

互动功能发起端 SDK

概述

本项目是保利威直播互动功能发起端（即开播端、管理端等）逻辑层 SDK，可以基于本 SDK 开发互动功能的发起界面。

使用说明

安装

```
npm i -S @polyv/interactions-launch-sdk
```

引入

```
import * as InteractionsLaunchSDK from '@polyv/interactions-launch-sdk';
```

全局初始化

注意：无论使用哪个互动功能，都要先执行全局初始化。全局初始化只需要执行一次。

按照以下步骤进行全局初始化：

1. 设置日志打印等级。

日志会打印到控制台并上传到 polyv 服务器。日志上传服务默认开启，可以通过 `Logger.disableUploadLog()` 关闭。如无特别原因，请保持打开日志上传，以便我们帮助您定位线上问题。

```
InteractionsLaunchSDK.Logger.setLogLevel(InteractionsLaunchSDK.Logger.LogLevel.TRACE);
```

2. 使用 `setChatInfo()` 方法传入 polyv 聊天室实例对象和数据。

互动功能发起端 SDK 内部依赖于 polyv 聊天室服务。您可以基于 polyv 聊天室 SDK 快速接入该服务。

`PolyvChatRoom` 是聊天室 SDK 提供的全局对象，更多使用细节请查看[这篇文档](#)。

`setChatInfo()` 方法支持多次调用。如，在聊天室重连后需要重新设置 `chat` 和 `socketId` 属性。

```
// 初始化聊天室 SDK
var chatroom = new PolyvChatRoom({
  version: '2.0', // 注意这里要使用 '2.0'
  roomId: '', // 房间号
  userId: '', // 用户在聊天室的 uid
  pic: '', // 用户头像
  nick: '', // 用户昵称
  userType: '', // 用户类型: 'teacher' 老师 / 'assistant' 助教 / 'manager' 管理员
  token: '', // 校验码
  // 其它具体参数信息请参考 polyv 聊天室 SDK
});

// 在 connect 事件回调中拿到聊天室实例和 socketId 后再传入到互动 SDK 中
chatroom.chat.on('connect', () => {
  const socket = chatroom.chat.socket;

  // 设置聊天室实例对象和数据
  InteractionsLaunchSDK.setChatInfo({
    // 聊天室实例对象。
    chat: socket,
    // 连接聊天室的 socketId
    socketId: socket.id,
    // 用户在聊天室的 uid
    userId: '',
  });
});
```

3. 使用 `setChannelInfo()` 方法初始化频道信息。

该方法支持多次调用。**注意：当 `sessionId` 发生变化时（一般是直播状态发生改变时发生），必须调用该接口更新配置。**如，`setChannelInfo({ sessionId: 'new sessionId' })`。

此外，在切换房间时，也可以调用该接口更新 `channelId` 和 `roomId`。

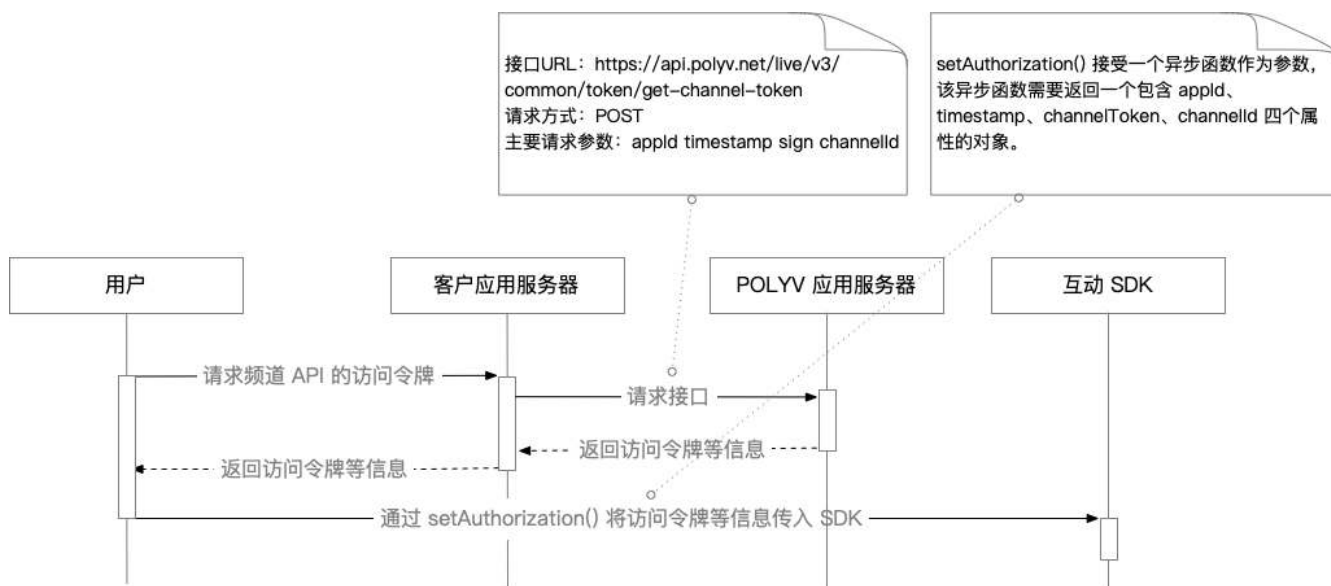
```
InteractionsLaunchSDK.setChannelInfo({
  channelId: '',
  roomId: '',
  // 频道类型。默认为 'normal'。
  channelType: '',
  // 直播场次 id
  sessionId: '',
});
```

4. 设置授权。SDK 内部请求频道 API 接口时需要使用到访问令牌等信息。

`setAuthorization()` 接受一个异步函数作为参数，该异步函数需要返回一个包含 `appId`、`timestamp`、`channelToken`、`channelId` 四个属性的对象。

`appId` 是账号 `AppId`；`timestamp` 是 13 位时间戳；`channelId` 是频道号；`channelToken` 的获取请参考[获取频道API的访问令牌](#)。

获取授权流程介绍：



```
InteractionsLaunchSDK.setAuthorization(async () => {
    // 此处异步获取频道API的访问令牌，并返回互动 SDK 中实现授权所需的参数。
    return {
        appId: '',
        timestamp: '',
        channelId: '',
        channelToken: '',
    };
});
```

5. （可选）如果有域名定制的需求，可设置自定义域名。

```
InteractionsLaunchSDK.setCustomDomain({
    // 可选。聊天室接口域名。默认为 'apichat.polyv.net'
    chat: '',
    // 可选。直播业务域名。默认为 'api.polyv.net'
    api: '',
});
```

6. 按需要实例化各互动功能类。

```
// 以签到功能为例
// 1. 实例化 CheckIn 类
const checkInCore = new InteractionsLaunchSDK.CheckIn();
// 2. 注册事件监听函数
checkInCore.on('show-result', handler1);
```

API 文档

点击[此处](#)可查看 API 文档。

功能实践

此处仅提供各功能参考文档链接。

- AnswerCard 答题卡: [参考文档](#)
- CheckIn 签到: [参考文档](#)
- Timer 计时器: [参考文档](#)
- LotteryNew 抽奖: [参考文档](#)
- Notice 公告: [参考文档](#)
- Questionnaire 问卷: [参考文档](#)
- RunAnswer 抢答: [参考文档](#)

版本更新

暂无