

腾讯云 企业版(Enterprise) SDK 基于 LiteAVSDK 统一框架设计和实现，该框架包TRTC、AI美颜在内的多项功能：

环境要求

- Flutter 1.12 及以上版本。
- Android 端开发：
 - Android Studio 3.5及以上版本。
 - App 要求 Android 4.1及以上版本设备。
- iOS 端开发：
 - Xcode 11.0及以上版本。
 - 请确保您的项目已设置有效的开发者签名。

前提条件

- 您已 [注册腾讯云](#) 账号，并完成 [实名认证](#)。
- 进入 [美颜特效服务开通申请页](#)，填写申请信息并单击【提交申请】。生成的 License 信息中的 Key 和 LicenseUrl 将在 [步骤4](#) 集成的 SDK 中用到。特别需要注意的是 **iOS bundle ID** 和 **Android 应用包名称 (package name)** 一经绑定License就不能修复。

！

- 您可在 iOS 工程配置中的 **General->Identity** 中获取 bundle ID，如下所示：

Identity

Display Name	<input data-bbox="564 1216 1059 1256" type="text" value="\${PRODUCT_NAME}"/>
Bundle Identifier	<input data-bbox="564 1267 1059 1308" type="text" value="de.telekom.entertain-tv-mobil"/>
Version	<input data-bbox="564 1319 1059 1359" type="text" value="1.0"/>
Build	<input data-bbox="564 1370 1059 1411" type="text" value="1.0"/>

