

GameFunSDK API Document

(iOS)

Name	Date	version	remark
Yuanjun	2024.08.30	1.0.0	

1. Description and definition

1.1 API call

Import <GameFunSDK/GameFunSDK.h>

Every API is called by [GameFunManager defaultManager] instance

1.2 Callback

`error` is a common field for every callback class. if `error` equals to `nil`, it means the api call is success.

1.3 Callback classes:

1.3.1 Login Callback

```
class: GAMFUNUserCompletionHandler
fields:
userInfo //the user information
error
```

1.3.2 Receive Notification Callback

```
class: GAMFUNNotificationCompletionHandler  
fields:  
result      //the notification message  
error
```

1.3.3 Logout Callback

```
class: GAMFUNLogoutCompleteHandler  
fields:  
success      // BOOL type, indicate operation result  
error
```

1.3.4 Get Friends Callback

```
class: GAMFUNFriendsCompletionHandler  
fields:  
list          //fields list, contains from 0 -> N friend object  
error
```

1.3.5 Share Callback

```
class: GAMFUNShareCompleteHandler  
fields:  
success      // BOOL type, indicate operation result  
error
```

1.3.6 Payment Callback

```
class: GAMFUNOrderProgressHandler  
fields:  
GAMFUNOrderStatus //Payment order status
```

1.3.7 Payment Result Callback

```
class: GAMFUNOrderCompletionHandler
fields:
orderInfo //the payment order object
error
```

1.3.8 AppVersion Info Callback

```
class: GAMFUNVersionCompletionHandler
fields:
hasNewVersion //has new version
versionInfo //the version detail information. class model <GAMFUNAppVersion>
error
```

1.3.9 Common Callback

```
class: GAMFUNCompletionHandler
fields:
result //operation result
error
```

1.3.10 Account Kicked Callback

```
class: GAMFUNAccountKickedHandler
fields:
information //information of notification
```

1.3.11 Account Banned Callback

```
class: GAMFUNAccountBannedHandler
fields:
information //the reason of banned
time
```

1.4 Errors

GAMFUNPlatformErrorType_Unknown	unknown error
GAMFUNPlatformErrorType_AuthorizeFailed	authorized failed
GAMFUNPlatformErrorType_ForUserInfoFailed	request user data failed
GAMFUNPlatformErrorType_LoginFailed	login failed
GAMFUNPlatformErrorType_FriendsListFailed	get friends failed
GAMFUNPlatformErrorType_MessageSendFailed	send message failed
GAMFUNPlatformErrorType_PayFailed	payment failed
GAMFUNPlatformErrorType_ShareFailed	share failed
GAMFUNPlatformErrorType_NotInstall	app is not installed
GAMFUNPlatformErrorType_NotNetWork	network exception
GAMFUNPlatformErrorType_SourceError	third error
GAMFUNPlatformErrorType_NoPermission	no permissions
GAMFUNPlatformErrorType_UnauthorizedRequest	unauthorized request, need login.

2. Setup Initialize (must)

2.1 Launch

Methods in AppDelegate are implemented like below:

2.1.1 Setup Configuration:

The launch animation for SDK-1.0.5 has been removed and the specific launch animation needs to be implemented by the game developers.

```
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

    // set SDK Api Environment
```

```

[[GameFunManager defaultManager] setServerDomainTest];

// set Pay Enviroment

[[GameFunManager defaultManager] setPayEnvironmentTest];

// open log

[[GameFunManager defaultManager] openLogInfo];


// launch SDK

NSString *appId = @"xxxxx";

NSString *appSecret = @"xxxxxx";

NSString *gameid = @"xxxxxxx";

[[GameFunManager defaultManager] setupWithAppId:appId Secret:appSecret
GameId:gameid Application:application Options:launchOptions];

return YES;
}

```

2.1.2 External Callback:

```

- (BOOL)application:(UIApplication *)app openURL:(NSURL *)url
options:(NSDictionary<UIApplicationOpenURLOptionsKey,id> *)options {

    return [[GameFunManager defaultManager] application:app openURL:url
options:options];
}

```

2.1.3 App Events:

```

- (void)applicationDidEnterBackground:(UIApplication *)application {

    [[GameFunManager defaultManager] applicationDidEnterBackground:application];
}

- (void)applicationWillEnterForeground:(UIApplication *)application {

    [[GameFunManager defaultManager] applicationWillEnterForeground:application];
}

- (void)applicationDidBecomeActive:(UIApplication *)application {

    [[GameFunManager defaultManager] applicationDidBecomeActive:application];
}

- (void)applicationWillTerminate:(UIApplication *)application {

    [[GameFunManager defaultManager] applicationWillTerminate:application];
}

