

亮眼云 VV SDK

开发手册

V1.0

北京亮眼云视科技有限公司

2018 年 2 月 9 日

目录

目录	2
版权声明	3
版本说明	4
1. 产品介绍	5
2. 开发指南	5
SDK 系统运行环境	5
开发流程	6
使用流程	6
API 接口	7
创建实例	7
销毁实例	7
回收资源	7
初始化 ShineVVSDK	7
释放 SDK	11
设定渲染窗口	11
加入房间	12
退出房间	12
修改本地视频状态	13
修改本地音频状态	13
3. 常见问题及错误码	21
常见问题	21

错误码列表.....	21
------------	----

版权声明

1. 本版权声明是北京亮眼云视科技有限公司关于 VV SDK 软件产品的全部版本（包括已有版本及未来更新版本）及与该软件作品全部版本有关的源代码、目标代码、文档资料以及任何由北京亮眼云视科技有限公司基于软件技术维护或支持服务所提供的数据库及查询方式、数据、资料等（以下统称：本软件作品）做出的法律声明。
2. 本软件作品的著作权、商标权等知识产权属于北京亮眼云视科技有限公司所有，受《中华人民共和国著作权法》、《计算机软件保护条例》、《知识产权保护条例》和相关国际版权条约、法律、法规，以及其它知识产权法律和条约的保护。
3. 任何单位和个人未经北京亮眼云视科技有限公司书面授权，不得以任何目的（包括但不限于学习、研究等非商业用途）修改、使用、复制、截取、编纂、编译、上传、下载等或以任何方式和媒介复制、转载和传播本软件作品的任何部分，否则将视为侵权，北京亮眼云视科技有限公司保留依法追究其法律责任的权利。

版本说明

版本号	更新时间	主要内容或修改
1.0	2018.2.9	1. VV SDK 的整体介绍； 2. VV SDK 的主要 API 接口说明。

1. 产品介绍

亮眼云 VV 系列产品包括亮眼会议以及亮眼课堂等软件，提供了丰富的会议/课堂功能，包括多人音视频、电子白板、多格式文档在线浏览等。

- 1、全互动音视频会议/课堂，支持 1v1、1vN 多种在线会议/课堂形式；
- 2、支持 Windows/Linux/安卓/iOS/Web 方式进入课堂；
- 3、主持人可对整个会议/课堂过程进行主持和控制（禁言、解禁、禁视、解禁视频）；
- 4、支持 IM 实时聊天，显示当前会议/课程进行时间；
- 5、支持电子白板包括画笔、形状、线条、颜色等工具，实时多方联动显示；
- 6、支持多种格式课件实时共享，包括 PPT/PDF/DOC 文件、图片、MP4 视频文件等；
- 7、支持课件在线演示实时多方联动；

VV SDK 产品为开发者提供了面向不同平台标准的 API 接口，方便开发者快速应用集成包含以上功能的亮眼云音视频通信和电子白板互动功能。

2. 开发指南

开发者集成使用 VV SDK 的 API 接口功能，首先在亮眼云公司主页（<http://www.shinevv.com>）联系公司客服注册和开通账户，通过开通后的账号登录亮眼云系统，获取开发者账户授权，并开通相应系统的应用权限。

SDK 系统运行环境

VV SDK 的运行需要以下软硬件环境：

操作系统	Windows 32 位或 64 位系统
音视频设备	USB 摄像头/集成摄像头/耳麦/音箱
最低配置	CPU : 1.8G Hz 内存 : 1G 硬盘 : 5G

开发流程

流程说明：

1. 下载亮眼云 VV SDK 压缩包，包括
2. 在客户端工程内引入 shinevv.dll.lib 库文件，代码中加入 shinevv.h 头文件。

使用流程

业务流程参考 Demo 中的 SDK 应用实例，包括以下步骤

- 1、调用 CreateShinevvInstance，创建 ShineVV 实例句柄；
- 2、初始化 SDK，连接亮眼云服务器；
- 3、创建本地音视频以及远端音视频的渲染窗口；
- 4、进入在线会议/课堂房间；
- 5、进行在线音视频通话，包括禁言，禁视等操作；
- 6、退出软件，销毁 ShineVV 实例；
- 7、回收资源

