

# iOS SDK Document for Facebook share

## Import SDK files

/\*\*

FBSDK13.0.0 and above no longer support Xcode12, you must use xcode13.2.1 or above.

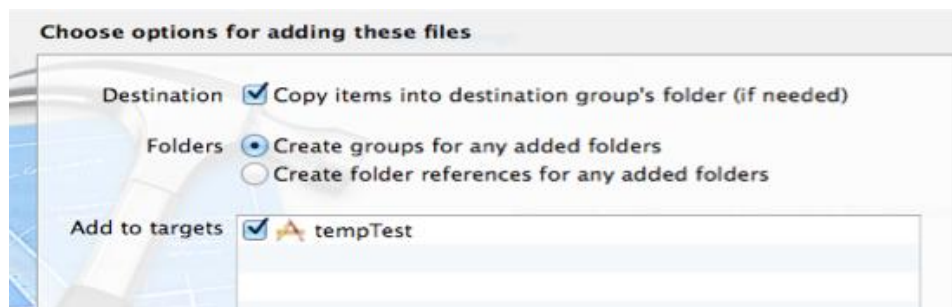
FBSDK15.1.0 and above no longer support Xcode13, you must use xcode14 or above.

FBSDK17.0.0 and above no longer support Xcode14, you must use xcode15.0 or above(FBSDK17.0.2 required xcode15.3+).

\*\*/

Import the FBSDKCoreKit.framework, FBAEMKit.framework, FBSDKCoreKit\_Basics.framework, FBSDKShareKit.framework files in the SDK folder into the access project and select the correct target.

(If you have access to Facebook login or statistics, the same file can be ignored and only a different framework file needs to be imported.)



FBSDK has changed to dynamic library version since 17.3.0, please load it as dynamic library, i.e. set FBxxxx.framework to Embed & Sign.

Check if the target->Build Setting-> Runpath search path contains @executable\_path/Frameworks, if it doesn't, please add it, if it does, please continue to the next step.

**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:**

FBSDKCoreKit.framework and FBAEMKit.framework and FBSDKCoreKit\_Basics.framework and FBSDKShareKit.framework are dynamic libraries, and TargetMembership needs to be linked to UnityFramework and Unity-iPhone at the same time.

UnityFramework only needs to be associated, it doesn't need to be and can't be set to Embed & Sign, otherwise it will report an error when submitting to the AppStore.

Unity-iPhone not only needs to be associated, it needs to be and must be set to Embed & Sign, otherwise it will crash on startup.

## Xcode configuration

FBSDK needs access to engineering support for Swift and OC mashups. You can choose one of the following 2 options

Option 1: You need to create a swift file in your project, and then follow the Xcode prompts to create an OC and swift bridge file.

Option 2: Go to Project->target->BuildSettings -> Search path->Library Search Paths, and add the code:

```
$(SDKROOT)/usr/lib/swift
```

```
$(TOOLCHAIN_DIR)/usr/lib/swift/$(PLATFORM_NAME)
```

Then Project->target->BuildSettings -> Linking -> Runpath search path and add the following code:

```
//needs to be first on the list
```

```
/usr/lib/swift
```

**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:**

All operations are performed on UnityFramework.

Option 1: A swift file needs to be created in the project, which can be associated to UnityFramework only or to both UnityFramework and Untiy-iphone.

Option 2: Go to Project->UnityFramework->BuildSettings -> Search path->Library Search Paths, and add the code:

```
$(SDKROOT)/usr/lib/swift
```

```
$(TOOLCHAIN_DIR)/usr/lib/swift/$(PLATFORM_NAME)
```

Then Project->UnityFramework->BuildSettings -> Linking -> Runpath search path and add the following code:

```
//needs to be first on the list
```

```
/usr/lib/swift
```

FacebookAppID: application parameters in the FB background;

FacebookDisplayNam: The name of the application configured in the FB background {game name};

FacebookClientToken: Check the Facebook background application settings->Advanced->Client password;

LSApplicationQueriesSchemes: Facebook related whitelist.

