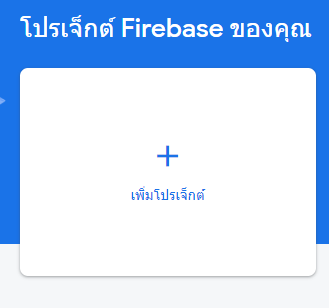
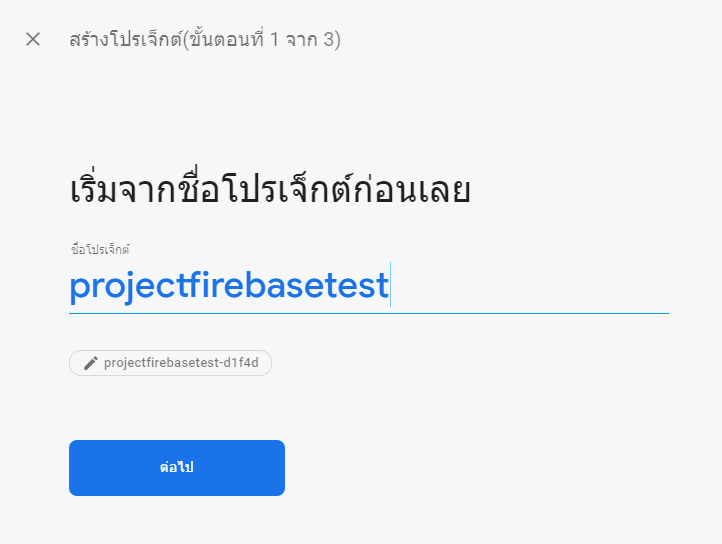
**ติดตั้ง Firebase**

