẩy của Apple (APN) cho các thiết bị Apple

- Giao thức push web cho các ứng dụng web

4. **FCM SDK** trên thiết bị của người dùng, nơi thông báo được hiển thị hoặc tin nhắn được xử lý theo trạng thái foreground/background của ứng dụng và bất kỳ logic ứng dụng có

liên quan nào.



Lifecycle flow

- **Đăng ký thiết bị để nhận tin nhắn từ FCM.** Một phiên bản của ứng dụng khách đăng ký nhận tin nhắn, nhận token thông báo đăng ký xác định duy nhất phiên bản ứng dụng.

- **Gửi và nhận tin nhắn**

- Tin nhắn được soạn, trong Notifications composer hoặc môi trường đăng tin cậy và yêu cầu tin nhắn được gửi đến FCM backend.
- FCM backend nhận yêu cầu tin nhắn, tạo ID tin nhắn và siêu dữ liệu khác và gửi nó đến lớp truyền tải cụ thể của nền tảng.
- Khi thiết bị trực tuyến, tin nhắn được gửi qua lớp truyền tải dành riêng cho nền tảng đến thiết bị.
- Trên thiết bị, ứng dụng khách sẽ nhận được tin nhắn hoặc thông báo.



- Khi một **notification messages** được gửi đi, user sẽ nhận được một object data như dưới đây:

```
{
  "message":{
    "token":"bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
    "notification":{
      "title":"Portugal vs. Denmark",
      "body":"great match!"
    }
  }
}
```

Notification messages được gửi đến mục thông báo khi ứng dụng ở chế độ background. Đối với các ứng dụng ở foreground, tin nhắn được xử lý bằng callback function.

- Đặt khóa thích hợp với các cặp key-value trị tùy chỉnh của bạn để gửi data payload đến ứng dụng khách.

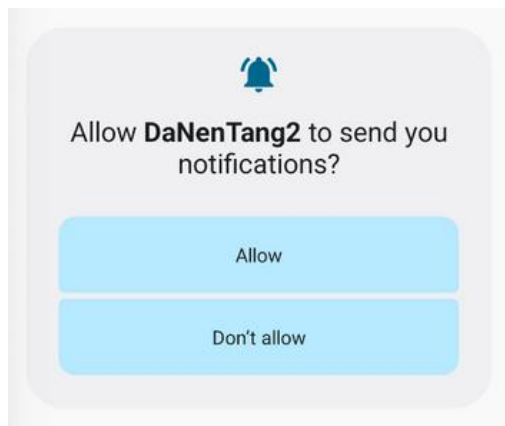
Ví dụ: đây là thông báo có định dạng JSON trong cùng một ứng dụng IM như trên, trong đó thông tin được đóng gói trong key dữ liệu chung và ứng dụng khách dự kiến sẽ diễn giải nội dung:

```
{
  "message":{
    "token":"bk3RNwTe3H0:CI2k_HHwgIpoDKCIZvvDMExUdFQ3P1...",
    "data":{
      "Nick" : "Mario",
      "body" : "great match!",
      "Room" : "PortugalVSDenmark"
    }
  }
}
```

- ☐ Cài đặt thư viện **@react-native-firebase/messaging**, đảm bảo rằng bạn đã cài đặt thư viện **@react-native-firebase/app** và setup firebase trước đó rồi

```
yarn add @react-native-firebase/messaging
```

- ☐ Cài đặt thư viện permission để xin quyền gửi notification trên ứng dụng

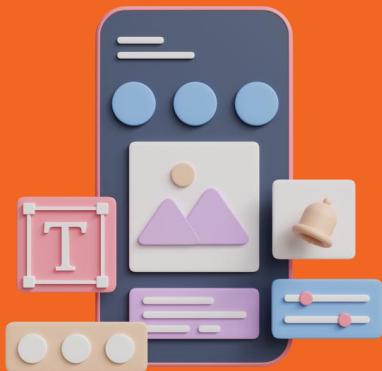


- ☐ Lưu ý, ở Android 13 giới thiệu permission mới để hiển thị thông báo. Điều này ảnh hưởng đến tất cả các ứng dụng chạy trên Android 13 trở lên sử dụng thông báo FCM.
- ☐ Bạn cần thêm permission mới này vào **AndroidManifest.xml**

```
<uses-permission android:name="android.permission.POST_NOTIFICATIONS"/>
```

