

PROGRAMACION EN ANDROID CON JAVA PRACTICA No. 9

Notificaciones Push en Android con Firebase Google Cloud Messaging.

Introducción

Las notificaciones push son una potente herramienta para comunicar un dispositivo cliente con Android y un servidor, mediante mensajes. El servidor es el que inicia la petición, es decir, el servidor notifica al dispositivo cliente sobre algún evento. Todo ello sin necesidad de estar ejecutando una tarea en background que este constantemente haciendo peticiones al servidor y comprobando si hay nuevos mensajes para mostrar al usuario.

Las notificaciones push tienen como ventaja frente a la técnica del polling (peticiones periódicas al servidor para averiguar si hay nuevos eventos pendientes de notificar) que consume menos recursos y las notificaciones llegan al instante y no al cabo de un periodo de tiempo.

Google ofrece un servidor gratuito desde el cuál gestionar estas notificaciones. Este servidor se llama Firebase Cloud Message for Android (FCM) una actualización de su antiguo Google Cloud Message for Android (GCM).

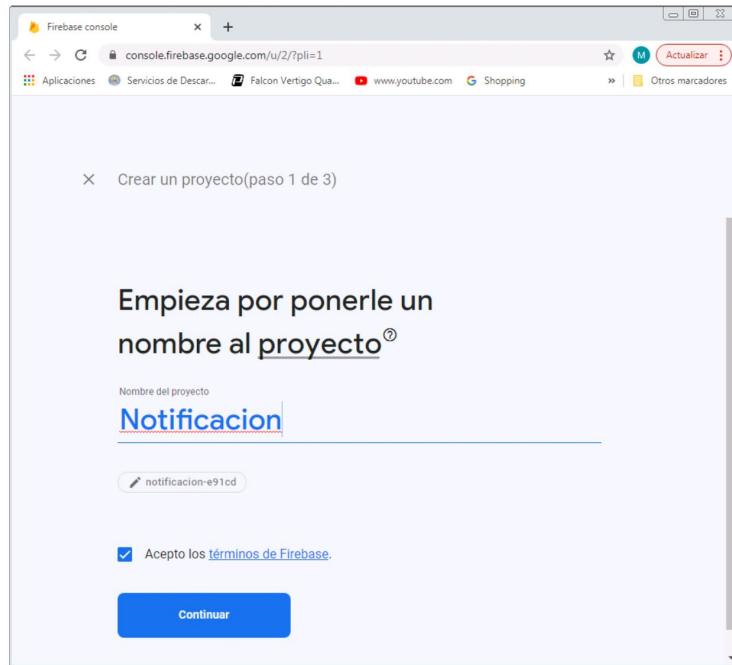
Las principales características del FCM son:

1. Permite a un servidor enviar mensajes a una aplicación Android.
2. No es necesario que la aplicación esté en ese momento lanzada para recibir mensajes. El sistema levantará la aplicación cuando reciba la notificación, siempre y cuando la aplicación esté correctamente configurada y tenga permisos para ello.
3. Pasa los mensajes en plano, es decir, sin formato alguno. Posteriormente la aplicación que se ha levantado es la que se encarga de personalizar el mensaje.
4. Requiere de un dispositivo que tenga como mínimo la versión 2.2 de Android y la Google Play Store instalada y actualizada, o en caso de que sea un emulador la versión 2.2 de Android y las Google API's.
5. También requiere tener instalado el Google Play Services. Para versiones anteriores a la 3.0 es necesario tener configurada una cuenta de Google en el dispositivo. No es necesario para versiones 4.0.4 o superiores.

