

畅言 android 平台 sdk 文档

北京搜狐新媒体信息技术有限公司

版本：CySDK V2. 2. 0

修订记录

时间	文档版本	修订人	备注
2014/01/07	2. 1. 0		新增评论框浮层，评论列表页，评论发表页和社交平台分享功能；原有 api 接口格式有变动
2014/03/20	2. 1. 1		增加评分功能，修改 sdk 初始化方式，防止泄露 appsecret
2014/04/11	2. 1. 2		优化单点登录功能流程
2014/08/11	2. 2. 0		优化评论框浮层、评论列表页、评论发表页和登陆页，新增分频道获取评论功能，优化顶踩接口
2014/12/16	2. 2. 1		1. 完善回复列表，增加消息推送功能 2. 增加“我做出的评论” 接口
2014/12/24	2. 2. 2		1. 增加是否启用消息推送功能的开关

2014/12/30	2. 2. 3		1. 修复评论列表重复显示的 bug 2. 新增 api 文档的详细说明 3. 增加消息推送中的注意事项 4. 修复列表中图片下载失败时有 toast 显示的 bug
2015/02/08	2. 2. 4		1. 修复图片缓存导致评论列表闪退的 bug。

目录

畅言 android 平台 sdk 文档.....	1
目录	2
一、 概述.....	4
名词解释.....	4
二、运行 demo 程序	4
三、快速接入.....	4
1.导入依赖文件.....	4
2.添加应用权限.....	5
3.添加畅言配置.....	5
4.初始化畅言 SDK	5
4.1.config——UI 类	6
4.2.config——Login 类	11
4.3.config——CommentSettings 类.....	12
5.调用 SDK 接口插入评论框浮层。	12
5.1 activity 嵌入	12
5.2 fragment 嵌入.....	12
6. SDK 中单点登录功能的使用	13
7. SDK 中消息推送的使用	14
四、api 调用示例及说明.....	15
1.畅言数据接口 api.....	15
1.1 不登录可访问的 api 接口	16

1.2 需要登录的 api.....	23
2.附带 UI 界面的 api 接口	26
2.1 评论列表界面.....	26
2.2 评论发表界面.....	27
2.3 用户评论中心界面(包括发出的评论和接收到的评论)	27
3. 消息推送.....	27
配置见 三.7.....	27
设置 notification 的图标	27
如何自定义 处理接收到消息推送.....	28
如果使用自定义的评论编辑页面.....	28
注意事项.....	28
附录	28
关于代码混淆.....	29

一、概述

畅言SDK为开发者提供app内评论功能的嵌入，通过使用畅言SDK，开发者可在app内拥有评论功能和分享功能。

本文档将对使用 SDK 时所用的配置、接口进行说明，并提供简单的实例，帮助开发者方便的使用 SDK。

名词解释

cyan_app_id	畅言分配给第三方应用的 appId，用来鉴别身份。登录 changyan.sohu.com 后，点击设置->高级，查看。
cyan_app_key	畅言分配给第三方应用的 appkey，用来进行数据加密和签名校验。登录 changyan.sohu.com 后，点击设置->高级，查看。
accesstoken	表示用户身份的 token，用于畅言 api 的调用。
oauth_redirect	使用 oauth 方式登录畅言的回调页面地址，登录 changyan.sohu.com 后，点击设置->高级，进行修改。

二、运行 demo 程序

- 1.下载畅言 SDK 的 zip 包，解压。
- 2.将 CySDK 2.2.0 中的 cyan_sdk_test 项目导入到 eclipse 中。
- 4.右键点击 cyan_sdk_test，run as->android application 即可运行。

