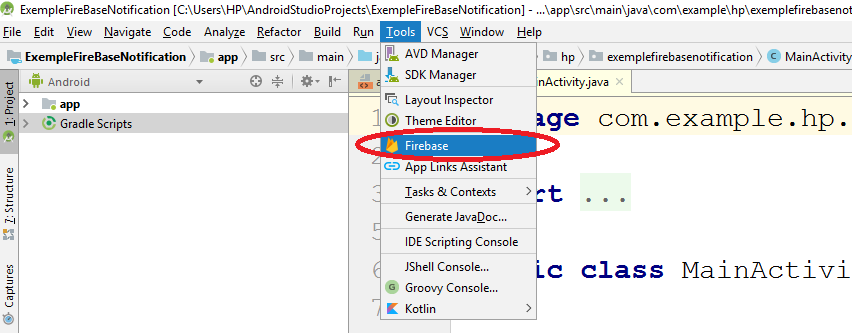
**Firebase**

# Définition

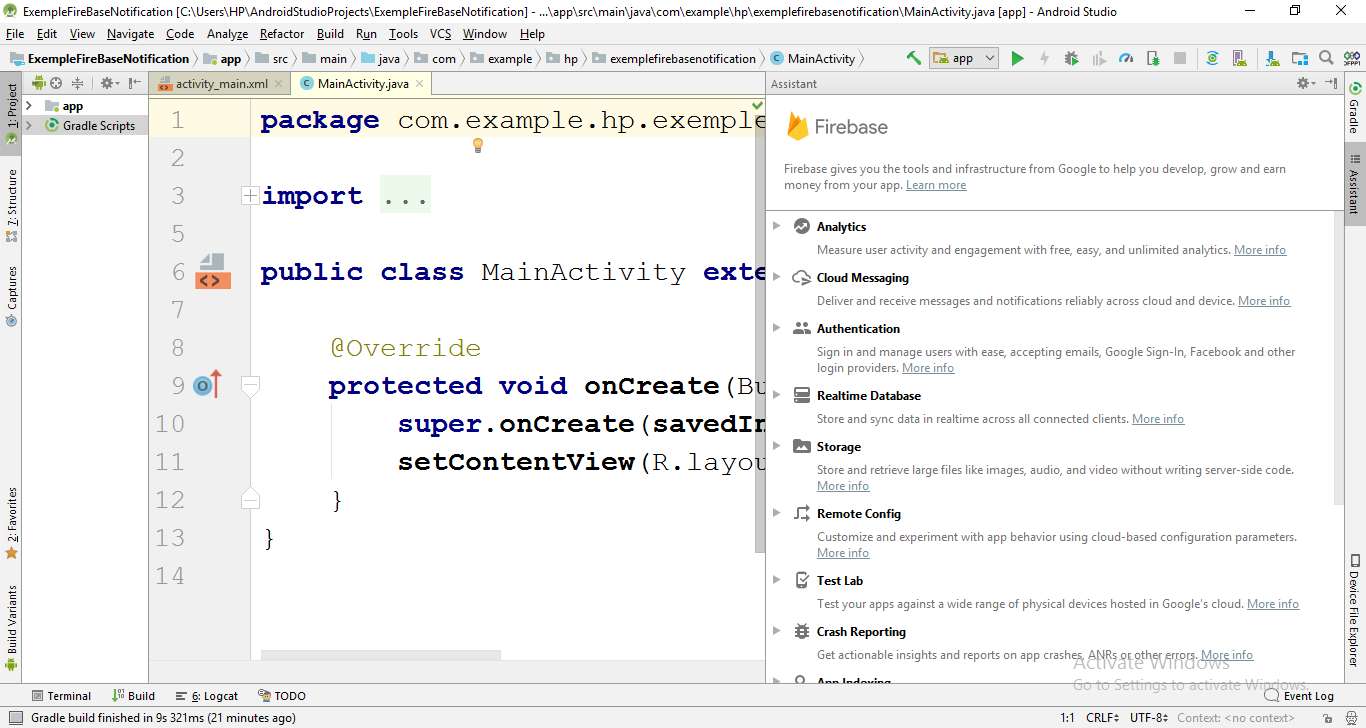
Firebase est un service cloud qui aide les développeurs à développer leurs applications.

