|  |  |
| --- | --- |
| NAVER Cafe SDK Integration Guide | |
|  |  |

|  |
| --- |
| Copyright |

Copyright © NAVER Corp. All Rights Reserved.

This document is an intellectual property of NAVER Corp.; unauthorized reproduction or distribution of this document, or any portion of it is prohibited by law.

This document is provided for information purpose only. NAVER Corp. has endeavored to verify the completeness and accuracy of information contained in this document, but it does not take the responsibility for possible errors or omissions in this document. Therefore, the responsibility for the usage of this document or the results of the usage falls entirely upon the user, and NAVER Corp. does not make any explicit or implicit guarantee regarding this.

Software products or merchandises mentioned in this document, including relevant URL information, conform to the copyright laws of their respective owners. The user is solely responsible for any results occurred by not complying with applicable laws.

NAVER Corp. may modify the details of this document without prior notice.

|  |
| --- |
| About This Document |

Purpose

This document describes how to integrate the NAVER Cafe SDK into applications.

Audience

This document is intended for developers who want to integrate the NAVER Cafe SDK into applications.

Contact Us

For any errors or inquiries regarding this document, please contact us via email.

Email: [dl\_gamesdkpartner@navercorp.com](mailto:dl_gamesdkpartner@navercorp.com)

Revision History

|  |  |
| --- | --- |
| Date | Description |
| 2015. 12. 23. | First official release of this document |

|  |
| --- |
| Conventions |

Notes

Note

A note describes useful information for users.

Cautions

Caution

A caution describes information that users should know in order to prevent system errors.

User Interface (UI) Text and User Input

* UI text: Use bold formatting like Top Menu > Sub Menu.
* User input: Use bold formatting like localhost.
* Replaceable user input: Use braces to enclose like http://www.naver.com/{company\_name}.

Source Code

Text for source code is written in black on a gray background.

COPYDATASTRUCT st;

st.dwData = PURPLE\_OUTBOUND\_ENDING;

st.cbData = sizeof(pp);

st.lpData = &pp;

::SendMes(GetTargetHwnd(), WM\_COPYDATA, (WPARAM)this->m\_hWnd, (LPARAM)&st);

|  |
| --- |
| Table of Contents |

[Overview 9](#_Toc438464763)

[Overview of the NAVER Cafe SDK 9](#_Toc438464764)

[Features and Characteristics of NAVER Cafe SDK 9](#_Toc438464765)

[Main Features 9](#_Toc438464766)

[Requirement 9](#_Toc438464767)

[For Android 11](#_Toc438464768)

[Development Environment 11](#_Toc438464769)

[Requirement 11](#_Toc438464770)

[Library File Structure 11](#_Toc438464771)

[Development Environment Setup 12](#_Toc438464772)

[How to Integrate 12](#_Toc438464773)

[Configure the AndroidManifest.xml File 12](#_Toc438464774)

[Initialize the NAVER Cafe SDK 12](#_Toc438464775)

[Start the NAVER Cafe SDK Activity 13](#_Toc438464776)

[Close the NAVER Cafe SDK Activity 13](#_Toc438464777)

[Compose Post 13](#_Toc438464778)

[Execute App Scheme 14](#_Toc438464779)

[Implement User ID Mapping 14](#_Toc438464780)

[Change Resource Image 14](#_Toc438464781)

[API Reference 14](#_Toc438464782)

[Glink 14](#_Toc438464783)

[init() 15](#_Toc438464784)

[isShowGlink() 15](#_Toc438464785)

[popBackStack() 16](#_Toc438464786)

[setGameUserId() 16](#_Toc438464787)

[startEvent() 16](#_Toc438464788)

[startImageWrite() 17](#_Toc438464789)

[startHome() 17](#_Toc438464790)

[startMenu() 18](#_Toc438464791)

[startNotice() 18](#_Toc438464792)

[startProfile() 19](#_Toc438464793)

[startVideoWrite() 19](#_Toc438464794)

[startWrite() 20](#_Toc438464795)

[stop() 20](#_Toc438464796)

[For iOS 21](#_Toc438464797)

[Development Environment 21](#_Toc438464798)

[Requirement 21](#_Toc438464799)

[Library File Structure 21](#_Toc438464800)

[Development Environment Setup 21](#_Toc438464801)

[How to Integrate 24](#_Toc438464802)

[Initialize the NAVER Cafe SDK 24](#_Toc438464803)

[Configure NAVER Login 24](#_Toc438464804)

[Start the NAVER Cafe SDK Activity 25](#_Toc438464805)

[Compose Post 25](#_Toc438464806)

[API Reference 25](#_Toc438464807)

[NCSDKManager 25](#_Toc438464808)

[@property (nonatomic, weak) id parentViewController 25](#_Toc438464809)

[(void)dismissViewController 26](#_Toc438464810)

[(void)dismissTopViewController 26](#_Toc438464811)

[(NCSDKManager \*)getSharedInstance 27](#_Toc438464812)

[(id)navercafeRootViewController 27](#_Toc438464813)

[(void)resetSharedInstance 27](#_Toc438464814)

[(void)presentArticlePostViewControllerWithMenuId 28](#_Toc438464815)

[(void)presentArticlePostViewControllerWithType 28](#_Toc438464816)

[(void)presentMainViewController 29](#_Toc438464817)

[(void)presentMainViewControllerWithArticleId 29](#_Toc438464818)

[(void)presentMainViewControllerWithTabIndex 30](#_Toc438464819)

[(void)presentViewController 30](#_Toc438464820)

[(void)setGameUserId 30](#_Toc438464821)

[(void)setNaverLoginClientId 31](#_Toc438464822)

[NCSDKLoginManager 31](#_Toc438464823)

[@property (nonatomic, weak) UIViewController \*rootViewController 32](#_Toc438464824)

[(NSString \*)accessToken 32](#_Toc438464825)

[(NSString \*)accessTokenExpireTime 32](#_Toc438464826)

[(BOOL)finishNaverLoginWithURL 33](#_Toc438464827)

[(NCSDKLoginManager \*)getSharedInstance 33](#_Toc438464828)

[(void)isLoginWithFinish 33](#_Toc438464829)

[(BOOL)isValidAccessTokenExpireTimeNow 34](#_Toc438464830)

[(void)loginWithFinish 34](#_Toc438464831)

[(void)logout 35](#_Toc438464832)

[(void)refreshAccessToken 35](#_Toc438464833)

[(void)refreshAccessTokenWithFinish 35](#_Toc438464834)

[(void)requestDeleteToken 36](#_Toc438464835)

[(void)setIsInAppOauthEnable 36](#_Toc438464836)

[(void)setIsNaverAppOauthEnable 36](#_Toc438464837)

[For Unity 38](#_Toc438464838)

[Development Environment 38](#_Toc438464839)

[Requirement 38](#_Toc438464840)

[Library File 38](#_Toc438464841)

[Development Environment Setup 38](#_Toc438464842)

[How to Integrate 42](#_Toc438464843)

[Execute 42](#_Toc438464844)

[Implement User ID Mapping 42](#_Toc438464845)

[Change AFNetworking 42](#_Toc438464846)

[API Reference 43](#_Toc438464847)

[GLinkUnity 43](#_Toc438464848)

[executeArticlePost() 43](#_Toc438464849)

[executeArticlePostWithImage() 43](#_Toc438464850)

[executeArticlePostWithVideo() 44](#_Toc438464851)

[executeMain() 44](#_Toc438464852)

|  |
| --- |
| Table of Tables and Figures |

Table List

[Table 1 Libraries required using NAVER Cafe SDK for Android 11](#_Toc438464853)

[Table 2 Libraries required using NAVER Cafe SDK for iOS 21](#_Toc438464854)

Figure List

[Figure 1 NAVER Cafe SDK View 9](#_Toc438464855)

# Overview

## Overview of the NAVER Cafe SDK

The NAVER Cafe SDK lets you easily integrate NAVER Cafe into your mobile games, regardless of game engine. Once it is integrated, mobile gamers do not have to leave games to use community features (NAVER Cafe).

