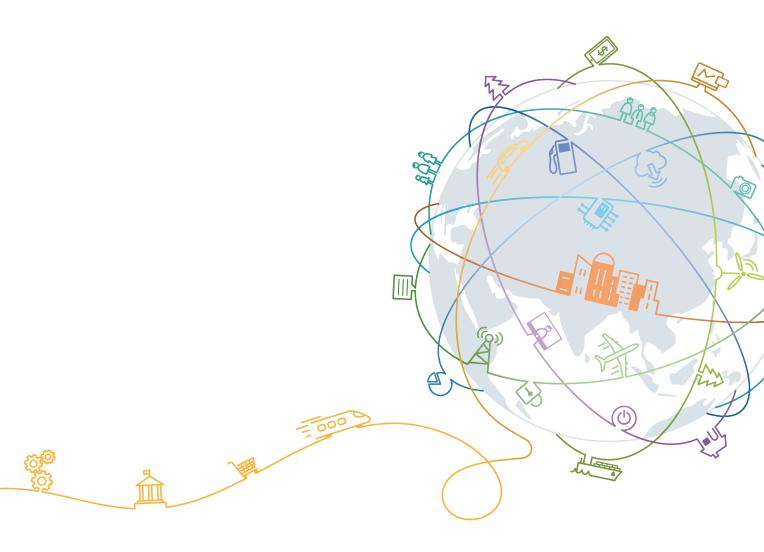
EC SDK Demo 19.1.10 指南(Android)

文档版本 03

发布日期 2019-07-30





版权所有 © 华为技术有限公司 2019。 保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。 本文档提及的其他所有商标或注册商标,由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束,本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定,华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用指导,本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编: 518129

网址: http://e.huawei.com

目 录

1 简介	
- 1921 2 快速体验	
2.1 业务帐号申请	
2.2 开发环境准备	2
2.3 SDK 导入	
2.4 调试运行 Demo	3
3 Demo 代码框架	4
3.1 文件夹结构	
3.2 逻辑框架	ϵ
4 Demo 功能列表	
5 如何裁剪	10
5.1 裁剪说明	
5.2 示例一: 裁剪 "IPT 业务"	
5.3 示例二: 裁剪"企业通讯录"	
5.4 示例三: 裁剪"CTD业务"	24
6 FAQ	31
7 修订记录	32

 $oldsymbol{1}$ $_{lpha _{ lpha } \wedge }$

Android SDK 提供了一套完整的接口集合,开发者可以通过调用 Android SDK (以下简称SDK) 开放的API,在自己的应用程序中快速集成音视频呼叫,音视频会议,桌面共享等能力。

2 快速体验

2.1 业务帐号申请

使用华为 eSDK 公有云环境

开发前,请申请华为云账号。申请方法请参见开发流程。

2.2 开发环境准备

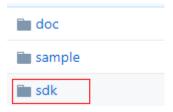
开发工具

在开发前,请准备好以下开发工具。

- 操作系统: Windows 7专业版。
- Android Studio: Android Studio3.0及以上版本。

SDK 软件包

● 访问https://github.com/Huawei/CloudEC_Client_API_Demo_Android, 获取sdk文件夹。



Demo 源码

● 访问https://github.com/Huawei/CloudEC_Client_API_Demo_Android, 获取 sample文件夹。

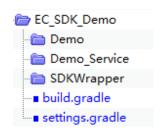


2.3 SDK 导入

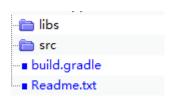
步骤1 将下载的eSDK API Android SDK软件包解压缩到磁盘:



步骤2 将下载的Demo源代码解压后拷贝至工作目录:



步骤3 将 "步骤1"解压后的软件包拷贝至 "SDKWrapper" 文件夹:



----结束

2.4 调试运行 Demo

步骤1 打开Android Studio, 导入2.3 章节中处理好的Demo工程。

步骤2 根据Android Studio的Gradle提示,以及本地的Gradle和Android SDK版本,修改工程的build.gradle配置。

□ 说明

主要修改为gradle版本号、compileSdkVersion、buildToolsVersion。

步骤3 Build。

□□说明

一般情况下,构建过程无错误,若存在报错,请根据Android studio提示进行修正。

步骤4 连接Android调试真机,安装编译好的EC_SDK_Demo.apk。

□ 说明

由于Android Studio3.xxx和Android Studio2.xxx编译有差别,会导致3.xxx编译出的apk在运行时报 NoClassDefFoundError错误。可参考网上multidex如何配置来解决。

