**Thanks for your purchase.**

**// FIRST OF ALL ///////////////////////////**

**You need Dotween (avalaible for free on the asset store) to use this template.**

**If that is not downloaded, the game does not function and you wont be able to get it to work.**

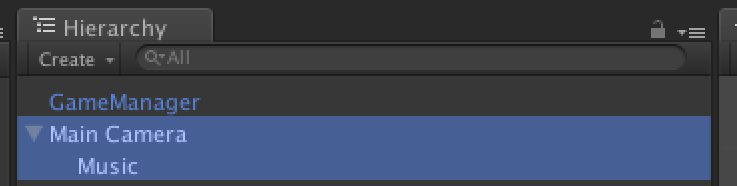
**// FIRST OF ALL ///////////////////////////**

**To begin, open the scene «NinjaHero».**

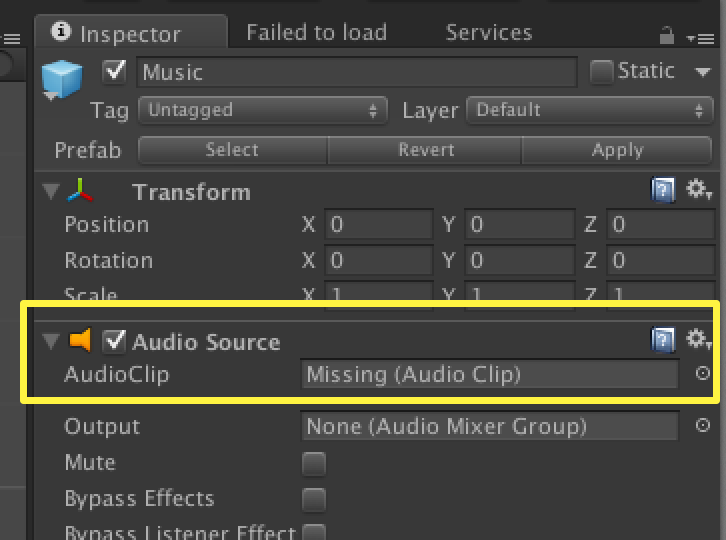


**1 - Music:**

To add a background music: Find the GameObject « Main Camera »,

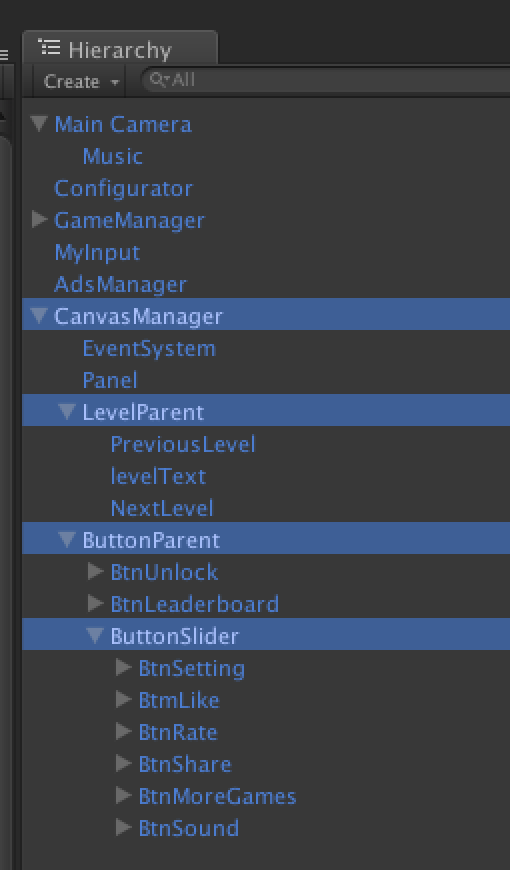


and add your music to his child « Music ».

