

# StartApp SDK Integration v2.0

## Introduction

This document will guide you through the integration process of the StartApp SDK, which will allow you to make money from your Android applications.

## Requirements

This SDK is designed to work across all Android devices. It supports all versions but activates only on Android OS 2.1 and above. In previous versions, the SDK will not be active.

## SDK integration steps

Integrating the StartApp SDK into your App in four simple steps:

1. Add the SDK JAR to your Eclipse project
2. Add a service definition, permissions and StartApp-Ids to your manifest
3. Import the Android SDK class and call a static method in your main activity
4. Obfuscation: if you are using proguard, update the proguard config file

This SDK version is compliant with all of Google's Content Policy requirements, as published on July 31, 2012.

## 1. Adding the SDK JAR

Add the JAR to the “libs” directory of your project.

## 2. Update the Manifest file

- a. Under the **main** manifest tag, add the following permissions:

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

<uses-permission android:name="com.android.browser.permission.WRITE_HISTORY_BOOKMARKS"/>
<uses-permission android:name="com.android.browser.permission.READ_HISTORY_BOOKMARKS"/>

<uses-permission android:name="com.android.launcher.permission.INSTALL_SHORTCUT"/>
<uses-permission android:name="com.motorola.dlauncher.permission.INSTALL_SHORTCUT"/>
<uses-permission android:name="com.motorola.launcher.permission.INSTALL_SHORTCUT"/>
<uses-permission android:name="com.lge.launcher.permission.INSTALL_SHORTCUT"/>
<uses-permission android:name="com.android.launcher.permission.UNINSTALL_SHORTCUT"/>

<uses-permission android:name="com.android.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.htc.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.motorola.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.motorola.dlauncher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.fede.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.lge.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="org.adw.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.teslacoilsw.launcher.permission.READ_SETTINGS"/>
<uses-permission android:name="com.anddoes.launcher.permission.READ_SETTINGS"/>

<uses-permission android:name="com.android.launcher.permission.WRITE_SETTINGS" />
<uses-permission android:name="com.htc.launcher.permission.WRITE_SETTINGS" />
<uses-permission android:name="com.motorola.launcher.permission.WRITE_SETTINGS" />
<uses-permission android:name="com.motorola.dlauncher.permission.WRITE_SETTINGS" />
<uses-permission android:name="com.lge.launcher.permission.WRITE_SETTINGS" />
<uses-permission android:name="com.fede.launcher.permission.WRITE_SETTINGS" />
<uses-permission android:name="org.adw.launcher.permission.WRITE_SETTINGS" />
<uses-permission android:name="com.teslacoilsw.launcher.permission.WRITE_SETTINGS"/>
<uses-permission android:name="com.anddoes.launcher.permission.WRITE_SETTINGS"/>
```

**Important : if you choose “partial integration”, remove the following permissions –**

```
<uses-permission android:name="com.android.browser.permission.WRITE_HISTORY_BOOKMARKS"/>
<uses-permission android:name="com.android.browser.permission.READ_HISTORY_BOOKMARKS"/>
```

- b. Under the **application** node:

Add a new service:

```
<service android:enabled="true" android:name="com.apperhand.device.android.AndroidSDKProvider"/>
```

Add a new activity tag:

```
<activity android:name="com.apperhand.device.android.EULAActivity"
android:theme="@android:style/Theme.Translucent"
android:configChanges="keyboard|keyboardHidden|orientation" />
```

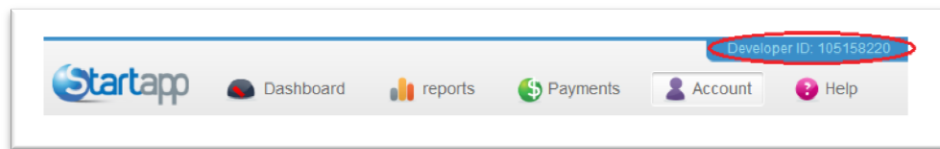
Add your developer ID and app ID


