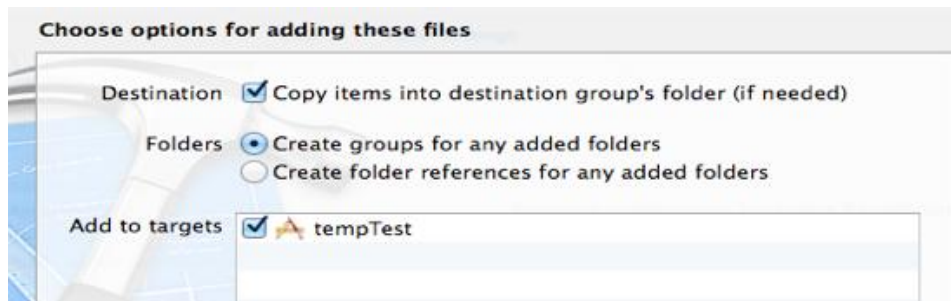


iOS-SDK接入文档之Appsflyer篇

引入 SDK 文件

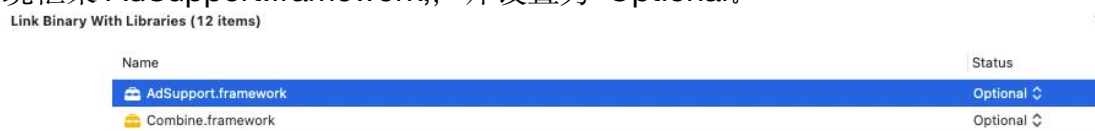
将 SDK 文件夹下 AppsFlyerLib.framework 文件引入到工程，并选择正确的 target



包含 **UnityFramework** 动态库特别说明:

AppsFlyerLib.framework 为静态库 TargetMembership 需要关联到 UnityFramework。

使用 **AppsFlyer** 需要在 Target > Build Phases > Link Binary With Libraries 加入系统框架 **AdSupport.framework**，并设置为 **Optional**。



如果引入 appflyer 报错，找不到头文件，请解压 SDK 内附带 appsflyersdk 压缩包并替换 appsflyerSDK 文件。

向 AppsFlyer 发送 SKAN 回帖副本

配置您的应用程序，将后退副本发送到 AppsFlyer。

要注册 AppsFlyer 端点:

1. 将 `NSAdvertisingAttributionReportEndpoint` 键添加到应用程序的 `info.plist`。
2. 将密钥的值设置为 `https://appsflyer-skadnetwork.com/`。

根据苹果的说法，您只能设置一个端点。收到的回帖副本可在[回帖副本报告](#)中找到。

SDK 内部已经投递事件(接入方无需再处理)

AppsflyerSDK内部已经投递安装、启动等事件，本SDK内已经投递的事件

1. 注册事件: AFEventCompleteRegistration;

2. 登录事件:AFEEventLogin;
3. 购买事件:AFEEventPurchase。

接口调用

引入 REDeLoginKit.h

```
#import <JYouLoginKit/REDeLoginKit.h>
```

初始化接口

****Appsflyer 的接入首先需要去 Appsflyer 后台去获取 devkey,iTunes connect 后台获取 AppleAppID, 然后调用我们初始化接口, 建议在应用系统代理启动方法 didFinishLaunchingWithOptions 中调用此方法初始化 Appsflyer. ****

```
/**
 * 初始化 appsflyer
 * @param devKey AppsFlyer devKey
 * @param appID itunes connect 分配的 appID
 * @param Sandbox 支付验证是否用沙盒模式, 不传默认为 NO(正式模式), 注意:
 * 出正式提审包时一定要设置为 NO
 * * @param currency 设置 AF 统计货币类型, 可传 nil 或空字符串则使用默认的
 * USD
 */
+ (void)setAppsFlyerDevKey:(NSString *)devKey appleAppID:(NSString *)appID useReceiptValidationSandbox:(BOOL)Sandbox currencyType:(NSString *)currency;
```

参数 devKey 在 AppsFlyer 后台, 参数 appID 是该应用的 iTunes ID 但不包含" id"。

代码示例:

```
[REDeLoginKit setAppsFlyerDevKey:@"xxx" AppID:@"xxxx" useReceiptValidationSandbox:YES];
```

投递自定义事件接口

```
/**
 *AppsFlyer 自定义事件接口 (选接)
 *
 * @param eventName 自定义事件名从 AppsFlyerLib.framework 的 `AppsFlyerLib.h` 的预定义常量中获取
 * @param values 参数字典 (可传空), key 值从 `AppsFlyerLib.h` 的预定义常量中获取
 * 调用事例:
 * [REDeLoginKit afTrackEvent:AFEEventPurchase
 *   withValues: @{AFEEventParamRevenue : @200,
 *                  AFEEventParamCurrency : @"USD",
 *                  AFEEventParamQuantity : @2,
 *                  AFEEventParamContentId: @"092",
 *                  AFEEventParamReceiptId: @"9277"}];
 */
```

```
+ (void)afTrackEvent:(NSString *)eventName withValues:(NSDictionary
*_Nullable)values;
```

应用跳转回调(必接)

类: REDeLoginKit

函数: +(void)application:(UIApplication *)application openURL:(NSURL *)url
options:(NSDictionary *)options;
功能: 处理第 3 方应用回调结果

在- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url
options:(NSDictionary<UIApplicationOpenURLOptionsKey,id> *)options; 中调用。
类: REDeLoginKit

函数: + (void)application:(UIApplication *)application openURL:(NSURL *)url
sourceApplication:(NSString *)source application:(id)annotation;
功能: 处理第 3 方应用回调结果

在- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url
sourceApplication:(NSString *)sourceApplication annotation:(id)annotation; 中调用。

示例:

```
- (BOOL)application:(UIApplication *)application openURL:(NSURL
*)url sourceApplication:(NSString*)sourceApplication
annotation:(id)annotation
{
    [REDeLoginKit application:application openURL:url
sourceApplication:sourceApplication annotation:annotation];
    return YES;
}
- (BOOL)application:(UIApplication *)application openURL:(NSURL
*)url options:(NSDictionary<UIApplicationOpenURLOptionsKey,id>
*)options {
    [REDeLoginKit application:application openURL:url
options:options];
    return YES;
}
```

配置 Appsflyer 是否投递登录流程事件(选接)

1.配置文件配置(优先级低于代码配置)

SSBundle.bundle/ColorStyle.plist 添加/修改键值对 logEvtReport:value(0 关闭 1 开启)

2.代码配置(优先级高于配置文件开启)

/** 配置 appsflyer 是否需要上报 SDK 内登录流程事件 */

+ (void)configNeedAppsFlyerLoginEvent:(BOOL)need;

启用 AppsFlyer 调试模式(选接)

您可以在初始化 AppsFlyer 前通过将 AppsFlyerLib 属性 isDebug 设置为 true 来启用调试日志:

```
[AppsFlyerLib shared].isDebug = true;
```

示例:

```
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary
*)launchOptions {
    [AppsFlyerLib shared].isDebug = true;
    [REDeLoginKit setAppsFlyerDevKey:@"xxxx"
appleAppID:@"xxxx" useReceiptValidationSandbox:NO
currencyType:@"USD"];
}
```

日志关键字[com.appsflyer.serial].