



UNITY GAME TEMPLATE STACK BREAKER

USER GUIDE



VERSION 1.0
CLAWBEAR GAMES
clawbeargames@gmail.com

Table of Contents

I. TEMPLATE INTRODUCTION	3
II. TEMPLATE SETUP.....	4
III. REQUIREMENT PACKAGES	6
IV. GAMEPLAY CUSTOMIZATION	7
1. INGAME MANAGER	7
2. PLAYER CONTROLLER.....	7
V. LEVEL CUSTOMIZATION.....	8
VI. SERVICES CONFIGURATION.....	11
1. SOUND MANAGER.....	11
2. AD MANAGER	12
3. UNITY ADS CONTROLLER	15
4. ADMOB CONTROLLER.....	16
5. SHARE MANAGER	17
6. LEADERBOARD MANAGER	19
7. CUSTOMIZING UI	21

I. TEMPLATE INTRODUCTION



Stack Breaker is an exciting endless level-base game in which you touch and hold the screen to make player fall down on a colored block, try not to hit the black block or you will die, try to break as many block as you can to enable immortal mode and complete the level. The game is ready to release straight out of the box, and it can also be easily customized to make it even more engaging to your players. Supports for PC/Mac, iOS, Android, etc...!

The game is made with Unity C# and optimized for mobile devices. This template provides you a full-featured, ready-for-release game source code that you can customize and build your own game in no time!

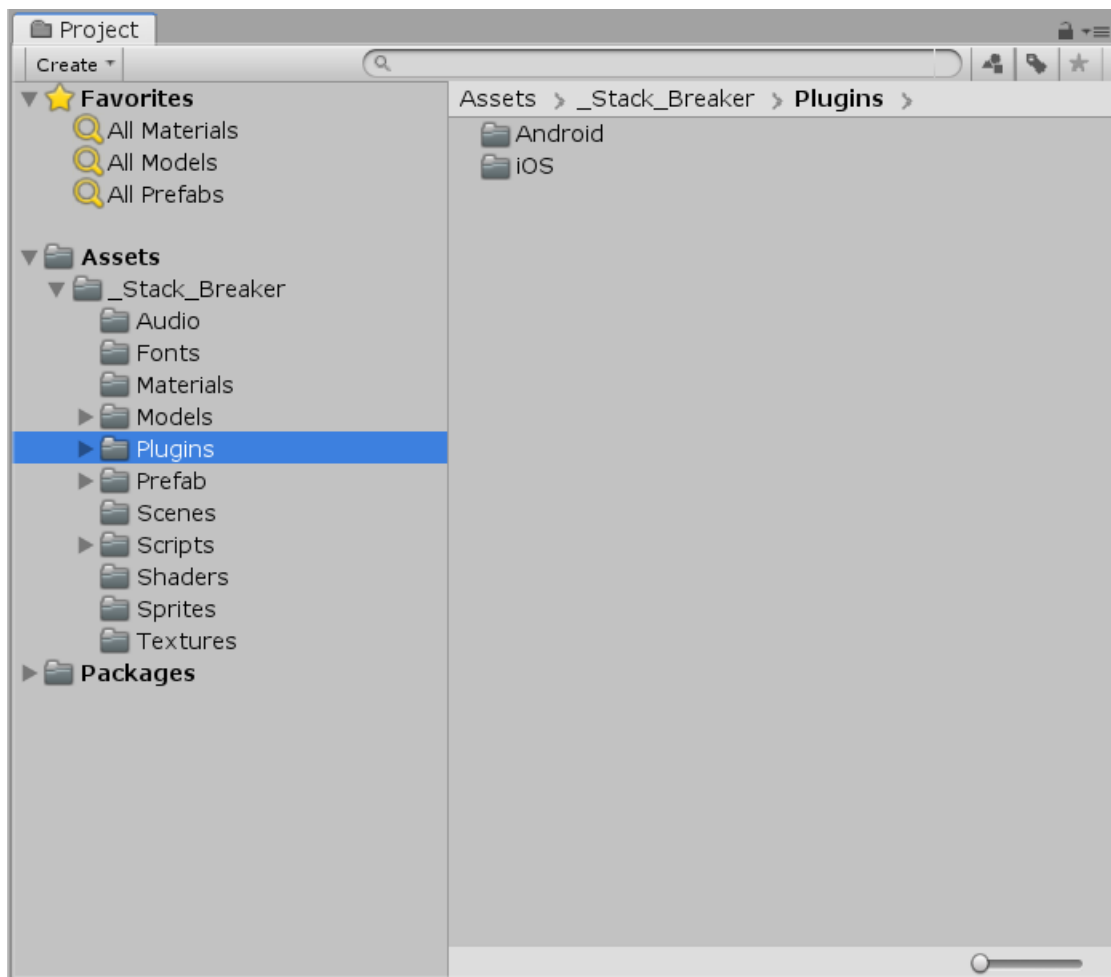
Highlight features:

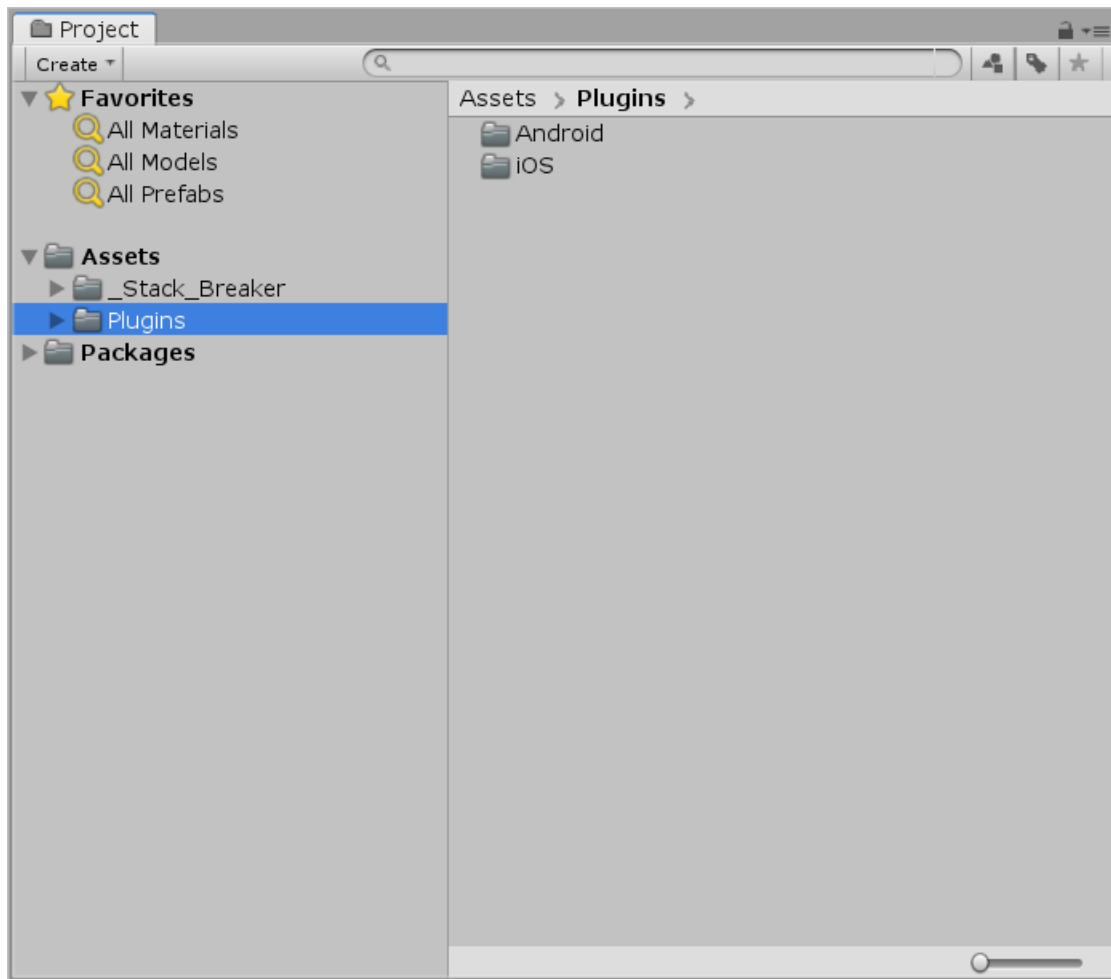
- ❖ Addictive one-touch gameplay
- ❖ Smooth control, eye-catching graphics
- ❖ 100++ different difficulty levels
- ❖ Very easy to add new levels or edit existing ones
- ❖ Leaderboard system using Dreamlo
- ❖ Watch ad to revive
- ❖ Multiple ad networks: Admob and Unity Ads (banner, interstitial and rewarded video) ready to use
- ❖ Native share Android/iOS
- ❖ Facebook/Twitter share
- ❖ Optimized for mobile
- ❖ Detailed documentation and commented C# code
- ❖ Free-to-use assets (fonts, sounds, music, models, etc.)
- ❖ Ready to publish out-of-the-box

II. TEMPLATE SETUP

This template was designed for mobile (Android, iOS, Windows Phone...) so after imported the package to unity, you need to switch to Android or iOS, or Window Phone.

