# RJWechatManager

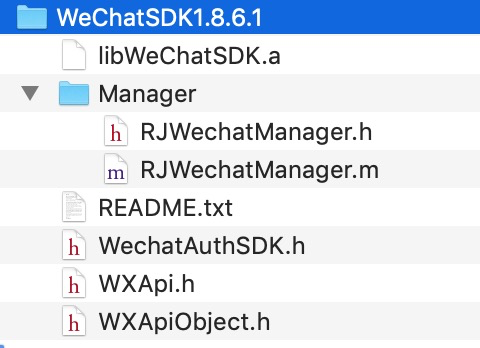
# 接入参考指南

# 修订记录

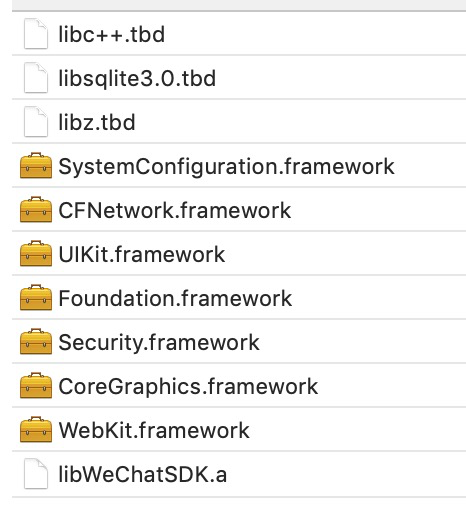
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 归档日期 | 版本 | 版本说明 | 升级指导 | 开发人员 |
| 2020-02-25 | 1.0.1 | 添加app分享 | 重新接入 | 陈荣杰 |

#### **1 集成**

1）导入工程



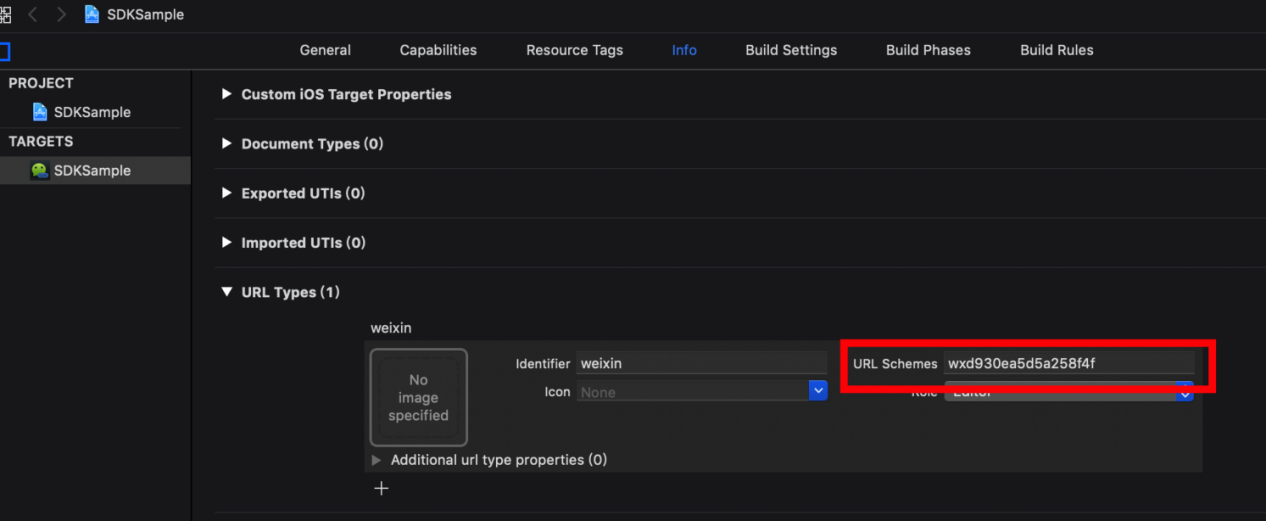
2）链接下面所有静态库



3）在Build Setting，在"Other Linker Flags"中加入"-ObjC -all\_load"

