Zig Zag 2D Documentation

Welcome to the Zig Zag 2D documentation. Below you will find a brief overview of the important scripts used in the game and how they are setup.

If there is anything that needs further explaining or there is something that is not working please don't hesitate to send me an e-mail at **nich.fagan@gmail.com.**

GameManager.cs

The GameManager is where all the action happens. It handles all the game logic such as generating the tunnel walls, moving the player, checking for collision detection, etc.

Inspector Properties:

General Settings	
Game Camera	The camera the game is rendered on. This is used to get the width and height of the world that is shown by the camera.
Start UI	The GameObject that displays the start screens UI.
Player Select UI	The GameObject that displays the player select screens UI.
Game UI	The GameObject that displays the game UI.
Over UI	The GameObject that displays the game over screens UI.
Show Collision Boxes In Scene	If this is checked, then the collision boxes for the walls and the player will be visible in the Scene window.
Player Settings	
Player Infos	The list of players that can be used in the game.
Player Infos - Player Prefab	The prefab of the Player that will be instantiated and used in the game.
Player Infos - Locked	If this is checked then this player info is locked and cannot be selected on the player select screen until it becomes unlocked.
Player Infos - Unlock Amount	The amount of drops that must be spend on this player info for it to be unlocked
Player Infos - Icon	The sprite to use on the player select screen.

Player Infos - Icon Color	The color of the icon Image on the player select screen.
Speed	The speed that the player moves at in the game.
Score Multiplier	The multiplier that is applied to the score. The higher this is the faster the game score will increment.
Drop Settings	
Drop Chance %	The chance that a drop will be placed in the tunnel when ever a new length of tunnel is created.
Drop Collect Amount	The amount the drop collected count increases every time the player collects a drop.
Drop Edge Padding	The amount of space between a drop and the edge of the tunnel.
Drop Prefab	The prefab to instantiate for each drop.
Background Settings	
Bkg Material	The Material to use when creating the background Mesh and Mesh Renderer.
Bkg Move Speed	The speed that the background moves at.
Bkg Repeat Size	The size each tile of the background texture.
Tunnel Settings	
Tunnel Material	The Material to use for the tunnel.
Texture Fills Tunnel	If this is check then instead of the tunnel being “empty” with tunnel walls, the tunnel becomes a filled track.
Align Texture With Tunnel	If this is checked then the texture of the tunnel will align with the edges of the tunnel.
[Unit / Texture] Size	If “Align Texture With Tunnel” is checked then this will be “Unit Size”, if not then it will be “Texture Size”. In either case it is the size of each texture tile.
Min Tunnel Size	The minimum size/width of the tunnel. If “Align Texture With Tunnel” is checked then this is a multiple of “Unit Size”.
Max Tunnel Size	The maximum size/width of the tunnel. If “Align Texture With Tunnel” is checked then this is a multiple of “Unit Size”.
Min Tunnel Length	The minimum length of a tunnel segment. If “Align Texture With Tunnel” is checked then this is a multiple of “Unit Size”.

Max Tunnel Length	The maximum length of a tunnel segment. If “Align Texture With Tunnel” is checked then this is a multiple of “Unit Size”.

Player.cs / Drop.cs

These two scripts are identical in functionality but remain separate incase additional functionality where to be added to either of them. The Player script is attached to each player prefab and the drop script is attached to the drop prefab. All they do is provide a Collision Size that is used during collision detection.

Inspector Properties:

Collision Size	The size of the collision radius to be used then checking collisions between the player / walls and the player / drops.

GameUI.cs

Handles displaying / updating the UI that appears when the game is being played.

Inspector Properties:

Game Camera	The camera the game is rendered on. This is used to get the width and height of the world that is shown by the camera.
Parent Canvas	The Canvas that the UI is a child of / being rendered on.
Current Score Text	The UI.Text component that the current score will be displayed in.
High Score Marker	The GameObject that displays the position in the tunnel where the players high score is at.
Average Score Marker	The GameObject that displays the position in the tunnel where the players average score is at.
High Score Marker Text	The UI.Text component that the high score will be displayed in.
Average Score Marker Text	The UI.Text component that the average score will be displayed in.