<https://console.firebase.google.com/?hl=th>

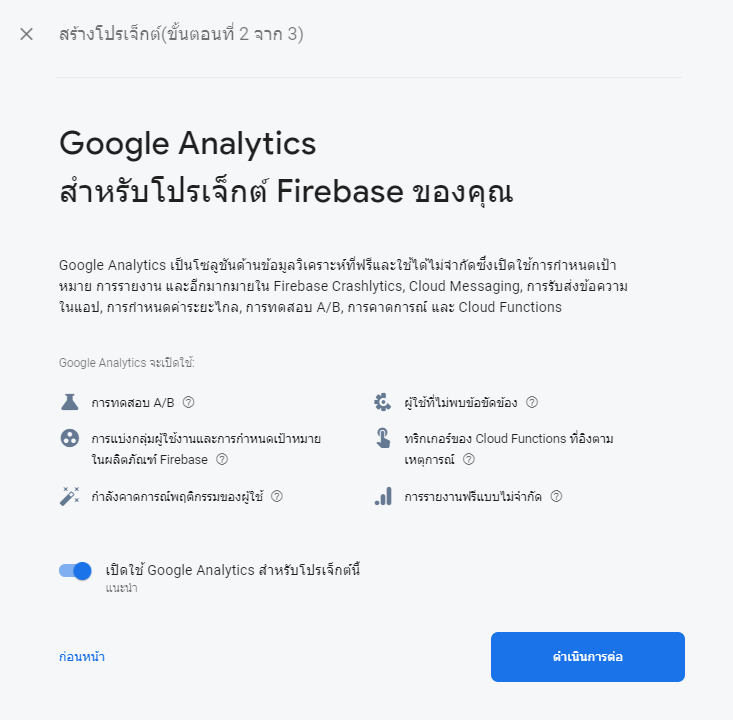
ทำการสร้าง Project Firebase

****

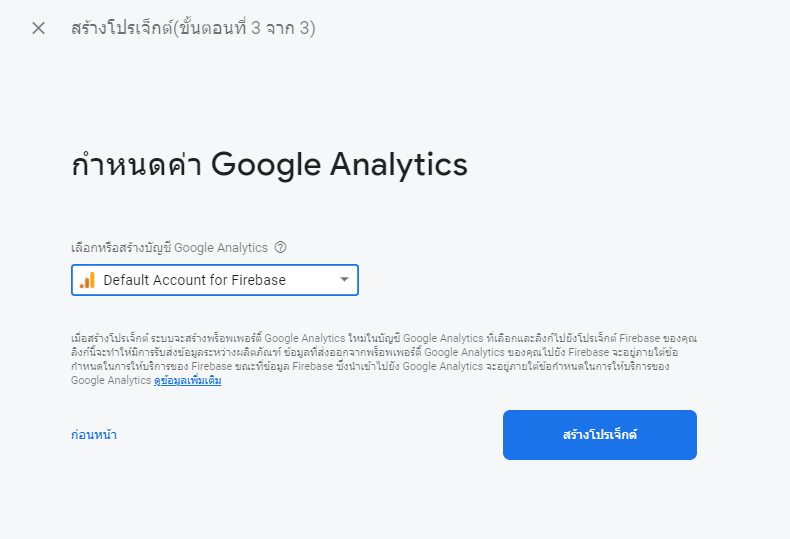
ตั้งชื่อ Project



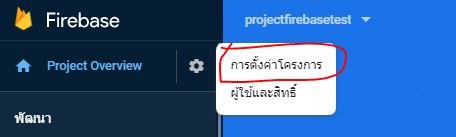
กดดำเนินการต่อ



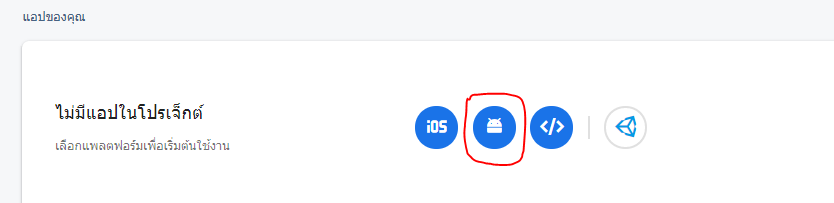
เลือก Default Account for Firebase แล้วกดสร้างโปรเจ็กต์



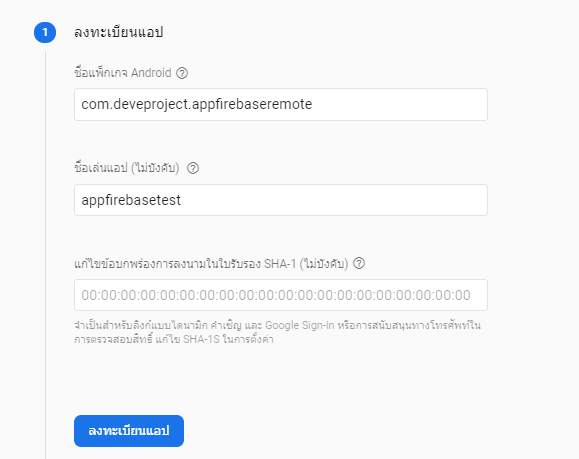
เมื่อสร้างเสร็จแล้ว ให้ไปหน้า การตั้งค่าโครงการ



