

趣拍SDK阿里云标准版 for Android 开发文档

一、环境搭建

注：新版短视频SDK需要AndroidStudio环境。

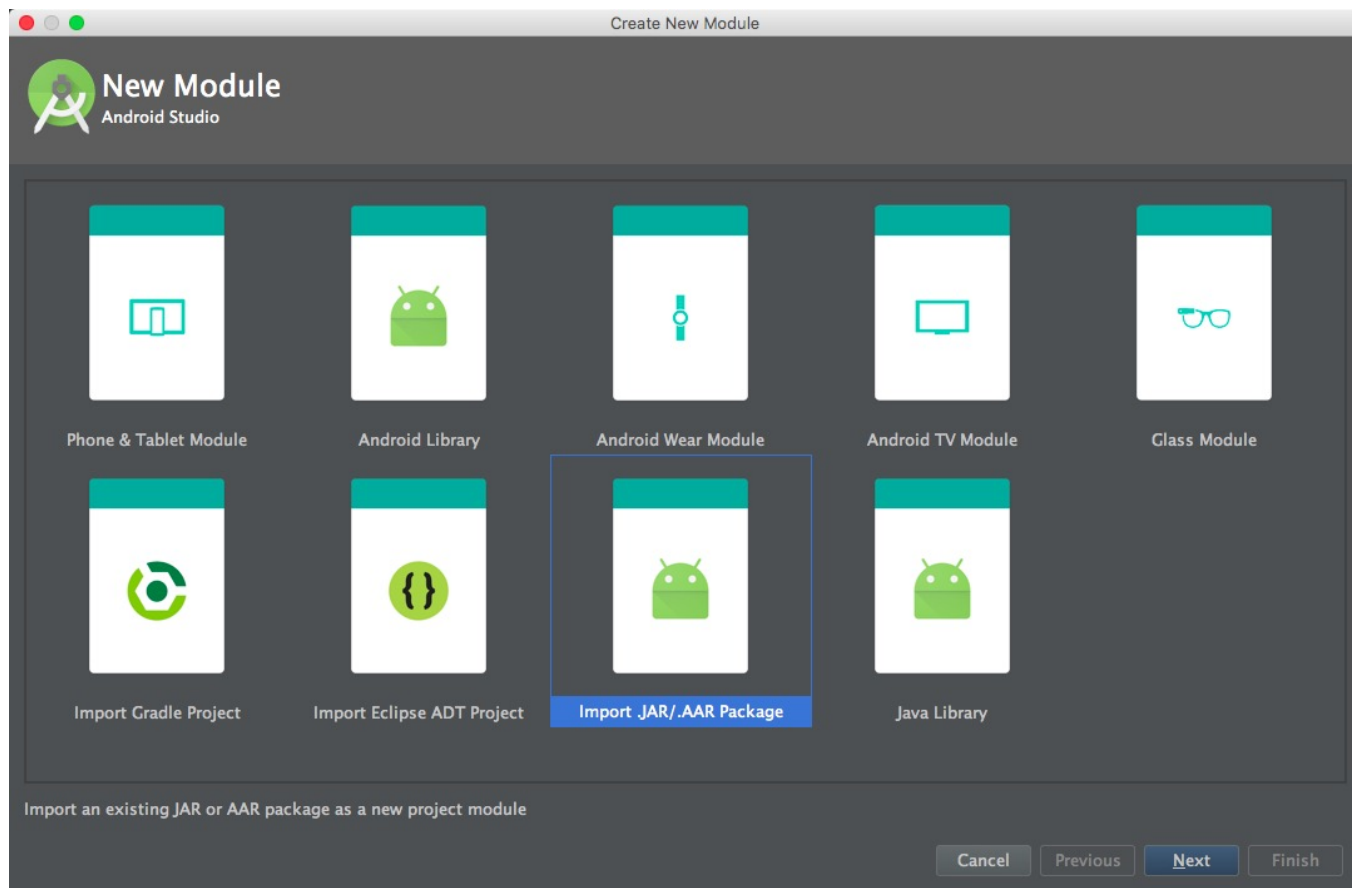
1.1 下载SDK，解压后得到01-lib（为SDK库文件），02-demo（为SDK demo文件），03-doc（为SDK 集成文档）。

