

TTkGameSDK API Document

(iOS)

| Name | Date | version | remark |
|---------|-----------|---------|---|
| Yuanjun | 2020.3.18 | 1.0.0 | created |
| Yuanjun | 2020.3.27 | 1.0.1 | improvements |
| Yuanjun | 2020.3.31 | 1.0.2 | add game invite |
| Yuanjun | 2020.4.07 | 1.0.3 | add single sign-on |
| Yuanjun | 2020.4.12 | 1.0.4 | improvements |
| Yuanjun | 2020.4.17 | 1.0.5 | 1.launch animation deleted (2.1.1) 2.add agreement view (4.1) & delete 4.2 of last version 3.Domain Changed 4.add login with apple (4.2) |
| Yuanjun | 2020.4.27 | 1.0.6 | improvements |
| Yuanjun | 2020.5.13 | 1.0.7 | improvements |
| Yuanjun | 2020.5.27 | 1.0.8 | SDK rename 1.GameManager class rename with TTkGameManager 2.TTGCLoginType rename with TTGCLoginType_TTk |
| Yuanjun | 2020.5.29 | 1.1.0 | improvements |
| Yuanjun | 2020.6.24 | 1.1.1 | 1. add ttk login 2. add twitter login (4.5) 3. add ttk friends and share |
| Yuanjun | 2020.7.06 | 1.1.2 | add notification of account banned (1.3.11 & 4.11) |
| Yuanjun | 2020.7.10 | 1.1.3 | 1. fix AppsFlyer 2. get available token (4.12) |
| Yuanjun | 2020.7.14 | 1.1.4 | bug fix |

| | | | |
|---------|-----------|--------|---|
| Yuanjun | 2020.7.15 | 1.1.5 | add thirdUid in TTGCUserModel (4.8) |
| Yuanjun | 2020.7.20 | 1.1.6 | 1. get product detail (6.3) |
| Yuanjun | 2020.7.22 | 1.1.7 | fix bug of SDK codeSign |
| Yuanjun | 2020.7.27 | 1.1.8 | improvements |
| Yuanjun | 2020.8.03 | 1.1.9 | 1. login with Botim (4.6) 2. botim friends and share (7.3) |
| Yuanjun | 2020.8.04 | 1.1.10 | improvements |

1. Description and definition

1.1 API call

Import <TTkGameSDK/TTkGameSDK.h>

Every API is called by [TTkGameManager 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: TTGCUserCompletionHandler
fields:
userInfo //the user information
error
```

1.3.2 Receive Notification Callback

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

1.3.3 Logout Callback

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

1.3.4 Get Friends Callback

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

1.3.5 Share Callback

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

1.3.6 Payment Callback

```
class: TTGCOrderProgressHandler  
fields:  
TTGCOderStatus //Payment order status
```

1.3.7 Payment Result Callback

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

1.3.8 AppVersion Info Callback

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

1.3.9 Common Callback

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

1.3.10 Account Kicked Callback

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

1.3.11 Account Banned Callback

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

1.4 Errors

| | |
|---|-----------------------------------|
| TTGCPlatformErrorType_Unknown | unknown error |
| TTGCPlatformErrorType_AuthorizeFailed | authorized failed |
| TTGCPlatformErrorType_ForUserInfoFailed | request user data failed |
| TTGCPlatformErrorType_LoginFailed | login failed |
| TTGCPlatformErrorType_FriendsListFailed | get friends failed |
| TTGCPlatformErrorType_MessageSendFailed | send message failed |
| TTGCPlatformErrorType_PayFailed | payment failed |
| TTGCPlatformErrorType_ShareFailed | share failed |
| TTGCPlatformErrorType_NotInstall | app is not installed |
| TTGCPlatformErrorType_NotNetwork | network exception |
| TTGCPlatformErrorType_SourceError | third error |
| TTGCPlatformErrorType_NoPermission | no permissions |
| TTGCPlatformErrorType_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

    [[TTGameManager defaultManager] setServerDomainTest];
}
```

```

// set Pay Enviroment

[[TTkGameManager defaultManager] setPayEnvironmentTest];

// open log

[[TTkGameManager defaultManager] openLogInfo];


// launch SDK

NSString *appId = @"xxxxx";

NSString *appSecret = @"xxxxxx";

NSString *gameid = @"xxxxxxx";

[[TTkGameManager 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 [[TTkGameManager defaultManager] application:app openURL:url
options:options];
}