- ☐ Để hiện popup yêu cầu permission thông báo ứng dụng như hình trên bạn cần cài đặt thêm thư viện **react-native-permissions**

npm i react-native-permissions



## LẬP TRÌNH ĐA NỀN TẢNG 2

BÀI 8: GIỚI THIỆU VỀ FCM VÀ TÙY CHỈNH NOTIFICATION

PHẦN 2: GỬI NOTIFICATION, TÙY CHỈNH NOTIFICATION CHO ỨNG DỤNG

- ☐ Gửi notification đến ứng dụng ở trạng thái foreground và background
- ☐ Tùy chỉnh notification hiển thị trên ứng dụng



- Viết hàm **requestUserPermission** để hiển thị popup yêu cầu người dùng cho phép thông báo

```
import PushNotification from 'react-native-push-notification';
```

```
const requestUserPermission = async () => {  
  try {  
    requestNotifications(['alert', 'sound']).then(({status, settings}) => {  
      // ... Làm gì đó ở đây  
    });  
  } catch (error) { console.log('requestUserPermission', error) }  
};
```

- Gọi hàm **requestUserPermission** trong hook **useFCM** tại **App.tsx**, mỗi khi người dùng được mở lên sẽ gọi hàm này

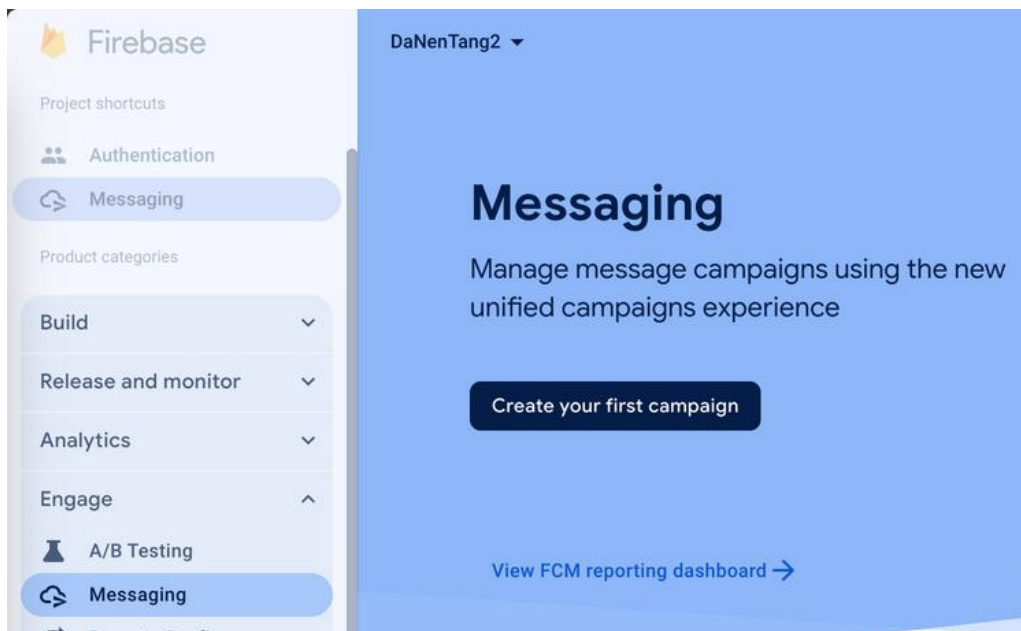
```
function App() {  
  const {requestUserPermission} = useFCM();  
  
  useEffect(() => {  
    requestUserPermission();  
  }, []);  
}
```

