FGL SDK for HTML5: Official Construct 2 Plugin

The official FGL Construct 2 Plugin allows the easy use of the FGL SDK features from within Construct 2.

The condition, action, and expression names are designed to match the function names in the Javascript SDK, so you may also refer to the Javascript SDK implementation guide and API reference for additional information.

Version 1.6 August 7th 2014

Obtaining the Plugin

Download the plugin package from FGL.com/html5

Package contents:

fglsdk.c2addon
The FGL SDK Plugin - drag this file into Construct 2

FGL Plugin for Construct 2.pdf
This Guide

construct2-sample
A sample Construct 2 project

Quick Start

1. Drag the `fglsdk.c2addon` file into your Construct 2 window. This will prompt you to install the plugin. Restart Construct 2 when installation is complete.
2. Add the `FGL` object to your game - right click in the **Objects pane**, and click **Insert new object**. Choose `FGL` under the **Platform speficic** heading.
3. You can now begin to add SDK functionality to your game's event sheets.

   Open the construct2-sample capx file and view the event sheets to see how to properly implement the SDK features.

Construct 2 Reference

The following pages describe each condition, action, and expression available on the FGL Construct 2 object.

Contents

- Conditions
  - Is branding enabled
  - Is cross-promotion enabled
  - Is premium
  - Has achievement
  - Is unlock enabled
  - On unlock success
  - On unlock failure
- Actions
  - DisplayScoreboard
  - HandleBrandingClick
  - InitiateUnlockFunction
  - ShowAd
  - ShowMoreGames
  - SubmitScore
  - GrantAchievement
- Expressions
  - BrandingLogo

Conditions

Is branding enabled

This checks if sponsor branding is enabled. The condition will be true if sponsor branding should be shown - check this before showing any sponsor branding.

Is cross-promotion enabled

This checks if cross-promotion is enabled. The condition will be true if a more games button should be shown - check this before creating a more games button or showing the more games cross-promotion menu.

Is premium

This checks if the game is running a premium (aka paid-for) version. This condition will be true if the player has purchased the game through an in-app unlock. Check this before enabling premium features, hiding 'buy' buttons and disabling ads in the premium version of your game.

Has achievement

This checks if the player has previously been granted an achievement.

Conditions

Is unlock enabled

This checks if unlocking a premium version of the game is enabled. This condition will be true if it should be possible for the player to attempt to buy a premium version of the game. Check this before creating any 'buy premium version' buttons or options.

On unlock success

This will trigger after an InitiateUnlockProcess action, if the player has successfully purchased the premium version of the game. The 'Is premium' condition will also become true. Use this to update your game to the premium version immediately and do anything else you want upon successfully purchasing the game.

On unlock failure

This will trigger after an InitiateUnlockProcess action, if the player has failed to purchase the premium version of the game for any reason. This is purely optional as the SDK will show anything necessary to inform the user that the purchase failed.

Actions

DisplayScoreboard

Displays the scoreboard for the game.

HandleBrandingClick

Contains behaviour that should occur when sponsor branding is clicked. If sponsor branding is showing, please add a click condition to the sponsor branding sprite and use this action for that event. Normally this will take the user to the sponsor's webpage.

InitiateUnlockFunction

This will begin the unlock process for buying the premium version of the game. This should only ever be called if the 'Is unlock enabled' condition is true (in fact, if this condition is false or if 'Is premium' is true then all 'buy' buttons should be hidden).

ShowAd

Show an ad from the SDK's ad rotations. This should never be called in premium versions (if the 'Is Premium' condition is true or if 'On unlock success' has fired previously during the game).

Actions

ShowMoreGames

Show the SDK cross-promotion (more games) menu. This should only ever be called if the 'Is cross-promotion enabled' condition is true (in fact, all 'more games' buttons should be hidden if this condition is false).

SubmitScore

Submit a score to the leaderboard. This will also display the leaderboard after submitting the score.

GrantAchievement

Grant an achievement to the player.

Expressions

BrandingLogo

Showing the sponsor branding logo requires loading the logo from a url using this expression. To do so, add an action, then select the object that you want to use for the Branding Logo (should be a Sprite object but can have any image as it will be overwritten). Choose 'Web' - 'Load image from URL'. In the 'URI' field, type the following:

```
FGL.BrandingLogo
```

Make sure that 'Resize to image size' is selected, then hit OK. Your chosen Sprite object should now load the sponsor branding logo as appropriate. It is highly recommended that this is done with a 'System' - 'On start of layout' condition as in the provided sample project. Finally, be sure to check if the 'Is branding enabled' condition is true before showing any sponsor branding!