OverUI.cs

Handles displaying / updating the UI that appears when the game is over.

Inspector Properties:

Current Score Text	The UI.Text component that the current score will be displayed in.
High Score Text	The UI.Text component that the high score will be displayed in.
Average Score Text	The UI.Text component that the average score will be displayed in.
Coins Amount Text	The UI.Text component that the amount of drops collected will be displayed in.

PlayerSelectUI.cs

Handles displaying / updating the UI that appears when the the player clicks the SELECT PLAYER button on the OverUI screen.

Inspector Properties:

Coins Amount Text	The UI.Text component that the amount of drops collected will be displayed in.
Item Prefab	The PlayerSelectListItem prefab that will be instantiated and placed in the list for each Player Infos in the GameManager.
List Content	The Transform that all the instantiated Item Prefabs will be placed in.

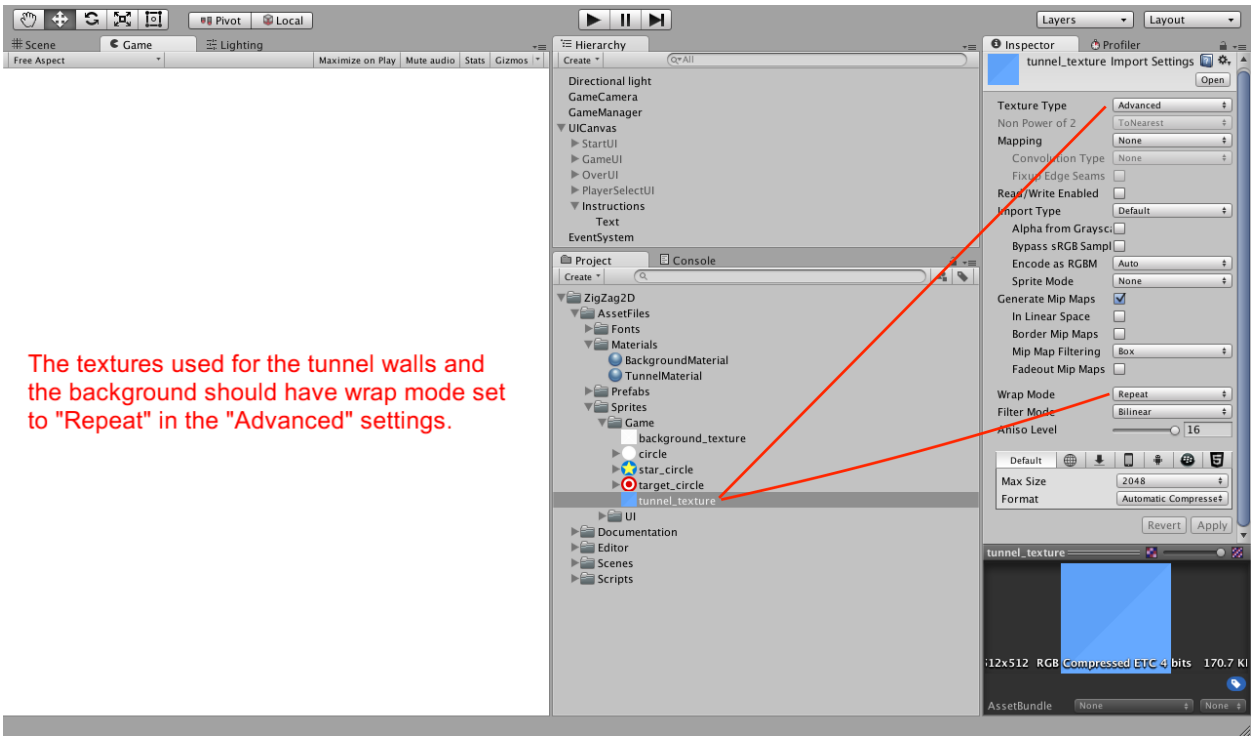
PlayerSelectListItem.cs

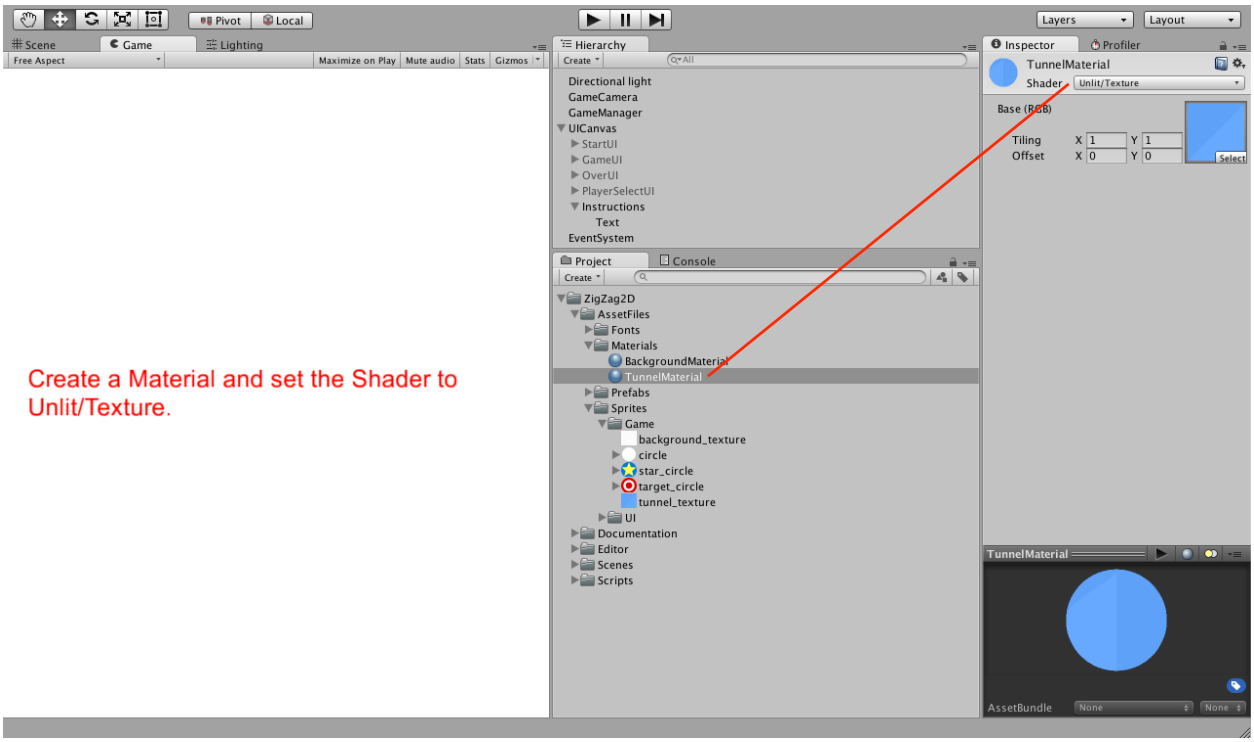
Handles displaying a single item in the player select screens list.