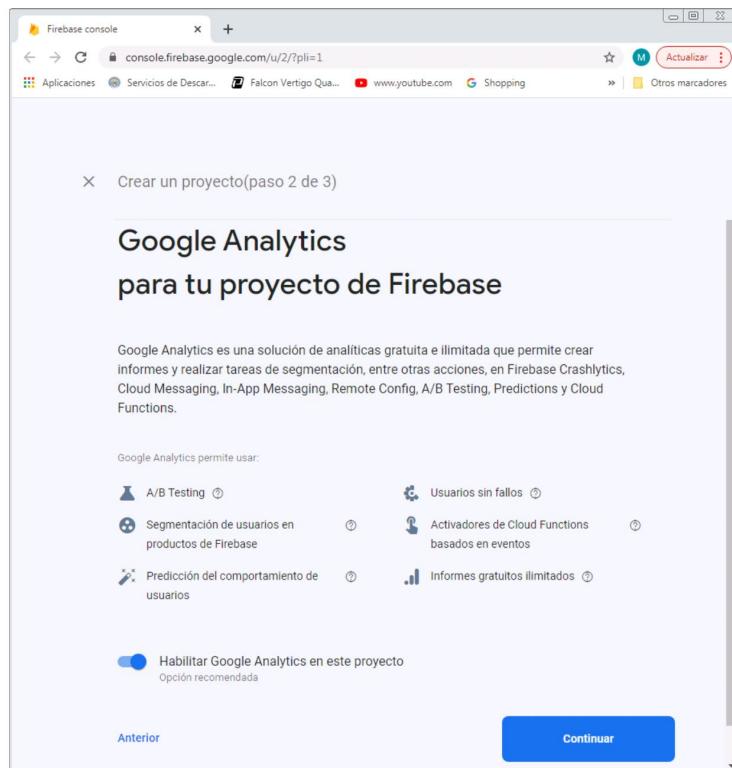
Activar el servicio de Firebase Cloud Messaging (FCM)

Debemos Acceder a la Consola de Firebase desde la url (<https://firebase.google.com/>) y haciendo clic sobre la opción Got to console. Para ello debemos contar con una cuenta de google e ingresaremos a ella.

Tras pulsar Crear Nuevo Proyecto nos aparecerá una ventana donde podremos crear un nuevo proyecto. En esta tenemos que introducir el nombre del proyecto y hacer clic sobre el botón Continuar.



En la siguiente ventana hacemos clic en el botón Continuar.



En la siguiente ventana seleccionamos el país donde estamos creando el servicio, y aceptamos los términos y condiciones de Google, continuamos haciendo clic en el botón Crear Proyecto, y nuevamente en el botón Continuar.

Crear un proyecto(paso 3 de 3)

Configurar Google Analytics

Ubicación de Analytics ⓘ

Colombia

Términos de Google Analytics y configuración de uso compartido de datos

Usar la configuración predeterminada para compartir datos de Google Analytics. [Learn more](#)

- ✓ Comparte tus datos de Analytics con nosotros para ayudarnos a mejorar nuestros productos y servicios.
- ✓ Comparte tus datos de Analytics con nosotros para habilitar las comparativas.
- ✓ Comparte tus datos de Analytics con Google para habilitar el servicio de asistencia técnica.
- ✓ Comparte tus datos de Analytics con los especialistas en cuentas de Google.

Acepto los [términos de protección de datos entre responsables del tratamiento de datos y responsables del tratamiento de datos de mediciones](#), y reconozco que estoy sujeto a la [política de consentimiento de usuarios finales de la Unión Europea](#). Es obligatorio marcar esta casilla si vas a compartir tus datos de Analytics para mejorar los productos y servicios de Google. [Más información](#)

Acepto los [términos de Google Analytics](#).

Al crear el proyecto, también se creará una propiedad de Google Analytics que se asociará a tu proyecto de Firebase. De esta forma, se habilitará el flujo de datos entre los productos. Los datos que se exportan de tu propiedad de Google Analytics a Firebase están sujetos a los términos del servicio de Firebase, mientras que los datos que se importan de Firebase a Google Analytics se rigen por los términos del servicio de Google Analytics. [Más información](#)

