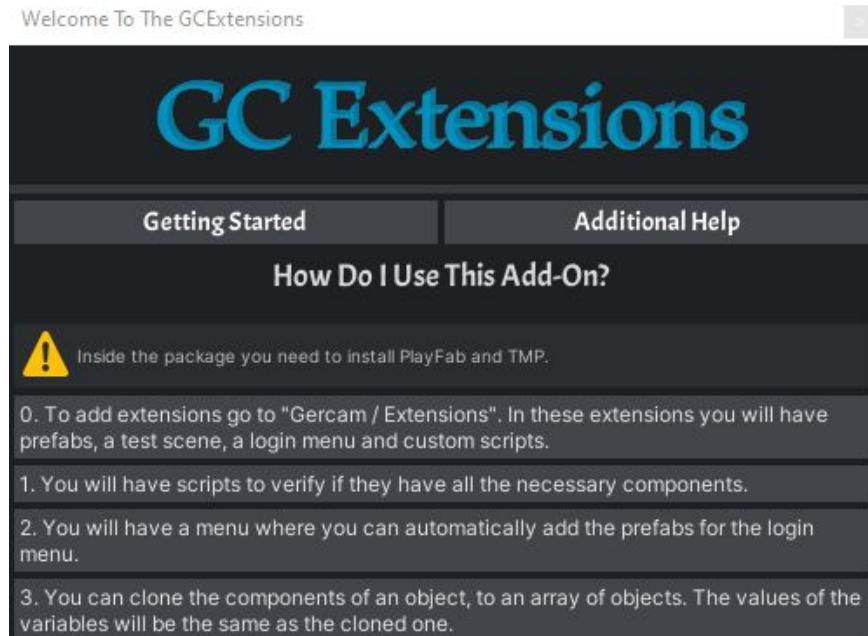


## **GUIDE:**

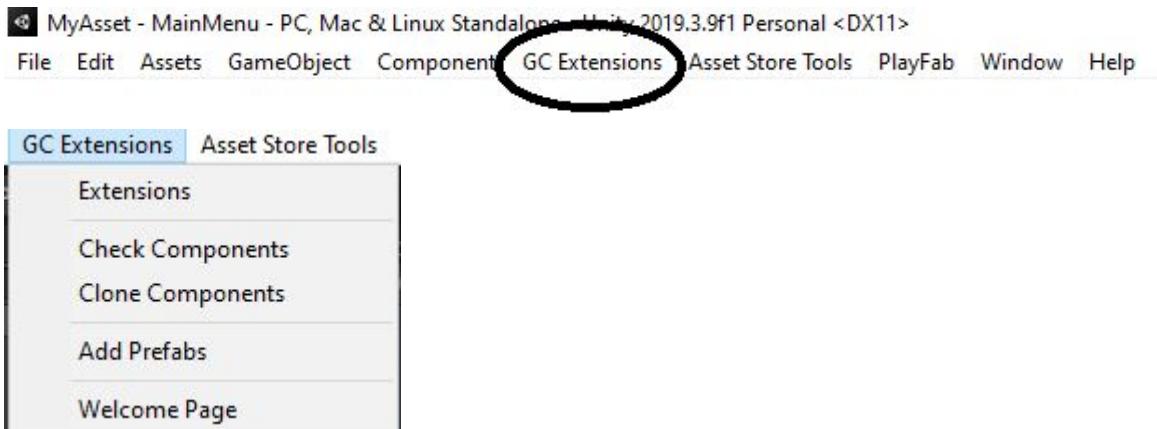
1. First Steps
2. Windows
3. Installation of playfab
4. Setup Playfab
5. Add extensions of GCExtensions for playfab
6. Setup login scene
7. Setup game scene

# First Steps

When you install the asset, the first thing that appears is a welcome window, where I explain that you have this asset. Then in this same window there is a help section, where you can find how to communicate with me.



After importing the asset you will see a menu called GCExtensions, there are all the available windows of this asset.