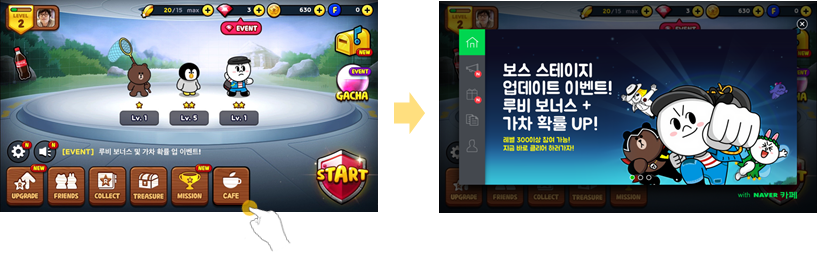


Figure 1 NAVER Cafe SDK view

## Features and Characteristics of NAVER Cafe SDK

### Main Features

The main features of the NAVER Cafe SDK include the following:

* Supporting both iOS and Android
* Supporting the following game engines: Unity 4 or later and Cocos2d-x 2.1 or later
* Offering a separate floating area for using community features
* Using NAVER Login that supports OAuth 2.0 protocol to authorize users

### Requirement

#### Operating System

The NAVER Cafe SDK supports the following platforms:

* Android: Android 4.2 (Jelly Bean) API level 17 or later
* iOS: iOS 7.0 or later (ARMv7, ARMv7s, and ARM64 supported)

#### Using NAVER Login to Authorize Users

You must have a client ID and a client secret to authorize users by using NAVER Login when you use the NAVER Cafe SDK.

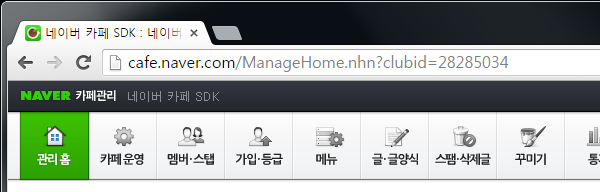
* Client ID: A value issued when you register your game to implement NAVER Login to your game app.
* Client secret: A value issued when you register your game to implement NAVER Login to your game app.

You can obtain a client ID and a client secret when you register your application in the NAVER Login Developers site. See documentation in the site below for information on how to register your application and check issued client ID and client secret or contact us at dl\_gamesdkpartner@navercorp.com.

* NAVER Login Developers: <https://nid.naver.com/devcenter/main.nhn> (Korean only)

#### NAVER Cafe ID

In order to use the NAVER Cafe SDK, you are required to have the NAVER Cafe ID. You can see your ID via URL shown in red rectangle below in the Community Settings page, a value of the clubid parameter.



# For Android

## Development Environment

### Requirement

#### Development Tool

* IDE: Android Studio or Eclipse
* Android Support Library v7

#### Adding Other Libraries

To use the NAVER Cafe SDK, you need to add the following libraries to a project before building your project.

Table 1 Libraries required using NAVER Cafe SDK for Android

|  |  |
| --- | --- |
| Library | Description |
| NAVER Login library | * Included in the NAVER Cafe SDK library (version 4.1.4). * URL for download: https://static.nid.naver.com/images/web/devcenter/3rdparty\_login\_library\_android\_4.1.4.zip |
| NAVER Volleyer | * Included in the NAVER Cafe SDK library (version 2.0.1). * URL for download: <http://mvnrepository.com/artifact/com.navercorp.volleyextensions/volleyer> |
| Volley | * Included in the NAVER Cafe SDK library (version 1.0.2). * URL for download: [http://mvnrepository.com/artifact/com.mcxiaoke.volley/library](http://mvnrepository.com/artifact/com.mcxiaoke.volley/library/) |
| Google Gson | * Included in the NAVER Cafe SDK library (version 2.3.1). * URL for download: <http://mvnrepository.com/artifact/com.google.code.gson/gson> |
| Glide | * Included in the NAVER Cafe SDK library (version 3.6.1). * URL for download: <http://mvnrepository.com/artifact/com.github.bumptech.glide/glide> |
| Otto | * Included in the NAVER Cafe SDK library (version 1.3.8). * URL for download: <http://mvnrepository.com/artifact/com.squareup/otto> |

### Library File Structure

The NAVER Cafe SDK library for Android consists of the following structure:

* lib: The NAVER Cafe SDK library and required libraries
  + cafeSdk-x.x.x.aar: The NAVER Cafe SDK library file that can be used in Android Studio
  + cafeSdk-x.x.x.zip: The NAVER Cafe SDK library that can be used in Eclipse
  + library: A library folder that can be used for the NAVER Cafe SDK
* sample: A folder for sample projects that use the NAVER Cafe SDK library
  + navercafesdk-sample-android-studio: A folder for the NAVER Cafe SDK sample projects that can be used in Android Studio
  + navercafesdk-sample-eclipse-master: A folder for the NAVER Cafe SDK sample projects that can be used in Eclipse

### Development Environment Setup

To use the NAVER Cafe SDK, you must first set up a development environment as follows:

#### Android Studio

* + - * 1. Copy the NAVER Cafe SDK library file (cafeSdk-x.x.x.aar) to the libs folder in your Android project.
        2. Copy the NAVER Login library file (naveroauthlogin-4.x.x.jar) to the libs folder in your Android project.
        3. Click File > Project Structure in the menu, click App > Dependencies in the Project Structure dialog box, and add the libraries. You can also manually add them to the build.gladle file in your Android project as follows:

compile 'com.android.support:support-v13:23.1.0'

compile 'com.navercorp.volleyextensions:volleyer:2.0.1'

compile 'com.google.code.gson:gson:2.3.1'

compile 'com.github.bumptech.glide:glide:3.6.1'

compile 'com.squareup:otto:1.3.8'

#### Eclipse

* + - * 1. Extract the NAVER Cafe SDK library file (cafeSdk-x.x.x.zip).
        2. Add library files required for building your project to the libs folder, a part of the extracted contents.
        3. Add activities used in NAVER Login and the NAVER Cafe SDK to the AndroidManifest.xml file in your Android project as follows:

<activity

android:name="com.naver.glink.android.sdk.ui.VideoPlayActivity"

android:screenOrientation="sensorLandscape"

android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen" />

<activity

android:name="com.nhn.android.naverlogin.ui.OAuthLoginActivity"

android:screenOrientation="sensorLandscape"

android:theme="@android:style/Theme.Translucent.NoTitleBar" />

<activity

android:name="com.nhn.android.naverlogin.ui.OAuthLoginInAppBrowserActivity"

android:label="OAuth2.0 In-app"

android:screenOrientation="sensorLandscape" />

## How to Integrate

### Configure the AndroidManifest.xml File

Add the code below to the AndroidManifest.xml file in your Android project.