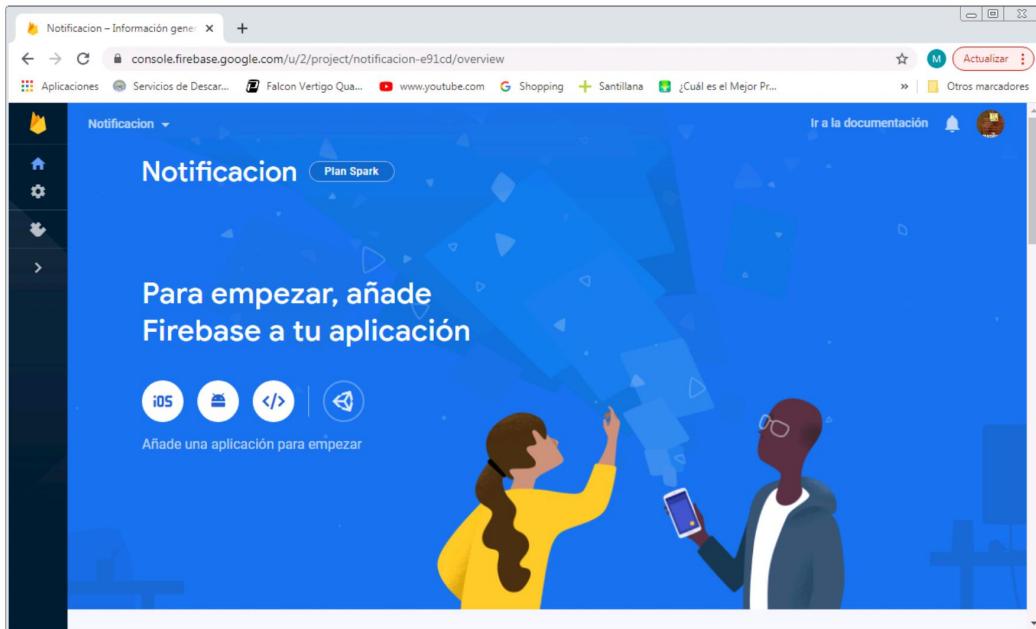
Anterior Crear proyecto

Notificación

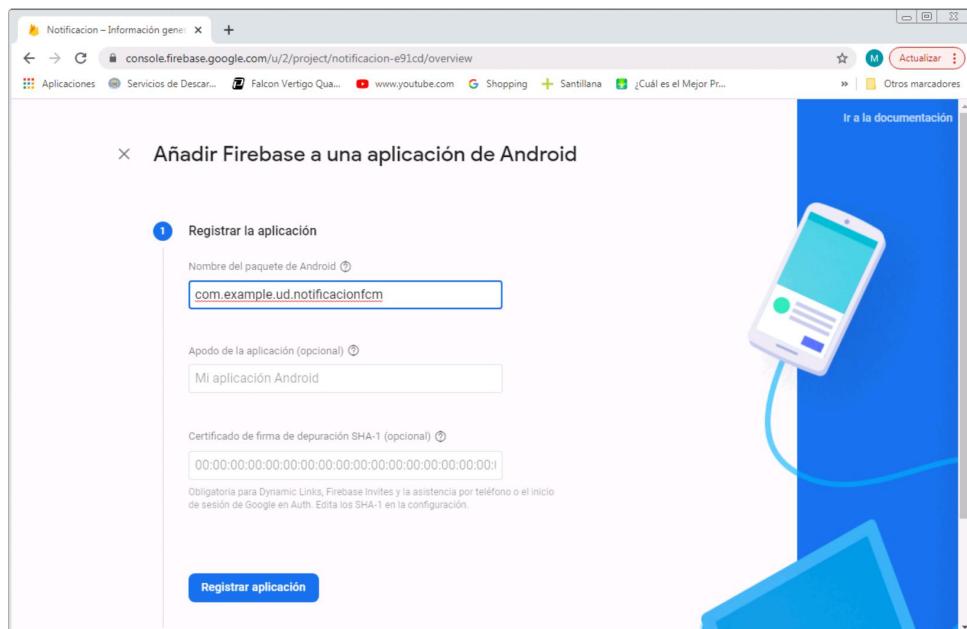
✓ El nuevo proyecto está listo.