步骤5 完成服务器地址、端口设置,输入登录帐号和密码,完成登录。

----结束

3 Demo 代码框架

3.1 文件夹结构

图 3-1 文件夹结构

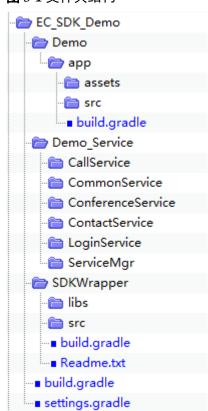
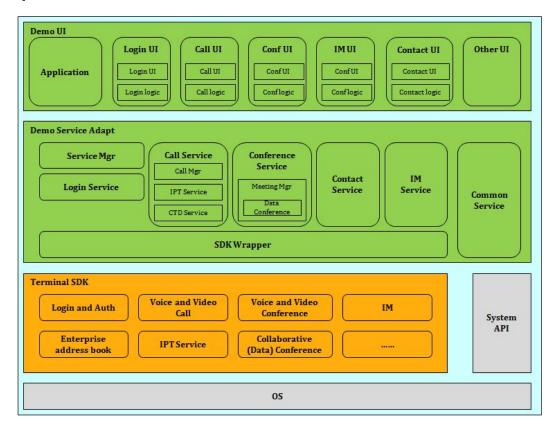


表 3-1 文件夹说明

文件夹	说明
Demo	UI 工程库文件夹,包含Demo的UI层业务逻辑、 用户界面呈现和资源文件(图片、音频和文字 等)。
Demo_Service\CallService	"音视频呼叫服务"工程库文件夹,包含基于 SDK实现音视频呼叫、IPT业务和CTD的业务代 码。
Demo_Service\CommonService	"公共服务"工程库文件夹,包含其他业务依赖的的公共功能实现代码,如字符串处理、 Crash输出、日志、广播消息等。
Demo_Service\ConferenceService	"音视频会议服务"工程库文件夹,包含基于 SDK实现音视频会议(包含数据会议)的业务代 码。
Demo_Service\ContactService	"联系人服务"工程库文件夹,包含基于SDK 实现企业联系人(地址本)的业务代码
Demo_Service\LoginService	"登录服务"工程库文件夹,包含基于SDK实现Demo所有业务逻辑总体登录入口的实现代码。
Demo_Service\ServiceMgr	"服务管理"工程库文件夹,包含Demo对具体业务的配置(是否支持某业务)、初始化和去初始化等服务管理的实现代码。
SDKWrapper\libs	SDK存放路径。

3.2 逻辑框架



$oldsymbol{4}$ Demo 功能列表

功能项	功能子项	是否开放	Demo是否支持
统一鉴权与登录	普通登录	Y	Y
	第三方鉴权登录	Y	N
	注销	Y	Y
音频呼叫	发起音频呼叫	Y	Y
	挂断/取消/拒绝呼叫	Y	Y
	接听音频呼叫	Y	Y
	设置(或取消)麦克风闭音	Y	Y
	发送DTMF信号	Y	Y
视频呼叫	发起视频呼叫	Y	Y
	挂断/取消/拒绝呼叫	Y	Y
	接听视频呼叫	Y	Y
	发起音频转视频	Y	Y
	发起视频转音频	Y	Y
	同意或拒绝音频转视频	Y	Y
	打开本地视频预览	Y	N
	关闭本地视频预览	Y	N
	开启或关闭本地摄相头	Y	Y
设备管理	切换麦克风	Y	N
	切换扬声器	Y	Y
	切换摄相头	Y	Y
	设置通话音量	Y	N

保持&恢复	保持和恢复音频通话	Y	Y
	保持和恢复视频通话	Y	Y
偏转	呼叫偏转	Y	Y
转移	呼叫盲转 Y		Y
登记IPT业务	登记或取消登记呼叫等待	Y	Y
	登记或取消登记DND	Y	Y
	登记或取消登记前转	Y	Y
CTD	发起CTD通话	Y	Y
会议管理	预约会议	Y	Y
	查询会议列表	Y	Y
	查询会议详情	Y	Y
基础会控	创建即时会议	Y	Y
	获取会议信息	Y	Y
	邀请与会者	Y	Y
	删除与会者	Y	Y
	挂断与会者	Y	Y
	重播与会者	Y	Y
	结束会议	Y	Y
	离开会议	Y	Y
	闭音会场	Y	Y
	闭音与会者	Y	Y
	锁定会议	Y	Y
	释放主席	Y	Y
	申请主席	Y	Y
	延长会议	N	N
	设置或取消举手	Y	Y
	显示发言人	Y	Y
	会议列表一键入会	Y	Y
	会议接入码入会	Y	Y
	IVR入会	Y	Y
渐进式会议	两转三会议	Y	N