In the package property, enter the same value that you provided for Android Intent, which was configured when you registered your application to implement NAVER Login. You also need to configure permissions to access the Glink class, an object to control the NAVER Cafe SDK.

<!—The package should have the same value that you provided for Android Intent, which was configured when you registered your application to implement NAVER Login.-->

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.naver.glink.sample">

<!—Permissions to access Glink -->

<uses-permission android:name="android.permission.INTERNET"/>

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE"/>

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

### Initialize the NAVER Cafe SDK

Use the Glink.init() method to initialize the NAVER Cafe SDK.

The values for client ID and client secret were obtained when you registered your application to implement NAVER Login.

/\*\*

\* Initialize SDK with information received from "NAVER Login Developers".

\* Initialization must be carried out before calling other methods of Glink.

\* NAVER Login Developers: https://nid.naver.com/devcenter/main.nhn (Korean only)

\*/

final int cafeId = 28266581;

final String consumerKey = "client ID";

final String consumerSecret = "client secret";

Glink.init(consumerKey, consumerSecret, cafeId);

### Start the NAVER Cafe SDK Activity

There are five tabs in the NAVER Cafe SDK area. You should select one you want to execute when the NAVER Cafe SDK activity starts.

For example, the Glink.startHome() method is used to execute the Home tab in the code below.

// Execute the Home tab when the NAVER Cafe SDK activity starts.

Glink.startHome(activity);

The other tabs are as follows: Notice, Event, Menu, and Profile

The description below shows every method to execute each tab.

* Glink.startNotice(): Executes the Notice tab.
* Glink.startEvent(): Executes the Event tab.
* Glink.startMenu(): Executes the Menu tab.
* Glink.startProfile(): Executes the Profile tab.

### Close the NAVER Cafe SDK Activity

You can use the Glink.stop() or Glink.popBackStack() method to implement a functionality that closes the NAVER Cafe SDK activity.

* Glink.stop(): Closes all activities at once and exits the NAVER Cafe SDK.
* Glink.popBackStack(): Closes activities in the order they are arranged in a back stack (backStack). The NAVER Cafe SDK exits after all activities are closed. Activities that can be arranged in backStack are as follows: a post view, writing a post, and search.

The code below shows how to close the NAVER Cafe SDK activity with the Glink.stop() method.

// Close with the stop() method.

Glink.stop(activity);

The code below shows how to close the NAVER Cafe SDK activity with the Glink.popBackStack() method.

// Close with the popBackStack() method.

Glink.popBackStack(activity);

### Compose Post

Use the Glink.startWrite() method to implement a post compose activity.

// Start a post compose activity with the default subject and body.

int menuId = 4; // If 0, a menu is not selected.

String text = "Start a post compose activity with the default subject and body.";

Glink.startWrite(MainActivity.this, "subject", text);

Use the Glink.startImageWrite() or Glink.startVideoWrite() method if you want to make the post compose activity execute with an image or video attachment.

### Execute App Scheme

If you have a banner image that links to an app scheme, you can configure OnClickAppSchemeBannerListener as described below to implement a functionality that executes an app scheme when a touch event occurs.

// Configure the app scheme touch listener.

Glink.setOnClickAppSchemeBannerListener(new Glink.OnClickAppSchemeBannerListener() {

@Override public void onClickAppSchemeBanner(String appScheme) {

// Receive the app scheme string specified in Community Settings from the NAVER Cafe SDK.

// Write code to handle an app scheme.

}

});

### Implement User ID Mapping

Use the Glink.setGameUserId() method to implement a functionality that maps user's game ID to NAVER Cafe ID.

// Map user's game ID and NAVER Cafe ID.

// The mapped game ID is displayed in the Profile.

Glink.setGameUserId(this, "gameUserId", "game ID");

### Change Resource Image

You can change the resource images contained in the NAVER Cafe SDK library as follows:

* + - * 1. Extract the NAVER Cafe SDK library file (.aaa file).
        2. Change image files located under /res/drawable-xhdpi folder.
        3. Compress the folder into a new .aaa file (NAVER Cafe SDK library).
        4. Build a project, using a newly created .aaa file.

Caution

Make sure that the size of a new image is the same as that of an old one.

## API Reference

### Glink

Class that controls the NAVER Cafe SDK. With the methods of this class, you can implement functionalities such as initializing, starting, and stopping the NAVER Cafe SDK.

The Glink class provides the following methods:

* init()
* isShowGlink()
* popBackStack()
* setGameUserId()
* startEvent()
* startImageWrite()
* startHome()
* startMenu()
* startNotice()
* startProfile()
* startVideoWrite()
* startWrite()
* stop()

### init()

#### Description

Initializes the NAVER Cafe SDK.

#### Syntax

public static void init(String clientId, String clientSecret, int cafeId);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| clientId | String | Y | Client ID issued when you registered your application to implement NAVER Login. |
| clientSecret | String | Y | Client secret issued when you registered your application to implement NAVER Login. |
| cafeId | int | Y | Cafe ID that can be found in the URL of the Community Settings page, a value of the clubid parameter. |

#### Return Value

None

#### Code Example

//  Initialize the NAVER Cafe SDK with the NAVER Login registration information and NAVER Cafe ID.

Glink.init(“abcd”, "aaaa", 33);

### isShowGlink()

#### Description

Checks whether the NAVER Cafe SDK activity is opened.

#### Syntax

public static boolean isShowGlink(Activity activity);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

* true: State in which the NAVER Cafe SDK activity is opened
* false: State in which the NAVER Cafe SDK activity is not opened

Code Example

// Check whether the NAVER Cafe SDK activity is opened.

// If true, it is opened; if false, it is not opened.

Glink.isShowGlink(this);

### popBackStack()

#### Description

Closes activities in the order they are arranged in a back stack (backStack). The NAVER Cafe SDK exits after all activities are closed. Activities that can be arranged in backStack are as follows: a post view, writing a post, and search.

#### Syntax

public static void popBackStack(Activity activity);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

None

Code Example

// Close the NAVER Cafe SDK activities one by one.

Glink.popBackStack (activity);

### setGameUserId()

#### Description

Maps user's game ID to NAVER Cafe ID.

#### Syntax

public static void setGameUserId(Activity activity, String gameUserId, String fieldName);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |
| gameUserId | String | N | User's game ID |
| fieldName | String | N | Display ID in the Profile (default: user's game ID) |

#### Return Value

None

#### Code Example

// Map user's game ID and NAVER Cafe ID.

// The mapped game ID is displayed in the Profile.

Glink.setGameUserId(this, "gameUserId", "game ID");

### startEvent()

#### Description

Starts the NAVER Cafe SDK when the Event tab is selected.

#### Syntax

public static void startEvent(Activity activity);

#### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

None

Code Example

// Execute the Event tab when the NAVER Cafe SDK activity starts.

Glink.startEvent(activity);

### startImageWrite()

#### Description

Opens a post compose activity with an image attachment.

#### Syntax

public static void startImageWrite(Activity activity, int menuId, String subject, String text, String imagery);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |
| menuId | int | N | Menu ID (default: -1); it can be found in the URL of the NAVER Cafe Menu, a value of the menuid parameter. |
| subject | String | N | Post subject |
| text | String | N | Body of the post |
| imageUri | String | N | Image file path (with URI scheme) |

Return Value

None

Code Example

// Start a post compose activity with the default subject, body, and an image. Use the URI scheme for an image.

int menuId = 4; // If 0, a menu is not selected.

