腾讯云微金小云客服

SDK

产品文档





【版权声明】

©2013-2017 腾讯云版权所有

本文档著作权归腾讯云单独所有,未经腾讯云事先书面许可,任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】



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。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况,部分产品、服务的内容可能有所调整 。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模式的承诺或保证。



文档目录

文	[档声明	2
SI	DK	4
	SDK接入指南(Android)	4
	SDK接入指南(iOS)	8



SDK

SDK接入指南(Android)

管理人员登录 微金小云客服管理系统 ,在【系统管理】>【App 接入】单击 Android SDK

接入说明,接入指引如下:

2	微众银行							
	服务监控	业务参数 客服管理	用户管理	微信接入	APP接入			
	统计分析	接入信息						
	会话管理	APP ID						
	系统管理	APP KEY 						
	知识库	有效期 2016-09-06 3	至 2016-09-07					
		SDK信息						
		IOS SDK下载 0.1 0.1 0.1						
		IOS SDK接入说明 >		Android SDK接入说明)	>			
-								
	■ 云客版sdk-sit-v1zip ^ ■ WBCouldServicezip ^							

SDK 接入 (Android)

接入前请先确保后台接入正常, 接入配置为 Android Studio。



云客服 SDK (WeCcsSdk) 最低支持到 Android API 14:Android 4.0(ICS),请在构建项目时注意。

WeCcsSdk 将以 AAR 文件的形式提供。

需要添加下面文档中所示的依赖(将提供的两个 aar 文件加入到 App 工程的'libs'文件夹下面,并且在 build.gradle 中添加下面的配置:

```
android{
 //...
 repositories {
 flatDir {
 dirs 'libs' //this way we can find the .aar file in libs folder
 }
 }
}
//添加依赖
dependencies {
 //0. appcompat-v7
 compile 'com.android.support:appcompat-v7:23.1.1'
 //1. 云客服SDK
 compile(name: 'webank-sdk-ccs-openPro-release.aar ', ext: 'aar')
 //2. WeBank图片选择器
 compile(name: 'webank_image_picker-release', ext: 'aar')
 // 3. 依赖的第三方jar包
 compile 'com.google.code.gson:gson:2.3.1' //网络请求json解析
 compile 'com.squareup.okhttp:okhttp:2.4.0' //网络请求
 compile 'com.github.bumptech.glide:glide:3.7.0' //for 图片选择器
}
```

混淆配置

将 SDK 中的 webank-ccs-proguard-rules.pro 和 x5_proguard.cfg.pro 拷贝到主工程根目录下 , 然后通过"-include webank-ccs-proguard-rules.pro" 加入到您的混淆文件中。

接口调用



SDK 代码调用的入口为: com.webank.mbank.ccs.WeCcsSdk 这个类。

```
public class WeCcsSdk {
    /**
    * 在Application.onCreate()的时候初始化
    */
    public void init(Context app);
    ...
    /**
    * 需要打开云客服界面时调用此接口,通过Bundle进行参数传递。
    *支持的参数见后面的说明
    */
    public void launch(Activity ctx, Bundle data) {
    // ...
    }
}
```

WeCcsSdk.launch()的第二个参数用来传递数据,可以将参数打包到 data(Bundle)中,必须传递的参数包括:

//这些都是WeCcsSdk的成员常量,作为传递数据的key KEY_USER_ID = "user_id"; //user id KEY_SIGNATURE = "signature"; //签名信息 KEY_NONCE="nonce"; //32位随机字符串 KEY_APP_ID = "app_id"; //APP_ID

其他选填的参数被封装在 com.webank.mbank.ccs.model.User 对象中(它是一个 Parcelable 对象),包括 nickName(昵称),cardNo(证件号码),userName(用户名),sex(性别:男1女 2),phoneNum(手机号码)等,可以通过 KEY_USER_INFO 这个名称放到 bundle 中,这些参数可以不传入。

接入实例



自定义Application.onCreate()中: WeCcsSdk.getInstance().init(this); # 在MainActivity中点击某个按钮的代码逻辑: Bundle bundle = new Bundle(); bundle.putString(WeCcsSdk.KEY_NONCE, "随机码xxx"); bundle.putString(WeCcsSdk.KEY_USER_ID, "UserIdxxx"); bundle.putString(WeCcsSdk.KEY_SIGNATURE, "签名信息xxx"); bundle.putString(WeCcsSdk.KEY_APP_ID,"app_id"); WeCcsSdk.getInstance().launch(MainActivity.this, bundle);



SDK接入指南(iOS)

管理人员登录 微金小云客服管理系统 , 在【系统管理】>【App 接入】单击 iOS SDK

接入说明,接入指引如下:

ĉ	微众银行								
	服务监控	业务参数	客服管理	用户管理	微信接入	APP接入			
	统计分析	接入信息							
	会话管理	APP ID							
	系统管理	APP KEY							
		状态 	有效 2016-09-06 至 20	016-09-07					
		SDK信 息							
			IOS SDK下載 0.1 Android SDK下载 0.1						
		IOS SDK接入:	尻明 >		Android SDK接入记	99 >			
=	■ 云客服sdk-sit-v1zip へ ■ WBCouldServicezip へ								

SDK 接入(iOS)

