iOS 平台 SDK 文档

编号：WJOAUTH_IOS_SDK

版本：WJOAUTH _IOS_SDK V1.0.0

修订记录：

时间	文档版本	修订人	备注
2018/05/14	1.0.0	周美丽	初稿

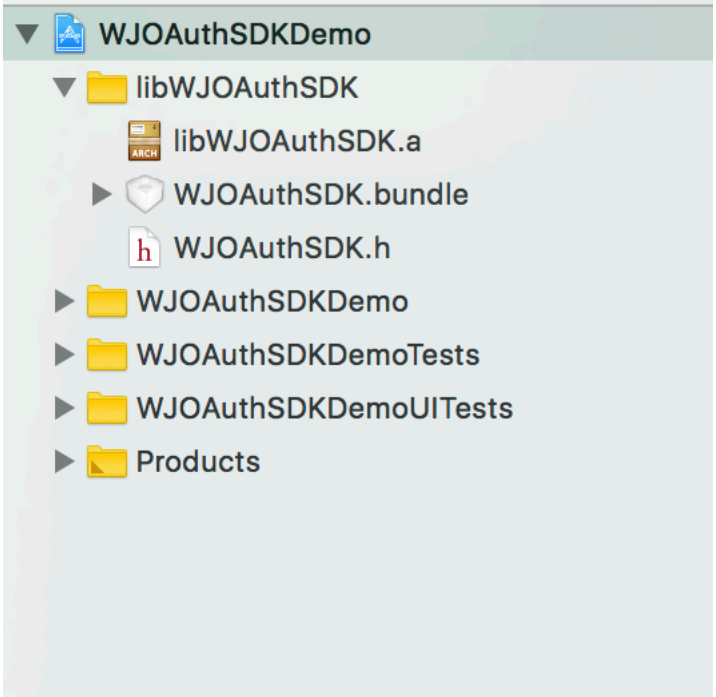
目录

WJOAuth SDK

一、SDK 接入设置

1、添加 SDK 文件到工程

将从 GitHub 上下载的 libWJOAuthSDK 文件夹添加至工程，其中包含 WJOAuthSDK.h 文件以及 libWJOAuthSDK.a 和 WJOAuthSDK.bundle，统共 3 个文件。



2、在工程中引入静态库之后，需要在编译时添加–objC 编译选项

避免静态库中类加载不全造成程序崩溃。方法：程序 Target->Buid Settings- >Linking 下 Other Linker Flags 项添加-ObjC。

3、添加 FrameWork 文件到工程

在工程中修改 Other Linker Flags 后，需要修改编译步骤的链接库设置，避免链接阶段由于库的设置错误导致程序崩溃。方法：程序 Target->Buid Phases->Link Binary With Libraries 下添加以下 Framework 至工程中。需要添加的 Frameworks 为: SystemConfiguration.framework。

4、定义应用 Oauth2.0 认证所需的几个常量

AppKey:第三方应用申请的 appkey,用来身份鉴证、显示来源等;AppRedirectURL:应用回调页,在进行 Oauth2.0 登录认证时所用。对于 Mobile 客户端应用来说,是不存在 Server 的,故此处的应用回调页地址只要与应用回调页中的 url 地址保持一致就可以了,如图所示:

```
#define kAppKey         @"3573952935"
#define kRedirectURI    @"http://your_callback_uri"
```

5、注册 appkey(clientid)

程序启动时,在代码中向 WJOAuthSDK 注册你的 Appkey，可设置授权界面语言，默认跟随系统。

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    // Override point for customization after application launch.

    [WJOAuthSDK setLanguageType:WJLanguageTypeChinese];

    [WJOAuthSDK registerApp:kAppKey];

    return YES;
}
```

二、应用场景代码示例

1、oauth2.0 授权认证

调用[WJOAuthSDK sendRequest:delegate:]的方法后会跳转到 WJOAuthSDK，进入登录界面进行授权，当授权完成后会回调给第三方应用程序,第三方实现 WJOAuthSDKDelegate 的相关方法监听此次请求的 response. 此中 state 内容为用户自定义(可不填写)，WJOAuthSDK 回调 Response 中会通过 requestState 包含原 request.state 中的所有数据，用于数据传输过程中校验相关的上下文环境数据。

2、从第三方应用向 WJOAuthSDK 发送请求

代码示例如:

```
WJOAuthRequest *request = [WJOAuthRequest request];
request.state = @"Verification";
request.redirectURI = kRedirectURI;
[WJOAuthSDK sendRequest:request delegate:self];
```

同上此中 state 内容为用户自定义(可不填写)，WJOAuthSDK 回调 Response 中会通过 requestState 包含原 request.state 中的所有数据，用于数据传输过程中校验相关的上下文环境数据。

具体调用示例参见demo