```

2.1.4 Push Register:

```
- (void)application:(UIApplication *)application
didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken {

    [[GameFunManager defaultManager] registerDeviceToken:deviceToken];
}

- (void)application:(UIApplication *)application
didReceiveRemoteNotification:(NSDictionary *)userInfo fetchCompletionHandler:(void
(^)(UIBackgroundFetchResult))completionHandler {

    [[GameFunManager defaultManager] application:application
didReceiveRemoteNotification:userInfo];
}
```

2.1.5 Universal Link:

```
- (BOOL)application:(UIApplication *)application continueUserActivity:(NSUserActivity
*)userActivity restorationHandler:(void (^)(NSArray<id<UIUserActivityRestoring>> *
_Nullable))restorationHandler {

    return [[GameFunManager defaultManager] application:application
continueUserActivity:userActivity restorationHandler:restorationHandler];
}
```

2.2 Check App Version (optional)

After SDK is launched, be sure to check the version to ensure that the current version is available. If you need to force upgrade, you can go to the appstore to download the latest version, otherwise the SDK function may not be available.

example

```
- (void)checkVersion {

    [[GameFunManager defaultManager] checkAppVersionCompletion:^(BOOL hasNewVersion,
GAMFUNAppVersionModel * _Nullable versionInfo, NSError * _Nullable error) {

        if (hasNewVersion) {

            BOOL forceUpdate = versionInfo.forceUpdate;

            if (forceUpdate) {

                // If you need to force update, you cannot continue to use it, and you need
                to jump to the appstore to update.
            }
        }
    }
}
```

```

        // Open appstore

        [[GameFunManager defaultManager]
openAppStoreWithStoreIdentifier:@"xxx"];

    } else {

        // If update is not required, you can optionally prompt for an upgrade.

    }

}

}];

}

```

2.2.1 Check Version

Method	checkAppVersionCompletion
Callback	GAMFUNVersionCompletionHandler hasNewVersion // has new version <BOOL> versionInfo // the version detail information. class model <GAMFUNAppVersion> build // app build code version // app new version versionInfo // the version information forceUpdate // need to force upgrade <BOOL> currentVersionCode // current version code

2.2.2 Open AppStore

Method	openAppStoreWithStoreIdentifier
Params	identifier //App Store item identifier (NSNumber) of the product

3. API: Configuration

3.1 Enbale Log Output

```

[[GameFunManager defaultManager] openLogInfo];

```

3.2 Disable Log Output(default)

```
[[GameFunManager defaultManager] closeLogInfo];
```

3.3 Set Test Environment

```
[[GameFunManager defaultManager] setServerDomainTest];
```

3.4 Set Production Environment(default)

```
[[GameFunManager defaultManager] setServerDomainProduction];
```

3.5 Set Sandbox Payment

```
[[GameFunManager defaultManager] setPEnvTest];
```

3.6 Set Production Payment(default)

```
[[GameFunManager defaultManager] setPEnvProduction];
```

3.7 Customer Service (Messenger)

```
[[GameFunManager defaultManager] contactusWithMessenger];
```

4. API: Login&User Info

User must agree the User Agreement & Privacy before do Login action

4.1 Show Agreement View

```
- (IBAction)agreement:(id)sender {
```



```
[[GameFunManager defaultManager] showAgreementView];
}
```

4.2 Login with Apple

Login with Apple only supports iOS13 and higher.

Method	loginWithAppleCompletion
Callback	GAMFUNUserCompletionHandler

example:

```
- (IBAction)appleLogin:(id)sender {
    GAMFUNHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

    [[GameFunManager defaultManager] loginWithAppleCompletion:^(id _Nonnull userInfo,
NSError * _Nonnull error) {
        if (!error) {
            //login success

            GAMFUNHUD_SUCCESS(@"success")

            [weakSelf closeLoginView];
        } else {
            //error info

            GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
        }
    }];
}
```

