

# SDK集成

本文主要介绍如何快速地将启业云乐高SDK（QYCLegoSDK iOS）集成到您的项目中，按照如下步骤进行配置，就可以完成 SDK 的集成工作。

## 开发环境要求

- Xcode 9.0+。
- iOS 9.0 以上的 iPhone 或者 iPad 真机。
- 项目已配置有效的开发者签名。

## 集成 QYCLegoSDK

您可以选择使用 CocoaPods 自动加载的方式，或者先下载 SDK，再将其导入到您当前的工程项目中。

## CocoaPods

### 1. 安装 CocoaPods

在终端窗口中输入如下命令（需要提前在 Mac 中安装 Ruby 环境）：

```
1 | sudo gem install cocoapods
```

### 2. 创建 Podfile 文件

进入项目所在路径，输入以下命令之后项目路径下会出现一个 Podfile 文件。

```
1 | pod init
```

### 3. 编辑 Podfile 文件

编辑 Podfile 文件，使用 CocoaPod 官方源，支持选择版本号。

```
1 | platform :ios, '9.0'
2 | source 'https://github.com/CocoaPods/Specs.git'
3 | source 'http://git.qpaas.com/PaasPods/PaasSpecs.git'      # 组件索引库
4 |
5 | target 'App' do
6 |   pod 'Lego_H5_Framework', '~> 1'
7 | end
```

## 4. 更新并安装 SDK

在终端窗口中输入如下命令以更新本地库文件，并安装 LiteAVSDK：

```
1 | pod install
```

或使用以下命令更新本地库版本：

```
1 | pod update
```

pod 命令执行完后，会生成集成了 SDK 的 `.xcworkspace` 后缀的工程文件，双击打开即可。

## 5. Build Settings配置

选中 Build Settings 项，搜索 `Other Linker Flags`。添加 `-ObjC`、`-all_load`。

## 手动集成

1. 下载 [QYCLegoSDK](#)，下载完成后进行解压。
2. 打开您的 Xcode 工程项目，选择要运行的 target，选中 **Build Phases** 项，单击 **Link Binary with Libraries** 项展开，单击底下的【+】添加依赖库。
3. 依次添加所下载的 `QYCLegoSDK` 及其所需依赖库：

```
1 | CoreGraphics.framework
2 | CoreGraphics.framework
3 | Security.framework
4 | WebKit.framework
```

4. 选中 Build Settings 项，搜索 `Other Linker Flags`。添加 `-ObjC`、`-all_load`。
5. 授权相机相册等使用权限

若使用 SDK 的扫一扫、录音、拍视频等功能，需要授权麦克风、摄像头等使用权限。在 App 的 Info.plist 中添加，即可在系统弹出授权对话框中显示提示信息。

- Privacy - Camera Usage Description
- Privacy - Photo Library Usage Description
- Privacy - Location Always and When In Use Usage Description
- Privacy - Location When In Use Usage Description
- Privacy - Microphone Usage Description
- Privacy - Speech Recognition Usage Description

## 使用

在项目需要使用 SDK API 的文件里，添加模块引用。

```
1 | @import Lego_H5;
```

## 初始化

如果已经在其他途径登录了启业云平台了，可以直接是使用下面的方法初始化

```
1 | /// 注册乐高 + 乐高初始化
2 | /// @param url 基础URL
3 | /// @param userId 用户Id
4 | /// @param entId 企业Id
5 | /// @param callback 回调
6 | - (void)registerWithBaseURL:(NSString *)url
7 |                      userId:(NSString *)userId
8 |                      entId:(NSString *)entId
9 |                      callback:(void (^)(BOOL result, id response))callback;
```

如果没有对启业云平台进行鉴权，必须要使用下面方法初始化

```
1 | /// 初始化
2 | /// @param baseUrl 基础URL
3 | /// @param userId 用户ID
4 | /// @param secret 访问启业云的secret，在启业云官网申请
5 | /// @param key 访问启业云的key，在启业云官网申请
6 | /// @param success 回调
7 | /// @param failure 回调
8 | - (void)sdkRegisterLoginWithBaseURL:(NSString *)baseUrl
9 |                      userId:(NSString *)userId
10 |                      accessSecret:(NSString *)secret
11 |                      accessKey:(NSString *)key
12 |                      success:(void (^)(NSDictionary *result))success
13 |                      failure:(void (^)(NSError *result))failure;
```

注意，accessSecret、accessKey 需要在启业云官网申请

## 手写签名

### 原生调用

```
1 | @import Lego_H5;
2 |
3 | @interface ViewController () <QYCSignatureVCDelegate>
4 | @end
```

```

5
6 ...
7 QYCSignatureVC *vc = [[QYCSignatureVC alloc] init];
8 vc.delegate = self;
9 vc.modalPresentationStyle = UIModalPresentationFullScreen;
10 [self presentViewController:vc animated:YES completion:nil];
11 ...
12
13 - (void)signatureVC:(UIViewController *)vc commitSignatures:(NSArray *)signatures {
14     //...
15 }

```

## Webview调用（SDK的默认方式）

```

1 import { jsBridgeReady } from "@qycloud/js-bridge";
2 jsBridgeReady(function(bridge) {
3     bridge.callHandler(
4         "chooseImage",
5         {
6             autograph: false, // 是否是手写签名
7             signType: 'one' // 手写签名个数
8         },
9         function(res) {
10             var localIds = res.localIds; // 返回选定照片的本地ID列表，localId可以作为img标签的
            src属性显示图片
11             // res = {localIds:[{fileName:'',fileUrl:''},{fileName:'',fileUrl:''}]}
12         }
13     );
14 });

```

## 预览

- 返回：返回上级页面
- 重写：清空
- 撤销：撤销上一步
- 提交：完成提交

陈子

返回

重写

撤销

提交

