# PLUG_Event Operation Guide for Immediate Rewards

PLUG

**General**

# Overview

1. If you integrate the game ID mapping function of PLUG,

[User's game IDs can be mapped with their community IDs](#)

2. If you integrate the callback API function of PLUG,

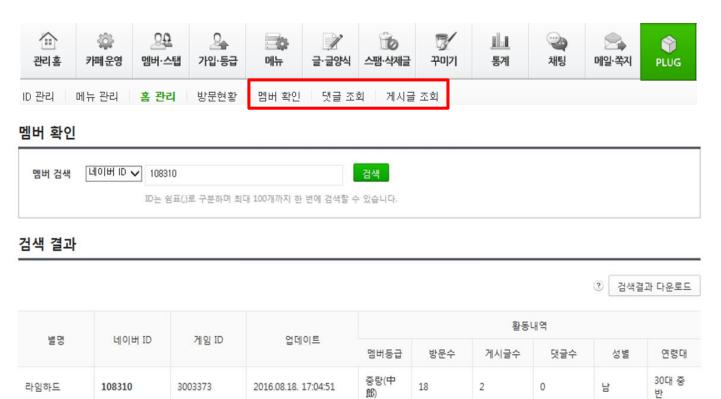[Reward for attending the community event can be offered immediately](#)

→ Not required to check matching between the community ID used to attend the event and its associated game ID

→ Additional follow-up action is not required after event completion as you immediately provide rewards

→ Immediate rewards can increase participants and thereby lead to a drastic increase in the number of signups to your community (signups increased by 40-50% during an event)

# 1-1. Map Game IDs with Cafe Nicknames_Korea

- Integrating the PLUG function (game ID mapping) enables you to map game IDs with Cafe nicknames

  ※ Integration is only available for users who completed signups or added posts/comments on PLUG inside the game app
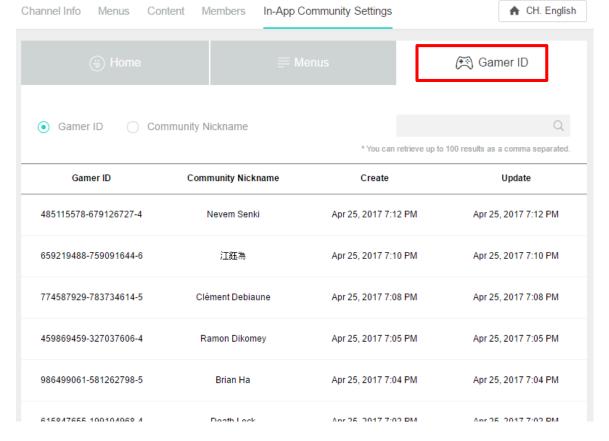
  For users who have done these actions on NAVER Cafe (either PC or mobile), integration does not work

- Integrated data can be seen in  [멤버 확인], [댓글 조회], and [게시글 조회]

  (Downloadable in Excel file format)

# 1-2. Map Game IDs with Community Nicknames_Global<sup>General</sup>

- Integrating the PLUG function (game ID mapping) enables you to map game IDs with community nicknames

  ※ Integration is only available for users who completed signups or added posts/comments on PLUG inside the game app

  For users who have done these actions on global community (https://www.plug.game), integration does not work

- Integrated data can be seen in [Gamer ID] after going to [Channel > In-App Community Settings]

# 1-3. Game ID Mapping Integration Guide

- Once integrated, it will be available for both Korean NAVER Cafe and the global community

### [Unity]

게임 아이디를 카페 아이디와 연동하려면 syncGameUserId() 메서드를 사용한다.

```csharp
void syncGameUserId(string gameUserId)
```

다음은 게임 아이디와 카페 아이디를 연동하는 예다.

```csharp
/**
 * 게임 아이디와 카페 아이디를 연동한다.
 *
 * @param gameUserId 게임 아이디
 */
GLink.sharedInstance().syncGameUserId("usergameid");
```

### [Cocos2d-x]

게임 아이디를 카페 아이디와 연동하려면 syncGameUserId() 메서드를 사용한다.

```cpp
static void syncGameUserId(std::string gameUserId)
```

다음은 게임 아이디와 카페 아이디를 연동하는 예다.

```cpp
/**
 * 게임 아이디와 카페 아이디를 연동한다.
 *
 * @param gameUserId 게임 아이디
 */
cafe::CafeSdk::syncGameUserId("usergameid");
```