4.3 Login with GameCenter

Method	loginWithGameCenterCompletion
Callback	GAMFUNUserCompletionHandler

example:

```
- (void)gamecenterLogin {
    GAMFUNHUD_NO_Stop(@"login...")
}
```

```

    __weak __typeof(self) weakSelf = self;

    [[GameFunManager defaultManager] loginWithGameCenterCompletion:^(id _Nonnull
userInfo, NSError * _Nonnull error) {

        if (!error) {

            //login success

            GAMFUNDHUD_HINT(@"success")

            [weakSelf closeLoginView];

        } else {

            //error info

            GAMFUNDHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

        }

    }];
}

```

4.4 Login with Facebook

Method	loginWithFacebookCompletion
Callback	GAMFUNUserCompletionHandler

example:

```

- (void)facebookLogin {

    GAMFUNDHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

    [[GameFunManager defaultManager] loginWithFacebookCompletion:^(id _Nonnull
userInfo, NSError * _Nonnull error) {

        if (!error) {

            //login success

            GAMFUNDHUD_HINT(@"success")

            [weakSelf closeLoginView];

        } else {

            //error info

            GAMFUNDHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

        }

    }];
}

```

```
}
```

4.5 Login with Twitter

Method	loginWithTwitterCompletion
Callback	GAMFUNUserCompletionHandler

example:

```
- (IBAction)twitterLogin:(id)sender {

    GAMFUNHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

    [[GameFunManager defaultManager] loginWithTwitterCompletion:^(id _Nonnull userInfo,
NSError * _Nonnull error) {

        if (!error) {

            //登录成功

            GAMFUNHUD_SUCCESS(@"success")

            [weakSelf closeLoginView];

        } else {

            //查看 error 信息

            GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

        }

    }];

}
```

4.6 Login with Botim

Method	loginWithBotimCompletion
Callback	GAMFUNUserCompletionHandler

example:

```
- (IBAction)botimLogin:(id)sender {

    if ([[GameFunManager defaultManager] botimIsInstall]) {

        GAMFUNHUD_NO_Stop(@"login...")

        __weak __typeof(self) weakSelf = self;
```

```

[[GameFunManager defaultManager] loginWithBotimCompletion:^(id _Nonnull userInfo,
NSError * _Nonnull error) {

    if (!error) {

        //login success

        GAMFUNDHUD_SUCCESS(@"success");

        [weakSelf closeLoginView];

    } else {

        //error info

        GAMFUNDHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

    }

}];

} else {

    // no app please hide login button

}

}

```

4.7 Login with Google

Method	loginWithGoogleCompletion
Callback	GAMFUNUserCompletionHandler

example:

```

- (IBAction)googleLogin:(id)sender {

    GAMFUNDHUD_NO_Stop(@"login...")

    __weak __typeof(self)weakSelf = self;

    [[GameFunManager defaultManager] loginWithGoogleCompletion:^(id _Nonnull userInfo,
NSError * _Nonnull error) {

        if (!error) {

            //login success

            GAMFUNDHUD_SUCCESS(@"success");

            [weakSelf closeLoginView];

        } else {

            //error info

            GAMFUNDHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

        }

    }];

}

```

```

    }

    }];
}

```

4.8 Login with Guest

Method	guestLoginCompletion
Callback	GAMFUNUserCompletionHandler

example

```

- (void)guestLogin {

    GAMFUNHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

    [[GameFunManager defaultManager] guestLoginCompletion:^(id _Nonnull userInfo,
NSError * _Nonnull error) {

        if (!error) {

            //login success

            GAMFUNHUD_HINT(@"success")

            [weakSelf closeLoginView];

        } else {

            //error info

            GAMFUNHUD_Stop

        }

    }];

}

```

4.9 Get User Info

Method	userInfoCompletion
Callback	GAMFUNUserCompletionHandler
Params	GAMFUNUserModel <div> <div>userType</div> <div>user type (GAMFUNLoginType by 4.10)</div> </div>

	userId	user's identifier
	thirdUid	third-party user identifier
	nickname	nickname
	photoUrl	head image url
	pushOff	push switch 0:push switch on 1:push switch off

example

```
- (void)getUserInfo {
    GAMFUNDHUD_NO_Stop(@"")

    [[GameFunManager defaultManager] userInfoCompletion:^(id _Nullable userInfo,
NSError * _Nullable error) {
        if (!error) {
            GAMFUNDHUD_HINT(@"success")

            GAMFUNUserModel *model = userInfo;

            [self freshUserInfo:model];
        } else {
            //error info

            GAMFUNDHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
        }
    }];
}
```

