

Push Notifications (PN) con Firebase Cloud Messages(FCM)

La solución de Google Firebase Cloud Messaging ofrece una serie de servicios que amplían las posibilidades ofrecidas por el sistema de Push Notifications nativo de iOS, Apple Push Notification service (APNs). Para conseguirlo añade una capa más al protocolo APN permitiendo servicios añadidos como:

- Analytics sobre las notificaciones push
- El mismo sistema de generación /gestión de mensajes se puede usar para dispositivos iOS y Android (Apn funciona solo para dispositivos iOS)
- Puede enviar mensajes de notificaciones y mensajes de datos
- Puede enviar mensajes desde la aplicación al servidor (mensajes de respuesta, implementación de chat services, etc)

Para usar FCM, es necesario habilitar y configurar tanto el servicio de APN como de FCM. Para habilitar el servicio APN en una app iOS es necesario añadir la capability PN desde el proyecto XCode y al mismo tiempo es necesario tener un certificado APN asociado con la app en el portal de Apple developer.

Cómo funciona la red APN y cómo enviar una notificación desde el terminal:

Para que una app reciba PN es necesario que tenga asignado un **“device apn token”**.

El device apn token es un código que cada aplicación puede obtener al pidiéndoselo a la red APN la primera vez que se ejecuta la app y guardándolo o imprimiéndolo en consola para poderlo usar para enviar notificaciones desde otras herramientas (por ejemplo desde el terminal) para depurar y probar cosas.

La app tendrá que pedir este token a la red APN con la llamada:

```
UIApplication.shared.registerForRemoteNotifications()
```

Para poder enviar PN directamente desde el terminal. Se necesitarán 4 parámetros::

1. El app *bundleId*
2. El device apn token
3. Un certificado (en formato .pem) que autorice la entidad que genera el mensaje (en nuestro caso nosotros desde el terminal. Este certificado se obtiene desde la página del Apple developer, en la sección del identificador de la App. Quién tiene acceso a este certificado está autorizado porque solo los miembros de la cuenta pueden acceder. Entonces tener este certificado permite ser autorizado a enviar mensajes de notificación para esa aplicación. Notar que el certificado que se puede descargar desde la página apple está en formato .p12 mientras la

petición http para enviar mensaje lo requiere en format .pem. Usare el siguiente comando para convertir de formato:

```
a. openssl pkcs12 -in MyApp_APNS_Certificate.p12 -out myapp-push-cert.pem -nodes -clcerts
```

Para saber más sobre como crear el certificado APN en el portal Apple: [Establishing a Certificate-Based Connection to APNs](#)

4. El mensaje en formato json textual, por ejemplo: '{"aps":{"alert":"Hello from APNs!","sound":"default"}}'

Por ejemplo con el siguiente comando el dispositivo identificado por deviceToken recibirá la notificación y al abrirla se lanzará la app identificada por el bundleId:

```
curl -http2 --cert <certificado.p12> -H "apns-topic: <bundleID>" -d '{"aps":{"alert":"Hello from APNs!","sound":"default"}}' https://api.development.push.apple.com/3/device/<deviceToken>
```

Importante: el usuario tiene que habilitar las notificaciones para la app desde las configuraciones.

Firebase Cloud Messaging (FCM)

- Para usar el servicio FCM, además de habilitar el uso del servicio APN, hay que habilitar y configurar el servicio Firebase Cloud Messages:
 - Se tiene que tener una cuenta Firebase y un proyecto Firebase con el servicio Cloud Message
 - FCM necesita establecer una conexión certificate-based para conectarse a la APN de Apple.
 - El proveedor del servicio PN (firebase) necesita el “provider certificate” creado en el portal Apple dev como credenciales para poder usar la red APN. Hay que subirlo desde la pagina de configuración del proyecto firebase en el apartado “Cloud Messaging”
 - NB: De momento se ha creado un certificado para entorno de desarrollo. Sin embargo, con el mismo método se puede crear uno para entorno de producción.

APP de prueba

- La App de prueba que se ejecuta en el dispositivo tiene que usar el framework firebase.
- El proyecto tiene que tener el fichero de configuración de los servicios firebase que se puede descargar desde la cuenta del proyecto firebase (ver documentación firebase)
- //La app pide al usuario los permisos necesarios para usar el servicio Push Notification
- La app pide al usuario los permisos necesarios que enviarle notificaciones

```
UNUserNotificationCenter.current().requestAuthorization()
```

El Usuario tiene que manualmente dar su permiso en el diálogo que se le presenta.

- La app comunicará con la red APN para recoger el token APN asignado por la red (identificador del dispositivo):

```
UIApplication.shared.registerForRemoteNotifications()
```

- La app reenviará el token APN al backend firebase:

```
self.forwardTokenToServer(tokenString: deviceTokenString)
```

Para detectar errores

<https://firebase.googleblog.com/2017/01/debugging-firebase-cloud-messaging-on.html>