注意：

- demo包含短视频的demo和直播的demo，提供的AAR包是短视频和直播的合并SDK，纯粹的短视频需求用户只需要Qupaisdk-release.aar导入即可。
- demo依赖的第三方库为必须依赖的库不可删减

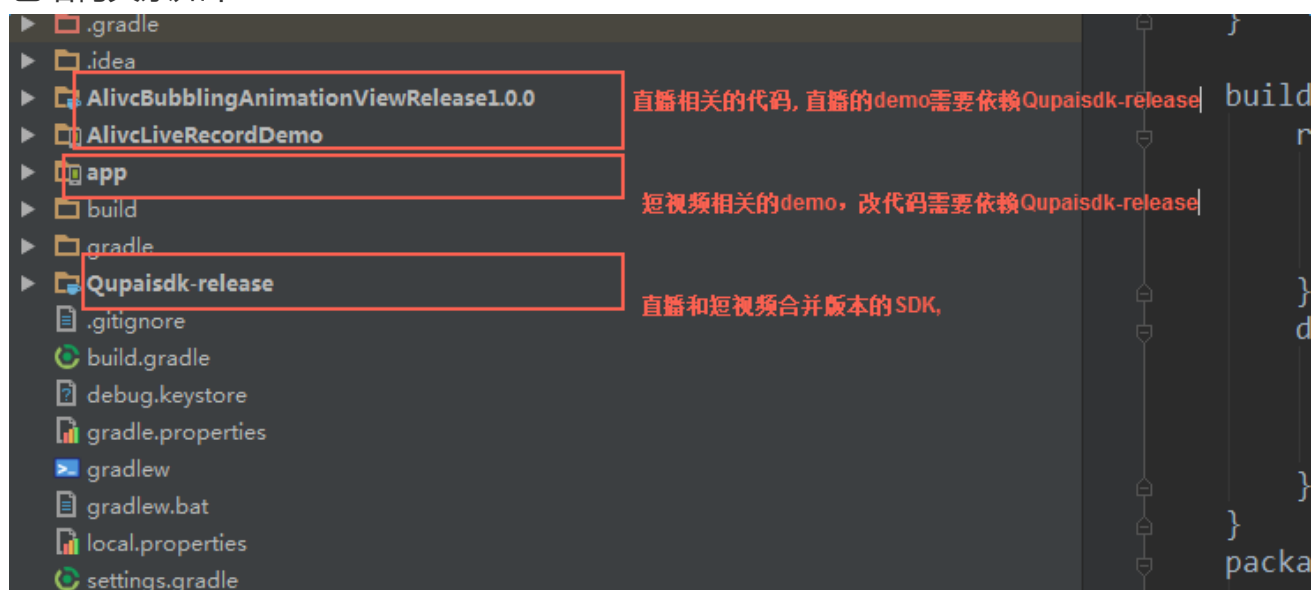
1.2 将01-libQupaisdk-release目录下的aar文件导入到开发者项目中。

打开File，选择New，点击New Module，再选择导入AAR包。



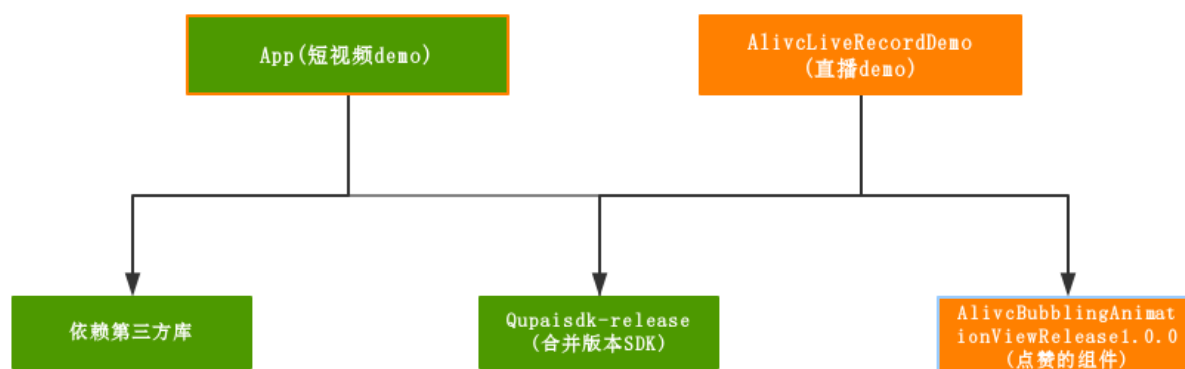
1.3 SDK提供的demo的目录结构：

- 包结构关系如下：



- 依赖关系如下：

阿里云短视频+直播demo依赖关系



1.3 获取license

1) 进入控制台

2) 点击应用管理

3) 在已经创建的应用中点击获取license或联系商务 (已使用云服务的用户可以直接获取标准版license、其他版本license需通过商务授权获得, 有需要请联系商务)