三、快速接入

1.导入依赖文件

将 CySDK 2.2.0 中 lib 文件夹下的所有 jar 包和 CySDK 中 bin 文件夹下的 CySDK.jar 复制到

工程的 libs 目录下。

2.添加应用权限

在 AndroidManifest.xml 中添加以下权限配置

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

3.添加畅言配置

在 AndroidManifest.xml 中添加如下配置项（可参考 CySDK 2.1.0 下的 AndroidManifest.xml）

```
<!-- 评论列表界面 -->
<activity android:name="com.sohu.cyan.android.sdk.activity.CyanCommentActivity"/>
<!-- 发表评论页面 -->
<activity android:name="com.sohu.cyan.android.sdk.activity.CyanPostCommentActivity"/>
<!-- 社会化账号登陆界面-->
<activity android:name="com.sohu.cyan.android.sdk.activity.OAuthActivity"></activity>
```

畅言的注册地址: <http://changyan.sohu.com/register>

4.初始化畅言 SDK

```
Config config = new Config();
config.ui.toolbar_bg = Color.WHITE;
config.comment.showScore = true;
config.comment.uploadFiles = true;
config.comment.anonymous_token = "z6e0xTejZBmqP-dQcAGN2lWmTZu8_yk9fW_w7oUvJP8";
config.login.loginActivityClass = AppLoginActivity.class;
CyanSdk.register("cy2uLn0jfoQS", "16bd0e2533092b96e3e55958ef19d08a" ,
"http://10.2.58.251:8081/login-success.html", config);
cyanSdk = CyanSdk.getInstance(this.getActivity());
```

高亮部分修改为畅言授权的 appid、appkey 以及后台设置的 callbackUri，登录 changyan.sohu.com, 点击 设置 -> 高级 查看 配置项，callbackUri 在 页面 <http://changyan.sohu.com/setting/common/further> 中设置，

调用API接口

回调地址

确定

当您需要使用畅言OAUTH2.0认证时，需填写回调地址，详情参见：[OAUTH认证文档](#)

如果之前没有设置过 `oauth` 的回调地址，可以将之设置为网站的域名。

下划线部分为自定义畅言显示选项，需要在 `config` 中进行设置，详细参数说明如下：

4.1.config——UI 类

1. 颜色设置

可设置评论框浮层的背景色、按钮色、文字颜色等，不修改则使用默认值。

`toolbar_bg` : `toolbar` 背景色

`toolbar_border` : `toolbar` 边框色

`toolbar_btn`: `toolbar` 按钮颜色

`before_clk` : 顶按钮点击之前颜色

`after_clk`: 顶按钮点击之后颜色

`list_title`: 热门评论、最新评论标题的文字和边框颜色

`login_bg`: 登陆界面背景颜色

`login_btn`: 登陆界面按钮颜色

`login_btn_text`: 登陆界面按钮文字颜色

`login_input` : 登陆界面昵称显示框背景颜色

`login_input_text`: 登陆界面昵称显示框文字颜色

`edit_cmt_bg`: 评论输入框背景颜色

替代机型。其他一些国家也减少了其打算购买的F-35战斗机数量。目前，已经建成了65架F-35战斗机，其中一架属于荷兰。美国国防部以承诺控制成本对此作出回应。提供的数据显示F-35战斗机设计和工厂的变化，可能会使其单位成本每架8500万美元的价格更加便宜。



图1 bottomtoolbar颜色参数指示图



图2 评论展示界面颜色参数指示图



图3 评论界面颜色参数指示图



图4 登陆界面颜色参数指示图

2. 风格设置

评论显示方式分为盖楼（floor）和缩进（indent）两种方式，通过 `style` 设置，默认值为“floor”，当选择indent风格时，需设置depth和sub_size的值，depth表示缩进显示的深度，sub_size 表示缩进显示的评论条数。

具体展示形式如下



图5 盖楼风格



图6 缩进风格(depth = 1, sub_size=20)

设置方法：修改 config

```
config.ui.style="indent";  
config.ui.depth = 1;  
config.ui.sub_size = 20;
```