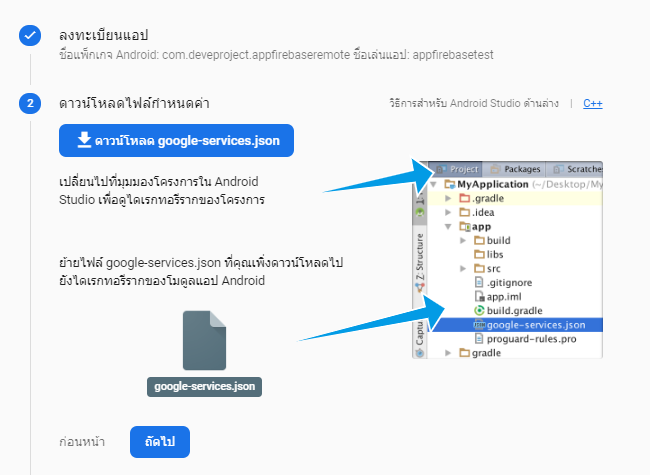
ทำการเพิ่มแอปโปรเจ็กต์



ใส่ชื่อแพ็กเกจและชื่อแอป แล้วกดลงทะเบียนแอป

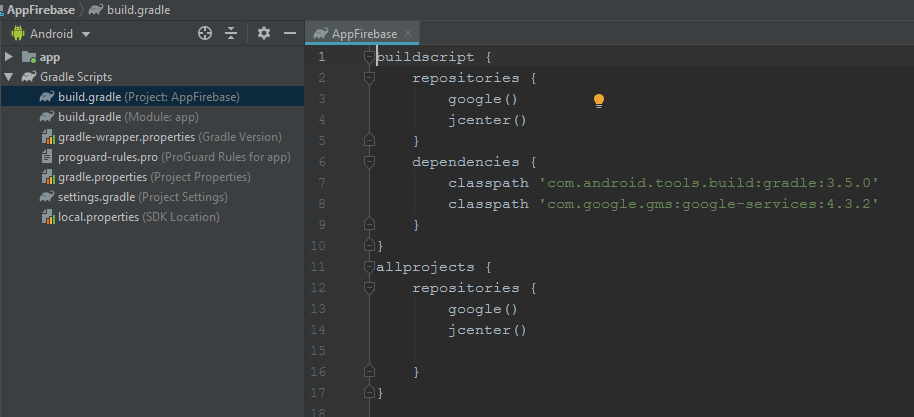


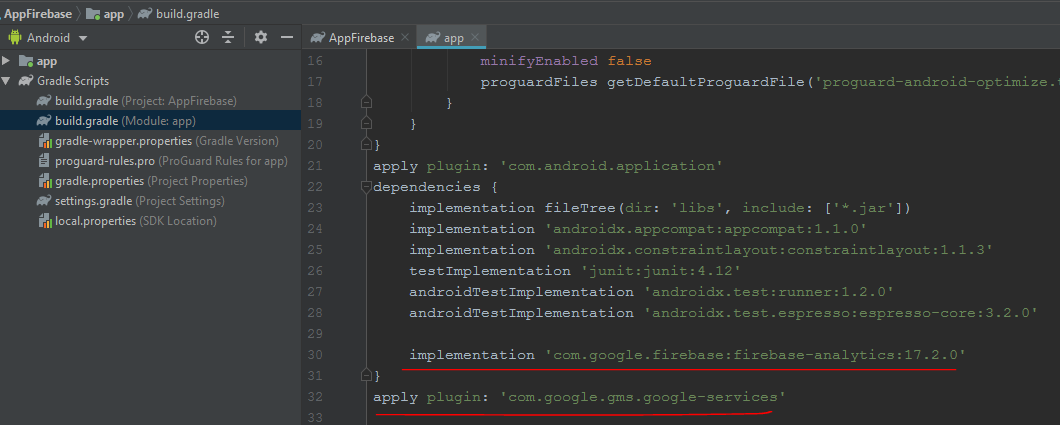
ดาวโหลดไฟล์ google-services.json แล้วเอาไปใส่ไว้ตรงตำแหน่งที่ภาพบอก เสร็จแล้วกดถัดไป



ทำตามคำแนะนำไปเรื่อยๆ







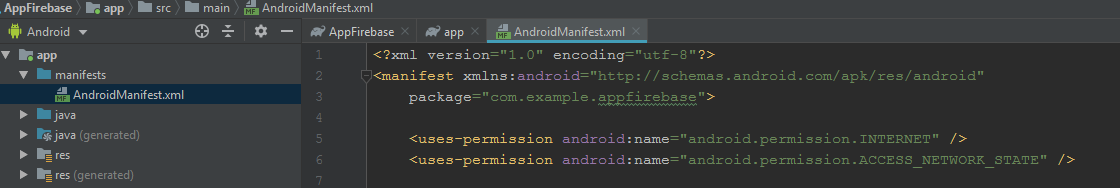
เมื่อถึงขั้นตอนที่ 4 ให้เราทำการ run app ลงในมือถือ หรือ emulator ของเรา หรือจะกด ข้ามขั้นตอนนี้ ไปก่อนก็ได้

เสร็จสิ้นการติดตั้ง firebase ลงใน App android

อย่าลืมประกาศ Permission สำหรับใช้งาน Internet ใน AndroidManifest.xml

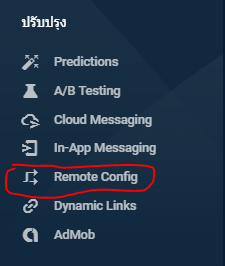
<uses-permission android:name="android.permission.INTERNET" />