Continuar

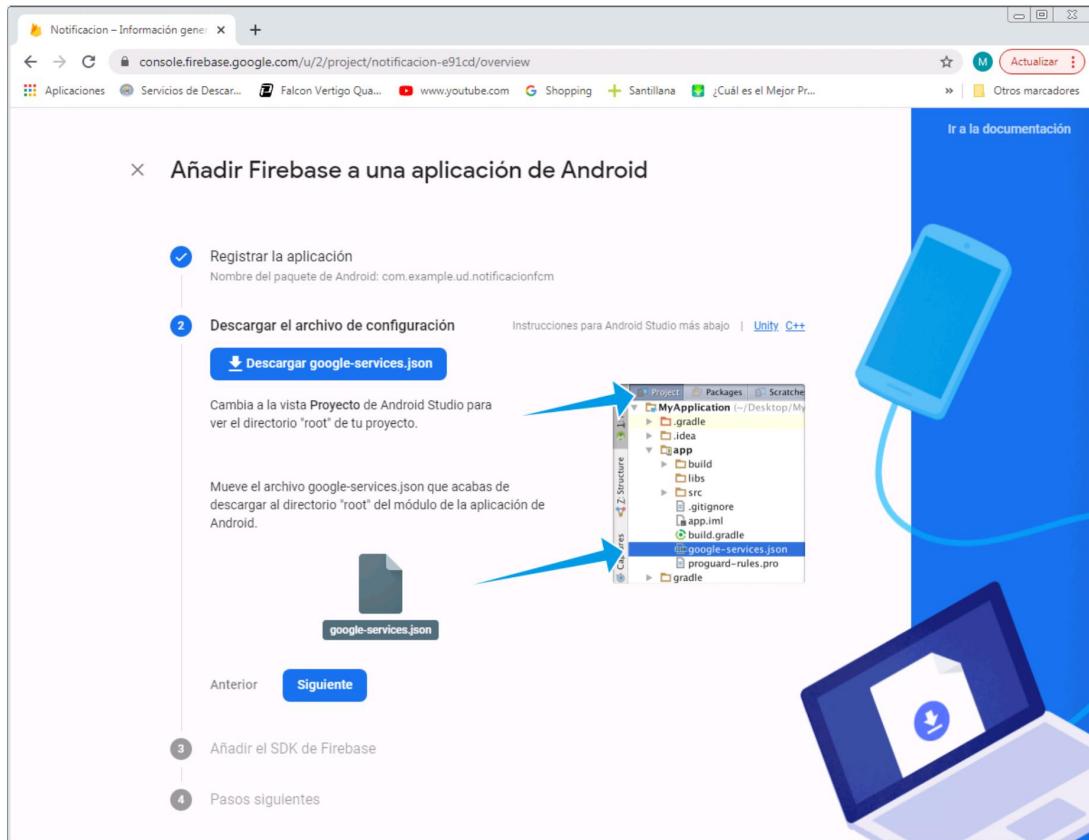
En la nueva ventana vamos a mantenernos en el proyecto Notificacion y hacemos clic sobre el icono de android