如果成功获取了license, 在运行app时还显示license无效, 请先清除下工程再运行appQupaisdk-release.aar在Qupaisdk-release文件夹中。

1.4 开发者Android项目添加对QupaiSDK的依赖。

```
1. dependencies {
2.     compile 'com.android.support:appcompat-v7:22.2.0'
3.     compile 'com.android.support:support-v13:22.2.0'
4.     compile 'com.fasterxml.jackson.core:jackson-databind:2.6.4'
5.     compile 'com.google.dagger:dagger-compiler:2.0.1'
6.     compile 'com.google.dagger:dagger:2.0.1'
7.     compile 'com.android.support:recyclerview-v7:22.2.0'
8.     compile 'com.nostra13.universalimageloader:universal-image-
loader:1.9.4'
9.     compile 'com.google.code.findbugs:jsr305:3.0.0'
10.    compile project(':Qupaisdk-release')
11.    compile project(':oss-android-sdk')
12. }
```

设置AndroidManifest.xml文件，声明使用权限

```
1. <uses-permission
2.     android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
3.
4. <uses-permission
5.     android:name="android.permission.ACCESS_NETWORK_STATE" />
6.
7. <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" /
8. >
9.
10. <uses-permission android:name="android.permission.READ_PHONE_STATE" />
11.
12. <uses-permission android:name="android.permission.READ_SETTINGS" />
13.
14. <uses-permission android:name="android.permission.WRITE_SETTINGS" />
15.
16. <uses-permission android:name="android.permission.INTERNET" />
17.
18. <uses-permission
19.     android:name="android.permission.READ_EXTERNAL_STORAGE" />
20.
21. <uses-permission android:name="android.permission.GET_TASKS" />
```

在Module:app的build.gradle中添加：

```

1.  android{
2.      //其它配置
3.      packagingOptions {
4.          exclude 'META-INF/LICENSE'
5.          exclude 'META-INF/NOTICE'
6.      }
7.  }

```

1.5 需要引入的库 在工程中引入01-Demo中libs目录下有一系列的so包：

```
armeabi-v7a/libgnustl_shared.so
```

```
armeabi-v7a/libqupai-media-jni.so
```

```
armeabi-v7a/libqupai-media-thirdparty.so
```

注意：

a:保证so在默认的目录里面.默认gradle读取目录为：src/main/jniLibs/abi/XXX.so

b:目前仅提供armeabi-v7a的so。开发者自行做适配，官方文

档：<https://developer.android.com/ndk/guides/abis.html?hl=is> 参考博

文：<http://www.cnblogs.com/shansheng/p/5200998.html>

二、SDK使用方法说明

2.0 加载底层动态库(必须)

```

1.  for (String str : new String[]{"gnustl_shared", "qupai-media-thirdparty",
2.      "qupai-media-jni"}) {
3.      System.loadLibrary(str);
4.  }

```

2.1 创建实例

```

1.  QupaiService qupaiService = QupaiManager.getQupaiService(v.getContext());