接入前请先确保后台接入正常, 接入配置为 iOS。

WeBankService SDK 最低支持到 iOS7.0,请在构建项目的时候注意。



引用资源文件 WBCloudCustomerService.bundle 和 WBCloudCustomerService.framework 到项目, SDK

Required 依赖 libspeex.a, libogg.a; Optional 依赖

Photos.framework, libc++.tbd.需要在【BuildPhases】>【Link Binary With Libraries】中添加。

	General	Capabilities	Resource Tags	Info	Build Settings	Build Phases	Build Rules				
+								🕞 Filter			
▼ Target Dependencies (0 items)											
					Add target depend	dencies here					
	+ -										
Compile Sources (41 items)										×	
Link Binary With Libraries (5 items)										×	
	Name								Status		
	🚔 Photos.	framework							Optional 🛟		
	libc++.t	bd							Optional 🛟		
	📑 libogg.a	I							Required 🗘		
	📑 libspeex	c.a							Required 🗘		
	🚔 WBClou	dCustomerService	.framework						Required 🗘		
	+ -				Drag to reorder f	rameworks					

SDK 需要使用相机,相册和录音权限,请在 info.plist 中添加 Privacy -Microphone Usage Description, Privacy -Camera Usage Description, Privacy -Photo Library Usage Description 设置语言,在工程中【project】>【info】>【Localization】>【language】中 add 一个简体中文 Chinese(Simplified)

调用 SDK 接口

SDK 的功能通过 WBService 这个类的方法进行调用,详细接口说明如下:

@class WBCloudCustomerService;

@protocol WBCloudCustomServiceDelegate < NSObject >

@required

/**

拉起需要在哪一个页面拉起云客服页面

*/

-(nonnull UIViewController *)getViewController;

@optional

-(void)serviceWillpopViewController:(nonnull WBCloudCustomerService *)service;

-(void)serviceDidpopViewController:(nonnull WBCloudCustomerService *)service;



/**

需要传入的是用户信息具体的传递方法直接使用

@"id_no": @"421112121100001201" 证件号,如果没有请传@""
@"user_name": @"张三" 姓名,如果没有请传@""
@"phone_num": @"18800000000" 电话号, 如果没有请传@""
@"sex":@"1" 性别: 男1 女2,如果没有传递@""
@"nick_name":@"yournickname" 昵称
*/

*/

@interface WBCloudCustomerUserInfo : NSObject

@property (nonatomic, copy) NSString *id_no;//证件号,如果没有请传@""
@property (nonatomic, copy) NSString *user_name;//姓名,如果没有请传@""
@property (nonatomic, copy) NSString *phone_num;//电话号, 如果没有请传@""
@property (nonatomic, copy) NSString *nick_name;//昵称, 如果没有请传@""
@property (nonatomic, copy) NSString *sex;//性别: 男1 女2,如果没有传递@""

@end

@interface WBCloudCustomerService : NSObject

/**

* WBCloudCustomerService 的delegate

*/

@property (nonatomic, weak) id<WBCloudCustomServiceDelegate> delegate;

/**

*

WBCloudCustomerService全局单例,请不要使用init方法创建WBCloudCustomerService对象,使用全局单例 */

+(instancetype)sharedInstance;

/**

* 云客服sdk拉起页面的接口,需要传入以下参数.在调用该方法以前,需要首先通过[WBCloudCustomerService



sharedInstance].delegate = xxx;

*

- * @param appid 合作方使用的APPID
- * @param nonce 每次请求使用的唯一的nonce信息
- * @param sign 将nonce签名以后的信息注意,nonce和sign两个信息都必须由合作方后台传递给合作方iOS APP.
- * @param version 版本信息,默认使用@"1.0.0"
- * @param userId 每个用户唯一标识
- * @param userInfo 每个客户的身份姓名性别等重要信息

*/

 -(void)startCustomServcie:(NSString *)appid nonce:(NSString *)nonce sign:(NSString *)sign version:(NSString *)version userId:(NSString *)userId userInfo:(WBCloudCustomerUserInfo *)userInfo;
 @end

具体调用 SDK 的步骤:

1. App 在希望调用 SDK 的时候,在该 UIViewController 中实现 delegate 方法。

```
#pragma mark - WBServiceDelegate
-(UIViewController *)getViewController{
return self;
```

}

2. 设置 delegate、UI 风格、当前 SDK 运行的模式,测试的时候请使用 WBSDKEnvDebug,如果生产上线请设置 WBSDKEnvRelease。

[[WBCloudCustomerService sharedInstance] set WeBankSDKEnv:WBCloudCustomServiceEnvDebug]; [WBCloudCustomerService sharedInstance].delegate = self;

3. 创建客户信息类的实例对象,具体含义见 SDK 注释。



创建一个WBCloudCustomerUserInfo对象并设置相应的属性,等待传入sdk

4. 最后将 SDK 所需要的参数通过 StartService 方法传递给单例, 拉起 SDK。

[[WBCloudCustomerService sharedInstance] startCustomServcie:appid nonce:self.nonce sign:self.sign version:@"1.0.0" userId:self.userId.text userInfo:[WBCloudCustomerUserInfo new]];

- 5. 在 SDK 服务完成以后,会调用 delegate,并从当前页面退出,具体的回调方法是:
 - -(void)serviceWillpopViewController:(nonnull WBService *)service;
 - -(void)serviceDidpopViewController:(nonnull WBService *)service;