CEF IOS SDK

开发文档（预览）

Azure China Engineering

2018/10

Contents

[简介 2](#_Toc524702460)

[目录结构 2](#_Toc524702461)

[工程配置 3](#_Toc524702462)

[验证码OTP 7](#_Toc524702463)

[社交登录验证 8](#_Toc524702464)

# 简介

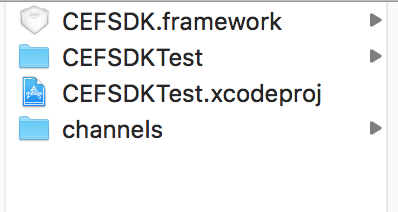
Customer Engagement Fabric (CEF) 是一个App开发者平台，用统一的API帮助开发者快速搭建与用户相关的通用功能。目前提供：

**验证类：**验证码 One-Time-Password (OTP), 社交登录验证

开发者可以选择启用单个或多个功能。

## 目录结构

获取CEFSDK-iOS, 解压压缩包之后结构如图所示：



其中CEFSDK.framewrok是CEF的SDK文件，CEFSDKTest 文件夹是测试工程的代码，CEFSDKTest.xcodeproj是测试工程的工程文件， channels里面包含了需要用到的第三方SDK, 包括如下3个，也可以自行从各个SDK官方网站下载。



|  |  |  |
| --- | --- | --- |
| SDK | 版本 | 官方下载网址 |
| 微博(libWeiboSDK) | 3.1.4 | https://github.com/sinaweibosdk/weibo\_ios\_sdk |
| 腾讯(TencentOpenAPI) | 3.3.3 | http://wiki.open.qq.com/wiki/mobile/SDK下载 |
| 微信(WeChatOpenSDK) | 1.8.1 | https://open.weixin.qq.com/cgi-bin/showdocument?action=dir\_list&t=resource/res\_list&verify=1&id=open1419319164&token=&lang=zh\_CN |