### [Unreal]

게임 아이디를 카페 아이디와 연동하려면 SyncGameUserId() 메서드를 사용한다.

```cpp
static void SyncGameUserId(FString GameUserId)
```

다음은 게임 아이디와 카페 아이디를 연동하는 예다.

```cpp
/**
 * 게임 아이디와 카페 아이디를 연동한다.
 *
 * @param gameUserId 게임 아이디
 */
void ACafeSDKSampleProjectGameMode::InitGame(const FString& MapName, const FString&
Options, FString& ErrorMessage)
{
...
    UCafeSdkBlueprintLibrary::SyncGameUserId("userGameID");
...
}
```

Also provided as native code for Android and iOS

# 2-1. Event Operation Guide Using Callback API

- A callback API handles various events occurred in PLUG, including start, stop, join, post, comment, and so on

- How to operate a community event using the callback API

[Prerequisite]

- Define callbacks in actions of signup, post, or comment, which are likely to be used for the community event

→ This turns into status where whether to sign up, board ID (menuId), and post ID (articleId) can be passed as soon as a user completes the signup process or adds a post/comment on PLUG inside the game app

[Event Operations]

1. Planning (event type, period, reward)
2. Preparation

2-1. Developer: Set an event-specific and real-time reward to be delivered to user's mailbox while preventing duplicate submissions

- For signups: Set rewards to be submitted for signups

- For adding posts: Set the ID (menuId) of a board to be used for the event and compare two IDs (the board ID and the board ID in which the post is added) if they match each other. Then, submit a reward if two IDs are identical.

- For adding comments: Set the ID (articleId) of a post to be used for the event and compare two IDs (post ID and the post ID in which the comment is added) if they match each other. Then, submit a reward if two IDs are identical.

2-2. Community Manager: Create an event announcement in the board including the fact that real-time reward will be provided. It needs to be emphasized that rewards will be available only if entering the event is made through PLUG inside the game app

2-3. Game User: See the reward sent to their mailbox after attending an event

# 2-2. Operational Case Using Callback API_Korea



Attracting at least 10,000 users with callback

# 2-3. Operational Case Using Callback API_Global

**Endless Frontier**
👥 134,776

● Event results of callback for signups

- 20,000 community users in its first three days

- 130,000 community users in 4 months since its release



ef.ekkorr 매니저 · 2017년 4월 6일 (목)

## In-game Community Membership Event

Thanks for choosing Ekkorr. Join the in-game community to receive 3,000 gems and a 5-Star voucher in your inbox. Hope you all have another great day! Thank you.

👁 274349                           ♡ 7,364    💬 27088   ↱

https://www.plug.game/endlessFrontier-en/events

# 2-4. Callback API Integration Guide

※ The callback API is described as "listener" in the Android guide while "delegate" in the iOS guide



한국어    https://plug.gitbooks.io/plug-sdk-android/ko/[%ED%95%9C]-Callback-Listener.html

English    https://plug.gitbooks.io/plug-sdk-android/en/[EN]-Callback-Listener.html

中文    https://plug.gitbooks.io/plug-sdk-android/zh/[ZH]-Callback-Listener.html

日本語    https://plug.gitbooks.io/plug-sdk-android/ja/[JA]-%E3%82%B3%E3%83%BC%E3%83%AB%E3%83%90%E3%83%83%E3%82%AF%E3%83%AA%E3%82%B9%E3%83%8A%E3%83%BC.html



한국어    https://plug.gitbooks.io/plug-sdk-ios/content/ko/[%ED%95%9C]-Callback-Delegate.html

English    https://plug.gitbooks.io/plug-sdk-ios/en/[EN]-Callback-Delegate.html

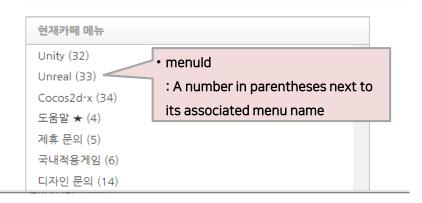中文    https://plug.gitbooks.io/plug-sdk-ios/zh/[ZH]-Callback-Delegate.html

日本語    https://plug.gitbooks.io/plug-sdk-ios/ja/[JA]-Callback-Delegate.html

■ Korean NAVER Cafe



• menuId
: A number in parentheses next to its associated menu name

• articleId
: A number before the post title

# Reference. How to Find "menuId" and "articleId"_Global

■ PLUG



- menuId

  : A number before the menu name

- articleId

  : A number before the post