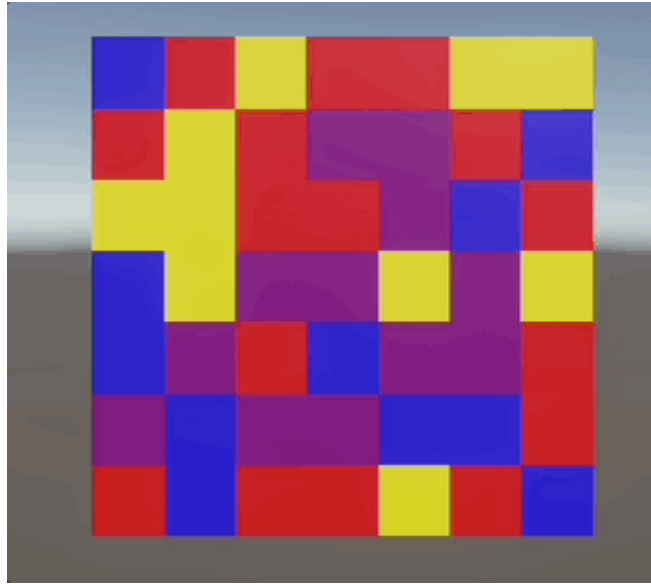


ReadMe

.....

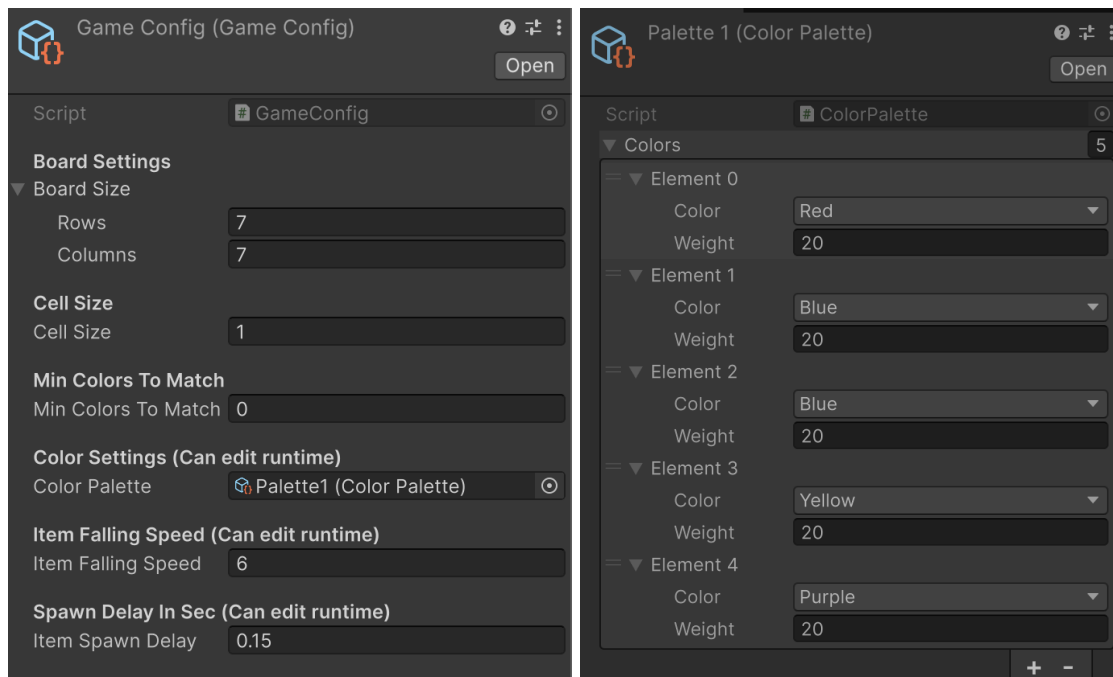
Tap Match

A game prototype made with Unity 6.



by [Bismy](#) on November 04

The game can be played by tapping on any cell. Different parameters of the game can easily be configured using Unity scriptable objects. See reference below.



Game

This is the main class that controls the game board overall flow. It creates the game board and reacts to user inputs by finding matches and changing the game board.

Board

This maintains the list of cells in the game board in a **2D array data structure** which is suitable to perform an efficient search function of neighboring cells. It also controls the destruction and creation of colored items in the cells. It uses **object pooling** technique to minimize the amount of game objects created in the game by reducing the load to the garbage collector.

CommandInvoker

The game is using the **command design pattern** to schedule the falling action of items in the game board. This is excellent to keep track of different time bound animating actions taking place in the game. For example it can also help detect the stability (idle) of the game board by looking at the number pending commands at a certain time.

EventSystem

This is a custom event system developed to ensure smooth communication flow that avoids **unnecessary coupling** between classes. It is used to listen to user inputs.

MatchFinder

This using the **BreadthFirstSearch** searching algorithm to quickly find the matching items closed the cell that the user tapped. The 2D array represents a graph where each element is connected to 4 other elements. BFS is an efficient algorithm to traverse a graph data structure.

RandomWeightedColor & Color Palette

This is using the weighted random picker algorithm to pick a random color based on the weights assigned to it. This is good for adjusting the **difficulty** of a particular level by making some colors appear more often than others.

.....

Thanks,

Bismy.