	升级会议	Y	Y
基本控制	加入数据会议	Y	Y
	离开数据会议	Y	Y
	结束数据会议	Y	Y
屏幕共享	观看桌面共享	Y	Y
程序共享	观看应用程序共享	Y	Y
文档	观看文档共享	Y	Y
白板	观看白板共享	Y	Y
标注	观看标注	Y	Y
聊天	发送公共即时消息	Y	Y
网络地址本	查询企业联系人	Y	Y
	查询部门	Y	N
	获取和设置个人头像	Y	Y

5 如何裁剪

5.1 裁剪说明

EC SDK Demo(Android)基于功能特性进行模块划分,可以根据开发者的实际需要,在修改少量代码的情况少,对不需要的功能进行裁剪。

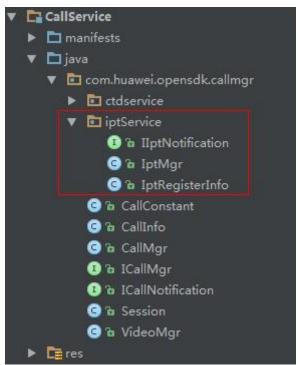
5.2 示例一: 裁剪 "IPT 业务"

□说明

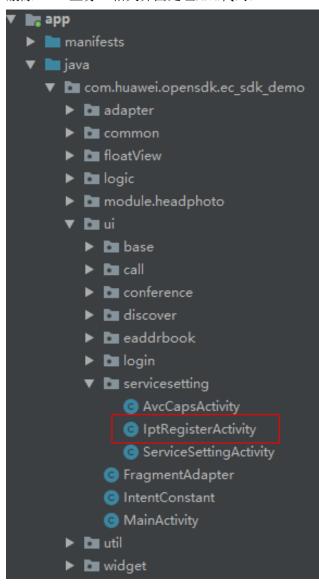
无特殊说明, 文中红框 内容均表示需删除内容。

步骤1 删除无需使用的文件和文件目录。

1. 删除"IPT业务"相关业务层处理代码:



2. 删除"IPT业务"相关界面处理Java代码:



∭说明

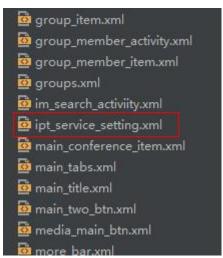
- IptRegisterActivity文件需全部删除。
- ServiceSettingActivity文件中的case值和case里面的内容需要删除。
- 3. 删除"IPT业务"相关引用代码:

□说明

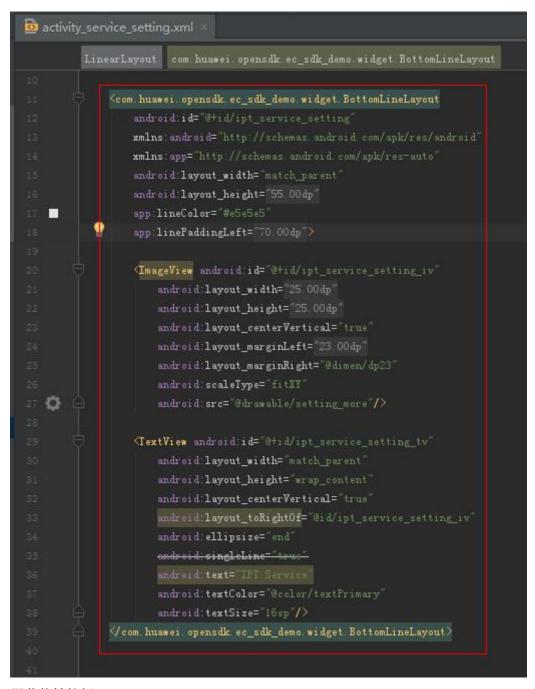
- ServiceMgr工程下,ServiceNotify文件里面的ipt界面通知需要删除。
- 4. 删除"IPT业务"相关界面*layout.xml文件。

