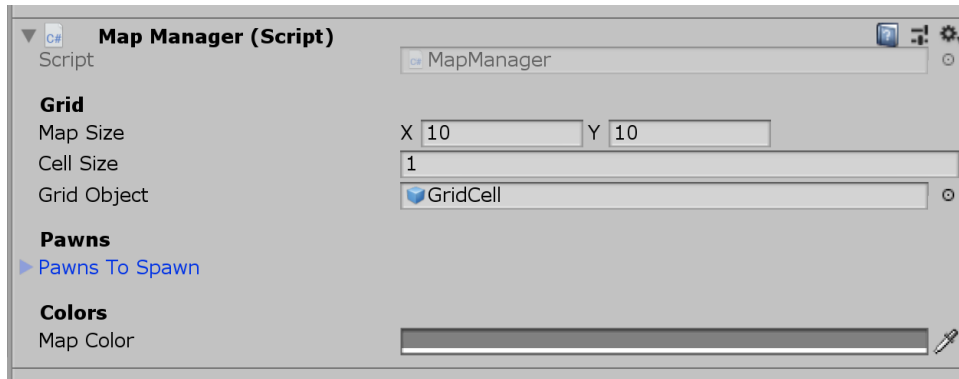


## Map Setup

Open Scenes->SampleScene

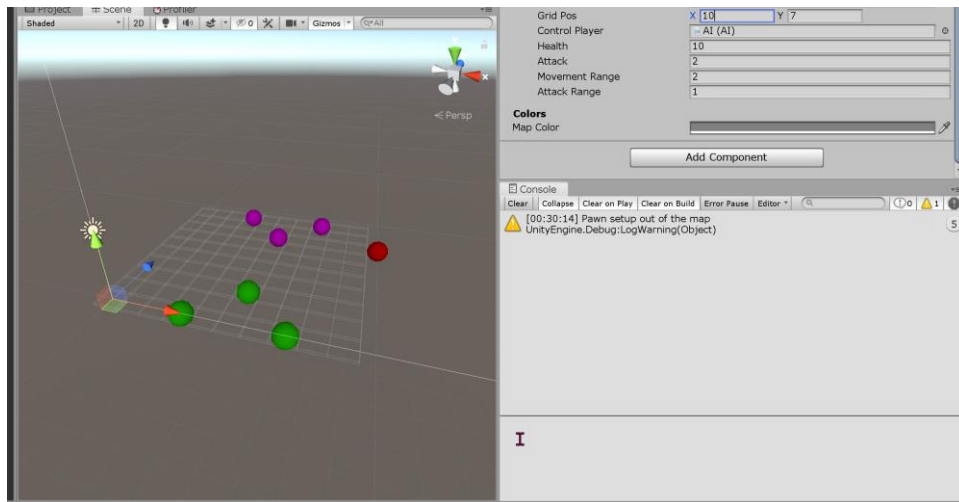
Click on MapManager



- Map size -> The grid NxM size
- Cell size -> The size that each grid will be assigned logically (caution with this because the object sizes)
- Grid Object -> Object that will be spawned as a cell. **Make sure it has the same size as the Cell Size!!** If you don't want to worry about sizes ask me and I could scale sizes given the `Renderer.BoundingBox`, also, **the object needs to have a renderer!** (to change color)
- Pawns to Spawn -> List of data structures to spawn a pawn

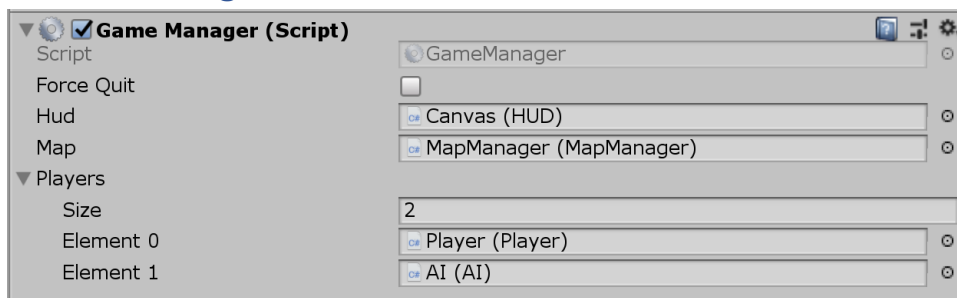


- Pawn Object -> The pawn prefab to spawn, (Needs a Pawn component!)
- Grid Pos -> position of the grid where to spawn
- Control Player -> Which Player will be the owner (Player, or AI)
- Health -> Health that the pawn will have
- Attack, Movement range, Attack Range -> Same as health



The map and pawns will be drawn on the scene view, the color of the pawns will be the player's color, if any pawn setup couldn't be spawned it will change the color to red (as you can see in the image outside of the grid) and show a warning in the console

