**MerculetSDK**

**使用文档（iOS）**

目录

一 快速集成 3

1.1 获取appKey、accountKey、accountSecret 3

1.2 导入SDK 3

1.2.1 使用Cocoapods安装SDK 3

1.2.2 下载SDK并集成 3

1.3 初始化SDK 5

二 自定义事件统计 6

2.1 自定义事件统计 6

三 注意事项 6

3.1 支持ATS 6

# 快速集成

## 获取appKey、accountKey、accountSecret

登录官网（http://merculet.io/），按照步骤提示注册应用，可获得appKey、accountKey和accountSecret。

## 导入SDK

导入SDK有以下两种方法，第一种使用Cocoapods安装SDK，第二种直接下载SDK并集成，选择其中一种即可。

### 使用Cocoapods安装SDK

使用Cocoapods可以方便的统一管理第三方库：https://cocoapods.org

安装Cocoapods，并在项目根目录下创建Podfile文件，在Podfile文件中添加如下内容：

pod 'MerculetSDK'

//如果使用bitcode，使用pod 'MerculetSDKBitcode'

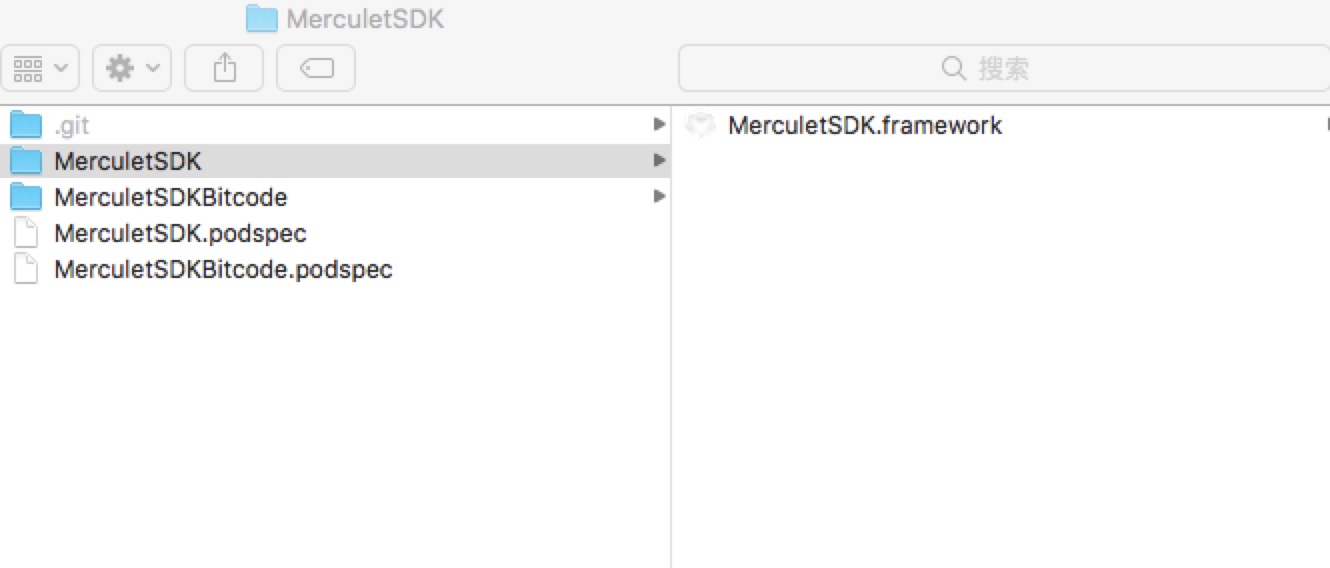
在terminal下运行命令如下：

pod install

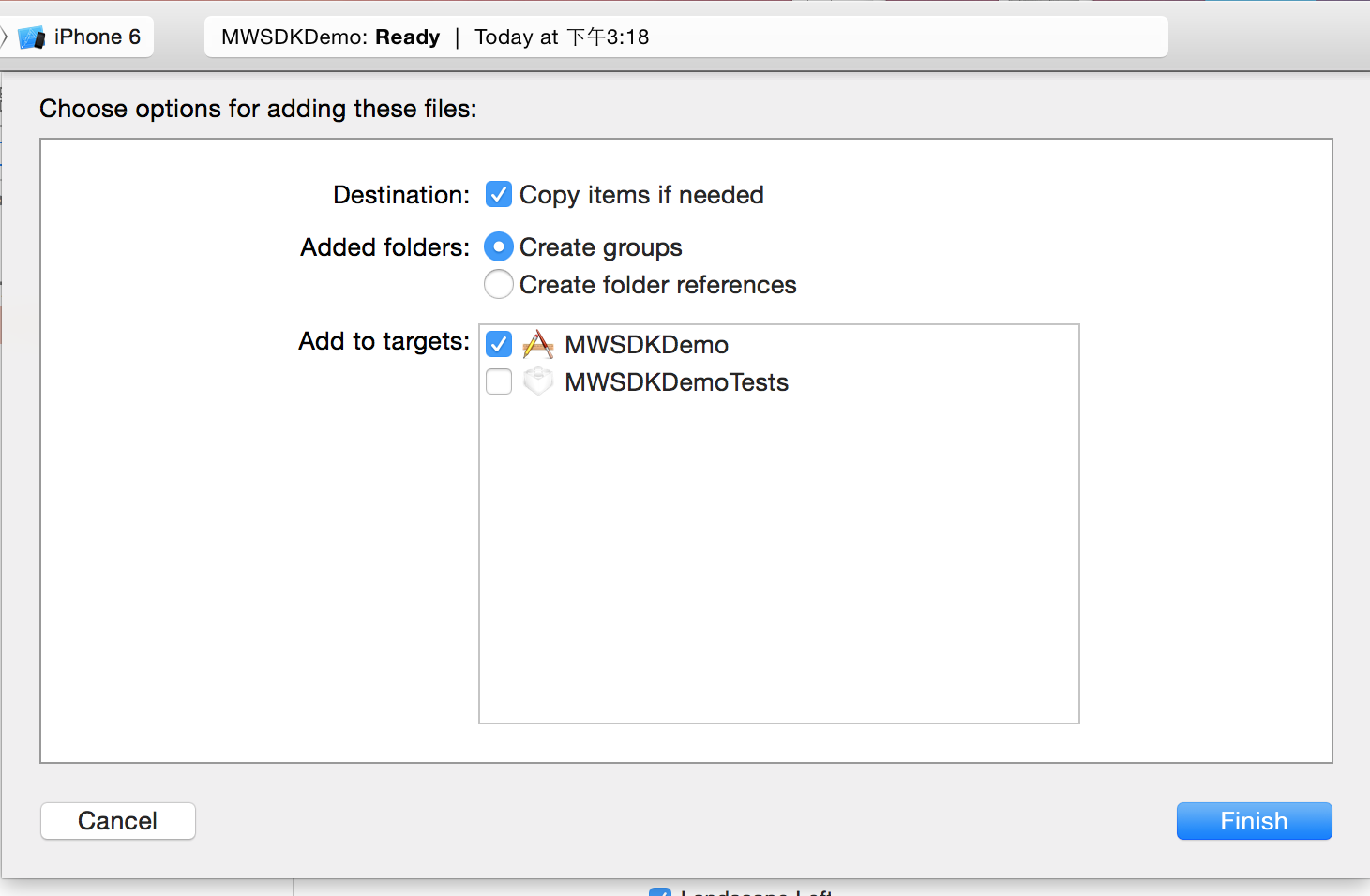
### 下载SDK并集成

1. **下载SDK并集成**

下载并解压最新版本SDK压缩包，解压后如下图：



拖动.framework文件夹到工程中，拖到工程中后，弹出以下对话框，勾选“**Copy items into destination group's folder(if needed)**”，并点击“**Finish**”按钮，如图：



注意：请务必在上述步骤中选择“**Create groups for any added folders**”单选按钮组。如果您选择“Create folder references for any added folders”，一个蓝色的文件夹引用将被添加到项目并且将无法找到它的资源。

1. **添加依赖库**

如果使用了Cocoapods集成的SDK，可以忽略此步骤

AdSupport.framework

CoreGraphics.framework

CoreFoundation.framework

CFNetwork.framework

Security.framework

libz.tbd

libsqlite3.0.tbd

libc++.tbd

## 初始化SDK

在AppDelegate中，增加头文件的引用

#import <MerculetSDK/MWApi.h>