```

2.1.3 App Events:

```

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

    [[TTkGameManager defaultManager] applicationDidEnterBackground:application];
}

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

    [[TTkGameManager defaultManager] applicationWillEnterForeground:application];
}

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

    [[TTkGameManager defaultManager] applicationDidBecomeActive:application];
}

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

    [[TTkGameManager defaultManager] applicationWillTerminate:application];
}

```

2.1.4 Push Register:

```

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

- (void)application:(UIApplication *)application
didReceiveRemoteNotification:(NSDictionary *)userInfo fetchCompletionHandler:(void
(^)(UIBackgroundFetchResult))completionHandler {
    [[TTkGameManager 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 [[TTkGameManager defaultManager] application:application
continueUserActivity:userActivity restorationHandler:restorationHandler];
}

```

2.2 Check App Version

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 {
    [[TTkGameManager defaultManager] checkAppVersionCompletion:^(BOOL hasNewVersion,
TTGCApVersionModel * _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
            }
        }
    }
}

```

```

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

        } else {

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

        }

    }

    }];
}

```

2.2.1 Check Version

| | |
|-----------------|---|
| Method | checkAppVersionCompletion |
| Callback | TTGCVersionCompletionHandler hasNewVersion // has new version <BOOL> versionInfo // the version detail information. class model <TTGCApVersion> 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

```

[[TTkGameManager defaultManager] openLogInfo];

```


3.2 Disable Log Output(default)

```
[[TtkGameManager defaultManager] closeLogInfo];
```

3.3 Set Test Environment

```
[[TtkGameManager defaultManager] setServerDomainTest];
```

3.4 Set Production Environment(default)

```
[[TtkGameManager defaultManager] setServerDomainProduction];
```

3.5 Set Sandbox Payment

```
[[TtkGameManager defaultManager] setPayEnvironmentTest];
```

3.6 Set Production Payment(default)

```
[[TtkGameManager defaultManager] setPayEnvironmentProduction];
```

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 {  
    [[TtkGameManager defaultManager] showAgreementView];  
}
```

4.2 Login with Apple

Login with Apple only supports iOS13 and higher.

| | |
|-----------------|----------------------------|
| Method | loginWithAppleCompletion |
| Callback | TTGCUUserCompletionHandler |

example:

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

    __weak __typeof(self) weakSelf = self;

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

            [weakSelf closeLoginView];
        } else {
            //error info
            TTGCHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
        }
    }];
}
```

4.3 Login with GameCenter

| | |
|-----------------|-------------------------------|
| Method | loginWithGameCenterCompletion |
| Callback | TTGCUUserCompletionHandler |

example:

```
- (void)gamecenterLogin {
    TTGCHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

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

```

        if (!error) {

            //login success

            TTGCHUD_HINT(@"success")

            [weakSelf closeLoginView];

        } else {

            //error info

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

        }

    }

}];
}

```

4.4 Login with Facebook

| | |
|-----------------|-----------------------------|
| Method | loginWithFacebookCompletion |
| Callback | TTGCUUserCompletionHandler |

example:

```

- (void)facebookLogin {

    TTGCHUD_NO_Stop(@"login...")

    __weak __typeof(self)weakSelf = self;

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

        if (!error) {

            //login success

            TTGCHUD_HINT(@"success")

            [weakSelf closeLoginView];

        } else {

            //error info

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

        }

    }

}];
}

```

4.5 Login with Twitter

| | |
|-----------------|----------------------------|
| Method | loginWithTwitterCompletion |
| Callback | TTGCUUserCompletionHandler |

example:

```
- (IBAction)twitterLogin:(id)sender {
    TTGCHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

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

            TTGCHUD_SUCCESS(@"success")

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

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

4.6 Login with Botim

| | |
|-----------------|----------------------------|
| Method | loginWithBotimCompletion |
| Callback | TTGCUUserCompletionHandler |

example:

```
- (IBAction)botimLogin:(id)sender {
    if ([[TTkGameManager defaultManager] botimIsInstall]) {
        TTGCHUD_NO_Stop(@"login...")

        __weak __typeof(self) weakSelf = self;

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

```

        if (!error) {

            //login success

            TTGCHUD_SUCCESS(@"success");

            [weakSelf closeLoginView];

        } else {

            //error info

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

        }

    }

}];

} else {

    // no app please hide login button

}

}

```

4.7 Login with Guest

| | |
|-----------------|---------------------------|
| Method | guestLoginCompletion |
| Callback | TTGCUserCompletionHandler |

example

```

- (void)guestLogin {

    TTGCHUD_NO_Stop(@"login...")

    __weak __typeof(self) weakSelf = self;

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

        if (!error) {

            //login success

            TTGCHUD_HINT(@"success")

            [weakSelf closeLoginView];

        } else {

            //error info

            TTGCHUD_Stop

        }

    }

}];