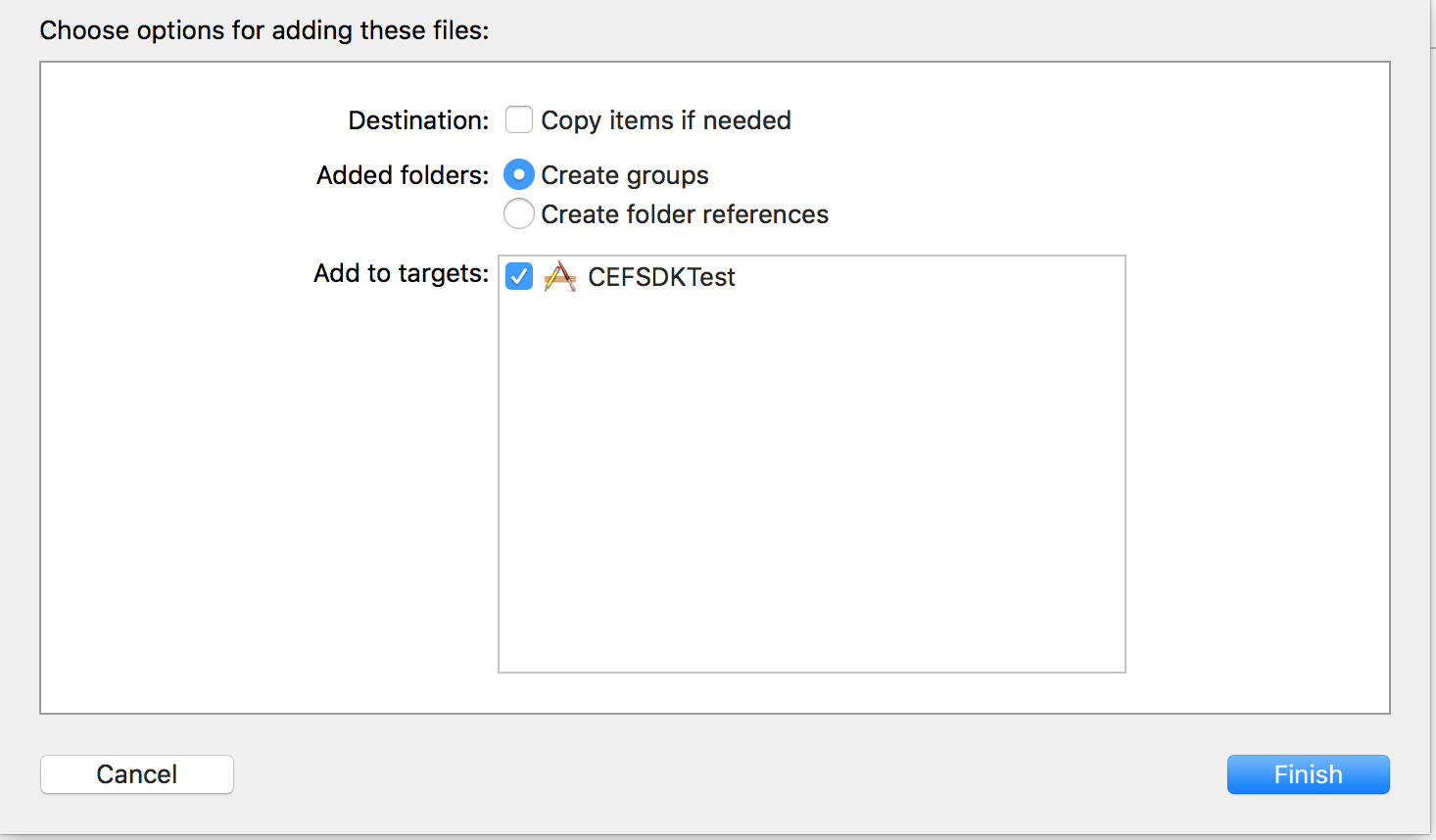
# 工程配置

1. 配置个人开发者账号

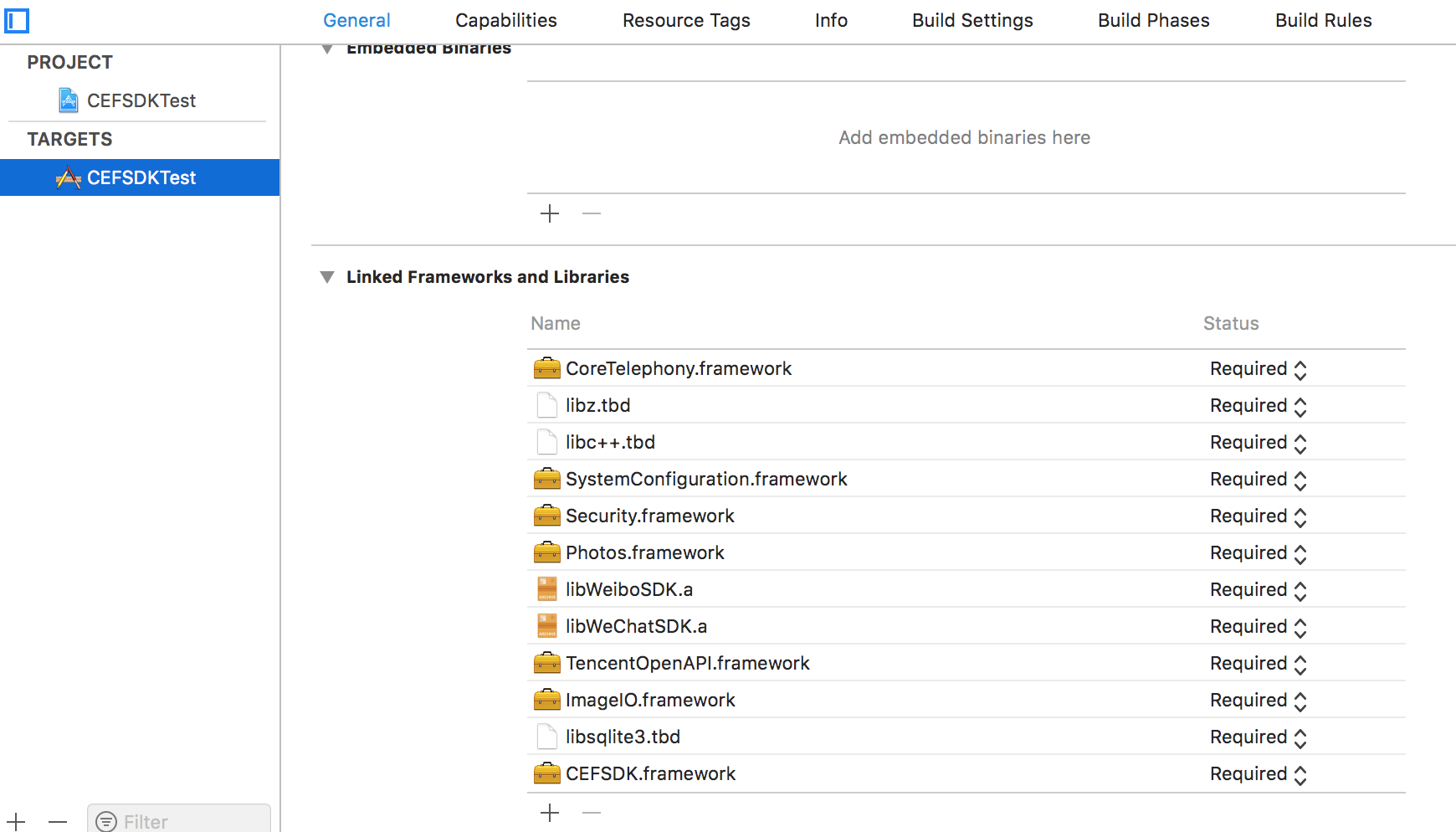
在 Xcode 7 中，苹果放宽了对开发者的限制，开发者无需注册开发者账号也可以使用真机进行调试了，只需要在 Xcode -> Preferences -> Accounts 里登陆自己的 Apple ID，在项目的 General选项卡，展开 Identity 选择 Personal Team，连上真机运行程序，如果有 Fix Issue 点击稍后即可成功运行。