# Push Notification

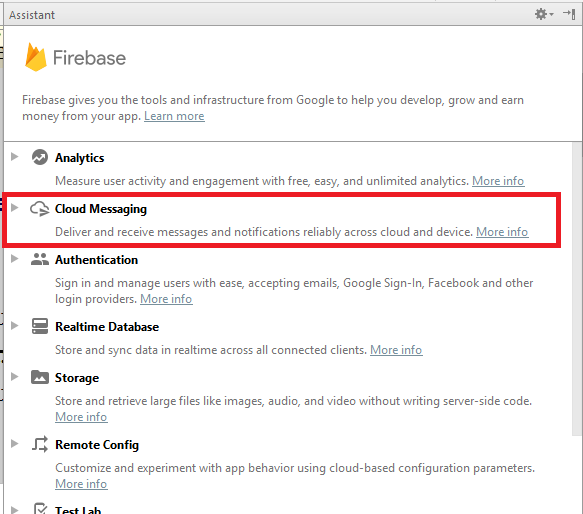
1. Créer un nouveau projet sous Android studio
2. Accéder au menu suivant :



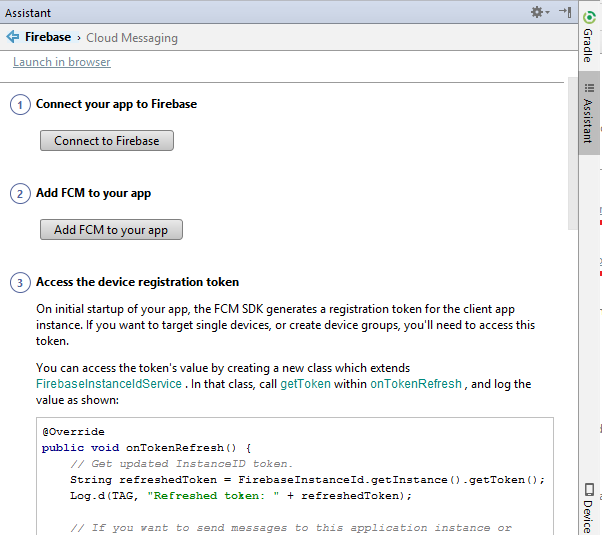
La clique sur ce menu permet d’ouvrir l’assistant du Firebase :



Dans l’assistant du Firebase, Cliquer sur Cloud Messaging :



1. Dans l’assistant du cloud Messaging, Cliquer sur le bouton **Connect to Firebase** puis sur le bouton **Add FCM to your app.**



1. Créer la classe **FirebaseMessagingService** qui hérite de la classe **FirebaseMessagingService.** Dans cette classe il faut redéfinir la méthode **onMessageReceived :**

**public class** FirebaseMessagingService **extends** com.google.firebase.messaging.FirebaseMessagingService {  
 @Override  
 **public void onMessageReceived**(RemoteMessage **remoteMessage**) {  
 Notification.Builder n =**new** Notification.**Builder**(**getApplicationContext**());  
 n.**setContentTitle**(**remoteMessage**.**getNotification**().**getTitle**());  
 n.**setContentText**(**remoteMessage**.**getNotification**().**getBody**());  
 n.**setSmallIcon**(R.drawable.***i1***);  
 n.**setDefaults**(Notification.***DEFAULT\_SOUND***);  
 NotificationManager mNotification = (NotificationManager) **getSystemService**(***NOTIFICATION\_SERVICE***);  
  
 mNotification.**notify**(1,n.**build**());  
 }

@Override  
 **public void onNewToken**(String s) {  
 *// Get updated InstanceID token.* super.onNewToken(s);

Log.***d***(**"DeviceToken ==> "**, s);

}  
  
}

1. Ajouter dans le fichier manifest.xml le code suivant :

<**service  
 android:name=".FirebaseMessagingService"**>  
 <**intent-filter**>  
 <**action android:name="com.google.firebase.MESSAGING\_EVENT"**/>  
 </**intent-filter**>  
</**service**>

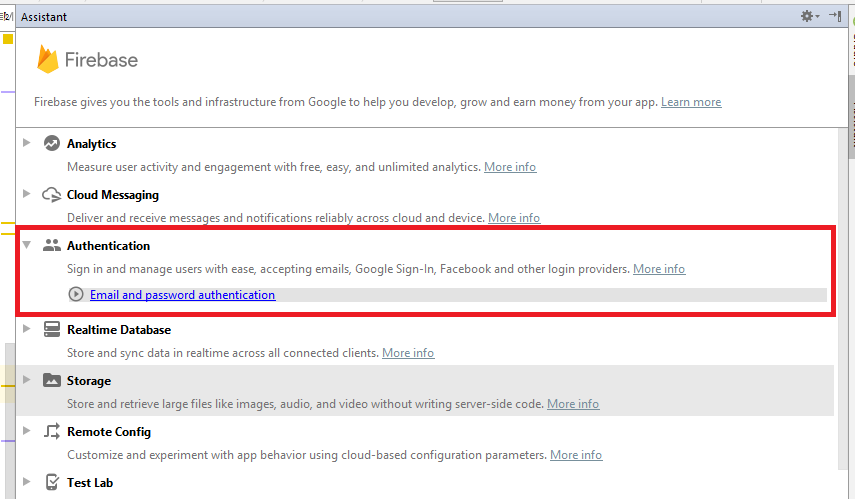
Vous pouvez utiliser cette function qui permet d’afficher une notification quel que soit la version d’Android de l’appareil :