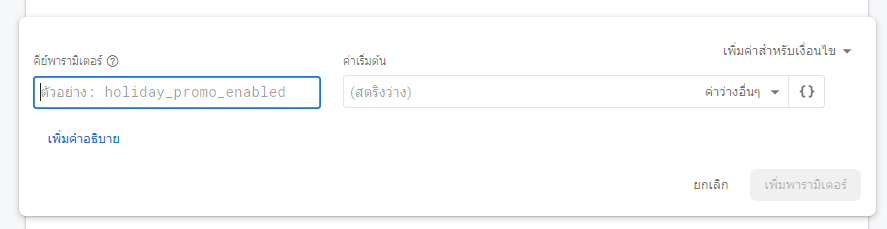
<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />



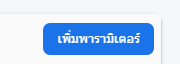
**ติดตั้ง Firebase Remote**

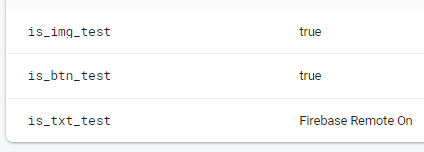
เลือก Remote Config



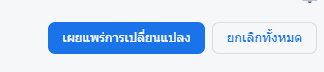


กดเพิ่ม พารามิเตอร์



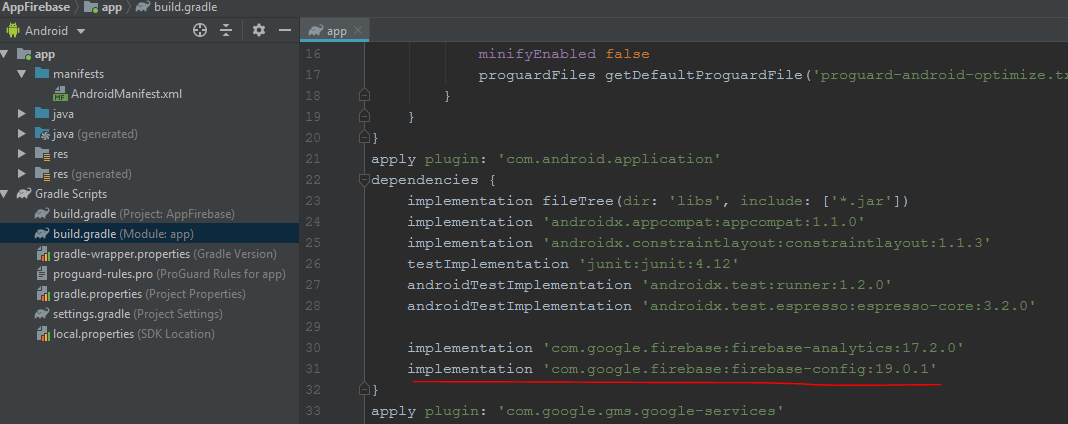


เมื่อเพิ่มพารามิเตอร์เสร็จแล้วให้กด เผยแพร่การเปลี่ยนแปลง



เพิ่มคำสั่ง firebase-config ไว้ใน dependencies และทำการ Sync

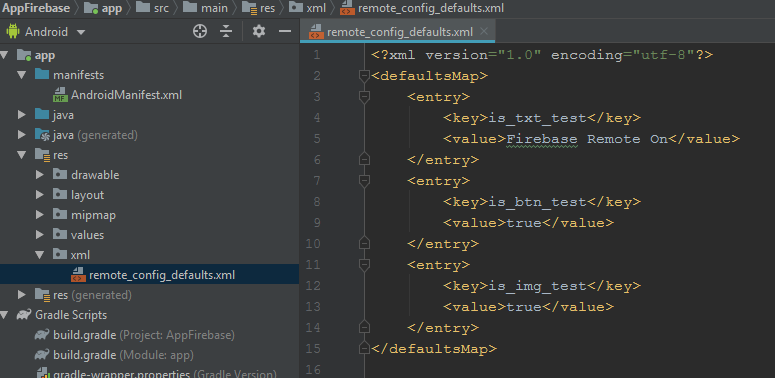
implementation 'com.google.firebase:firebase-config:19.0.1'



ไปที่ Code app ของเรา ทำการสร้างไฟล์ remote\_config\_defaults.xml ที่โฟลเดอร์ res/xml เพื่อประกาศและเก็บค่าเริ่มต้น

