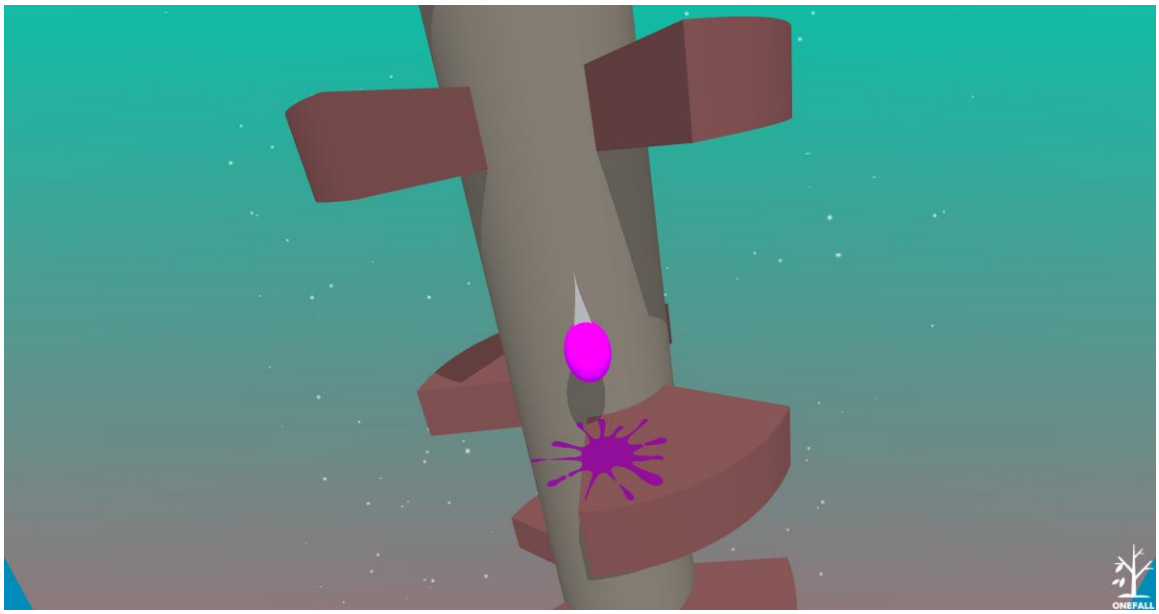
USER GUIDE

JUMPY HELIX TEMPLATE

Onefall Games

Table of Contents

I. INTRODUCTION



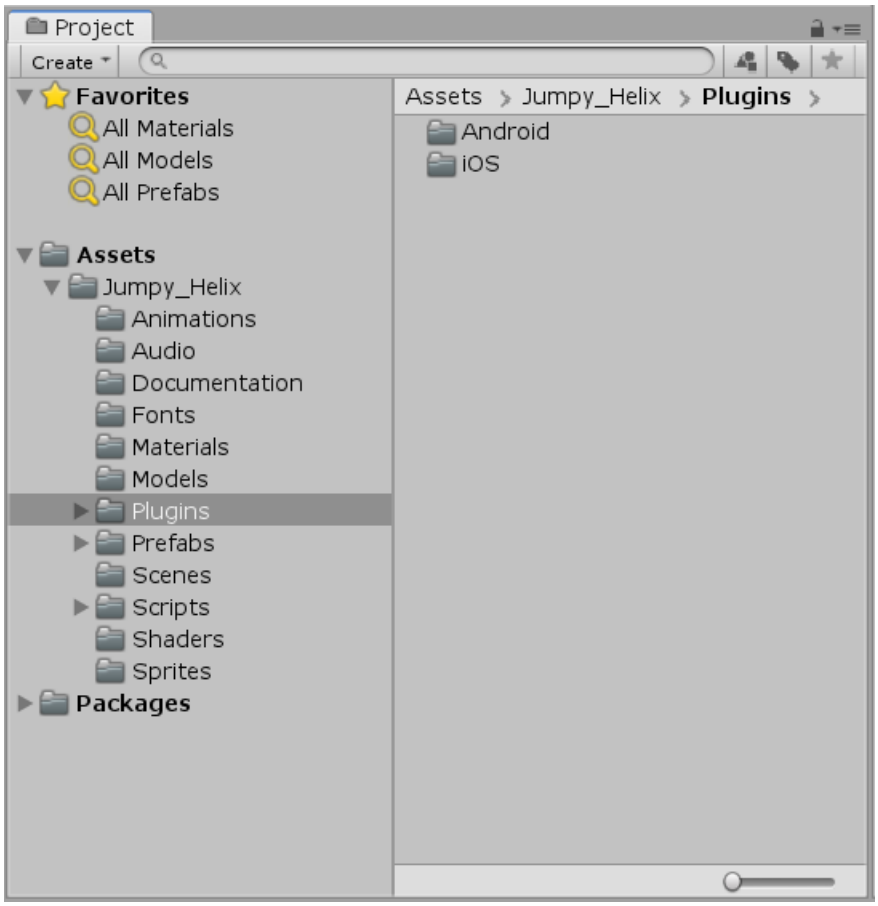
Jumpy Helix is an exciting endless arcade game in which you touch on left screen to spin the pillar left, touch on right screen and hold to spin the pillar right. Try to fall through the cracks and avoid the dead pieces. Be careful of the time, you need to go all the pieces as quick as possible. The 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! Some features:

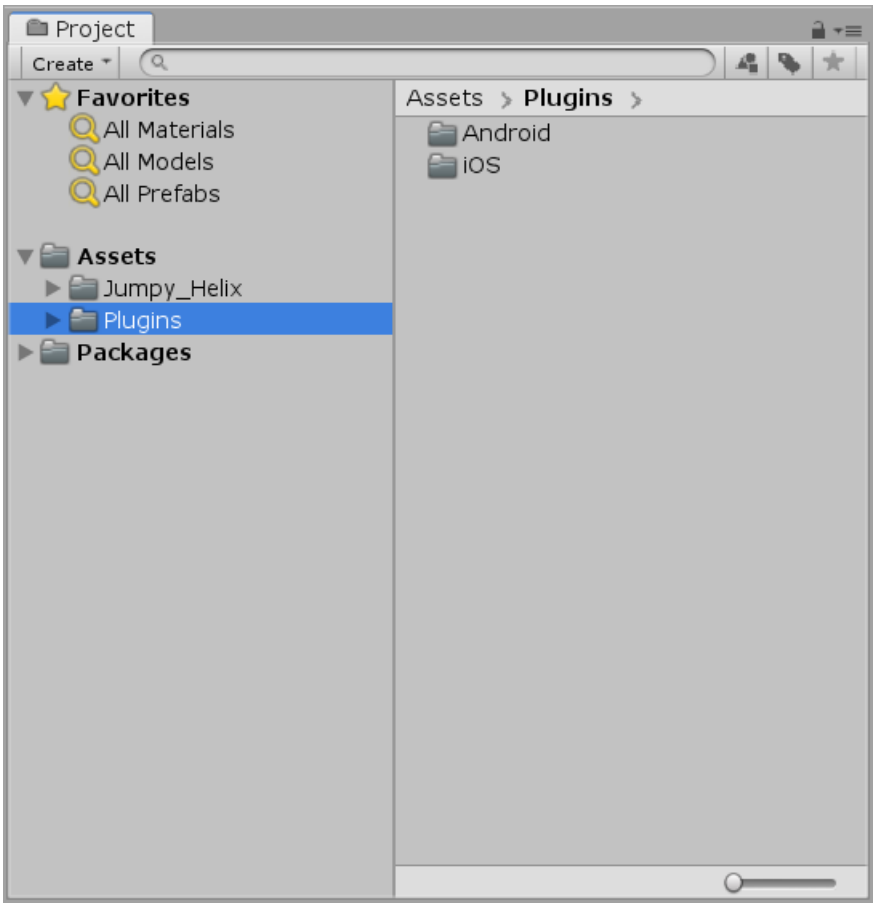
- Addictive one-touch gameplay
- Smooth control, easy to modify
- Eye-catching effects
- Endless level, easy to modify each level
- Break pieces when you go through more than 2 helix pieces
- Different color for each type of piece
- Native share for Android/IOS
- Facebook/Twitter share
- Multiple ad networks: Admob and Unity Ads
- Free-to-use assets (fonts, sounds, music, model, etc.)
- Optimized for mobile
- Ready to publish out-of-the-box

II. TEMPLATE SETUP

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

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





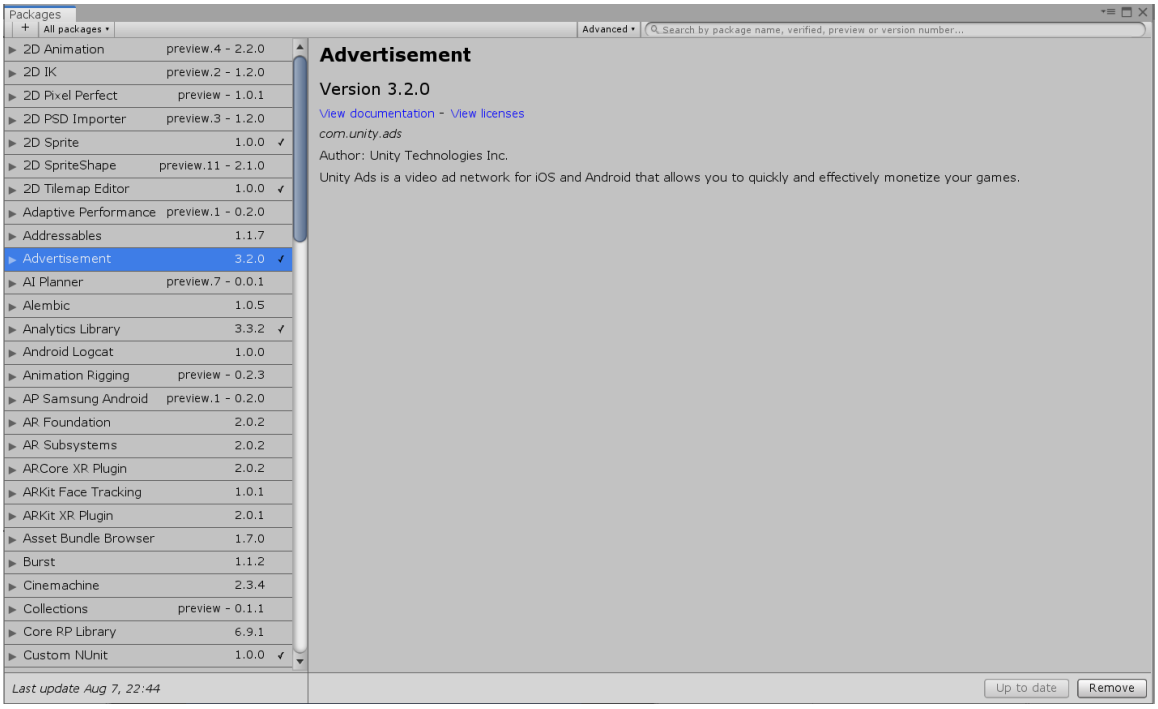
The template contains only one scene. You need to start from scene Gameplay.