4.2.config——Login 类

可设置登陆相关选项，不修改则使用默认值。

SSOLogin：是否允许单点登录

SSO_Assets_Icon：单点登陆图标的名称(需将图片文件放在 assets 文件夹中)，SSOLogin 为 true 时必须设置。

loginActivityClass：单点登录的 Activity，SSOLogin 为 true 时必须设置。

Custom_oauth_login：第三方登录是否自己获得授权，默认设置为 false，即授权给畅言。

QQ_btn_listener/SINA_btn_listener：(Custom_oauth_login 为 true 时，必须设置。)

用户自己绑定在 QQ/SINA 登录按钮上的监听事件。

QQ/SINA/SOHU：Custom_oauth_login 为 false 时，选取第三方登录方式。

4.3.config——CommentSettings 类

可设置评论相关选项，不修改则使用默认值。

showScore: 是否允许用户评分

uploadFiles: 是否允许上传附件

useFace: 是否使用表情功能

anonymous_token: 匿名登录的 access token, 为 null 时, 不允许匿名登录, 在后台设置->wap 设置中获取这个 token。

游客匿名token

IRTU3LghBcOymAaPaURsUIJNaGo0GdQw60H6ZSyC\

复制

游客昵称

22222

hotsize: 显示的热门评论数

latestsize: 显示的最新评论数

pagesize: 加载更多时显示的评论数

5.调用 SDK 接口插入评论框浮层。

目前 SDK 开发的评论框浮层支持嵌入到 `LinearLayout` 和 `RelativeLayout` 两种布局中, `LinearLayout` 的嵌入需要将布局中的其他 view 整体包裹在一个布局中, 并设置 `layout_weight` 属性为 1, `layout_height` 为 0。请参照 `cyan_sdk_test` 项目下的 `res/toolbar_demo.xml`, 是如何实现在 `LinearLayout` 中实现评论框浮层的嵌入的。`RelativeLayout` 的嵌入不需要做特殊处理。

5.1 activity 嵌入

如果在 activity 中直接嵌入评论框浮层, 那么可以直接调用 `addCommentToolbar(final Activity activity, final String topicSourceId, String topicTitle, String topicUrl)` 接口, `topicSourceId` 是文章在你自己系统的 id, `topicTitle` 是文章标题, `topicUrl` 是文章链接。

5.2 fragement 嵌入

`com.example.cyan_sdk_test.CyActivity.java` 中的 `FixToolBarSectionFragment` 类展示了如何在 `Fragment` 中嵌入评论框浮层。先要在 `onCreate` 方法中出事化畅言 SDK 实例, 畅言 SDK 采用单例模式, 调用畅言 SDK 的 api 之前需要先初始化, 然后在 `onCreateView` 中调用 `CyanSDK`

的 `addCommentToolbar` (`ViewGroup rootView`, `String topicSourceId`, `String topicTitle`, `String topicUrl`)，需要将 `fragment` 的 `rootView` 转为 `ViewGroup` 类型传入，`topicSourceId` 是文章在你自己系统的 id，`topicTitle` 是文章标题，`topicUrl` 是文章链接。

```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {
    View rootView = inflater.inflate(R.layout.toolbar_demo, container, false);
    WebView webview = (WebView) rootView.findViewById(R.id.webview);
    webview.loadUrl("http://mil.sohu.com/20131103/n389458829.shtml");
    cyanSdk.addCommentToolbar((ViewGroup) rootView, "20131125", "F-35战机",
        "http://mil.sohu.com/20131103/n389458829.shtml");
    return rootView;
}
```

6. SDK 中单点登录功能的使用

初始化 SDK 时需要设置单点登录相关配置项：

```
Config config = new Config();
// 是否在发表评论时允许使用单点登录
config.login.SSOLogin = true;
// 登录图标名称(需将图片文件放在assets文件夹中)
config.login.SSO_Assets_Icon = "ico31.png";
// 登录界面activity的class
config.loginActivityClass = AppLoginActivity.class;
CyanSdk.register("cy2uLn0jfoQS", "16bd0e2533092b96e3e55958ef19d08a",
    "http://10.2.58.251:8081/login-success.html", config);
```