□说明

以下只是示例,包含了主要相关layout.xml文件,并非所有,开发者可以根据需要进一步删除。



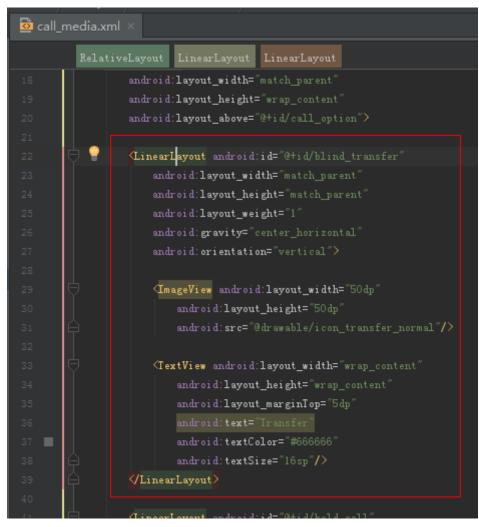
5. 屏蔽IPT注册界面入口。



6. 屏蔽偏转按钮

```
all_media.xml
         RelativeLayout LinearLayout FrameLayout
                      android:layout_weight="1"
                      android:visibility="gone">
                      ImageView android:id="@+id/video_accept_call_btn"
                          android:layout_width="73dp"
                          android:layout_height="73dp"
                          android:scaleType="fitXY"
                          android:layout_gravity="center"
                          android: src="@drawable/video_accept"/>
                  ⟨/FrameLayout⟩
                  (FrameLayout android:id="@+id/divert_call_area"
                      android:layout_width="0dp"
                      android:layout_height="wrap_content"
                      android:layout_weight="1"
                      android:visibility="gone">
                      ImageView android:id="@+id/divert_call_btn"
                          android:layout_width="73dp"
                          android:layout_height="73dp"
                          android:scaleType="fitXY"
                          android:layout_gravity="center"
                          android:background="@drawable/icon_transfer_normal" />
                  ⟨FrameLayout⟩
              ⟨LinearLayout⟩
```

7. 屏蔽盲转按钮



8. 删除偏转,盲转界面按钮

```
BaseMediaActivity

protected ImageView mRejectBtn;
protected FrameLayout mAudioAcceptCallArea;
protected FrameLayout mVideoAcceptCallArea;
protected FrameLayout mDivertCallArea;
private Timer mDismissDialogTimer;
private static final int CANCEL_TIME = 25000;

protected LinearLayout mPlateButton;
protected LinearLayout mHoldCallButton;
protected LinearLayout mMuteArea;
protected LinearLayout mPlateArea;
```

----结束

5.3 示例二: 裁剪"企业通讯录"

∭说明

- 1、无特殊说明, 文中红框 内容均表示需删除内容。
- 2、因"即时消息"依赖"企业通讯录"来实现头像更新上传功能,所以,裁剪"企业通讯录"后,头像更新上传功能不可用。

步骤1 修改Gradle中的相关配置。

1. 删除"settings.gradle"中的依赖: ':Demo_Service:ContactService';

```
'Demo_Service:ContactService'
```

2. 删除相关"build.gradle"中的依赖:

build.gradle (Module: app)

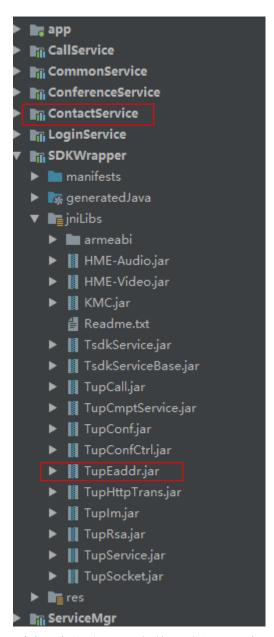
```
dependencies {
    compile fileTree(include: ['*.jar'], dir: 'libs')
    compile com android.support:support-v4:25.3.1'
    compile com android.support:support-v13:25.3.1'
    compile com android.support:appcompat-v7:25.3.1'
    compile project(':Demo_Service:CallService')
    compile project(':Demo_Service:LoginService')
    compile project(':Demo_Service:ConferenceService')
    compile project(':Demo_Service:ServiceMgr')
    compile project(path: ':Demo_Service:ContactService')
```

build.gradle (Module: ServiceMgr)

