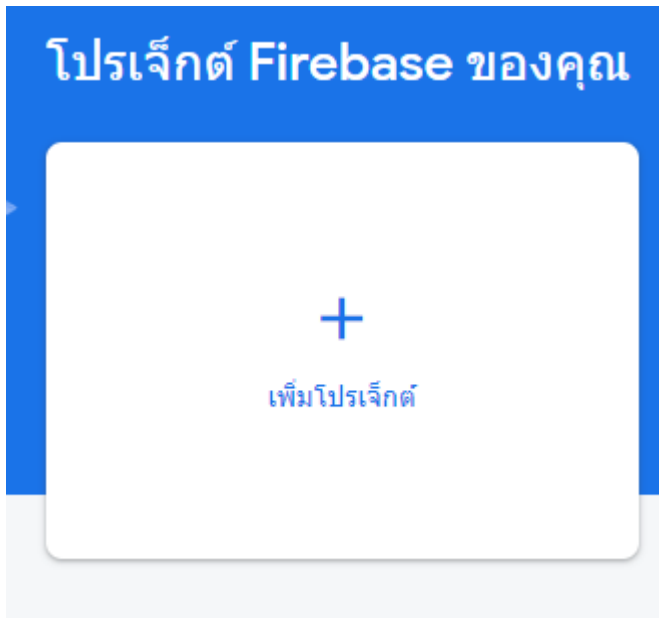




## ติดตั้ง Firebase

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

ทำการสร้าง Project Firebase



ตั้งชื่อ Project




× สร้างโปรเจกต์(ขั้นตอนที่ 1 จาก 3)

## เริ่มจากชื่อโปรเจกต์ก่อนเลย

ชื่อโปรเจกต์

projectfirebasetest

 projectfirebasetest-d1f4d

ต่อไป

กดดำเนินการต่อ



× สร้างโปรเจกต์(ขั้นตอนที่ 2 จาก 3)

# Google Analytics

## สำหรับโปรเจกต์ Firebase ของคุณ

Google Analytics เป็นโซลูชันด้านข้อมูลวิเคราะห์ที่ฟรีและใช้ได้ไม่จำกัดซึ่งเปิดใช้การกำหนดเป้าหมาย การรายงาน และอีกมากมายใน Firebase Crashlytics, Cloud Messaging, การรับส่งข้อความในแอป, การกำหนดค่าระยะไกล, การทดสอบ A/B, การคาดการณ์ และ Cloud Functions

Google Analytics จะเปิดใช้:



การทดสอบ A/B ①



ผู้ใช้ที่ไม่พบข้อขัดข้อง ①



การแบ่งกลุ่มผู้ใช้งานและการกำหนดเป้าหมาย  
ในผลิตภัณฑ์ Firebase ①



ทริกเกอร์ของ Cloud Functions ที่อิงตาม  
เหตุการณ์ ①



กำลังคาดการณ์พฤติกรรมของผู้ใช้ ①



การรายงานฟรีแบบไม่จำกัด ①



เปิดใช้ Google Analytics สำหรับโปรเจกต์นี้  
ขณะนี้

[ก่อนหน้า](#)

[ดำเนินการต่อ](#)




เลือก Default Account for Firebase แล้วกดสร้างโปรเจกต์

× สร้างโปรเจกต์(ขั้นตอนที่ 3 จาก 3)