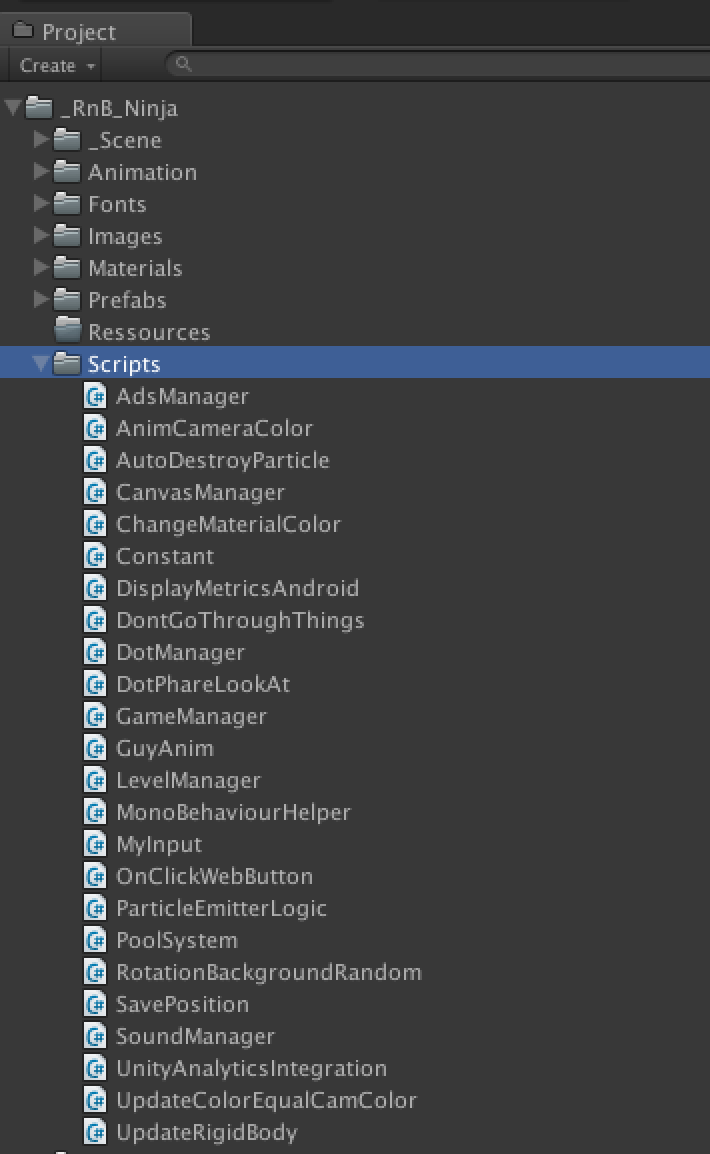


**2- Menu:**

The menu is in the GameObject « CanvasManager ». Check the childs, they are named logically.