- 您可在 Android 工程下的 **Mainfest.xml** 文件中获取应用包名称 (**package name**)，如下所示：

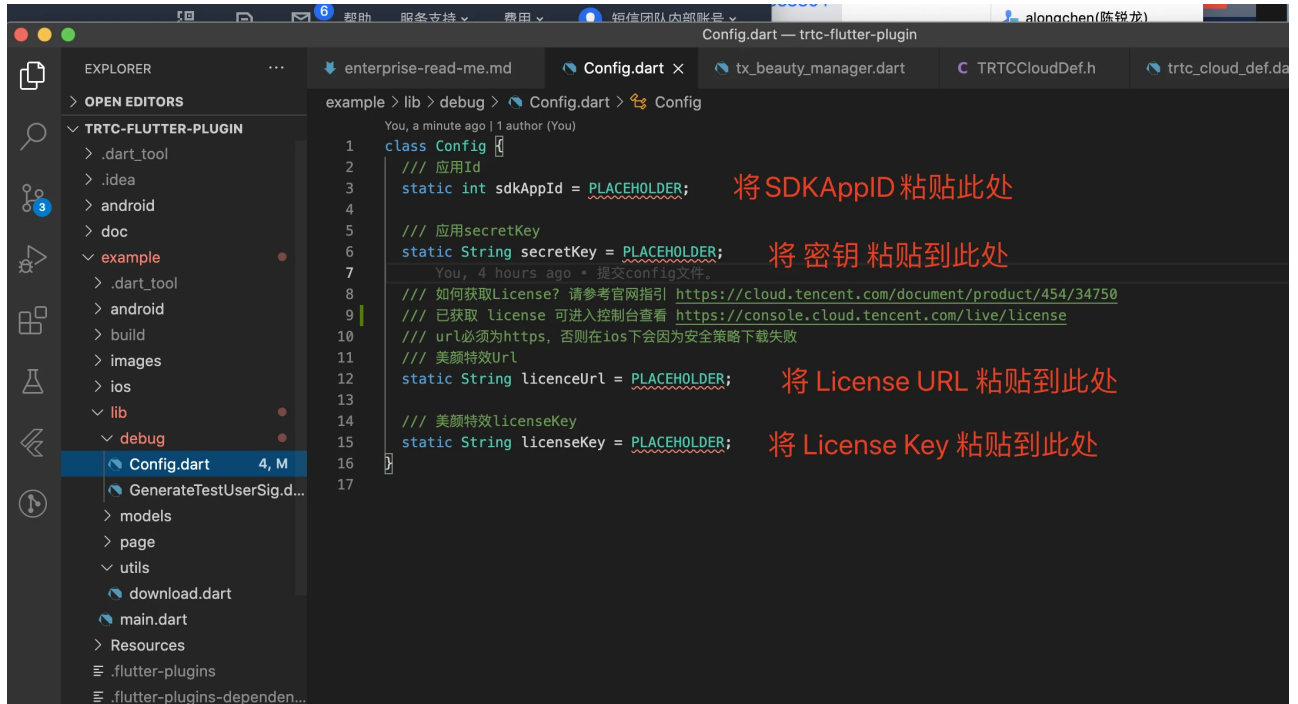
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.huawei.player"
    android:versionCode="20181111"
    android:versionName="1.0">
```

操作步骤

步骤1：配置 Demo 工程文件

- 企业版下载后需要美颜特效 SDK 的解压密码和授权 License 才能运行，解码密码和授权 License 请根据 [美颜特效 SDK 购买流程](#) 获取。

2. 解压下载的源码包。
 3. 找到并打开 `example/lib/debug/config.dart` 文件。
 4. 设置 `config.dart` 文件中的相关参数：
- SDKAPPID: 默认为 PLACEHOLDER, 请设置为实际的 SDKAppID。
 - SECRETKEY: 默认为 PLACEHOLDER, 请设置为实际的密钥信息。
 - licenceUrl: 默认为 PLACEHOLDER, 请设置为实际的美颜特效Licence Url
 - licenseKey: 默认为 PLACEHOLDER, 请设置为实际的美颜特效licenseKey。



5. 返回实时音视频控制台，单击【粘贴完成，下一步】。
6. 单击【关闭指引，进入控制台管理应用】。

!

- 本文提到的生成 UserSig 的方案是在客户端代码中配置 SECRETKEY, 该方法中 SECRETKEY 很容易被反编译逆向破解, 一旦您的密钥泄露, 攻击者就可以盗用您的腾讯云流量, 因此该方法仅适合本地跑通 **Demo** 和功能调试。
- 正确的 UserSig 签发方式是将 UserSig 的计算代码集成到您的服务端, 并提供面向 App 的接口, 在需要 UserSig 时由您的 App 向业务服务器发起请求获取动态 UserSig。更多详情请参见 [服务端生成 UserSig](#)。
- 申请 License。进入 [美颜特效服务开通申请页](#), 填写申请信息并单击【提交申请】。生成的 License 信息中的 Key 和 LicenceUrl

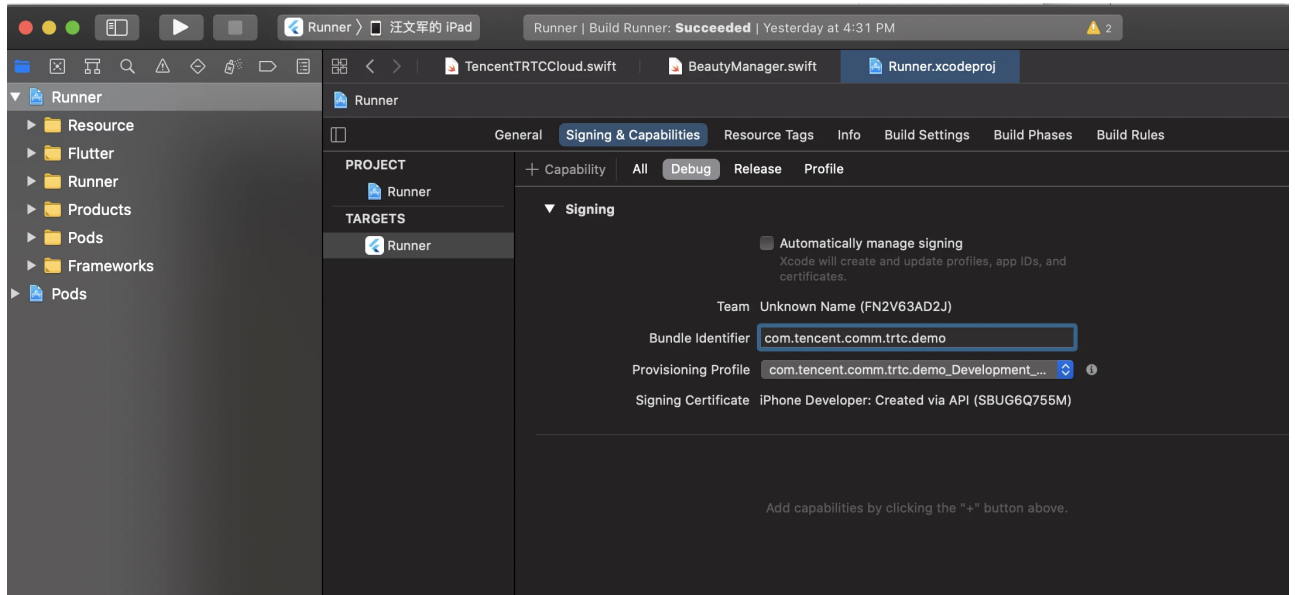
步骤2: 编译运行

Android端

1. 进入根目录, 安装依赖 `flutter pub get`
2. 使用 Android Studio (3.5及以上的版本) 打开源码工程。
3. 单击【运行】即可。

iOS端

1. 进入根目录，安装依赖 `flutter pub get`
2. 使用 XCode（11.0及以上的版本）打开源码目录下的 `/example/ios/Runner.xcworkspace`。
3. 您可在 iOS 工程配置中的 General->Identity 中获取 bundle ID，如下所示：



4. 编译并运行 Demo 工程即可。

demo工程介绍

目录结构

