

# ToTokGameSDK 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

## 1. Description and definition

### 1.1 API call

Every API is called by `[ToTokGameManager 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: TTGCOOrderProgressHandler  
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: TTGCCCompletionHandler  
fields:  
result //operation result  
error
```

## 1.4 Errors

TTGCPlatformErrorType_Unknown	unkown error
TTGCPlatformErrorType_AuthorizeFailed	authorized failed
TTGCPlatformErrorType_ForUserInfoFailed	request user data failed
TTGCPlatformErrorType_LoginFailed	login failed
TTGCPlatformErrorType_FriendsListFailed	get firends 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

## 2. Setup Initialize (must)

### 2.1 Launch

Methods in AppDelegate are implement like below:

#### 2.1.1 Setup Configuration:

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

    // set SDK Api Enviroment

    [[ToTokGameManager defaultManager] setServerDomainTest];

    // open log

    [[ToTokGameManager defaultManager] openLogInfo];

    // launch SDK

    NSString *appId = @"xxxxx";
    NSString *appSecret = @"xxxxxx";
    NSString *gameid = @"xxxxxxx";

    [[ToTokGameManager defaultManager] setUpWithAppId:appId Secret:appSecret
GameId:gameid Application:application Options:launchOptions AnimationHandle:^(BOOL
isFinish) {

        //launch animation finished and load your UI

    }];

    return YES;
}
```

### 2.1.2 External Callback:

```
- (BOOL)application:(UIApplication *)app openURL:(NSURL *)url
options:(NSDictionary<UIApplicationOpenURLOptionsKey,id> *)options {
    return [[ToTokGameManager defaultManager] application:app openURL:url
options:options];
}
```

### 2.1.3 App Events:

```
- (void)applicationDidEnterBackground:(UIApplication *)application {
    [[ToTokGameManager defaultManager] applicationDidEnterBackground:application];
}

- (void)applicationWillEnterForeground:(UIApplication *)application {
    [[ToTokGameManager defaultManager] applicationWillEnterForeground:application];
}

- (void)applicationDidBecomeActive:(UIApplication *)application {
    [[ToTokGameManager defaultManager] applicationDidBecomeActive:application];
}

- (void)applicationWillTerminate:(UIApplication *)application {
    [[ToTokGameManager defaultManager] applicationWillTerminate:application];
}
```

### 2.1.4 Push Register:

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

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

    [[ToTokGameManager 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

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

            } else {

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

            }

        }

    }
};
}

```

### 2.2.1 Check Version

Method	checkAppVersionCompletion
--------	---------------------------

<b>Callback</b>	TTGCVersionCompletionHandler
	hasNewVersion // has new version <BOOL>
	versionInfo // the version detail information. class model <TTGCAppVersion>
	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
<b>Params</b>	identifier //App Store item identifier (NSNumber) of the product

## 3. API: Configuration

### 3.1 Enbale Log Output

```
[[ToTokGameManager defaultManager] openLogInfo];
```

### 3.2 Disable Log Output(default)

```
[[ToTokGameManager defaultManager] closeLogInfo];
```

### 3.3 Set Test Environment

```
[[ToTokGameManager defaultManager] setServerDomainTest];
```

### 3.4 Set Production Environment(default)

```
[[ToTokGameManager defaultManager] setServerDomainProduction];
```

## 3.5 Set Sandbox Payment

```
[[ToTokGameManager defaultManager] setPayEnvironmentTest];
```

## 3.6 Set Production Payment(default)

```
[[ToTokGameManager defaultManager] setPayEnvironmentTest];
```

# 4. API: Login&User Info

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

## 4.1 User Agreement Url

```
NSString *urlString = [[ToTokGameManager defaultManager] getAgreementUrlString];
```

## 4.2 Privacy Url

```
NSString *urlString = [[ToTokGameManager defaultManager] getPrivacyUrlString];
```

## 4.3 Login with GameCenter

Method	loginWithGameCenterCompletion
Callback	TTGCUserCompletionHandler

**example:**

```
- (void)gamecenterLogin {  
    TTGCHUD_NO_Stop(@"login...")  
    __weak __typeof(self) weakSelf = self;
```



```

[[ToTokGameManager 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
<b>Callback</b>	TTGCUUserCompletionHandler

### example:

```

- (void)facebookLogin {

    TTGCHUD_NO_Stop(@"login...")

    __weak __typeof(self)weakSelf = self;

    [[ToTokGameManager 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 Guest

Method	guestLoginCompletion
<b>Callback</b>	TTGCUserCompletionHandler

### example

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

    __weak __typeof(self)weakSelf = self;

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

            TTGCHUD_HINT(@"success")

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

            TTGCHUD_Stop
        }
    }];
}
```

## 4.6 Get User Info

Method	userInfoCompletion
<b>Callback</b>	TTGCUserCompletionHandler
<b>Params</b>	TTGCUserModel  userType                      user type (TTGCLoginType by 3.7) userId                        user's identifier nickname                      nickname photoUrl                      head image url

	pushOff	push switch 0:push switch on 1:push switch off
--	---------	--

## example

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

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

            TTGCUserModel *model = userInfo;

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

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

# 4.7 Get Login Type

Method	loginType
<b>Return</b>	TTGCLoginType

Definition:

Enum	TTGCLoginType
Description	Enum type for Login
Enum Values	
TTGCLoginType_unloggedIn	not login
TTGCLoginType_ToTok	ToTok user
TTGCLoginType_GameCenter	GameCenter player
TTGCLoginType_Facebook	Facebook user
TTGCLoginType_Guest	ToTokGame guest

### example:

```
- (void)showLoginView {

    if ([[ToTokGameManager 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];

        });

    }

}
```

## 4.8 Notification of offline

Method	accountKickedNotification
<b>Callback</b>	TTGCAccountKickedHandler

### example:

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

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

    TTGCHUD_HINT(information);

    [self showLoginView];

}];
```

## 5. API: Logout

Method	logout
<b>Callback</b>	TTGCLogoutCompleteHandler

### example:

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

    [[ToTokGameManager 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: <ToTokGame Server API Document (to Game Developer)>

### example:

```

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

    [[ToTokGameManager 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	TTGCOOrderCompletionHandler TTGCOOrderModel by 5.1

### example:

```

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

```

## 7. API: SNS & Share

### 7.1 ToTok Friends

Not yet open

## 7.2 Share

Share to Facebook, WhatsApp.

ToTok Sharing is not yet open.

### 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";
    [[ToTokGameManager 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:@"totokgame_logo_1024"];

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

    obj.photoImage = image;

    [[ToTokGameManager 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";

    [[ToTokGameManager 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.TTGCSocialWAImages 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";
    [[ToTokGameManager 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:@"totokgame_logo_1024"];
    obj.photos = @[image];
    [[ToTokGameManager defaultManager] whatsAppShareMessage:obj completion:^(BOOL success,
NSError * _Nullable error) {
        if (success) {
        } else {
        }
    }];
}
```

```

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

}];
}

```

## 8. API: Push

### 8.1 Register

in AppDelegate:

```

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

```

### 8.2 Receive Push Message

Method	getRemoteNotification
Callback	TTGCNotificationCompletionHandler

**example:**

```

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

```