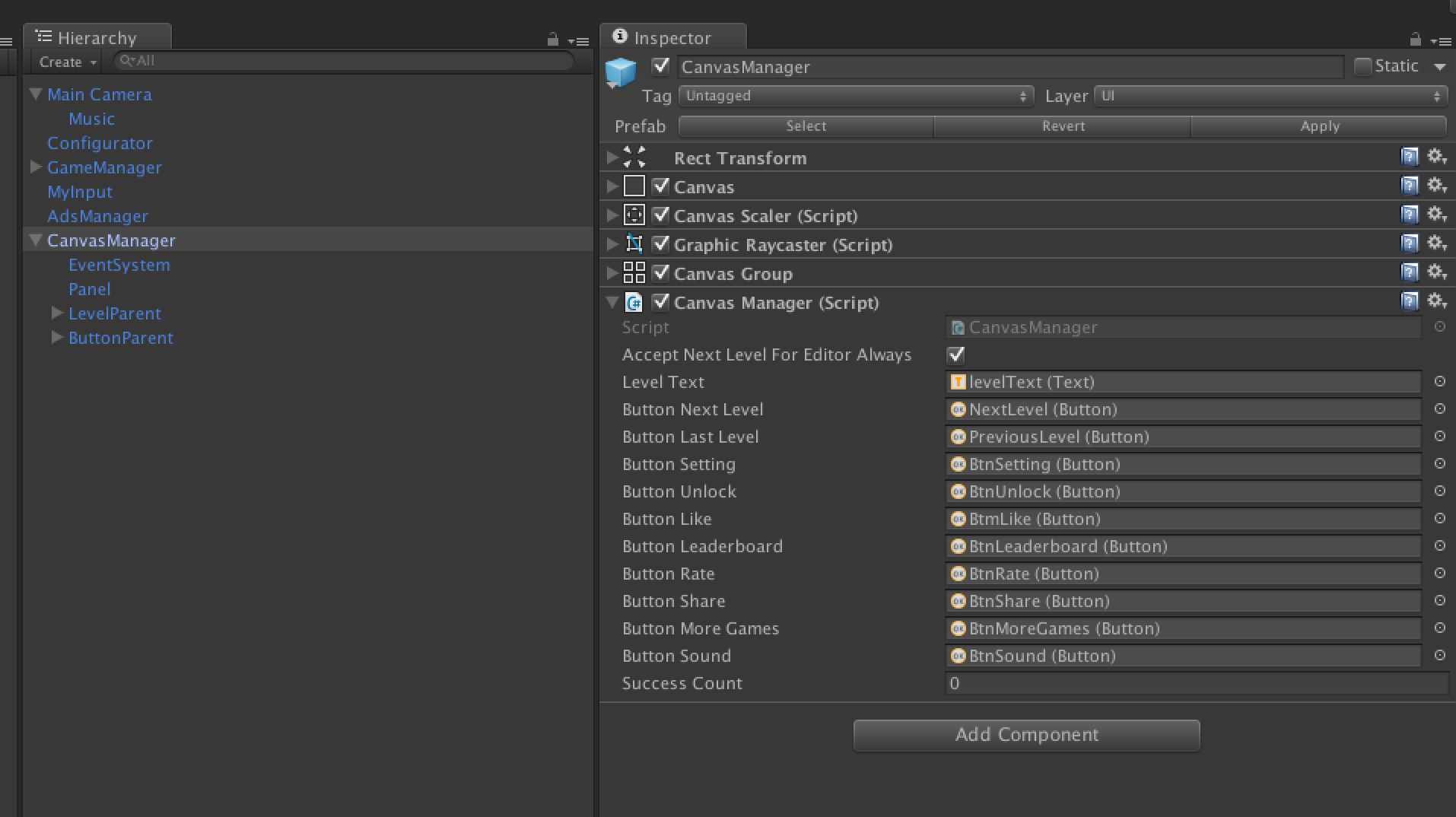


**3 - Scripts:**



**CanvasManager.cs:**

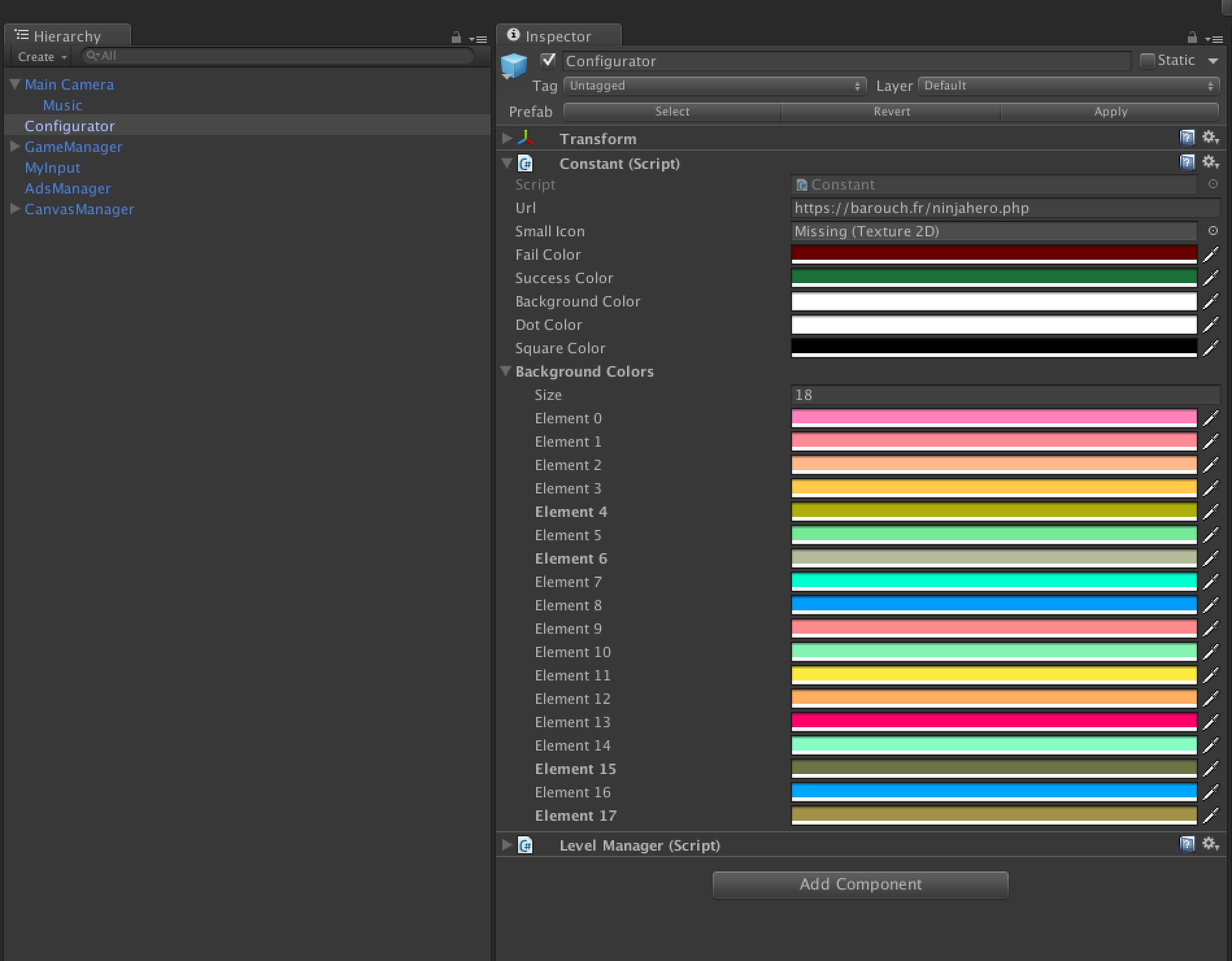
This script is attached to the GameObject « CanvasManager ».



**Constant.cs:**

Some constants use in the app.

This script is attached to the GameObject « Configurator ».