AdnroidMainfest.xml 需要添加的配置项：

```
<!--登录界面 activity -->
<activity android:name="com.example.cyan_sdk_test.AppLoginActivity"></activity>
```

登录成功后需要调用的 sdk 接口，以实现畅言的同步登录：

```
AccountInfo account = new AccountInfo();
//应用自己的用户id
account.isv_refer_id = "10001";
account.nickname = "testuser";
account.img_url =
"http://assets.changyan.sohu.com/upload/asset/scs/images/pic/pic42_null.gif";
CyanSdk.getInstance(AppLoginActivity.this).setAccountInfo(account, callBack);
```

其中 `callBack` 接口是同步登陆畅言成功后需要做的回调操作。

畅言后台开启单点登录开关

在后台设置->PC 端设置->单点登录中，勾选“开启单点登录服务”，

单点登录

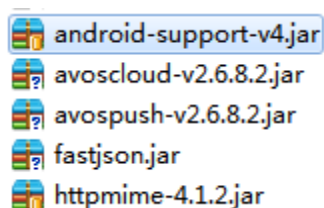
☐ 开启单点登录服务

注意，只有当下边的三个接口URL都设置正确时，单点登录才会生效
如何实现单点登录，详情参见：[单点登录文档](#)

以保证用户数据与站点的同步。

7.SDK 中消息推送的使用

引入 jar 包



AndroidManifest.xml 中配置 activity

```
<activity
    android:name="com.sohu.cyan.android.sdk.activity.RepliesActivity"
    android:launchMode="singleTask">
```

AndroidManifest.xml 中配置服务

```
<!-- 畅言消息推送 服务 -->
<service android:name="com.avos.avoscloud.PushService" />
<receiver
    android:name="com.sohu.cyan.android.sdk.push.CyanPushReceiver" >
    <intent-filter>
        <action
            android:name="android.intent.action.BOOT_COMPLETED" />
        <action
            android:name="android.intent.action.USER_PRESENT" />
        <action
            android:name="android.net.conn.CONNECTIVITY_CHANGE" />
        <action
            android:name="action.com.sohu.cyan.push.receive" />
    </intent-filter>
</receiver>
```

AndroidManifest.xml 中配置权限

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"
/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"
/>
<uses-permission android:name="android.permission.VIBRATE" />
```

AndroidManifest.xml 中<application>配置

```
android:name="com.sohu.cyan.android.sdk.push.CyanPushApplication"
```

或者

```
public class MyApplication extends CyanPushApplication {
    @Override
    public void onCreate() {
        super.onCreate();
    }
}
android:name="包名.MyApplication"
```

AndroidManifest.xml 中注册 RepliesActivity

```
<activity
android:name="com.sohu.cyan.android.sdk.activity.RepliesActivity"
android:launchMode="singleTask">
```

Java 代码中启用消息推送(需要写在 CyanSdk.register(...)之后)

```
PushManager.setUseService(Context, boolean);
```

四、api 调用示例及说明

1. 畅言数据接口 api

调用畅言 api 的数据接口需要先将 CySDK 2.1.0 目录中 libs 文件夹下的 fastjson-1.1.36.jar 和 httpmime-4.1.2.jar 复制到工程的 libs 目录下。

然后在项目的 AndroidManifest.xml 中加入以下配置信息:

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

调用api接口前需要初始化畅言sdk,sdk初始化配置详见:

```
Config config = new Config();
CyanSdk.register("cy2uLn0jfoQS", "16bd0e2533092b96e3e55958ef19d08a",
"http://10.2.58.251:8081/login-success.html", config);
cyanSdk = CyanSdk.getInstance(this.getActivity());
```

高亮部分修改为畅言授权的 appid、appkey、callbackUri, 登录 changyan.sohu.com,点击设置

->高级查看配置项, callbackUri 在页面 <http://changyan.sohu.com/setting/further> 中设置。

SDK 的 api 使用示例参见 com.example.cyan_sdk_test.CyActivity.java 中的

ApiSectionFragment。

畅言 SDK 的网络数据请求采用异步方式，需要开发者自己实现 `CyanRequestListener` 接口处理返回数据，数据的反序列化采用泛型方式，需要开发者指定 `CyanRequestListener` 的泛型类型。`CyanRequestListener` 中有 `onRequestSucceeded` 和 `onRequestFailed` 两个接口，前者处理正常响应，处理异常响应，下文中 `api` 参数介绍中出现该实现接口的类，不再详细说明。

1.1 不登录可访问的 api 接口

接口名称	loadTopic	
描述	首次加载文章页面时调用，创建在畅言中的文章信息，加载最新最热列表。	
参数	名称	说明
	topicSourceId	应用提供的文章唯一标识，与 <code>topicUrl</code> 二选一，优先使用 <code>topicSourceId</code>
	topicUrl	文章 url 地址，与 <code>topicSourceId</code> 二选一
	topicTitle	文章标题
	topicCategoryId	新闻分类/频道 id，可选参数，不使用传 <code>null</code>
	pageSize	查询的最新评论数量
	hotSize	查询的最热评论数量
	style	评论展示风格，可选择盖楼模式（ <code>floor</code> ）或缩进模式（ <code>indent</code> ），传 <code>null</code> 时默认盖楼模式
	depth	列表缩进深度，默认为 1，即只缩进一次
	sub_size	子列表最大显示几条
	listener	实现 <code>CyanRequestListener</code> 接口类的实例
返回值	无	

listener 回调 方 法 onRequestS ucceeded 的 参数值	TopicLoadResp.java		
	属性	类型	说明
	topic_id	long	评论 id
	total_page_no	int	总页数
	participation_sum	int	参与数
	share_sum	int	分享数
	cmt_sum	int	评论总数
	hots	ArrayList<Comment>	最热评论列表
	comments	ArrayList<Comment>	最新评论列表
	error_code	int	错误码
	error_msg	String	错误信息
	Comment.java		
	属性名	类型	说明
	content	String	内容
	create_time	long	评论时间
	comment_id	long	评论 id
	support_count	int	顶数
	oppose_count	int	参数
	reply_count	int	回复数
	top	boolean	是否置顶
	ip	String	
	ip_location	String	地域信息
	from	String	
	passport	Passport	评论用户信息
	score	float	文章评分
	metadata	String	
	childComment	Comment	
	comments	ArrayList<Comment>	盖楼列表
	attachments	ArrayList<Comment>	图片附件

	reply_id	long	被回复评论 id

接口名称	getCommentCount		
描述	查询单个文章评论数，可实时查询		
参数	名称	说明	
	topicSourceIds	应用提供的文章唯一标识，与 topicUrl，topicIds 三选一	
	topicUrl	文章 url 地址，与 topicSourceId，topicIds 三选一	
	topicIds	文章在畅言中的 ID,通过 loadTopic 接口获取,与 topicSourceId，topicIds 三选一，优先使用 topicIds，其次使用 topicSourceIds	
	listener	实现 CyanRequestListener 接口类的实例	
返回值	无		
listener 回调方法 onRequestSucceeded 的参数值	TopicCountResp.java		
	属性值	类型	说明
	count	int	单个文章评论数
	error_code	int	错误码
	error_msg	String	错误信息

接口名称	getCommentCount	
描述	批量获取文章评论数，批量接口无法实时更新缓存，所以此接口查询评论数无法实时	
参数	名称	说明
List<String>	topicSourceIds	应用提供的文章唯一标识，与 topicUrl， topicIds 三选一
List<String>	topicUrl	文章 url 地址，与 topicSourceId， topicIds 三选一
