

App Launcher

AppLauncher plugin allows you to launch another app from your unity application/game.

App Launcher can launch Settings, Facebook, LinkedIn, WhatsApp, Skype, Twitter, SMS. This plugin can also launch any other application in android by its package name and in iOS by its url scheme.

Launch any specific setting in the following list

iOS and Android :

- Default Setting

Android only :

- Sound Setting
- Accessibility Settings
- Airplane Settings
- Date Time Settings
- Bluetooth Settings
- WIFI Settings
- Keyboard Settings

Integration Guide

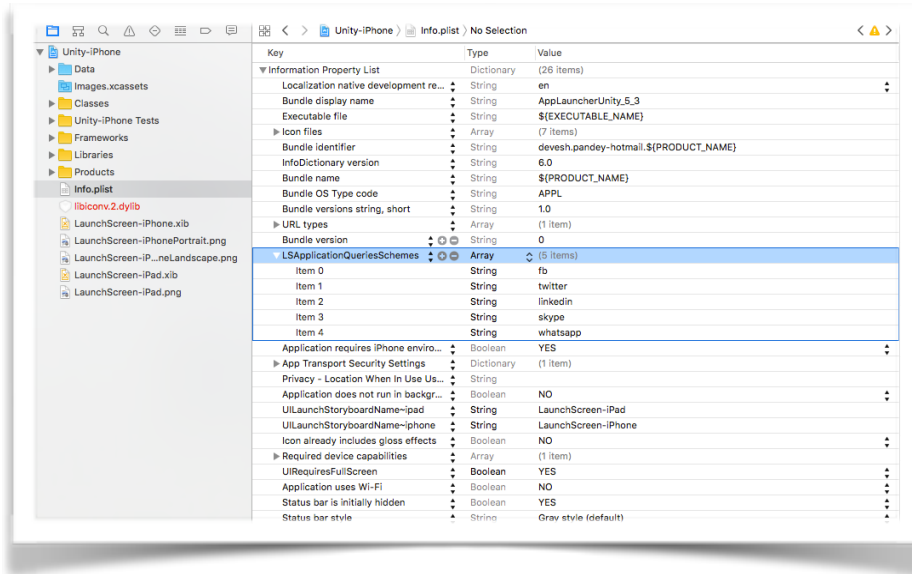
Just import App Launcher in your project and call function of the plugin as per your need. See the function details below.

You need to add some keys to info.plist if you are building for iOS 9 or above.

Add keys in info.plist for Facebook, LinkedIn, WhatsApp, Twitter like below

- Export xcode package
- Select info.plist
- Add LSApplicationQueriesSchemes key
- Add these items - fb, skype, linkedin, whatsapp, twitter

<http://unitydevelopers.blogspot.in/2017/06/launch-another-ios-app-from-your-app.html>



Function Details

1. Launch Facebook App

`public static bool LaunchFacebookApp(string id, FacebookProfileType pageType, string callback)`

id : Facebook id to launch

pageType : Page type

callback : Name of game object of callback script.

Description : This function will launch Facebook app and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchFacebookApp("1300757898", FacebookProfileType.PROFILE, "AppLauncherEvent");`

2. Launch LinkedIn App

`public static bool LaunchLinkedInApp(string id, LinkedInProfileType pageType, string callback)`

id : LinkedIn id of profile to launch

pageType : Page type

callback : Name of game object of callback script.

Description : This function will launch LinkedIn app and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchLinkedInApp("devesh-pandey-b2089738", LinkedInProfileType.INDIVIDUAL, "AppLauncherEvent");`

3. Launch Email App

`public static bool LaunchEmailApp(string emailId, string subject, string body, string callback)`

emailId : email id to send email.

subject : subject of the email.

body : text message to send.

callback : Name of game object of callback script.

Description : This function will launch email app with subject & message and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchEmailApp("example@gmail.com", "Demo Email", "This is message body.", "AppLauncherEvent");`

4. Launch WhatsApp

`public static bool LaunchWhatsApp(string msg, string callback)`

msg : Message to send.

callback : Name of game object of callback script.

Description : This function will launch WhatsApp app with message to send and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchWhatsApp("Hello","AppLauncherEvent");`

5. Launch Twitter

`public static bool LaunchTwitter(string screenName, string callback)`

screenName : Twitter user name.

callback : Name of game object of callback script.

Description : This function will launch twitter and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchTwitter("devesh_pandey19" ,"AppLauncherEvent");`

6. Launch SMS

`public static bool LaunchSMS(string phoneNo, string msg, string callback)`

phoneNo : Receiver phone number.

msg : Message to send.

callback : Name of game object of callback script.

Description : This function will launch SMS and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchSMS("9898989898", "Hello World!", "AppLauncherEvent");`

7. Launch Settings

`public static bool LaunchSettings(SettingType type, string callback)`

type : setting type to launch.

(it will work in Android only and will be ignored in iOS as per apple guidelines its not allowed.)

callback : Name of game object of callback script.

Description : This function will launch settings and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchSettings(SettingType.DATE_SETTING,"AppLauncherEvent");`

iOS : Main setting will launch

Android : Date setting will launch

8. Launch any other app

`public static bool LaunchApp(string urlScheme, string callback)` for iOS

`public static bool LaunchApp(string packageName, string callback)` for Android

urlScheme (iOS): Launch app in iOS by url scheme.

packageName : Launch by package name.

callback : Name of game object of callback script.

Description : This function will launch any other app and return true if successfully launched, otherwise false.

Example :

`AppLauncher.LaunchApp("skype://", "AppLauncherEvent");` (iOS)

`AppLauncher.LaunchApp("com.skype.raider", "AppLauncherEvent");` (Android)

Notes : If you want to launch any other app in iOS then you have to make a entry of url scheme keyword in info.plist as described in Integration Guide section.

Suppose you want to launch twitter app then you have to add "twitter" in the items of **LSApplicationQueriesSchemes**.

9. Launch any other app (for Android only)

```
public static bool LaunchAppWithParameters(string packageName,  
Dictionary<string, string> parameters, string callback)
```

packageName : Launch by package name.

parameters : parameters name & value dictionary.

callback : Name of game object of callback script.

Description: This function will launch any other app with parameter and return true if successfully launched, otherwise false.

Example :

```
Dictionary<string, string> nameAndValue = new Dictionary<string, string>();  
nameAndValue.Add("appId", "id");  
AppLauncher.LaunchApp("com.skype.raider", nameAndValue,  
"AppLauncherEvent");
```

Note : in iOS you can launch any other application with parameter by using query string. Here is example of launching whatsapp

```
NSURL *whatsappURL = [NSURL URLWithString:@"whatsapp://send?  
text=Hello%20World!"];  
if ([[UIApplication sharedApplication] canOpenURL: whatsappURL]) {  
    [[UIApplication sharedApplication] openURL: whatsappURL];  
}
```

Events Handling

Create a empty game object in hierarchy and name it as you want, say “AppLauncherEvent” and pass this name as parameter in each function for callback.

Create a script and add these 2 function in it and put this script on AppLauncherEvent game object.

Or

You can use AppLauncherEvent prefab provided in Assets/App Launcher/Prefab/ folder. Just drag it in you hierarchy and everything will be ready for you.

OnSuccess will be called when application launched successfully

```
void OnSuccess(string message)
{
}
}
```

OnError will be called when unable to launch application

```
void OnError(string message)
{
}
}
```

If you have any queries then you can write me at devesh.pandey19@gmail.com