After that, go to Assets/_Stack_Breaker and move the folder Plugins out of _Stack_Breaker folder.





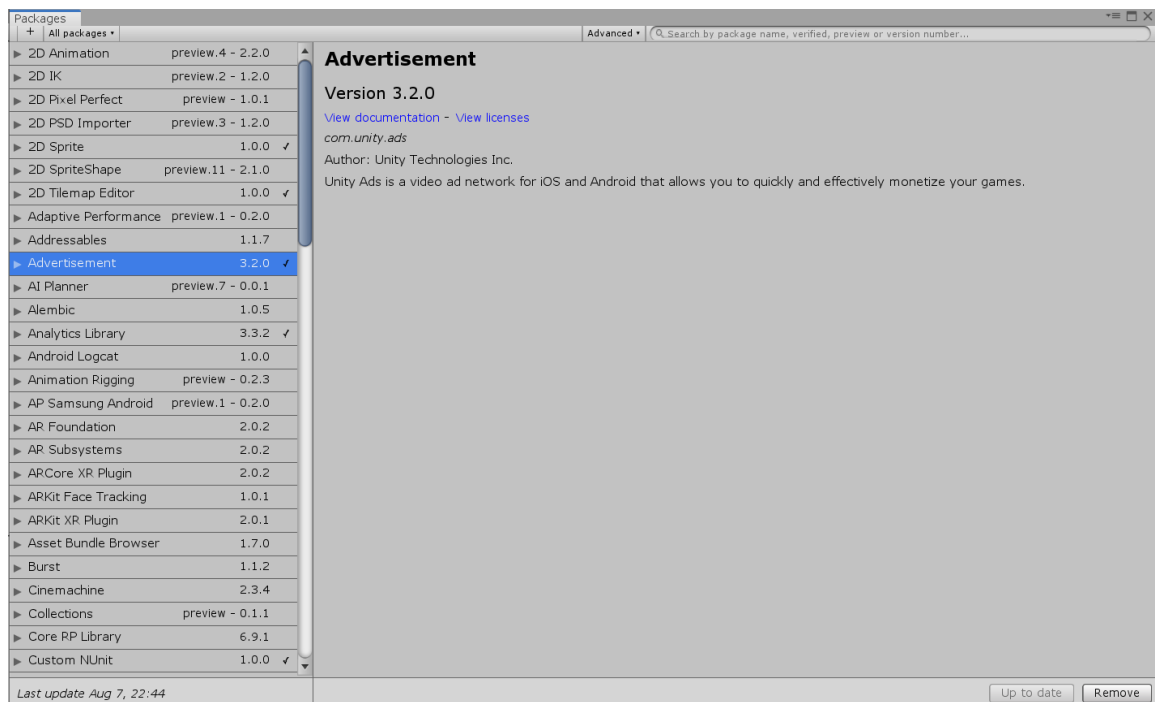
The template contains 3 scenes: Home, Loading, Ingame under the path Assets/_Stack_Breaker /Scenes. You need to start from Home scene.

Scenes In Build	
<input checked="" type="checkbox"/> _Stack_Breaker/Scenes/Home	0
<input checked="" type="checkbox"/> _Stack_Breaker/Scenes/Loading	1
<input checked="" type="checkbox"/> _Stack_Breaker/Scenes/Ingame	2

III. REQUIREMENT PACKAGES

When you open this template, at first you will see some errors in Console window, that's because this template requires some packages to run. You have to install these packages below to have this template run smoothly. Please follow these instructions:

- Open Package Manager by going to Window/Package Manager. If you don't see Package Manager button, then you have to re-install your Unity completely because it's the problem that Unity Technologies still not able to fix.



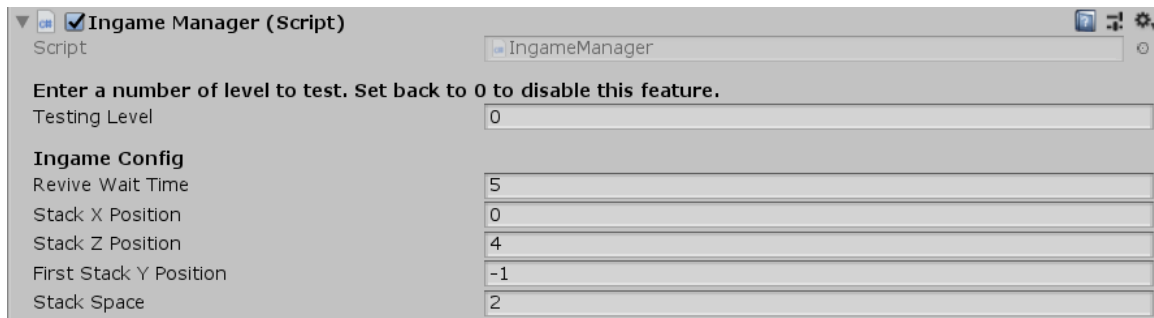
- These are packages you need to install:
 - ❖ Advertisement version 3.4.2 or higher.
 - ❖ Analytics Library version 3.2.2 or higher.
 - ❖ Text Mesh Pro version 1.3.0 or higher.

After you installed these packages, there's maybe some errors in Console window. If you see errors in Console window, go to File - > Build Setting -> Player Setting and change the Scripting Run Time Version to .Net 4.x Equivalent.

IV. GAMEPLAY CUSTOMIZATION

1. Ingame Manager

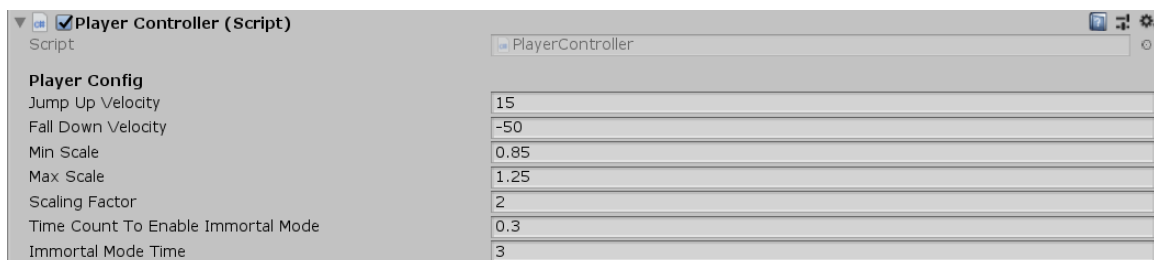
Most of important gameplay parameters can be configured within the IngameManager component which is attached to a game object also named IngameManager in the hierarchy. You can find IngameManager object in InGame scene.



- ❖ *Testing Level*: the level you need to test before building the game. If you set it to 0, the script will not load the testing level.
- ❖ *Revive Wait Time*: the delay time for revive feature last.
- ❖ *Stack X Position*: the position of x axis of all stacks will be created in the game.
- ❖ *Stack X Position*: the position of z axis of all stacks will be created in the game.
- ❖ *First Stack Y Position*: the position of y axis of the first stack will be created in the game.
- ❖ *Stack Space*: the space between each stack (only effected on y axis).

2. Player Controller

Most of important parameters of player can be configured within the PlayerController component which is attached to a game object named Player in the hierarchy.