III. REQUIREMENT PACKAGES

When you open this template, at first you will some errors in Console window, that's

because this template requires some packages to run. You have to install these packages bellow to have this template run smoothly. Please follow these instructions:

- Open Package Manager by go 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 3.2.0 or higher.
  - Analytics Library version 3.2.2 or higher.
  - Text Mesh Pro version 1.3.0 or higher.

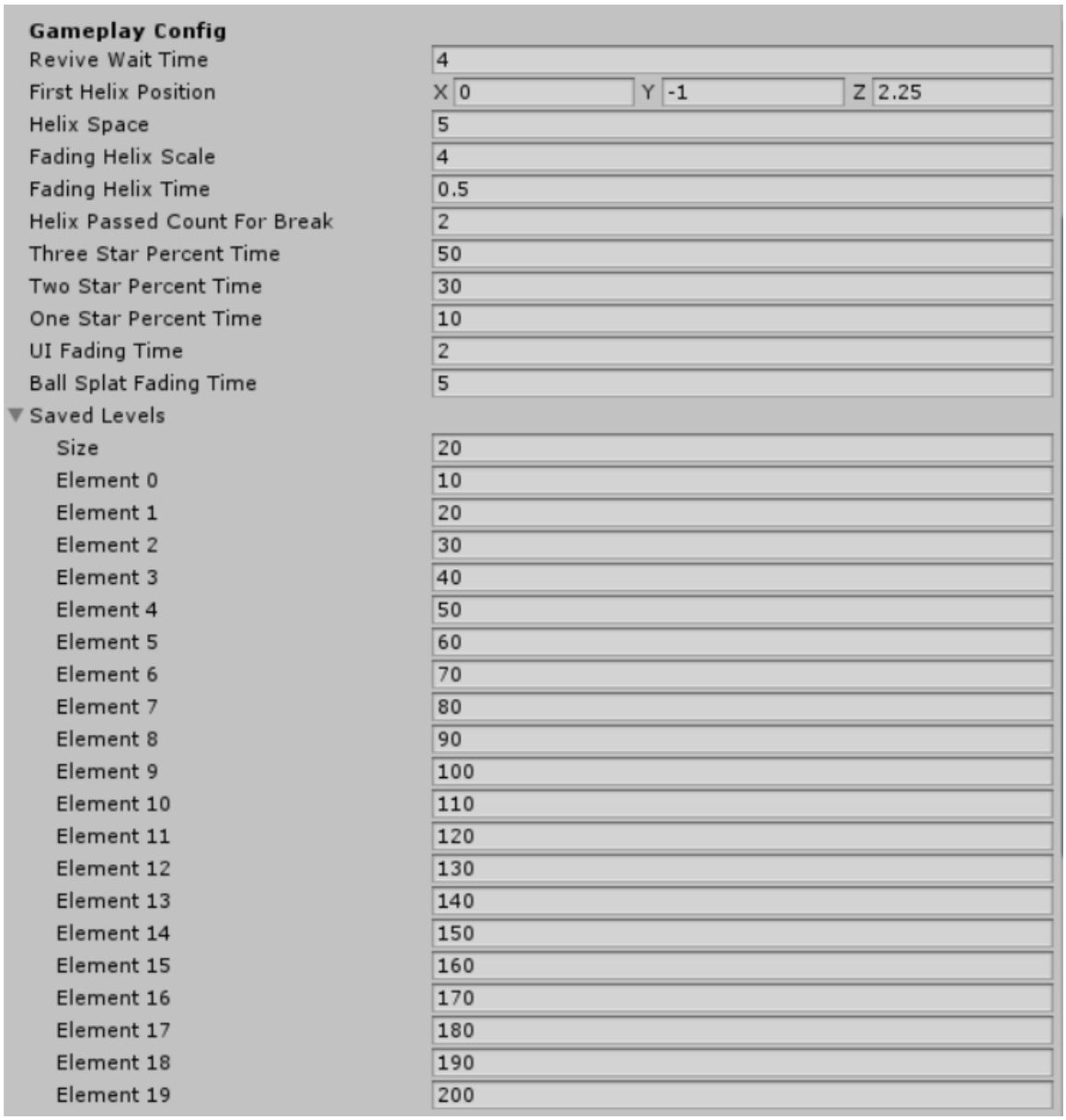
After you installed these packages, there's maybe some errors in Console window. If you see errors in Console window, go to File - > Build Setting -> Player Setting and change the Scripting Run Time Version to .Net 4.x Equivalent.

IV. TEMPLATE CUSTOMIZATION

1. Gameplay Tweaking

1.1. Game Manager

Most of important gameplay parameters can be configured within the GameManager component which is attached to a game object also named GameManager in the hierarchy.