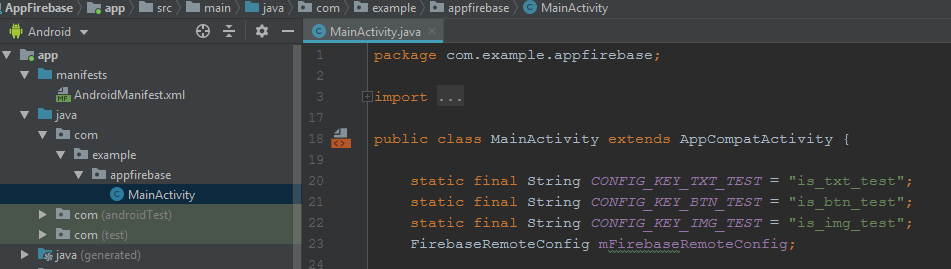
โดยที่ค่า parameter และ value จะต้องตรงกับที่เราตั้งค่าไว้ใน firebase remote config

<?xml version="1.0" encoding="utf-8"?>  
<defaultsMap>  
  
 <entry>  
 <key>is\_txt\_test</key>  
 <value>Firebase Remote On</value>  
 </entry>  
  
 <entry>  
 <key>is\_btn\_test</key>  
 <value>true</value>  
 </entry>  
  
 <entry>  
 <key>is\_img\_test</key>  
 <value>true</value>  
 </entry>  
  
</defaultsMap>



ทำการประกาศตัวแปร FirebaseRemoteConfig และ Key Config Remoteกันก่อน

static final String *CONFIG\_KEY\_TXT\_TEST* = "is\_txt\_test";  
static final String *CONFIG\_KEY\_BTN\_TEST* = "is\_btn\_test";  
static final String *CONFIG\_KEY\_IMG\_TEST* = "is\_img\_test";  
FirebaseRemoteConfig mFirebaseRemoteConfig;



ตามด้วยการประกาศรับค่า Instance ของ FirebaseRemoteConfig

mFirebaseRemoteConfig = FirebaseRemoteConfig.*getInstance*();

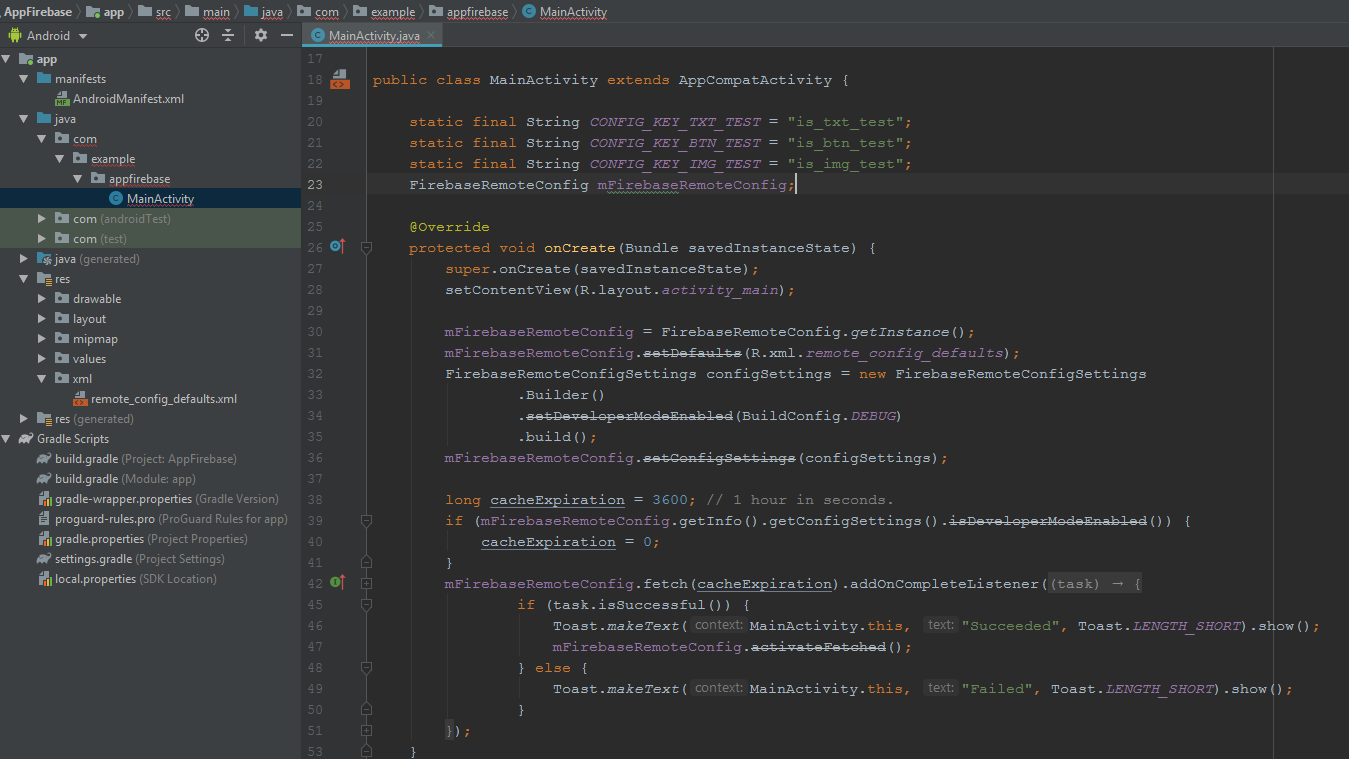
mFirebaseRemoteConfig.setDefaults(R.xml.*remote\_config\_defaults*);