List<Long>	topicIds	文章在畅言中的 ID，通过 loadTopic 接口获取，与 topicSourceId， topicIds 三选一，优先使用 topicIds，其次使用 topicSourceIds
	listener	实现 CyanRequestListener 接口类的实例
返回值	无	
listener 回调方法 onRequest Succeeded 的参数值	TopicsCountResp.java	
	属性值	类型
	result	Map<String, TopicCount>
	error_code	int
	error_msg	String
	TopicCount.java	
	属性名	类型
	id	long
	sid	String
	sum	int
	comments	int
	shares	int
	likes	int
	parts	int
	说明	
	畅言中的新闻 id	
	source id	
	评论数，不推荐使用，请使用 comments	
	评论数	
	分享数	
	喜欢数	
	参与数	

接口名称	getTopicComments	
描述	分页查询文章评论数据	
参数	名称	说明

	topicId	畅言分配的文章唯一 ID，通过 loadTopic 接口获取													
	pageSize	分页大小													
	pageNo	页数													
	style	评论展示风格，可选择盖楼模式（floor）或缩进模式（indent），传 null 时默认盖楼模式													
	depth	列表缩进深度，默认为 1，即只缩进一次													
	sub_size	子列表最大显示几条													
	listener	实现 CyanRequestListener 接口类的实例													
返回值	无														
listener 回调方法 onRequestSucceeded 的参数值	TopicCommentsResp.java <table><tr><td>属性名</td><td>类型</td><td>说明</td></tr><tr><td>topic_id</td><td>long</td><td>文章 id</td></tr><tr><td>comments</td><td>List<Comment></td><td>评论</td></tr><tr><td>cmt_sum</td><td>int</td><td>评论总数</td></tr></table>			属性名	类型	说明	topic_id	long	文章 id	comments	List<Comment>	评论	cmt_sum	int	评论总数
属性名	类型	说明													
topic_id	long	文章 id													
comments	List<Comment>	评论													
cmt_sum	int	评论总数													

接口名称	commentAction	
描述	评论顶踩接口	
参数	名称	说明
	topicId	畅言分配的文章唯一 ID，通过 loadTopic 接口获取
	commentId	评论 id
	type	操作类型，枚举类 CommentActionType
	listener	实现 CyanRequestListener 接口类的实例
返回值	无	
listener 回调方法 onRequestSucceeded 的参数值	CommentActionResp.java 服务器返回的 JSON 数据: {}	

tSucceeded 的参数值	
-----------------	--

接口名称	commentReplies														
描述	获取一条评论的所有回复														
参数	名称	说明													
	topicId	畅言分配的文章唯一 ID，通过 loadTopic 接口获取													
	commentId	评论 id													
	pageSize	分页大小													
	pageNo	页数													
	listener	实现 CyanRequestListener 接口类的实例													
返回值	无														
listener 回调方法 onRequestsucceeded 的参数值	CommentReplyResp.java <table><tr><td>属性名</td><td>类型</td><td>说明</td></tr><tr><td>comments</td><td>List<Comment></td><td>一条评论的所有回复</td></tr><tr><td>error_code</td><td>int</td><td>错误码</td></tr><tr><td>error_msg</td><td>String</td><td>错误信息</td></tr></table>			属性名	类型	说明	comments	List<Comment>	一条评论的所有回复	error_code	int	错误码	error_msg	String	错误信息
属性名	类型	说明													
comments	List<Comment>	一条评论的所有回复													
error_code	int	错误码													
error_msg	String	错误信息													

startAuthorize 为登录接口，使用时需要在 AndroidManifest.xml 中注册畅言的 LoginActivity