- ❑ Để nhận message từ Firebase sử dụng hàm **messaging().onMessage()**

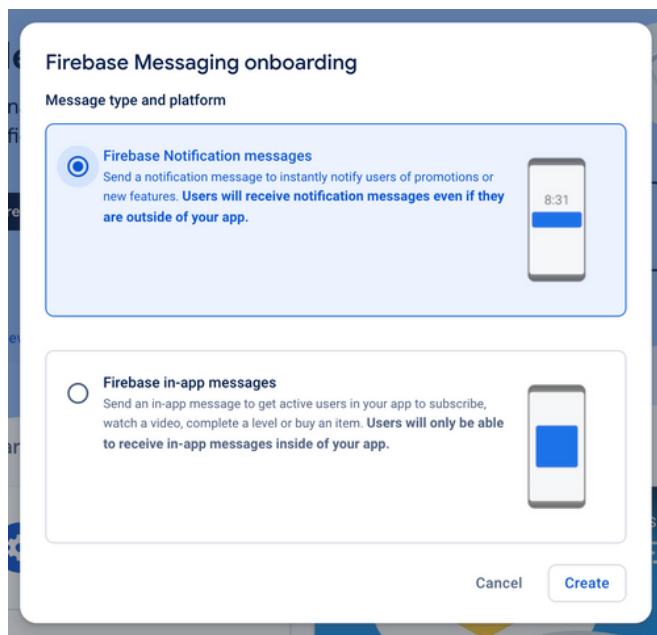
```
import messaging from '@react-native-firebase/messaging';
```

```
useEffect(() => {  
  const unsubscribe = messaging().onMessage(async remoteMessage => {  
    console.log('onMessage ', remoteMessage);  
  });  
  
  return () => { unsubscribe() };  
}, []);
```

- ☐ Sau nó trở lại Firebase console, đã tạo từ slide 7. Vào mục **Message** chọn **Create your first campaign**



- ☐ Popup hiện ra, chọn **Firestore Notification messages**. Sau đó nhấn **Create**



☐ Nhập nội dung **Notification messages type** theo ví dụ ở đây:

## 1 Notification

Notification title ?

Tiêu đề notification

Notification text

Nội dung notification

Notification image (optional) ?

Example: <https://yourapp.com/image.png>



Notification name (optional) ?

Enter optional name

### Device preview

This preview provides a general idea of how your message will appear on a mobile device. Actual message rendering will vary depending on the device. Test with a real device for actual results.

Send test message

Initial state

Expanded view



- ☐ Chọn app bạn muốn gửi thông báo tới, ở bản demo này tôi chọn **com.danentang2**. Sau đó nhấn **Next**


2 Target

User segment

Topic

Target user if...

App

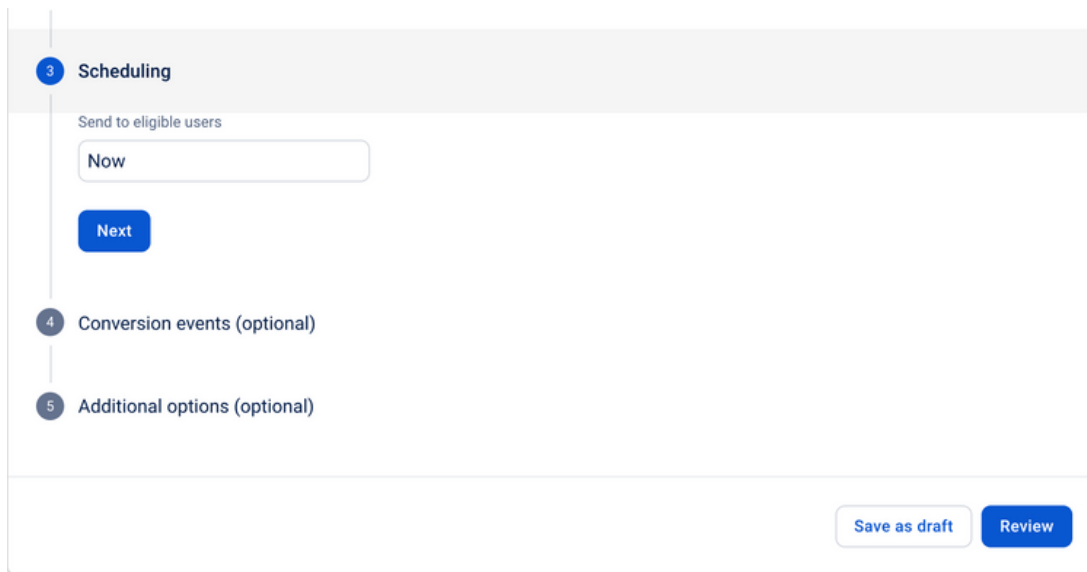
 com.danentang2

Target another app

Fewer than 100 users match this condition ?