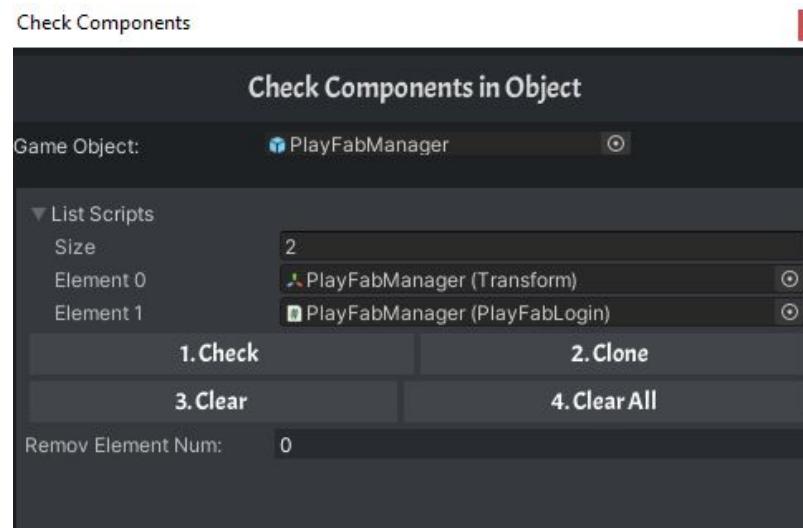
## Windows

The first time you install the asset you will not have the Add prefabs window, since this window is achieved when you add the playfab extension.

What is each window for:

**Extensions**→ This window is mainly used to remember that if you want to use this extension you must install playfab and the TMP before installing the extensions for playfab.

**Check Components**→ This window is used to view all the components that an object has. To know all the components it has, you must put the object in the input gameObject. When you put it you will see 4 options.



1. **Check**→ It is used to see all the components that the object has.
2. **Clone**→ It opens a window that serves to clone all the components, below we explain how it works.
3. **Clear**→ In the Remove input you enter the number of the List Scripts array and hit Clear, then it will only remove the

element that you have told it.

4. **Clear All**→ Remove all components from the object.

**Clone Components**→ It is used to clone all the components of an object to an array of objects.

1. First you will have to select all those objects that you want to be cloned.
2. Once selected you have to click on "Add Objects to Clone".
3. Select the object that you want to copy the components.
4. You have a button to select all the components you want to copy and you can remove which components you don't want to copy.
5. Then you give "1. Copy Components" and then a "2. Paste Components".

**Add Prefabs**→ In this window you can put in your scene those prefabs you want (only those of the playfab).



Remember to instal TMP, to install go in unity Window>TextMeshPro and install both

## Installation of PlayFab

1. Go to this link ([PlayFab](#)). You can also access from the GC Extensions> Extensions> Add Playfab window. In this link you will give Download Editor Extensions.
2. It will download a package. You will import it into your project and a window will open for you. If it does not open you will go to Windows> PlayFab> EditorExtensions.
3. You will need to log in. Once started you will go to the SDK section of the PlayFab Editor Extensions, and download the SDK.
4. After installing the SDK, you will go to the Settings / Project section, and you will select the studio, the title ID and we will remove the Compress API Data.

## Setup PlayFab

### Check Player

For all GC Extensions for playfab scripts to work, we will go to our playfab title id (web page), and we will go to Automation> Cloud Script> Revisions. At the bottom of the code we will put this:

```
// Compare if the player is online or offline.  
handlers.SetOnlineState = function(args, context)  
{  
    var isOnline = args.OnlineState;  
    server.UpdateUserData({PlayFabId: currentPlayerId, Data: {"IsOnline" :  
isOnline}});  
};
```

## Rule

First of all we need to add a template for the rule. We will go to the playfab web page Contents> Email Templates> New Email Template.

The safe options that you have to put are the following:

Template type → Email verification

Email body → paste this → <p>**Please click** <a href="\$ConfirmationUrl\$">**here**</a>  
**confirm your email. Thanks!**</p>

IMPORTANT, Words in bold are the ones you can modify.

We also need to add a rule. We will go to the website of our title ID, Automation> Rules> New Rule. And we will leave it like this:

The screenshot shows the 'Edit Rule' page in the PlayFab web interface. At the top, there is a navigation bar with tabs: Cloud Script, Rewarded Ads, A/B Tests, Rules (which is selected and highlighted in grey), and Scheduled Tasks. Below the navigation bar, the page title is 'Edit Rule' and the breadcrumb path is 'Rules > Edit Rule'. The main configuration area starts with a 'Name \*' field containing 'Verify Email'. Next is an 'Event type \*' dropdown set to 'com.playfab.player\_updated\_contact\_email'. A note below the dropdown states 'Custom namespaces must start with "title.16175."'. There is also a link to 'Learn about built-in PlayStream events'. Under the 'CONDITIONS' section, there is a '+ Add group' button. Under the 'ACTIONS' section, there are two dropdowns: 'Type' set to 'Send email' and 'Email template' set to 'Verify Email Template'. Below these dropdowns is a 'Remove action' button. At the bottom of the actions section are '+ Add action', 'Save action' (in a blue button), and 'Cancel' buttons.