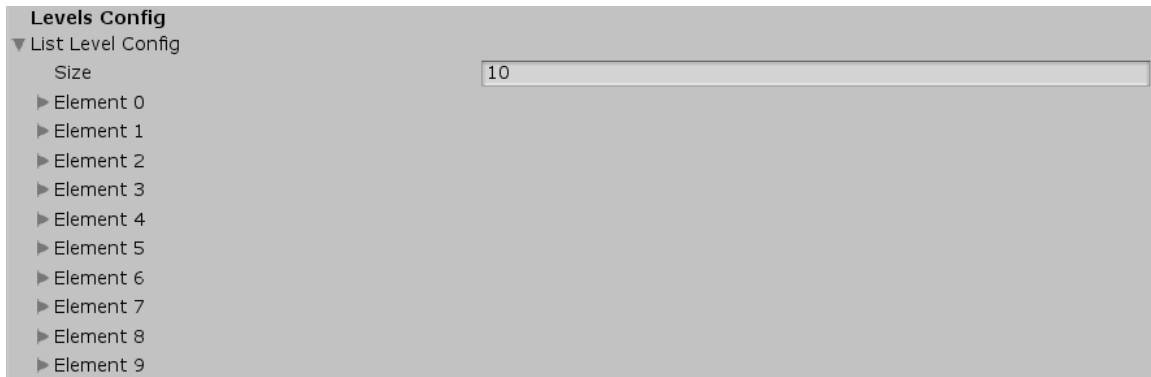


- ❖ *Jump Up Velocity*: the velocity of player when it jumps.
- ❖ *Fall Down Velocity*: the velocity of player when it falls.
- ❖ *Min Scale*: the minimum scale of the player.

- ❖ *Max Scale*: the maximum scale of the player.
- ❖ *Scaling Factor*: the scaling factor of the player.
- ❖ *Time Count To Enable Immortal Mode*: the time player needs to enable immortal mode. This parameter only counts when user holding the screen (player breaking the stack), not bumping on the stack.
- ❖ *Immortal Mode Time*: the time for player performs immortal mode.

V. LEVEL CUSTOMIZATION

This is a level-base game template and the goal of the game is you have to complete a level and start a new level, try to collect coins and avoid traps and obstacles. All the parameters of the level were show on the inspector for you to adjust and modify. They are stored in IngameManager script and you can adjust all the levels in the inspector of IngameManager object which you can find it in Ingame scene.



As you can see, there's a list of 10 configuration parameters of levels, these parameters are already designed and ready to use. If you want to add new configs, just resize the list and hit Enter. Now I will show you the parameters of each level.

▼ Element 0

Level Number Config

Min Level

Max Level

Background Sprite Config

Background Sprite

Colors Config

Player Color

Deadly Part Color

Background Top Color

Center Pillar Config

Min Center Pillar Rotating Speed

Max Center Pillar Rotating Speed

Min Center Pillar Rotating Time

Max Center Pillar Rotating Time

Stack Configs

► List Stack Config

Level Number Config: this section is about the configuration of level number, each level on this range will use these config parameters below to create the level.

- ❖ *Min Level:* the minimum level of this config.
- ❖ *Max Level:* the maximum level of this config.

All the level between this range will use these parameters below to create the level.

Background Sprite Config: this section is about the configuration of the sprite using as background in this level.

- ❖ *Background Sprite:* the sprite using as background.

Colors Config: this section is about the configuration of all the colors needed in this level.

- ❖ *Player Color:* the color of the player.
- ❖ *Deadly Part Color:* the color of deadly part in this level.

Center Pillar Config: this section is about the configuration of the center pillar. These parameters below you allow you config the rotation of the pillar.

- ❖ *Min Center Pillar Rotating Speed & Max Center Pillar Rotating Speed:* the minimum and maximum rotating speed of the center pillar. The actual value will be randomized between these two values.

- ❖ *Min Center Pillar Rotating Time & Max Center Pillar Rotating Time:* the minimum and maximum rotating time of the center pillar. The actual value will be randomized between these two values.

At first, the center pillar will random a direction to rotate with a randomized speed, it will rotate for a randomized time and stop, then randomized again the time and the speed, then rotate the opposite direction at the start. It will keep rotating like that until player completed the level. It will not stop if the player died.

Stack Configs: this section is about the configuration of all stacks will be created in this level. As you can see there's 3 elements of *List Stack Config*, so there's will be 3 blocks of stacks will be created. Each block has its own stack number, which define how many stacks will be created and other parameters to config the behavior of that block of stacks. You can add more block by resize the *List Stack Config* and config the

List Stack Config: this is the list of all configuration parameters of the block of the stacks. Here's the explanation of all parameters.

The screenshot shows a configuration window titled "Stack Configs". It contains three main sections:

- ▼ List Stack Config**:
 - Size: 3
- ▼ Element 0**:
 - Min Stack Number: 20
 - Max Stack Number: 25
 - Stack Type: STACK_8_PARTS (with a dropdown arrow)
 - Stack Color: A blue color bar with a selection icon.
 - First Stack Angle: A slider set to 10.
 - Rotation Change Amount: 5
- ▼ List Index Of Deadly Part**:
 - Size: 2
 - Element 0: 0
 - Element 1: 1

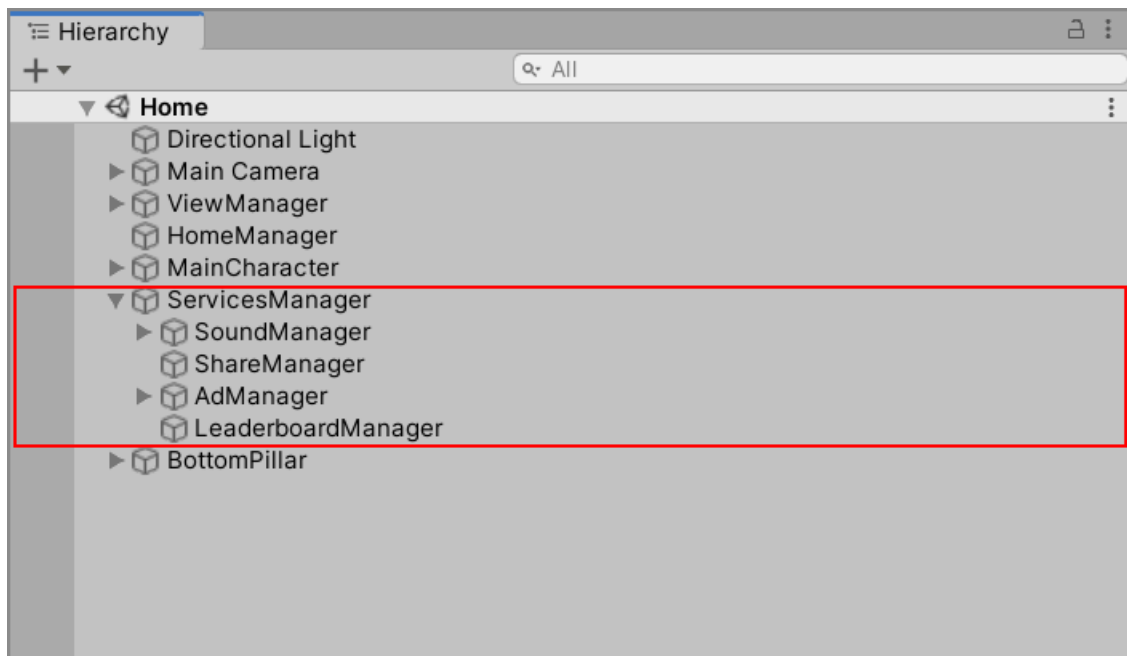
- ❖ *Min Stack Number & Max Stack Number:* the minimum and maximum stack will be created in this level. The actual value will be randomized between these two values.
- ❖ *Stack Type:* the type of stack you want to create. Currently, the template has 5 stack's type. STACK_6_PARTS, STACK_8_PARTS, STACK_10_PARTS, STACK_12_PARTS and STACK_16_PARTS. The number on the name represent the number of the parts the stack has. Example: STACK_6_PARTS has 6 parts on it.
- ❖ *First Stack Angle:* the first angle of first stack of the block.
- ❖ *Rotation Change Amount:* the change amount of the angle of each stack. Example: *First Stack Angle* is 10 and *Rotation Change Amount* is 5, so the first stack will be created with the angle 10, then the next stack will be created with the angle 15, then the next is angle 20, then

angle 25...Of course, you can set the parameter like -5, then the angle of the stacks will be 10, 5, 0, -5, -10...