2. 将下载的SDK解压后导入到工程中

将SDK放在项目文件夹里，打开Xcode的项目，直接拖进去就可以。注意拖进去的时候要选择create groups.



3. 添加依赖库

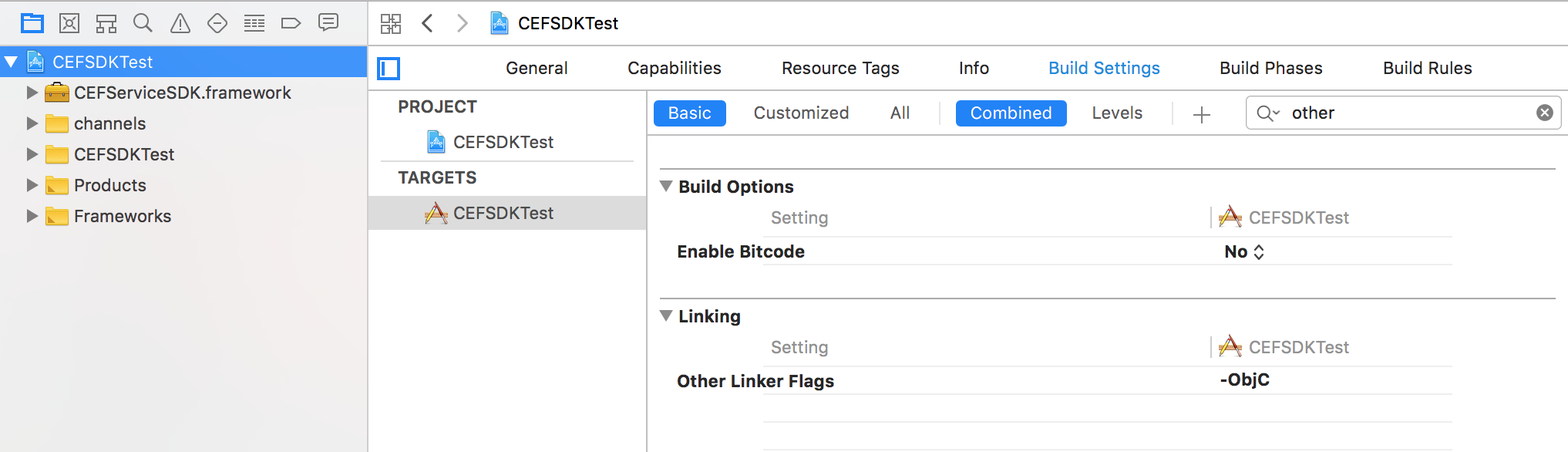


拖进去的第三方SDK和CEF SDK 会自动包含在依赖库里面，除此之外，还有以下一些系统的依赖库需要添加

|  |  |  |
| --- | --- | --- |
| Channel | 依赖库 |  |
| 新浪微博SDK | photos.framework  ImageIO.framework  Libsqlite3.tbd  SystemConfiguration.framework  Security.framework |  |
| 腾讯SDK | Libsqlite3.tbd  SystemConfiguration.framework  Security.framework |  |
| 微信SDK | libsqlite3.tbd  SystemConfiguration.framework  Security.framework  libc++.tbd  Libz.tbd  CoreTelePhony.framework |  |

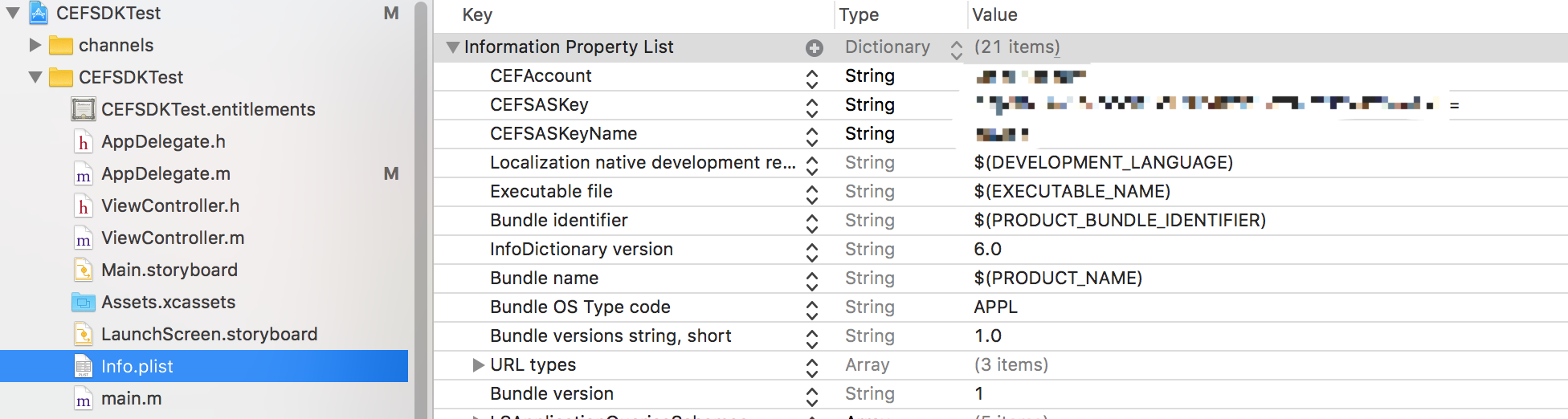
4. 配置编译设置

在项目的Build Settings中，把Enable Bitcode 设置为No, Other Linker Flags 添加 -ObjC.