```
├── android    // trtc flutter plugin android
│   ├── gradle    //Android gradle
│   ├── libs      //trtc android plugin 目录, 存放trtc的原生android sdk
│   └── src        //trtc flutter plugin android 代码目录
├── doc        //trtc flutter sdk文档目录
│   └── api        //trtc flutter plugin api目录
├── example    //企业版 demo 目录
│   ├── Resources //企业版 demo 的资源文件
│   ├── android   //企业版 demo android 目录
│   ├── build     //
│   ├── images    //企业版 demo 的图片资源
│   ├── ios       //企业版 demo 的 ios工程目录
│   └── lib        //企业版 demo 的实现代码
├── ios        //trtc flutter plugin ios目录
│   ├── Assets    //
│   └── Classes   //trtc flutter plugin ios 代码目录
└── lib        //trtc flutter plugin dart 接口代码
```

主要文件

- lib/trtc_cloud-腾讯云视频通话功能的主要接口类
- lib/trtc_cloud_video_view-提供渲染视频TRTCCloudVideoView的widget
- lib/tx_audio_effect_manager-腾讯云视频通话功能音乐和人声设置接口

- lib/tx_beauty_manager-美颜及动效参数管理
- lib/tx_device_manager-设备管理类
- lib/trtc_cloud_def-腾讯云视频通话功能的关键类型定义
- lib/trtc_cloud_listener-腾讯云视频通话功能的事件回调监听接口

功能调用

初始化SDK

```
// 创建 TRTCCloud 单例
trtcCloud = await TRTCCloud.sharedInstance();
// 获取设备管理模块
txDeviceManager = trtcCloud.getDeviceManager();
// 获取美颜管理对象
txBeautyManager = trtcCloud.getBeautyManager();
// 获取音效管理类
txAudioManager = trtcCloud.getAudioEffectManager();
```

美颜特效License

在工程配置完之后，您需集成美颜特效 License 才能开始使用功能。在调用 SDK 的相关接口前调用如下所示方法进行 License 的设置。

```
// 配置 美颜License, licenceUrl必须使用https, 否则ios下回下载
// licenceUrl 下载url 和 key。授权 License 请根据 美颜特效 SDK [购买流程]
// (https://cloud.tencent.com/document/product/647/32689#Enterprise) 获取。
txBeautyManager.setLicence(licenceUrl, licenseKey);
```

美颜动效（动效贴纸、AI 抠图、美妆、手势）

购买美颜动效素材后，您可以获得对应效果的素材包。每一个素材包就是一个独立的目录，目录里包含了很多资源文件。每个素材包因其复杂度不同，文件数量和大小尺寸也各不相同。

为了节省安装包体积，我们建议您将素材包上传到您的服务器上，并将下载地址配置在您的 App 中，例如：http://yourcompany.com/hudongzhibo/AISpecial/**/{动效名}.zip。在 App 启动后，下载并解压素材包到 Documents 目录下。完成解压后，即可通过以下接口开启动效效果：