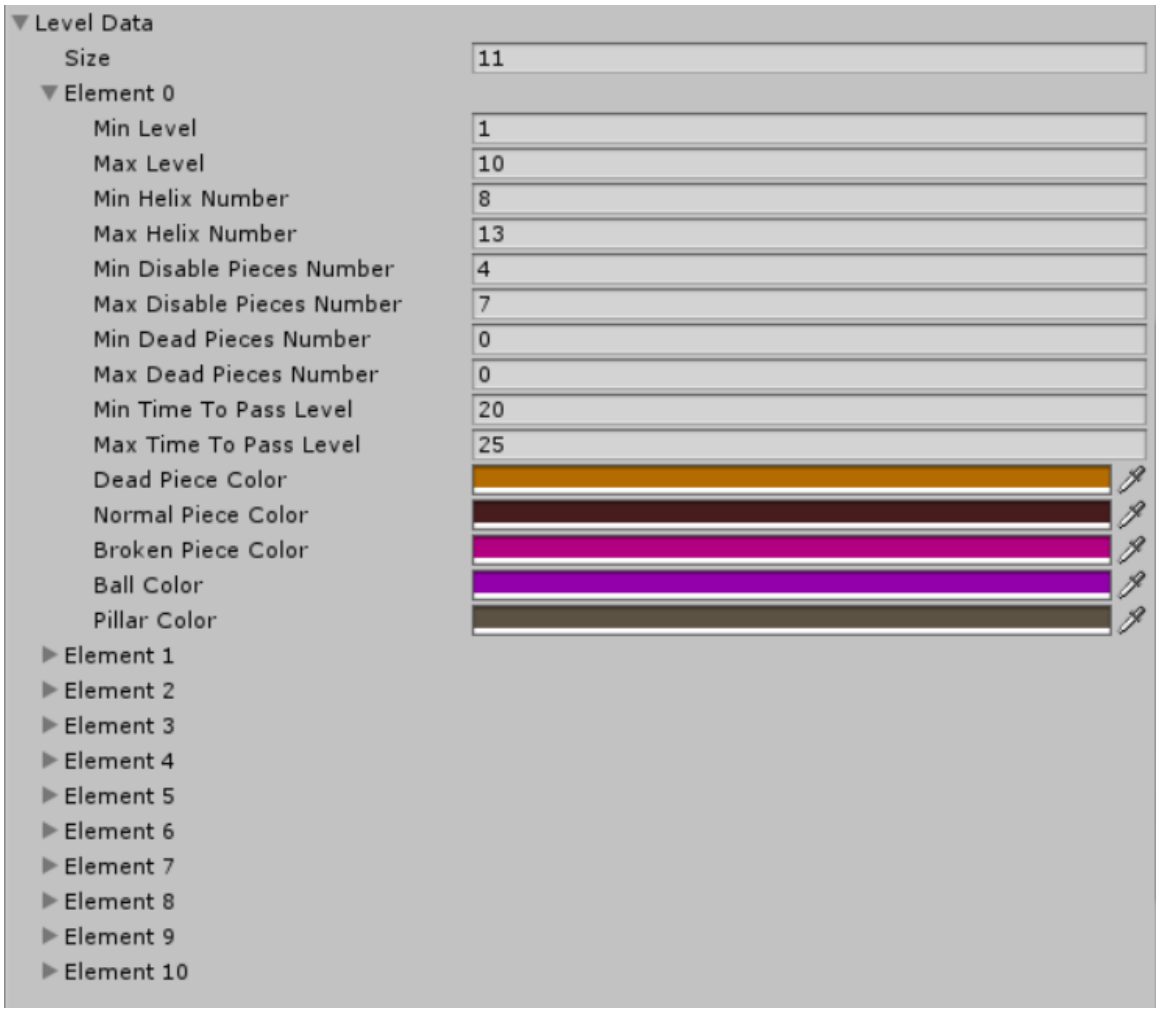


You can tweak the gameplay by modifying following variables:

- *Revive Wait Time*: delay time for Revive feature till it ends.
- *First Helix Position:* the position of the first helix when the game started.
- *Helix Space:* the space between each helix.
- *Fading Helix Scale:* the scale factor of fading helix.
- *Fading Helix Time:* the time for fading helix.
- *Helix Passed Count For Break:* the passed count for break helix. For example: the value is 2, that mean every time you go through equals or more

than 2 helix, then you hit a helix, it will shatter to pieces.

- *Three Star Percent Time:* the percent time left you need to reach to get three stars. Example: the percent is 50, the time for the level is 10 seconds. Then you need to win the level within 5 seconds to get three stars.
- *Two Star Percent Time:* the percent time left you need to reach to get two stars. Example: the percent is 30, the time for the level is 10 seconds. Then you need to win the level within 7 seconds to get two stars.
- *One Star Percent Time:* the percent time left you need to reach to get one stars. Example: the percent is 10, the time for the level is 10 seconds. Then you need to win the level within 9 seconds to get one stars.
- *UI Fading Time:* time for fading out UI.
- *Ball Splat Fading Time:* time for fading ball's splat.
- *Saved Levels:* the array of levels that will be saved like checkpoints.
- *Level Data:* this struct allow you config how the level will be generated.