String text = "Start a post compose activity with the default subject, body, and an image.\nYou should use the URI scheme for an image.";

String path = "your image uri";

Glink.startImageWrite(MainActivity.this, menuId, "subject", text, path);

### startHome()

#### Description

Opens the NAVER Cafe SDK activity when the Home tab is selected.

#### Syntax

public static void startHome(Activity activity);

#### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

#### Return Value

None

#### Code Example

// Execute the Home tab when the NAVER Cafe SDK activity starts.

Glink.startHome(activity);

### startMenu()

#### Description

Opens the NAVER Cafe SDK activity when the Menu tab is selected.

#### Syntax

public static void startMenu(Activity activity);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

None

Code Example

// Execute the Menu tab when the NAVER Cafe SDK activity starts.

Glink.startMenu(activity);

### startNotice()

#### Description

Open the NAVER Cafe SDK activity when the Notice tab is selected.

#### Syntax

public static void startNotice(Activity activity);

#### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

None

Code Example

// Execute the Notice tab when the NAVER Cafe SDK activity starts.

Glink.startNotice(activity);

### startProfile()

#### Description

Opens the NAVER Cafe SDK activity when the Profile tab is selected.

#### Syntax

public static void startProfile(Activity activity);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

None

Code Example

// Execute the Profile tab when the NAVER Cafe SDK activity starts.

Glink.startProfile(activity);

### startVideoWrite()

#### Description

Opens a post compose activity with a video attachment.

#### Syntax

public static void startVideoWrite(Activity activity, int menuId, String subject, String text, String videoUri);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |
| menuId | int | N | Menu ID (default: -1); it can be found in the URL of the NAVER Cafe Menu, a value of the menuid parameter. |
| subject | String | N | Post subject |
| text | String | N | Body of the post |
| videoUri | String | N | Video file path (with URI scheme) |

Return Value

None

Code Example

// Start a post compose activity with the default subject, body, and a video. Use the URI scheme for a video.

int menuId = 4; // If 0, a menu is not selected.

String text = "Start a post compose activity with the default subject, body, and a video.\nYou should use the URI scheme for a video.";

String path = "your video uri";

Glink.startVideoWrite(MainActivity.this, menuId, "subject", text, path);

### startWrite()

#### Description

Opens a post compose activity.

#### Syntax

public static void startWrite(Activity activity, int menuId, String subject, String text);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |
| menuId | int | N | Menu ID (default: -1); it can be found in the URL of the NAVER Cafe Menu, a value of menuid parameter. |
| subject | String | N | Post subject |
| text | String | N | Body of the post |

Return Value

None

Code Example

// Start a post compose activity with the default subject and body.

int menuId = 4; // If 0, a menu is not selected.

String text = "Start a post compose activity with the default subject and body.";

Glink.startWrite(MainActivity.this, "subject", text);

### stop()

#### Description

Closes all activities at once and exits the NAVER Cafe SDK.

#### Syntax

public static void stop(final Activity activity);

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| activity | Activity | Y | Context object of an activity that executes the method |

Return Value

None

Code Example

// Close with the stop() method.

Glink.stop(activity);

# For iOS

## Development Environment

### Requirement

#### Development Tool

* IDE: Xcode 6.0 or later

Caution

Automatic reference counting (ARC) is enabled in this library.

#### Adding Other Libraries

To use the NAVER Cafe SDK, you need to add the following libraries to a project before building your project.

Table 2 Libraries required using NAVER Cafe SDK for iOS

|  |  |
| --- | --- |
| Library | Description |
| NAVER Login library | * Included in the NAVER Cafe SDK library (version 4.0.6). * URL for download: https://static.nid.naver.com/images/web/devcenter/3rdparty\_login\_library\_ios\_4.0.6.zip |
| AFNetworking 1.0 or later | * Included in the NAVER Cafe SDK library (version 2.6.1). * URL for download: <https://github.com/AFNetworking/AFNetworking> |

### Library File Structure

The NAVER Cafe SDK library for iOS consists of the following structure:

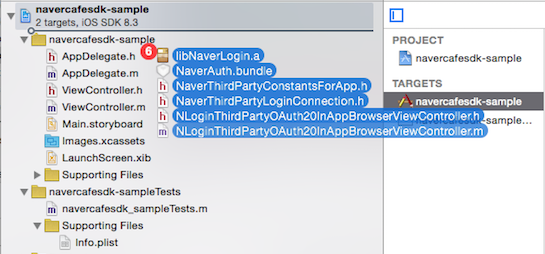
* lib: A folder for the NAVER Cafe SDK library
  + NaverCafeSDK.bundle
  + NaverCafeSDK.framework
* sample: A folder for sample projects that use the NAVER Cafe SDK library and for required libraries
  + external-lib: A folder for the NAVER Login library and the AFNetworking library
  + navercafesdk-sample-ios: A folder for the NAVER Cafe SDK sample projects

### Development Environment Setup

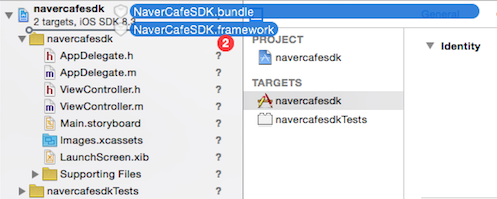
#### Configuring Xcode

To use the NAVER Cafe SDK, you must first set up a development environment in Xcode as follows:

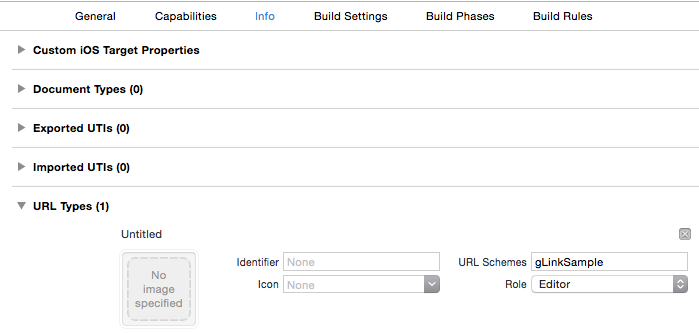
* + - * 1. Extract the NAVER Login library file.
        2. Add the NAVER Login library to your Xcode project.



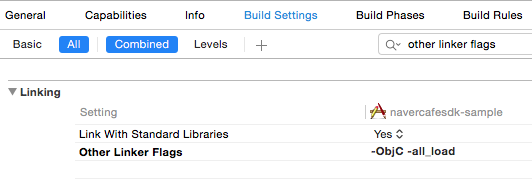
* + - * 1. Add the AFNetworking library to your Xcode project.
        2. Extract the NAVER Cafe SDK library file.
        3. Add both NaverCafeSDK.framework and NaverCafeSDK.bundle to your Xcode project.



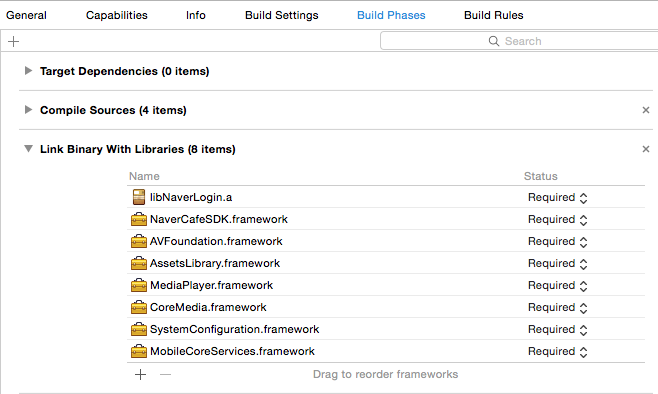
* + - * 1. Register a URL scheme that you entered when you registered your application to implement NAVER Login in your project.