**public void showNotification(String title, String message) {  
  
 if (Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*O*) {  
 String CHANNEL\_ID = "channel 1";  
 NotificationManagerCompat notificationManager = NotificationManagerCompat.*from*(getApplicationContext());  
  
 NotificationChannel channel1 = new NotificationChannel(CHANNEL\_ID, "Channel 1", NotificationManager.*IMPORTANCE\_HIGH*);  
 channel1.setDescription("this is channel 1");  
  
 NotificationManager manager = getSystemService(NotificationManager.class);  
 manager.createNotificationChannel(channel1);  
  
 Notification notification = new NotificationCompat.Builder(getApplicationContext(), App.*CHANNEL\_1\_ID*)  
 .setSmallIcon(R.drawable.*ic\_launcher\_background*)  
 .setContentTitle(title)  
 .setContentText(message)  
 .setPriority(NotificationCompat.*PRIORITY\_HIGH*)  
 .build();  
  
 notificationManager.notify(1, notification);  
 } else {  
 Notification.Builder n = new Notification.Builder(getApplicationContext());  
 n.setContentTitle("Alerte de test");  
 n.setContentText("Message de mon alerte ");  
 n.setSmallIcon(R.drawable.*ic\_launcher\_background*);  
 n.setDefaults(Notification.*DEFAULT\_SOUND*);  
 NotificationManager mNotification = (NotificationManager) getSystemService(*NOTIFICATION\_SERVICE*);  
  
 mNotification.notify(1, n.build());  
 }  
}**

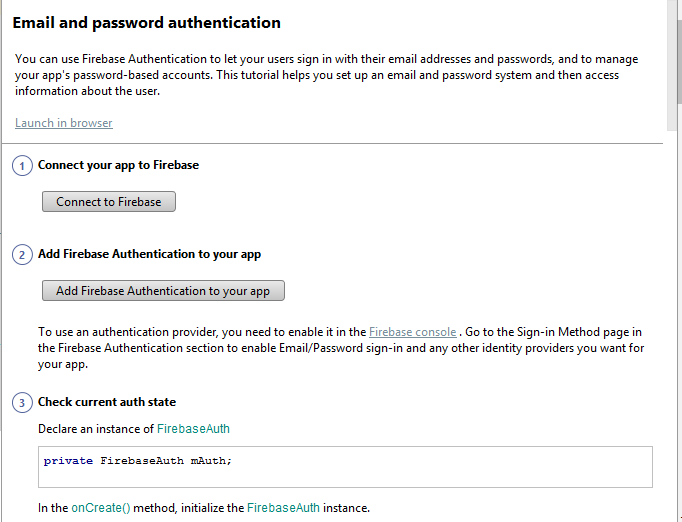
# Authentication par email et password

Les étapes pour réaliser une authentification par email et password sont :

1. Dans l’assistant de Firebase, cliquer sur **Authentification** puis sur **Email and password authentification.**



1. Cliquer ensuite sur **Connect to Firebase** puis sur le button **Add Firebase Authentification**



1. Déclarer la variable suivant dans mainActivity.java

**private** FirebaseAuth **mAuth**;

Dans la méthode Oncreate

**mAuth** = FirebaseAuth.***getInstance***();

1. Pour ajouter un utilisateur :

**mAuth.createUserWithEmailAndPassword(mail,pwd).addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {  
 @Override  
 public void onComplete(@NonNull Task<AuthResult> task) {  
 if (task.isSuccessful()) {  
Log.*d*("Create User ==>", "createUserWithEmail:success");  
 FirebaseUser user = mAuth.getCurrentUser();  
  
 } else {  
 *// If sign in fails, display a message to the user.* Log.*w*("Create User ==>", "createUserWithEmail:failure", task.getException());  
 Toast.*makeText*(MainActivity.this, "Authentication failed.",  
 Toast.*LENGTH\_SHORT*).show();  
 }**

1. Pour l’existence d’un utilisateur :

**mAuth**.**signInWithEmailAndPassword**(mail,pwd).**addOnCompleteListener**(**this**, **new** OnCompleteListener<AuthResult>() {  
 @Override  
 **public void onComplete**(@NonNull Task<AuthResult> **task**) {  
 **if** (**task**.**isSuccessful**()) {  
 *// Sign in success, update UI with the signed-in user's information* Toast.***makeText***(MainActivity.**this**, **"Authentication success."**,  
 Toast.***LENGTH\_SHORT***).**show**();  
 FirebaseUser user = **mAuth**.**getCurrentUser**();  
  
 } **else** {  
 *// If sign in fails, display a message to the user.* Log.***w***(**"Create User ==>"**, **"createUserWithEmail:failure"**, **task**.**getException**());  
 Toast.***makeText***(MainActivity.**this**, **"Authentication failed."**,  
 Toast.***LENGTH\_SHORT***).**show**();  
 }  
  
 }  
});