API 接口

创建实例

- `void* CreateShinevvInstance();`
- 调用参数说明：

参数名	参数类型	说明
无		

- 返回值说明：

参数类型	说明
void*	返回应用实例句柄，结果为空表示调用失败。

销毁实例

- `void DestoryShinevvInstance();`
- 调用参数说明：

参数名	参数类型	说明
无		

- 返回值说明：

参数类型	说明
无	

回收资源

- `void ClearShinevv();`
- 调用参数说明：

参数名	参数类型	说明
无		

● 返回值说明：

参数类型	说明
无	

初始化 ShineVVSDK

```
void InitSDK(void* handle, const char* pRoomId, MediaType eMediaType,
const char* pMemberId, const char* pDisplayName, const char* pServerIp,
int nPort, const char* pToken,
void(*OnJoined)(ErrorCode eCode),
void(*OnDisConnected>(),
void(*OnNewMemberJoined)(const char* pMemberId, const char*
pDisplayName),
void(*OnMemberLeft)(const char* pMemberId, const char* pDisplayName),
void(*OnModifyLocalAudioStatus)(bool bOpen),
void(*OnModifyLocalVideoStatus)(bool bOpen),
void(*OnMemberEnableVideo)(const char* pMemberId, const char*
pDisplayName),
void(*OnMemberDisableVideo)(const char* pMemberId, const char*
pDisplayName, void* pRenderWin),
```


void(*OnMemberUnMuteAudio)(const char* pMemberId, const char*

pDisplayName),

void(*OnMemberMuteAudio)(const char* pMemberId, const char*

pDisplayName),

void(*OnSessionError)(const char* reason))

● **调用参数说明：**

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
pRoomId	字符串	房间号
eMediaType	枚举值	媒体会话类型
pMemberId	字符串	成员标识符，必须在当前房间中保持唯一
pDisplayName	字符串	成员昵称
pServerIp	字符串	服务器 IP
nPort	整型	服务器端口
pToken	字符串	认证信息,此为预留接口，目前还没有认证，这里可以给一个任意字符串
OnJoined	函数指针	加入房间回调,调用 JoinRoom 后会触发
OnDisConnected	函数指针	断开连接回调,当此回调被触发时，要依次调用 LeaveRoom()，ReleaseSdk()来释放资源
OnNewMemberJoined	函数指针	新成员加入房间回调，在此回调中调用 SetRenderWindow()接口来设置新成员视频渲

		染窗口句柄
OnMemberLeft	函数指针	成员离开房间回调
OnModifyLocalAudioStatus	函数指针	本地音频状态改变回调， JoinRoom(),ModifyAudioStatus(),都会触发
OnModifyLocalVideoStatus	函数指针	本地视频状态改变回调， JoinRoom(),ModifyVideoStatus(),都会触发
OnMemberEnableVideo	函数指针	成员打开视频回调
OnMemberDisableVideo	函数指针	成员关闭视频回调
OnMemberUnmuteAudio	函数指针	成员打开音频回调
OnMemberMuteAudio	函数指针	成员关闭音频回调
OnSessionError	函数指针	会话失败回调，会话失败的原因包括设备被占用，网络堵塞导致消息超时，消息内容错误等等

● 媒体会话类型说明：

/**

* 媒体会话类型

* @Audio 语音通话

* @Video 视频通话

*/

```
enum MediaType {  
  
    Audio,  
  
    Video  
  
};
```

- 返回值说明：

参数类型	说明
无	

释放 SDK

- `void ReleaseSdk(void* handle);`

- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄

- 返回值说明：

参数类型	说明
无	

设定渲染窗口

- `void SetRenderWindow(void* handle, const char* pMemberId, void* pRenderWin);`

- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
pMemberId	字符串	成员标识符
pRemoteRenderWin	指针	视频渲染窗口句柄

- 返回值说明：

参数类型	说明
无	

加入房间

- void JoinRoom(void* handle);

- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄

- 返回值说明：

参数类型	说明
无	

退出房间

- void LeaveRoom(void* handle);

- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄

- 返回值说明：

参数类型	说明
无	

修改本地视频状态

- `void ModifyVideoStatus(void* handle, bool bOpen);`
- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
bOpen	布尔	true，显示本地视频；false，关闭本地视频

