

阿里云

移动用户反馈 开发指南

文档版本：20211207

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1.Android SDK接入	05
2.iOS SDK接入	11

1.Android SDK接入

1 SDK下载

如果使用Maven库方式不需要下载，方法见2.1中Maven库快速集成（远程同步）

如果要手动集成，进入[EMAS控制台](#)下载SDK。

2 准备工作

2.1 Android Studio开发

1、Maven库快速集成（远程同步）（推荐方式）

在 `Project` 根目录下 `build.gradle` 文件中配置Maven库URL：

```
allprojects {
    repositories {
        jcenter()
        maven {
            url 'http://maven.aliyun.com/nexus/content/repositories/releases/'
        }
    }
}
```

在对应的 `module` 下的 `build.gradle` 文件中添加对应依赖：

```
dependencies {
    .....
    compile 'com.aliyun.ams:alicloud-android-feedback:3.3.7'
    .....
}
```

2、手动集成

复制下载SDK文件夹下的 `.aar` 和 `.jar` 到libs目录下，在 `build.gradle` 文件中添加如下配置：

```
repositories {
    flatDir {
        dirs 'libs'
    }
}
```

添加本地依赖：

```
dependencies {
    compile(name: 'alicloud-android-feedback-3.3.7', ext: 'aar')
    compile(name: 'alicloud-android-rest-1.6.0-open', ext: 'aar')
    compile(name: 'alicloud-android-sender-1.1.2', ext: 'aar')
    compile(name: 'alicloud-android-logger-1.0.1', ext: 'aar')
    compile files('libs/alicloud-android-utdid-2.5.2.jar')
    compile files('libs/alicloud-android-beacon-1.0.7.jar')
    compile files('libs/alicloud-android-crashdefend-0.0.5.jar')
    //自行依赖support-v4
}
```

注意

切记添加support-v4的依赖。

如果遇到UT DID冲突可参考：《[阿里云-移动云产品SDK UT DID冲突解决方案](#)》。

2.2 权限

- 需要相册、相机和录音的权限。
- Android 6.0需要开发者自己的方式实现权限获取，更低版本的使用声明即可。
- 如果用户进行需要权限的操作时，用户还未授权应用相关权限，SDK会直接向用户申请相关权限。应用可以通过如下方式，注册回调监听，在SDK申请权限之前，弹出提示窗，说明权限用途。

```

// 监听拍照操作 权限申请
FeedbackAPI.setPermissionInterrupt(FeedbackAPI.ACTION_CAMERA, new IPermissionRequestInterrupt() {
    @Override
    public void interrupt(Context context, String action, String[] permissions, InterruptCallback callback) {
        showDialog(context, "相机", "拍照问题进行反馈", callback);
    }
});
// 监听相册读取操作 权限申请
FeedbackAPI.setPermissionInterrupt(FeedbackAPI.ACTION_ALBUM, new IPermissionRequestInterrupt() {
    @Override
    public void interrupt(Context context, String action, String[] permissions, InterruptCallback callback) {
        showDialog(context, "相册", "选择问题照片进行反馈", callback);
    }
});
// 监听录音操作 权限申请
FeedbackAPI.setPermissionInterrupt(FeedbackAPI.ACTION_AUDIO, new IPermissionRequestInterrupt() {
    @Override
    public void interrupt(Context context, String action, String[] permissions, InterruptCallback callback) {
        showDialog(context, "录音", "录制语音描述进行反馈", callback);
    }
});

// 这里是示意代码，具体请替换为应用统一的权限说明样式
private void showDialog(Context context, String permission, String message, final InterruptCallback callback) {
    final AlertDialog.Builder normalDialog = new AlertDialog.Builder(context);
    normalDialog.setTitle("即将进行敏感权限授权");
    normalDialog.setMessage(permission + "权限作用：" + message);
    normalDialog.setPositiveButton("确定",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                // 用户同意申请权限，执行权限申请操作
                callback.goOnRequest();
            }
        });
    normalDialog.setNegativeButton("取消",
        new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                // 用户不同意申请权限，停止权限申请操作
                callback.stopRequest();
            }
        });
    // 显示
    normalDialog.show();
}

```

2.3 混淆配置

应用进行混淆打包时，应加入如下混淆配置：

```
-keep class com.taobao.** {*;}
-keep class com.alibaba.** {*;}
-dontwarn com.taobao.**
-dontwarn com.alibaba.**
-keep class com.ut.** {*;}
-dontwarn com.ut.**
-keep class com.ta.** {*;}
-dontwarn com.ta.**
```