* + - * 1. Go to Build Settings > Other Linker Flags and specify the -ObjC –all\_load option so that you can use a static library.



* + - * 1. Add the following libraries to Link Binary With Libraries under Build Phases.
  + MobileCoreServices.framework
  + SystemConfiguration.framework
  + MediaPlayer.framework
  + AVFoundation.framework
  + CoreMedia.framework



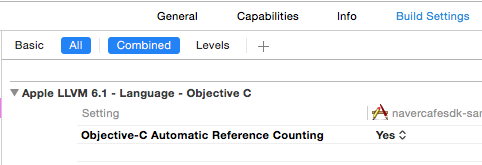
To use a Cocos2d-x engine, add the following libraries.

* + MobileCoreServices.framework
  + SystemConfiguration.framework
  + MediaPlayer.framework
  + AVFoundation.framework
  + CoreMedia.framework
  + GameController.framework
  + AssetsLibrary.framework
  + Security.framework

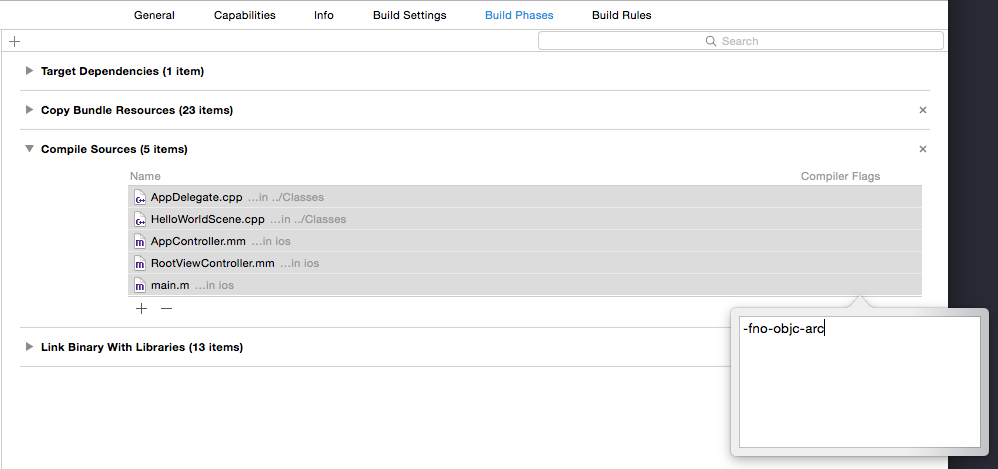
#### Converting a Project from MRC to ARC

If you are using a Cocos2d-x engine in a manual reference counting (MRC) environment, you must convert it to an automatic reference counting (ARC) environment as follows:

* + - * 1. Go to Build Settings > Objective-C Automatic Reference Counting and specify YES.



* + - * 1. Go to Build Phases > Compile Sources and set Compile Flags of target files to -fno-objc-arc.



## How to Integrate

### Initialize the NAVER Cafe SDK

Initialize the NAVER Cafe SDK as described below. The values for client ID and client secret were obtained when you registered your application to implement NAVER Login.

// ViewController

#import <NaverCafeSDK/NCSDKManager.h>

- (void)viewDidLoad {

// Initialize the NAVER Cafe SDK with the NAVER Login registration information and NAVER Cafe ID.

[[NCSDKManager getSharedInstance] setNaverLoginConsumerKey:@"client ID"

naverLoginConsumerSecret:@"client secret"

cafeId:00000000];

}

### Configure NAVER Login

Write code below to configure the app delegate so that the login information can be configured in the NAVER Cafe SDK when the NAVER Login process is completed.

// AppDelegate

#import <NaverCafeSDK/NCSDKLoginManager.h>

- (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(NSString \*)sourceApplication annotation:(id)annotation {

// Configure the login information to the NAVER Login object.

return [[NCSDKLoginManager getSharedInstance] finishNaverLoginWithURL:url];

}

### Start the NAVER Cafe SDK Activity

To start the NAVER Cafe SDK activity, you must first set up a view controller for the NAVER Cafe SDK and execute the presentMainViewController method.

#import <NaverCafeSDK/NCSDKManager.h>

// Execute the NAVER Cafe Home.

[[NCSDKManager getSharedInstance] setParentViewController:self];

[[NCSDKManager getSharedInstance] presentMainViewController];

If you use the presentMainViewControllerWithTabIndex method, instead of using the presentMainViewController method, you can start the NAVER Cafe SDK activity when a specific tab is selected. The presentMainViewControllerWithArticleId method lets the NAVER Cafe SDK activity start when a specific post is opened.

### Compose Post

Use the presentArticlePostViewControllerWithMenuId method to implement a post compose activity.

// Execute a post compose activity.

[[NCSDKManager getSharedInstance] setParentViewController:self];

[[NCSDKManager getSharedInstance] presentArticlePostViewControllerWithMenuId:0 subject:@"post subject" content:@"body of the post" filePath:@"document"];

## API Reference

### NCSDKManager

Class that controls the NAVER Cafe SDK. With the methods of this class, you can implement functionalities such as initializing, starting, and stopping the NAVER Cafe SDK. It contains the parentViewController class, the top view controller that manages the NAVER Cafe SDK.

The NCSDKManager class provides the following methods:

* dismissViewController
* dismissTopViewController
* getSharedInstance
* navercafeRootViewController
* resetSharedInstance
* presentArticlePostViewControllerWithMenuId
* presentArticlePostViewControllerWithType
* presentMainViewController
* presentMainViewControllerWithArticleId
* presentMainViewControllerWithTabIndex
* presentViewController
* setGameUserId
* setNaverLoginClientId

### @property (nonatomic, weak) id parentViewController

#### Description

View controller that manages the NAVER Cafe SDK. It is a parent class in which the NAVER Cafe SDK is executing.

Syntax

@property (nonatomic, weak) id parentViewController;

Parameters

None

Return Value

UIViewController object

Code Example

[[NCSDKManager getSharedInstance] setParentViewController:self]

### (void)dismissViewController

#### Description

Dismisses a view controller, presented above the view controller that manages the NAVER Cafe SDK.

#### Syntax

- (void)dismissViewController:(id)viewController;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| viewController | id | Y | View controller presented above the view controller that manages the NAVER Cafe SDK |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] dismissViewController:self];

### (void)dismissTopViewController

#### Description

Dismisses the top view controller among current ones, presented above the view controller that manages the NAVER Cafe SDK.

#### Syntax

- (void)dismissTopViewController;

Parameters

None

Return Value

None

Code Example

[NCSDKManager getSharedInstance] dismissTopViewController];

### (NCSDKManager \*)getSharedInstance

#### Description

Gets a NAVER Cafe SDK instance (singleton instance).

#### Syntax

+ (NCSDKManager \*)getSharedInstance;

Parameters

None

Return Value

NCSDKManager object

Code Example

[NCSDKManager getSharedInstance]

### (id)navercafeRootViewController

#### Description

Gets the top view controller object of the NAVER Cafe SDK.