```

2.2 初始化参数-initRecord

SDK初始化成功之后，将参数进行初始化设置。– 建议在初始化成功的时候调用，建议只调用一次。测试demo中为可配置，所以可多次调用。

示例：

```
1. //UI设置参数 UISettings _UISettings = new UISettings() {
2.     @Override
3.     public boolean hasEditor() {
4.         return true; //是否需要编辑功能
5.     }
6.
7.     @Override
8.     public boolean hasImporter() {
9.         return true; //是否需要导入功能
10.    }
11.
12.    @Override
13.    public boolean hasGuide() {
14.        return false; //是否启动引导功能，建议用户第一次使用时设置为true
15.    }
16.
17.    @Override
18.    public boolean hasSkinBeautifier() {
19.        return true; //是否显示美颜图标
20.    }
21. };
22. //压缩参数
23. MovieExportOptions movie_options = new MovieExportOptions.Builder()
24.     .setVideoBitrate(mVideoBitrate)
25.     .configureMuxer("movflags", "+faststart")
26.     .build();
27. //输出视频的参数
28. ProjectOptions projectOptions = new ProjectOptions.Builder() //输出视频
    宽高目前只能设置1:1的宽高，建议设置480*480.
29.     .setVideoSize(480, 480) //帧率 .setVideoFrameRate(30)
30. //时长区间
31.     .setDurationRange(2, 8) .get();
32. //缩略图参数,可设置取得缩略图的数量，默认10张
33. ThumbnailExportOptions thumbnailExportOptions =new
    ThumbnailExportOptions.Builder().setCount(10).get();
34. VideoSessionCreateInfo info =new VideoSessionCreateInfo.Builder()
35. //水印地址，如"assets://Qupai/watermark/qupai-logo.png"
36.     .setWaterMarkPath(waterMarkPath)
37. //水印的位置
38.     .setWaterMarkPosition(1)
```

```

39. //摄像头方向,可配置前置或后置摄像头
40. .setCameraFacing(Camera.CameraInfo.CAMERA_FACING_BACK)
41. //美颜百分比,设置之后内部会记住,多次设置无效
42. .setBeautyProgress(80)
43. //默认是否开启
44. .setBeautySkinOn(true)
45. .setMovieExportOptions(movie_options)
46. .setThumbnailExportOptions(thumbnailExportOptions)
47. .build();
48. //初始化,建议在application里面做初始化,这里做是为了方便开发者认识参数的意义
49. qupaiService.initRecord(info,projectOptions,_UISettings);

```

2.3 拍摄API

```

1. /** * 调用拍摄 Activity *
2. @param activity当前的activity *
3. @param requestCode 为避免重复定义requestCode,请开发者自行定义requestCode *
4. @param isFirstRecord 是否是第一次调用拍摄 */
5. void showRecordPage(Activity activity,int requestCode,boolean
   isFirstRecord);

```

示例：

```

1. qupaiService.showRecordPage(MainActivity.this,
   RequestCode.RECORDE_SHOW, isGuideShow);

```

2.4 添加音乐API

```

1. /** * 添加音乐 *
2. @param id 音乐ID , id为int型的整数 *
3. @param name 音乐名称 *
4. @param musicPathName 音乐保存的路径的名称-传递文件路径,如:assets://Qupai/music/Athena 或者 file:///XXX */
5. void addMusic(int id,String name,String musicPathName);