✓ LSAApplicationQueriesSchemes	⇅	Array	(5 items)	
Item 0		String	fbapi	
Item 1		String	fb-messenger-api	
Item 2		String	fbauth2	
Item 3		String	fbshareextension	
Item 4		String	fb-messenger-share-api	
DTSDKBuild	⇅	String	19A339	
FacebookAdvertiserIDCollectionEnabled	⇅	Boolean	1	
Bundle version string (short)	⇅	String	1.0.0	
> CFBundleSupportedPlatforms	⇅	Array	(1 item)	
> Supported interface orientations	⇅	Array	(1 item)	
BuildMachineOSBuild	⇅	String	21A559	
DTPlatformBuild	⇅	String	19A339	
Bundle OS Type code	⇅	String	APPL	
DTXcodeBuild	⇅	String	13A1030d	
Localization native development region	⇅	String	English	
MinimumOSVersion	⇅	String	9.0	
Bundle version	⇅	String	3	
Icon already includes gloss effects	⇅	Boolean	YES	
Status bar is initially hidden	⇅	Boolean	YES	
FacebookAppID	⇅	String	{{FacebookAppID}}	替换成游戏的fbappid
> UIDeviceFamily	⇅	Array	(2 items)	
Launch screen interface file base name	⇅	String	LaunchScreen	
Bundle identifier	⇅	String	com.firefantasyxx.ios	
FacebookDisplayName	⇅	String	{{FacebookDisplayName}}	游戏的fb应用名
DTXcode	⇅	String	1310	
FacebookClientToken	⇅	String	{{FacebookClientToken}}	游戏的fbclienttoken

## URL Types

Document Types (0)

Exported UTIs (0)

Imported UTIs (0)

URL Types (1)

Untitled

Identifier: None

Icon: None

URL Schemes: fb1836507973298860

Role: Editor

fb追加FacebookAppID

Additional url type properties (0)

+

## Interface call

```
import REDeLoginKit.h
```

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

## Facebook Like and share

```
/**
 * FB sharing of sdk server configuration
 */
+ (void)shareWith:(NSString *)serverId roleId:(NSString *)roleId;
```

Function: Enter the user center page to display information and operation portals associated with other platform accounts.

## Facebook Native image sharing

To use FB native picture sharing, you need to configure the parameters in the project info.plist. For the specific configuration, refer to the Facebook login section.

### 1.1 share picture

```
/**
 * FB image sharing
 * image image to be shared
 * caption Create a title for the shared photo. If you pass nil,
there is no title. The FB platform does not allow you to pre-fill
in the shared content. If you fill in here, it will not be
displayed on FB.
 * completeResults
    share success status=0, error = nil
    share fail status=1, error != nil
    cancel share status=2, error = nil
 * tip:
    1. Photo size must be less than 12MB
    2. Users need to install the native iOS version of the Facebook
app with version 7.0 or above
 */
+ (void)fbSharePhoto:(UIImage *)image caption:(NSString *)caption
tag:(NSString *)tag completeWithResults:(void (^)(NSError
*error, NSInteger status))completeResults;
```

Function: Share pictures to FB App.

\* tip:

1. The photo size must be less than 12MB and cannot be named in Chinese
2. The user needs to install the native iOS version of the Facebook app with version 7.0 or above

Code example:

```
[REDeLoginKit fbSharePhoto:@"1080-1920.jpg" caption:@"133"
completeWithResults:^(NSError *error, NSInteger status) {
    if (status == 0) {
        NSLog(@"Successful sharing");
    }else if(status == 1){
        NSLog(@"Sharing failed %@",error.description);
    }else{
        NSLog(@"User canceled sharing");
    }
}
}];
```

Facebook link image sharing

```
/**
 * FB image sharing
 * imageUrl image link to be shared
 * caption Create a title for the shared photo. If you pass nil,
there is no title. The FB platform does not allow you to pre-fill
in the shared content. If you fill in here, it will not be
displayed on FB.
 * completeResults
    share success status=0, error = nil
    share fail status=1, error != nil
    cancel share status=2, error = nil
 * tip:
    1. Users need to install the native iOS version of the Facebook
app with version 7.0 or above
 */
+ (void)fbShareUrlPhoto:(NSString *)imageUrl caption:(NSString
*)caption tag:(NSString *)tag completeWithResults:(void(^)(NSError
*error,NSInteger status))completeResults;
```

Function: Share link image to FB App.

\*tip:

The user needs to install the native iOS version of the Facebook app with version 7.0 or above

## Facebook URL sharing

To use FB url sharing, you need to configure the parameters in the project info.plist. For the specific configuration, refer to the Facebook login section.

```
/**
 * FB url share
 * urlString url link to share
 * quote Highlighted text, displayed with the link, such as a
prominent citation in an article, can be empty
 * tag The hashtag, which can be empty, will be displayed in the
share dialog, so the user can decide whether to delete it before
posting
 * completeResults
    share success status=0, error = nil
    share fail status=1, error != nil
    cancel share status=2, error = nil
 */
```

```
+ (void)fbShareUrl:(NSString *)urlString quote:(NSString *)quote
tag:(NSString *)tag completeWithResults:(void (^)(NSError
*error, NSInteger status))completeResults;
```

Function: Share url link to FB App.

Facebook Video sharing

Function: Share video to FB App.

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 system callback methods.

For example:

```
- (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;
}
```