4.10 Get Login Type

Method	loginType
Return	GAMFUNLoginType

Definition:

Enum	GAMFUNLoginType
Description	Enum type for Login
Enum Values	
GAMFUNLoginType_unloggedIn	not login
GAMFUNLoginType_TTk	TTk user

GAMFUNLoginType_GameCenter	GameCenter player
GAMFUNLoginType_Facebook	Facebook user
GAMFUNLoginType_Google	Google user
GAMFUNLoginType_Twitter	Twitter user
GAMFUNLoginType_Guest	Game guest

example:

```
- (void)showLoginView {
    if ([[GameFunManager defaultManager] loginType] == GAMFUNLoginType_unloggedIn) {
        //unlogged in and show login UI
        [self backgroundUI];
        [self loginView];
    } else {
        dispatch_after(dispatch_time(DISPATCH_TIME_NOW, (int64_t)(2.5 * NSEC_PER_SEC)),
dispatch_get_main_queue(), ^{
            [self getUserInfo];
        });
    }
}
```

4.11 Notification of offline

Method	accountKickedNotification
Callback	GAMFUNAccountKickedHandler

example:

```
[[GameFunManager defaultManager] accountKickedNotification:^(NSString * _Nullable
information) {
    //Your account has been logged in on another device, please login again.
    GAMFUNHUD_HINT(information);
    [self showLoginView];
}];
```

4.12 Notification of banned

Method	accountBannedNotification
Callback	GAMFUNAccountBannedHandler

example:

```
[[GameFunManager defaultManager] accountBannedNotification:^(NSString * _Nullable
information, NSString * _Nullable time) {

    //Your account has been banned.

}];
```

4.13 Get Available Token

Method	getAvailableAccessToken
Callback	GAMFUNUserCompletionHandler

example:

```
[[GameFunManager defaultManager] getAvailableAccessToken:^(id _Nullable userInfo,
NSError * _Nullable error) {

    if (userInfo) {

        GAMFUNUserModel *userModel = userInfo;

        NSLog(@"userid: %@", userModel.userId);

        NSLog(@"token: %@", userModel.accessToken);

    }

}];
```

4.14 Set Migration Code

Method	setupMigCode
--------	--------------

Note:

If you want to migrate users, be sure to use this API to set the migration code before logging in.

example:

```
[[GameFunManager defaultManager] setupMigCode:@"Your Mig Code"];
```

4.15 Custom Login with UI

Method	loginWithCustomUICompletion
Callback	GAMFUNUserCompletionHandler

example:

```
- (IBAction)customLogin:(id)sender {
    GAMFUNHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

    [[GameFunManager defaultManager] loginWithCustomUICompletion:^(id _Nonnull userInfo,
NSError * _Nonnull error) {
        if (!error) {
            //登录成功

            GAMFUNHUD_SUCCESS(@"success")

            [weakSelf closeLoginView];
        } else {
            //查看 error 信息

            GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
        }
    }];
}
```

5. API: Logout

Method	logout
Callback	GAMFUNLogoutCompleteHandler

example:

```
- (IBAction)logout:(id)sender {
```

```

GAMFUNHUD_NO_Stop(@"logout...")

[[GameFunManager defaultManager] logout:^(BOOL success, NSError * _Nullable error) {

    if (success) {

        GAMFUNHUD_HINT(@"success")

        [self showLoginView];

    } else {

        //error info

        NSLog(@"%@",error.userInfo);

        GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

    }

}];

}

```

6. API: Game Payment

6.1 Product Purchase

Order Status Definition:

GAMFUNOrderStatus_ProductQuerying	query product
GAMFUNOrderStatus_Generating	order init
GAMFUNOrderStatus_Generated	order init finish
GAMFUNOrderStatus_ProductPurchasing	order in trading progress
GAMFUNOrderStatus_ProductPurchased	order purchased
GAMFUNOrderStatus_ReceiptChecking	order verifying

6.1.1 IAP

Method	1. buyProductWithSKU: Progress: Completion: 2. buyProductWithSKU: WithName: Progress: Completion:(v1.1.17 or
--------	---

	later)
Params	
sku	String,product id
name	String,product name or description
Progress	GAMFUNOrderProgressHandler
Completion	GAMFUNOrderCompletionHandler <pre> class GAMFUNOrderModel orderId orderRef //transaction_id channel //1 env //0=sandbox, 1=production sku //product id amount //price createTime payTime cancelTime status //0=order created, 1=order payed, 2=order reset </pre>