You can change all the color in this game object :

\* Fail color

\* Success Color

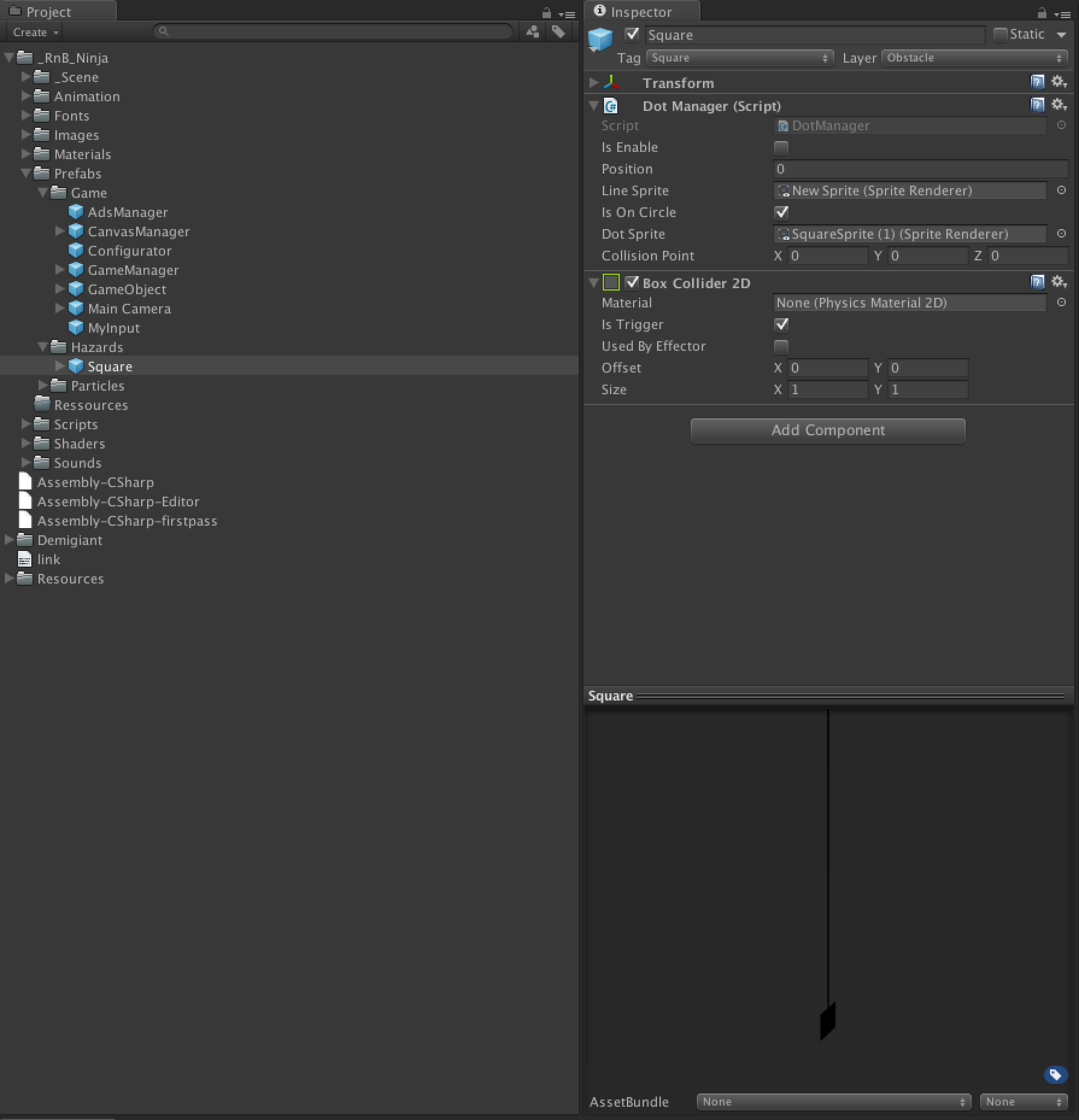
\* Background Color

\* Dot Color

\* Background Colors

**DotManager.cs:**

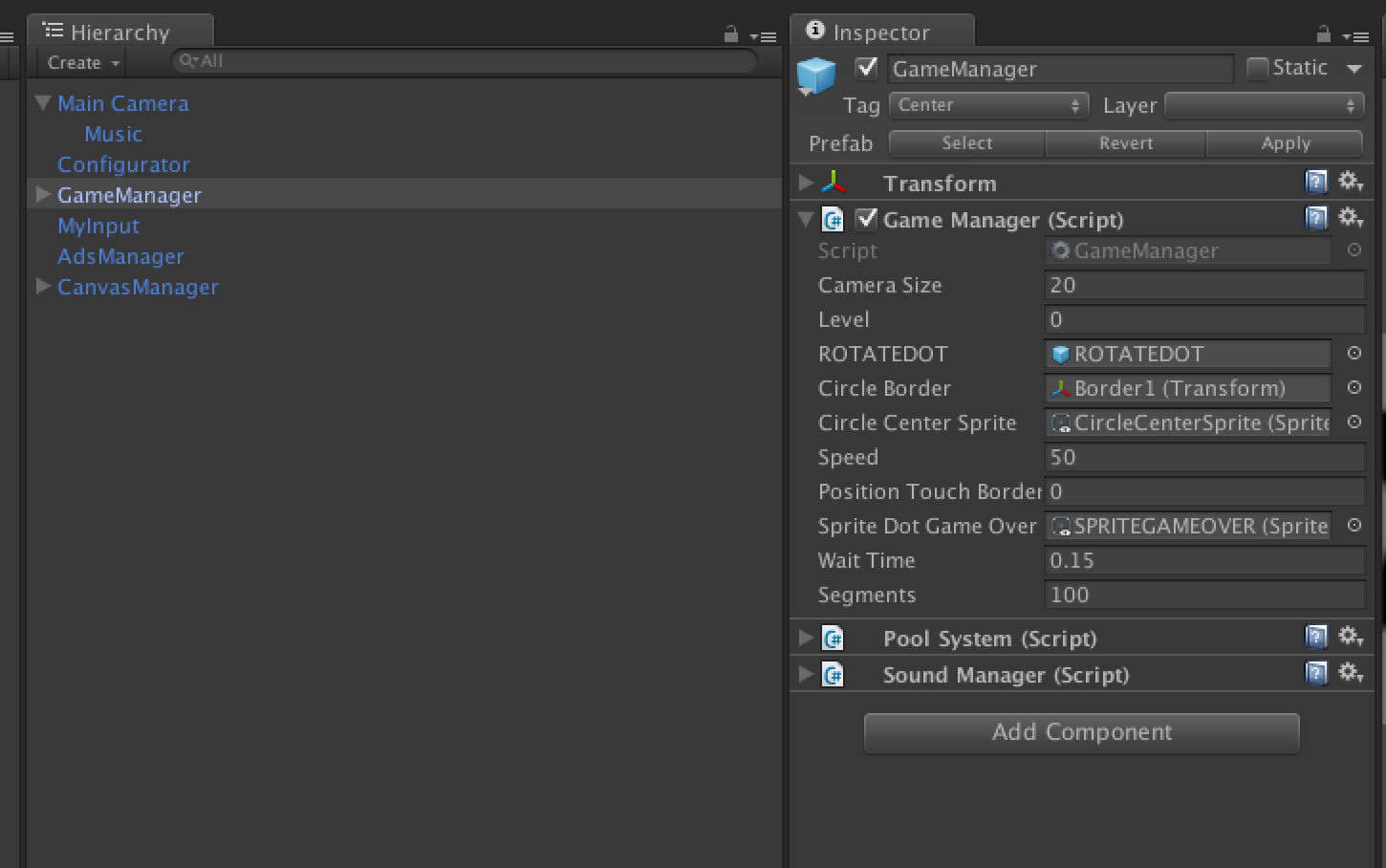