## 2 info配置

#### 1）在 Xcode 中，选择你的工程设置项，选中“TARGETS”一栏，在“info”标签栏的“URL type“添加“URL scheme”为你所注册的应用程序 id（如下图所示）



#### 2）在Xcode中，选择你的工程设置项，选中“TARGETS”一栏，“info”标签栏的“LSApplicationQueriesSchemes“添加weixin 和weixinULAPI（如下图所示）

## 

#### 在info.plist中添加“App Transport Security Settings“设置（如下图所示）



#### **3在代码中使用微信开发工具包**

要使你的程序启动后微信终端能响应你的程序，必须在代码中向微信终端注册你的 id。（如下图所示，在 AppDelegate 的 didFinishLaunchingWithOptions 函数中向微信注册 id）。

1）配置APPID、UNIVERSAL\_LINK和 AppSecret

#define APP\_ID @"wx9ecf48468f517cbc"

#define UNIVERSAL\_LINK @"https://shop.lanhuzi.com/"

#import "RJWechatManager.h"

2）重写下列方法

- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions {

//向微信注册

[WXApi registerApp:APP\_ID universalLink:UNIVERSAL\_LINK];

return YES;

}

-(BOOL)application:(UIApplication \*)app openURL:(NSURL \*)url options:(NSDictionary<UIApplicationOpenURLOptionsKey,id> \*)options

{//ios13以下重写该方法

    return [WXApi handleOpenURL:url delegate:[RJWechatManager sharedManager]];

}

- (BOOL)application:(UIApplication \*)application continueUserActivity:(NSUserActivity \*)userActivity restorationHandler:(void(^)(NSArray<id<UIUserActivityRestoring>> \* \_\_nullable restorableObjects))restorationHandler {

    return [WXApi handleOpenUniversalLink:userActivity delegate:[RJWechatManager sharedManager]];

}

-(void)scene:(UIScene \*)scene openURLContexts:(NSSet<UIOpenURLContext \*> \*)URLContexts

API\_AVAILABLE(ios(13.0)){//ios13 重写该方法

    NSEnumerator \*enumerator = [URLContexts objectEnumerator];

    UIOpenURLContext \*context;

    while (context = [enumerator nextObject]) {

        [WXApi handleOpenURL:context.URL delegate:[RJWechatManager sharedManager]];

    }

}

## 4 使用RJWechatManager

Import “RJWechatManager.h”

登录接口：

示例：

添加代理<RJWecatDelegate>

[RJWechatManager sharedManager].delegate = **self**;

[RJWechatManager sendAuthRequest:nil viewController:**self** delegate:[RJWechatManager sharedManager]];

登录回调：

登录成功，获取用户信息

-(**void**)ResponseOnResp:(NSDictionary \*)dict

{

    NSString \*access\_token  =  dict[@"access\_token"]; *// 接口调用凭证(有效期2h)*

    NSString \*openid        =  dict[@"openid"];

    [**self** onLoginResult:@"success" uid:openid sign:access\_token cid:[**self** getChannelName] para:**nil** url:**nil**];

}

## 5 App微信分享

**@brief** 发送Auth请求到微信，支持用户没安装微信，等待微信返回onReq

+(**void**)sendAppContentTitle:(NSString \*)title

                   extInfo:(NSString \*)extInfo

                       Url:(NSString \*)url

               Description:(NSString \*)description

                     Scene:(**enum** WXScene)scene;

|  |  |  |
| --- | --- | --- |
| 参数 | 描述 | 其他 |
| title | 标题消息 |  |
| extInfo | 额外消息 传给第三方处理 |  |
| url | 微信终端会回传给第三方程序处理 |  |
| description | 内容 |  |
| scene | 发送场景（会话，朋友圈等） |  |

示例：

[RJWechatManager sendAppContentTitle:@"这是测试标题" extInfo:@"11" Url:@"https://giqi.github.io/" Description:@"这是测试内容" Scene:WXSceneSession];