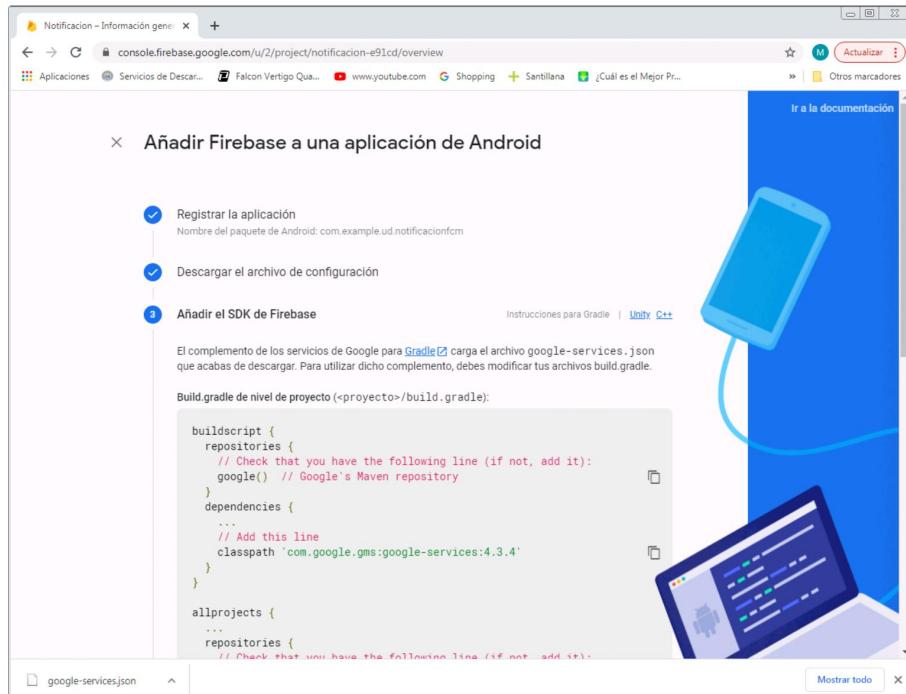


En la nueva ventana colocamos el nombre completo del paquete que vamos a crear, en nuestro caso colocaremos por el momento: **com.example.ud.notificacionfcm**, dejaremos vacíos los demás campos y haremos clic sobre el botón Registrar Aplicación.



Salvamos en nuestro equipo el archivo que automáticamente se descarga (google-services.json), este archivo se debe guardar en la raíz app del proyecto que crearemos más adelante. Daremos clic en el botón Siguiente, hasta que nos aparezca la opción de hacer clic sobre el botón Ir a la Consola.





Añadir Firebase a una aplicación de Android

- Registrar la aplicación
- Descargar el archivo de configuración
- Añadir el SDK de Firebase

El complemento de los servicios de Google para [Gradle](#) carga el archivo google-services.json que acabas de descargar. Para utilizar dicho complemento, debes modificar tus archivos build.gradle.

Build.gradle de nivel de proyecto (<proyecto>/build.gradle):

```
buildscript {
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
    }
    dependencies {
        ...
        // Add this line
        classpath 'com.google.gms:google-services:4.3.4'
    }
}

allprojects {
    ...
    repositories {
        // Check that you have the following line (if not, add it):
    }
}
```

Build.gradle de nivel de aplicación (<proyecto>/<app-module>/build.gradle):