3 开始使用

3.1 反馈组件初始化

```
FeedbackAPI.init(application, appkey, appSecret);
```

参数说明：

- application：Application对象。
- appkey：阿里云上应用的唯一标识，阿里云控制台可查询到。
- appSecret：阿里云上应用的appSecret，阿里云控制台可查询到。

注意

- 为避免在日志中泄漏参数 `appkey` / `appsecret` 或App运行过程中产生的数据，建议线上版本关闭SDK调试日志。
- 由于所有用户使用统一的SDK接入，在接入过程中需要在代码中设置 `appkey` / `appsecret` 参数，而此类参数与计量计费密切相关，为防止恶意反编译获取参数造成信息泄漏，建议您开启混淆，并进行App加固后再发布上线。

3.2 打开用户反馈界面

另起一个Activity打开用户反馈H5界面：

```
FeedbackAPI.openFeedbackActivity();
```

如果需要调用结果（例如显示加载中）：

```
FeedbackAPI.openFeedbackActivity(final Callable success, final Callable fail)
```

如果希望使用Fragment方式打开：

```
FragmentManager fm = getSupportFragmentManager();
final FragmentTransaction transaction = fm.beginTransaction();
final Fragment feedback = FeedbackAPI.getFeedbackFragment();
// must be called
FeedbackAPI.setFeedbackFragment(new Callable() {
    @Override
    public Object call() throws Exception {
        transaction.replace(R.id.content, feedback);
        transaction.commit();
        return null;
    }
}/*success callback*/, null/*fail callback*/);
```

说明

fragment的使用务必使用上述 `setFeedbackFragment` 的回调在xml中声明fragment那种方式，暂不支持包装fragment的activity；务必调用 `FeedbackAPI.cleanFeedbackFragment()`；此时的Fragment是 support-v4包中的Fragment。

3.3 获取反馈未读数

```
FeedbackAPI.getFeedbackUnreadCount(IUnreadCountCallback callback)
```

参数callback: `onSuccess(int count)` 回调方法count指示当前未读消息数。

3.4 设置标题栏中“历史反馈”的字号

```
FeedbackAPI.setHistoryTextSize(20); //单位为sp
```

说明

3.1.0版本中需要将控制台中历史反馈字号设置为0，此方法才会生效。

3.5 设置返回按钮的图片

```
FeedbackAPI.setBackIcon(R.drawable.back);
```

3.6 沉浸式任务栏

默认使用沉浸式任务栏，如需要关闭可以使用 `FeedbackAPI.setTranslucent(false)`;

说明

3.1.0版本中需要将控制台中沉浸式任务栏设置为true，此方法才会生效。

3.7 设置默认联系方式

```
FeedbackAPI.setDefaultUserContactInfo( "1300000XXXX" );
```

3.8 用户昵称

- 设置用户昵称后，用户反馈信息时携带该昵称，最终在控制台反馈会话中可查看。
- v3.1.7新增接口。

```
FeedbackAPI.setUserNick("xxx");
```

3.9 语音反馈

需要使用3.1.0以上版本的SDK，并且在控制台>系统配置>Mobile客户端配置>全局配置中将是是否开启语音打开。

3.10 设置反馈消息自定义参数

```
JSONObject extInfo = new JSONObject();  
extInfo.put("key", "value");  
FeedbackAPI.setAppExtInfo(extInfo);
```

4 Demo

[Demo地址](#)

说明

Demo工程中，`FeedbackAPI.init` 被调用了多次，这是为了能够达到实时更新配置的效果，实际使用上建议只在Application中，调用 `FeedbackAPI.init` 方法一次即可。

2.iOS SDK接入

在这个集成教程中，我们使用已创建的Demo应用，向您展示如何快速地集成用户反馈组件功能。

前提条件

在这个集成教程中，我们使用已创建的Demo应用，向您展示如何快速地集成用户反馈组件功能。

一般地，您需要入驻成为开发者并且创建应用后，才能集成反馈功能并且发布您的App。

注意

- 您完全可以先跳过前置准备，使用我们预先申请的Demo应用，来完成快速集成。
- 但是在您正式发布应用之前，您务必将一些内容替换为您所申请应用的特定内容。对于这些需要替换的内容，会在教程中以注意点的形式标识出。
- 用户反馈SDK支持的系统最低版本为iOS7。
- 用户反馈SDK支持 bitcode（bitcode仅在 Xcode7及以上显示并默认开启）。
- 用户反馈SDK及其依赖库均使用HTTPS协议进行加密传输。