在- (BOOL)application: didFinishLaunchingWithOptions:方法中调用registerApp方法来初始化SDK，如下图所示：

#import "AppDelegate.h"

#import <MerculetSDK/MWApi.h>

@implementation AppDelegate

- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions

{

//如果您创建应用时使用storyBoard可以省略此步骤

self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];

self.window.rootViewController = viewController;

[self.window makeKeyAndVisible];

**//初始化SDK，必写**

**NSString \* appkey = @"fb61da1dc41f4e1f8d978a9e7547edda";**

**NSString \* accountKey = @"4a0fa34013cc4ffba60c240ad7afe453";**

**NSString \* accountSecret = @"5838f70dbe2e4044a3fffd5d709f23c2";**

**[MWAPI registerApp:appkey accountKey:accountKey accountSecret:accountSecret];**

return YES;

}

注意：必须设置rootViewController，否则会导致无法弹出活动界面。如果您创建应用时使用storyBoard可以省略此步骤，系统会自动设置rootViewController。

# 自定义事件统计

## 自定义事件统计

**传入用户的userId（必填项）和邀请码（选填项）**

NSString \*userId = @"XXX";// 必填

NSString \*invitation = @"XXX";// 选填

[MWAPI setUserOpenId:userId invitationCode:invitation];

[MWAPI cancelUserOpenId];

注意：由于SDK是跟踪用户的行为的，所以确保在传入了userId，否则SDK不会收集任何信息。

**取消对用户的追踪**

[MWAPI cancelUserOpenId];

注意：不能将userOpenId直接设置成nil或空字符串。

**统计指定行为**

[MWAPI setCustomAction:@"eventName" attributes:@{@"key1":@"value1",@"key2":@"value2"…}];

注意：eventName和KeyValue参数需要先在后台管理上注册，才能参与正常的数据统计

# 注意事项

## 支持ATS

苹果：2017 年1月1日后所有iOS应用必须启用ATS。  
SDK已启用ATS。