

Rhythm N Jump

Game documentation and HowTo guide



1. Introduction.....	2
2. Getting started	2
3. How to play.....	2
4. Reskin	3
5. Basic Game Settings	3
Directory structure	3
Gameplay tweaking.....	4

1.Introduction

Rhythm N Jump has simplistic, minimalist art style gameplay. Endless, random and faster!

Features:

- Works on all platforms, PC, Mac, iOS, Android, etc
- Two game modes.
- Easy to Reskin.
- Simple to understand C# code.
- All assets included: graphics, sounds, and code.

2.Getting started

Open the blank project you want to **import** assets into. Choose Assets > **Import Package** > plus the name of this **package** you want to **import**, and the **Import Unity Package** dialog box displays, with all the items in the **package** pre-checked, ready to install.

To test the game, search start scene by project panel of unity editor to found the scene file named "start". You can also found the scene files under Assets/Rhythm N Jump/Scenes

Double click on it to active the scene. Then you can run the game correctly.

If you import this asset to Unity3d 2019, please follow this step to switch the scripting runtime version to .Net 4.x:

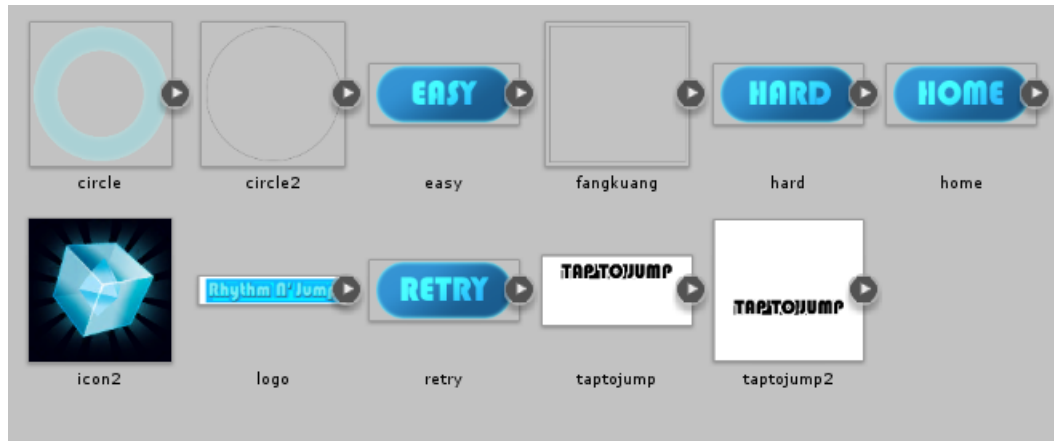
Go to Edit>Project Settings>Player, then select Other Settings tab and in Configuration block try switching Scripting Runtime Version to .NET 4.x Equivalent.

3.How to play

Tap the side of the big one to jump. Only big can jump over the small one!

4.Reskin

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

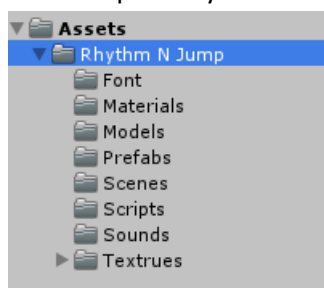


All other resources were under Assets/Rhythm N Jump/Models

5.Basic Game Settings

Directory structure

When import Rhythm N Jump your will see this directory structure in you Unity Project View.



- **Font:** Holds the font file used in the game.
- **Materials:** Holds the physics material used in the game.
- **Models:** Holds the FBX format models used in the game.
- **Prefabs:** Holds all the prefabs used in the game.
- **Scenes:** The scene that runs in the game is game.
- **Scripts:** Holds all the scripts used in the game. Each prefab contains one or more of these scripts. All code is written in C#, and the game logic is clear so you can easily understand

these codes.

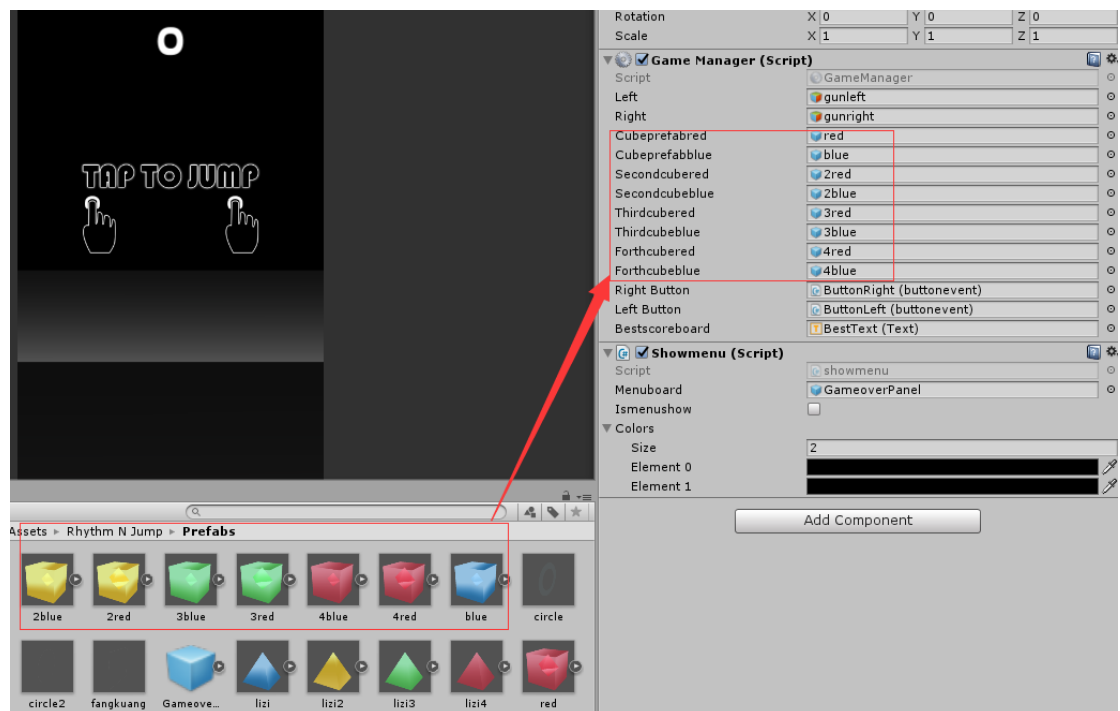
- **Sounds:** Holds all the sounds used in the game.
- **Textures:** Holds all the textures used in the game.

Gameplay tweaking

Most of important gameplay parameters can be configured in component: GameManager.

GameManager

The GameManager component is attached to a game object also named GameManager in the hierarchy.



You can tweak the gameplay by modifying following variables:

- Left: The left point of the box spawn position.
- Right: The right point of the box spawn position.

Please contact zhang_frank@hotmail.com if there is anything unclear.

THANK YOU AND GOOD LUCK WITH YOUR GAMES!