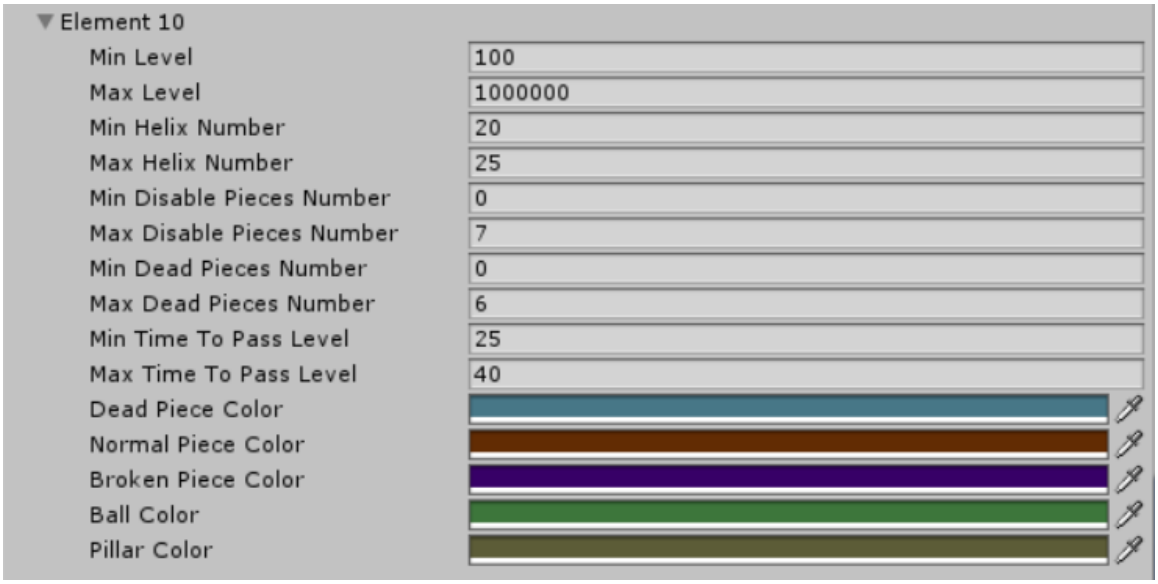


- *Min Level* & *Max Level:* the minimum and maximum level. Each level between these two values will apply these following config parameters.
- *Min Helix Number* & *Max Helix Number:* the minimum and maximum helix object will be generated when the game stared. The actual value will be randomized between these two values.
- *Min Disable Piece Number* & *Max Disable Piece Number:* minimum and maximum disable piece of one helix. The actual value will be randomized between these two values. **REMEMBER: EACH HELIX HAS 12 PIECES. DO NOT ADJUST THESE TWO VALUES TOO HIGH, IT WILL CAUSE ERRORS.**
- *Min Dead Piece Number* & *Max Dead Piece Number:* minimum and maximum dead piece of one helix. The actual value will be randomized between these two values. **REMEMBER: EACH HELIX HAS 12 PIECES. DO NOT ADJUST THESE TWO**

VALUES TOO HIGH, IT WILL MAKE THE HELIX IMPOSSIBLE TO GO THROUGH.

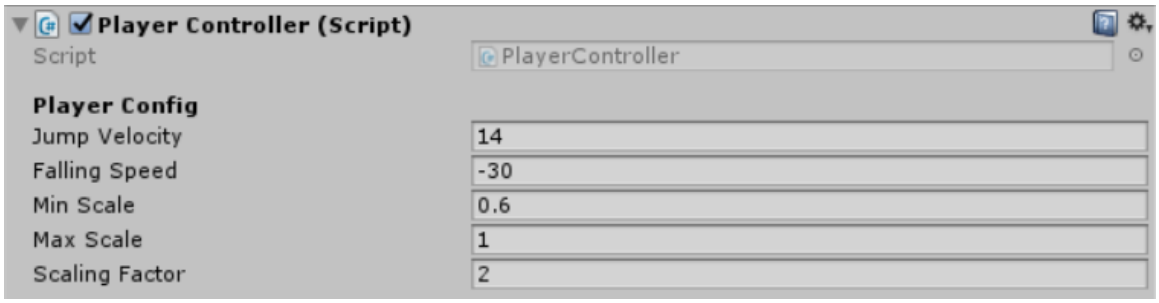
- *Min Time To Pass Level* & *Max Time To Pass Level:* minimum and maximum time to win the level. The actual value will be randomized between these two values.
- *Dead Piece Color:* the color of dead piece (the piece that if you hit them, you will die).
- *Normal Piece Color:* the color of normal piece.
- *Broken Piece Color:* the color of broken piece.
- *Ball Color:* color of the ball (main player).
- *Pillar Color:* color of the pillar.

Currently, *Level Data* has 10 elements, you can add more config by resize the array and edit the new element. You can make endless level by edit *Max Level* something like 1000000. See the image bellow as example.



1.2. Player Controller

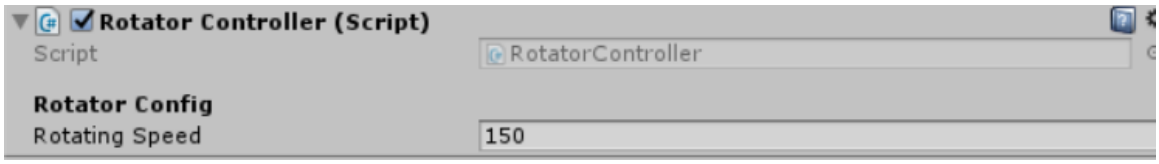
The Player object in the hierarchy contains a PlayerController component, in which you can customize the player (main character) behavior.



- *Jump Velocity*: the jump velocity when the game started.
- *Falling Speed*: falling speed of the ball (main player).
- *Min Scale*: the minimum scale when the ball falling down.
- Max Scale: the maximum scale when the ball bouncing up.
- *Scaling Factor:* how fast the ball will scale.

1.3. Rotator Controller

The Rotator object in the hierarchy contains a RotatorController component, in which you can customize the rotator behavior.