```
<activity android:name="com.sohu.cyan.android.sdk.activity.OAuthActivity"></activity>
```

调用 startAuthorize 接口的 activity 需要实现 onActivityResult 接口，接受登录是否成功的返回信息，参考 com.example.cyan_sdk_test.LoginActivity。

接口名称	startAuthorize	
描述	登录接口，获取用户信息，用户收到的回复，调用评论发表接口前需要先登录	
参数	名称	说明
	platform_id	通过 oauth 方式登录的第三方平台，见 com.sohu.cyan.android.sdk.util.Constants.PlatformId
	activity	调用登录接口的 activity

返回值	无
-----	---

接口名称	getAnonymousUserInfo		
描述	根据 Config 中设置的 anonymous_token，获取匿名用户信息接口，包括匿名用户的昵称，头像地址等信息。		
参数	名称	说明	
	listener	实现 CyanRequestListener 接口类的实例	
返回值	无		
listener 回调方法 onRequestSucceeded 的参数值	UserInfoResp.java		
	属性名	类型	说明
	nickname	String	昵称
	from	String	账号来源
	img_url	String	头像
	sign	String	
	latest_reply_sum	int	最近回复数
	comment_sum	int	发表评论数
	user_id	long	用户 id

接口名称	anonymousSubmitComment	
描述	发表评论接口	
参数	名称	说明
	topicId	畅言分配的文章 ID，通过 loadTopic 接口获取
	content	评论内容
	replyId	回复的评论，默认为 0
	attachUrl	附件图片地址，通过 attachUpload 接口上传后获取
	score	打分
	appType	平台类型，40:ipone,41:ipad,43:android
	accessToken	匿名用户 token，在站长后台 wap 设置中生成
	listener	实现 CyanRequestListener 接口类的实例

返回值	无								
listener 回调方法 onRequestSucceeded 的参数值	<div>SubmitResp.java</div> <table><tr><th>属性名</th><th>类型</th><th>说明</th></tr><tr><td>id</td><td>long</td><td></td></tr></table> <div>服务器返回 JSON 样例</div> <pre>{ id: 1001 }</pre>			属性名	类型	说明	id	long	
属性名	类型	说明							
id	long								

1.2 需要登录的 api

接口名称	submitComment							
描述	发表评论接口							
参数	名称	说明						
	topicId	畅言分配的文章 ID，通过 loadTopic 接口获取						
	content	评论内容						
	replyId	回复的评论，默认为 0						
	attachUrl	附件图片地址，通过 attachUpload 接口上传后获取						
	score	打分						
	appType	平台类型，40:ipone,41:ipad,43:android						
	listener	实现 CyanRequestListener 接口类的实例						
返回值	无							
listener 回调方法 onRequestSucceeded 的参数值	<div>SubmitResp.java</div> <table><tr><th>属性名</th><th>类型</th><th>说明</th></tr><tr><td>id</td><td>long</td><td></td></tr></table> <div>服务器返回 JSON 样例</div> <pre>{ id: 1001 }</pre>		属性名	类型	说明	id	long	
属性名	类型	说明						
id	long							

	}
--	---

接口名称	attachUpload								
描述	发表评论接口								
参数	名称	说明							
	file	要上传的图片							
	listener	实现 CyanRequestListener 接口类的实例							
返回值	无								
listener 回调方法 onRequestSucceeded 的参数值	<div>AttachementResp.java</div> <table><tr><td>属性名</td><td>类型</td><td>说明</td></tr><tr><td>url</td><td>String</td><td>附件的 url</td></tr></table> <div>服务器返回 JSON,例如:</div> <pre>{ url : "http://img.itc.cn/comment/odYRJeX_jpg" }</pre>			属性名	类型	说明	url	String	附件的 url
属性名	类型	说明							
url	String	附件的 url							

接口名称	getUserInfo	
描述	获取用户信息接口，包括用户的昵称，头像地址，最新回复数，发表评论总数等信息。	
参数	名称	说明
	listener	实现 CyanRequestListener 接口类的实例
返回值	无	

listener 回调方法 onRequestSucceeded 的参数值	UserInfoResp.java		
	属性名	类型	说明
	nickname	String	昵称
	from	String	账号来源
	img_url	String	头像
	sign	String	
	latest_reply_sum	int	最近回复数
	comment_sum	int	发表评论数
	user_id	long	用户 id

接口名称	getUserComments	
描述	获取用户做出的评论	
参数	名称	说明
	pageNo	页数
	pageSize	分页大小
	listener	实现 CyanRequestListener 接口类的实例
返回值	无	
listener 回调方法 onRequestSucceeded 的参数值	UserCommentResp.java	
	属性名	类型
	total_number	int
	comments	List<ReplySendComment>
ReplySendComment.java 参考 Comment.java		

接口名称	getUserNewReply	
描述	获取用户收到的最新回复	
参数	名称	说明
	pageNo	页数

	pageSize	分页大小	
	listener	实现 CyanRequestListener 接口类的实例	
返回值	无		
listener 回调方法 onRequestSucceeded 的参数值	UserReplyResp.java		
	属性名	类型	说明
	total_num	int	收到的最新回复数
	replies	List<ReplyRcvComment>	最新回复列表
	ReplyRcvComment.java 参考 Comment.java		

接口名称	logOut
描述	退出登录
返回值	无

2. 附带 UI 界面的 api 接口

2.1 评论列表界面

使用 viewComment 或 getCommentBtn 需要在项目的 AndroidManifest.xml 中注册 CyanCommentActivity