## กำหนดค่า Google Analytics

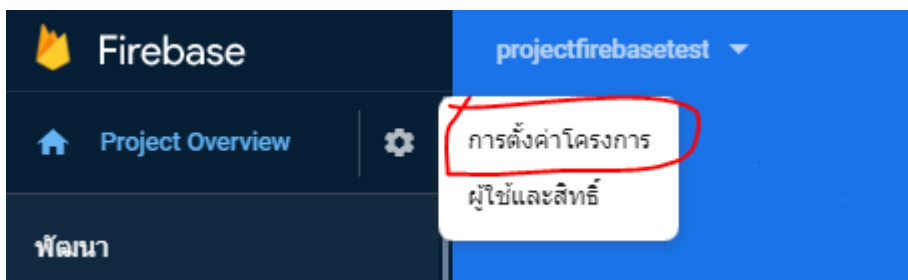
เลือกหรือสร้างบัญชี Google Analytics ⑦

 Default Account for Firebase

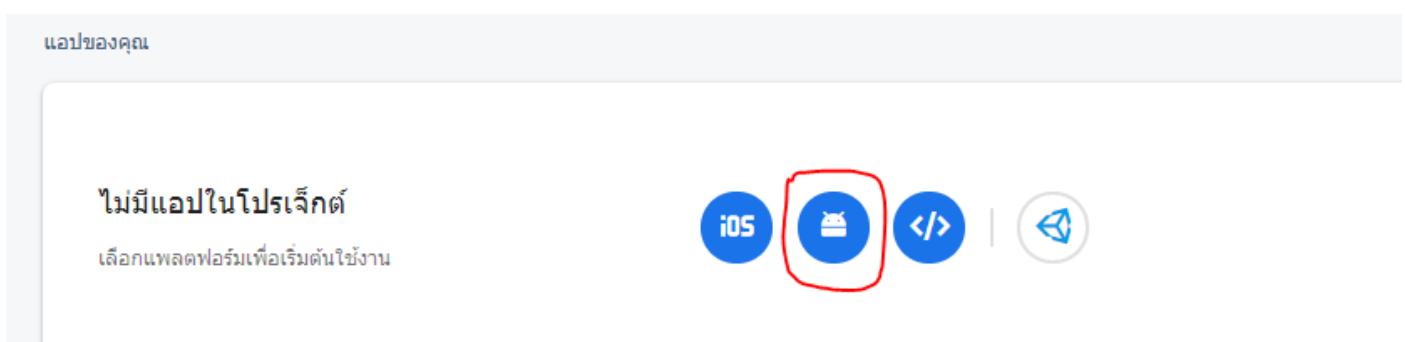
เมื่อสร้างโปรเจกต์ ระบบจะสร้างพร็อพเพอร์ตี้ Google Analytics ใหม่ในบัญชี Google Analytics ที่เลือกและลิงก์ไปยังโปรเจกต์ Firebase ของคุณ  
ลิงก์นี้จะทำให้การรับส่งข้อมูลระหว่างผลิตภัณฑ์ ข้อมูลที่ส่งออกจากพร็อพเพอร์ตี้ Google Analytics ของคุณไปยัง Firebase จะอยู่ภายใต้ข้อ  
กำหนดในการให้บริการของ Firebase ขณะที่มีข้อมูล Firebase ซึ่งนำเข้าไปยัง Google Analytics จะอยู่ภายใต้ข้อกำหนดในการให้บริการของ  
Google Analytics [ดูข้อมูลเพิ่มเติม](#)

[ก่อนหน้า](#) [สร้างโปรเจกต์](#)

เมื่อสร้างเสร็จแล้ว ให้ไปหน้า การตั้งค่าโครงการ



ทำการเพิ่มแอปโปรเจกต์





ใส่ชื่อแพ็กเกจและชื่อแอป แล้วกดลงทะเบียนแอป

1

ลงทะเบียนแอป

ชื่อแพ็กเกจ Android ⓘ

com.deveproject.appfirebaseremote

ชื่อเล่นแอป (ไม่บังคับ) ⓘ

appfirebasetest

แก้ไขข้อบกพร่องการลงนามในใบรับรอง SHA-1 (ไม่บังคับ) ⓘ

00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

จำเป็นสำหรับลิงก์แบบไดนามิก สำคัญ และ Google Sign-In หรือการสนับสนุนทางโทรศัพท์ในการตรวจสอบสิทธิ์ แก้ไข SHA-1S ในการตั้งค่า

ลงทะเบียนแอป

ดาวน์โหลดไฟล์ google-services.json แล้วเอาไปใส่ไว้ตรงตำแหน่งที่ภาพบอก เสร็จแล้วกดถัดไป