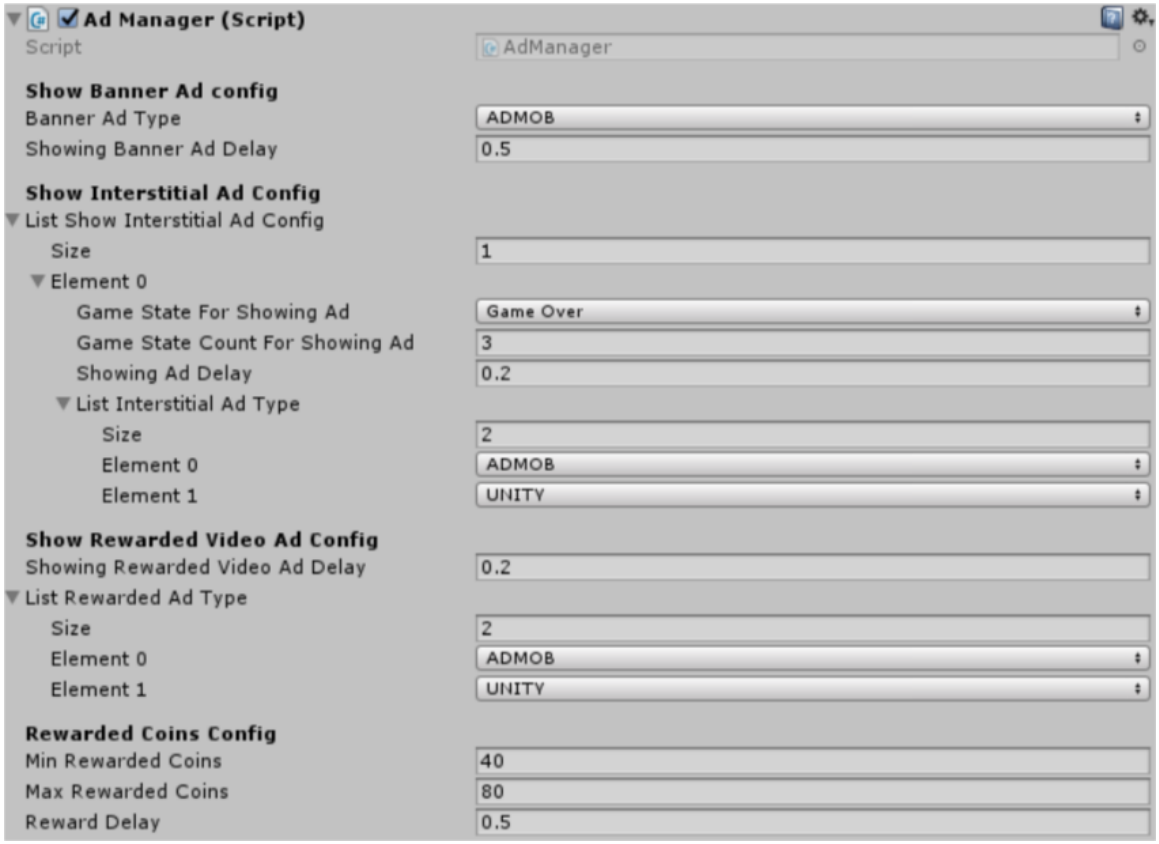


- *Rotating Config*: rotating speed of the rotator.

2. Advertisement

2.1. Ad Manager

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



Show 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 and Unity Ads.

- *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 it still unstable, so I recommend using Admob for banner ad).
- *Showing Banner Ad Delay:* delay time to show banner ad.

Show 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.

Show 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.

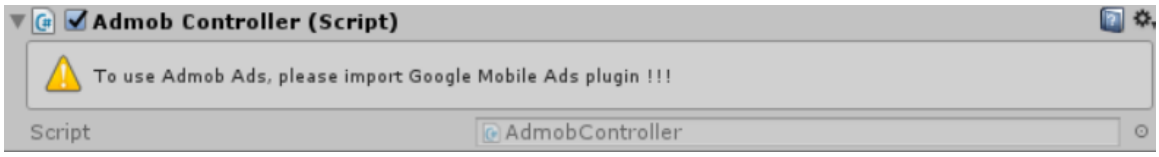
- *Showing Rewarded Video Ad Delay:* the delay time for showing rewarded video ad.
- *List Rewarded Ad Type:* the list of rewarded video ad type. The first item of this list will be the first priority of interstitial ad. Example: in the image you can see 2 items: ADMOB and UNITY. ADMOB is the first item 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.

Rewarded Coins Config: this is the section where you can config how many coins will be rewarded after user watch the rewarded ad.

- *Min Rewarded Coins* & *Max Rewarded Coins:* minimum and maximum of rewarded coins, the actual coins will be randomized between these two values.
- *Reward Delay:* the delay time to show rewarded video.

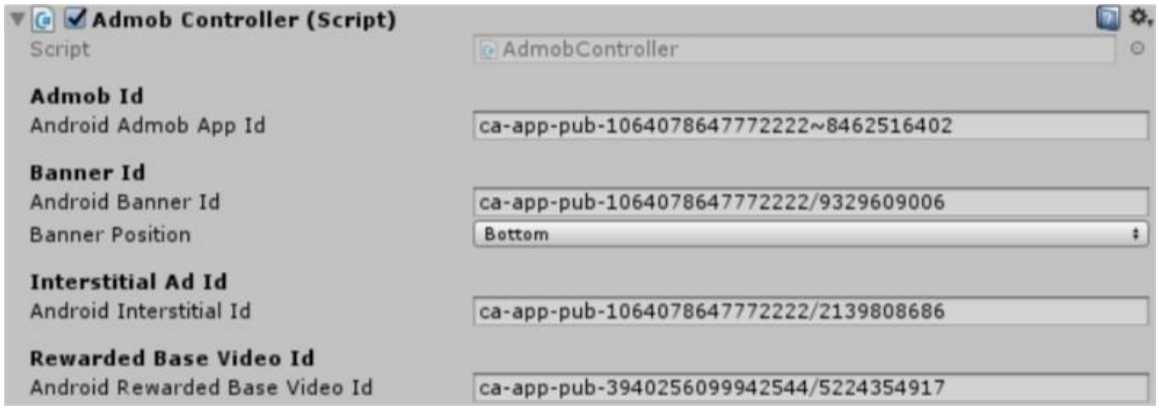
2.2. Admob Controller

The AdManager object in hierarchy of scene Gameplay 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 go to this link, 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.