```
https://st1.xiangji.qq.com/yunmaterials/{动效名}.zip
https://st1.xiangji.qq.com/yunmaterials/{动效名}.png
```

强烈建议您将动效资源放在自己的服务器上，以防小直播变动造成不必要的影响。当解压完成后，即可通过以下接口开启动效效果。

```
//动效名。
String motionKey = "video_xiaofu";
//判断动效文件夹是否存在。
bool isExitDir = await TXUtils.isMotionDirExit(motionKey);
//如果动效文件夹存在直接返回动效文件夹路径，否则下载该动效文件并且解压返回动效文件夹目录。
String fullPath = isExitDir
    ? await TXUtils.getMotionDirByKey(motionKey)
    : await TXUtils.downloadMotionZipFile(Platform.isAndroid
        ?
        "https://dldir1.qq.com/hudongzhibo/AISpecial/Android/170/video_xiaofuAndroid.zip"
        :
        "https://dldir1.qq.com/hudongzhibo/AISpecial/ios/160/video_xiaofu.zip");
//选择使用哪一款 AI 动效挂件，该接口仅在【企业版 SDK】
(https://cloud.tencent.com/document/product/647/32689#Enterprise) 中生效
//有些挂件本身会有声音特效，通过此 API 可以关闭这些特效播放时所带的声音效果。
// 参数：fullPath 特效文件夹的全路径。
txBeautyManager.setMotionTpl(fullPath);
```

绿幕功能

使用绿幕需要先准备一个用于播放的 mp4 文件，通过调用以下接口即可开启绿幕效果：

```
// 设置绿幕背景视频,此处的绿幕功能并非智能抠背，需要被拍摄者的背后有一块绿色的幕布来辅助产生特效
// file 视频文件路径。支持 MP4；nil 表示关闭特效。
// 注意：支持asset资源路径和绝对路径，带"/"开头会被识别为绝对路径
txBeautyManager.setGreenScreenFile("Resources/goodluck.mp4");
```

大眼瘦脸

大眼和瘦脸通过以下方法设置：

```
//设置大眼级别，该接口仅在【企业版 SDK】
(https://cloud.tencent.com/document/product/647/32689#Enterprise) 中生效
//eyeScaleLevel 大眼级别，取值范围0 - 9； 0表示关闭，1 - 9值越大，效果越明显。
txBeautyManager.setEyeScaleLevel(eyeScaleLevel)

//设置瘦脸级别，该接口仅在【企业版 SDK】
(https://cloud.tencent.com/document/product/647/32689#Enterprise) 中生效
//faceSlimLevel 瘦脸级别，取值范围0 - 9； 0表示关闭，1 - 9值越大，效果越明显。
txBeautyManager.setFaceSlimLevel(double faceSlimLevel);
```

相关文档链接

- [实时音视频（TRTC）API文档](#)
- [Demo体验](#)
- [美颜特效 SDK 文档](#)

常见问题

- [两台手机同时运行 Demo，为什么看不到彼此的画面？](#)
- [防火墙有什么限制？](#)
- [iOS 打包运行 Crash？](#)
- [iOS 无法显示视频（Android 正常）？](#)
- [更新 SDK 版本后，iOS CocoaPods 运行报错？](#)
- [Android Manifest merge failed 编译失败？](#)
- [因为没有签名，真机调试报错？](#)
- [对插件内的 swift 文件做了增删后，build 时查找不到对应文件？](#)
- [Run 报错“Info.plist, error: No value at that key path or invalid key path: NSBonjourServices”？](#)
- [Pod install 报错？](#)
- [Run 的时候 iOS 版本依赖报错？](#)