Step1 下载SDK包

SDK下载地址：<https://emas.console.aliyun.com/#/sdkDownload>

如果您已经获得Demo包，则可以跳过这一步。

Demo地址，打开[Demo下载页面](#)

Step2 集成SDK

下载SDK并集成

- 将第一步中下载的文件，直接拖入你的Xcode工程中。参照[Demo](#)中的做法。
- 在弹出的对话框中勾选Copy items if needed后，点击Finish。

如果遇到 UT DID 冲突可参考：《[阿里云-移动云产品SDK UT DID冲突解决方案](#)》。

注意

确认目的目录中包含 `YWFeedbackBundle.bundle`，以免造成返回箭头丢失的问题。

使用 Cocoapods 安装 SDK

[CocoaPods](#) 安装完成后，还需要在你的项目根目录创建一个Podfile文件，添加如下内容：

```
# 指定 Master 仓库和阿里云仓库
source 'https://github.com/CocoaPods/Specs.git'
source 'https://github.com/aliyun/aliyun-specs.git'

# 添加依赖
target 'YourTargetName' do
  pod 'AlicloudFeedback', '~> 3.2.0'
end
```

(~>为模糊指定版本号方式，~> 3.2.0 表明引用版本位于3.2.0 <= version < 3.3.0之间的最新版本SDK，用户可参考 [Podfile Syntax Reference](#)，根据项目需要指定SDK版本。)

在terminal下运行命令如下：

```
pod install
```

已安装旧版的，请运行（不要用-no-repo-update选项）：

```
pod update
```

如果安装时发现无法找到对应的库，请首先尝试到[CocoaPods](#)上搜索，如果能够搜索到对应的版本，那就是本地环境的问题，可以尝试运行如下命令：

```
pod repo update
```

然后再尝试安装。

命令执行完成后即可完成用户反馈SDK的下载安装。

注意

运行demo前请在 `YWLoginController.m` 文件中填入您的账号信息：

```
static NSString * const kAppKey = @"*****";
static NSString * const kAppSecret = @"*****";
```