```
<activity android:name="com.sohu.cyan.android.sdk.activity.CyanCommentActivity"/>
```

接口名称	viewComment	
描述	调用畅言 SDK 的评论列表界面和发表评论界面	
参数	名称	说明
	topicSourceId	文章在 app 中的唯一标识
	topicTitle	文章标题
	context	调用列表页接口的 context
返回值	无	

2.2 评论发表界面

使用 editComment 需要在项目的 AndroidManifest.xml 中注册 CyanCommentActivity

```
<activity android:name="com.sohu.cyan.android.sdk.activity.CyanPostCommentActivity"/>
```

接口名称	editComment	
描述	调用畅言 SDK 的发表评论界面	
参数	名称	说明
	context	
	topicId	文章在畅言系统中的 ID，通过 loadTopic 方法获取（必传）
	replyId	被回复的评论的 Id，默认为 0
	replyNick	被回复的用户名称，默认为 null
返回值	无	

2.3 用户评论中心界面(包括发出的评论和接收到的评论)

需要在项目的 AndroidManifest.xml 中注册 RepliesActivity

```
<activity android:name="com.sohu.cyan.android.sdk.activity.RepliesActivity"/>
```

接口名称	showUserSendAndRcvComments	
描述	调用畅言 SDK 的用户评论中心界面(包括发出的评论和接收到的评论)	
参数	名称	说明
	des	0:跳转到评论中心的接收到的评论页面 1: 跳转到评论中心的做出的评论页面
返回值	无	

3.消息推送

配置见 三.7

设置 notification 的图标

```
PushManager.setNotificationIcon(Context ctx,int DrawableId);
```

如何自定义 处理接收到消息推送.

```
public class YourReceiver extends BroadcastReceiver {  
  
    @Override  
    public void onReceive(Context context, Intent intent) {  
  
        if (intent.getAction().equals(PushConstants.PUSH_RECEIVE_ACTION)){  
            JSONObject json = new JSONObject(intent.getExtras().getString(  
                "com.avos.avoscloud.Data"));  
            final String message = json.getString("message");//收到的消息  
                ...//自己的处理代码  
        }  
    }  
}
```

在 androidManifest.xml 中的

```
<receiver  
android:name="com.sohu.cyan.android.sdk.push.CyanPushReceiver" >  
改为  
<receiver android:name="*.YourReceiver" >
```

如果使用自定义的评论编辑页面

回复别人的评论成功后, 需要在 new CyanRequestListener<SubmitResp>()的回调方法 public void onRequestSucceeded(final SubmitResp SubmitData)

做如下操作:

PushManager.pushMessage(context , user_id + "", yourStr);

user_id:为被回复的用户 user_id.

yourStr:为你要推送的内容

注意事项

1.需要在 androidmanifest 中配置 application,receiver,service

2.PushManager.setUseService(Context, Boolean); 需要 写 在

CyanSdk.register(...)之后

附录

关于代码混淆

畅言 sdk 中使用了 java 反射和泛型，因此在对项目进行代码混淆时需要保证 sdk jar 包中的某些类不被混淆。代码混淆配置文件请参考 demo 工程下的 proguard-project.txt 文件。