

云联络中心

音视频客服





【版权声明】

©2013-2025 腾讯云版权所有

本文档(含所有文字、数据、图片等内容)完整的著作权归腾讯云计算(北京)有限责任公司单独所有,未经腾讯云 事先明确书面许可,任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成 对腾讯云著作权的侵犯,腾讯云将依法采取措施追究法律责任。

【商标声明】

🕗 腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算(北京)有限责任公司及其关联公司所有。本文档涉及的第三方主体的 商标,依法由权利人所有。未经腾讯云及有关权利人书面许可,任何主体不得以任何方式对前述商标进行使用、复 制、修改、传播、抄录等行为,否则将构成对腾讯云及有关权利人商标权的侵犯,腾讯云将依法采取措施追究法律责 任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况,部分产品、服务的内容可能不时有所调整。 您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定,除非双方另有约定,否则, 腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务,及相应的技术售后服务,任何问题请联系 4009100100或 95716 。



文档目录

音视频客服

集成音视频客服

效果展示

集成用户端音频客服

集成用户端视频客服

跳转小程序方式接入

API 文档

用户端 SDK 下载和发布日志(App)

视频会话 CDR 推送

音视频客服 集成音视频客服 效果展示

最近更新时间: 2023-05-15 16:38:30

音频通话效果





视频通话效果





集成用户端音频客服

最近更新时间: 2025-06-30 19:46:12

本文主要介绍如何快速集成腾讯云联络中心音频客服,只要按照如下步骤进行配置,就可以跑通用户端的音频客服。

前提条件

- 您已 注册腾讯云 账号,并完成 实名认证。
- 您已开通云联络中心服务,并创建了云联络中心实例。

集成用户端音频客服前置准备

1. 登录 云联络中心管理端,选择登录**云联络中心 > 管理端 > 音频客服 > 技能组管理 > 添加技能组**。

● 电话外呼 □	工作台 🗧 管理端 (1	L) 服务记录 🔛 新手配算	置指引		
器 概览	技能组管理				
🔜 数据分析					
□ 服务记录	+ 添加技能组	3 请知	前入技能组名称或客服姓名、曲 Q	+ 添加新客服	
↔ 实时监控	技能组名称	接待上限	操作	客服姓名	邮箱
○ 在线客服	test	1	编辑删除	a	@tencent.com
◎ 电话客服 ~	共 1 条	10 🔻 条 / 页	₩ ◀ 1 /1页 ▶ ₩	共 1 条	
◎ 音频客服 ^					
 技能组管理 2 					
。 IVR管理					
• 自助服务					
• 放音文件管理					
• 渠道管理					
• 音频客服设置					



2. 添加 新客服 到对应的技能组中。

◎ 电话外呼		工作台 管理端	L 服务记录 ▲ 新手配	置指引				
器 概览		技能组管理						
🖮 数据分析								
□ 服务记录		+ 添加技能组	谱	输入技能组名称或客服姓名、曲	Q	+ 添加新客服	2、添加新客服	
🐼 实时监控		技能组名称	接待上限	操作		客服姓名	邮箱	
○ 在线客服	Ŷ	test	ı 1 ,	选择技能組		you	.@tencent.com	
ⓒ 电话客服	~	共 1 条	10 🔻 条 / 页	₩ ◀ 1 /1页 ▶	H	共 1 条		
♥ 音频客服	^							
• 技能组管理								
。 IVR管理								
• 自助服务								
。 放音文件管理								
。 渠道管理								
• 音频客服设置								

3. 新增 IVR, 选择登录的云联络中心 > 管理端 > 音频客服 > IVR管理 > 呼入 IVR > 新建。

📞 电话外呼	I 🗆	工作台	■ 管理端 1	L 服务记录	▲ 新手配置指引
器 概览		IVR管理			
山 数据分析		呼入IVR	自助服	务IVR	
1 服务记录		新建	3		
⑦ 实时监控		名称			更新时间
○ 在线客服	~				
ⓒ 电话客服	~				
ⓒ 音频客服	^				
• 技能组管理					
・ IVR管理 2					
• 自助服务					
• 放音文件管理					
• 渠道管理					

4. 新建 IVR,在 IVR 画布左上角输入 IVR 名称,如: 音频客服 IVR。根据您的场景需要拖拽 IVR 模块到画布区域 合适的位置释放,连接模块并在各个模块填写相应信息。IVR 流程必须以开始模块为第一个模块,以结束模块为



最后一个模块。

《 电话外呼] I	作台	●● 管理端	服务记录
🂵 数据分析		•	÷[音频通话IVR MAIN	
🗐 服务记录		拖z	动以新	增节点	
💮 实时监控		基码	诎模块		T 144
○ 在线客服	~	8	F\$	条件判断	71%
☺ 电话客服	~		ß	服务时间	
♥ 音频客服	^		=	接口调用	
• 技能组管理				放音	1: 产品咨询
。 IVR管理		多	功能模	块	2: 产品售后
。 自助服务		::	-C	分支	
。 放音文件管理			ť	ASR导航	↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

5. 新增配置音频入口。选择管理端 > 渠道管理 > 配置音频入口 > 新增,输入音频入口名称,选择 IVR。

《 电话外呼		│ 工作台 🕂 管理端 🕒 服务记录 🎦 新手配置指引	
器 概览		渠道管理	
<u>ы</u> 数据分析		配置音频入口 绑定小程序	
🗐 服务记录		新增 3	
🔗 实时监控			
○ 在线客服	~	音频入口名称	音频,
论 电话客服	~		
🛇 音频客服	~	共 0 条	
• 技能组管理			
。 IVR管理			
• 自助服务			
• 放音文件管理			
・ 渠道管理 2			
• 音频客服设置			
⊠ 视频客服	~		



6. 复制音频入口 ID。

● 电话外呼] 工作台 🗧 管理	里端 🕒 服务记录	新手配置指引	I			
田 概览		渠道管理						
: 🛄 数据分析		配置音频入口	绑定小程序					
圓 服务记录		新増						
④ 实时监控								
○ 在线客服	~	音频入口名称			音频入口ID		关联的IVR	操作
♥ 电话客服	~	音频通话测试			ioYTSfp6xWKAEQuG	复制音频入口ID	音频通话IVR	编辑册
ⓒ 音频客服	^	共 1 条						10 💌
• 技能组管理								
。 IVR管理								
· 自助服务								
• 放音文件管理								
• 渠道管理								

腾讯云联络中心提供了多种方式集成音频客服,包括跳转到 TCCC 联络小程序平台 完成音频客服功能,同时也提供 了小程序 SDK、Android SDK 方便开发者实现音频客服相关功能。

跳转到 TCCC 联络小程序平台

我们提供了 TCCC 联络小程序平台,可以方便您在微信小程序里、企业微信里、自有 App 内打开 **TCCC 联络小程** 序 完成音频客服。

跳转参数说明

拉起 TCCC 联络小程序平台需要用到以下参数。 请先记录这些参数,后文将会详细介绍具体开发方式。

参数	值	备注
appl d	wxf3ac5a9dfc918430	固定值
path	/pages/call/call??sdkAppId= {YourAppID}&channeIId={YourChanneIID}	请将具体参数值修改为您自 己的

path 参数说明

参数	类型	必填	备注
sdkAppId	String	是	腾讯云联络中心应用 ID
channelld	String	是	视频入口 ID,参考 复制音频入口 ID 。