```

```
}
```

4.8 Get User Info

| | |
|----------|--|
| Method | userInfoCompletion |
| Callback | TTGCUUserCompletionHandler |
| Params | <div>TTGCUUserModel</div> <div><div>userType</div><div>user type (TTGCLoginType by 3.7)</div></div> <div><div>userId</div><div>user's identifier</div></div> <div><div>thirdUid</div><div>third-party user identifier</div></div> <div><div>nickname</div><div>nickname</div></div> <div><div>photoUrl</div><div>head image url</div></div> <div><div>pushOff</div><div>push switch 0:push switch on 1:push switch off</div></div> |

example

```
- (void)getUserInfo {  
    TTGCHUD_NO_Stop(@"")  
    [[TTkGameManager defaultManager] userInfoCompletion:^(id _Nullable userInfo,  
NSError * _Nullable error) {  
        if (!error) {  
            TTGCHUD_HINT(@"success")  
            TTGCUUserModel *model = userInfo;  
            [self freshUserInfo:model];  
        } else {  
            //error info  
            TTGCHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);  
        }  
    }  
};  
}
```

4.9 Get Login Type

| | |
|--------|-----------|
| Method | loginType |
|--------|-----------|

| | |
|---------------|---------------|
| Return | TTGCLoginType |
|---------------|---------------|

Definition :

| | |
|--------------------------|---------------------|
| Enum | TTGCLoginType |
| Description | Enum type for Login |
| Enum Values | |
| TTGCLoginType_unloggedIn | not login |
| TTGCLoginType_TTk | TTk user |
| TTGCLoginType_GameCenter | GameCenter player |
| TTGCLoginType_Facebook | Facebook user |
| TTGCLoginType_Twitter | Twitter user |
| TTGCLoginType_Guest | Game guest |

example :

```
- (void)showLoginView {
    if ([[TTkGameManager defaultManager] loginType] == TTGCLoginType_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];
        });
    }
}
```

4.10 Notification of offline

| | |
|-----------------|---------------------------|
| Method | accountKickedNotification |
| Callback | TTGCAccountKickedHandler |

example:

```
[[TTkGameManager defaultManager] accountKickedNotification:^(NSString * _Nullable
information) {

    //Your account has been logged in on another device, please login again.

    TTGCHUD_HINT(information);

    [self showLoginView];

}];
```

4.11 Notification of banned

| | |
|-----------------|---------------------------|
| Method | accountBannedNotification |
| Callback | TTGCAccountBannedHandler |

example:

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

    //Your account has been banned.

}];
```

4.12 Get Available Token

| | |
|-----------------|---------------------------|
| Method | getAvailableAccessToken |
| Callback | TTGCUserCompletionHandler |

example:

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

    if (userInfo) {

        TTGCUserModel *userModel = userInfo;

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

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

    }

}];
```


5. API: Logout

| | |
|----------|---------------------------|
| Method | logout |
| Callback | TTGCLogoutCompleteHandler |

example:

```
- (IBAction)logout:(id)sender {
    TTGCHUD_NO_Stop(@"logout...")

    [[TTkGameManager defaultManager] logout:^(BOOL success, NSError * _Nullable error) {
        if (success) {
            TTGCHUD_HINT(@"success")
            [self showLoginView];
        } else {
            //error info
            NSLog(@"%@",error.userInfo);
            TTGCHUD_HINT([error.userInfo objectForKey:@"errorMsg"]);
        }
    }];
}
```

6. API: Game Payment

6.1 Product Purchase

Order Status Definition:

| | |
|----------------------------------|---------------------------|
| TTGCOderStatus_ProductQuerying | query product |
| TTGCOderStatus_Generating | order init |
| TTGCOderStatus_ProductPurchasing | order in trading progress |
| TTGCOderStatus_ProductPurchased | order purchased |

| | |
|--------------------------------|-----------------|
| TTGCOderStatus_ReceiptChecking | order verifying |
|--------------------------------|-----------------|

| | |
|------------|--|
| Method | buyProductWithSKU: Progress: Completion: |
| Params | |
| sku | String,product id |
| Progress | TTGCOderProgressHandler |
| Completion | TTGCOderCompletionHandler <pre> class TTGCOderModel { 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> |

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

example:

```

- (IBAction)pay:(id)sender {
    TTGCHUD_NO_Stop(@"pay...")

    [[TTkGameManager defaultManager] buyProductWithSKU:@"LangChaoJD001"]

    Progress:^(TTGCOderStatus orderStatus) {

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