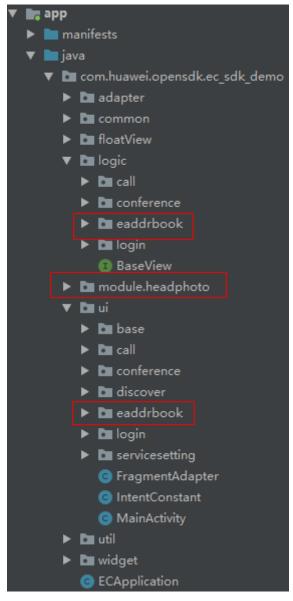
```
dependencies {
    compile fileTree(include: ['*.jar'], dir: 'libs')
    compile project(path: ':Demo_Service:CallService')
    compile project(path: ':Demo_Service:LoginService')
    compile project(path: ':Demo_Service:ConferenceService')
    compile project(path: ':Demo_Service:ContactService')
}
```

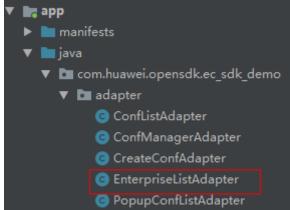
步骤2 删除无需使用的工程、包、文件和文件目录。

1. 删除"ContactServer"工程库和"TupEaddr.jar"包:



2. 删除"企业通讯录"相关界面处理Java代码:



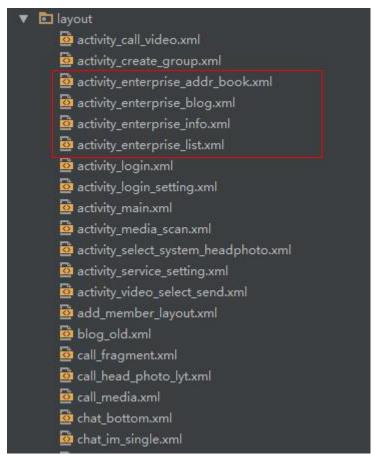


∭说明

- ServiceMgr工程下, ServiceNotify文件里面的eaddr界面通知需要删除。
- 3. 删除"企业通讯录"相关界面*layout.xml文件。

□说明

以下只是示例,包含了主要相关layout.xml文件,并非所有,开发者可以根据需要进一步删除。



步骤3 删除工程库中对以上步骤中内容的引用。

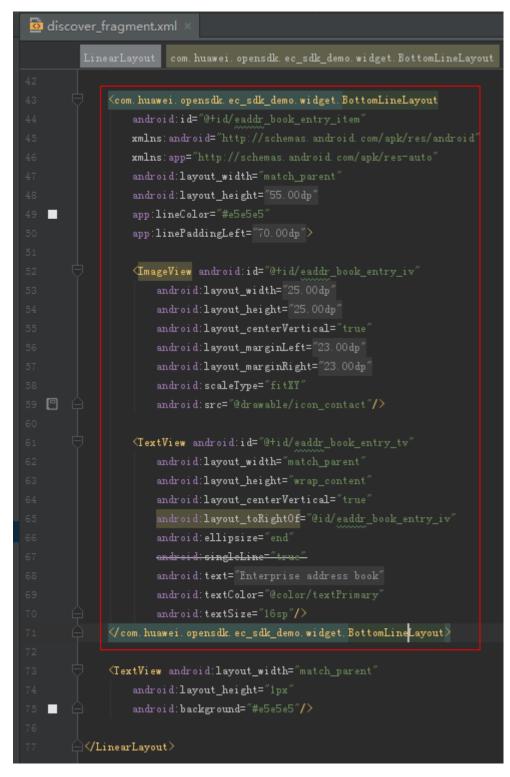
如:



□ 说明

开发者可以通过"build"的方式,找到相关引用,再进行屏蔽。

步骤4 屏蔽主界面入口。



步骤5 屏蔽其他无需使用的界面入口。

步骤6 删除其他"企业通讯录"相关资源文件。

----结束

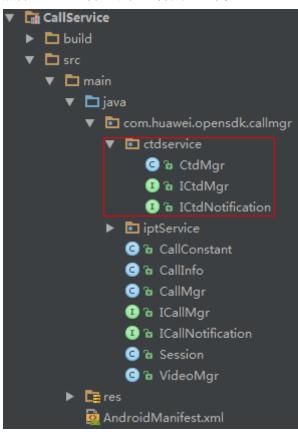
5.4 示例三: 裁剪 "CTD 业务"