#### Syntax

- (id)navercafeRootViewController;

Parameters

None

Return Value

UIViewController object

Code Example

[[NCSDKManager getSharedInstance] navercafeRootViewController]

### (void)resetSharedInstance

#### Description

Removes an NCSDKManager object.

#### Syntax

+ (void)resetSharedInstance;

Parameters

None

Return Value

None

Code Example

[NCSDKManager resetSharedInstance]

### (void)presentArticlePostViewControllerWithMenuId

#### Description

Opens a post compose activity.

#### Syntax

- (void)presentArticlePostViewControllerWithMenuId:(NSInteger)menuId

subject:(NSString \*)subject

content:(NSString \*)content;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| menuId | NSInteger | Y | Menu ID (default: 0); it can be found in the URL of the NAVER Cafe Menu, a value of the menuid parameter. |
| subject | NSString | N | Post subject |
| content | NSString | N | Body of the post |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] presentArticlePostViewControllerWithMenuId:1

subject:@"subject"

content:@"body"];

### (void)presentArticlePostViewControllerWithType

#### Description

Opens a post compose activity with an attachment.

#### Syntax

- (void)presentArticlePostViewControllerWithType:(GLArticlePostType)type

menuId:(NSInteger)menuId

subject:(NSString \*)subject

content:(NSString \*)content

filePath:(NSString \*)filePath;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| type | GLArticlePostType | Y | Attachment file types   * 1: images * 2: videos |
| menuId | NSInteger | Y | Menu ID (default: 0); it can be found in the URL of the NAVER Cafe Menu, a value of the menuid parameter. |
| subject | NSString | N | Post subject |
| content | NSString | N | Body of the post |
| filePath | NSString | Y | Attached file path |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] presentArticlePostViewControllerWithType:kGLArticlePostTypeVideo

menuId:1

subject:@"subject"

content:@"body"

filePath:@"private/var/mobile/Applications/0D1657F9-EACF-4D64-BC8A-4E01EB4FF247/tmp/trim.2CC623C7-78C3-4597-BA75-9BA12BFEF333.MOV"];

### (void)presentMainViewController

#### Description

Opens the NAVER Cafe SDK activity.

#### Syntax

- (void)presentMainViewController;

Parameters

None

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] presentMainViewController];

### (void)presentMainViewControllerWithArticleId

#### Description

Opens the NAVER Cafe SDK activity when a specific post is opened.

#### Syntax

- (void)presentMainViewControllerWithArticleId:(NSUInteger)articleId;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| articleId | NSUInteger | Y | Post ID (default: 0) |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] presentMainViewControllerWithArticleId:10];

### (void)presentMainViewControllerWithTabIndex

#### Description

Opens the NAVER Cafe SDK activity when a specific tab is selected.

#### Syntax

- (void)presentMainViewControllerWithTabIndex:(NSUInteger)tabIndex;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| tabIndex | NSUInteger | Y | Index values of selected tabs (default: 0)   * 0: Home * 1: Notice * 2: Event * 3: Menu * 4: Profile |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] presentMainViewControllerWithTabIndex:1];

### (void)presentViewController

#### Description

Presents another view controller on top of currently displayed view controller of the NAVER Cafe SDK.

#### Syntax

- (void)presentViewController:(id)viewController;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| viewController | id | Y | ID of a view controller object |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] presentViewController:self];

### (void)setGameUserId

#### Description

Connects user's game ID with NAVER Cafe ID and displays the ID in the Profile.

#### Syntax

- (void)setGameUserId:(NSString \*)gameUserId fieldName:(NSString \*)fieldName;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| gameUserId | NSString | Y | User's game ID |
| fieldName | NSString | N | Display ID in the Profile (default: user's game ID) |

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] setGameUserId:@"abc3251235" fieldName:@"game ID"];

### (void)setNaverLoginClientId

#### Description

Sets a NAVER Login object.

#### Syntax

- (void)setNaverLoginClientId:(NSString \*)naverLoginClientId

naverLoginClientSecret:(NSString \*)naverLoginClientSecret

cafeId:(NSInteger)cafeId;

Parameters

None

Return Value

None

Code Example

[[NCSDKManager getSharedInstance] setNaverLoginConsumerKey:@"Consumer ID"

naverLoginConsumerSecret:@"Secret ID"

cafeId:00000000];

### NCSDKLoginManager

Class that controls the NAVER Login functionalities within the NAVER Cafe SDK.

The NCSDKLoginManager class provides the following methods:

* accessToken
* accessTokenExpireTime
* finishNaverLoginWithURL
* getSharedInstance
* isLoginWithFinish
* isValidAccessTokenExpireTimeNow
* loginWithFinish
* logout
* refreshAccessToken
* refreshAccessTokenWithFinish
* requestDeleteToken
* setIsInAppOauthEnable
* setIsNaverAppOauthEnable

### @property (nonatomic, weak) UIViewController \*rootViewController

#### Description

View controller that manages logins in a web view.

#### Syntax

@property (nonatomic, weak) UIViewController \*rootViewController;

Parameters

None

Return Value

UIViewController object

Code Example

[[NCSDKLoginManager getSharedInstance] setRootViewController:self];

### (NSString \*)accessToken

#### Description

Gets an access token that was received from NAVER servers after authorizing an app with NAVER Login.

#### Syntax

- (NSString \*)accessToken;

Parameters

None

Return Value

Access token

Code Example

[[NCSDKLoginManager getSharedInstance] accessToken];

### (NSString \*)accessTokenExpireTime

#### Description

Gets expiration time of an access token.

#### Syntax

- (NSString \*)accessTokenExpireTime;

Parameters

None

Return Value

Expiration time of an access token

Code Example

[[NCSDKLoginManager getSharedInstance] accessTokenExpireTime];

### (BOOL)finishNaverLoginWithURL

#### Description

Starts the app delegate after completing the NAVER Login process.

#### Syntax

- (BOOL)finishNaverLoginWithURL:(NSURL \*)url;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| url | NSURL | Y | The URL scheme called back to the app delegate after completing the NAVER Login process |

Return Value

* true: Login success
* false: Login failure

Code Example

- (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(NSString \*)sourceApplication annotation:(id)annotation {

return [[NCSDKLoginManager getSharedInstance] finishNaverLoginWithURL:url];

}

### (NCSDKLoginManager \*)getSharedInstance

#### Description

Gets a NAVER Login instance (singleton instance).

#### Syntax

+ (NCSDKLoginManager \*)getSharedInstance;

Parameters

None

Return Value

NCSDKLoginManager object

Code Example

[NCSDKLoginManager getSharedInstance]

### (void)isLoginWithFinish

#### Description

Checks whether an access token remains valid, which was received from NAVER servers after authorizing an app with NAVER Login.

#### Syntax

- (void)isLoginWithFinish:(void (^)(BOOL successACToken))finish;

Parameters

None

Return Value

* true: Valid access token
* false: Invalid access token because it has expired.

Code Example

[[NCSDKLoginManager getSharedInstance] isLoginWithFinish:^(BOOL successACToken) {

if (successACToken) {

}

}];

### (BOOL)isValidAccessTokenExpireTimeNow

#### Description

Checks whether an access token exists and remains valid. Note that you cannot check the validity of an access token if it has expired on the server.

