

Easy Alert for iOS

An easy to integrate plugin for displaying native iOS UIAlertView in Unity

Document History

| Date | Author | Version |
|----------------------|--------------------------|--------------------------|
| 11 March 2014 | Preet Kamal Singh Minhas | Initial Release |
| 04 March 2015 | Preet Kamal Singh Minhas | 1.1, updated for Unity 5 |

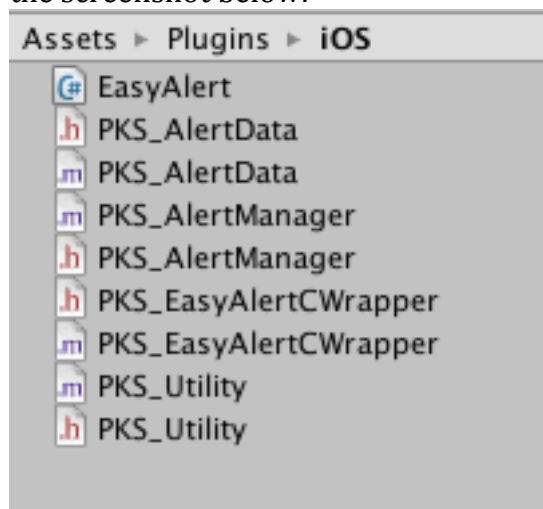
- Document History 1**
- Introduction 3**
- Integrating Easy Alert with your Unity project..... 3**
- Using Easy Alert 3**
- Easy Alert exposed methods 4**
- Alert Styles..... 4**
- Easy Alert events..... 4**
- Advanced 4**
- Plugin Architecture 4**
- Suggestions/Queries? 5**

Introduction

This plugin aims to make the process of displaying native alerts in iOS simple & easy.

Integrating Easy Alert with your Unity project

1. Once you have imported the EasyAlert package into your project, you must ensure that Assets->Plugins->iOS folder contains the files listed in the screenshot below:



2. The EasyAlert API provides the selected button index & the user input string values in a callback. In case you want to receive the callback, your scene must have a GameObject with the EasyAlert.cs script attached to it. This script receives all the callbacks from the native code.
3. You are now ready to use Easy Alert in your project!

Using Easy Alert

1. Display an alert by using the following method call:
EasyAlert.ShowNativeAlert(int tag, AlertStyle style, string title, string message, string cancelButtonTitle, string[] otherButtonTitles);
2. In case you want to receive the alert callbacks, the scene must have a GameObject with the EasyAlert.cs script attached to it.
3. Configure the event handler in your script. Sample JS code:

```
function Start () {  
    //start listening for the event  
    EasyAlert.alertDismissedEvent += AlertDismissed;
```

```

    }

    function OnDestroy() {
        //stop listening to the event
        EasyAlert.alertDismissedEvent -= AlertDismissed;
    }

    function AlertDismissed(alertTag : int, buttonIndex : int, string0 : String, string1 : String) {
        Debug.Log(String.Format("Tag {0}, buttonIndex {1}, string0 {2}, string1 {3}", alertTag, buttonIndex, string0, string1));
    }

```

Easy Alert exposed methods

//Displays the native alert

```

public static void ShowNativeAlert(int tag, AlertStyle style, string title, string message, string cancelButtonTitle, string[] otherButtonTitles);

```

Alert Styles

Easy Alert supports all 4 types of alerts available in iOS 5+:

```

public enum AlertStyle{
    Default = 0,
    SecureTextInput,
    PlainTextInput,
    LoginAndPasswordInput
};

```

Easy Alert events

//The delegate definitions

```

public delegate void AlertDismissedEventHandler(int alertTag, int buttonIndex, string string0, string string1);

```

//Events

//Called when an alert is dismissed

```

public static event AlertDismissedEventHandler alertDismissedEvent;

```

Advanced

Plugin Architecture

PKS_EasyAlertCWrapper : C Wrapper over the Objective-C methods

PKS_AlertManager : Methods to expose UIAlertView functionality as a plugin

PKS_Utility : Utility methods to convert char* to NSString* and vice versa.

Suggestions/Queries?

Please write in to contact@marchingbytes.com