Next

- ☐ Nhấn nút **Review**, sau đó có một Popup hiện lên để xác nhận gửi notification. Bạn nhấn nút **Publish** trên popup đó



The screenshot shows a multi-step interface for scheduling a notification. A vertical line on the left indicates the progress through five steps:

- 3 Scheduling**: This step is currently active and highlighted with a grey background. It contains the text "Send to eligible users" and a text input field with the value "Now". Below the input field is a blue button labeled "Next".
- 4 Conversion events (optional)**: This step is visible but not active.
- 5 Additional options (optional)**: This step is visible but not active.

At the bottom right of the interface, there are two buttons: "Save as draft" (outlined) and "Review" (solid blue).

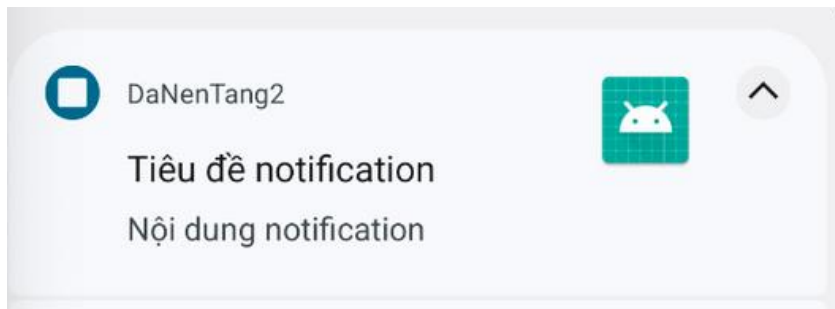


❖ Lưu ý: thời gian gửi notification này có thể bị delay lên đến 15 phút.

□ Đây là response hàm **onMessage** nhận được notification từ firebase console

```
onMessage {"collapseKey": "com.danentang2", "data": {}, "from": "182591633460", "messageId":  
"0:1701524662173735%a9cab6dfa9cab6df", "notification": {"android": {}, "body": "Nội dung notification",  
"title": "Tiêu đề notification"}, "sentTime": 1701524610802, "ttl": 2419200}
```

- Tiếp theo, bây giờ chúng ta sẽ muốn thông báo hiển thị nhận được từ FCM lên ứng dụng giống như hình dưới đây, bạn cần cài đặt thêm thư viện **react-native-push-notification**



☐ Cài đặt thư viện

```
npm install --save react-native-push-notification
```

- ❑ Để push notification hoạt động, bạn cần tạo **createChannel** trên Android

```
PushNotification?.createChannel(  
  {  
    channelId: 'notification-channel-id',  
    channelName: 'notification-channel',  
    soundName: 'default',  
  },  
  created => {  
    console.log('notification-channel-id: ', created);  
  },  
);
```

- Tiếp theo bạn sử dụng hàm **configure** để bắt sự kiện từ prop **onNotification** người dùng nhấn lên notification trên thanh thông báo.

```
PushNotification.configure({  
  onNotification(notification) {  
    console.log('On click notificatoin ', notification);  
  },  
  permissions: {  
    alert: true,  
    badge: true,  
    sound: true,  
  },  
});
```

- ❑ Trở lại hàm **onMessage** khi nhận được thông báo, bạn gọi hàm **localNotification** để push thông dựa trên dữ liệu thông báo nhận được hiện lên cho người dùng thấy

```
const unsubscribe = messaging().onMessage(async remoteMessage => {  
  console.log('onMessage ', remoteMessage);  
  PushNotification.localNotification({  
    channelId: 'notification-channel-id',  
    title: remoteMessage.notification?.title || "",  
    bigText: remoteMessage.notification?.body || "", //content for Android  
    message: remoteMessage.notification?.body || "", //content for Ios  
    ...remoteMessage,  
  }));
```

- ☐ Ngoài ra, bạn có thể làm thêm các nội dung khác lên thông báo như thay đổi âm thanh thông báo, thay đổi icon, thêm nội dung,... để tìm hiểu thêm chi tiết, các bạn truy cập trang

<https://www.npmjs.com/package/react-native-push-notification>

- ☐ Giới thiệu về FCM
- ☐ Cài đặt thư viện
- ☐ Ở bài này, các bạn sẽ học được cách gửi notification đến ứng dụng ở chế độ foreground và background
- ☐ Tùy chỉnh notification hiển thị trên ứng dụng





# Kết thúc