6.1.2 Xsolla

Method	xslBuyProductWithSKU: Currency: Completion:(v2.0.0 or later)
Params	
sku	String,product id
currency	String,pay currency
Completion	GAMFUNOrderCompletionHandler <pre> class GAMFUNOrderModel orderId orderRef //transaction_id channel //1 env //0=sandbox, 1=production </pre>

	sku //product id amount //price createTime payTime cancelTime status //0=order created, 1=order payed, 2=order reset
--	---

6.1.3 Payment Selector (recommend 2)

Method	1. selectBWithProductSKU: Name: Currency: Completion: (v2.1.0 or later) <div>Note: The following methods require the game's order id, which is used for server notification. (v2.2.3 or later)</div> 2. selectBWithProductSKU: Name: Currency: GameOrder: Progress: Completion:
Params	
sku	String,product id
name	String,product name
currency	String,pay currency
gameOrder	String,game order id
progress	GAMFUNOrderProgressHandler
Completion	GAMFUNOrderCompletionHandler <pre> class GAMFUNOrderModel orderId //SDK order id orderRef //transaction_id cpOrderId //game order id channel //1 env //0=sandbox, 1=production sku //product id amount //price createTime </pre>

	payTime cancelTime status //0=order created, 1=order payed, 2=order reset
--	---

remark:

the Game Server should send the product to user's account after the order be payed, then check the order status by server api, Order status check is necessary for anti-spam

Server API: <GameFun Server API Document (to Game Developer)>

example:

```

[[GameFunManager defaultManager] selectBWithProductSKU:@"com.amuse.game.coin"
Name:@"ios(apple)" Currency:@"AED" GameOrder:@"123456" Progress:^(GAMFUNOrderStatus
orderStatus) {

    NSLog(@"order status: %ld", (long)orderStatus);

} Completion:^(id _Nullable orderInfo, NSError * _Nullable error) {

    if (error) {

        NSLog(@"pay error\n%@", error);

    } else {

        if (orderInfo) {

            GAMFUNOrderModel *model = orderInfo;

            NSLog(@" orderId   : %@", model.orderId);

            NSLog(@" orderRef   : %@", model.orderRef);

            NSLog(@" cpOrderId  : %@", model.cpOrderId);

            NSLog(@" channel   : %@", model.channel);

            NSLog(@" sku       : %@", model.sku);

            NSLog(@" status    : %@", model.status);

        }

    }

}];

```

6.2 Query Order

Method	queryOrderWithOrderId: Completion:
Params	
orderId	the order's id
Completion	GAMFUNOrderCompletionHandler GAMFUNOrderModel by 6.1

example:

```
- (IBAction)query:(id)sender {  
    [[GameFunManager defaultManager] queryOrderWithOrderId:@"xxx"  
Completion:^(id _Nullable orderInfo, NSError * _Nullable error) {  
        if (orderInfo) {  
            //get order info  
        } else {  
            //error message  
        }  
    }  
};  
}
```

6.3 Get Product Info

Method	getProductDetailWithSKU: Completion:
Params	
sku	String,product id
Completion	GAMFUNProductCompletionHandler class GAMFUNProductModel sku; //The string that identifies the product to the Apple App Store. price; //The cost of the product in the local currency. localeCurrency; //The currency code associated with the locale. localizedTitle; //The name of the product. localizedDescription; //A description of the product.

example:

```
- (void)getInfo:(NSString *)sku {

    [[GameFunManager defaultManager] getProductDetailWithSKU:sku

Completion:^(id _Nullable productInfo, NSError * _Nullable error) {

    if (productInfo) {

        GAMFUNProductModel *model = productInfo;

        NSLog(@" sku           : %@", model.sku);

        NSLog(@" price           : %@", model.price);

        NSLog(@" localeCurrency    : %@", model.localeCurrency);

        NSLog(@" localizedTitle    : %@", model.localizedTitle);

        NSLog(@" localizedDescription : %@", model.localizedDescription);

    } else {

        // error info

    }

}

}];

}
```

7. API: SNS & Share

7.1 Field Description

GAMFUNBotimFriendModel

Params	Note
bimUid	botim user id
name	friend's name
photoUrl	friends' avatars

GAMFUNBotimPlayingFriendModel

Params	Note
bimUid	botim user id
userId	user id of friend
name	friend's name
photoUrl	friends' avatars

GAMFUNSocialBimMessage

Params	Note
bimUid	botim user id
contenString	shared text content
imageUrl	shared web picture address
title	shared title

7.2 Share (Third-Party)

Share to Facebook, WhatsApp.