Step3 添加依赖系统库

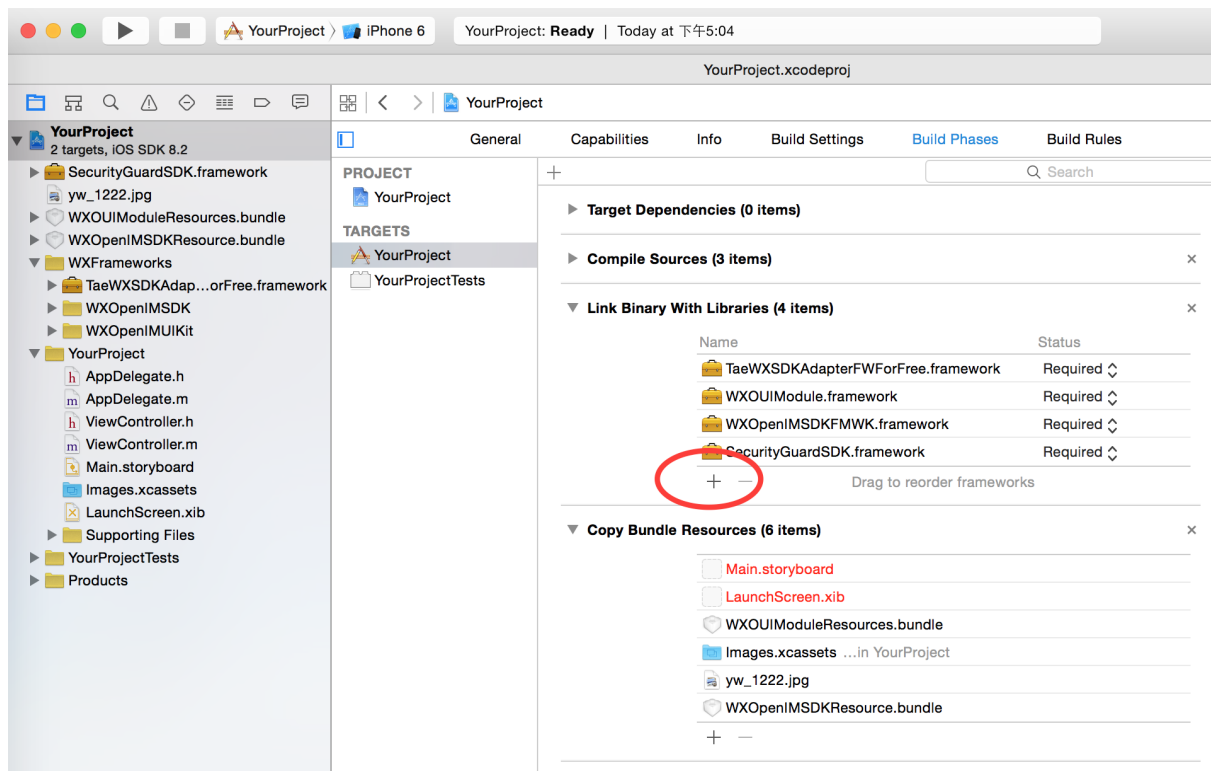
YWFeedbackKit依赖于这些系统库，您需要将他们添加到工程中：

- 系统公共库：
 - libz.tbd
 - libresolv.tbd
 - libsqlite3.tbd
 - CoreMotion.framework
 - CoreTelephony.framework
 - SystemConfiguration.framework

注意

- Cocoapods集成方式不需要手动添加依赖系统库。
- 在Xcode7之前的版本, 没有 `.tbd` 的库, 而是 `.dylib` 。

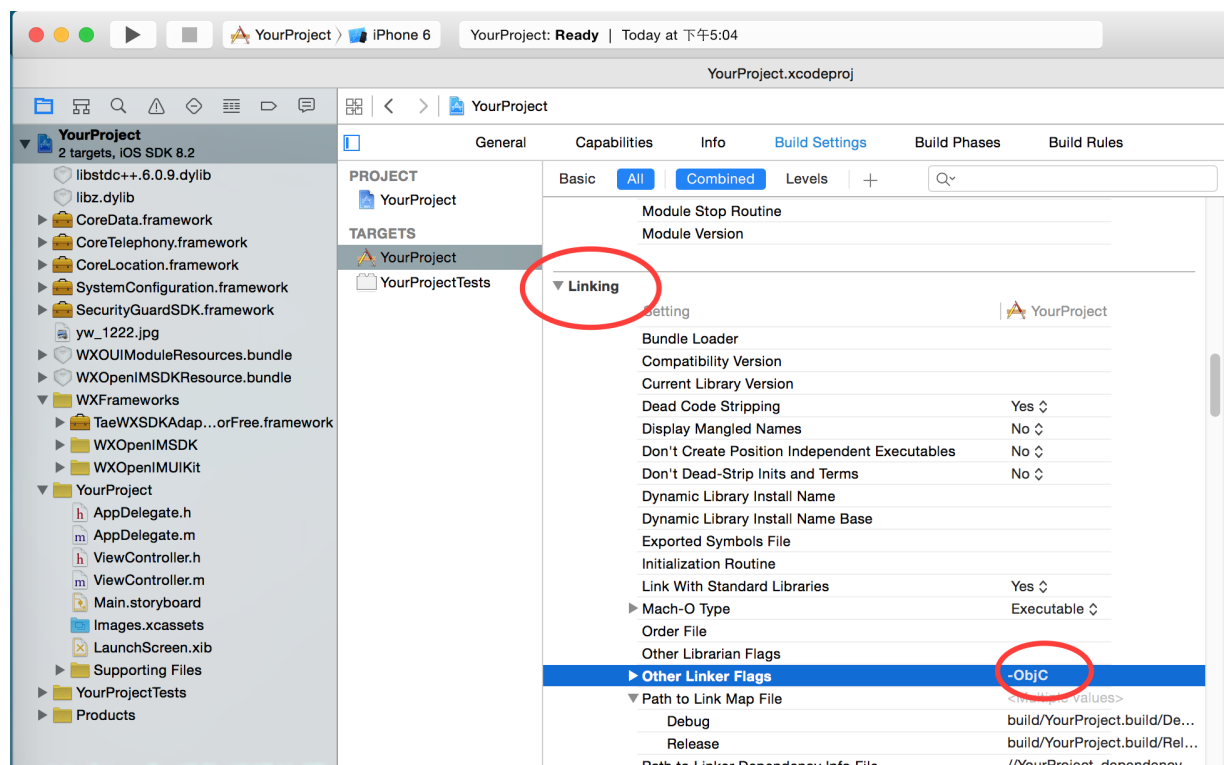
不知道如何添加系统库? 请参考下图:



Step4 修改编译选项

在 `Target->Linking->Other Linker Flags` 中添加 `-ObjC` 选项。

不知道如何修改编译选项? 请参考:



注意

请注意 `-ObjC` 的大小写。

Step5 iOS 10中隐私权限设置

iOS 10对隐私权限更加严格，需要添加权限说明（privacy description）。如果不做设置，可能会导致崩溃、审核不通过等情况。设置如下：

在info plist中增加字段：

```
<key>NSCameraUsageDescription</key>
<string>访问相机</string>
<key>NSPhotoLibraryUsageDescription</key>
<string>访问相册</string>
```

配置完后如图所示：

▼ Information Property List	Dictionary	(20 items)
Privacy - Camera Usage Description	String	访问相机
Privacy - Photo Library Usage Des...	String	访问相册

注意

- 上图中的访问相机/访问相册可修改。
- 如出现点击拍照、相册闪退的情况，确认是否已添加权限说明。

Step6 初始化SDK

- 引入头文件：

```
#import <YWFeedbackFMWK/YWFeedbackKit.h>
#import <YWFeedbackFMWK/YWFeedbackViewController.h>
```

- 调用初始化方法：

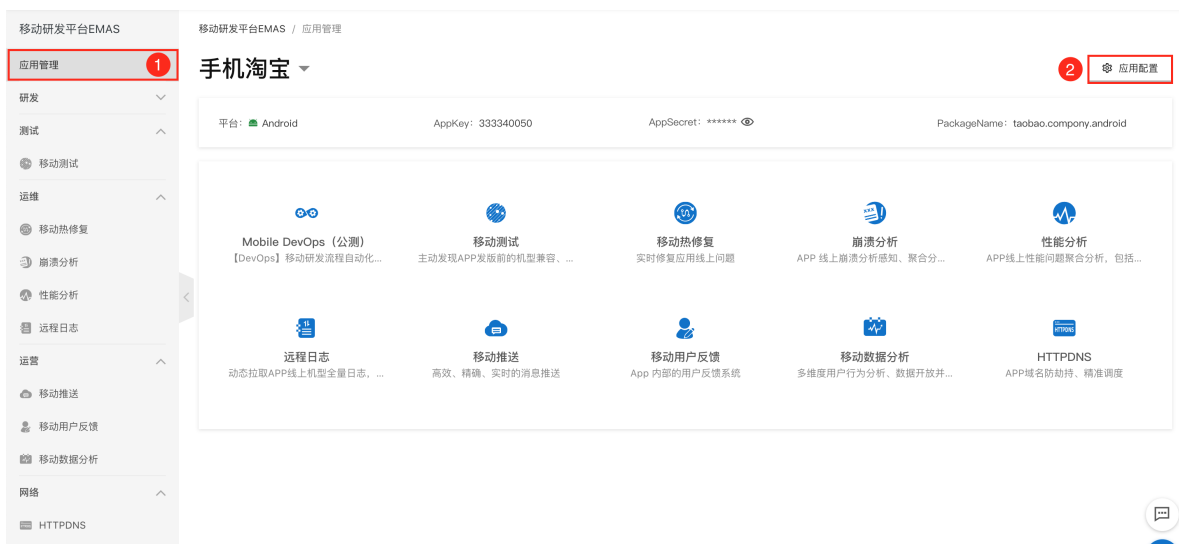
```
Objective-C

// 无需配置信息的初始化方法
- (instancetype)autoInit;

YWFeedbackKit *feedbackKitt = [[YWFeedbackKit alloc] autoInit];
```

下载SDK统一配置文件

- 在控制台应用管理页面，单击应用配置按钮，打开编辑配置右侧栏；在编辑配置右侧栏，单击下载Android/iOS应用配置按钮，下载App的配置文件 AliyunEmasServices-Info.plist，如下图所示：



- Xcode中，把下载的 AliyunEmasServices-Info.plist 文件拖入对应App Target下即可，在弹出框勾选 Copy items if needed。
- 【附】 AliyunEmasServices-Info.plist 配置文件，包含SDK初始化所需的配置信息，用户只需要调用无需手动输入配置信息的 autoInit 初始化接口。设置业务方扩展的反馈数据：

```
/// 业务方扩展反馈数据，可设置任意字段，在创建反馈页面前设置，可在后台扩展信息中查看
@property (nonatomic, strong, readonly) NSDictionary *extInfo;
```

