
MORE Application Controller SDK

Version 0.16.08.15

2016.08.15

Table of Contents

MORE Application Controller SDK.....	1
<i>Version 0.16.08.15</i>	1
1. Introduction.....	3
1.1. Overview	3
2. Service.....	4
2.1. Application Controller SDK	4
2.2. Use SDK Library	4
2.3. Application Function	8
3. Parameter Definition	10
3.1. OnCallbackResult Function Parameter.....	10
3.2. Handler 與 Callback listener 參數對應.....	11

1. Introduction

1.1. Overview

MORE Application Controller 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 MORE Application Controller SDK 加入到自己的 APP 專案裡，透過 API 的呼叫即可安裝 App 或解除安裝 App，且能主動通知開發者目前智慧型裝置是否有使用者正在安裝或解除安裝 App，目前提供 Android 版本 SDK Library。

2. Service

2.1. Application Controller SDK

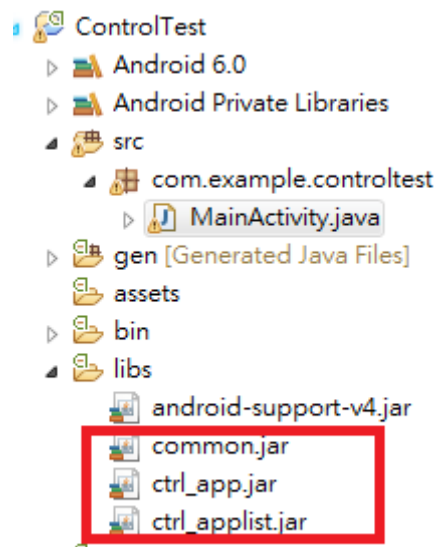
Application Controller SDK 主要提供給開發者一個功能強大的安裝 App 跟解除安裝 App 的 SDK，且能主動通知開發者目前智慧型裝置是否有使用者正在安裝或解除安裝軟體。

2.2. Use SDK Library

在使用 Application Controller SDK 前，需要先透過 AndroidManifest 取得網路與寫入外部記憶體的權限並註冊 Application Receiver，此模組會透過 Handler 或 Callback Listener 主動通知開發者目前 App 安裝或解除安裝狀態。

使用步驟：

1. 先將以下 jar 檔加入到 Android 開發專案的 libs 資料夾



2. 在 AndroidManifest.xml 的 `<manifest>` 裡新增以下幾行

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

3. 在 AndroidManifest.xml 的< application 裡新增以下幾行

```
<receiver android:name="sdk.ideas.ctrl.app.PackageReceiver" >
    <intent-filter>
        <action android:name="android.intent.action.PACKAGE_ADDED" />
        <action android:name="android.intent.action.PACKAGE_REMOVED" />
    </intent-filter>
</receiver>
```

4. 匯入以下 library

```
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.util.Log;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import sdk.ideas.common.CtrlType;
import sdk.ideas.common.OnCallbackResult;
import sdk.ideas.ctrl.app.ApplicationHandler;
import java.util.HashMap;
```

5. 設定Application Handler

```
ApplicationHandler mApplicationHandler = new ApplicationHandler(this);
```

6. 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 CtrlType.MSG_RESPONSE_APPLICATION_HANDLER:
                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

```
mApplicationHandler.setHandler(theHandler);
```

7. Callback Listener 方式

```

mApplicationHandler.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);
    }
});

```

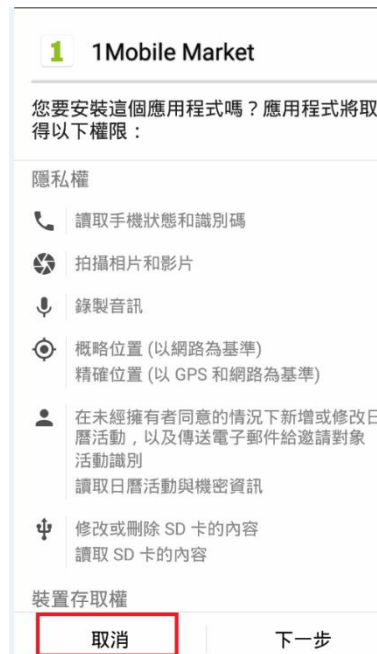
8. 啟動 Application Receiver，此會開始監聽是否有安裝軟體或解除安裝軟體。

```
mApplicationHandler.startListenAction();
```