All the dot prefabs have a « DotManager.cs » attached.



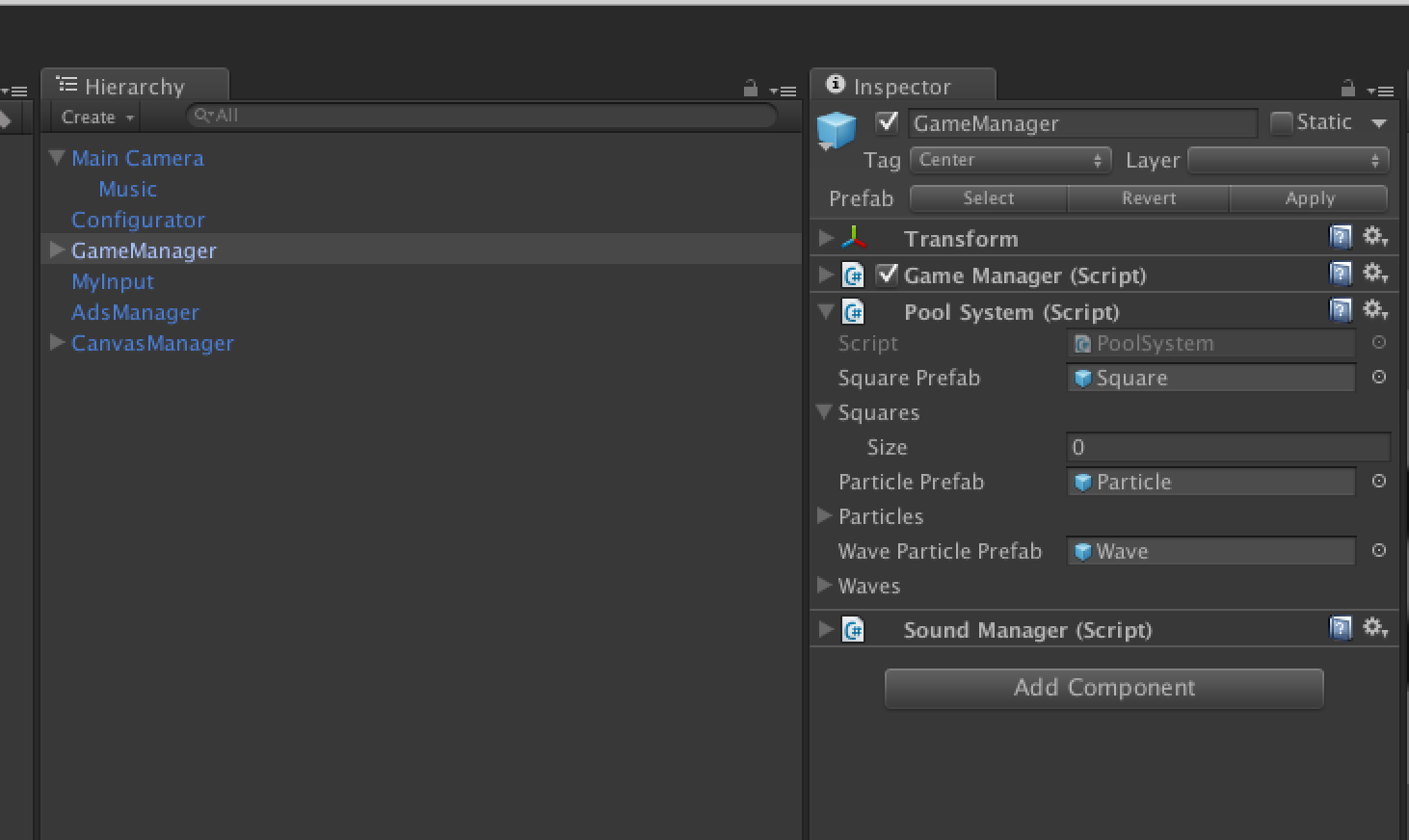
This script is responsible to detect collision, enable the hasard type (write or black, write to destroy, black to avoid).