- ❖ *List Index Of Deadly Part*: the index of deadly part on each stack, this parameter based on the parameter *Stack Type*. So, if you set the *Stack Type* to `STACK_6_PARTS`, then the stack will have 6 parts on it, so the index of each part will be 0, 1, 2, 3, 4, 5. **REMEMBER, THE INDEX WILL START AT 0 SO, YOU CAN'T HAVE THE INDEX 6.** As you can see, I set the index to 0 and 1, so the deadly will be the first part and the second part. If I set the index to 1, 3, 5 then the deadly part will be the second part (index 1), the 4th (index 3) and the 6th (index 5). You can set the index to 6 because the index count from 0 to 5.

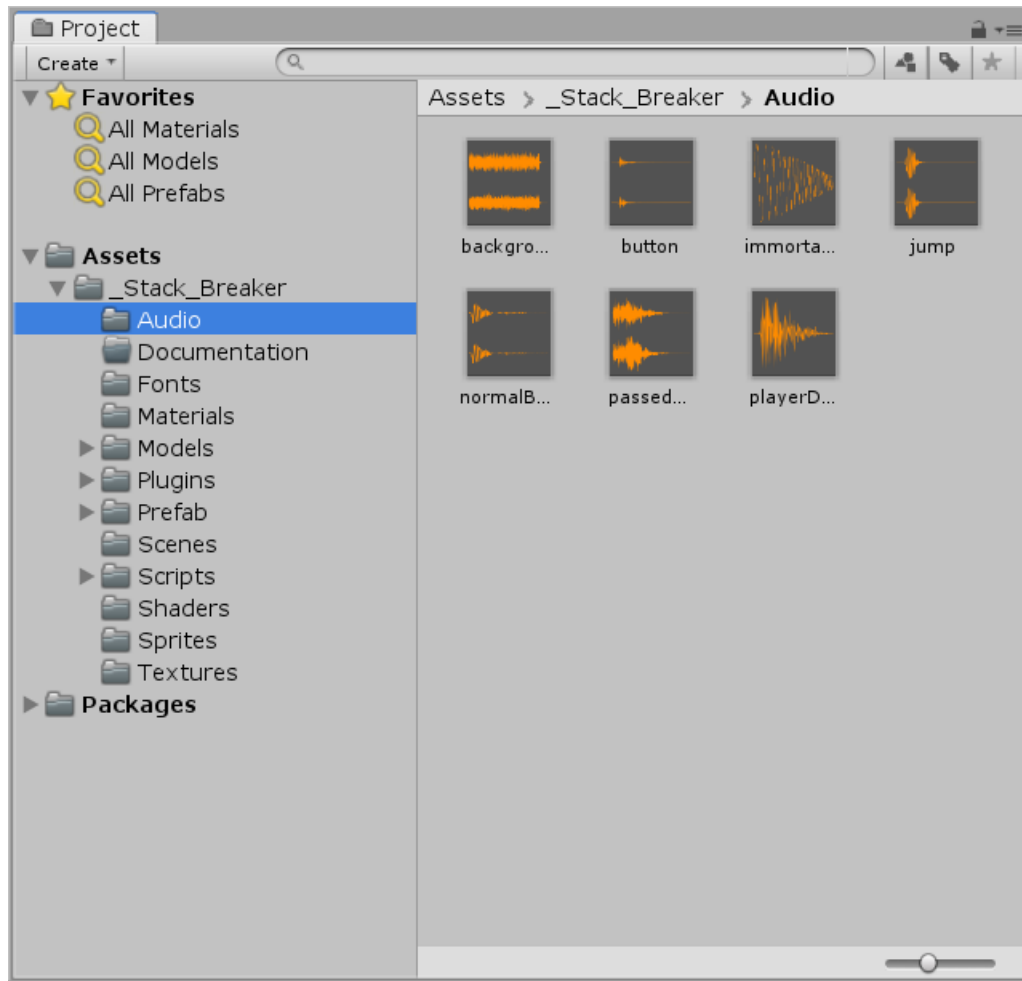
VI. SERVICES CONFIGURATION

This template comes with many features like sounds, sharing, multiple ads...ect. You can find all the scripts for these features under `ServicesManager` object in Home scene.



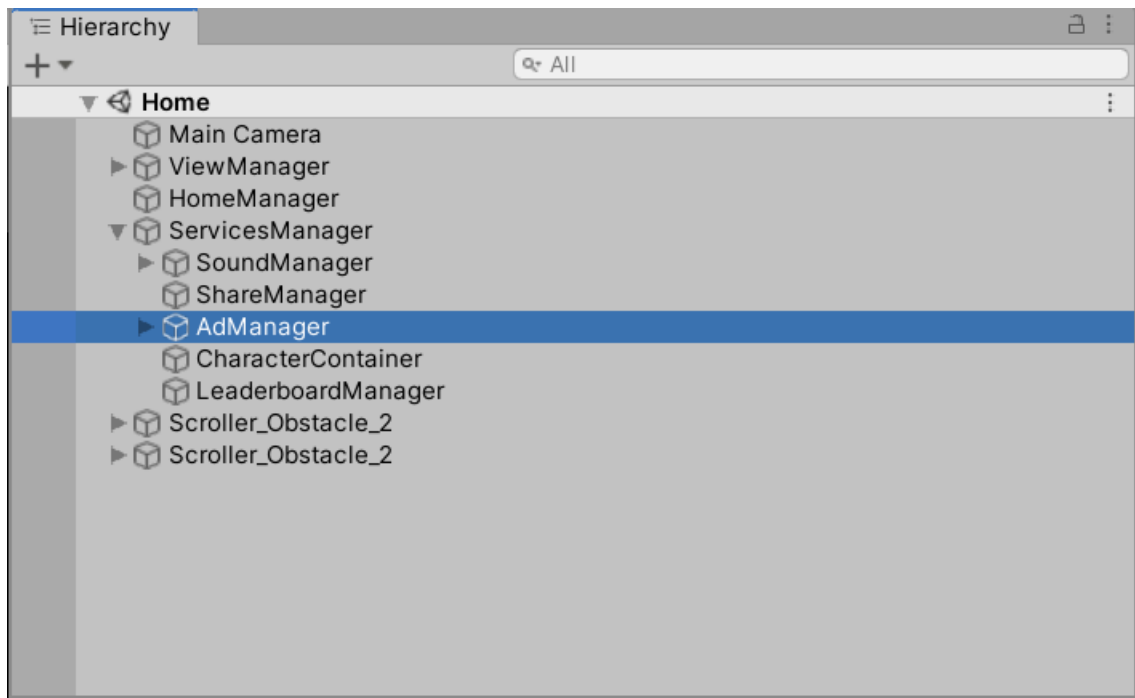
1. Sound Manager

All sounds included in this game are free-to-use in commercial projects and are located under the path `Assets/_Stack_Breaker/Audio` folder.



2. Ad Manager

The AdManager object in hierarchy of scene Home contains AdManager component, in which you can customize which type of ads you want to use and how you want to show the ads.



▼ **Ad Manager (Script)**

Script AdManager

Banner Ad config

Banner Ad Type ADMOB

Showing Banner Ad Delay 0.5

Interstitial Ad Config

▼ List Show Interstitial Ad Config

Size 2

▼ Element 0

Game State For Showing Ad Ingame_Game Over

Game State Count For Showing Ad 2

Show Ad Delay 1

▼ List Interstitial Ad Type

Size 2

Element 0 UNITY

Element 1 ADMOB

▼ Element 1

Game State For Showing Ad Ingame_Pause

Game State Count For Showing Ad 1

Show Ad Delay 0.5

▼ List Interstitial Ad Type

Size 2

Element 0 UNITY

Element 1 ADMOB

Rewarded Video Ad Config

Showing Rewarded Video Ad Delay 0.2

▼ List Rewarded Ad Type

Size 1

Element 0 UNITY

Rewarded Coins Config

Min Rewarded Coins 100

Max Rewarded Coins 150

Reward Delay 0.2

- ❖ **Banner Ad Config:** this is the section where you can control which type of banner ad you want to show. Currently, the template support for 2 banner ad type: Admob ad Unity.
 - **Banner Ad Type:** the type of banner ad you want to show. (Unity Ads just released Unity Monetization 3.0 which included Banner Ad, but seems like its still unstable, so I recommend using Admob for banner ad).
 - **Showing Banner Ad Delay:** delay time to show banner ad.

- ❖ **Interstitial Ad Config:** this is the section where you can control which type of interstitial ad you want to show and how you want to

how it. Currently, the template support for 2 interstitial ad type: Admob and Unity Ads.