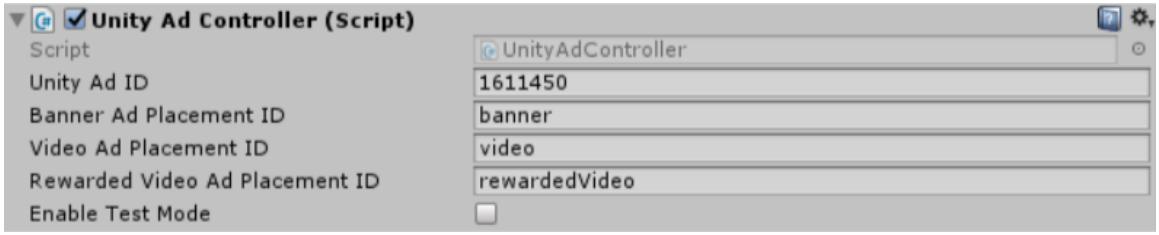


Now you can config admob id and all the ad units as you want. Currently, the platform using is Android, that why all the ids in the image showing for Andoid, of course it will show the ids for iOS when you switch the build platform to iOS.

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

2.3. Unity Ad Controller

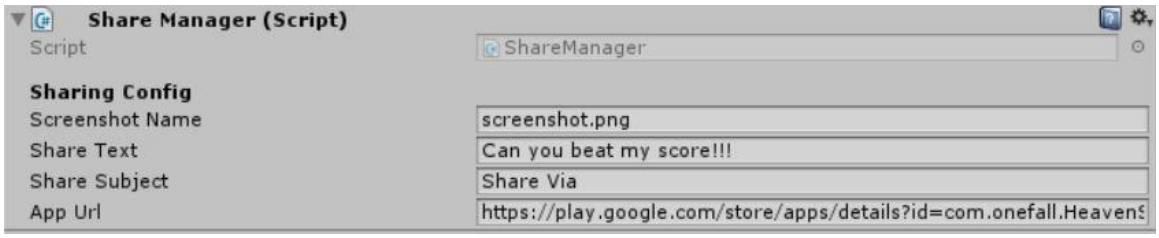
The template already had unity ads sdk included, so you don't need to import any thing else to use unity ads, just put your project id and your placement ids and you will good to go.



- *Unity Ad ID:* the id of your unity ad project. You can find all of your ad projects here.
- *Banner Ad Placement ID:* the banner ad placement id of your ad project.
- *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.

3. Share Manager

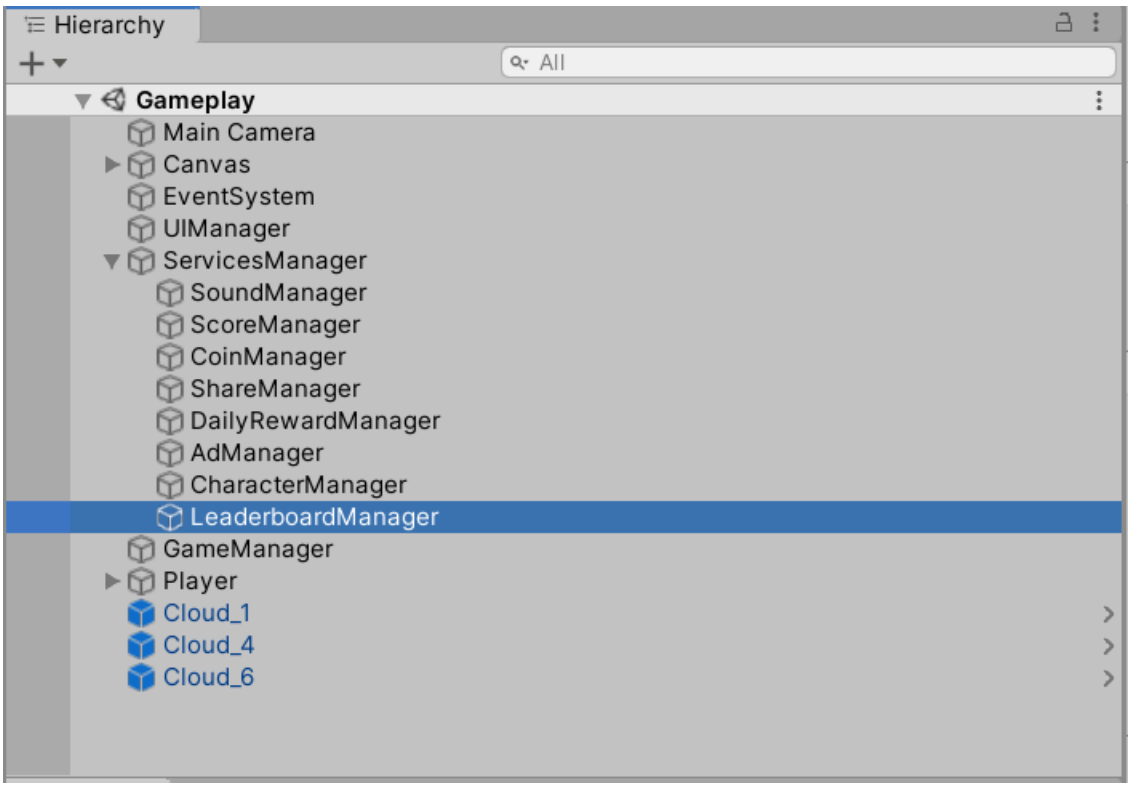
All information for sharing feature can be config in ShareManager game object. It contains the information likes shreenshot's name, text, subject and url…You can config these features from the ShareManager object in the hierarchy.