**GameManager.cs:**

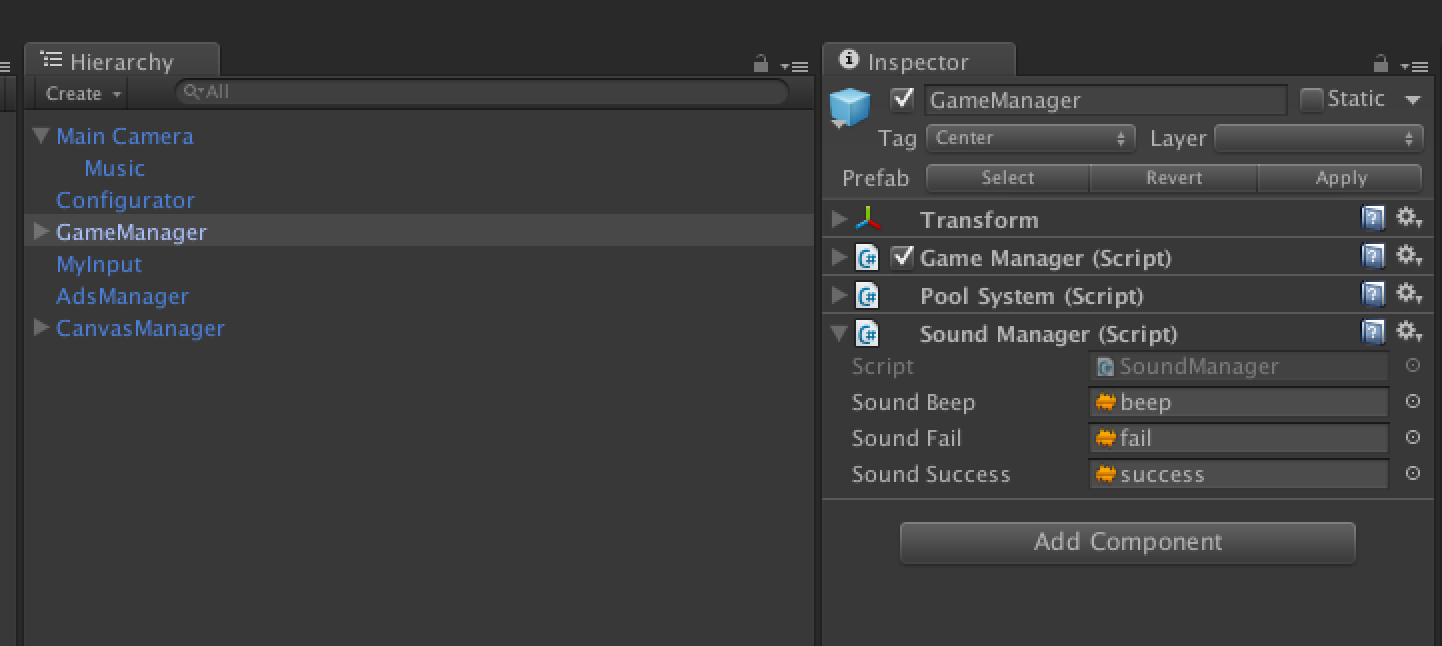
This script is attached to the GameObject « GameManager ».



This script is responsible to create game, and detect all the event in the game : Success / Game Over.