List Show Interstitial Ad Config: this is the list contains all the config parameters of showing interstitial ad.

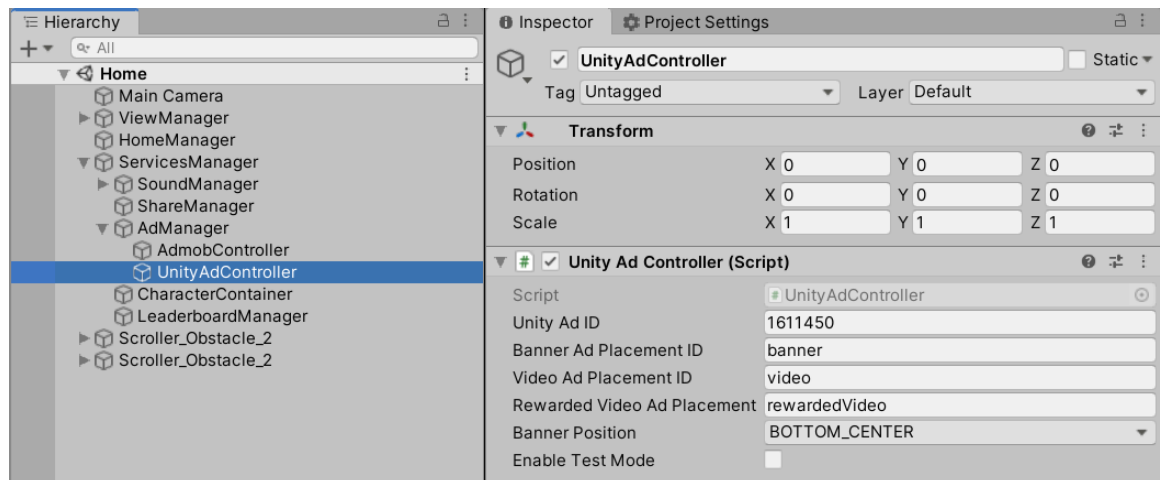
- *Game State For Showing Ad*: the game state you want to show the ad.
- *Game State Count For Showing Ad*: the number of game state that the game go through to show ad. Example: if the value is 2 and *Game State For Show Ad* is Game Over, that mean the ad will show up after 2 times of game over.
- *Show Ad Delay*: the delay for showing the ad.
- *List Interstitial Ad Type*: the list of interstitial ad type. The first item of this list will be the first priority of interstitial ad. Example: in the image you can see 2 items: ADMOB and UNITY. ADMOB is the first item so, *Ad Manager* will load rewarded video ad of Admob first, if there's no Admob's rewarded video to load, then *Ad Manger* will continue to load Unity interstitial ad.

❖ *Rewarded Video Ad Config*: this is the section where you can control which type of rewarded video ad you want to show and how you want to how it. Currently, the template support for 2 rewarded video ad type: Admob and Unity Ads.

- *Showing Rewarded Video Ad Delay*: the delay time for showing rewarded video ad.
- *List Rewarded Ad Type*: the list of rewarded video ad type. The first item of this list will be the first priority of interstitial ad. Example: in the image you can see 2 items: ADMOB and UNITY. ADMOB is the first item so, *Ad Manager* will load rewarded video ad of Admob first, if there's no Admob's rewarded video to load, then *Ad Manger* will continue to load Unity rewarded video ad.

3. Unity Ads Controller

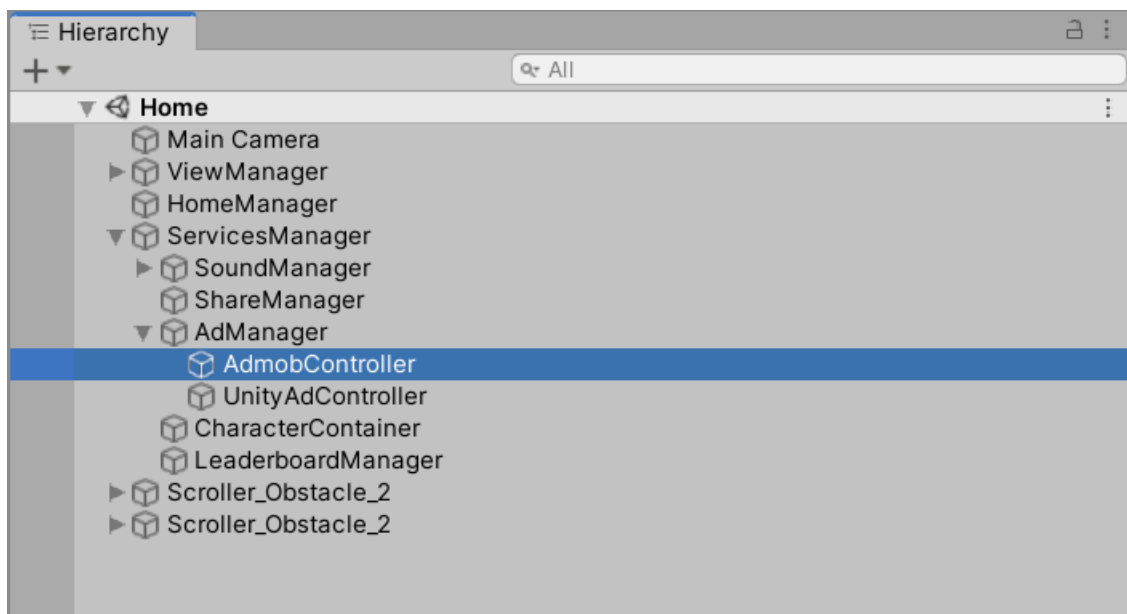
The template already had unity ads sdk included, so you don't need to import anything else to use unity ads, just put your project id and your placement ids and you will good to go.

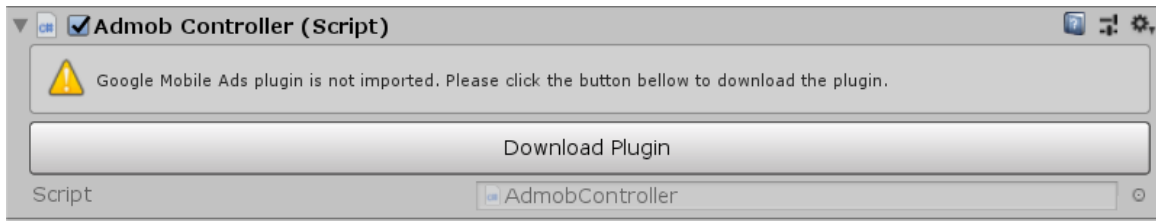


- ❖ *Unity Ad ID*: the id of your unity ad project. You can find all of your ad projects [here](#).
- ❖ *Video Ad Placement ID*: the video ad placement id of your ad project.
- ❖ *Rewarded Video Ad Placement ID*: the rewarded video ad placement id of your ad project.

4. Admob Controller

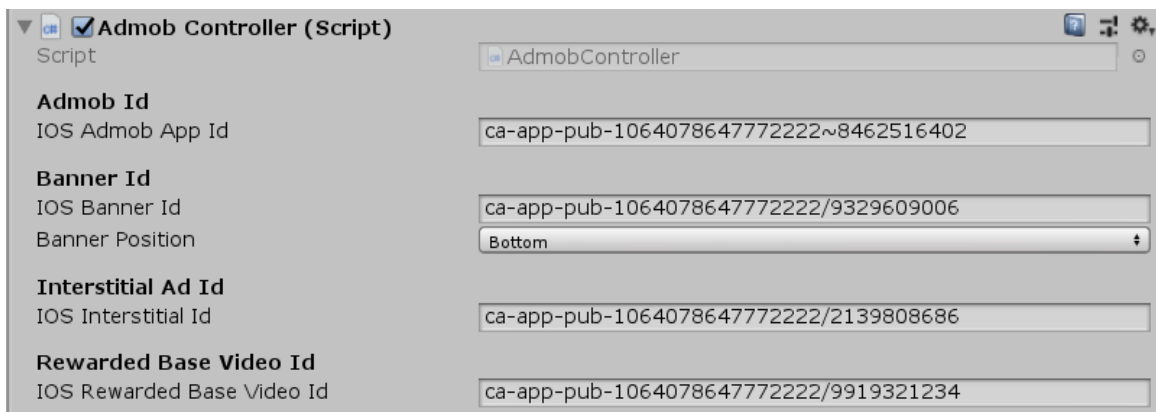
The AdmobManager object in hierarchy of scene Home contains AdmobController component, in which you can customize parameters like admob id, ad units...