```

```

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

        if (orderInfo) {

            TTGCHUD_HINT(@"success")

        }

        if (error) {

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

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

        }

    }

}];
}

```

6.2 Query Order

| | |
|------------|---|
| Method | queryOrderWithOrderId: Completion: |
| Params | |
| orderId | the order's id |
| Completion | TTGCOderCompletionHandler TTGCOderModel by 6.1 |

example:

```

- (IBAction)query:(id)sender {

    [[TTkGameManager 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 | TTGCPProductCompletionHandler <pre> class TTGCPProductModel 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. </pre> |

example:

```

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

    [[TTkGameManager defaultManager] getProductDetailWithSKU:sku
Completion:^(id _Nullable productInfo, NSError * _Nullable error) {

    if (productInfo) {

        TTGCPProductModel *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

TTGCBotimFriendModel

| Params | Note |
|--------|------|
|--------|------|

| | |
|----------|------------------|
| bimUid | botim user id |
| name | friend's name |
| photoUrl | friends' avatars |

TTGCBotimPlayingFriendModel

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

TTGCSocialBimMessage

| 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.TTGCSocialFBInvite quote //message text |
| | 2.TTGCSocialFBImages photoImage //photo image <UIImage> |
| | 3.TTGCSocialFBFriendsGameInvite inviteString //invite message text |
| completion | TTGCShareCompleteHandler |

example:

```
// share invite link to facebook

- (IBAction)fbShare:(id)sender {

    TTGCSocialFBInvite *obj = [[TTGCSocialFBInvite alloc] init];

    obj.quote = @"Join us";

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

        if (success) {

        } else {

            if (error) {

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

            }

        }

    }];

}

//share image to facebook

- (IBAction)fbpic:(id)sender {

    UIImage *image = [UIImage imageNamed:@"game_logo_1024"];

    TTGCSocialFBImages *obj = [[TTGCSocialFBImages alloc] init];

    obj.photoImage = image;

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

        if (success) {

        } else {

            if (error) {

                TTGCHUD_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 {
```

```

TTGCSocialFBFriendsGameInvite *obj = [[TTGCSocialFBFriendsGameInvite alloc] init];
obj.inviteString = @"join us";

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

    if (success) {

        TTGCHUD_SUCCESS(@"sent the message");

    } else {

        if (error) {

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

        }

    }

}];
}

```

7.2.2 Share To WhatsApp

| | |
|------------|--|
| Method | whatsAppShareMessage: completion: |
| Params | |
| Message | 1.TTGCSocialWAInvite contentString // whatsApp message text |
| | 2.TTGCSocialWAIImages photos //NSArray image <UIImage> |
| completion | TTGCShareCompleteHandler |

example:

```

// share invite link to whatsapp
- (IBAction)whatsAppShare:(id)sender {

    TTGCSocialWAInvite *obj = [[TTGCSocialWAInvite alloc] init];

    obj.contentString = @"play games";

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

        if (success) {

        } else {

            if (error) {


```

```

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

}

}];

}

// share picture to whatsapp
- (IBAction)wapic:(id)sender {

    TTGCSocialWAImages *obj = [[TTGCSocialWAImages alloc] init];

    UIImage *image = [UIImage imageNamed:@"game_logo_1024"];

    obj.photos = @[image];

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

        if (success) {

        } else {

            if (error) {

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

            }

        }

    }];

}

```

7.3 Share (Botim)

7.3.1 Get Botim Friends

| | |
|----------|---------------------------------|
| Method | getFriendsFromBotimCompletion |
| Callback | TTGCBimFriendsCompletionHandler |

example:

```

- (void)getFriends {

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

        // list of botim friends

        // user model TTGCBotimFriendModel

```



```

    }];
}

```

7.3.2 Get Botim Playing Game Friends

| | |
|----------|--------------------------------------|
| Method | getPlayingFriendsFromBotimCompletion |
| Callback | TTGCBimFriendsCompletionHandler |

example:

```

- (void)getFriends {

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

        // list of botim friends

        // user model TTGCBotimPlayingFriendModel

    }];
}

```

7.3.3 Share Botim Message

| | |
|----------|--|
| Method | sendToMessageToBotim:(id)message Completion: |
| Callback | TTGCShareCompleteHandler |

example:

```

- (void)sendMessageToFriend {

    TTGCSocialBimMessage *model = [[TTGCSocialBimMessage 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";

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

```

```
    }];  
}
```

8. API: Push

8.1 Register

in AppDelegate:

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

8.2 Receive Push Message

| | |
|----------|-----------------------------------|
| Method | getRemoteNotification |
| Callback | TTGCNotificationCompletionHandler |

example:

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