การดึงค่า config จาก server นั้น ปกติ 1 ชม. จะจำกัดได้แค่ 5 ครั้ง ซึ่งเวลาเราพัฒนาอยู่ มันจะต้องเกินกว่า 5 ครั้งแน่นอน ดังนั้นเราจึงจำเป็นต้อง enable developer mode เพื่อที่จะ request ได้เรื่อยๆ โดยแค่วาง source code ด้านล่างนี้ต่อจากการ getInstance ได้เลย

FirebaseRemoteConfigSettings configSettings = new FirebaseRemoteConfigSettings  
 .Builder()  
 .setDeveloperModeEnabled(BuildConfig.*DEBUG*)  
 .build();  
mFirebaseRemoteConfig.setConfigSettings(configSettings);

เรื่อง .fetch() ที่มีค่าเริ่มต้นที่ 12 ชม. นั้นก็นานเกินไป นานเกินกว่าที่เราจะรอได้ในการพัฒนา ดังนั้นให้เพื่อให้เราดึงค่าแบบ realtime ได้ในขณะพัฒนา ให้ตั้งค่า expiration ของ cache ให้เป็น 0 แต่ก็ควรเช็คด้วยว่าเราอยู่ใน developer mode แล้วจึงค่อยดำเนินการ

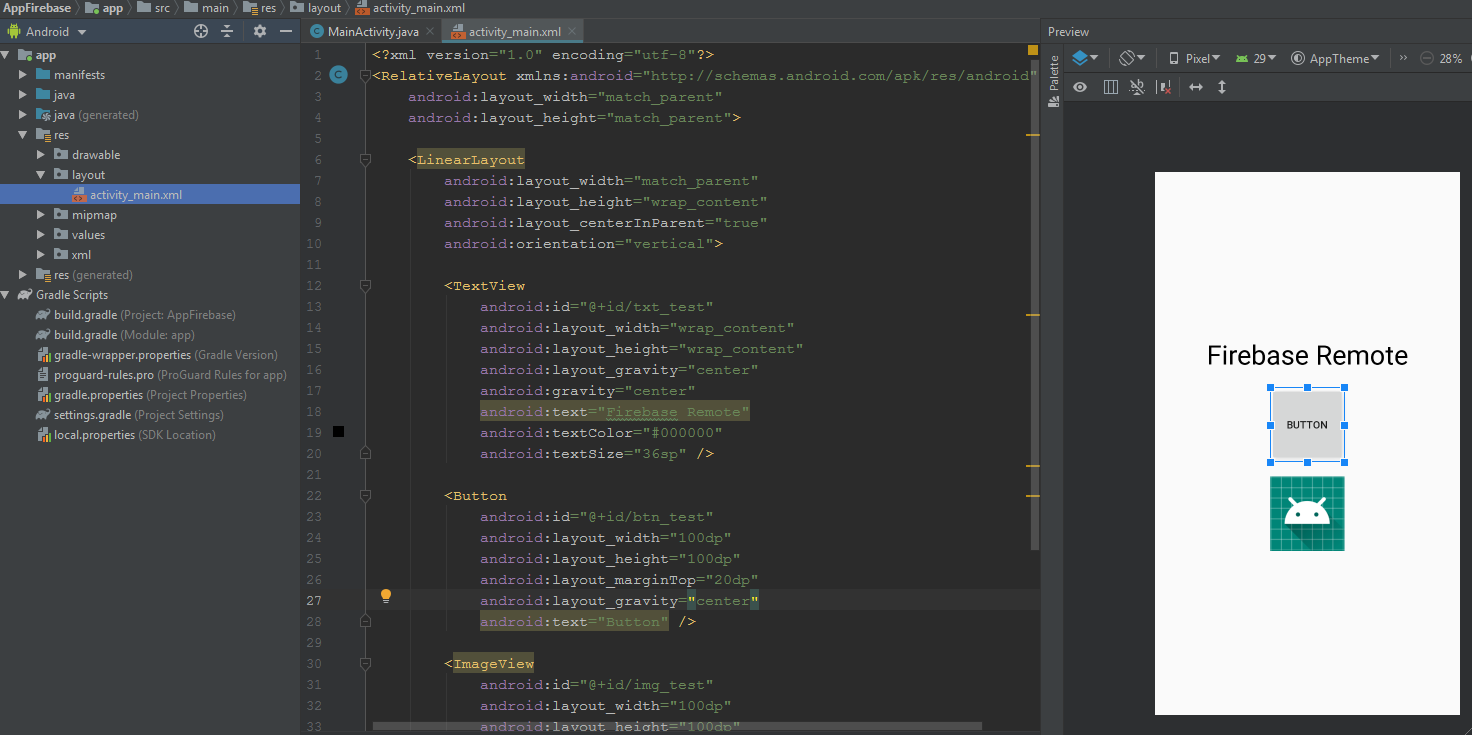
long cacheExpiration = 3600; // 1 hour in seconds.  
if (mFirebaseRemoteConfig.getInfo().getConfigSettings().isDeveloperModeEnabled()) {  
 cacheExpiration = 0;  
}  
mFirebaseRemoteConfig.fetch(cacheExpiration).addOnCompleteListener(new OnCompleteListener<Void>() {  
 @Override  
 public void onComplete(@NonNull Task<Void> task) {  
 if (task.isSuccessful()) {  
 Toast.*makeText*(MainActivity.this, "Succeeded", Toast.*LENGTH\_SHORT*).show();  