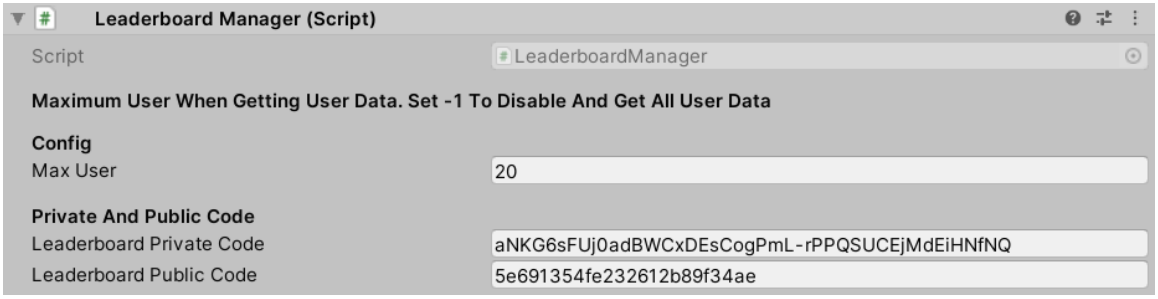


- *Screenshot Name:* the name of screenshot for sharing feature.
- *Share Text:* the text for sharing feature.
- *Share Subject:* the subject for sharing feature.
- *App Url:* the url of the app (Google Play on Android and App Store on IOS).

4. Leaderboard Feature

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.

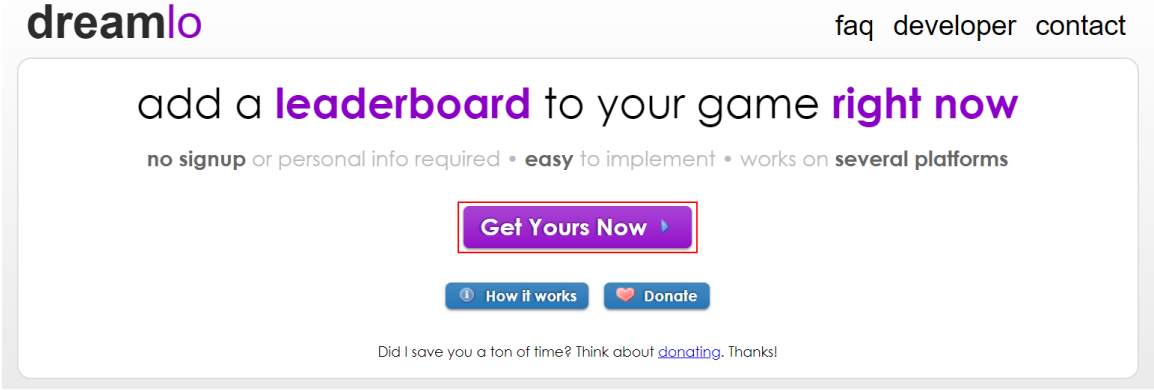




- *Max User:* the maximum user show in the leaderboard. If you set it to -1, it will show all the user that played the game and report to leaderboard table. We suggest you to limit it to 50 or 100, because if your game have 1000 downloads, which mean you have 1000 user in the leaderboard and when you 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.



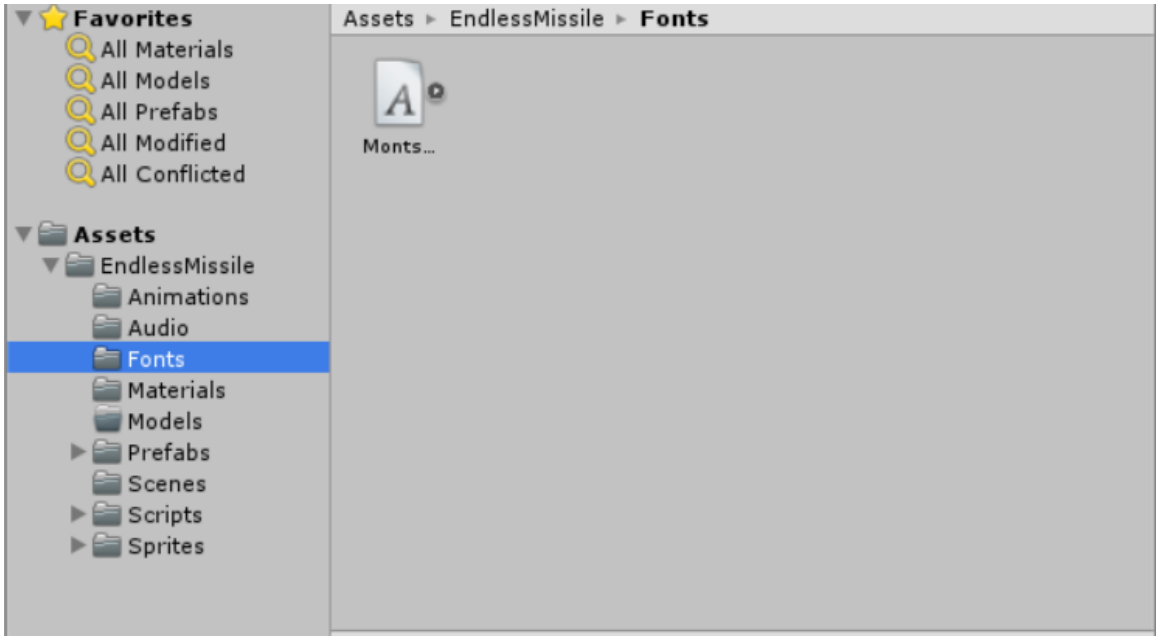
After that, you will be direct to a page that you can get your private and public code.

Copy 2 codes and paste it on LeaderboardManager component. That's it.

5. Customizing UI

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

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



6. Sounds

All sounds included in this game are free-to-use in commercial projects and are located under the *Audio* folder.

THANK YOU AND GOOD LUCK WITH YOUR GAMES!