We will need to activate an Add-on. Go to Add-on>SMTP. Follow the instructions given by Google for this service ([Link](#)).

When you get to the Select devices option, select "Other (custom name)" and you will put PlayFab. You will give okey and a window will appear with a password, select the password and copy it. Then we will go to the SMTP window of playfab and we will put the following

HostName → smtp.gamil.com

Port number → 587

Username → Email

Password → The one you have copied previously

## Add extensions of GCExtensions for playfab

Once everything about playfab and the setup is installed, we will install the GCExtensions extensions, GC Extensions> Extensions> Add Extensions for Playfab.

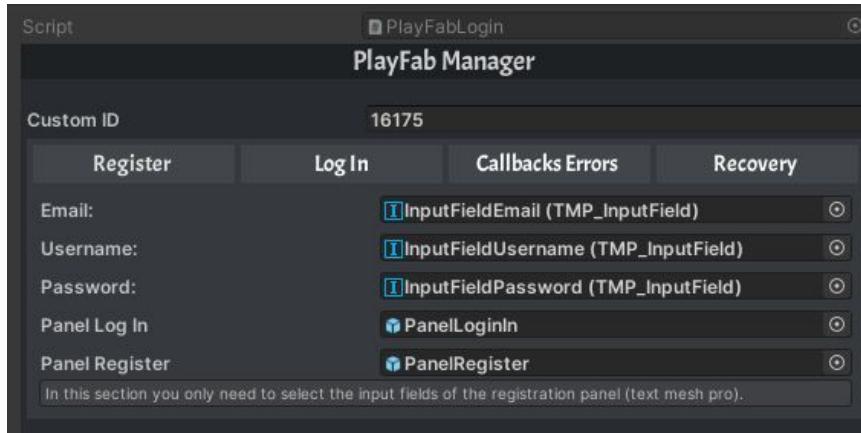
## Setup login scene

Inside the scene we will need to add the prefabs, GCExtensions> Add prefabs> We will add the first and second of the section "Prefabs for PlayFab login".

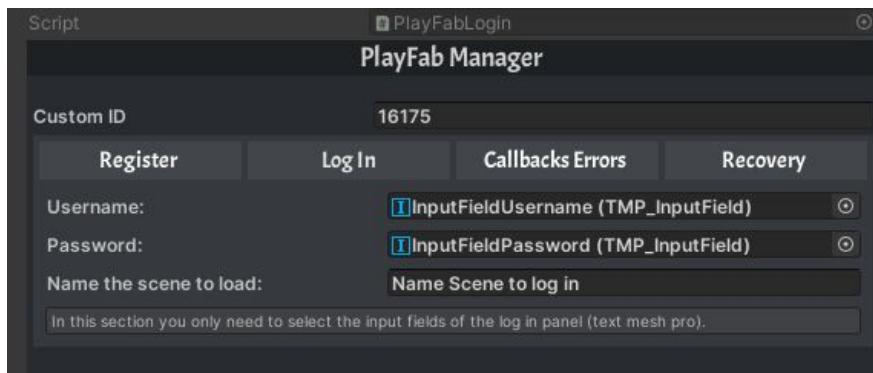
We select the PlayFab Manager from the hierarchy, and we will put all the components of the Canvas\_MainMenu

In the Custom ID we will put the Id of our study. To know which one we have, we will go to the playfab website, we will click on the gear and we will click on "My studios and title". We will get a list of all the studies and in each one of them the ID, we will copy that ID and put it in the Custom Id.

