

---

## **MORE APP Tracker SDK**

---

*Version 0.16.03.28*

---

*2016.03.28*

---

# Table of Contents

MORE APP Tracker SDK.....	1
<i>Version 0.16.03.28</i> .....	1
Table of Contents .....	2
1. Introduction.....	3
1.1. Overview .....	3
2. Service.....	4
2.1. Event Tracking Service .....	4
2.1.1. Use SDK Library .....	4
2.1.2. Tracker Function .....	7
3. Parameter Definition.....	7
3.1. OnCallbackResult Function Parameter.....	7
3.2. Handler 與 Callback listener 參數對應.....	8

# **1. Introduction**

## **1.1. Overview**

Ideas APP Tracker SDK 主要提供 Smart Mobile Device 更快速與簡潔的開發模組，APP 開發人員只要將 Ideas APP Tracker SDK 加入到自己的 APP 專案裡，透過 API 的呼叫即可執行 APP 使用者行為紀錄服務，目前提供 Android 版本 SDK Library。

## 2. Service

### 2.1. Event Tracking Service

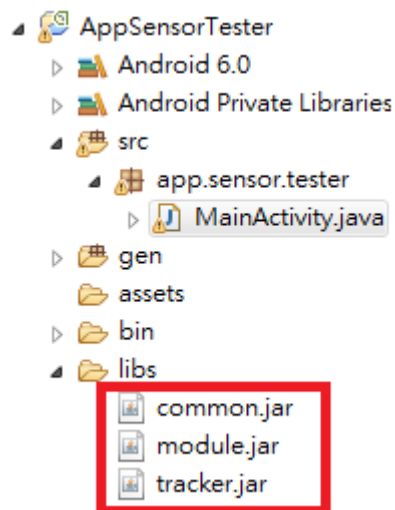
More Event Tracking Service 主要提供 APP 傳送使用者資料與使用行為資料，結合 S.E.R.的分類，會將 APP 傳送過來的資料歸類為不同的類別來分析，透過 Tracker 模組，Tracker API 會呼叫 S.E.R. API 將資料儲存於 S.E.R. Server，目前 More Event Tracking Service 會依各家廠商希望分析的資料放入 Tracker 中。

#### 2.1.1. Use SDK Library

Tracker 在傳送 Event 資料前，須先將 More SDK Library 加入到自己的 APP 專案，API 使用時先呼叫 startTracker Function，startTracker 執行成功後，即可開始呼叫 track function 傳送 Event 資料，如果要停止服務則呼叫 stopTracker function。

使用步驟：

1. 先將 tracker.zip 中的 JAR 檔加入到 Android 開發專案的 libs 資料夾



2. 在 AndroidManifest.xml 加入以下權限

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.GET_ACCOUNTS" />
```

3. import tracker class

```
import sdk.ideas.common.OnCallbackResult;
import sdk.ideas.tracker.Tracker;
```

4. 本 SDK 提供 Handler 和 Callback Listener 方法來去知道程式是否有成功

執行，開發者可以以自己喜好去做設定，以下是 Handler 方法範例

1) 先宣告Handler

```
private Handler theHandler = new Handler()
{
    @Override
    public void handleMessage(Message msg)
    {
        switch (msg.what)
        {
            Case 1029:
                Log.d("handler response: ", "Result: " + String.valueOf(msg.arg1) + " From: "
                    + String.valueOf(msg.arg2) + " Message: " +
                    ((HashMap<String,String>)msg.obj) );
                break;
            default:
                break;
        }
    }
};
```

2) 設定此 handler

```
tracker.setHandler(theHandler);
```

5. Callback Listener 方式

```
tracker.setOnCallbackResultListener(new OnCallbackResult()
{
    @Override
    public void onCallbackResult(int result, int what, int from, HashMap<String,
String> message)
    {
        Log.d("Listener response: ", "Result: " + String.valueOf(result) + " What:
" + String.valueOf(what)
            + " From: " + String.valueOf(from) + " " + message);
    }
});
```

6. 請先 startTracker

```
startTracker(String app_id)
```

7. 在要傳送資料處執行tracker，例如以下Sample，可依開發者想分析的資料進行彈性化調整。

```
HashMap<String, String> parm = new HashMap<String,
String>();
    parm.put("TYPE", type);
    parm.put("SOURCE_FROM", source_from);
    parm.put("PAGE", page);
    parm.put("PRODUCTION", production);
    parm.put("PRICE", price);
tracker.track(parm);
```

### 2.1.2. Tracker Function

Event Tracking API	
Function	Description
<b>void</b> startTracker(String app_id)	Tracking 服務開始，登入 APP ID
<b>void</b> startTracker(String app_id, String fb_id, String fb_name, String fb_email)	如果有 Facebook id、nickname 與 Email 則可用此 function。
<b>void</b> track(HashMap<String, String> parm)	傳送 Tracking 資料
<b>void</b> stopTracker()	Tracking 服務停止
<b>void</b> setHandler(Handler handler)	設定 Handler，參數對應可參考 3.2
<b>void</b> setOnCallbackResultListener(OnCallbackResult listener)	設定 callback listener，其參數參考 3.1

## 3. Parameter Definition

### 3.1. OnCallbackResult Function Parameter

result :

Value	Description
0	UNKNOWN ERROR
1	SUCCESS
-1	IO EXCEPTION
-2	ILLEGAL ARGUMENT EXCEPTION
-3	CLIENT PROTOCOL EXCEPTION
-4	UNSUPPORTED ENCODING

	EXCEPTION
-5	ILLEGAL STRING LENGTH OR NULL
-6	NOT CALL START TRACKER

**what :**

Value	Description
1029	TRACKER HANDLER

**from :**

Value	Description
0	StartTracker Function
1	Track Function
2	StopTrack Function

**message :**

key	Description
“message”	敘述執行成功或錯誤訊息

### 3.2. Handler 與 Callback listener 參數對應

Handler	OnCallbackResult
msg.what	<b>int</b> what
msg.arg1	<b>int</b> result
msg.arg2	<b>int</b> from
msg.obj	HashMap<String, String> message