微信小程序内进行跳转



使用 wx.navigateToMiniProgram 拉起 **TCCC 联络小程序**即可,请参见 <mark>打开另一个小程序</mark>。 代码示例:

```
const sdkAppId = '140xxxx'; // 请修改为自己的SDK APP ID。
const channelId = 'xxx'; // 请修改为自己的音频入口 ID
wx.navigateToMiniProgram({
    appId: 'wxf3ac5a9dfc918430',
    //path参数请参照前文说明填写
    path:`/pages/call/call?sdkAppId=${sdkAppId}&channelId=${channelId}`,
    success(res) {
        // 打开成功
    },
    fail(err) {
        // 打开失败
    },
})
```

企业微信里发起音频客服

- 1. 调用 wx.agentConfig 注入应用权限;
- 3. 调用 launchMiniprogram 打开小程序;
 详细请参考 跳转到小程序。

App 内发起音频客服

使用微信提供的 OpenSDK 拉起 TCCC 联络小程序,请参见 App 拉起小程序功能。

- Android 开发示例
- iOS 开发示例

导入代码片段

如果您还未申请到小程序,或者想先测试预览跳转效果。我们也提供了小程序代码片段,方便您快速跳转 **TCCC 联 络小程序** 平台。



1. 在微信开发者工具选择项目的界面中,左侧可以选择代码片段页卡,单击右上角的导入按钮。

• •			
Q 输入项目名称	代码片段		
小程序项目	代码片段是一种可分享的小项目 了無	释详 情	2 导入 管理
小程序		tccc-callout-demo	tccc-callout-demo
小游戏	+	/Users/gavinwjwang/WeChatPr 2	/Users/gavinwjwang/WeChatPr 3
代码片段 1			
公众号网页项目 其他	tccc-callout-demo /Users/gavinwjwang/WeChatPr 1	minicode-1 /Users/gavinwjwang/WeChatPr 4	minicode-1 /Users/gavinwjwang/WeChatPr 5
	minicode-1 /Users/gavinwjwang/WeChatPr 6	test /Users/gavinwjwang/WeChatPr 7	NodesRefError /Users/gavinwjwang/WeChatPr 8
注销 >			

2. 粘贴 https://developers.weixin.qq.com/s/daMLfYmC7Zzl 到输入框中。

腾讯云

● ● Q 输入项目名称 小程序项目	导入代码片段 链接或 ID		导入 管理
小程序 小游戏 代码片段		取消 确定	c-callout-demo .ers/gavinwjwang/WeChatPr
公众号网页项目 其他	tccc-callout-demo /Users/gavinwjwang/WeChatPr 1	minicode-1 /Users/gavinwjwang/WeChatPr 4	minicode-1 /Users/gavinwjwang/WeChatPr 5
	minicode-1 /Users/gavinwjwang/WeChatPr 6	test /Users/gavinwjwang/WeChatPr 7	NodesRefError /Users/gavinwjwang/WeChatPr 8
浸 注销 >			

3. 导入代码后,输入您的 SDK APP ID 和 在线客服入口 ID 即可选择真机调试。运行效果如下



●●●●● WeChat중	14:59	100% (),
腾讯云 昭 此Demo用于: 心F	乎叫中用户站 演示如何拉起開 月户端小程序接	端Demo 鹭讯云呼叫中 _钱 入
^{跳转参数}	<mark>入您的SD</mark> _{请输入sdkAp}	K APP ID
渠道入口ID <mark>输入</mark>	^{请输入渠道入} 、您的音频, 音频	□ID 入
渠道类型	视频 在线	
更多详细说明请参	参考官网文档	

小程序音频客服集成

我们提供了小程序组件,可以方便您在小程序中快速集成音频客服。您可参考下面相关文档。

- 腾讯云联络中心用户端小程序集成指引
- 腾讯云联络中心用户端小程序 SDK API 概览以及示例

Android 音频客服集成

本文主要介绍如何快速地将腾讯云联络中心用户端音频客服(TCCC User SDK(Android))集成到您的项目 中,只要按照如下步骤进行配置,就可以完成 SDK 的集成工作。

开发环境要求

- Android Studio 3.5+。
- Android 4.1 (SDK API 16)及以上系统。

集成 Android端 SDK (aar、jar)

手动下载(aar、jar)



目前我们暂时还未发布到 mavenCentral ,您只能手动下载 SDK 集成到工程里:

- 1. 下载最新版本 TCCC User SDK。
- 2. 在工程根目录下的 build.gradle 中,指定本地仓库路径。



implementation fileTree(dir: "libs", includes: ['*.aar', '*.jar'])

3. 在 app/build.gradle的defaultConfig 中,指定 App 使用的 CPU 架构。



```
    ● 说明:
    目前 TCCC User SDK 支持 armeabi , armeabi-v7a 和 arm64-v8a。
```



4. 在 app/src/AndroidManifest.xml 中,指定 App 不允许应用参与备份和恢复基础架构。

T	CCCSimpleDemo $ angle$ app $ angle$ src $ angle$ main $ angle$ $rac{40}{400}$ AndroidMa	anifest.xml 🔨 🔺 app 🔻 🛄 HUAWEI LIO-AL00 👻 🕨 🔅 🛱 🖏 🦚 義 🔳
ect	$\blacksquare \text{ Project } \bullet \qquad \bigoplus \ \underline{\underline{\underline{\underline{5}}}} \ \underline{\underline{\underline{5}}} \ \underline{\underline{5}} \ \underline{\underline{5}} \ \underline{\underline{5}} \ \underline{\underline{5}} \ \underline{\underline{5}} \ \underline{5} \ 5$	AndroidManifest.xml ×
ProJ	TCCCSimpleDemo ~/work/ugit/TCCC/TC	1 xml version="1.0" encoding="utf-8"?
	> 🖿 .gradle	<pre>2</pre>
¥	> 🖿 .idea	<pre>3 package="com.tencent.tcccsimpledemo"></pre>
mmo	🕆 📑 app	4
5	> build	5 application
	> libs	6 android:allowBackup="false"
e	✓ src	7 A android:icon="@minman/ic_launchen"
nag	> androidTest	
Mai	🗠 🖿 main	8 android:label="TCCC Simple Demo"
Irce	> 🖿 java	9 🚾 android:roundIcon="@mipmap/ic_launcher_round"
SOL		10 android:supportsRtl="true"
¥ A	AndroidManifest.xml	11 android:theme="@style/Theme.TCCCSimpleDemo">
10	test	12 🖯 <activity< td=""></activity<>
	🧑 .gitignore	13 android:name=".CallingActivity"
	🔊 build.gradle	14 android:exported="false" />
	🗧 proguard-rules.pro	
	> 🖿 gradle	16 android:name=" MainActivity"

5. 单击 I Sync Now, 完成 TCCC User SDK 的集成工作。

配置 App 权限

在 AndroidManifest.xml 中配置 App 的权限,TCCC SDK 需要以下权限:

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"
/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"
/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
/>
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS"
/>
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
</uses-permission android:name="android.
```

设置混淆规则

在 proguard-rules.pro 文件,将 TCCC SDK 相关类加入不混淆名单:

-keep class com.tencent.** { *; }

Android 音频客服集成相关文档



- 快速跑通云联络中心用户端 Android Demo
- 快速集成云联络中心用户端Android SDK
- 云联络中心用户端Android SDK API 概览以及示例

iOS 音频客服集成

本文主要介绍如何快速地将腾讯云联络中心 iOS User SDK 集成到您的项目中,只要按照如下步骤进行配置,就 可以完成 SDK 的集成工作。

开发环境要求

- Xcode $9.0+_{\circ}$
- iOS 9.0 以上的 iPhone 或者 iPad 真机。
- 项目已配置有效的开发者签名。

集成 iOS SDK

方案一: 使用 CocoaPods

1. 安装 CocoaPods

在终端窗口中输入如下命令(需要提前在 Mac 中安装 Ruby 环境):

sudo gem install cocoapods

2. 创建 Podfile 文件

进入项目所在路径,输入以下命令行之后项目路径下会出现一个 Podfile 文件。

pod init

3. 编辑 Podfile 文件

根据您的项目需要编辑 Podfile 文件:

```
platform :ios, '8.0'
target 'App' do
    pod 'TCCCCloud_Ios', :podspec =>
'https://tccc.qcloud.com/assets/doc/user/release/TCCCCloud_Ios.podspec'
    end
```

4. 更新并安装 SDK

• 在终端窗口中输入如下命令以更新本地库文件,并安装 SDK:



pod install

• 或使用以下命令更新本地库版本:

pod update

pod 命令执行完后,会生成集成了 SDK 的 .xcworkspace 后缀的工程文件,双击打开即可。

方案二:手动下载

- 1. 下载最新版本 TCCC User SDK。
- 2. 打开您的 Xcode 工程项目,选择要运行的 target,选中 Build Phases 项。



3. 单击 Link Binary with Libraries 项展开,单击底下的 "+"号图标去添加依赖库。





4. 依次添加下载的 TCCCCloud.Framework、ImSDK_CPP.framework、TXFFmpeg.xcframework、 TXSoundTouch.xcframework及其所需依赖库 GLKit.framework、AssetsLibrary.framework、 SystemConfiguration.framework、libsqlite3.0.tbd、CoreTelephony.framework、 AVFoundation.framework、OpenGLES.framework、Accelerate.framework、 MetalKit.framework、libresolv.tbd、MobileCoreServices.framework、libc++.tbd、 CoreMedia.framework。





 95. 单击 General,选择 Frameworks,Libraries,and Embedded Content,检查 TCCCCloud.framework 所需要动态库 TXFFmpeg.xcframework、 TXSoundTouch.xcframework、ImSDK_CPP.framework 是否已经添加,是否正确选择选择 Embed & Sign,如果没有单击底下的 "+"号图标依次添加。





6. 在工程 target 中 Build Settings 的 Other Linker Flags 增加 - ObjC 配置。

늘 🛛 🎞 Q 🛆 🔗 🗗	 器(く >) 🗙 Main.story	yboard (Base)	🛃 tccc-user-ios-example.:	codeproj	m [*] ViewControl	ler.mm		Œ
✓	K tccc-user-ios-example							
tccc-user-ios-example	∎ Gene	ral Signing	& Capabilities Resource Tags	Info Bu	uild Settings Build	Phases Bu	uild Rules	
> 🚞 Debug		+ Basic	Customized All Combined	Levels		😑 Other L	inker Flags	8
[™] ViewController.mm	PROJECT			-				
h AppDelegate.h	🛃 tccc-user-ios-exam	Linking						
M AppDelegate.m		✓ Linking			A toco-user	r-ios-evample		
h SceneDelegate.h	TARGETS	~	Other Linker Flags		-ObiC			
m SceneDelegate.m	A tccc-user-ios-exam		Debug		-ObjC			
h ViewController.h			Release		-ObjC			
🗙 Main.storyboard								
🖾 Assets.xcassets								
🔀 LaunchScreen.storyboard								
🖽 Info.plist								

配置 App 权限

 如需使用 SDK 提供的音视频功能,需要给 App 授权麦克风的使用权限。在 App 的 Info.plist 中添加以下两项,分别对应麦克风在系统弹出授权对话框时的提示信息。选择Privacy – Microphone Usage Description,并填入麦克风使用目的提示语。



M toco-user-ios-example							
■ Gen	eral Signing & Cap	pabilities Resource Tags Info B	Build S	ettings Bu	ild Phases Build Ru	ules	
PROJECT	✓ Custom iOS	Target Properties					
🛃 tccc-user-ios-exam		Кеу		Туре	Value		
		Bundle name	٥		\$(PRODUCT_NAME	=)	
TADGETS		Bundle identifier	\$		\$(PRODUCT_BUND	LE_IDEN	
TARGETS		InfoDictionary version	٥		6.0		
🛃 tccc-user-ios-exam		Main storyboard file base name	٥		Main		
	J	Bundle version	٥		\$(CURRENT_PROJ	ECT_VER	
		Launch screen interface file base na	🗘		LaunchScreen		
		Executable file	\$		\$(EXECUTABLE_NA	AME)	
		Application requires iPhone environ	· \$		YES	\$	
		> Supported interface orientations (iP	• \$		(3 items)		
		Application supports indirect input e	🗘		YES	\$	
		> Application Scene Manifest	\$		(2 items)		
		Bundle OS Type code	\$		\$(PRODUCT_BUND	LE_PACł	
		Development localization	\$		\$(DEVELOPMENT_	LANGUA(
		> Supported interface orientations (iP	• \$		(4 items)		
		Bundle version string (short)	Ŷ	Otring			
		Privacy - Microphone Usage D 🛟	00		◇ 需要访问您的麦克风	才可以正	
	> Document T	ypes (0)					
	> Exported Ty	pe Identifiers (0)					

2. 如需 App 进入后台仍然运行相关功能,可在 XCode 中选中当前工程项目,并在 Capabilities 下将设置项 Background Modes 设定为 ON,并勾选 Audio, AirPlay and Picture in Picture,如下图所示:

📁 🗵 🎞 Q 🛆 🧇 🕼 🗆 🗐	₩ < > 🛛 🗛 tccc-	user-ile.xcodeproj m [°] ViewController.mm h GenerateTestUserSig.h M GenerateTestUserSig.n	₹∣⊞
✓	🔼 tccc-user-ios-example		< 🔺 >
The second se	🚺 Gen	eral Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules	
~ 🚞 Debug		+ Capability All Debug Release	
h GenerateTestUserSig.h	PROJECT		
M GenerateTestUserSig.m	🛃 tccc-user-ios-exam	✓ Signing	
m [*] ViewController.mm		V Automatically manage signing	
h AppDelegate.h	TARGETS	Xcode will create and update profiles, app IDs, and certificates	
M AppDelegate.m	▲ tccc-user-ios-exam		
h SceneDelegate.h		Team None 🙂	
M SceneDelegate.m		Bundle Identifier com.tencent.tcccsdk.ios.demo.tccc-user-ios-exan	
h ViewController.h		v ios	
🗙 Main.storyboard		Provisioning Profile Xcode Managed Profile	
🖾 Assets.xcassets			
🗙 LaunchScreen.storyboard			
🖽 Info.plist 🛛 🕅		Status 🛕 Signing for "tccc-user-ios-example" requires a	
m main.m		in the Signing & Capabilities editor.	
> 📰 Products			
🗸 📷 Frameworks			
츰 CoreMedia.framework		V 😂 Background Modes	W
⊘ libc++.tbd		Mades 🔽 Audio, AirDiau and Disture is Disture	
音 MobileCoreServices.frame		Modes V Audio, AirPlay, and Picture in Picture	
🔗 libresolv.tbd		Voice over IP	
🚔 MetalKit.framework		External accessory communication	
👄 Accelerate.framework		Uses Bluetooth LE accessories	
👄 OpenGLES.framework		Acts as a Bluetooth LE accessory	



代码实现

目前我们只提供了通过 C++ 接口,可以用下面代码引入头文件