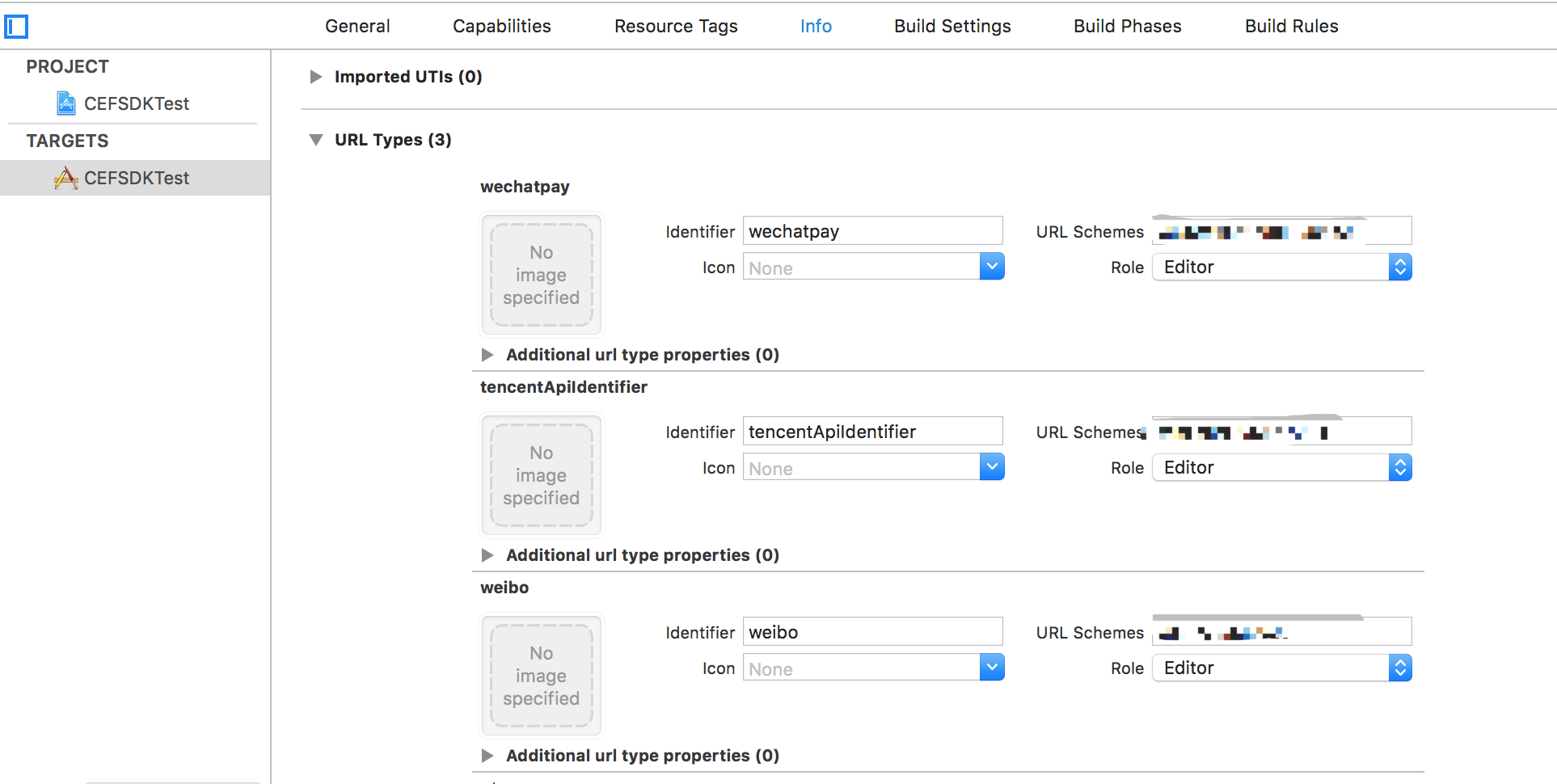
5. 设置CEF 账号信息

在项目的Info.plist中配置CEF的账号信息，包括CEFAccount, CEFSASKey 以及CEFSASKeyName



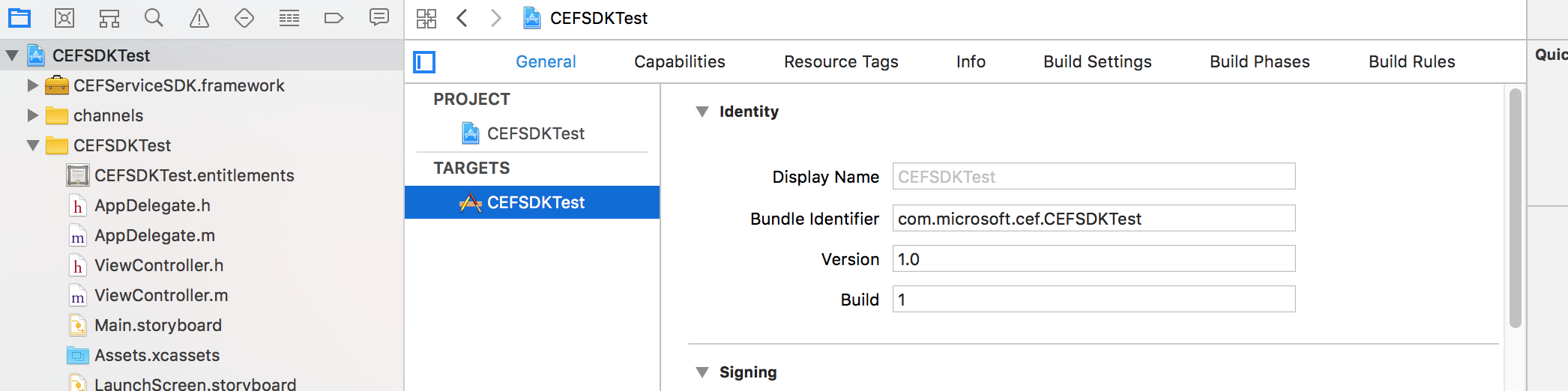
6. 配置第三方平台所对应的应用的信息

1） 在Info -> URL Types中，配置3个平台的indentifier和URLSchemes, 如下图所示：



|  |  |  |
| --- | --- | --- |
| SDK | Identifier | URL Schemes |
