

DU Ad Platform_SDK for Android接入手册

Version: DuVideoSDK_1.1.0

前提:

DuVideo SDK 支持的最低版本为 API Level 14 - Android 4.0 (ICE_CREAM_SANDWICH)

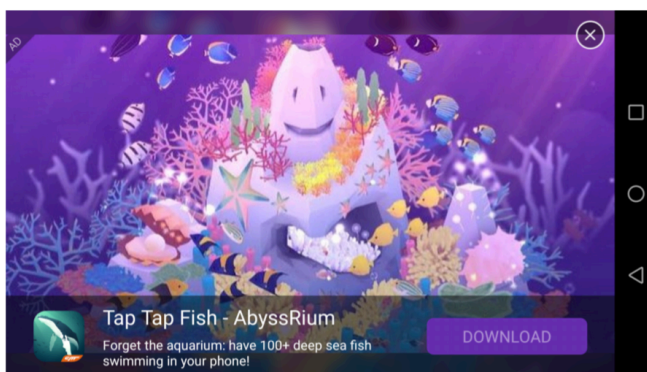
DuVideo SDK 需要依赖 DU Ad Platform_SDK HW1.1.0.0 或 CW1.0.9.9 (含) 以上版本。

在接入 **DuVideo** 之前需要完成 **HW** 或 **CW** 初始化，加载，代码混淆三个部分。

DU Ad Platform_SDK for Android接入手册

1. 获取身份
2. 加载与配置
 - 2.1 导入 SDK 文件
 - 2.2 配置 AndroidManifest.xml
3. 初始化
4. 视频广告使用
 - 4.1 构造视频广告对象
 - 4.2 获取视频实例
 - 4.3 注册视频广告回调
 - 4.4 进行广告拉取
 - 4.5 播放视频广告

视频广告示例（播放页与播放完成页）：



1. 获取身份

请参照 HW 或 CW 版 DUADplatform SDK 文档 第 3 章;

申请 DuVideo 广告位时，需申请的广告位类型为「视频」。

类型 - 奖励广告 为强制用户观看，不可跳过。

类型 - 非奖励广告 不强制用户观看，视频播放 3s 后用户可选择跳过。

2. 加载与配置

请严格按照本章进行配置，否则有可能会运行异常。

请参照 HW 或 CW 版 DUADplatform SDK 文档第 4 章，完成后继续阅读本文档。

2.1 导入 SDK 文件

拷贝 SDK aar 包放到你的安卓工程文件根目录的 libs 目录下，然后配置 build.gradle：

```
repositories {  
    flatDir {  
        dirs 'libs'  
    }  
}  
  
dependencies {
```

```

compile fileTree(include: ['*.jar'], dir: 'libs')
compile(name: 'DuappsAd-xw-xxx-release', ext: 'aar')
compile(name: 'DuVideoSDK-xxx-release', ext: 'aar')

//该依赖用于提取广告主题颜色,具体版本号请根据项目情况更改, 建议配置为21以上。
compile 'com.android.support:palette-v7:23.4.0'

/*如果依赖添加导致编译报错, 请更改为
compile 'com.android.support:palette-v7:23.4.0' {
    transitive = false
}
*/
}
*注: flatDir 指定的位置 即为aar存放的位置

```

2.2 配置 AndroidManifest.xml

添加DuVideo的额外权限（该权限用于获取 WiFi 连接状态，并根据平台设置判断是否下发广告）：

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

3. 初始化

1. 创建 Json 文件，将 Placement_ID 与广告位 ID 建立对应关系。具体格式如下：

```

{
  "video": [
    {
      "pid": "YOUR_DAP_PLACEMENT_ID",
      "fbid": "YOUR_FACEBOOK_PLACEMENT_ID",
      "amid": "YOUR_ADMOB_PLACEMENT_ID"
    }
  ]
}

```

注：如果开发者某广告位不需要 Facebook 广告, 请将该广告位对应的 "fbids" 部分删除。如需绑定 FBID, 请确保 Facebook Audience Network 版本不低于 FB 4.23.0。

如果开发者某广告位不需要 AdMob 广告, 请将该广告位对应的 "amid" 部分删除。

2. 在 application 的 onCreate() 方法中使用 DuVideoSDK.init()

接口说明：

```
public static void init(Context context, String pidsjson)
```

参数	说明
Context context	ACTIVITY CONTEXT
String pidsjson	Placement_ID 与广告位 ID 的对应关系

注：请在完成 `DuAdNetwork.init` 后进行 `DuVideoAdSDK.init`，两个方法可以使用同一 json 文件，SDK 能够自动进行识别

代码示例：

```
public void onCreate() {
    super.onCreate();
    //初始化 DAP SDK
    DuAdNetwork.init(this, getConfigJSON(getApplicationContext()));
    //初始化视频 SDK
    DuVideoAdSDK.init(this, getConfigJSON(getApplicationContext()));
}
```

4. 视频广告使用

4.1 构造视频广告对象

```
public DuVideoAd
```

4.2 获取视频实例

```
public static DuVideoAd DuVideoAdsManager.getVideoAd(Context mContext, int pid)
```

参数	说明
Context context	ACTIVITY CONTEXT
int pid	广告位ID，该 pid 注册在 json 的 video 数组中

代码示例：

```
DuVideoAd duVideoAd = DuVideoAdsManager.getVideoAd(this, YOUR_PID);
```

4.3 注册视频广告回调

```
public void setListener(DuVideoAdListener adListener);
```

参数	说明
DuVideoAdListener adListener	回调函数返回获取广告错误，获取广告的数据，广告点击事件。

```
public interface DuVideoAdListener {  
    //视频广告加载完成时回调，可以调用playAd()方法播放广告  
    void onAdPlayable();  
  
    //广告异常回调，错误码同 HW&CW 版本  
    void onAdError(AdError error);  
  
    //广告开始播放时回调  
    void onAdStart();  
  
    //广告播放完成后回调，已废弃，功能由以下3个回调代替  
    void onAdEnd(AdResult result);  
  
    //广告被点击时回调  
    void onAdClick();  
  
    //广告播放完成时回调  
    void onVideoCompleted();  
  
    //广告被关闭时回调  
    void onAdClose();  
}
```

4.4 进行广告拉取

```
public void load()
```

此接口只需调用一次，视频广告会在后台线程持续拉取，拉取到广告后会通过回调通知。请在执行 `load` 操作前先进行设置好数据监听接口。

注：视频文件拉取需要时间较长，建议在广告对象创建后立即进行数据监听和 `load` 操作

4.5 播放视频广告

```
public boolean isAdPlayable()
```

返回当前是否有可以播放的广告，有返回true，没有则返回false

```
public void playAd(Context context)
```

播放广告

注：视频广告将根据设备的屏幕方向自动旋转。

Java代码示例：

```
if (duVideoAd.isAdPlayable()) {  
    duVideoAd.playAd(mContext);  
}
```

Kotlin代码示例：

```
if (duRewardedVideoAd.isAdPlayable) {  
    duRewardedVideoAd.playAd(this@MainKotlinActivity)  
    skipping_video.isEnabled = false  
}
```