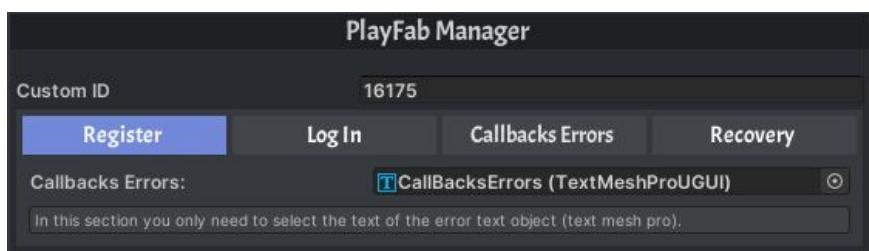
## Register Components



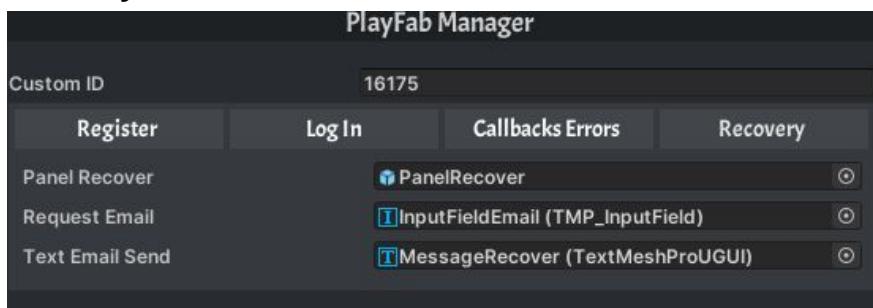
## Log In Components



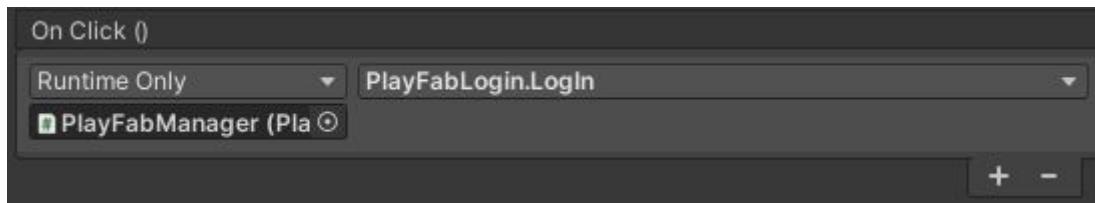
## Callbacks Errors



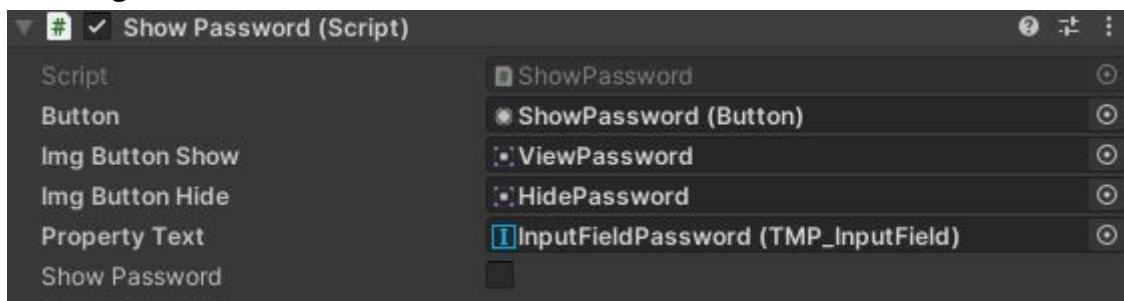
## Recovery



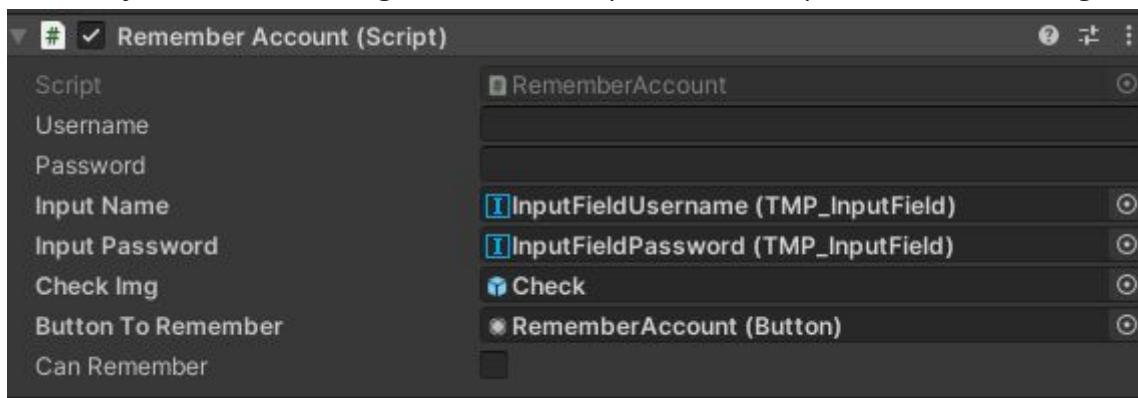
Inside the Canvas MainMenu, we will go to the Play button and the On Click () event the following.