## ลงทะเบียนแอป

ชื่อแพ็คเกจ Android: com.deveproject.appfirebaseremote ชื่อเล่นแอป: appfirebasetest

2

## ดาวน์โหลดไฟล์กำหนดค่า

วิธีการสำหรับ Android Studio ด้านล่าง | [C++](#)



ดาวน์โหลด google-services.json

เปลี่ยนไปที่มุมมองโครงการใน Android Studio เพื่อดูไดเรกทอรีรากของโครงการ

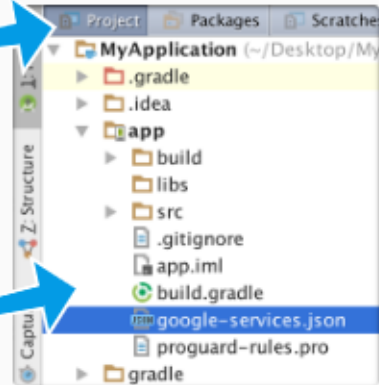
ย้ายไฟล์ google-services.json ที่คุณเพิ่งดาวน์โหลดไปยังไดเรกทอรีรากของโมดูลแอป Android



google-services.json

ก่อนหน้า

ถัดไป



ทำตามคำแนะนำไปเรื่อยๆ



build.gradle ระดับโครงการ (<โครงการ>/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:4.3.2'
    }
}

allprojects {
    ...
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
        ...
    }
}
```

build.gradle ระดับแอป (<โครงการ>/<app-module>/build.gradle):

```
apply plugin: 'com.android.application'

dependencies {
    // add the Firebase SDK for Google Analytics
    implementation 'com.google.firebase:firebase-analytics:17.2.0'
    // add SDKs for any other desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
...
// Add to the bottom of the file
apply plugin: 'com.google.gms.google-services'
```

สุดท้าย ให้กด "ซิงค์เลย" ในแถบที่ปรากฏใน IDE

Gradle files have changed since last build

[Sync now](#)



```
AppFirebase > build.gradle
Android
  app
    Gradle Scripts
      build.gradle (Project: AppFirebase)
      build.gradle (Module: app)
      gradle-wrapper.properties (Gradle Version)
      proguard-rules.pro (ProGuard Rules for app)
      gradle.properties (Project Properties)
      settings.gradle (Project Settings)
      local.properties (SDK Location)
1  buildscript {
2      repositories {
3          google()
4          jcenter()
5      }
6      dependencies {
7          classpath 'com.android.tools.build:gradle:3.5.0'
8          classpath 'com.google.gms:google-services:4.3.2'
9      }
10 }
11 allprojects {
12     repositories {
13         google()
14         jcenter()
15     }
16 }
17 }
```

```
AppFirebase > app > build.gradle
Android
  app
    Gradle Scripts
      build.gradle (Project: AppFirebase)
      build.gradle (Module: app)
      gradle-wrapper.properties (Gradle Version)
      proguard-rules.pro (ProGuard Rules for app)
      gradle.properties (Project Properties)
      settings.gradle (Project Settings)
      local.properties (SDK Location)
16 minifyEnabled false
17 proguardFiles getDefaultProguardFile('proguard-android-optimize.
18 }
19 }
20 }
21 apply plugin: 'com.android.application'
22 dependencies {
23     implementation fileTree(dir: 'libs', include: ['*.jar'])
24     implementation 'androidx.appcompat:appcompat:1.1.0'
25     implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
26     testImplementation 'junit:junit:4.12'
27     androidTestImplementation 'androidx.test:runner:1.2.0'
28     androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'
29
30     implementation 'com.google.firebase:firebase-analytics:17.2.0'
31 }
32 apply plugin: 'com.google.gms.google-services'
33 }
```

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" />
```