 mFirebaseRemoteConfig.activateFetched();  
 } else {  
 Toast.*makeText*(MainActivity.this, "Failed", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
});



เราลองมาทดสอบกันดูว่าจะสามารถใช้ Firebase Remote Config กันได้หรือไม่

ให้เราแก้ไขไฟล์ layout/activity\_main.xml ทำการประกาศ View ดังนี้ เพื่อทำการทดสอบ

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerInParent="true"  
 android:orientation="vertical">  
  
 <TextView  
 android:id="@+id/txt\_test"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:gravity="center"  
 android:text="Firebase Remote"  
 android:textColor="#000000"  
 android:textSize="36sp" />  
  
 <Button  
 android:id="@+id/btn\_test"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"  
 android:layout\_marginTop="20dp"  
 android:layout\_gravity="center"  
 android:text="Button" />  
  
 <ImageView  
 android:id="@+id/img\_test"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"  
 android:layout\_marginTop="20dp"  
 android:layout\_gravity="center"  
 android:src="@mipmap/ic\_launcher" />  
  
 </LinearLayout>  
</RelativeLayout>

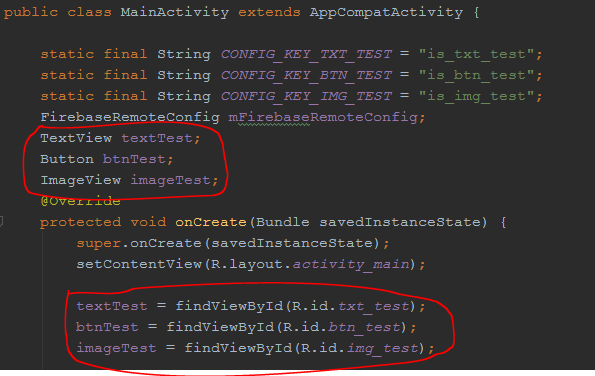


ประกาศตัวแปรให้ไฟล์ Java

TextView textTest;  
Button btnTest;  
ImageView imageTest;