□说明

无特殊说明, 文中红框 内容均表示需删除内容。

步骤1 删除无需使用的文件和文件目录。

1. 删除"CTD业务"相关业务层处理代码:



- 2. 删除"CTD业务"相关引用代码:
 - 删除LoginActivity引用

```
Log i( tag: "SDEDemo", msg: "onCreate: PUSH Process.");

return;

ServiceIlgr. getServiceMgr(). startService( context: this, appPath. mIdoProtocol);

Log. i( tag: "SDEDemo", msg: "onCreate: MAIN Process.");

Log. i( tag: "SDEDemo", msg: "onCreate: MAIN Process.");

Log. i( tag: "SDEDemo", msg: "onCreate: MAIN Process.");

CallMgr. getInstance(). regLoginEventNotification(LoginFunc. getInstance());

CallMgr. getInstance(). regCallServiceNotification(CallFunc. getInstance());

CtdMgr. getInstance(). regCallServiceNotification(ConfFunc. getInstance());

RestingMgr. getInstance(). regConfServiceNotification(ConfFunc. getInstance());

EnterpriseAddressBookMgr. getInstance(). registerNotification(EnterpriseAddrBookFunc. getInstance());
```

- 删除ServiceNotify引用

- 删除CallFunc引用

```
CallFunc onStartCtdCallResult()

//mPlayHandle = -1;

mMuteStatus = false;

public boolean isMuteStatus() { return mMuteStatus; }

public void setMuteStatus (boolean mMuteStatus) { this.mMuteStatus = mMuteStatus = mMuteStatus | this.mMuteStatus | this.mmut
```

```
CallFunc mHandler new Handler

import com. huawei. opensdk. ec_sdk_demo. util. ActivityUtil:
import com. huawei. opensdk. commonservice. common. LocContext:
import com. huawei. opensdk. ec_sdk_demo. util. FileUtil:

import java io. File;

public class CallFunc implements ICallNotification, ICtdNotification

f

private static final int UPGRADE_FAILED = 100;
private static final int CID_FAILED = 101;
private static final int CID_SUCCESS = 102;
private static final String RINGING_FILE = "ringing wav";
private static final String RING_BACK_FILE = "ring_back.wav";

private boolean mMuteStatus;
private String mFilePath;

urivate static CallFunc mInstance = new CallFunc():
```

- 删除CallFragment引用

□说明

此处的else也需要删除。

3. 屏蔽主界面入口:



4. 删除界面逻辑:

```
CallFragment

public class CallFragment extends AbsFragment implements View. OnClickListener

frivate ImageView mBtnVideo;
private String mSipNumber = "";
private ImageView mBtnVideo;
private ImageView mBtnDala;

private ImageView mCtdButton,
private ImageView mCtdButton,
private Boolean isChecked = false;
private static String number;
private int mCallID = 0;

public static String getNumber() { return number; }

public static void setNumber(String number) { CallFragment. number = number; }

public static void setNumber(String number) { CallFragment. number = number; }
```

```
CallFragment.java
             public void onDataLoad() { super.onDataLoad(); }
86 🧿
             public void onViewCreated()
                 super. onViewCreated();
                 mBtnCall = (ImageView) mView.findViewById(R.id.call_audio_btn);
                 mBtnVideo = (ImageView) mView.findViewById(R.id.call_video_btn);
                 mHideDial = (ImageView) mView.findViewById(R.id.hide_dial_btn);
                 mDialLayout = (LinearLayout) mView.findViewById(R.id.dial_plate_area);
                  mShowArea = (ImageView) mView.findViewById(R.id.show_dial_btn);
                 mDeleteNumberArea = (ImageView) mView.findViewById(R.id.delete_panel_btn);
                 mCtdButton = (ImageView) mView.findViewById(R.id.select_ctd_btn);
                 mDialPlateControl.showDialPlate();
                 mBtnCall.setOnClickListener(this);
                 mBtnVideo.setOnClickListener(this);
                 mHideDial.setOnClickListener(this);
                 mCtdButton.setOnClickListener(this);
                 mSipNumber = LoginMgr.getInstance().getTerminal();
```

----结束

6 faq

暂无。

7修订记录

发布日期	文档版本	修订说明
2019-07-30	3.0	配套19.1.10版本文档刷新发布。
2019-06-14	2.0	配套19.1.1版本文档刷新发布。
2018-06-06	1.0	配套6.1.0版本(暂仅支持融合会议组网), 文档首次发布。