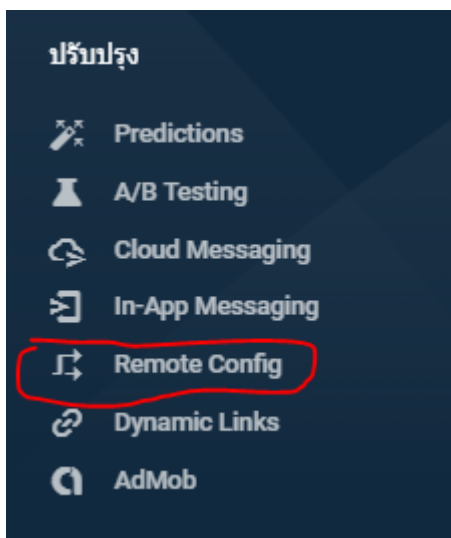




```
AppFirebase > app > src > main > AndroidManifest.xml
AndroidManifest.xml
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.appfirebase">
4
5     <uses-permission android:name="android.permission.INTERNET" />
6     <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
7
```

## ติดตั้ง Firebase Remote

เลือก Remote Config



คีย์พารามิเตอร์ ⓘ

ค่าเริ่มต้น

เพิ่มค่าสำหรับเงื่อนไข ▼

ตัวอย่าง: holiday\_promo\_enabled

(สตริงว่าง)

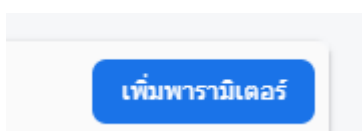
ค่าว่างอื่นๆ ▼ {}

เพิ่มคำอธิบาย

ยกเลิก

เพิ่มพารามิเตอร์

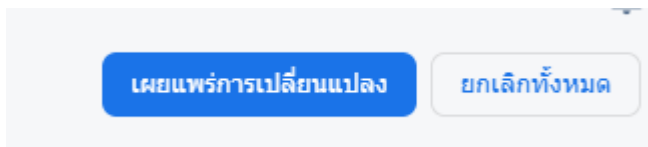
กดเพิ่ม พารามิเตอร์





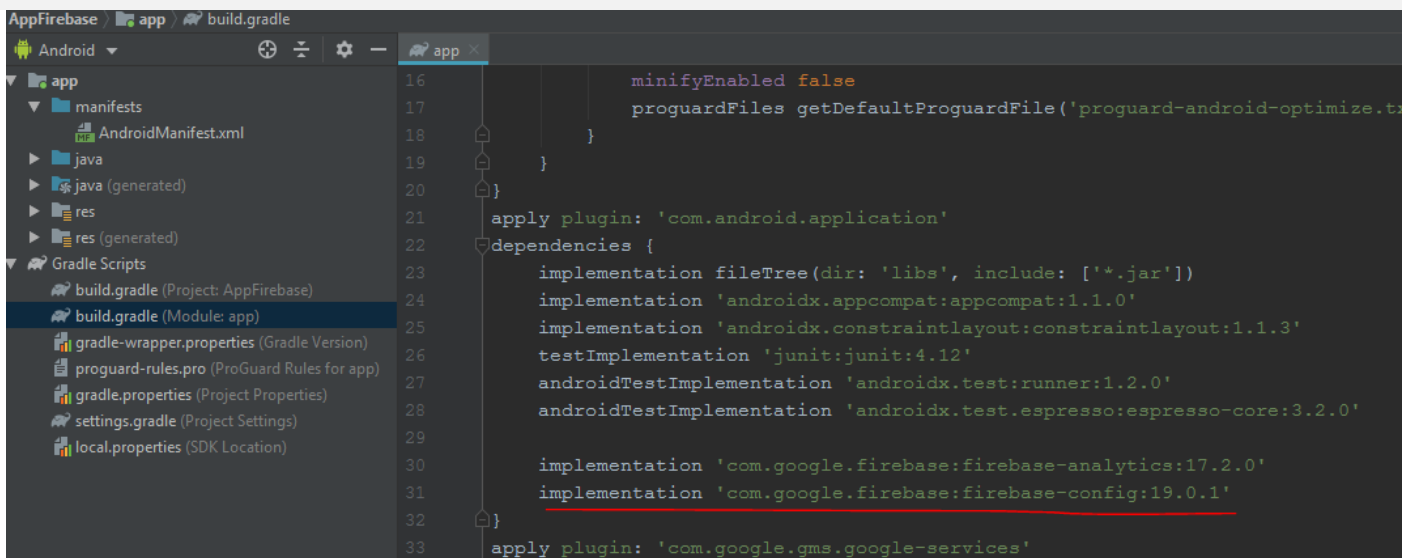
is_img_test	true
is_btn_test	true
is_txt_test	Firebase Remote On