```

示例：

```

1. if(qupaiService != null){
2.     qupaiService.addMusic(0, "Athena", "assets://Qupai/music/Athena");

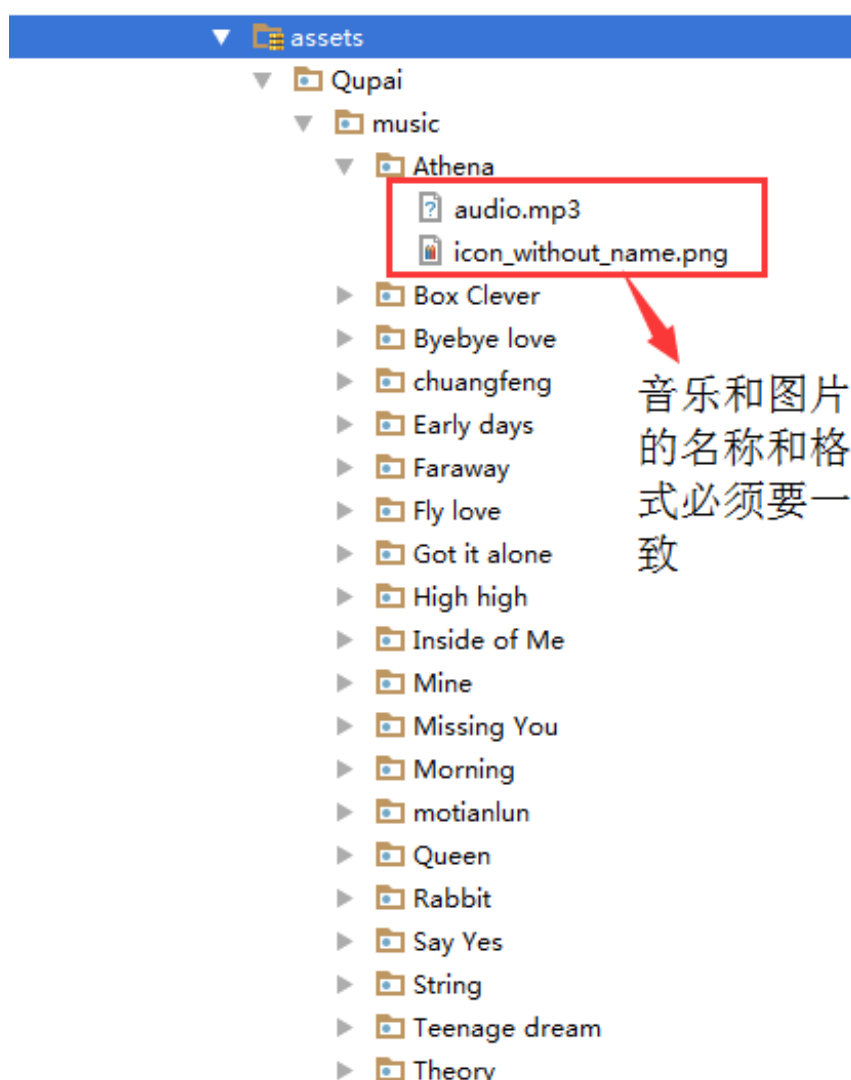
```

```

3.     qupaiService.addMusic(1, "Box Clever", "assets://Qupai/music/Box Clever");
4.     qupaiService.addMusic(2, "Byebye love",
5.     "assets://Qupai/music/Byebye love");
6.     qupaiService.addMusic(3, "chuangfeng",
7.     "assets://Qupai/music/chuangfeng");
8.     qupaiService.addMusic(4, "Early days", "assets://Qupai/music/Early days");
9.     qupaiService.addMusic(5, "Faraway", "assets://Qupai/music/Faraway");
10.    }

```

背景音乐目前必须放在 `assets://Qupai/music` 下，一个目录代表一首背景音乐，目录下 `audio.mp3` 为音乐文件。



2.5 更多音乐下载页面实现

更多音乐页面的实现由开发者自己实现。开发者只需要传递一个intent给接口。就可以了，如果不需要更多音乐入口则传递空。保证灵活性，具体的音乐下载逻辑由开发者自己实现，下载之后调用addMusic 就可以了。 AddMusic 可以传递本地路径也可以传递 assets 目录下的文件。示例代码：

```
1. //是否需要更多音乐页面-如果不需要更多音乐可以干掉
2. Intent moreMusic = new Intent();
3. if (st_more_music.isChecked()) {
4.     moreMusic.setClass(MainActivity.this, MoreMusicActivity.class);
5. } else {
6.     moreMusic = null;
7. }
8. qupaiService.hasMoreMusic(moreMusic); //如果不设置就没有更多音乐。
9. }
```

2.6录制完成的处理方法

在Demo中录制完成后调用了清除草稿的功能，需要存文件的请开发者在删除之前执行move操作。

```
1. @Override protected void onActivityResult(int requestCode, int
   resultCode, Intent data) {
2.     if (resultCode == RESULT_OK) {
3.         RecordResult result = new RecordResult(data); //得到视频地址，和缩略图
           地址的数组，返回十张缩略图
4.         videoFile = result.getPath(); thum = result.getThumbnail(); result
           .getDuration();
5.         tv_result.setText("视频路径:" + videoFile + "图片路径:" + thum[0]);
6.
7.     } else {
8.         if (resultCode == RESULT_CANCELED) {
9.             Toast.makeText(MainActivity.this, "RESULT_CANCELED", Toast.
               LENGTH_LONG).show();
10.        }
11.    }
12. }
```

三、上传视频功能的使用

请使用阿里云OSS上传SDK，详见[OSS上传文档](#)

四、混淆配置

```
1. -ignorewarnings -keepattributes
   Annotation,EnclosingMethod,Signature,InnerClasses
2. -keep class com.duanqu.**
3. -keepclassmembers class com.duanqu.** { *; }
4. -keepnames class com.fasterxml.jackson.** { ; }
5. -dontwarn com.fasterxml.jackson.databind.* ````
```