As you can see in the image, there's no option to change ads units because the template did not have google mobile ads plugin included, so if you want to use Admob, please click to the Download Plugin button, download the latest version of google mobile ads and import it into the project.

After imported the plugin, select AdManager object in the hierarchy, wait for few seconds for the project rebuild and you will see this.

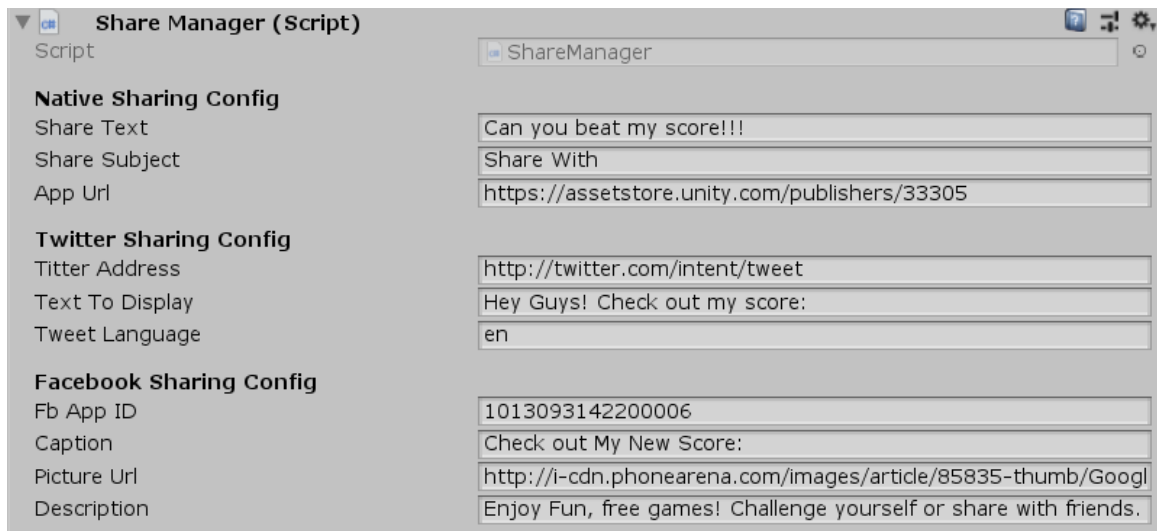
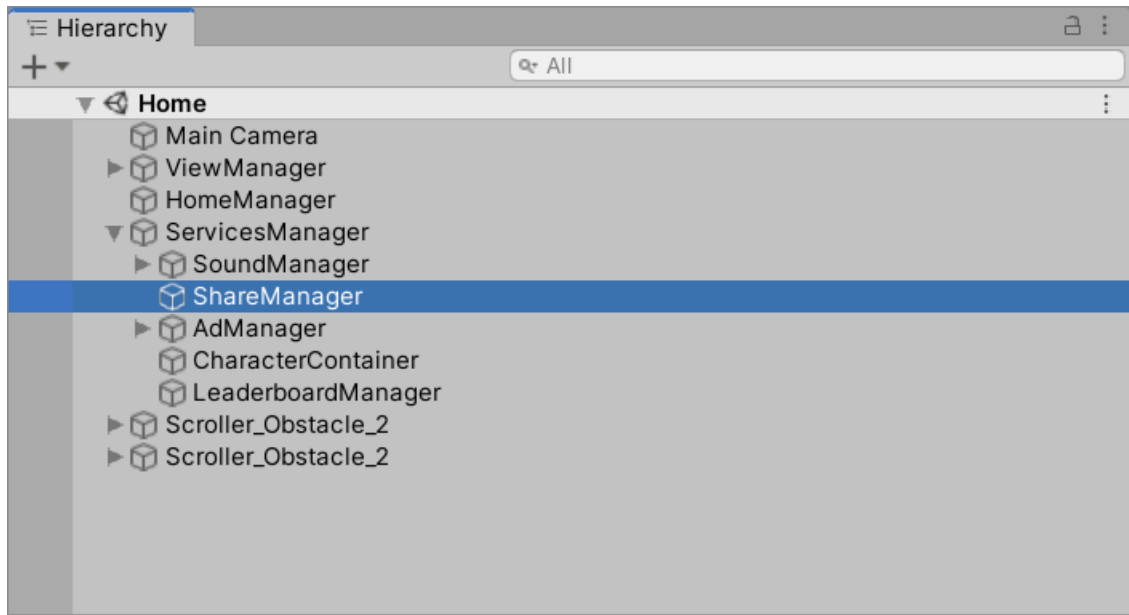


Now you can config admob id and all the ad units as you want. Currently, the platform using is iOS, that why all the ids in the image showing for iOS, of course it will show the ids for Android when you switch the build platform to Android.

- ❖ *IOS Admob App Id*: the android id of your admob app.
- ❖ *IOS Banner Id*: the banner ad unit of your admob app.
- ❖ *Banner Position*: the position of the banner ad.
- ❖ *IOS Interstitial Id*: the interstitial ad unit of your admob app.
- ❖ *IOS Rewarded Base Video Id*: the rewarded ad unit of your admob account.

5. Share Manager

All information for sharing feature can be config in ShareManager game object. It contains the information likes shreenshot's name, text, subject and url... You can config these features from the ShareManager object in the hierarchy.



Native Sharing Config:

- ❖ *Screenshot Name*: the name of screenshot for sharing feature.
- ❖ *Share Text*: the text for sharing feature.
- ❖ *Share Subject*: the subject for sharing feature.
- ❖ *App Url*: the url of the app (Google Play on Android and App Store on IOS).

Twitter Sharing Config:

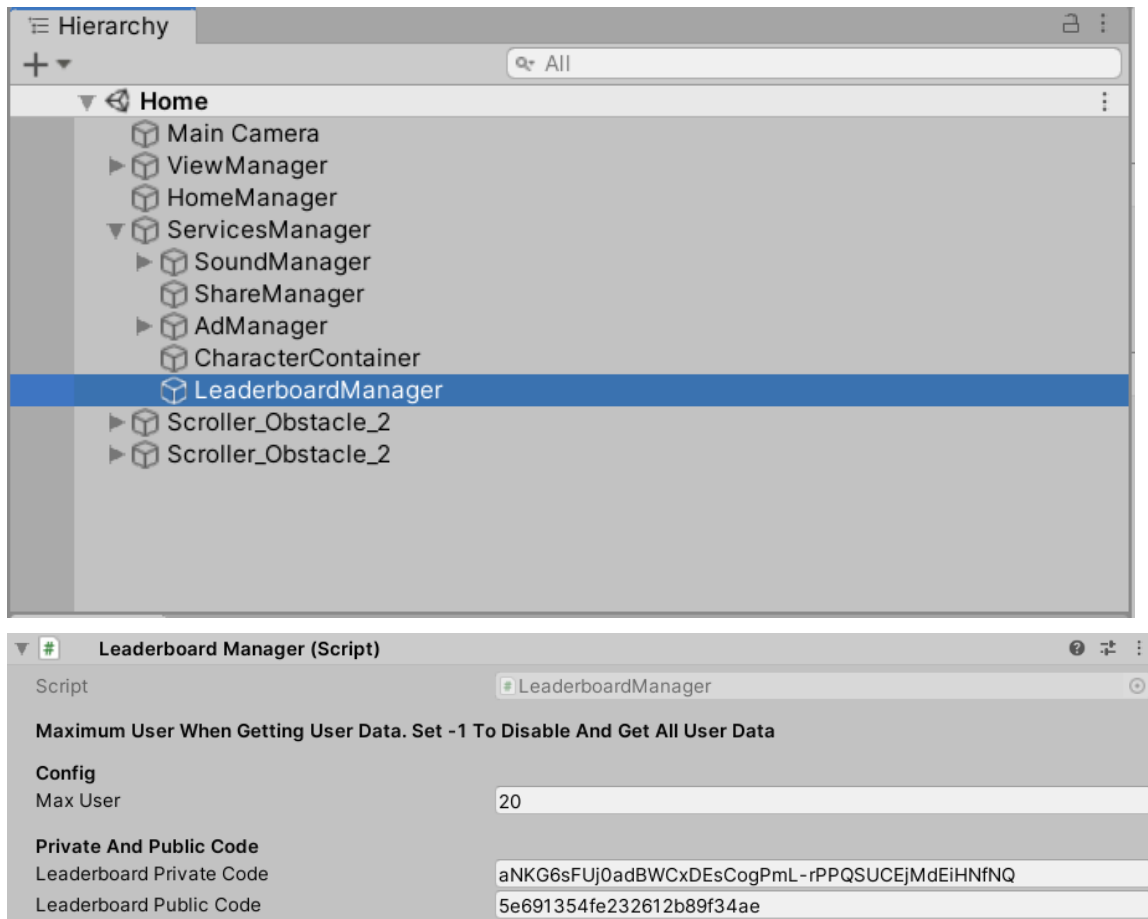
- ❖ *Twitter Address*: address of your twitter account.
- ❖ *Text To Display*: the text you want to display in the tweet.
- ❖ *Tweet Language*: language of the tweet you want to share.

