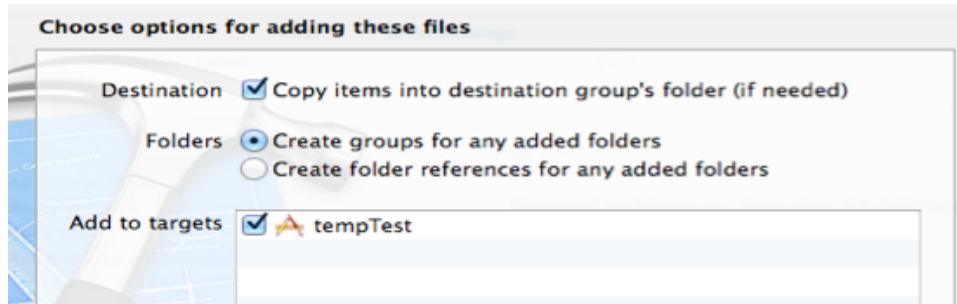


iOS-SDK接入文档之firebase推送篇

引入 SDK 文件

将 SDK 文件夹下 FirebaseMessaging 文件夹内文件引入到接入工程并选择正确的 target。

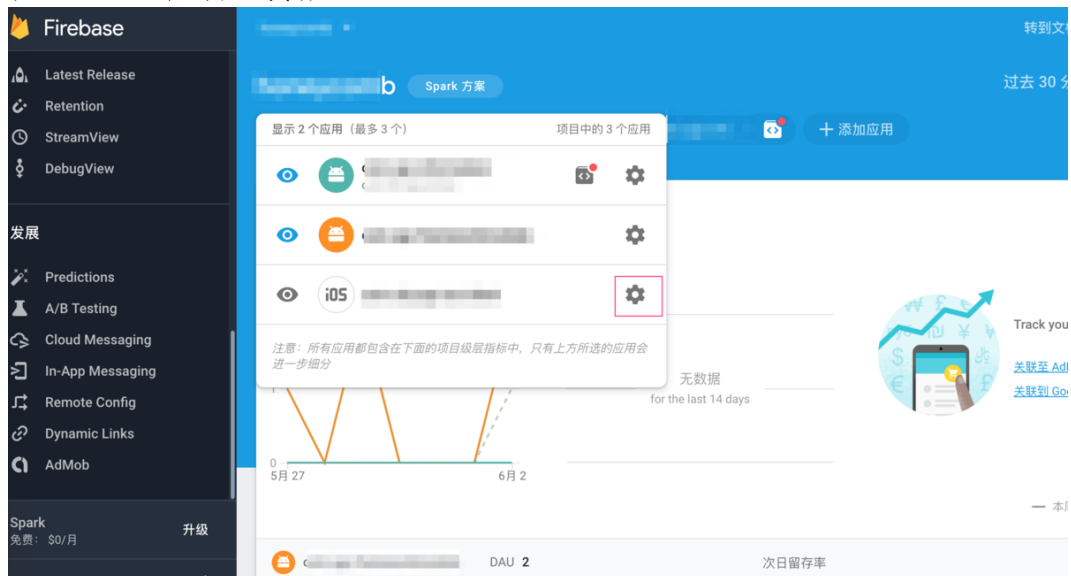


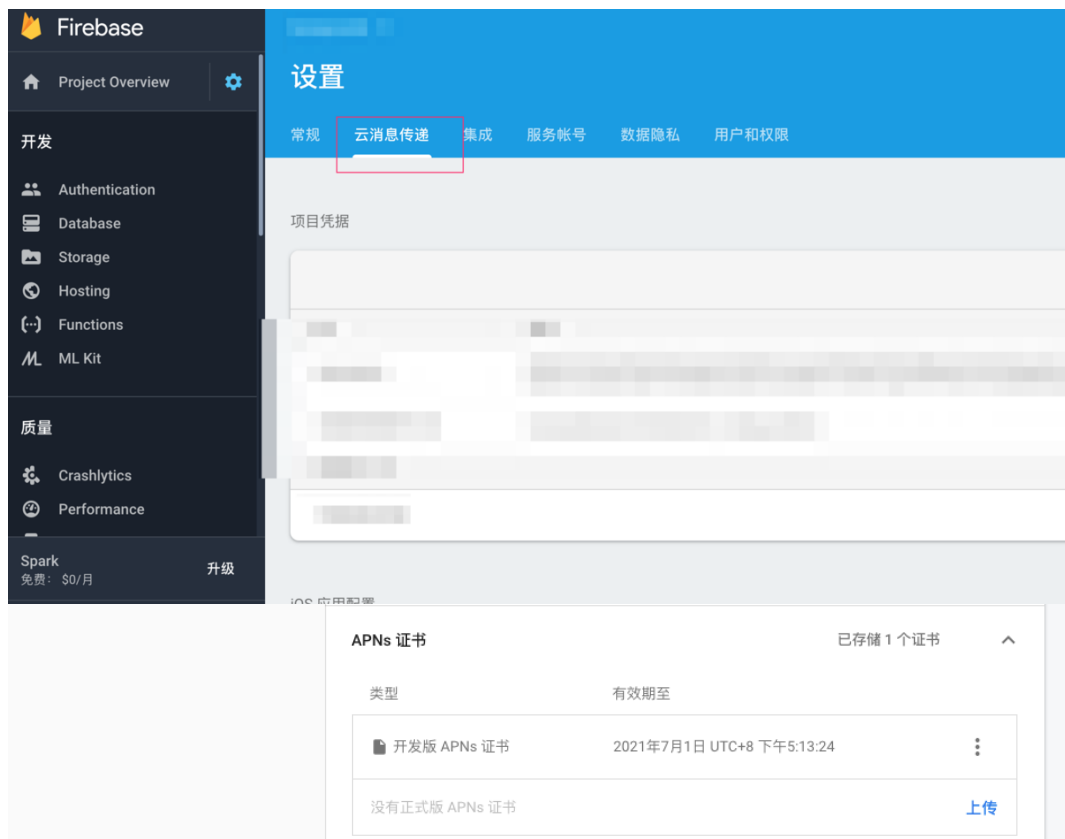
包含 **UnityFramework** 动态库特别说明:

FirebaseMessaging 文件夹下.framework 文件均为静态库, TargetMembership 需要关联到 UnityFramework。

firebase 后台及接入工程配置

1. 在 FireBase 后台上传推送 P12





2. 下载 googleService-Info.plist 文件
- 2.1 登录 Firebase 后台，然后打开您的项目。



2.2 点击  并选择项目设置。

2.3 在您的应用卡片中，从列表中选择您需要为其获取配置文件的应用的软件包 ID。

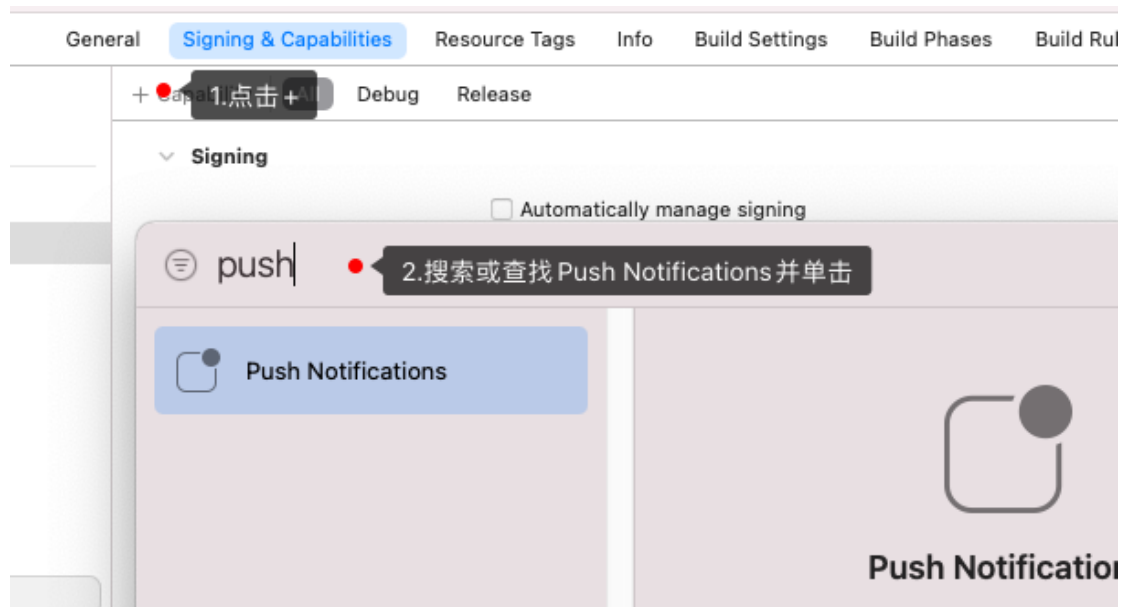


2.4 点击  GoogleService-Info.plist。

2.5 把 plist 文件放在 xcode 工程根目录下

注:没有引入此 plist 文件 xcode 工程会异常崩溃

3. 打开工程中推送配置
target->capabilities 添加 Push Notifications



4. 在 info.plist 配置 key: RegisterRemoteNotification (YES 时需要推送)

Information Property List	Dictionary	(20 items)
Localization native development re...	String	en
Bundle display name	String	QGOversea
Executable file	String	\$(EXECUTABLE_NAME)
▶ Icon files (iOS 5)	Dictionary	(0 items)
▶ CFBundleIcons~ipad	Dictionary	(0 items)
Bundle name	String	
Privacy - Photo Library Additions...	String	213
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
RegisterRemoteNotification	Boolean	NO

5. 在系统代理类里调用 SDK 的推送相关方法:

系统代理类 unity 工程一般为 UnityAppController; cocoa-2d 工程一般为 AppController

(必接)上传 deviceToken

(必接)+ (void)applicationRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken;

需要在- (void)application:(UIApplication *)application didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken 中添加这个代码上传 deviceToken 给 FIREBASE

tips:在系统代理类.m 或.mm 文件里面, 与此方法相同方法名的系统方法中调用此方法。(代理类中没有找到系统方法则新增一个, 输入方法名系统自动提示补全, 回车即可)。

(必接)+ (void)applicationDidFailToRegisterForRemoteNotifications;

需要在- (void)application:(UIApplication *)application didFailToRegisterForRemoteNotificationsWithError:(NSError *)error 添加, 当配置不全的时候会导致通知失败不走初始化

tips:在系统代理类.m 或.mm 文件里面, 与此方法相同方法名的系统方法中调用此方法。(代理类中没有找到系统方法则新增一个, 输入方法名系统自动提示补全, 回车即可)。