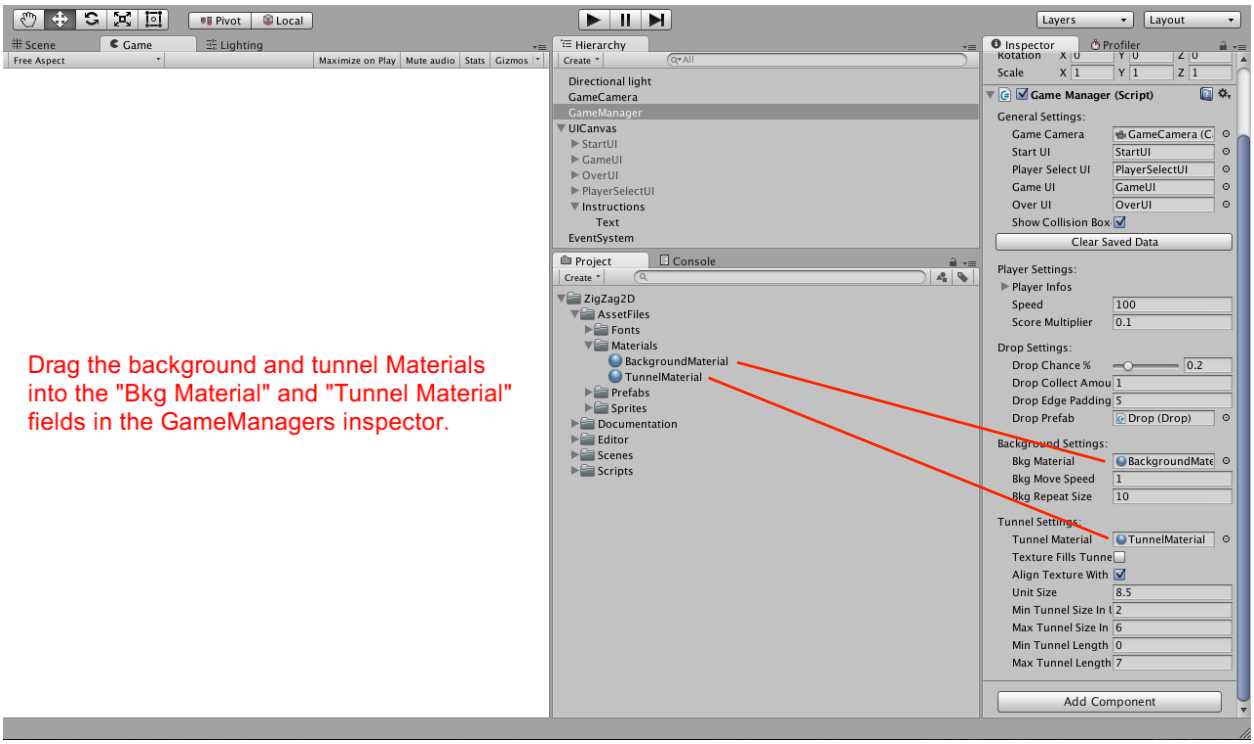
Inspector Properties:

Icon Image	The UI.Image component that will display the PlayerInfo.Icon sprite.
Coins Amount Text	The UI.Text component that the amount of drops collected will be displayed in.
Unlock Button	The GameObject that displays the unlock button.
Select Button	The GameObject that displays the select button.
Background Image	The UI.Image component that displays the icons background. This is the Image whose color will change when selected.
Normal Color	The color of the Background Image when the item is not selected.
Selected Color	The color of the Background Image when the item is selected.