- 返回值说明：

参数类型	说明
无	

修改本地音频状态

- `void ModifyAudioStatus(void* handle, bool bOpen);`
- 调用参数说明：

参数名	参数类型	说明
-----	------	----

handle	空指针	Shinevv 实例句柄
bOpen	布尔	true , 开启本地麦克风 ; false , 关闭本地麦克风

● 返回值说明：

参数类型	说明
无	

接收指定成员视频

- **void EnableMemberVideo(void* handle, const char* pMemberId, void(*EnableMemberVideoResult)(bool bSucc, const char* pMemberId));**

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
pMemberId	string	成员标识符
EnableMemberVideoResult	函数指针	接收指定成员视频结果回调

● 返回值说明：

参数类型	说明
无	

不接收指定成员视频

- **void DisableMemberVideo(void* handle, const char* pMemberId,**

void(*DisableMemberVideoResult)(bool bSucc, const char* pMemberId));

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
pMemberId	string	成员标识符
DisableMemberVideoResult	函数指针	不接收指定成员视频结果回调

● 返回值说明：

参数类型	说明
无	

接收指定成员音频

- **void EnableMemberAudio(void* handle, const char* pMemberId, void(*EnableMemberAudioResult)(bool bSucc, const char* pMemberId));**

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
pMemberId	string	成员标识符
EnableMemberAudioResult	函数指针	接收指定成员音频结果回调

● 返回值说明：

参数类型	说明
无	

不接收指定成员音频

- **void DisableMemberAudio(void* handle, const char* pMemberId, void(*DisableMembeAudioResult)(bool bSucc, const char* pMemberId));**

- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
pMemberId	string	成员标识符
DisableMemberAudioResult	函数指针	不接收指定成员音频结果回调

- 返回值说明：

参数类型	说明
无	

创建画板

void CreatePaletteInstance(void* handle, void* hwndParent, int nLeft, int nTop, int nWidth, int nHeight, bool bShow, void(*OnCreatePaletteResult)(bool bSucc));

- 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
hwndParent	空指针	成员标识符
nLeft	int	画板窗口初始水平位置
nTop	int	画板窗口初始垂直位置
nWidth	int	画板宽度
nHeight	int	画板高度
bShow	bool	画板窗口初始水平位置
OnCreatePaletteResult	函数指针	创建画板结果回调

● 返回值说明：

参数类型	说明
无	

销毁画板

void DestoryPaletteInstance(void* handle);

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄

● 返回值说明：

参数类型	说明
无	

显示或隐藏画板

```
Void ShowPalette(void* handle, bool bShow,
                void(*OnShowPaletteResult)(bool bSucc));
```

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
bShow	bool	是否显示画板
OnShowPaletteResult	函数指针	结果回调函数

● 返回值说明：

参数类型	说明
无	

设置画笔类型

```
Void SetPalettePenType(void* handle, PenType nType,
                      void(*OnSetPalettePenTypeResult)(bool bSucc));
```

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
nType	PenType	画板类型
OnSetPalettePenType eResult	函数指针	结果回调函数

● 返回值说明：

参数类型	说明
无	

设置画笔宽度

Void SetPalettePenWidth(void* handle, float fWidth,

void(*OnSetPalettePenWidthResult)(bool bSucc));

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
fWidth	float	画笔宽度
OnSetPalettePenWidth Result	函数指针	结果回调函数

● 返回值说明：

参数类型	说明
------	----

无	
---	--

设置画笔颜色

Void SetPalettePenColor(void* handle, unsigned char uRed,
unsigned char uGreen, unsigned char uBlue,
void(*OnSetPalettePenColorResult)(bool bSucc));

● 调用参数说明：

参数名	参数类型	说明
handle	空指针	Shinevv 实例句柄
uRed	unsigned char	红色分量
uGreen	unsigned char	绿色分量
uBlue	unsigned char	蓝色分量
OnSetPalettePenColorResult	函数指针	结果回调函数

● 返回值说明：

参数类型	说明
无	

3. 常见问题及错误码

常见问题

错误码列表

接口调用结果 code 说明

返回值	说明
VVSuccess	接口调用成功
VVConnectFail	连接服务器失败
VVCreateMediaSessionFail	创建媒体会话失败
VVAlreadyJoined	重复加入

```
enum ErrorCode {  
  
    VVSuccess = 0,  
  
    VVConnectFail,  
  
    VVCreateMediaSessionFail,  
  
    VVAlreadyJoined  
  
};
```