Facebook Sharing Config:

- ❖ *Fb App ID*: the id of your facebook app.
- ❖ *Caption*: the caption in your status.
- ❖ *Picture Url*: url of the picture you want to share. If you don't want to share pictures or just don't have any picture's url to share, leave this field empty.
- ❖ *Description*: the description you want to share.

6. Leaderboard Manager

All information for leaderboard feature can be config in LeaderboardManager game object. It contains 2 parameters for you to config the leaderboard. Currently, we used Dreamlo to handle leaderboard feature.

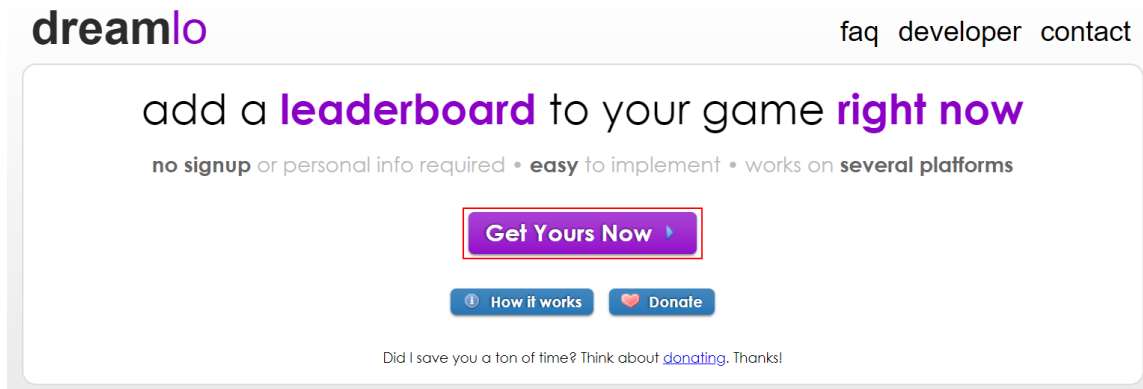


- ❖ *Max User*: the maximum user show in the leaderboard. If you set it to -1, it will show all the user that played the game and report to leaderboard table. We suggest you to limit it to 50 or 100, because if your game have 1000 downloads, which mean you have 1000 user in the leaderboard and when you load it all to the leaderboard view, the game will become laggy.
- ❖ *Leaderboard Private Code*: the leaderboard private code of

- Dreamlo.
- ❖ *Leaderboard Public Code*: the leaderboard public code of Dreamlo.

Now we will show you how to get the private and public code from Dreamlo.

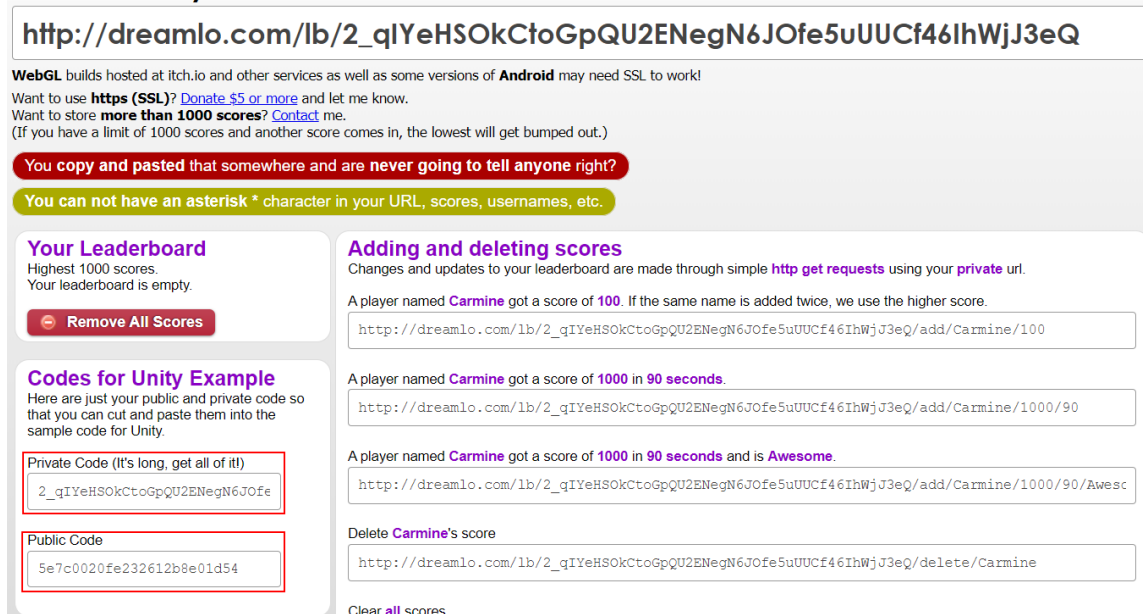
First, you need to go to Dreamlo.com, at the top of the front page, click to Get Yours Now button.



After that, you will be direct to a page that you can get your private and public code.



Here is your **private** url. Copy and paste this somewhere.
Do not tell anyone about this link.

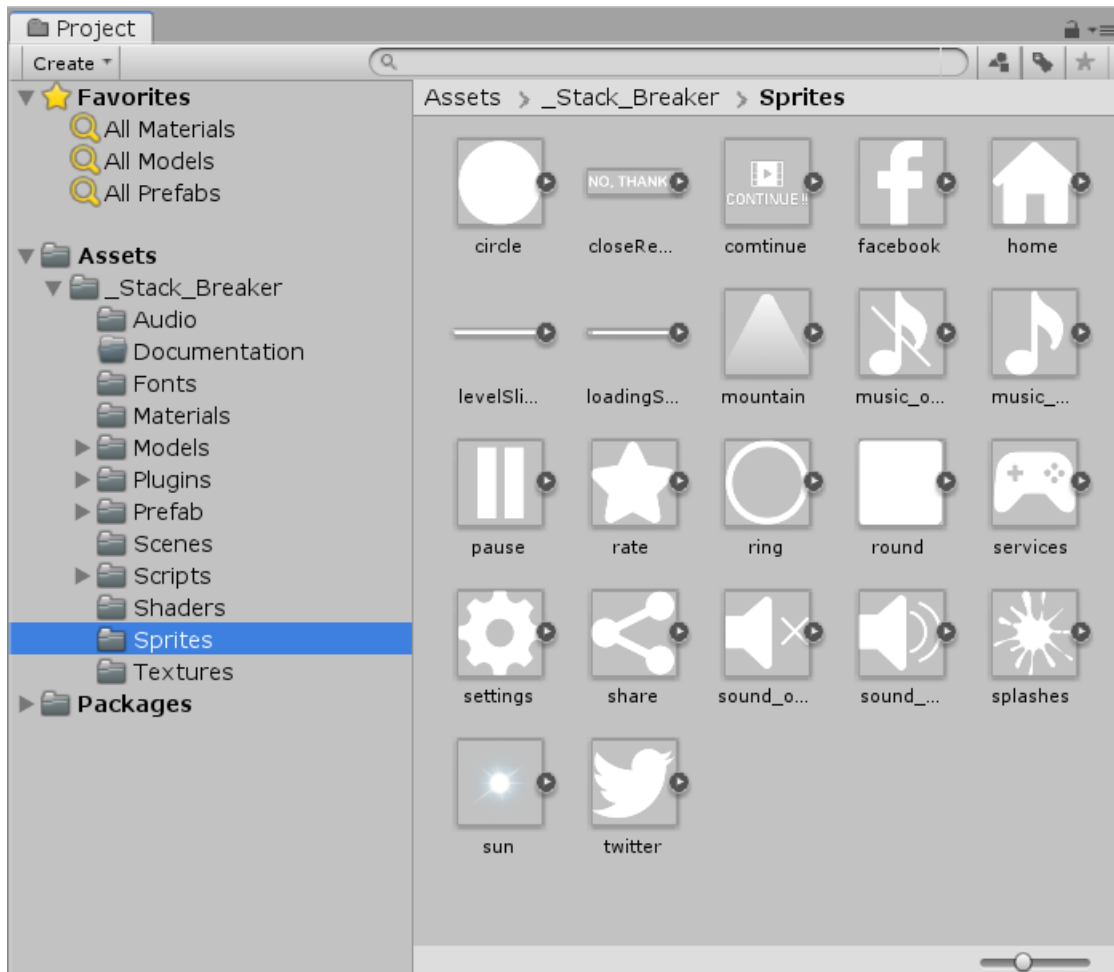


Copy 2 codes and paste it on LeaderboardManager component.