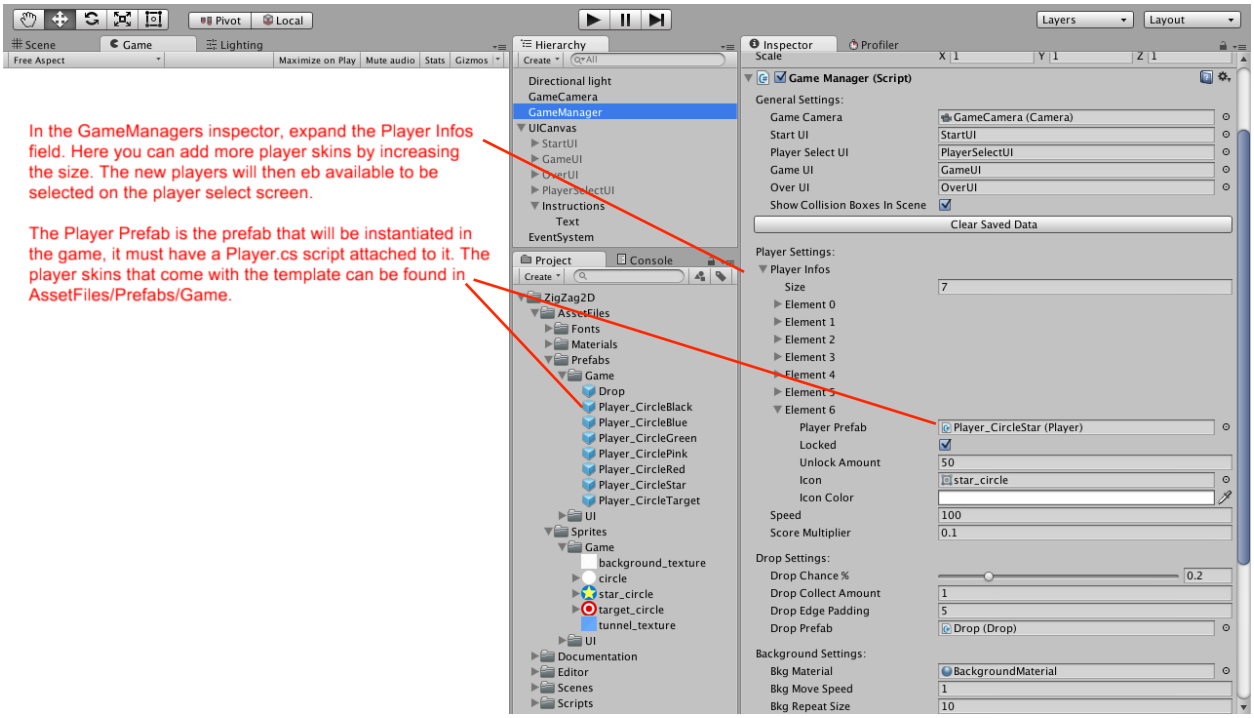
Setting Tunnel Wall / Background Textures