Java Kotlin

```
apply plugin: 'com.android.application'
// Add this line
apply plugin: 'com.google.gms.google-services'

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:26.1.0')

    // Add the dependency for the Firebase SDK for Google Analytics
    // When using the BoM, don't specify versions in Firebase dependencies
    implementation 'com.google.firebaseio:firebase-analytics'

    // Add the dependencies for any other desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```

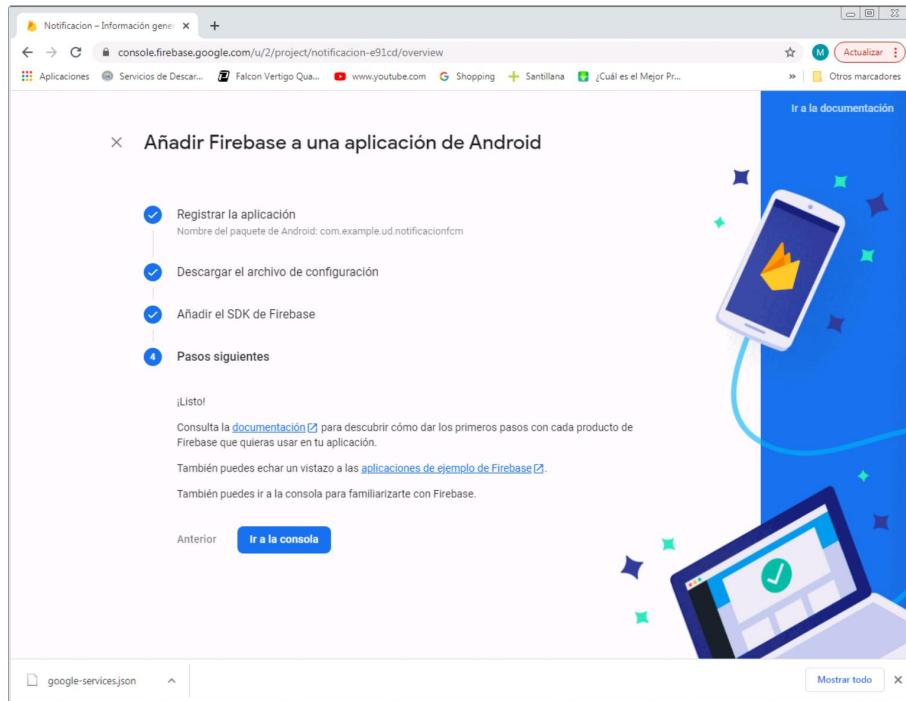
Si utilizas la lista de materiales de Firebase para Android, tu aplicación siempre usará versiones de bibliotecas de Firebase compatibles. [Más información](#)

Por último, pulsa Sincronizar ahora en la barra que aparece en el IDE:

Gradle files have changed since last sync [Sync now](#)

[Anterior](#) [Siguiente](#)

4 Pasos siguientes



No cierre el navegador y continúe creando la aplicación de Android.

APLICACION ANDROID

PASO1 Creación de la Aplicación

Ejecute el programa *Android Studio*. En la ventana que se despliega haga clic sobre el menú *File* y la Opción *New Project*.

En la nueva ventana seleccione la pestaña *Phone and Table*, elija la opción *Empty Activity* y haga clic en el botón *Next*.

En la siguiente ventana escriba como nombre del proyecto **NotificacionFCM** y como dominio escriba **com.example.ud.notificacionfcm**, seleccione la mínima versión de android de ejecución para su nueva aplicación (se recomienda la API 23 por defecto), seleccione como lenguaje Java y haga clic sobre el botón *Finish*.

Nota: el dominio y el nombre deben corresponder al escrito en la consola del Firebase.

Adición de archivos al proyecto

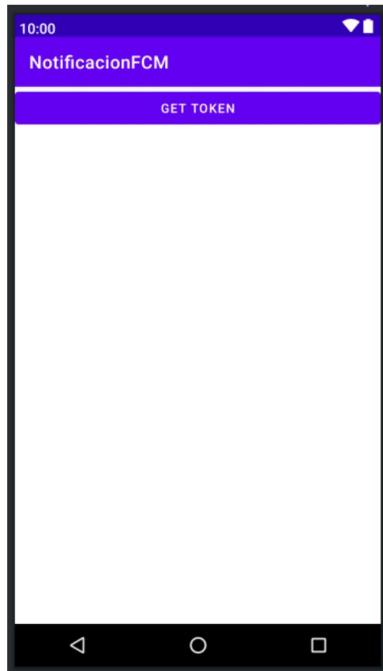
Copie el archivo info.png que acompaña la practica en la carpeta drawable del proyecto.
Copie el archivo google-services.json en la carpeta app del proyecto

PARTE 2.

Adición de Código

Modificación del archivo *activity_main.xml*

Nuestra actividad al final debería lucir de la siguiente manera:



Haga doble clic sobre el archivo *activity_main.xml* del paquete *layout*, y cambie el modo de visualización a texto, por medio de la pestaña Text. Modifique el código de la siguiente manera:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="171dp"
        tools:layout_editor_absoluteY="149dp">
```

```
<Button
    android:id="@+id/button"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="getToken01"
    android:text="GET TOKEN" />
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Modificación del archivo **build.gradle (Project: NotificacionFCM)**

Agregamos reglas para incluir el complemento google-services:

```
dependencies {
    classpath "com.android.tools.build:gradle:4.1.1"
    classpath 'com.google.gms:google-services:4.3.4'
}
```

Modificación del archivo **build.gradle (Module: app)**

Agregamos reglas para incluir firebase y complementos glade

```
plugins {
    id 'com.android.application'
    id 'com.google.gms.google-services'
}