| 微博(WeiboSDK) | weibo | wb+APPKey |
| 腾讯(TencentOpenAPI) | tencentApiIdentifier | tencent+APPKey |
| 微信(WeChatOpenSDK) | wechat | APPKey |

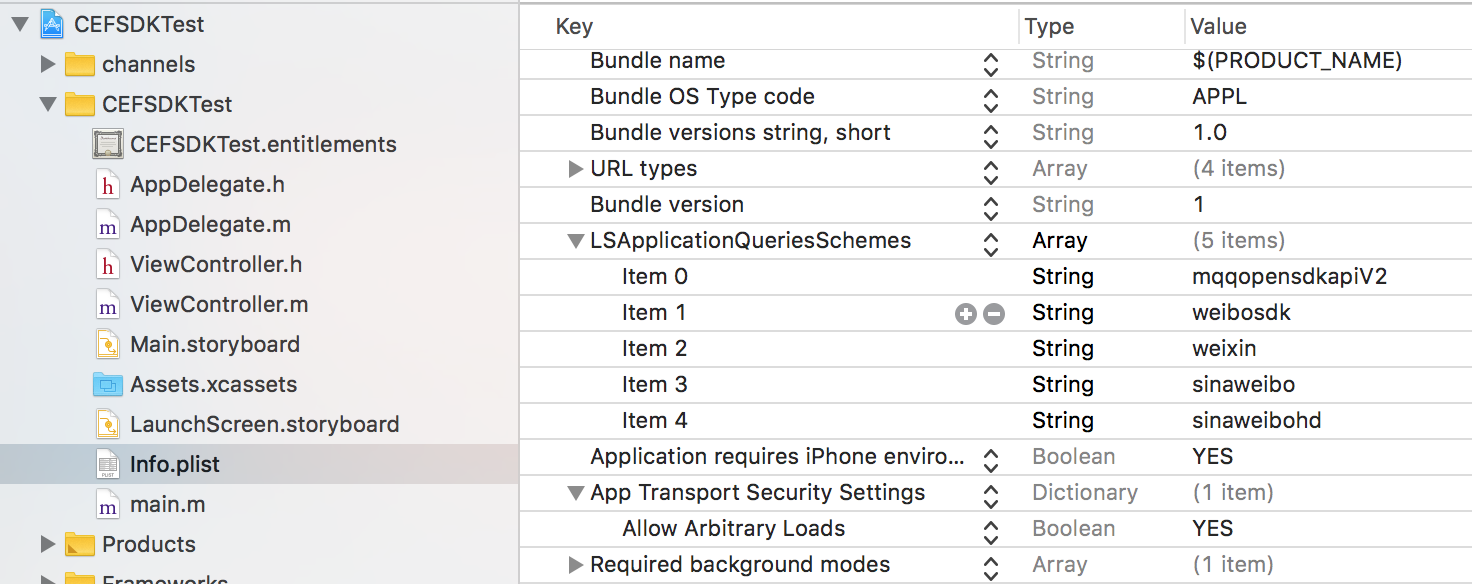
需要注意的是，有些平台的APP申请的时候，需要填App的Bundle ID, 这个创建的APP 的Bundle ID 要和申请时一致。如果使用我们的测试账号，这个Bundler ID必须为com.microsoft.cef.CEFSDKTest.



