

Match 3 - Unity Template



General Notes

This document refers to “gems” as the matchable items.

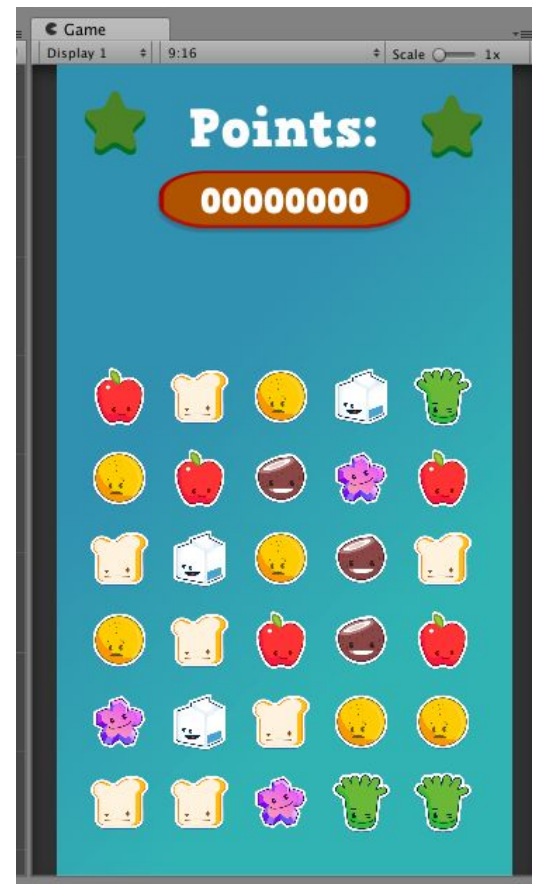
Note on v1.0: This is not finalized and may have minor bugs. However as a starting point, being a template it is functional.

For various arbitrary resolutions and may exhibit rendering and input issues. Make sure to work on these.

No Game Over checks. No block gravity, only new block fall in place.

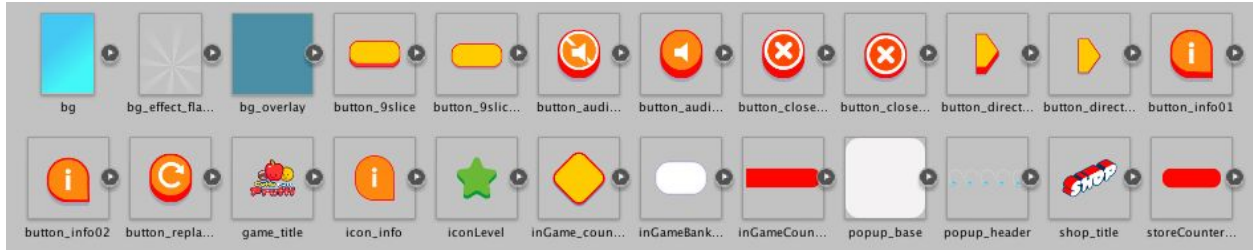
Make sure to set it up for portrait preview 9:16 aspect ratio, otherwise it would look correct.

Every gem/item will give 100 points upon match.



Asset details

There are many assets / resources included, many of which are not yet used. Mostly UI related graphic sprites.



Code details

Most of these classes are in fact static classes and many of them are also not MonoBehaviour because it's not necessary.

Core

As per name, this is the core of the code where most of the work and calculations are done. This is actually a MonoBehaviour and attached to a GameObject by the same name. We do this so we have one central point to Init and to do Update() over the entire code. However all the other methods are static.

GemScript

This script is attached to every single gem. This handles all the input even checks and reacts to those: Clicks and drags. Then it will initiate a swap if applicable. Another functionality is to check adjacent gems which we naturally need.

GemSwapper

This script is attached to the gem_swap GameObjects and used to facilitate the visual swap animation between gems.

GO

This class is for referencing GameObjects. For performance reasons its a bad idea to query GameObjects from the scene during runtime. As such, whenever referring to any GameObject, this class is used.

MyResources

This class is used to load in all the resources. We only load in the character (gem) sprites here. These are referred when changing and setting gem types from outside.

Speed

Using delta time can get confusing, so this class offers a simple way to get relative delta speeds for any kinds of movements. The `get()` method has an optional factor to adjust speed.

Util

Simple class for convenience utility methods and extension method. Take a look around, this guide will not go into detail on these. But many of them are used of course in the codebase.

Scene details

One scene. Mainly using Canvas because we use 2D UI elements.
2 Audio Sources, one for music, one for the single sound effect used.
The scene uses a matrix of `Unity.UI.Images`. These are used for the gems/items.

Gem Swap GameObjects:

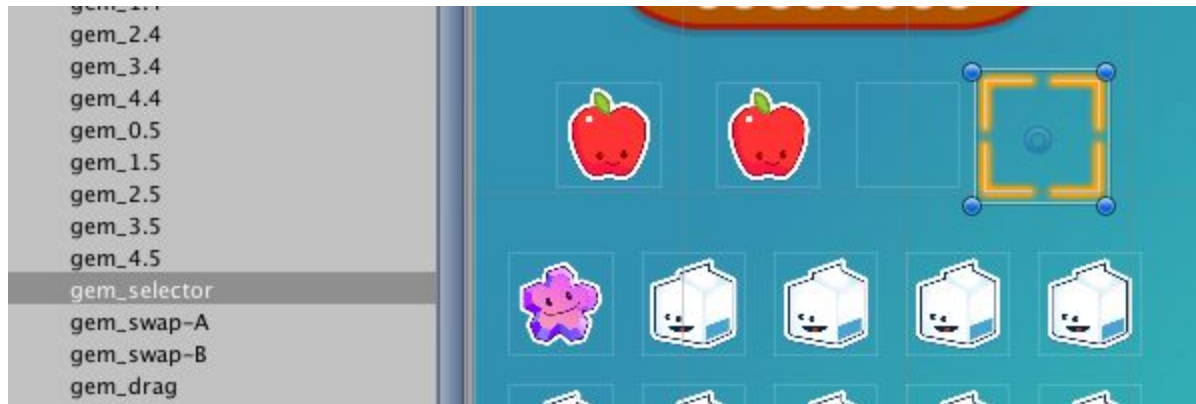
We want the matrix to stay fixed for data reasons, so we have these extra objects just for the animation.



These have the GemSwapper script attached.

Gem Selector:

This GameObject is a sprite graphic, which indicates the user that a gem has been selected, via click/tap. It will be moved over the gem and made visible.



Gem Drag:

This GameObject is basically just a rectangle used to check which gem the user drags over, while doing a drag and drop / swipe motion. The rectangle is moved where the mouse position is and checked where it overlaps with gems.