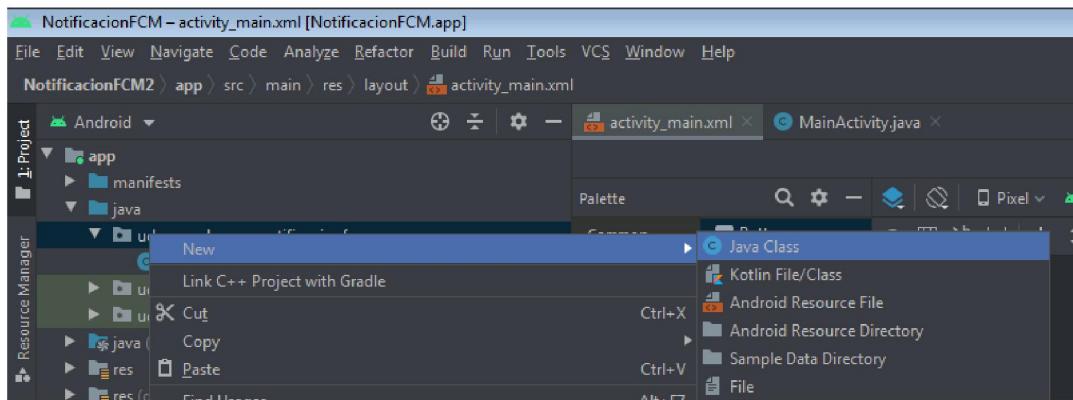
...
...

dependencies {

    implementation 'androidx.appcompat:appcompat:1.2.0'
    implementation 'com.google.android.material:material:1.2.1'
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
    implementation 'com.google.firebaseio:firebase-database:19.5.1'
    implementation 'com.google.firebaseio:firebase-messaging:21.0.0'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'
}
```

Creación de la clase **MyFirebaseMessagingService**

Hacemos clic sobre el proyecto en el paquete java y buscamos la opción Java Class del submenú New, en la nueva ventana que se despliega colocamos el nombre MyFirebaseMessagingService y damos enter.



Adicione el siguiente código a la clase:

```
public class MyFirebaseMessagingService extends FirebaseMessagingService {
    private static final String TAG = "MFBMS:";
    @Override
    public void onMessageReceived(@NonNull RemoteMessage remoteMessage) {
        super.onMessageReceived(remoteMessage);
        Log.d(TAG, "From: " + remoteMessage.getFrom());
        // Check if message contains a data payload.
        if (remoteMessage.getData().size() > 0) {
            Log.d(TAG, "Message data payload: " + remoteMessage.getData());
            Log.d(TAG, "Title: " + remoteMessage.getData().get("Title"));
            Log.d(TAG, "Mesage: " + remoteMessage.getData().get("Mesage"));
        }
        // Check if message contains a notification payload.
        if (remoteMessage.getNotification() != null) {
            Log.d(TAG, "Message Notification Title: " +
            remoteMessage.getNotification().getTitle());
            Log.d(TAG, "Message Notification Body: " +
            remoteMessage.getNotification().getBody());

            sendNotification(remoteMessage.getNotification());
        }
    }

    private void sendNotification(RemoteMessage.Notification notification) {
        //Obtenemos una referencia al servicio de notificaciones
        String ns = Context.NOTIFICATION_SERVICE;

        //Para versiones superiores o iguales a 26 (Oreo)
        if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.O){
            CharSequence NombreCanal = "Notificacion";
            NotificationChannel notificationChannel = new
            NotificationChannel("UDNOTIFICACION", NombreCanal,
            NotificationManager.IMPORTANCE_DEFAULT);
            NotificationManager notificationManager= (NotificationManager)
            getSystemService(ns);
            notificationManager.createNotificationChannel(notificationChannel);
        }
    }
}
```

```
}

// Obtenemos el contexto de la aplicación
Context contexto = getApplicationContext();

//Configuramos el Intent
Intent notIntent = new Intent(contexto, MainActivity.class);
// Usamos una PendingIntent para crear la notificación
PendingIntent contIntent = PendingIntent.getActivity(contexto, 0,
notIntent, 0);

//Configuramos la notificación que va a aparecer en la barra
int icono = android.R.drawable.stat_sys_warning;
CharSequence textoEstado = "¡Atención!";
CharSequence titulo = notification.getTitle();
CharSequence descripcion = notification.getBody();
long hora = System.currentTimeMillis();

