Q&A

1. 如何自定义一键分享列表?

在三种分享方式的接口下都带有一个 oneKeyShareList 参数,此参数用于自定义显示一键分享的平台类型,如下加粗部分代码所示:

```
id<ISSPublishContent> publishContent = [ShareSDK publishContent:CONTENT
                                           defaultContent:@""
                                                   image:[UIImage imageNamed:IMAGE NAME]
                                             imageQuality:0.8
                                               mediaType:SSPublishContentMediaTypeNews
                                                  title:@"ShareSDK"
                                                    url:@"http://www.sharesdk.cn"
                                             musicFileUrl:nil
                                                 extInfo:nil
                                                fileData:nil];
   //定制一键分享列表
   NSArray *oneKeyShareList = [ShareSDK
getShareListWithType:ShareTypeSinaWeibo,ShareTypeTencentWeibo,nil];
   [ShareSDK showShareActionSheet:self
                    shareList:nil
                      content:publishContent
                 statusBarTips:YES
               oneKeyShareList:oneKeyShareList
                shareViewStyle:ShareViewStyleDefault
                shareViewTitle:@"内容分享"
                       result:^(ShareType type, SSPublishContentState state, id<ISSStatusInfo>
statusInfo, id<ICMErrorInfo> error, BOOL end) {
                          if (state == SSPublishContentStateSuccess)
                             NSLog(@"发送成功");
                          }
                          else
                             NSLog(@"发送失败");
                          }
                          }]:
```

如果不需要一键分享可以传入 nil,默认一键分享列表则可以传入 [NSArray default0neKeyShareList]参数。

2. 微信和 QQ 的 AppKey 在哪里配置?

从微信和 QQ 开放平台申请下来 AppKey 后,首先打开*AppDelegate.m(*代表你的工程名字),然后在 didFinishLaunchingWithOptions 方法中写入注册应用的代码,如下:

```
[QQApi registerPluginWithId:@"QQ075BCD15"];
[ShareSDK registerApp:@"520520test"];
    然后确保 openURL 方法中加入如下代码:
- (BOOL)application:(UIApplication *)application handleOpenURL:(NSURL *)url
{
   return [ShareSDK handleOpenURL:url wxDelegate:self];
}
- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString
*)sourceApplication annotation:(id)annotation
{
   return [ShareSDK handleOpenURL:url wxDelegate:self];
}
    最后打开*-info.plist(*代表你的工程名字)文件。添加用于处理微信和 QQ 的回调 URL Scheme。
    如下图:
   n issuserinio.n

■ Supported interface orientations (ira

                                                               Array
                                                                         (± item)
    NSArray+ShareSDK.h
                                   ₩ URL types
                                                               Array
                                                                         (1 item)
   h ShareConfig.h

▼ Item 0 (Editor)

                                                               Dictionary
                                                                          (3 items)
   h ShareSDK.h
                                                               String
                                        Document Role
                                                                         Editor
   h ShareType.h
                                       ▼ URL Schemes
                                                                         (3 items)
                                                               Array
   F ShareViewStyle.h
                                                                         sinaweibosso.2279784657
                                          Item 0
                                                               String
    SSInheritValue.h
                                          Item 1
                                                               String
                                                                         wx6dd7a9b94f3dd72a
   SSNotifiationType.h
                                          Item 2
                                                                         QQ075BCD15
                                                               String
```

URL identifier

Status bar style

Icon already includes gloss effects

3. 如何查看错误码?