2) 配置对应平台的白名单

在项目Info.plist 文件中，添加LSApplicationQueriesSchemes, 类型为Array，然后给添加三个平台对应的白名单: mqqopensdkapiV2, weibosdk, weixin, sinaweibo,sinaweibohd.

App Transport Security Settings -> Allow Arbitrary Loads 的值，设置为YES。



# 验证码OTP

验证码，也称为OTP，广泛的应用于App账号验证，产品会员验证等场景，轻松排除了传统验证方式的弊端，大大提高了用户注册质量。

主要API：

生成验证码：

[CEFOTPManager generateOTPCode:phoneNumber templateName:@"OTPTemplate001" expireTime:300 codeLength:6 channel:CEFOTPChannel\_SMS completion:^(NSDictionary \*result, NSError \* \_Nullable error) {{}];

generateOTPCode参数说明：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| phoneNumber | NSString\* | 用户的手机号，目前只支持国内11位手机号码，+86和11位手机号，0086和11位手机号码 |
| templateName | NSString\* | 开发者自定义的模板名字，模板需要提前在服务器端设置好。 |
| expireTime | int | 验证码过期时间，以秒为单位。 |
| codeLength | int | 验证码长度 |
| channel | CEFOTPChannel，枚举 | 发送验证码的渠道，目前只支持SMS. |
| completion | 回调函数 | 请求失败,error不等于nil,包含有错误信息。  请求成功, result返回NSDictionay包含过期时间等信息 |

完成验证：

[CEFOTPManager verifyOTPCode:phoneNumber code:verifyCode channel:CEFOTPChannel\_SMS completion:^(CEFOTPResultCode result, NSError \* \_Nullable error){ }];

verifyOTPCode参数说明：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| phoneNumber | NSString\* | 用户的电话号码 |
| code | NSString\* | 验证码 |
| channel | CEFOTPChannel，枚举 | 发送验证码的渠道，目前只支持SMS. |
| completion | 回调函数 | 请求失败,error不等于nil,包含有错误信息。  请求成功, result返回CEFOTPResultCode,枚举类型,验证成功是SUCCESS, 失败是WRONGCODE,验证码过期是EXPIRED，并且每个验证码只能使用一次，验证过后再次验证时会失败。 |

# 社交登录验证

CEF提供统一的API帮助用户实现微信，QQ和微博登录，缩短开发者的开发时间。各个平台登录数据统一记到CEF后台，方便查阅和分析。

1. 在AppDelegate.m文件 didFinishLaunchingWithOptions, 初始化登录渠道

[CEFSocialLoginManager setupChannel:cred];

setupChannel参数说明：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| cred | CEFCredential | CEFCredential类需要先初始化相应的微信，QQ和微博等信息。CEF会根据CEFCredential 里面设置的平台信息来初始化相应的平台。如果没有设置或者没有设置完整的平台，就不会被初始化，后面也无法使用。 |

1. 在AppDelegate.m文件, openURL, 添加对打开App的处理如下:

- (BOOL)application:(UIApplication \*)app openURL:(NSURL \*)url options:(NSDictionary \*)options {

return [CEFSocialLoginManager handleOpenURL:url options:options];

}

- (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(nullable NSString \*)sourceApplication annotation:(id)annotation {

return [CEFSocialLoginManager handleOpenURL:url];

}

说明：

这个是由于iOS系统的要求，不在openURL里面添加相应的代码会导致从微信等第三方App跳转回本App时无法打开连接。

1. 在响应用户点击登录事件里调用 CEFSocialLoginManager loginWithChannel获得用户登录结果和平台上的用户信息。

[CEFSocialLoginManager loginWithChannel:CEFSocialLoginChannel\_WeChat completion:^(CEFResponseSocialLogin \*loginResult, CEFSocialLoginProfile \*socialProfile){}];

loginWithChannel参数说明：

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| channel | CEFSocialLoginChannel,枚举 | 用来说明登录的哪个第三方平台。 |
| completion | 回调函数 | 平台的登录结果loginResult，CEFResponseSocialLogin类型，  包含平台的登录结果，accessToken等  socialProfile, CEFSocialLoginProfile类型, 请求成功时包含平台上的用户信息，请求失败时等于nil. |