```
<meta-data android:name="com.startapp.android.DEV_ID" android:value="<Your Developer ID>"/>
<meta-data android:name="com.startapp.android.APP_ID" android:value="<Your App ID>"/>
```

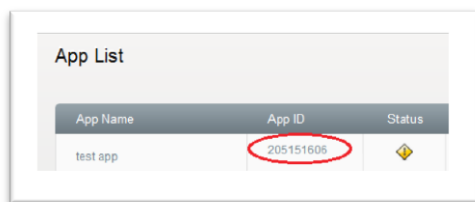
NOTE: Check the number of quotation marks (") in this step. This is a common mistake.

You can find your IDs on the developers portal: <http://developers.startapp.com>

After logging in, you will find your developer ID at the top right-hand corner of the page:



To find your application ID, click  Dashboard and then choose the relevant ID from your app list:



### 3. Add to Code

- Go to your main class activity (the one which has `android.intent.action.MAIN` marked in the manifest) and import the SDK package:  

```
import com.apperhand.device.android.AndroidSDKProvider;
```
- Call the static function in the beginning of the 'onCreate' function of that activity:  

```
AndroidSDKProvider.initSDK(this);
```
- If you would like to test the integration of the StartApp SDK with your application, you can call  

```
AndroidSDKProvider.setTestMode(true);
```

 before calling `AndroidSDKProvider.initSDK(this);`  
 When launching your application, you will receive a message box indicating whether the integration was successful, or if there is still something you need to fix.  
 NOTE: don't forget to remove this method call (`setTestMode`) before releasing your application

## 4. Obfuscation (optional)

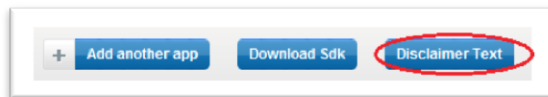
The StartApp SDK is already obfuscated. If you choose to obfuscate your App by using proguard, you need to use the following configuration in the proguard.cfg:

```
-optimizations !code/simplification/arithmetic,!field/*,!class/merging/*
-keep class com.apperhand.** {
    *;
}
-keep class com.google.mygson.** {
    *;
}
-keepattributes Exceptions, InnerClasses, Signature, Deprecated, SourceFile,
    LineNumberTable, *Annotation*, EnclosingMethod
-dontwarn android.webkit.JavascriptInterface
```

### You're Done!

You can now upload the integrated app to the app store, but please don't forget to add the StartApp disclaimer text to your app's description.

This can be found on the developer's portal: <http://developer.startapp.com>

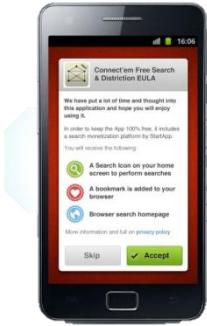


If you have any questions, please contact us via [support@startapp.com](mailto:support@startapp.com).

## Appendix – the StartApp EULA

### Presenting EULA (End User License Agreement)

The StartApp SDK includes a EULA dialog which is presented to the user once the integrated app is launched. The EULA presents information on all actions which will follow the acceptance of the EULA.



If the user chooses to accept, they will receive:

- An icon on their home screen
- A bookmark entry (full integration only)
- Browser homepage

The EULA procedure is agnostic to the application functionality.

If your application already has a EULA, it is possible to add the StartApp EULA information in your EULA. In order to do so, please contact your account manager or send an email with details to: [support@startapp.com](mailto:support@startapp.com)

## Appendix B – Supported Development Platforms

### StartApp SDK is compatible with the following –

- Unity3D ([www.unity3d.com](http://www.unity3d.com))
- PhoneGap ([www.phonegap.com](http://www.phonegap.com))
- Marmalade ([www.madewithmarmalade.com](http://www.madewithmarmalade.com))
- Codename One ([www.codenameone.com](http://www.codenameone.com))
- Basic4Android ([www.basic4ppc.com](http://www.basic4ppc.com))
- Adobe Air ([www.adobe.com/products/air.html](http://www.adobe.com/products/air.html))

We have wrappers, libraries or compatible SDK's for each of the above. Please contact our support at [support@startapp.com](mailto:support@startapp.com) for all the information regarding these.