#### Syntax

- (BOOL)isValidAccessTokenExpireTimeNow;

Parameters

None

Return Value

* true: Valid access token
* false: Invalid access token because it has expired.

Code Example

[[NCSDKLoginManager getSharedInstance] isValidAccessTokenExpireTimeNow];

### (void)loginWithFinish

#### Description

Starts the process of app authorization with NAVER Login and gets an access token.

#### Syntax

- (void)loginWithFinish:(void (^)(BOOL successACToken))finish;

Parameters

None

Return Value

* true: Valid access token
* false: Invalid access token because it has expired.

Code Example

[[NCSDKLoginManager getSharedInstance] isLoginWithFinish:^(BOOL successACToken) {

if (successACToken) {

}

}];

### (void)logout

#### Description

Deletes stored access token and refresh token and logs out.

#### Syntax

- (void)logout;//delete local access token

Parameters

None

Return Value

None

Code Example

[[NCSDKLoginManager getSharedInstance] logout];

### (void)refreshAccessToken

#### Description

Renews an access token with the refresh token that was received from NAVER servers after authorizing an app with NAVER Login.

#### Syntax

- (void)refreshAccessToken;

Parameters

None

Return Value

None

Code Example

[[NCSDKLoginManager getSharedInstance]refreshAccessToken];

### (void)refreshAccessTokenWithFinish

#### Description

Gets a refresh token that was received from NAVER servers after authorization an app with NAVER Login.

#### Syntax

- (void)refreshAccessTokenWithFinish:(void (^)(BOOL successACToken))finish;

Parameters

None

Return Value

* true: Success
* false: Failure

Code Example

[[NCSDKLoginManager getSharedInstance] refreshAccessTokenWithFinish:^(BOOL successACToken) {

if (successACToken) {

}

}];

### (void)requestDeleteToken

#### Description

Deletes stored access token and refresh token from both client and server.

#### Syntax

- (void)requestDeleteToken;//delete server authorization

Parameters

None

Return Value

None

Code Example

[[NCSDKLoginManager getSharedInstance] requestDeleteToken];

### (void)setIsInAppOauthEnable

#### Description

Runs an in-app browser and performs the login process.

#### Syntax

- (void)setIsInAppOauthEnable:(BOOL)enable;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| enable | BOOL | Y | Whether to run an in-app browser |

Return Value

None

Code Example

[[NCSDKLoginManager getSharedInstance] setIsInAppOauthEnable:YES];

### (void)setIsNaverAppOauthEnable

#### Description

Runs a NAVER app and performs the login process.

#### Syntax

- (void)setIsNaverAppOauthEnable:(BOOL)enable;

Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| enable | BOOL | Y | Whether to run a NAVER app |

Return Value

None

Code Example

[[NCSDKLoginManager getSharedInstance] setIsNaverAppOauthEnable:YES];

# For Unity

## Development Environment

### Requirement

#### Development Tool

* Game engine: Unity 4 or later

### Library File

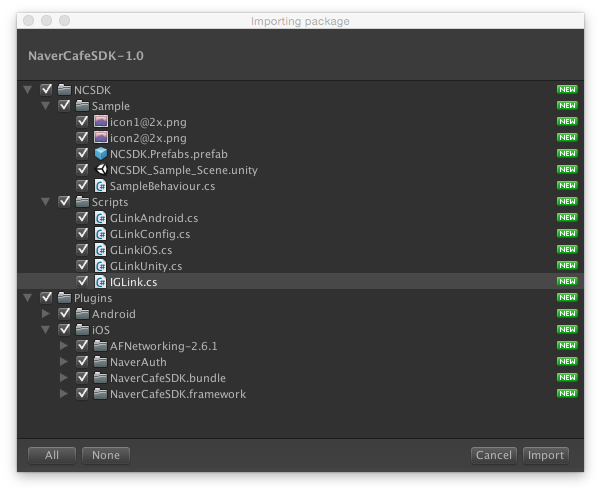
* NaverCafeSDK-1.0.unitypackage: NAVER Cafe SDK library for Unity

### Development Environment Setup

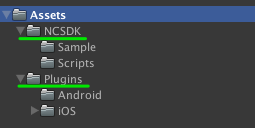
To use the NAVER Cafe SDK, you must first set up a development environment as follows:

#### Configuring Unity

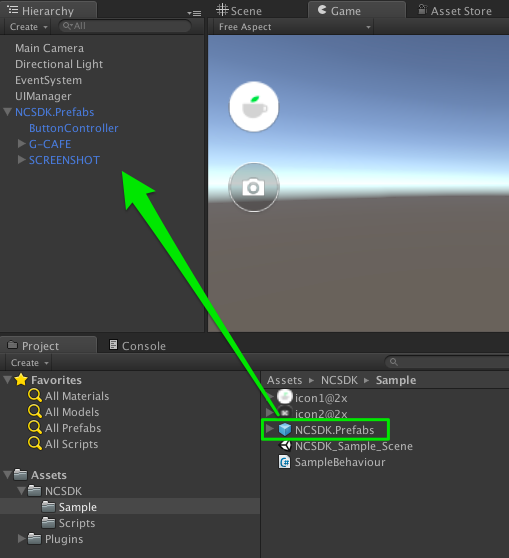
* + - * 1. Click Asset > Import package in the Unity menu and select the NaverCafeSDK-1.0.unityPackage file, which was downloaded, in the Custom Package dialog box.



* + - * 1. Click Import to create the NCSDK and Plugins folders in the Custom Package dialog box.



* + - * 1. In the NCSDK > Sample folder, drag and drop NCSDK.Prefabs to a desired scene. A default button provided by the NAVER Cafe SDK is created.



* + - * 1. In the NCSDK > Scripts folder, enter Cafe ID, client ID, and client secret in the GLinkConfig file.

//NCSDK/Scripts/GLinkConfig

static class GLinkConfig

{

public const string NaverLoginConsumerKey =

"Consumer ID";

public const string NaverLoginConsumerSecret =

"Secret ID";

public const int CafeId =

00000000;

}

#### Configuring Android

To use a Unity 5 engine, you must add the NAVER Cafe SDK library as follows:

* Add the NAVER Cafe SDK library file (cafeSdk-x.x.x.aar) and required library files to the Assets/Plugins/Android folder.

To use a Unity 4 engine, you must add the following additional configurations in Eclipse.

* + - * 1. Extract cafeSdk-x.x.x.zip which belongs to the NAVER Cafe SDK library.
        2. Add library files required for building your project to the libs folder, a part of the extracted contents.
        3. Import the extracted folder to your project in Eclipse.
        4. Click Android and select Is Library in the project configuration dialog box.
        5. Add activities used in the NAVER Cafe SDK to the AndroidManifest.xml file as follows:

<activity

android:name="com.nhn.android.naverlogin.ui.OAuthLoginActivity"

android:screenOrientation="sensorLandscape"

android:theme="@android:style/Theme.Translucent.NoTitleBar"/>

<activity

android:name="com.nhn.android.naverlogin.ui.OAuthLoginInAppBrowserActivity"

android:screenOrientation="sensorLandscape"

android:label="OAuth2.0 In-app"/>

<activity

android:name=".ui.VideoPlayActivity"

android:screenOrientation="sensorLandscape"

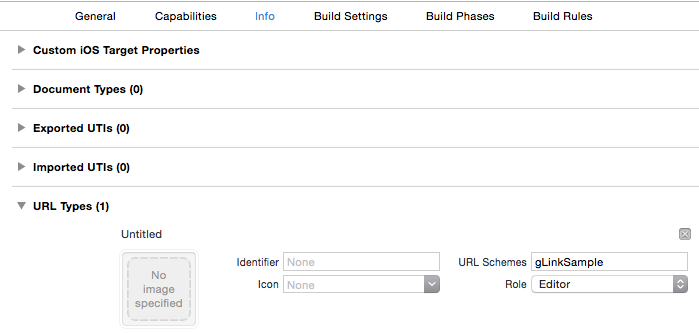
android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen"/>

* + - * 1. Build your project in Eclipse.
        2. Add the output to the Assets/Plugins/Android in your Unity project.
        3. Build your Unity project.