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>

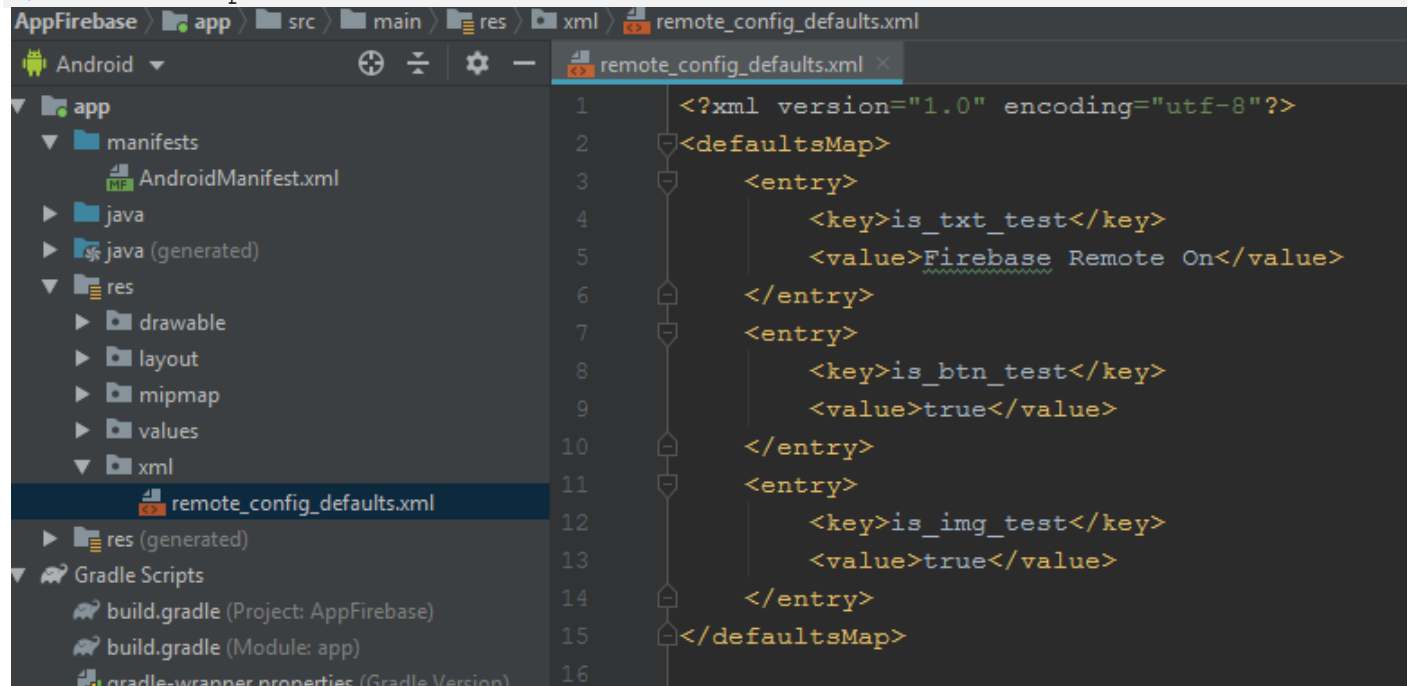
    <entry>
```



```
<key>is_btn_test</key>
<value>true</value>
</entry>

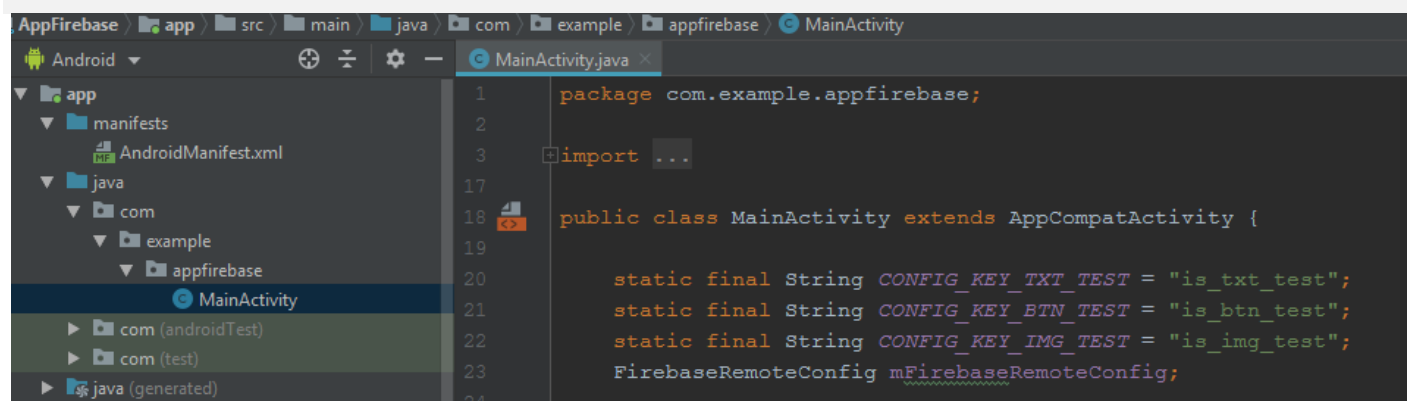
<entry>
  <key>is_img_test</key>
  <value>true</value>
</entry>
```

```
</defaultsMap>
```



ทำการประกาศตัวแปร `FirebaseRemoteConfig` และ `Key Config Remote` กันก่อน

```
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 โดยแควง **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** ได้  
ในขณะที่พัฒนา 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();  
        }  
    }  
});
```



```
AppFireBase / app / src / main / java / com / example / appfirebase / MainActivity
Android ▾
  app
    manifests
      AndroidManifest.xml
    java
      com
        example
          appfirebase
            MainActivity
            com (androidTest)
            com (test)
    java (generated)
    res
      drawable
      layout
      mipmap
      values
      xml
        remote_config_defaults.xml
    res (generated)
  Gradle Scripts
    build.gradle (Project: AppFireBase)
    build.gradle (Module: app)
    gradle-wrapper.properties (Gradle Version)
    proguard-rules.pro (ProGuard Rules for app)
    gradle.properties (Project Properties)
    settings.gradle (Project Settings)
    local.properties (SDK Location)

17
18 public class MainActivity extends AppCompatActivity {
19
20     static final String CONFIG_KEY_TXT_TEST = "is_txt_test";