**PoolSystem.cs:**

Instantiate and destroy prefabs are not good. So we use this to do that to simple enable and disable prefabs. Search on google « Object pooling » or write me if you want to understand the principle.

**SoundManager.cs:**

Reponsible to play all the FX in the game.

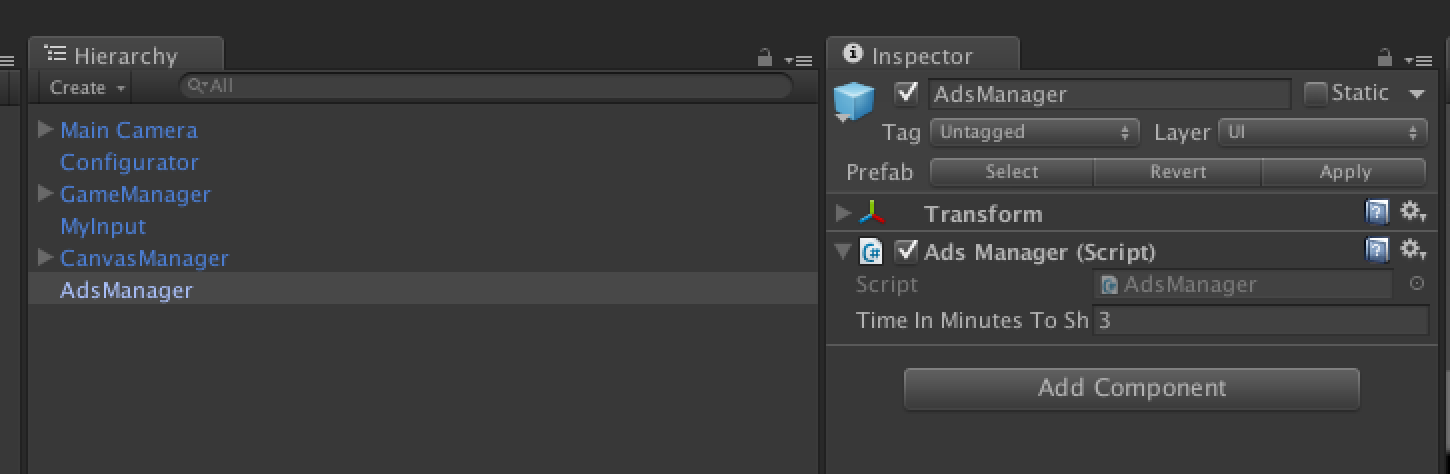
If you want to change them, do it here.

**MonobehaviorHelper.cs:**

Some class (like CanvasManager, GameManager..) derives from this class to have clearer code.

**AdsManager.cs :**

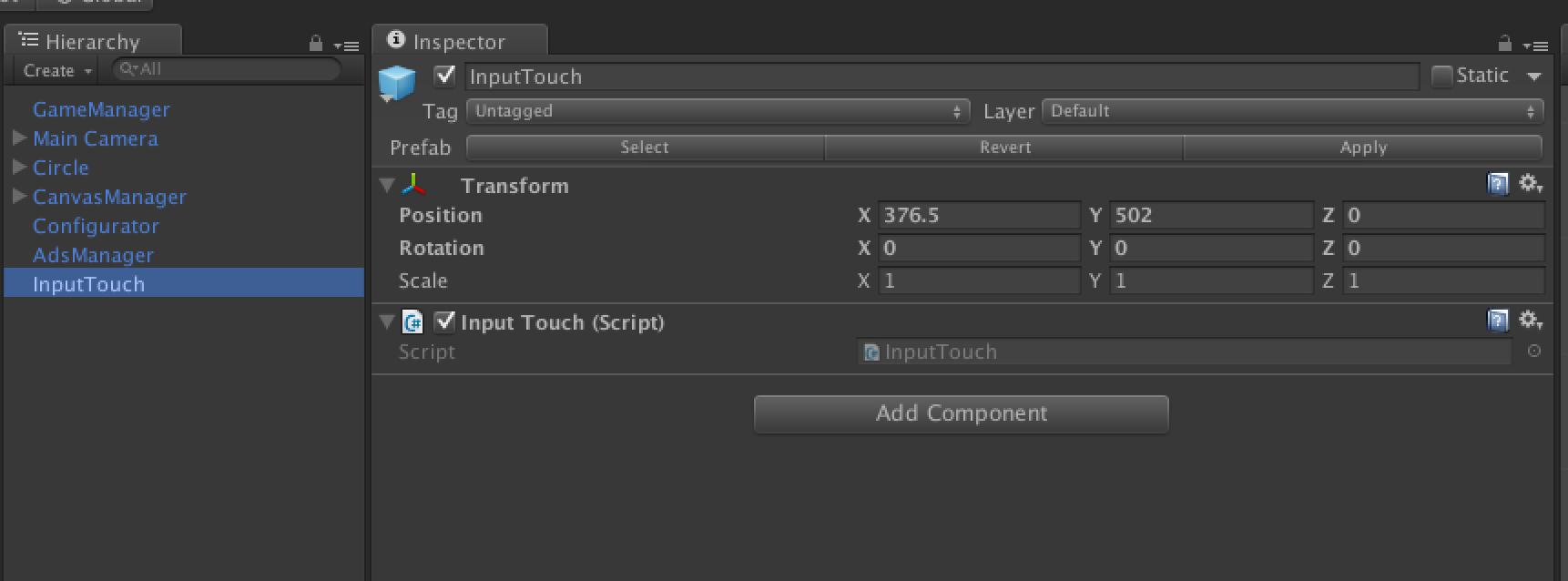
Script attached to the GameObject AdsManager, and responsible to display banners, interstitials, and rewarded videos in the game.