```
// 引入C++头文件
#include "TCCCCloud/tccc/include/ITCCCCloud.h"
// 使用tccc命名空间
using namespace tccc;
// 获取tcccSDK 单例
ITCCCCloud* tcccSDK = getTCCCShareInstance();
// 获取SDK版本号
const char * version = tcccSDK->getSDKVersion();
```

具体编码实现可参考 API 概览以及示例

iOS 音频客服集成相关文档

- 快速跑通云联络中心用户端 iOS Demo
- 快速集成云联络中心用户端 iOS SDK
- 云联络中心用户端 iOS SDK API 概览以及示例

常见问题

如何查看 TCCC 日志?

TCCC 的日志默认压缩加密,后缀为 .log。

- iOS:日志路径: sandbox/Documents/tccc
- Android 日志路径: /sdcard/Android/data/包名/files/tccc

在iOS下回调是否都在主线程

目前在iOS下回调都不在主线程,需要业务层面上判断并且把他转为主线线程

```
if ([NSThread isMainThread]) {
    // 在主线程,直接可以处理
    return;
}
dispatch_async(dispatch_get_main_queue(), ^{
    // 回调在非主线程。
});
```

TCCC 怎么校验生成的 UserSig 是否正确?

可通过云 API 调用生成UserSig,具体可查看 创建用户数据签名 接口文档。

集成用户端视频客服

最近更新时间: 2025-06-30 19:46:12

本文主要介绍如何快速集成腾讯云联络中心视频客服,只要按照如下步骤进行配置,就可以跑通用户端的视频客服。

前提条件

- 您已 注册腾讯云 账号,并完成 实名认证。
- 您已开通云联络中心服务,并创建了云联络中心实例。

集成视频客服前置准备

1. 登录 联络中心管理端,选择登录的联络中心 > 管理端 > 视频客服 > 技能组管理 > 添加技能组。

📞 电话外呼 📔 🗌		• 服务记录					S. 🖪 0/2 👩 🕸 🕅
── 数据分析	技能组管理						
□ 服务记录							
⊙ 实时监控	+ 添加技能组 3	请输入技能	组名称或客服: Q	adrien test			
○ 在线客服	技能组名称	接待上限	操作	客服姓名	郎箱	优先级 ()	操作
ⓒ 电话客服 →	adrien test	1	編辑 删除	'n. ("	Carrolline Services 30M	3 🧪	删除
◎ 音频客服 ~	gahou	1	编辑 删除	5		3 🧪	删除
回 视频客服 ^	gavin test	1	編辑 删除		-hanarti 'a@+ 108	1 🧪	删除
 · 投龍組管理 · 渠道管理 		1	編辑 删除		⑦ 添加新	客服	
• 视频客服设置	共 4 条 10 ▼ 条 /	/页 🛛 🔺 1	/1页 ▶ ⊨	共 3 条	10 👻	条/页 🛛 🖣	1 /1页 ▶ ⊮
◎ 通用设置							
🛆 客服管理							
○ 客户管理 ~							

2. 添加新客服到对应的技能组中。

电话外呼		工作台 📑 管理端	L 服务记录						<u> </u>) 0/2 👧 🖻
➡ 数据分析		技能组管理								
🗐 服务记录										
♂ 实时监控		+ 添加技能组	请输入	支能组名称或客服!	Q	gavin test				
○ 在线客服		技能组名称	接待上限	操作		客服姓名	邮箱	优先级 🛈	操作	
☺ 电话客服	×	adrien test	1	编辑 删除		gavinwjwang	gaviawig 💽	3 🧪	删除	
◎ 音频客服	~	gahou	1	编辑 删除			(1) 添	加新客服 2添;	加客服	
圖 视频客服	^						10	▼ 条 / 页 🛛 🕅	(1	/1页 ▶ ⊮
 技能组管理 		gavin test	1	编辑删除	1.选择	技能组				
- 渠道管理			1	编辑 删除						
- 视频客服设置		共 4 条 10 ▼ 条 /	页 🛛 🖣	1 /1页 ▶						
⑥ 通用设置										



3. 新增视频入口。选择管理端 > 渠道管理 > 配置视频入口 > 新增。

• 技能组管理	渠道管理			
- IVR管理	配置视频入口 绑定小程序			
• 满意度评价				
• 号码管理				
• 电话客服设置				
· 自助服务		05-		
• 收号设置		新增视频入口 12c		×
• 放音文件管理		视频入口名称 ★	请输入视频入口名称	
- 智能质检		关联技能组 d3	请选择技能组 🔻	3
◎ 音频客服 ~				
⊠ 视频客服 ^		f2c	4 确认 取	消
• 技能组管理				
· 渠道管理 1				
• 视频客服设置				
 ◎ 通用设置 × 				

4. 复制视频入口 ID。

📞 电话外呼 📔 🔤	_ 工作台 📕 管理端 🕒 服务记录 🛄 新手配]	置指引		
 技能组管理 	渠道管理			
。 IVR管理	配置视频入口 绑定小程序			
• 满意度评价	964 0			
。 号码管理	AV Free	复制视频入口ID		
• 电话客服设置	视频入口名称	视频入口ID	关联技能组	操作
• 自助服务	gahouliao	io 🗂	gavin test	编辑 删除
• 收号设置	adrien	۲ <u>۵</u>	adrien test	编辑 删除
• 放音文件管理		05	产品体验	编辑 期间全
• 智能质检	waahh	0.1	7 6674-302	348-428 0031994
◎ 音频客服 ~	dev	9fi⊡	gahou	编辑 删除
□□ 视频客服 ^	Android SDK		gavin test	编辑 删除
 技能组管理 	共 5 条			请选择 ▼ 条 / 页
 渠道管理 				
• 视频客服设置				
◎ 通用设置				

腾讯云联络中心提供了多种方式集成视频客服,包括跳转到 TCCC 联络小程序平台完成视频客服功能,同时也提供 了小程序 SDK、Android SDK 方便开发者实现视频客服相关功能。

跳转到 TCCC 联络小程序平台

我们提供了 TCCC 联络小程序平台,可以方便您在微信小程序里、企业微信里、自有 App 内打开 **TCCC 联络小程** 序 完成视频客服。

跳转参数说明

拉起 TCCC 联络小程序平台需要用到以下参数。 请先记录这些参数,后文将会详细介绍具体开发方式。

参数	值	备注
appl d	wxf3ac5a9dfc918430	固定值
path	/pages/video/video?sdkAppId= {YourAppID}&channelId={YourChannelID}	请将具体参数值修改为您自己 的

path 参数说明

参数	类型	必填	备注
sdkAppId	string	是	腾讯云联络中心应用 ID
channelld	string	是	视频入口 ID,参见 复制视频入口 ID 。

微信小程序内进行跳转

使用 wx.navigateToMiniProgram 拉起 **TCCC 联络小程序**即可,请参见 <mark>打开另一个小程序</mark>。 代码示例:

```
const sdkAppId = '140xxxx'; // 请修改为自己的SDK APP ID。
const channelId = 'xxx'; // 请修改为自己的视频入口ID
wx.navigateToMiniProgram({
    appId: 'wxf3ac5a9dfc918430',
    //path参数请参照前文说明填写
    path:`/pages/video/video?sdkAppId=${sdkAppId}&channelId=${channelId}`,
    success(res) {
        // 打开成功
    },
    fail(err) {
        // 打开失败
    },
})
```

企业微信里发起视频客服

- 1. 调用 wx.agentConfig 注入应用权限;
- 3. 调用 launchMiniprogram 打开小程序;
 详细请参见 跳转到小程序。

App 内发起视频客服

🔗 腾讯云

使用微信提供的 OpenSDK 拉起 TCCC 联络小程序,请参见 App 拉起小程序功能。

- Android 开发示例
- iOS 开发示例

导入代码片段

如果您还未申请到小程序,或者想先测试预览跳转效果。我们也提供了小程序代码片段,方便您快速跳转 **TCCC 联 络小程序** 平台。

1. 在微信开发者工具选择项目的界面中,左侧可以选择代码片段页卡,点击右上角的导入按钮。

••			
Q 输入项目名称	代码片段		
小程序项目	代码片段是一种可分享的小项目 了解	¥详情	2 导入 管理
小程序	_L_	tccc-callout-demo	tccc-callout-demo
小游戏	Т	/Users/gavinwjwang/WeChatPr 2	/Users/gavinwjwang/WeChatPr 3
代码片段			
公众号网页项目	tccc-callout-demo	minicode-1	minicode-1
其他	/Users/gavinwjwang/WeChatPr 1	/Users/gavinwjwang/WeChatPr 4	/Users/gavinwjwang/WeChatPr 5
	minicode-1 /Users/gavinwjwang/WeChatPr 6	test /Users/gavinwjwang/WeChatPr 7	NodesRefError /Users/gavinwjwang/WeChatPr 8
注销 >			



2. 粘贴 https://developers.weixin.qq.com/s/daMLfYmC7Zzl 到输入框中。

••	导入代码出码		
Q 输入项目名称			ר -
小程序项目	链接或 ID		导入 管理
小程序			
		取 用	c-callout-demo
小游戏		2	ers/gavinwjwang/weCnatPr
代码片段			
公众号网页项目	tccc-callout-demo	minicode-1	minicode-1
其他	/Users/gavinwjwang/WeChatPr 1	/Users/gavinwjwang/WeChatPr 4	/Users/gavinwjwang/WeChatPr 5
	minicode-1	taet	NodesBefError
	/Users/gavinwjwang/WeChatPr	/Users/gavinwjwang/WeChatPr	/Users/gavinwjwang/WeChatPr
注销 >			

3. 导入代码后,输入您的SDK APP ID 和 在线客服入口 ID 即可选择真机调试。运行效果如下



●●●●● WeChat夺	15:00	100% ()
腾讯云	呼叫中用户端	≣Demo
此Demo用于 心	[–] 演示如何拉起腾 用户端小程序接	讯云呼叫中 入
跳转参数 输之	入您的SDK。	APP ID
sdkAppId	请输入sdkApp	old
渠道入口ID <mark>输入</mark>	请输入渠道入I 您的视频入	
	音频	
渠道类型	视频	\checkmark
	在线	
更多详细说明请	参考官网文档	

小程序视频客服集成

我们提供了**腾讯云联络中心小程序 SDK**,方便您在微信小程序里集成视频客服功能。您可参考下面相关文档。

- 腾讯云联络中心用户端小程序集成指引
- 腾讯云联络中心用户端小程序 SDK API 概览以及示例

用户端 iOS 视频客服集成

如果您需要在您的 iOS 移动设备或者嵌入式集成视频客服,可参见下面步骤。

开发环境要求

- Xcode 9.0+。
- iOS 9.0 以上的 iPhone 或者 iPad 真机。
- 项目已配置有效的开发者签名。

集成 iOS SDK

方案一: 使用 CocoaPods

1. 安装 CocoaPods

在终端窗口中输入如下命令(需要提前在 Mac 中安装 Ruby 环境):

sudo gem install cocoapods

2. 创建 Podfile 文件

进入项目所在路径,输入以下命令行之后项目路径下会出现一个 Podfile 文件。

pod init

3. 编辑 Podfile 文件

根据您的项目需要编辑 Podfile 文件:

```
platform :ios, '8.0'
target 'App' do
    pod 'TCCCCloud_Ios', :podspec =>
'https://tccc.qcloud.com/assets/doc/user/release/TCCCCloud_Ios.podspec'
end
```

- 4. 更新并安装 SDK
 - 在终端窗口中输入如下命令以更新本地库文件,并安装 SDK:

pod install

○ 或使用以下命令更新本地库版本:

pod update

pod 命令执行完后,会生成集成了 SDK 的 .xcworkspace 后缀的工程文件,双击打开即可。

方案二:手动下载

- 1. 下载最新版本 TCCC User SDK。
- 2. 打开您的 Xcode 工程项目,选择要运行的 target,选中 Build Phases 项。



= X II Q A 🔗 🗗		🔡 < 📏 🕹 X Main.storyboard (Base)						
✓ ▲ tccc-user-ios-example		A tccc-user-ios-example						
Tecc-user-ios-example		■ Gene	neral Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules					
> 🔚 Debug			+ () Filter					
[™] ViewController.mm		PROJECT						
h AppDelegate.h		🛃 tccc-user-ios-exam	> Target Dependencies (0 items)					
M AppDelegate.m								
h SceneDelegate.h		TARGETS	Run Build Tool Plug-ins (0 items)					
M SceneDelegate.m		A tccc-user-ios-exam	> Compile Sources (5 items)					
h ViewController.h								
🔀 Main.storyboard			➤ Link Binary With Libraries (0 items)					
🖾 Assets.xcassets								
🔀 LaunchScreen.storybo	bard		> Copy Bundle Resources (3 items)					
🖽 Info.plist	?							

3. 单击 Link Binary with Libraries 项展开,单击底下的"+"号图标去添加依赖库。

General								
o o noral	Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules							
PROJECT	() Filter							
🖪 tccc-user-ios-exam	> Target Dependencies (0 items)							
	> Run Build Tool Plug-ins (0 items)							
TARGETS								
A tccc-user-ios-exam	Compile Sources (5 items)							
	✓ ink Rinary With ihraries (16 items)							
	Name	Status						
	CoreMedia.framework	Required \$						
	چ libc++.tbd	Required 🗘						
	AmobileCoreServices.framework	Required 🗘						
	ی≣ libresolv.tbd	Required 🗘						
	🚔 MetalKit.framework	Required 🗘						
	🚔 Accelerate.framework	Required 🗘						
	🚔 OpenGLES.framework	Required 🗘						
	🚔 AVFoundation.framework	Required 🗘						
	🚔 CoreTelephony.framework	Required 🗘						
	₂ libsqlite3.0.tbd	Required 🗘						
	🚔 SystemConfiguration.framework	Required 🗘						
	🚔 AssetsLibrary.framework	Required 🗘						
	🚔 GLKit.framework	Required 🗘						
	🚔 TXSoundTouch.xcframework	Required 🗘						
	TXFFmpeg.xcframework	Required 🗘						
	🙃 TCCCCloud.framework	Required 🗘						
	+ - Drag to reorder linked binaries							
	PROJECT ▲ tccc-user-ios-exam TARGETS ▲ tccc-user-ios-exam	PROJECT + * target Dependencies (0 items) * Run Build Tool Plug-ins (0 items) * Run Build Tool Plug-ins (0 items) * Compile Sources (5 items) * Link Binary With Libraries (16 items) * CoreMedia framework @ libc++.tbd @ MobileCoreServices.framework @ libresolv.tbd @ MetalKit.framework @ DepenGLES.framework @ DopenGLES.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ Libr.Histon @ CoreTelephony.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ SystemConfiguration.framework @ CoreTelephony.framework @ TyE						

4. 依次添加下载的 TCCCCloud.Framework、ImSDK_CPP.framework、TXFFmpeg.xcframework、 TXSoundTouch.xcframework 及其所需依赖库 GLKit.framework、AssetsLibrary.framework、



SystemConfiguration.framework、libsqlite3.0.tbd、CoreTelephony.framework、 AVFoundation.framework、OpenGLES.framework、Accelerate.framework、 MetalKit.framework、libresolv.tbd、MobileCoreServices.framework、libc++.tbd、 CoreMedia.framework。

🗸 🛃 tccc-user-ios-example	Kcc-user-ios-example								
tccc-user-ios-example	E Ge	eneral	Signing & Capa	abilities	Resource Tags	Info	Build Settings	Build Phases	Build Rules
> 🚞 Debug		+						🕞 Filter	
m [*] ViewController.mm	PROJECT								
h AppDelegate.h	Ccc-user-ios-exam > Target Dependencies (0 items)								
M AppDelegate.m									
h SceneDelegate.h	TARGETS	TARGETS							
M SceneDelegate.m	A tccc-user-ios-exam	tccc-user-ios-exam Compile Sources (5 items)							
h ViewController.h									
🗙 Main.storyboard		✓ Link Binary With Libraries (16 items)							
🖾 Assets.xcassets									0
🗙 LaunchScreen.storyboard				Indille					Status
🖽 Info.plist				- Co	oreMedia.framewor	k			Required 🗘
m main.m				lik =	oc++.tbd				Required 🗘
> 📰 Products				🚔 M	obileCoreServices.	ramewo	ork		Required 🗘
> 📷 Frameworks				,⊟ lik	presolv.tbd				Required 🗘
				🚔 M	etalKit.framework				Required 🗘
				🚔 Ad	ccelerate.frameworl	k			Required 🗘
				🚔 O	penGLES.framewor	k			Required 🗘
				🚘 A\	/Foundation.framew	vork			Required 🗘
				🚔 Co	oreTelephony.frame	work			Required 🗘
				,≣ lik	osqlite3.0.tbd				Required 🗘
				🚔 Sy	ystemConfiguration	.framew	ork		Required 🗘
				🚔 As	ssetsLibrary.framew	/ork			Required 🗘
				🚘 GI	LKit.framework				Required 🗘
				🚘 T)	KSoundTouch.xcfra	mework			Required 🗘
				🚘 Т)	XFFmpeg.xcframew	ork			Required 🗘
				🚘 то	CCCCloud.framewo	rk			Required 🗘
				+ -			Drag to reorde	er linked binaries	
		-							

 95. 单击 General,选择 Frameworks,Libraries,and Embedded Content,检查 TCCCCloud.framework 所需要动态库 TXFFmpeg.xcframework、 TXSoundTouch.xcframework、ImSDK_CPP.framework 是否已经添加,是否正确选择选择 Embed & Sign,如果没有单击底下的 "+"号图标依次添加。





6. 在工程 target 中 Build Settings 的 Other Linker Flags 增加 - ObjC 配置。

🗎 🛛 🖬 Q 🛆 🗇 🗗		器 \langle $ angle$ 🗙 Main.storyboard (Base)		🛃 tccc-	🛃 tccc-user-ios-example.xcodeproj		m⁺ ViewController.mm			Œ
Kore Karley K		Kara terestrest to the second								