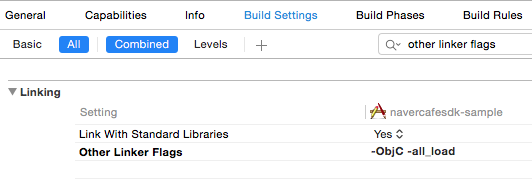
#### Configuring iOS

To develop applications for iOS, you must add the following configurations to your Xcdoe project.

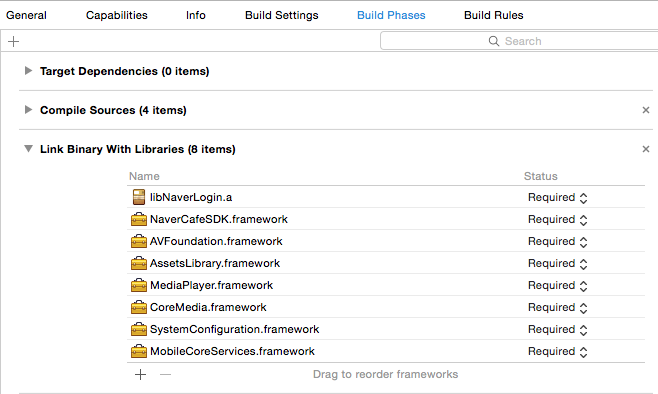
* + - * 1. Register a URL scheme that you entered when you registered your application to implement NAVER Login in your project.



* + - * 1. Go to Build Settings > Other Linker Flags and specify the -ObjC –all\_load option so that you can use a static library.



* + - * 1. Add the following libraries to Link Binary With Libraries under Build Phases.
  + MobileCoreServices.framework
  + SystemConfiguration.framework
  + MediaPlayer.framework
  + AVFoundation.framework
  + CoreMedia.framework
  + Security.framework
  + AssetsLibrary.framework



* + - * 1. Write code to call the app delegate when the NAVER Login process is completed as follows:

//AppDelegate

# import <NaverCafeSDK/NCSDKLoginManager.h>

- (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(NSString \*)sourceApplication annotation:(id)annotation {

return [[NCSDKLoginManager getSharedInstance] finishNaverLoginWithURL:url];

}

## How to Integrate

### Execute

Use the GlinkUnity.executeMain() method to start the NAVER Cafe SDK activity.

// Cafe Home

GlinkUnity.executeMain ();

Use the GlinkUnity.executeArticlePostWithImgae() method to implement a post compose activity.

//screenshot

GlinkUnity.executeArticlePostWithImgae(menuId, "post subject", "body of the post", image path);

Use the GlinkUnity .executeArticlePostWithImage() or GlinkUnity .executeArticlePostWithVideo() method if you want to make the post compose activity execute with an image or video attachment.

### Implement User ID Mapping

The NAVER Cafe SDK offers a functionality that maps user's game ID to NAVER Cafe ID. You can see the list of ID pairs in the Community Settings page if you access the page from a computer.

The example below shows how to implement an ID mapping functionality for iOS applications.

/\*

Map user's game ID to NAVER Cafe ID.

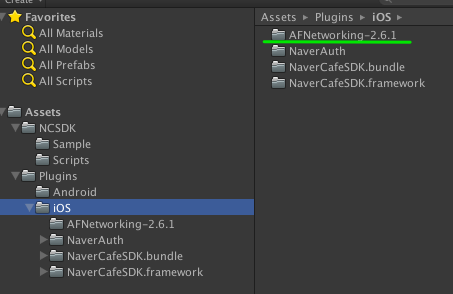
fieldName is a string to be displayed in the Profile (default: game ID)

\*/

- (void)setGameUserId:(NSString \*)gameUserId fieldName:(NSString \*)fieldName;

### Change AFNetworking

AFNetworking 2.6.1, a networking library, is included in NaverCafeSDK.unitypackage by default. You can delete this AFNetworking library or change it to other version for building your project if needed. Note, however, that the NAVER Cafe SDK supports AFNetworking version 1.x or later.



## API Reference

### GLinkUnity

Class that controls the NAVNER Cafe SDK. With the methods of this class, you can implement functionalities such as executing the NAVER Cafe SDK or post compose activity.

The GLinkUnity class provides the following methods:

* executeArticlePost()
* executeArticlePostWithImage()
* executeArticlePostWithVideo()
* executeMain()

### executeArticlePost()

#### Description

Opens a post compose activity.

#### Syntax

public static void executeArticlePost(int menuId, string subject, string content) {

sharedInstance().executeArticlePost (menuId, subject, content);

}

#### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| menuId | int | Y | Menu ID (default: -1); it can be found in the URL of the NAVER Cafe Menu; a value of the menuid parameter. |
| subject | string | Y | Post subject |
| content | string | Y | Body of the post |

#### Return Value

None

#### Code Example

GlinkUnity.executeArticlePost(28290504, "post subject","body of the post" );

### executeArticlePostWithImage()

#### Description

Opens a post compose activity with an image attachment.

#### Syntax

public static void executeArticlePostWithImage(int menuId, string subject, string content, string filePath) {

sharedInstance().executeArticlePostWithImage (menuId, subject, content, filePath);

}

#### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| menuId | int | Y | Menu ID (default: -1); it can be found in the URL of the NAVER Cafe Menu, a value of the menuid parameter. |
| subject | string | Y | Post subject |
| content | string | Y | Body of the post |
| filePath | string | Y | Image file path (with URI scheme) |

#### Return Value

None

#### Code Example

{

GlinkUnity.executeArticlePostWithImgae(28290504, "post subject", "post content","/navercafesdk/glink.png" );

}

### executeArticlePostWithVideo()

#### Description

Opens a post compose activity with a video attachment.

#### Syntax

public static void executeArticlePostWithVideo(int menuId, string subject, string content, string filePath) {

sharedInstance().executeArticlePostWithVideo (menuId, subject, content, filePath);

}

#### Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Required | Description |
| menuId | int | Y | Menu ID (default: -1); it can be found in the URL of the NAVER Cafe Menu, a value of the menuid parameter. |
| subject | string | Y | Post subject |
| content | string | Y | Body of the post |
| filePath | string | Y | Video file path (with URI scheme) |

#### Return Value

None

#### Code Example

{

GlinkUnity.executeArticlePostWithImgae(28290504, "post subject", "body of the post","/navercafesdk/glink.avi" );

}

### executeMain()

#### Description

Opens the NAVER Cafe SDK activity.

#### Syntax

public static void executeMain() {

sharedInstance().executeMain ();

}

#### Parameters

None

#### Return Value

None

#### Code Example

{

GlinkUnity.executeMain ();

}