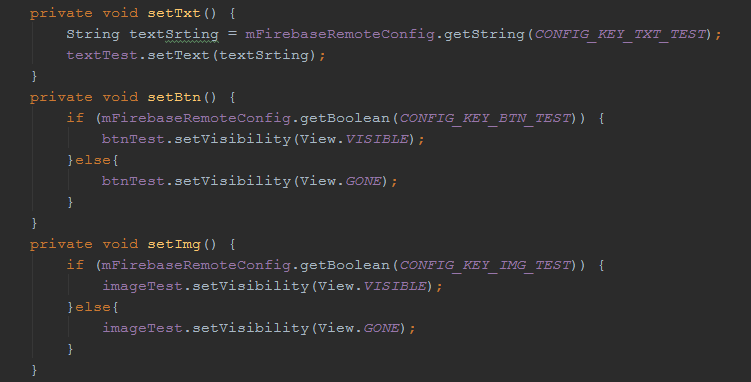
ประกาศใน onCreate

textTest = findViewById(R.id.*txt\_test*);  
btnTest = findViewById(R.id.*btn\_test*);  
imageTest = findViewById(R.id.*img\_test*);



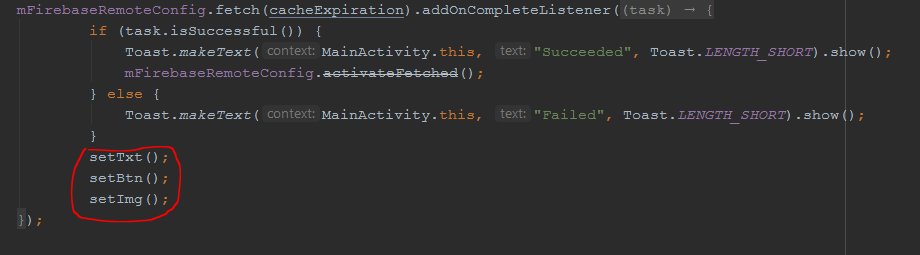
สร้างฟังก์ชั่นของตัวแปรที่เราจะทำการทดสอบ

private void setTxt() {  
 String textSrting = mFirebaseRemoteConfig.getString(*CONFIG\_KEY\_TXT\_TEST*);  
 textTest.setText(textSrting);  
}  
private void setBtn() {  
 if (mFirebaseRemoteConfig.getBoolean(*CONFIG\_KEY\_BTN\_TEST*)) {  
 btnTest.setVisibility(View.*VISIBLE*);  
 }else{  
 btnTest.setVisibility(View.*GONE*);  
 }  
}  
private void setImg() {  
 if (mFirebaseRemoteConfig.getBoolean(*CONFIG\_KEY\_IMG\_TEST*)) {  
 imageTest.setVisibility(View.*VISIBLE*);  
 }else{  
 imageTest.setVisibility(View.*GONE*);  
 }  
}

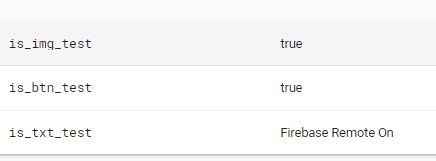


นำฟังก์ชั่นที่เราสร้างไปใส่ไว้ใน mFirebaseRemoteConfig.fetch ดังนี้

mFirebaseRemoteConfig.fetch(cacheExpiration).addOnCompleteListener(new OnCompleteListener<Void>() {  
 @Override  
 public void onComplete(@NonNull Task<Void> task) {  
 if (task.isSuccessful()) {  
 Toast.*makeText*(MainActivity.this, "Succeeded", Toast.*LENGTH\_SHORT*).show();  
 mFirebaseRemoteConfig.activateFetched();  
 } else {  
 Toast.*makeText*(MainActivity.this, "Failed", Toast.*LENGTH\_SHORT*).show();  
 }  
 setTxt();  
 setBtn();  
 setImg();  
 }  
});



เสร็จแล้วทำการ Run app ดู จะขึ้นแบบนี้



ลองทำการแก้ไขดู จะขึ้นดังภาพนี้