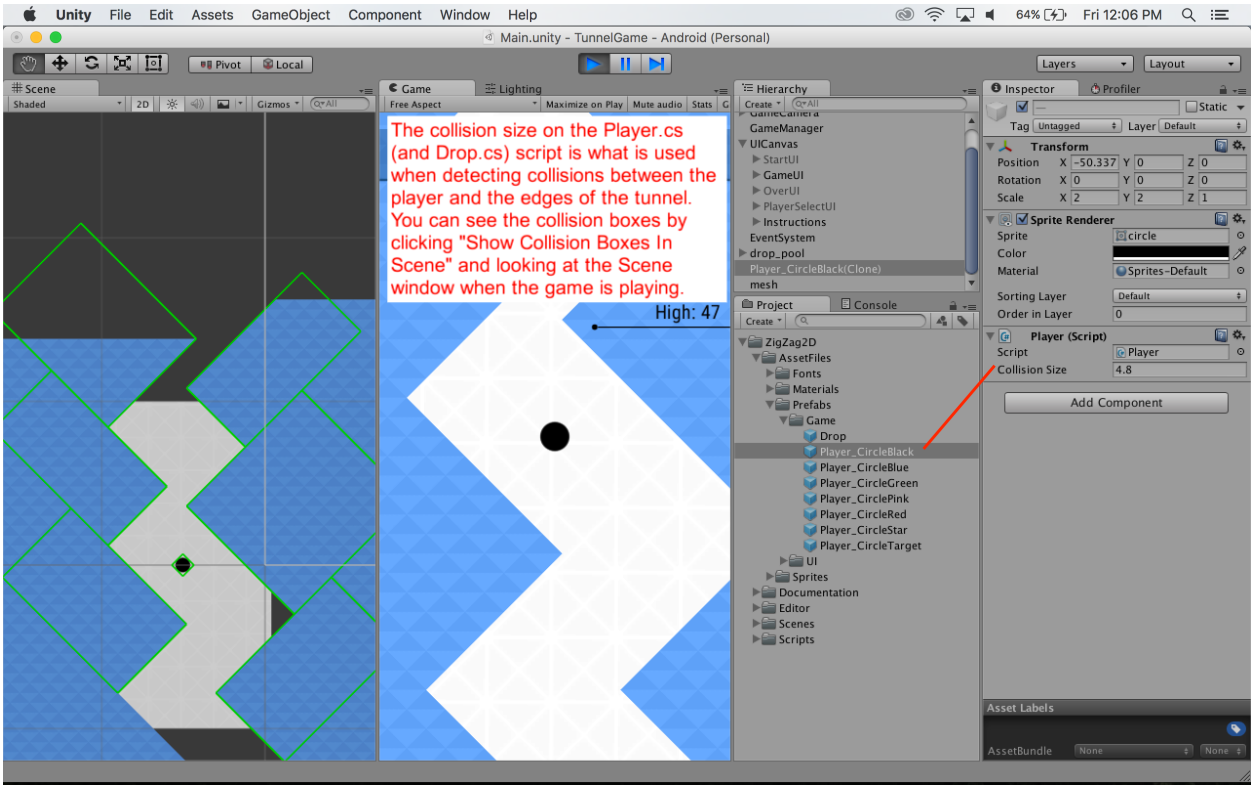

Create a Material and set the Shader to Unlit/Texture.


Drag the background and tunnel Materials into the "Bkg Material" and "Tunnel Material" fields in the GameManagers inspector.