SSPublishContentMediaType.h

N UserParamType.h

AGShareSDKDemo-Info.p InfoPlist.strings

AGShareSDKDemo

Supporting Files

所有带错误码返回的接口(包括分享、获取用户资料、关注用户等)都通过特定的 block 方法中的 error 进行返回。因此,在调用接口的时候指定 block 方法,然后根据返回结果来获取错误代码即可。下面加粗部分代码展示了如何捕获分享异常的错误码以及错误描述:

String

String

Boolean

YES

Opaque black style

error 参数是一个实现了 ICMErrorInfo 协议的对象, 其包含三个属性:

一个是 errorLevel, 用于指定错误级别; 有三种级别分别是 CMErrorLevelAPI, CMErrorLevelHTTP以及 CMErrorLevelNetwork。其中 CMErrorLevelAPI为调用 API 时返回错误,该错误码与各个平台返回错误码相同,详细可以参考各个平台的错误码描述。CMErrorLevelHTTP为 HTTP请求错误,错误码为 HTTP返回状态码,CMErrorLevelNetwork则表示网络错误。

第二个是 errorCode,包含了指定的错误代码。

第三个是 errorDescription,包含了对应错误码的描述信息。

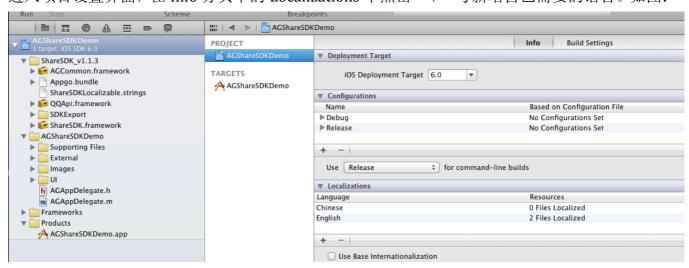
4. 如何获得成功或失败的信息?

要想知道一个接口的返回状态是否成功,可以通过每个带有 result 参数的接口传递一个 block 方法。然后由 result 返回的参数中判断是否成功,以及失败的错误信息。result 的 block 方法定义可 SDK 中的 EventHandler.h 头文件,下面加粗部分代码展示了,如何判断分享是否成功,如果不成功则打印错误日志:

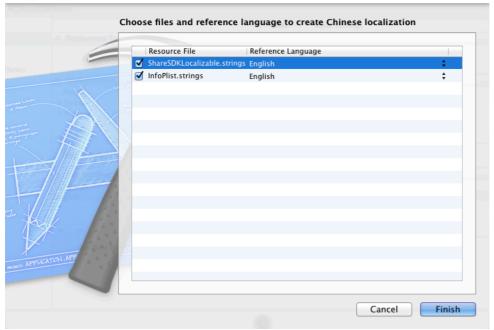
```
[ShareSDK showShareActionSheet:self
                   shareList:nil
                     content:publishContent
                statusBarTips:YES
              oneKeyShareList:[NSArray defaultOneKeyShareList]
               shareViewStyle:ShareViewStyleDefault
               shareViewTitle:@"内容分享"
                     result: ^(ShareType type, SSPublishContentState state,
id<ISSStatusInfo> statusInfo, id<ICMErrorInfo> error, BOOL end) {
                           if (state == SSPublishContentStateSuccess)
                           {
                               NSLog(@"分享成功");
                           else if (state == SSPublishContentStateFail)
                           {
                               NSLog(@"分享失败,错误码:%d,错误描述:%@", [error errorCode],
[error errorDescription]);
                           }
                           }];
```

5. 如何为 ShareSDK 实现本地化 UI 及提示信息?

进入项目设置界面,在 Info 分页下的 Localizations 中点击"+"号新增自己需要的语言。如图:



再弹出的界面上选中 ShareSDKLocalizable.strings 文件,点击 Finish 按钮即可为 SDK 新增对应语言的本地化信息。如图:



然后点击 ShareSDKLocalizable.strings 文件左边的小三角形按钮展开分支,选择需要编辑的对应语言版本即可完成本地化操作。如图:

```
"MSG_SYSTEM_VERSION_LOW_THAN_4_2" = "该功能不支持iOS 4.2以下系
AGShareSDKDemo
1 target, iOS SDK 6.0
                                                           "PRINT_JOB_NAME" = "内容分享";
▼ ShareSDK_v1.1.3
  AGCommon.framework
                                                           "INSTAPAPER_AUTH_TITLE" = "绑定Instapaper";
                                                           "INSTAPAPER_CANCEL_BUTTON" = "取消";
  Appgo.bundle
                                                           "INSTAPAPER_LOGIN_BUTTON" = "登录";
  ▼ ShareSDKLocalizable.strings
                                                           "INSTAPAPER_ACCOUNT_INPUT_PLACEHOLDER" = "帐号,如name@examp
        ShareSDKLocalizable.strings (English)
                                                           "INSTAPAPER_PASS_INPUT_PLACEHOLDER" = "密码";
                                                           "INSTAPAPER_ACCOUNT_INVALID" = "请输入登录帐号!":
          ShareSDKLocalizable.strings (Chinese)
                                                           "INSTAPAPER_PASS_INVALID" = "请输入登录密码!";
  QQApi.framework
  SDKExport
                                                           "WEIXIN_SESSION_NAME" = "微信好友"
                                                           "WEIXIN_TIMELINE_NAME" = "微信朋友圈";
"WEIXIN_MSG_SEND_SUCCESS" = "发送%@成功!";
  ShareSDK.framework
▼ ☐ AGShareSDKDemo
                                                           "WEIXIN MSG SEND FAIL" = "发送%@失败:%@";
   Supporting Files
                                                           "QQ_MSG_SEND_SUCCESS" = "发送QQ消息成功";
  External
                                                           "QQ MSG SEND FAIL" = "发送QQ消息失败:%@";
  Images
                                                           "SHARE_VIEW_CANCEL_BUTTON" = "取消";
"SHARE_VIEW_PUBLISH_BUTTON" = "发表"
     AGAppDelegate.h
                                                           "SHARE_VIEW_MSG_INVALID_CONTENT" = "请输入分享内容"; "SHARE_VIEW_DEF_PIC_PUBLISH_CONTENT" = "图片分享";
     m AGAppDelegate.m
Frameworks
                                                          "SHARE_VIEW_ACTIONSHEET_CLEAR_CONTENT" = "图万万字";
"SHARE_VIEW_ACTIONSHEET_CANCEL_ITEM" = "非空内容";
"SHARE_VIEW_ACTIONSHEET_CANCEL_ITEM" = "取消";
"SHARE_VIEW_ACTIONSHEET_REMOVE_PHOTO_ITEM" = "删除照片";
   Products
    AGShareSDKDemo.app
                                                           "SHARE_VIEW_ACTIONSHEET_VIEW_PHOTO_ITEM" = "观看照片"; "SHARE_VIEW_ACTIONSHEET_TAKE_PHOTO_ITEM" = "拍照...";
                                                           "SHARE_VIEW_ACTIONSHEET_SELECT_PHOTO_ITEM" = "选取照片...";
                                                           "FRIENDS_VIEW_TITLE" = "联系人";
"FRIENDS_VIEW_CANCEL_BUTTON" = "取消";
"EDTENDS_VIEW_SEADCHRAD_DLACEHOLDED" - "在既至人由神虚
```

6. 如何自定义分享菜单中的平台列表?

在 showShareActionSheet 方法中的第二参数 shareList 用于指定分享菜单中需要显示那些分享平台。如果传入 nil 则表示使用默认的分享平台列表,如果不需要某些平台时可以使用 ShareSDK中的 getShareListWithType 方法来构造一个符合自己需求的分享列表。下面加粗部分代码演示了只显示新浪微博和腾讯微博的分享列表:

NSArray *shareList = [ShareSDK getShareListWithType:ShareTypeSinaWeibo,
ShareTypeTencentWeibo, nil];

```
[ShareSDK showShareActionSheet:self shareList:shareList content:publishContent statusBarTips:YES
```

7. 豆瓣社区授权时页面出现"invalid_request:not_trial_user:xxxxxx Error 400"报错,如果解决?

由于所创建的 App 尚未通过审核,因此在授权用户不是测试用户的情况下进行授权就会出现如下报错页面:



解决方法是打开豆瓣开发者页面中的"我的应用"进入需要设置测试用户的 App,选择"测试用户"项,在右边界面的输入框填入报错界面中的 id 后,点击"添加"按钮添加测试用户即可。