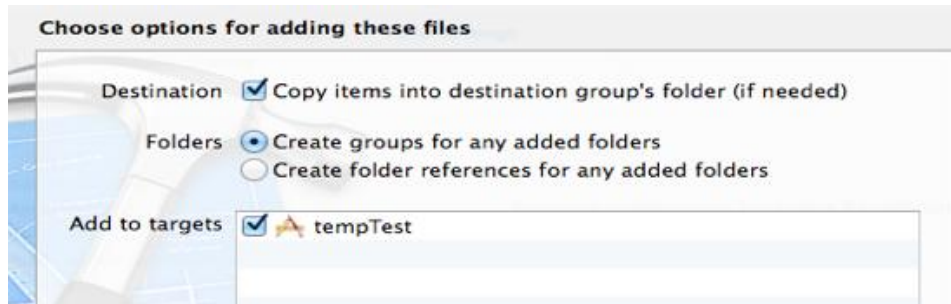


# iOS SDK Document for Appsflyer

## Import SDK files

Import the AppsFlyerLib.framework file under the SDK folder into the project, and select the correct target



**For the xcode project exported using Unity version 2019.3 and later, it contains the UnityFramework dynamic library. When importing the SDK file, you need to pay attention to:**

AppsFlyerLib.framework is a static library TargetMembership needs to be linked to UnityFramework.

Xcode Need add : Target > Build Phases > Link Binary With Libraries add system framework: **AdSupport.framework** and set as **Optional**.



If an error is reported when appflyer is introduced and the header file cannot be found, please unzip the appsflyersdk compressed package included in the SDK and replace the appsflyerSDK file.

The event has been delivered inside the SDK (the access party does not need to process it)

Events such as installation and startup have been delivered within AppsflyerSDK, and events have been delivered in this SDK

1. Register for events: AFEventCompleteRegistration;
2. Login event: AFEventLogin;
3. Purchase event: AFEventPurchase.

## Interface call

```
import REDeLoginKit.h
```

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

## Initialize the interface

**\*\*** To access AppsFlyer, you first need to go to the AppsFlyer background to obtain the APPID and devkey, and then call our initialization interface. It is recommended to call this method to initialize AppsFlyer when initializing the SDK.**\*\***

```
/**
 * init appsflyer
 * @param devKey AppsFlyer devKey
 * @param appID AppsFlyer appID
 * @param Sandbox Whether to use the sandbox mode for payment
verification, the default is NO (official mode) if it is not
passed, Note: It must be set to NO when the package is officially
submitted for review
 * * @param currency Set the AF statistical currency type, you
can pass nil or an empty string to use the default USD
 */
+ (void)setAppsFlyerDevKey:(NSString *)devKey
appleAppID:(NSString *)appID
useReceiptValidationSandbox:(BOOL)Sandbox currencyType:(NSString
*)currency;
```

The parameter devKey is in the AppsFlyer background, and the parameter appID is the iTunes ID of the application but does not contain "id".

Code example:

```
[REDeLoginKit setAppsFlyerDevKey:@"xxx" appleAppID:@"xxxx"
useReceiptValidationSandbox:YES currencyType:@"USD"];
```

## Delivery custom event interface

```
/**
 *AppsFlyer Custom event interface (optional)
 *
 * @param eventName The custom event name is obtained from the
predefined constants of `AppsFlyerLib.h` in
AppsFlyerLib.framework
 * @param values Parameter dictionary (can be passed empty), the
key value is obtained from the predefined constants of
`AppsFlyerLib.h`
```

```

Call example:
[REDeLoginKit afTrackEvent:AFEEventPurchase
    withValues: @{AFEEventParamRevenue : @200,
                  AFEEventParamCurrency : @"USD",
                  AFEEventParamQuantity : @2,
                  AFEEventParamContentId: @"092",
                  AFEEventParamReceiptId: @"9277"}];
*/
+ (void)afTrackEvent:(NSString *)eventName
withValues:(NSDictionary * _Nullable)values;

```

## Application jump callback (required)

Class: [REDeLoginKit](#)

Function: Process the callback result of the third party application

```

+ (void)application:(UIApplication *)application openURL:(NSURL
*)url options:(NSDictionary *)options;
+ (void)application:(UIApplication *)application openURL:(NSURL
*)url sourceApplication:(NSString *)source
annotation:(id)annotation;

```

The above methods need to be called in the following system callback methods:

```

- (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;
}

```

## Configure whether Appsflyer drops login flow events (optional)

1. Configuration file configuration (lower priority than code configuration)  
SSBundle.bundle/ColorStyle.plist add/modify key-value pair logEvtReport:value(0 off 1 on)
2. code configuration (priority higher than the configuration file to open)  
/\*\* Configure whether appsflyer needs to report the login process events in SDK \*/  
+ (void)configNeedAppsflyerLoginEvent:(BOOL)need;