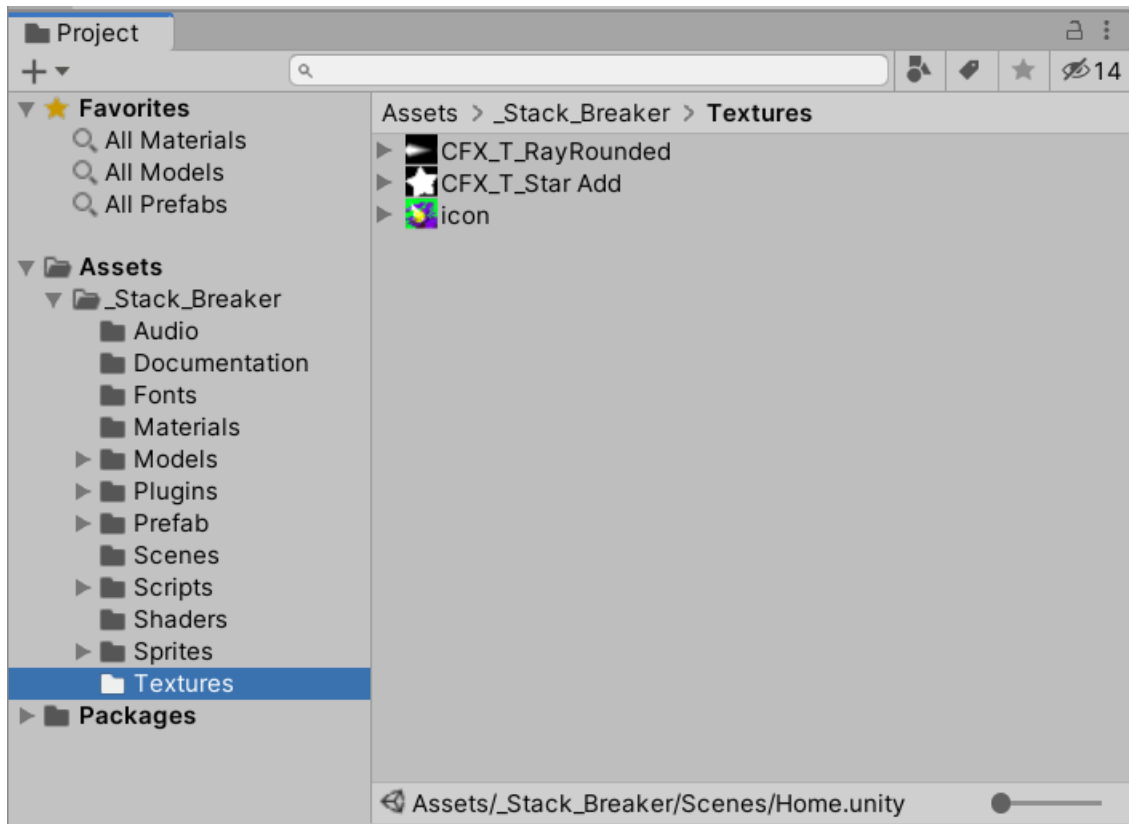
That's it.

7. Customizing UI

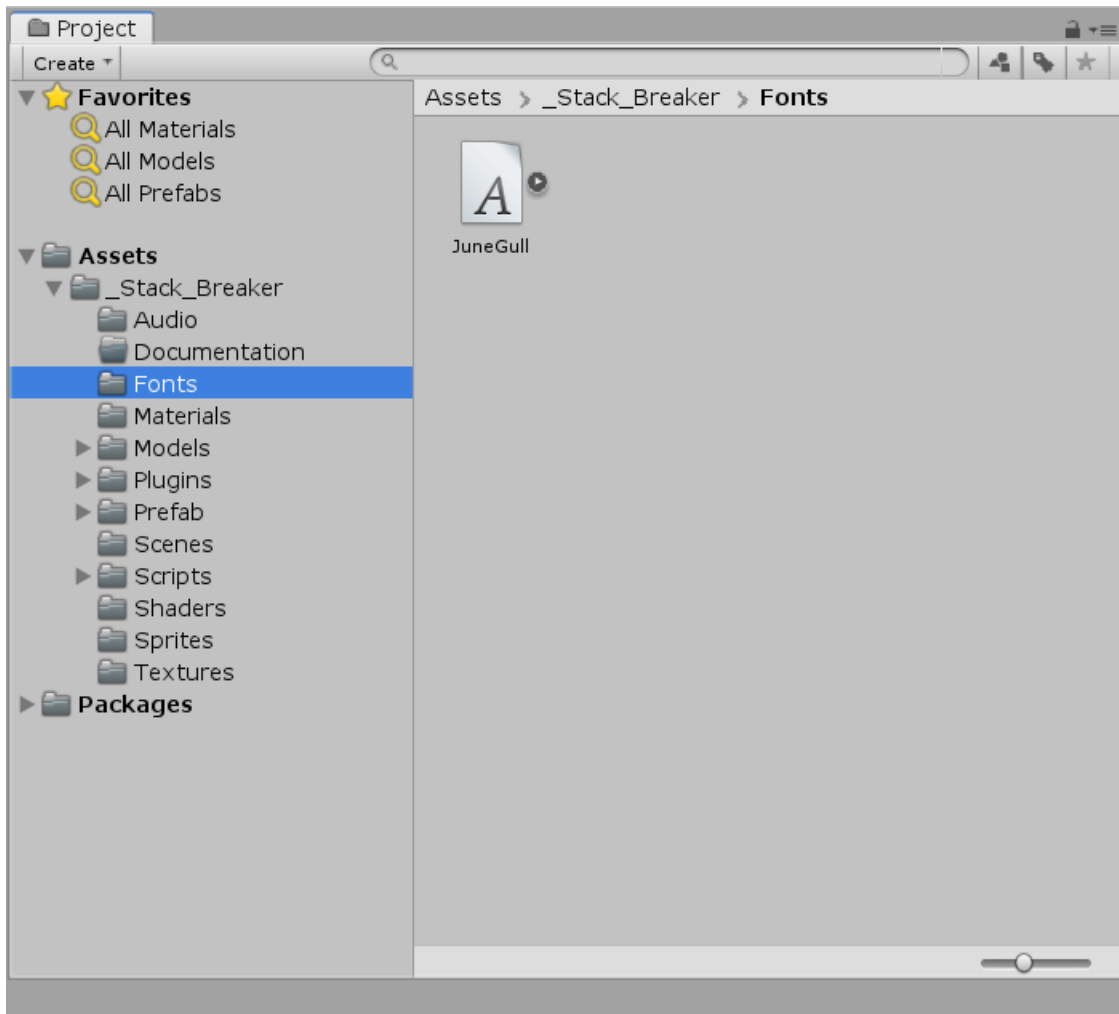
All sprites used in this game (for buttons and other UI components) are located under the path *Assets/_Stack_Breaker/Sprites* folder. You can replace them with your own sprites to modify the UI as you like.



All textures used in this game (for buttons and other UI components) are located under the path *Assets/_Stack_Breaker/Textures* folder. You can replace them with your own sprites to modify the images as you like.



All fonts used in this game are free-to-use in commercial projects. Fonts are located under the path *Assets/_Stack_Breaker/Fonts* folder together with appropriate license files.



THANK YOU AND GOOD LUCK WITH YOUR GAMES!