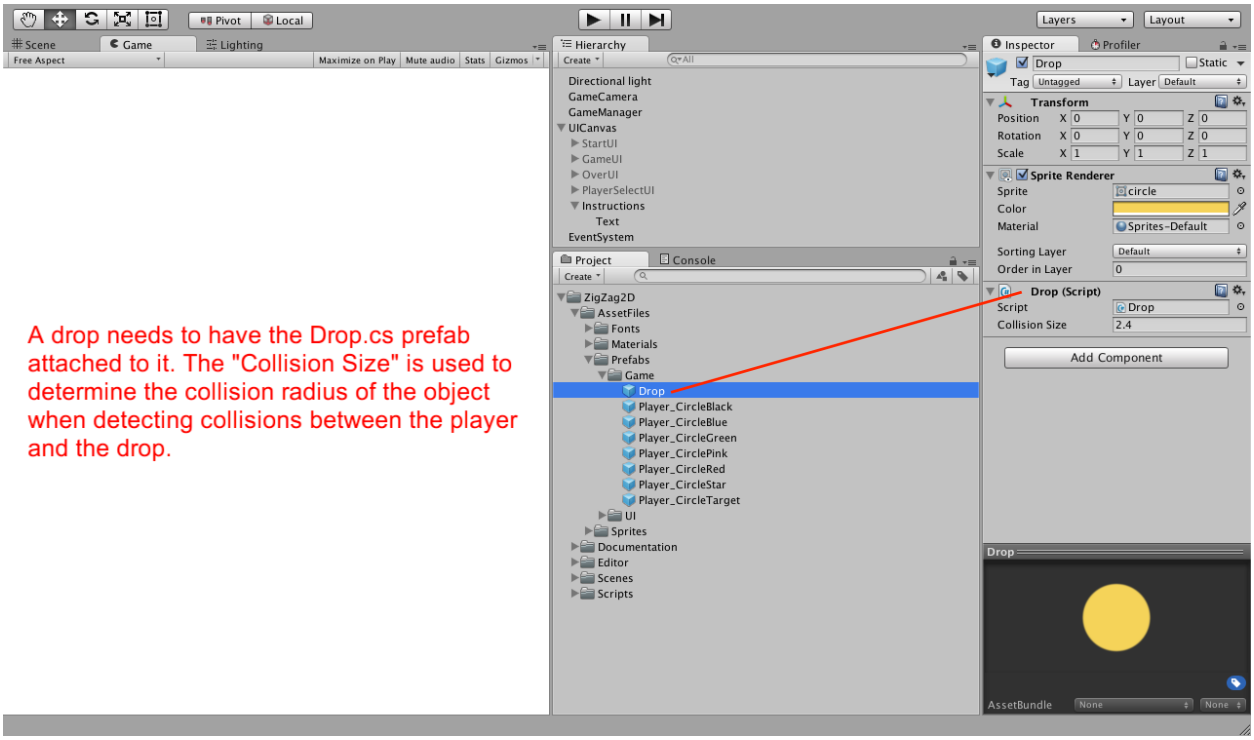
Setting Player Skins



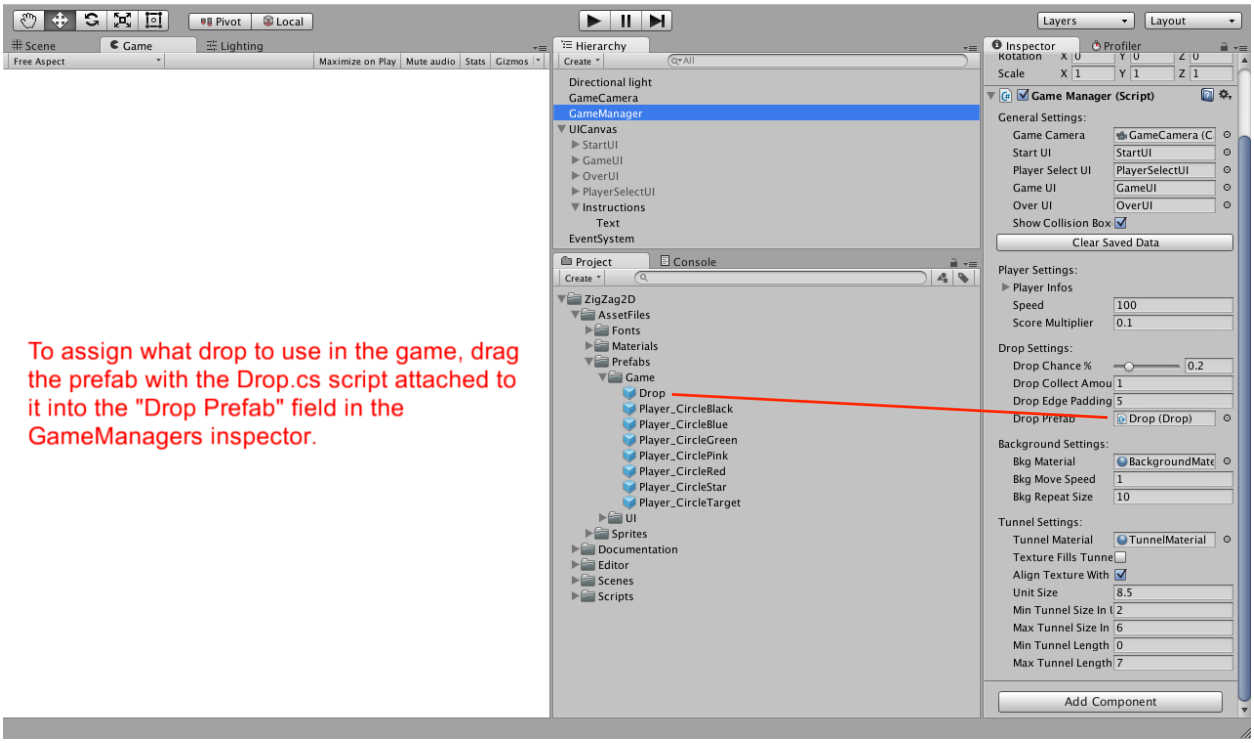
Setting Collision Sizes



Setting Drops



A drop needs to have the Drop.cs prefab attached to it. The "Collision Size" is used to determine the collision radius of the object when detecting collisions between the player and the drop.



To assign what drop to use in the game, drag the prefab with the Drop.cs script attached to it into the "Drop Prefab" field in the GameManagers inspector.