21     static final String CONFIG_KEY_BTN_TEST = "is_btn_test";
22     static final String CONFIG_KEY_IMG_TEST = "is_img_test";
23     FirebaseRemoteConfig mFirebaseRemoteConfig;
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27         super.onCreate(savedInstanceState);
28         setContentView(R.layout.activity_main);
29
30         mFirebaseRemoteConfig = FirebaseRemoteConfig.getInstance();
31         mFirebaseRemoteConfig.setDefaults(R.xml.remote_config_defaults);
32         FirebaseRemoteConfigSettings configSettings = new FirebaseRemoteConfigSettings
33             .Builder()
34             .setDeveloperModeEnabled(BuildConfig.DEBUG)
35             .build();
36         mFirebaseRemoteConfig.setConfigSettings(configSettings);
37
38         long cacheExpiration = 3600; // 1 hour in seconds.
39         if (mFirebaseRemoteConfig.getInfo().getConfigSettings().isDeveloperModeEnabled()) {
40             cacheExpiration = 0;
41         }
42         mFirebaseRemoteConfig.fetch(cacheExpiration).addOnCompleteListener((task) -> {
43             if (task.isSuccessful()) {
44                 Toast.makeText(context: MainActivity.this, text: "Succeeded", Toast.LENGTH_SHORT).show();
45                 mFirebaseRemoteConfig.activateFetched();
46             } else {
47                 Toast.makeText(context: MainActivity.this, text: "Failed", Toast.LENGTH_SHORT).show();
48             }
49         });
50     }
51 }
52
53 }
```

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"
```