 注意

- 不要将YWFeedbackKit实例设置为局部变量，请持有YWFeedback实例，以免被提前释放。
- 由于YWFeedbackKit是个宏，实际上是BCFeedbackKit，所以在Swift中，请使用BCFeedbackKit。

Step7 设置用户昵称

- 设置用户昵称后，用户反馈信息时携带该昵称，最终会在反馈控制台会话中显示。
- v3.3.5新增接口。

```
/**
 设置反馈用户名，请于SDK初始化完成后调用
  (该用户名，可以在移动反馈控制台反馈会话中查看，用于标识用户)
  @param nickname 反馈用户名
  */
- (void)setUserNick:(NSString *)nickname;
```

Step8 打开反馈页面

- 调用 YWFeedbackKit 实例如下接口创建反馈界面：

```
/// @brief 创建反馈页面回调Block
/// @params viewController 反馈页面
/// @return error 调用失败返回错误
typedef void (^YWMakeFeedbackViewControllerCompletionBlock) (YWFeedbackViewController * viewController, NSError *error);
/// @brief 创建反馈页面，默认为不显示弹出错误信息
- (void)makeFeedbackViewControllerWithCompletionBlock:(YWMakeFeedbackViewControllerCompletionBlock)completionBlock;
```

- 创建成功后将会返回 `YWFeedbackViewController`，你可以根据你App的实际结构，使用Present等不同的方式将其展示。
- 如果不使用系统默认alert展示错误提示，或者想自行进行错误提示的多语言化，可以自定义提示方式，只需要实现这个block即可。具体可参考 [Demo](#)。

```
/// @brief 反馈页面对外抛出Error的回调block。不设置此属性时，将使用默认Alert方式进行错误提示
/// @params viewController 反馈页面
/// @return error 失败error
@property (nonatomic, copy) void (^YWFeedbackViewControllerErrorBlock) (YWFeedbackViewController *viewController, NSError *error);
```

Step9 获取反馈未读数

- 调用如下接口获取未读数：

```
/// @brief 反馈未读消息数回调Block
/// @params unreadCount 未读消息数
/// @return error 调用失败返回错误
typedef void (^YWGetUnreadCountCompletionBlock) (NSInteger unreadCount, NSError *error);
/// @brief 请求反馈未读消息数
- (void)getUnreadCountWithCompletionBlock:(YWGetUnreadCountCompletionBlock)completionBlock;
```

- 调用成功后返回反馈未读数，如果返回有未读反馈消息，可在反馈入口添加未读数等方式提醒用户进行查看。

Step10 关闭用户反馈页面

在成功执行Step7后将获取到YWFeedbackViewController，设置属性closeBlock，当用户关闭用户反馈，将调用该block进行dismiss或pop。具体可参考 [Demo](#)。

```
[viewController setCloseBlock:^(UIViewController *aParentController){
    // 进行 dismiss 或者 pop，以及一些相关设置
    [aParentController dismissViewControllerAnimated:YES completion:nil];
}];
```

Step11 Mobile客户端配置

需要在控制台的 Mobile 客户端配置页面进行配置。

注意

上述接口在Demo中均有调用演示，具体可参考[Demo](#)

Step12 自定义UI

从v3.1.0版本开始增加了自定义UI的接口，包括导航栏字体，导航栏返回按钮的图标。

修改导航栏字体的API如下：

```
//YWFeedbackKit.h

/// @brief 如果不设置，默认为：`[UIFont boldSystemFontOfSize:13]`
@property (nonatomic, strong) UIFont *defaultCloseButtonTitleFont;

/// @brief 如果不设置，默认为：`[UIFont boldSystemFontOfSize:13]`
@property (nonatomic, strong) UIFont *defaultRightBarButtonItemTitleFont;
```

修改导航栏返回按钮图标的具体方法如下：

如果是手动集成，直接修改 `YWFeedbackBundle.bundle` 里的图片即可，如果是通过CocoaPods方式集成，那么需要修改 `YWFeedbackBundle.bundle` 命名为 `Customized.YWFeedbackBundle.bundle`，并修改 `Customized.YWFeedbackBundle.bundle` 内部的图片，然后拖拽到你自己的工程项目中即可。

常见错误码

错误码	含义
103	没有加-ObjC, Category没有加载上来。