7.2.1 Share To Facebook

Method	facebookShareMessage: completion:
Params	
Message	1.GAMFUNSocialFBInvite quote //message text
	2.GAMFUNSocialFBImages photoImage //photo image <UIImage>
	3.GAMFUNSocialFBFriendsGameInvite inviteString //invite message text
completion	GAMFUNShareCompleteHandler

example:

```
// share invite link to facebook
- (IBAction)fbShare:(id)sender {
    GAMFUNSocialFBInvite *obj = [[GAMFUNSocialFBInvite alloc] init];
    obj.quote = @"Join us";
    [[GameFunManager defaultManager] facebookShareMessage:obj completion:^(BOOL success,
NSError * _Nullable error) {
        if (success) {
        } else {
            if (error) {
                GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
            }
        }
    });
}
```



```

    }

    }

    }];
}

//share image to facebook
- (IBAction)fbpic:(id)sender {
    UIImage *image = [UIImage imageNamed:@"game_logo_1024"];
    GAMFUNSocialFBImages *obj = [[GAMFUNSocialFBImages alloc] init];
    obj.photoImage = image;
    [[GameFunManager defaultManager] facebookShareMessage:obj completion:^(BOOL success,
NSError * _Nullable error) {
        if (success) {
        } else {
            if (error) {
                GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
            }
        }
    }];
}

//send invite to facebook friends(This feature is only available if you are logged in with
Facebook)
- (IBAction)fbGameInviteFriends:(id)sender {
    GAMFUNSocialFBFriendsGameInvite *obj = [[GAMFUNSocialFBFriendsGameInvite alloc]
init];
    obj.inviteString = @"join us";
    [[GameFunManager defaultManager] facebookShareMessage:obj completion:^(BOOL success,
NSError * _Nullable error) {
        if (success) {
            GAMFUNHUD_SUCCESS(@"sent the message");
        } else {
            if (error) {
                GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
            }
        }
    }
}

```

```

    }];
}

```

7.2.2 Share To WhatsApp

Method	whatsAppShareMessage: completion:
Params	
Message	1.GAMFUNSocialWAInvite contentString // whatsApp message text
	2.GAMFUNSocialWAImages photos //NSArray image <UIImage>
completion	GAMFUNShareCompleteHandler

example:

```

// share invite link to whatsapp
- (IBAction)whatsAppShare:(id)sender {
    GAMFUNSocialWAInvite *obj = [[GAMFUNSocialWAInvite alloc] init];
    obj.contentString = @"play games";
    [[GameFunManager defaultManager] whatsAppShareMessage:obj completion:^(BOOL success,
NSError * _Nullable error) {
        if (success) {
        } else {
            if (error) {
                GAMFUNDHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
            }
        }
    }];
}

// share picture to whatsapp
- (IBAction)wapic:(id)sender {
    GAMFUNSocialWAImages *obj = [[GAMFUNSocialWAImages alloc] init];
    UIImage *image = [UIImage imageNamed:@"game_logo_1024"];
    obj.photos = @[image];
}

```

```

[[GameFunManager defaultManager] whatsappShareMessage:obj completion:^(BOOL success,
NSError * _Nullable error) {

    if (success) {

    } else {

        if (error) {

            GAMFUNHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);

        }

    }

}];
}

```

7.3 Share (Botim)

7.3.1 Get Botim Friends

Method	getFriendsFromBotimCompletion
Callback	GAMFUNBimFriendsCompletionHandler

example:

```

- (void)getFriends {

    [[GameFunManager defaultManager] getFriendsFromBotimCompletion:^(NSArray * _Nullable
list, NSError * _Nullable error) {

        // list of botim friends

        // user model GAMFUNBotimFriendModel

    }];

}

```

7.3.2 Get Botim Playing Game Friends

Method	getPlayingFriendsFromBotimCompletion
Callback	GAMFUNBimFriendsCompletionHandler

example:

```
- (void)getFriends {

    [[GameFunManager defaultManager] getPlayingFriendsFromBotimCompletion:^(NSArray *
_Nullable list, NSError * _Nullable error) {

        // list of botim friends

        // user model GAMFUNBotimPlayingFriendModel

    }];
}
```