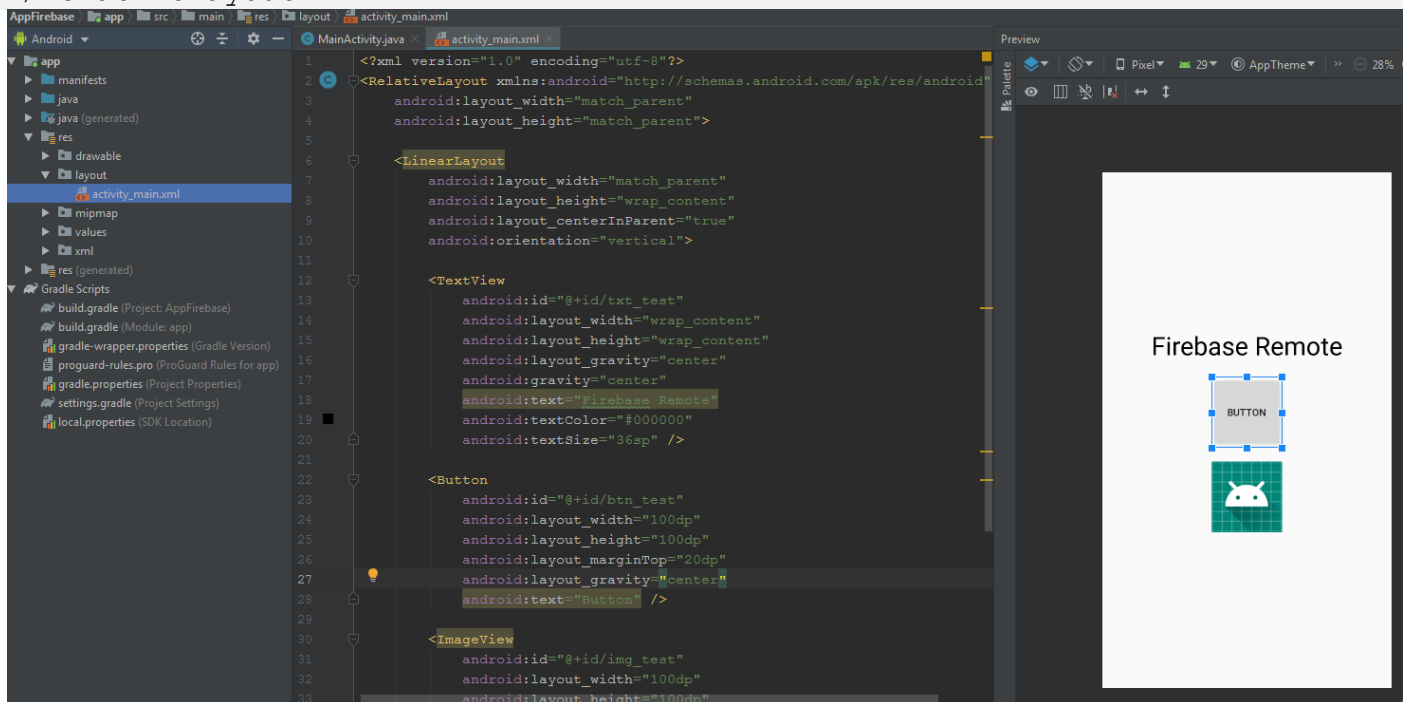
```
android:text="Firebase Remote"
android:textColor="#000000"
android:textSize="36sp" />
```

```
<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);
```



```
public class MainActivity extends AppCompatActivity {

    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;
    TextView textTest;
    Button btnTest;
    ImageView imageTest;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        textTest = findViewById(R.id.txt_test);
        btnTest = findViewById(R.id.btn_test);
        imageTest = findViewById(R.id.img_test);
    }
}
```