To more information about ads implementation, plase have a look to the \_Ads\_Integration\_Documentation.pdf.

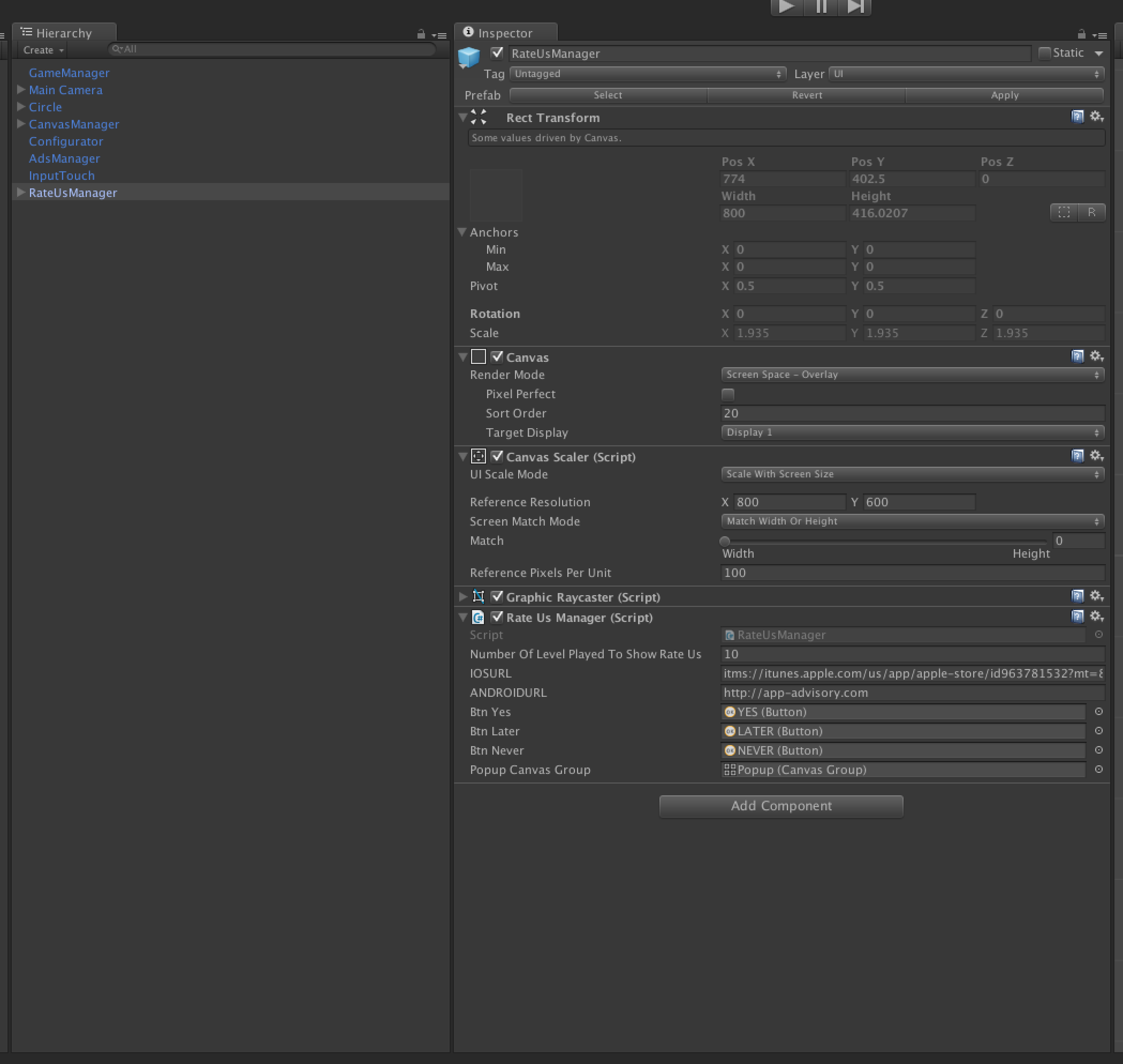
**InputTouch.cs :**

In charge to handle input in the game.



**RateUsManager.cs :**

In charge to handle input in the game.



Change the url to redirect to your app.

Change the « Number Of Level Played To Show Rate Us » to change the moment we prompt the pop up to ask to rate.

If you have any question, fell free to contact me :

contact@app-advisory.com

Thanks.