tccc-user-ios-example		∎ Ger	neral Signing	& Capabilities	Resource Tags	Info Bui	ild Settings Buil	d Phases Bu	ild Rules	
> 🖿 Debug			+ Basic	Customized	All Combined	Levels		😑 Other L	inker Flags	8
m [*] ViewController.mm		PROJECT				,				
h AppDelegate.h		🛃 tccc-user-ios-exam								
m AppDelegate.m			↓ Linking				A tece-use	er-ios-example		
h SceneDelegate.h		TARGETS	~	Other Linker Fla	as		-ObiC			
M SceneDelegate.m		A tccc-user-ios-exam		Debug			-ObjC			
h ViewController.h				Release			-ObjC			
🔀 Main.storyboard										
🖾 Assets.xcassets										
🔀 LaunchScreen.storyb	oard									
🖽 Info.plist										

配置 App 权限



 如需使用 SDK 提供的音视频功能,需要给 App 授权麦克风的使用权限。在 App 的 Info.plist 中添加以下两项,分别对应麦克风在系统弹出授权对话框时的提示信息。选择Privacy – Microphone Usage Description、Privacy – Camera Usage Description,并填入麦克风、摄像头使用目的提示语。

A tccc-user-ios-example						<
		Capabilities Resource Tags Info E				
PROJECT	✓ Custom iOS	Target Properties				
🛃 tccc-user-ios-exam		Кеу		Туре	Value	
		Bundle name	٥		\$(PRODUCT_NAME)	
TADGETS		Bundle identifier	٢		\$(PRODUCT_BUNDLE_IDENTIFI	
		InfoDictionary version	٥		6.0	
🛃 tccc-user-ios-exam		Main storyboard file base name	٥		Main	
		Bundle version	٥		\$(CURRENT_PROJECT_VERSION	
		Launch screen interface file base na	٥		LaunchScreen	
		Executable file	Ŷ	String	\$(FXECUTABLE NAME)	5
		Privacy - Microphone Usage Descri	٥		需要访问您麦克风才可以正常通话	
		Application requires iPhone environ	٥	Boolean	YES 🗘	
		> Supported interface orientations (iP	٥		(3 items)	
		Application supports indirect input e	٥		YES 🗘	
		> Required background modes	٥		(1 item)	
		> Application Scene Manifest	٥	Dictionary	(2 items)	
		Bundle OS Type code	٥	String	\$(PRODUCT BUNDLE PACKAGE	
		Privacy - Camera Usage Descripti 💲 🕻			需要访问您摄像头才可以正常通话	
		Default localization	$\hat{\mathbf{v}}$	String	DEVELOPMENT_LANGUAGE)	_
		> Supported interface orientations (iP	٥		(4 items)	
		Bundle version string (short)	\$		\$(MARKETING_VERSION)	
+ - 🕞 Filter	> Document 1	Types (0)				

2. 如需 App 进入后台仍然运行相关功能,可在 XCode 中选中当前工程项目,并在 Capabilities 下将设置项 Background Modes 设定为 ON,并勾选 Audio, AirPlay and Picture in Picture, 如下图所示:





代码实现

目前我们只提供了通过 C++ 接口,可以用下面代码引入头文件



iOS 音频客服集成相关文档

- 快速跑通云联络中心用户端 iOS Demo
- 快速集成云联络中心用户端 iOS SDK
- 云联络中心用户端 iOS SDK API 概览以及示例

用户端 Android 视频客服集成

如果您需要在您的 Android 移动设备或者嵌入式集成视频客服,可参考下面步骤。

下载 SDK 和 TCCCSimpleDemo 源码

根据实际业务需求 tccc-user-android-example 源码。

配置 TCCCSimpleDemo 工程文件

- 1. 找到并打开 Debug/src/main/java/com/tencent/debug/GenerateTestUserSig.java 文件。
- 2. 设置 GenerateTestUserSig.java 文件中的相关参数:
 - VIDEO_CHANNELID: 请设置为实际的 应用配置入口 APP ID。
 - SDKAPPID: 请设置为实际的腾讯云联络中心 SDKAppID。
 - SECRETID: 请设置为实际的密钥 ID 信息。
 - SECRETKEY: 请设置为实际的密钥 Key 信息。

월 🔲 Project 👻 😳 🚊 🌣 —	C Generat	eTestUserSig.java 😒
TCCCSimpleDemo ~/work/ugit/TCCC/TC	17 6	▲ 16 🛫 11 ∧ ∨ 🖁
🛛 📏 🖿 .gradle	18	public class GenerateTestUserSig {
) > 🖿 .idea	19	
> 🐂 app	20	/***
🖇 🕆 🛅 Debug	21	* 应用配置入口 APP ID
> > 🖿 build	22	*/
, > iiibs	23	public static final String VIDEO_CHANNELID = "PLACEHOLDER"; 粘贴应田配署入口APPID
v ™ src	24	
> 🖿 androidTest	25	* 塘讯云 SDKAnnId 重要装换为你自己账号下的 SDKAnnId
🖌 📉 main	26	* 进入路讯云呼叫中心[按制台][https://console.cloud.tencent.com/ccc.) 创建应用 即可振到 SDKAppId
🗸 🖿 java	27	· CARACTERIZATION CONTRACTOR CONT
Com.tencent.debug Com.tencent.debug	20	
G GenerateTestUserSig	28	
AndroidManifest.xml	29	public static final int SURAPPID = 0; ADVERTHY TO VERTIC
> test	30 E	/**
gitignore	31	* 计算签名用的加密密钥ID, [查看秘钥](https://console.cloud.tencent.com/cam/capi)
ar build.gradie	32	* 注意:该方案仅适用于调试Demo,正式上线前请将 UserSig 计算代码和密钥迁移到您的后台服务器上,以避免加密密钥泄露导致的流量盗F
consumer-rules.pro	33	* 文档: https://cloud.tencent.com/document/product/679/58260
proguard-rules.pro	34	*/
/ gradie	35	public static final String SECRETID = "PLACEHOLDER"; 粘帖腾讯云秘钥D
lo .grugnore	36	
aradle properties	37 0	/**
aradiew	38	* 计算签名用的加密密钥Key、[查看秘钥](https://console.cloud.tencent.com/capi)
gradlew.bat	39	* 注意:该方案仅适用于调试Demo,正式上线前请将 UserSig 计算代码和密钥迁移到您的后台服务器上,以避免加密密钥泄露导致的流量鉴[
a local properties	40	* 文档: https://cloud.tencent.com/document/product/679/58260
QuickStartDemo.md	41 0	• */
QuickStartSDK.md	42	public static final String SECRETKEY = "PLACEHOLDER", 粘贴腾讯元秘钥Key
g # README.md	1.3	
settings.gradle	40	
> IIII External Libraries	94	

△ 注意

- 本文提到的生成 UserSig 的方案是在客户端代码中配置 SECRETID、SECRETKEY,该方法中 SECRETID、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 TCCCSimpleDemo 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 创建用户数 据签名。

编译运行

使用 Android Studio (3.5及以上的版本) 打开源码工程 tccc-user-android-example ,单击运行即可。

Android 运行效果



基本功能如下图所示。



🕛 说明

UI 可能会有部分调整更新,请以最新版为准。

用户端 Android 音视频客服集成相关文档

- 快速跑通 Android Demo
- 快速集成 Android SDK
- Android SDK API 概览以及示例



跳转小程序方式接入

最近更新时间: 2023-07-25 15:27:53

腾讯云联络中心提供用户端音视频小程序,您可以直接跳转小程序或者公众号唤起小程序方式接入: 拉起云联络中心小程序需要用到以下参数:

参 数	值	备注
ap pld	wxf3ac5a9dfc918430	固定值,表示腾讯云联络中心 SaaS 小程序 ID
pat h	/pages/call/call? sdkAppId=1400000000&channeIId=xxx	唤起参数
id	gh_87f162feb200	腾讯云联络中心小程序原始 ID,仅 App 拉起 小程序时使用

path 页面类型:

入 腾讯 六

页面	备注
/pages/call/call	音频渠道页面
/pages/video/video	视频渠道页面
/pages/chat/chat	在线客服页面

渠道 ID 获取方式

视频渠道 ID: 请参考 使用视频客服 。 音频渠道 ID: 请参考 使用音频客服 。 在线渠道 ID: 请参考 使用在线客服 。

小程序内打开小程序进行外呼

推荐使用 wx.openEmbeddedMiniProgram 打开半屏小程序,减少二次确认步骤;接入详细指引请参考: 打开 半屏小程序.

代码示例:

```
const sdkAppId = '140xxxx'; // 请修改为自己的sdkAppId
const channelId = 'xxx'; // 请修改为自己的渠道 ID
wx.navigateToMiniProgram({
    appId: 'wxf3ac5a9dfc918430',
```



```
//path参数请参照前文说明填写
path:`/pages/call/call?sdkAppId=${sdkAppId}&channelId=${channelId}`,
success(res) {
    // 打开成功
},
fail(err) {
    // 打开失败
},
```



API 文档

最近更新时间: 2023-07-25 15:27:53

通用 API

login — 登录

参数		类型	必填	备注
	sdkAppl d	String	是	腾讯云联络中心应用 SdkAppld
	jsCode	String	是	通过 wx.login 获取
ontio	dataEnv	String	否	wx.getUserProfile的encryptedData
ns	datalv	String	否	wx.getUserProfile 的 iv
	mobileE nv	String	否	通过 button open-type="getPhoneNumber" 方式获取
	mobilelv	String	否	通过 button open-type="getPhoneNumber" 方式获取

checkLogin — 检查是否已登录

参数		类型	必填	备注
optio ns	sdkAppl d	String	是	腾讯云联络中心应用 SdkAppld

```
import { login, checkLogin } from 'tccc-wx-sdk/login'
const sdkAppId = '140000000'
// 登录示例代码
checkLogin({ sdkAppId })
   .then((result) => {
      if (result) {
          // 检查登录成功
      } else {
          throw new Error('登录已过期')
      }
}).catch(() => {
      wx.login({
          success: ({ code }) => {
      }
})
```



// 登录 tccc	
jsCode: code,	
sdkAppId	
1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1	
<pre>}).catch(e => {</pre>	
wx.showToast({	
icon: 'error',	
title: e.message	

音频呼叫 API

startCall — 发起音频通话

参数		类型	必填	备注	
options	channelld	String	是	IVR 入口 ID	



endCall — 挂断通话

this.selectComponent('#tcccSdk').endCall();

sendDigits — 发送 DTMF(双音多频信号)

参数



option s	digit s	String	是	DTMF digits 参数是一个字符串,可以包含字符 0-9、*、 #。

this.selectComponent('#tcccSdk').sendDigits(options)

视频呼叫 API

startSession — 发起视频通话

参数		类型	必填	备注
options	channelld	String	是	IVR 入口 ID



endSession — 挂断视频通话

this.selectComponent('#tcccSdk').endSession();

```
switchCamera — 切换摄像头
```

this.selectComponent('#tcccSdk').switchCamera();

```
disableCamera — 关闭摄像头
```

this.selectComponent('#tcccSdk').disableCamera();

enableCamera — 开启摄像头



chis.selectComponent('#tcccSdk').enableCamera();

会话通用 API

setSoundMode — 设置声音输出方式

参数		类型	必填	备注
soundMode	'ear' 'speaker'	string	是	speaker:扬声器,ear:听筒

muteAudio — 静音

this.selectComponent('#tcccSdk').muteAudio();

unmnuteAudio — 取消静音

this.selectComponent('#tcccSdk').unmnuteAudio();

通用事件

sessionStart — 会话开始

<tccc-wx-sdk bind:sessionstart="handleStart"></tccc-wx-sdk>
wx.showToast({
icon: 'success',
title: ' 通话开始 '

accepted — 对方接听





} }

sessionEnded — 会话结束

```
<tccc-wx-sdk bind:sessionEnded="handleSessionEnded"/>
handleSessionEnded({ closeBy }) {
    if (closeBy === 'admin') {
        wx.showToast({
            icon: 'none', // 自选合适的图标
            title: '系统原因挂断'
        })
    } else if(cloeBy === 'seat') {
        wx.showToast({
            icon: 'none', // 自选合适的图标
            title: '通话结束,对方已挂断'
        })
    } else {
        // 挂断成功
    }
}
```

error 一 通话错误

```
<tccc-wx-sdk bind:error="handleError"/>
handleError(error) {
    wx.showToast({
        icon: 'error',
        title: error.message || '发生错误,请重试'
    })
}
```



用户端 SDK 下载和发布日志(App)

最近更新时间: 2025-03-12 18:09:45

SDK 下载

腾讯云联络中心TCCC SDK 提供方为深圳市腾讯计算机系统有限公司,其个人信息处理规则见 SDK 个人信息保护 规则,以下为 SDK 当前可下载版本,合规使用说明见 SDK 合规使用指南 。

平台/引擎	版本	更新时 间	SDK / Demo 源码	接入文档	主要功能说明
用户端 iOS	2.11. 23	2023- 11-23	● SDK 下 载 ● Demo 下载	快速集成 SDK	支持用户端向员工发起音视频通 话。
用户端 Android	2.11. 23	2023- 11-23	● SDK 下 载 ● Demo 下载	快速集成 SDK	支持用户端向员工发起音视频通 话。

发布日志

Version 2.11.23 (2023-11-23)

- 新增支持视频客服。
- 新增部分功能以及接口。

Version 2.10.3 (2023-10-30)

- 增加关键日志上报以及耗时统计。
- 升级了关联的基础库。
- 修复了部分问题。

Version 2.0.03 (2023-06-30)

• 产品发布,支持音频客服。



视频会话 CDR 推送

最近更新时间: 2024-01-18 16:44:31

URL: https://{custom_url}?action=videoCdr&version=1

METHOD: POST

Content-Type: application/json;charset=utf8

REQUEST:

参数	类型	说明
SdkAppId	数值(长整型)	联络中心实例 ID
SessionId	字符串	会话 ID
StartTimestamp	数值(长整型)	整个会话开始时间戳(UNIX 秒级时间戳)
AcceptTimestamp	数值(长整型)	会话首次应答时间戳(UNIX 秒级时间戳)
EndedTimestamp	数值(长整型)	会话整体结束时间戳(UNIX 秒级时间戳)
QueuedSkillGroupI d	数值	会话进入排队技能组 ID
QueuedSkillGroup Name	字符串	会话进入排队的技能组名称
Duration	数值	会话整体服务时间,单位秒 EndedTimestamp- AcceptTimestamp
EndStatusString	字符串	结束状态,枚举值见下表
Userld	字符串	用户 ID
UserNickName	字符串	用户昵称
SeatUser	对象	客服信息,格式见下
UserRemark	字符串	用户备注
ClientData	字符串	客户端关联数据

EndStatusString 枚举值

参数	说明
ok	正常结束



seatGiveUp	座席未接
seatForward	座席转接
ringingGiveUp	会话振铃期间用户放弃
noSeatOnline	无座席在线
notWorkTime	非工作时间
error	系统错误

SeatUser 数据格式

参数	类型	说明
Mail	字符串	座席邮箱
Name	字符串	座席名称
Nick	字符串	座席昵称
Phone	字符串	座席电话号码
UserId	字符串	用户 ID
StaffNumber	字符串	座席工号
SkillGroupNameList	字符串数组	座席所属技能组列表

RESPONSE:

参数	类型	说明
ErrMsg	字符串	错误说明
ErrCode	数值	错误码

数据样例:

"SdkAppId":1400xx214,
"SessionId":"e97be0ab-1ef6-4ad2-a8c4-2b2bbfb18e55",
"StartTimestamp":1608130636,
"AcceptTimestamp":0,
"EndedTimestamp":1608130651,
"Duration":15,

"EndStatusString":"error", "UserId":"oZB6q5c-tuMEz-ZiHOAkhW59a4AI", "UserNickName":"**小君君**", "SeatUser":{ "Mail":"lululing@tencent.com", "Name":"lululing", "Nick":"璐璐", "Phone":"", "UserId":"lululing@tencent.com", "StaffNumber":"007", "SkillGroupNameList":["lulu"] }, "ClieptData":"dChpcyBpcyBbIGV/VW1wbCVh767bc2PmVY