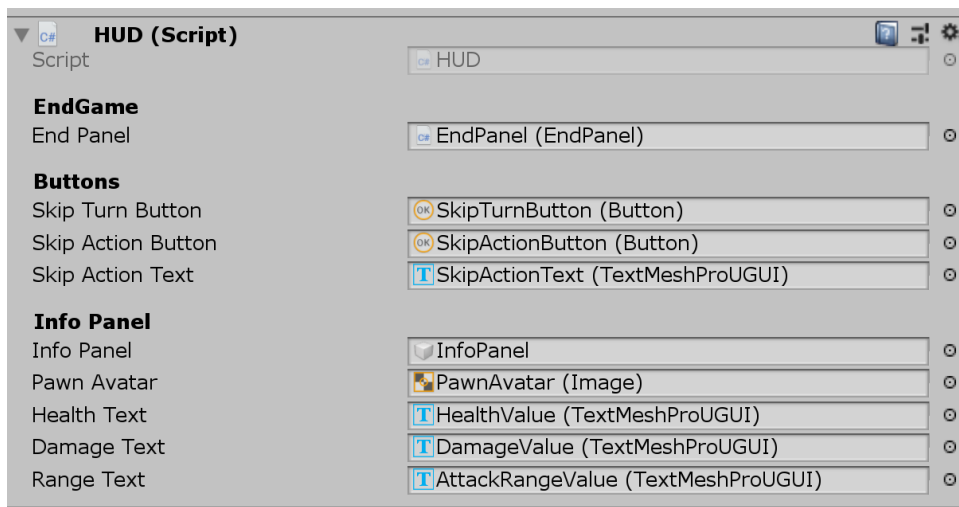
## Game Manager



The game manager needs a reference to the HUD and the MapManager

Also, the Players list to iterate over the turns (yes, theoretically this admits more than 2 players, but if one is wiped out the game ends), the first player on the list will be who starts the game!

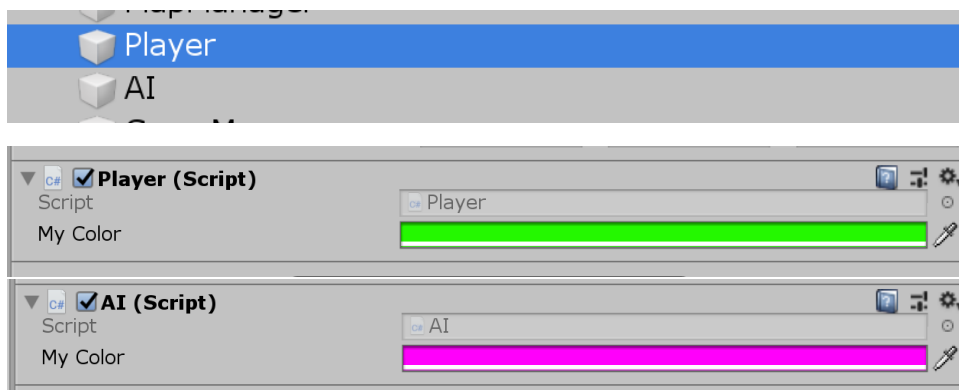
## HUD



HUD Manager, assign the buttons and texts it wants (I could split the info panel into another component, but as the HUD script is not too large I left it this way)

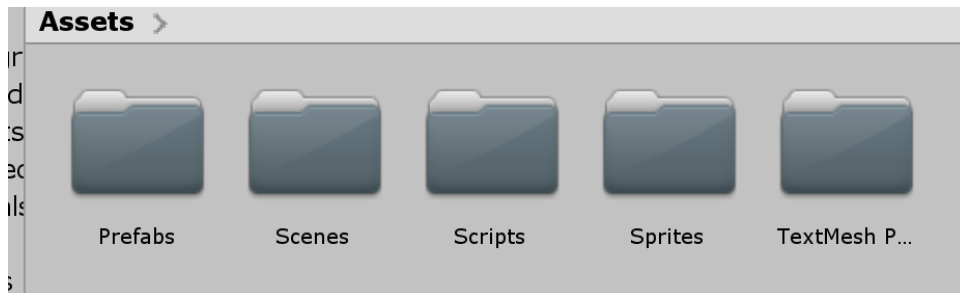
## Players

The players are simple empty gameobjects with a Player Component, change the color of the player here!

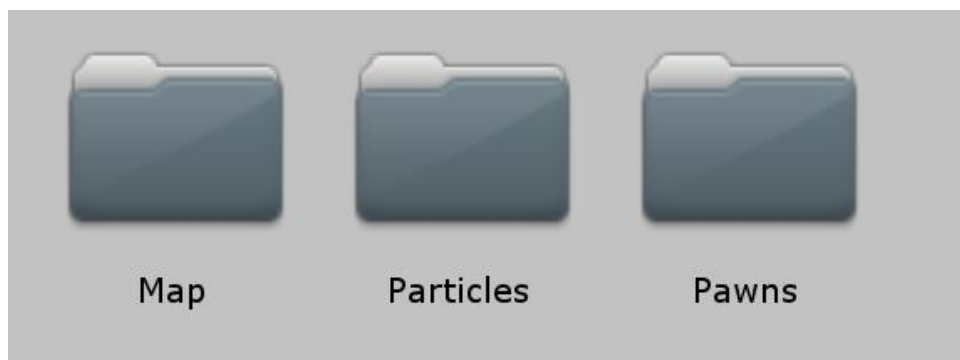


## Assets Folder Structure

I think it is logical but, I've splitted into these 4 folders (the TextMeshPro is automatically generated when you use any TextMesh)



The Prefabs folder is splitted in these



And the scripts folder is splitted in these