7.3.3 Share Botim Message

Method	sendToMessageToBotim:(id)message Completion:
Callback	GAMFUNShareCompleteHandler

example:

```
- (void)sendMessageToFriend {

    GAMFUNSocialBimMessage *model = [[GAMFUNSocialBimMessage alloc] init];

    model.bimUid = @"xxxxxxxxx";

    model.contentString = @"come and join us";

    model.title = @"play game";

    model.imageUrl =

@"http://t8.baidu.com/it/u=1484500186,1503043093&fm=79&app=86&f=JPEG?w=1280&h=853";

    [[GameFunManager defaultManager] sendToMessageToBotim:model Completion:^(BOOL success,
NSError * _Nullable error) {

        }];
}
```

8. API: Push

8.1 Register

in AppDelegate:

```

- (void)application:(UIApplication *)application
didRegisterForRemoteNotificationsWithDeviceToken:(NSData *)deviceToken {
    [[GameFunManager defaultManager] registerDeviceToken:deviceToken];
}

```

8.2 Receive Push Message

Method	getRemoteNotification
Callback	GAMFUNNotificationCompletionHandler

example:

```

- (void)recieveMessage {
    [[GameFunManager defaultManager] getRemoteNotification:^(id _Nullable result,
NSError * _Nullable error) {
        //Receive a push message and can customize the event based on the message.
        //Do something ...
    }];
}

```

9. API: Track

9.1 Field Description

GAMFUNTrackEvent (NS_ENUM)

Params	Note
GAMFUNTrackEvent_Activity	activity event
GAMFUNTrackEvent_Guide	guide event
GAMFUNTrackEvent_GuideCompletion	guide completion event
GAMFUNTrackEvent_Reward	reward event
GAMFUNTrackEvent_Rank	rank event

GAMFUNTrackEvent_VipInfo	vip info event
GAMFUNTrackEvent_AdClick	ad click event
GAMFUNTrackEvent_Other	other event

Track Event Param Keys

Keys Define	Values
GAMFUNTrackParamActivityId	af_activity_id
GAMFUNTrackParamActivityAction	af_activity_action
GAMFUNTrackParamGuideId	af_tutorial_id
GAMFUNTrackParamGuideDetail	af_tutorial_detail
GAMFUNTrackParamGuideSuccess	af_success
GAMFUNTrackParamRewardId	af_reward_id
GAMFUNTrackParamUserVip	af_user_vip
GAMFUNTrackParamAdId	af_ad_id

9.2 Track Event

Method	tracker_event: withValues:
--------	----------------------------

example:

```

- (IBAction)activity:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_Activity
withValues:@{GAMFUNTrackParamActivityId:@"activity1",
              GAMFUNTrackParamActivityAction:@"click"}];
}

- (IBAction)guide1:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_Guide
withValues:@{GAMFUNTrackParamGuideId:@"guide_test1",
              GAMFUNTrackParamGuideDetail:@"guide1"}];
}

- (IBAction)guide2:(id)sender {

```

```

[[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_Guide
withValues:@{GAMFUNTrackParamGuideId:@"guide_test2",
              GAMFUNTrackParamGuideDetail:@"guide2"}];
}

- (IBAction)guideCompletion:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_GuideCompletion
withValues:@{GAMFUNTrackParamGuideId:@"guide_test",
              GAMFUNTrackParamGuideSuccess:YES,
              GAMFUNTrackParamGuideDetail:@"test_finish"}];
}

- (IBAction)reward:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_Reward
withValues:@{GAMFUNTrackParamRewardId:@"reward_222"}];
}

- (IBAction)rank:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_Rank
withValues:@{@"af_rank":@"R0"}];
}

- (IBAction)vip:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_VipInfo
withValues:@{GAMFUNTrackParamUserVip:@"Vip5"}];
}

- (IBAction)ad:(id)sender {
    [[GameFunManager defaultManager] tracker_event:GAMFUNTrackEvent_AdClick
withValues:@{GAMFUNTrackParamAdId:@"ad_222"}];
}

```

9.3 Track Event(Customized)

Method	tracker_eventName: withValues:
--------	--------------------------------

10. API: Customer Service

10.1 Contact us With Messenger

Method	contactusWithMessenger
--------	------------------------

example:

```
[[GameFunManager defaultManager] contactusWithMessenger];
```