NotificationCompat.Builder mBuilder =
(NotificationCompat.Builder) new
NotificationCompat.Builder(contexto, "UDNOTIFICACION")
.setSmallIcon(icono)
.setLargeIcon(((BitmapDrawable)getResources()
.getDrawable(R.drawable.info)).getBitmap())
.setContentTitle(titulo)
.setContentText(descripcion)
.setContentInfo(textoEstado)
.setWhen(hora)
.setContentIntent(contIntent)
.setAutoCancel(true) //AutoCancel: cuando se pulsa la
notificación desaparece

.setColor(getResources().getColor(R.color.design_default_color_primary_dark))
//Para añadir sonido y vibración
.setVibrate(new long[] {1000, 500, 1000, 500})

.setSound(RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION))
;

//Enviamos la notificación
NotificationManagerCompat notManager =
NotificationManagerCompat.from(contexto);
notManager.notify(1, mBuilder.build());

}

@Override
public void onNewToken(@NonNull String s) {
    super.onNewToken(s);
    Log.e("TOKEN:", s);
}
}
```

Modificación del archivo MainActivity.java

Haga doble clic sobre el archivo *MainActivity.java* del paquete *java*, y agregue el siguiente metodo:

```
public void getToken01(View v){
    // Get token
    FirebaseMessaging.getInstance().getToken()
        .addOnCompleteListener(new OnCompleteListener<String>() {
            @Override
            public void onComplete(@NonNull Task<String> task) {
                if (!task.isSuccessful()) {
                    Log.w("TOKEN:", "Fetching FCM registration token failed",
                            task.getException());
                    return;
                }

                // Get new FCM registration token
                String token = task.getResult();

                // Log and toast
                String msg = token;
                Log.d("TOKEN:", msg);
                Toast.makeText(MainActivity.this, msg,
                        Toast.LENGTH_SHORT).show();
            }
        });
}
```

Ejecute la aplicación, haga clic sobre el botón GET TOKEN y copie el token, ya sea desde la ventana *Logcat* o desde la aplicación.

APLICACION PHP

PASO1

Creación de la Aplicación php para envio de mensaje

Sobre un servidor web cree un archivo php llamado *envioFCM.php*, sobre el copie el siguiente código:

```
<?php
// API access key from Google API's Console
define( 'API_ACCESS_KEY', 'KEY_DE_FIREBASE');
$registrationIds = ["TOKEN_DE_LA_APPLICACION"];

// prep the bundle
$msg = [
    'title'      => 'Programación Android',
    'body'       => 'Prueba 2'
];
```

```
$fields = [  
    'registration_ids' => $registrationIds,  
    'notification'      => $msg  
];  
$headers = [  
    'Authorization: key=' . API_ACCESS_KEY,  
    'Content-Type: application/json'  
];  
$fields = json_encode( $fields );  
  
$ch = curl_init();  
curl_setopt( $ch,CURLOPT_URL, 'https://fcm.googleapis.com/fcm/send' );  
curl_setopt( $ch,CURLOPT_POST, true );  
curl_setopt( $ch,CURLOPT_HTTPHEADER, $headers );  
curl_setopt( $ch,CURLOPT_RETURNTRANSFER, true );  
curl_setopt( $ch,CURLOPT_SSL_VERIFYPeer, false );  
curl_setopt( $ch,CURLOPT_POSTFIELDS, $fields );  
$result = curl_exec($ch );  
curl_close( $ch );  
  
echo $result;  
?>
```

Cambie el valor TOKEN_DE_LA_APPLICACION por el token copiado al ejecutar su aplicación.

Sobre el navegador conectado a la consola de Firebase, vaya a configuración y busque la pestaña de mensajería en la nube (Cloud Messaging) y copie la clave del servidor, reemplace en el código anterior la variable KEY_DE_FIREBASE con este valor.

Abra una nueva pestaña en el navegador y copie la siguiente ruta <http://localhost/envioFCM.php>, si todo se ejecuta correctamente el dispositivo debe recibir el mensaje en alrededor de 15 segundos.