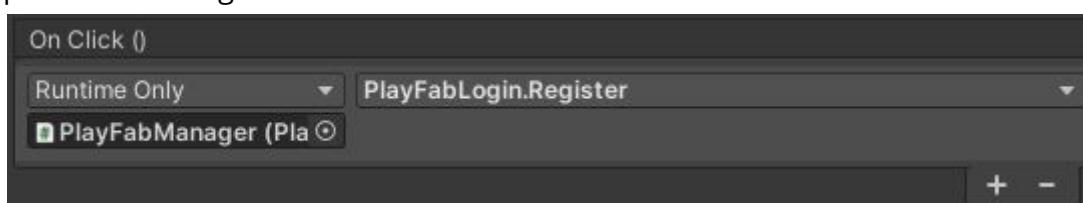
In the Login panel, in the show / hide password button, in the script we will put the following.



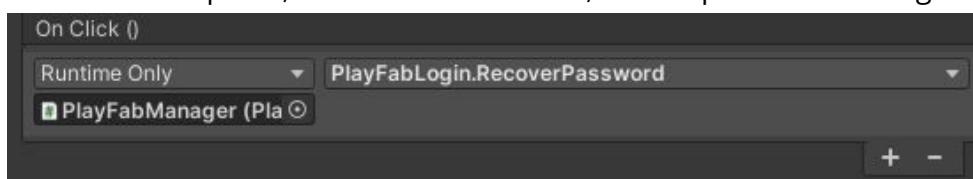
In the object called PanelLogin there is a script and we will put it in the following way.



In the registration panel, inside the registration button, in the OnClick () event we will put the following.



In the Recover panel, in the recover button, we will put the following event.



## Setup game scene

In the scene for the score there is a demo, just like the one for the login, but we explain the main thing to know how it works.

If we create it from 0, we will do the same, in a different scene we will add the prefabs.

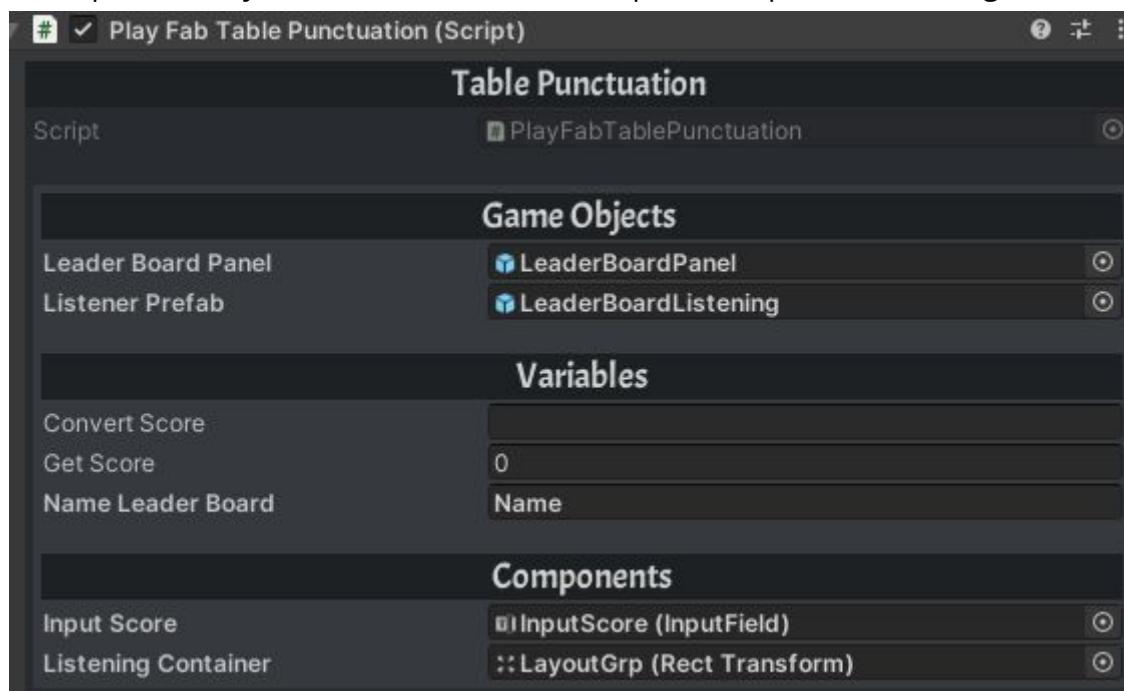
Inside the prefab LeaderBoardPanel> Update list, we will put the following events.



And in the AddScore button we will put the following event.



In the prefab "PlayFabPunctuation", in the script we will put the following.



IMPORTANT→ Inside the Assets / GCExtensions / GCExtensionsPlayFab / Scene folder, there are two demo scenes with all the setup