9. (可選功能)覆寫 Activity onActivityResult function，用此 function 可以讓 APP Controller SDK 知道當使用者按下取消時，能再彈跳出安裝或解除安裝視窗強制要求使用者安裝或解除安裝。

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
    super.onActivityResult(requestCode, resultCode, data);

    mApplicationHandler.onActivityResult(requestCode, resultCode, data);
}
```



10. 安裝軟體

```
mApplicationHandler.installApplicationThread(  
    "http://xx.xxxx.xx.xx.xx/app/android/xxxxxxxxx.apk", null, "  
onemobile.apk", 100);
```

11. 解除安裝軟體

```
mApplicationHandler.unInstallApplication("me.onemobile.andro  
id", 100);
```

2.3. Application Function

Application Controller API	
Function	Description
ApplicationHandler(Context context)	建構元
onActivityResult(int requestCode, int resultCode, Intent data)	當使用者按下取消/解除安裝/安裝按鍵時會觸發，這裡 requestCode 代表以下安

	裝及解除安裝的 appID
unInstallApplication(String packageName, int appID)	解除安裝 App，packageName 為此 App 的 package name
installApplication(String url,String savePath, String apkName, int appID)	<p>1. 此 method 誤用於 Main Thread 執行，不然會 ERROR</p> <p>2. savePath 為外部儲存記憶體，當其值為 null 時預設儲存於 Download/，此路徑不用絕對路徑，不用加入外部記憶體的絕對路徑。</p> <p>3.apkName 為下載此 app 的名稱，例如 xxx.apk</p> <p>4. 當下載完且安裝完時，app SDK 會自動把 apk 給刪除。</p> <p>5. appID 值為唯一的，可讓 SDK 知道使用者有沒有正確安裝或解除安裝開發者要求的軟體。</p>
installApplicationThread(String url, String savePath, String apkName, int appID)	<p>1. 此 function 已使用 Thread 執行</p> <p>2. savePath 為外部儲存記憶體，當其值為 null 時預設儲存於 Download/，此路徑不用絕對路徑，可不用填外部記憶體的絕對路徑。</p> <p>3.apkName 為下載此 app 的名稱，例如 xxx.apk</p> <p>4. 當下載完且安裝完時，app SDK 會自動把 apk 給刪除。</p>

	5. appID 值為唯一的，可讓 SDK 知道使用者沒沒有正確安裝或解除安裝開發者要求的軟體。
void startListenAction()	啟動 Application Receiver，此會開始監聽是否有安裝軟體或解除安裝軟體。 且會判斷是否是開發者自行安裝 App 或使用者自己安裝。請在 OnResume()加入此行。
void stopListenAction()	關閉 Application Receiver，將聽不到任何安裝或解除安裝成功或失敗資訊。請在 onPause() 和 onStop 加入此行。
void setHandler(Handler handler)	設定 Handler，參數對應可參考 3.2
void 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
1	SUCCESS
0	UNKNOWN ERROR
-1	IO EXCEPTION ERROR
-9	MALFORMED URL EXCEPTION ERROR

-10	PROTOCOL EXCEPTION ERROR
-11	PACKAGE NOT FIND ERROR(發生在解除安裝找不到指定 package 時)
-18	NO SPECIFY USE PERMISSION ERROR(未在 AndroidManifest.xml 給權限)
-20	DOWNLOAD ON MAIN THREAD(不可用主執行緒下載)

what :

Value	Description
1033	APPLICATION CONTROLLER HANDLER

from :

Value	Description
0	APPLICATION UNINSTALL BY SYSTEM
1	APPLICATION INSTALL BY SYSTEM
2	APPLICATION INSTALL BY USER
3	APPLICATION UNINSTALL BY USER

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	int what
msg.arg1	int result
msg.arg2	int from
msg.obj	HashMap<String, String> message