

VERTIGO DEMO PROJECT

I prepared a game where the player tries to rotate hexagonal tiles in order to create certain patterns. In this text i explained what i did and what i could not do in this project.

WHAT I DID ?

- ❖ The core mechanic added: Select a hexagonal group and turn them clockwise or counterclockwise. If a 3-hexagonal group of the same color occurs, they will explode, if not the hexagonal pieces returns to their original position.
- ❖ The game grid can be change from the Unity editor. The default grid is 8x9. To change the grid, there is a "Board" object in the "Level" scene. Grid Width and Grid Height can be found in the inspector of "Board" object. When they change the grid will change too. But camera size and position must be readjusted for gameplay.
- ❖ Colors and color count of the hexagons can be change from the Unity editor. By default, there are 5 colors. To change the color count there is a "Board" object in the "Level" scene. An array named "Hexagon Colors" can be found in the inspector of "Board" object. Size of the array can be change. After that color objects must added to array from prefabs folder. To add a new color, go to prefabs folder, drag and drop the "hexagon" prefab in the scene, change it to color you want from that object's sprite editor, create a new tag for that color and add it to this object. Drag and drop the object in the prefabs folder and new color is ready to use.
- ❖ Scoring is 5 times the exploded block count.
- ❖ A bomb hexagon appears on every 1000 score. The bomb's function is to count from a number every time an action is made on the board. When the number reaches zero, a bomb explodes and the game is lost. Every hexagon can be a bomb. In play mode all hexagons have a bool variable named isBomb. If it changes to true the hexagon becomes a bomb.
- ❖ The game is over when bomb explodes.
- ❖ New sounds can be added from Auido manager in the scene. Also the sound's name, volume and the pitch can be changed from Auido manager's inspector.

WHAT I COULD NOT DO ?

- ❖ I cannot control if there is a available move in the scene or not. I wrote a .cs file named deadLock. The idea was, after every move change every hexagons place in the grid with its neighbours and look for if there is a match or not. After that change back to it's original place and return the value. But it didn't work everytime. So i didn't use it but it can be found in scripts folder.