สร้างฟังก์ชันของตัวแปรที่เราจะทำการทดสอบ

```
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);
    }
}
}
```



```
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);
    }
}
```

นำฟังก์ชันที่เราสร้างไปใส่ไว้ใน mFirebaseRemoteConfig.fetch ดังนี้

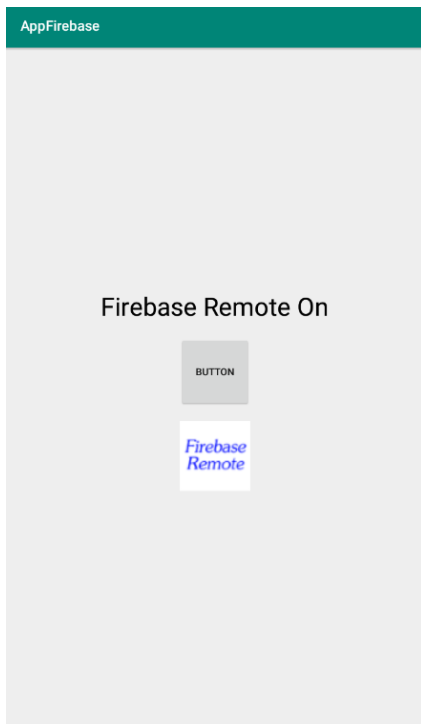
```
mFirebaseRemoteConfig.fetch(cacheExpiration).addOnCompleteListener(new
OnCompleteListener<Void>() {
    @Override
    public void onComplete(@NonNull Task<Void> task) {
        if (task.isSuccessful()) {
            Toast.makeText(MainActivity.this, "Succeeded",
Toast.LENGTH_SHORT).show();
     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





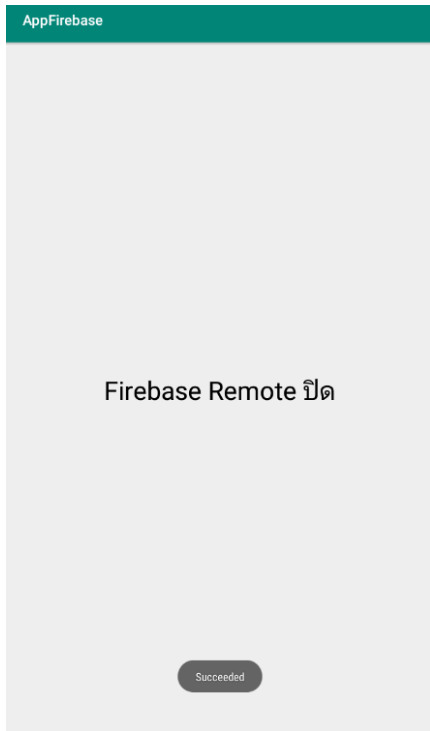
```
mFirebaseRemoteConfig.fetch(cacheExpiration).addOnCompleteListener((task) -> {  
    if (task.isSuccessful()) {  
        Toast.makeText(context: MainActivity.this, text: "Succeeded", Toast.LENGTH_SHORT).show();  
        mFirebaseRemoteConfig.activateFetched();  
    } else {  
        Toast.makeText(context: MainActivity.this, text: "Failed", Toast.LENGTH_SHORT).show();  
    }  
      
    setTxt();  
    setBtn();  
    setImg();  
});
```

เสร็จแล้วทำการ Run app ดู จะขึ้นแบบนี้



is_img_test	true
is_btn_test	true
is_txt_test	Firebase Remote On

ลองทำการแก้ไขดู จะขึ้นดังภาพนี้



is_img_test	false
is_btn_test	false
is_txt_test	Firebase Remote ปิด