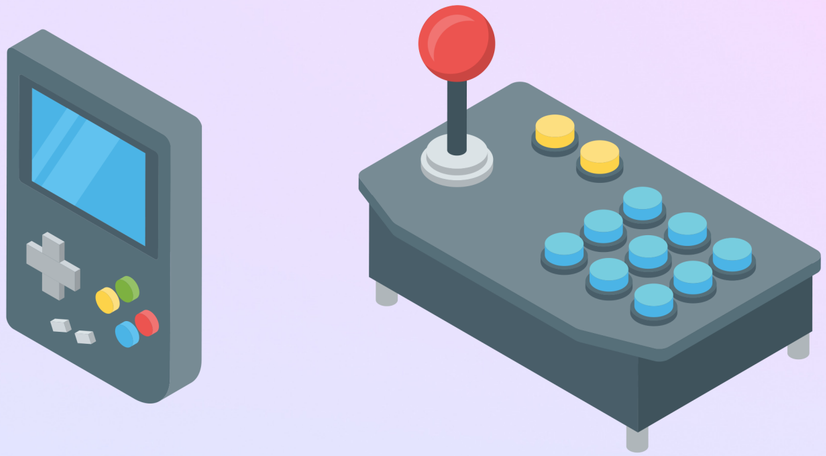


WOLVEDEN ACADEMY

NETWORKING AND MULTIPLAYER ONLINE GAMES

# MULTIPLAYER NEW ERA


BY PONGSATHORN KIATTICHAROENPORN (MEW)

WEEK 6 : CONNECTING ONLINE 1

# Presentation Update 1

**(Present your Progress Update Multiplayer Final Project)**

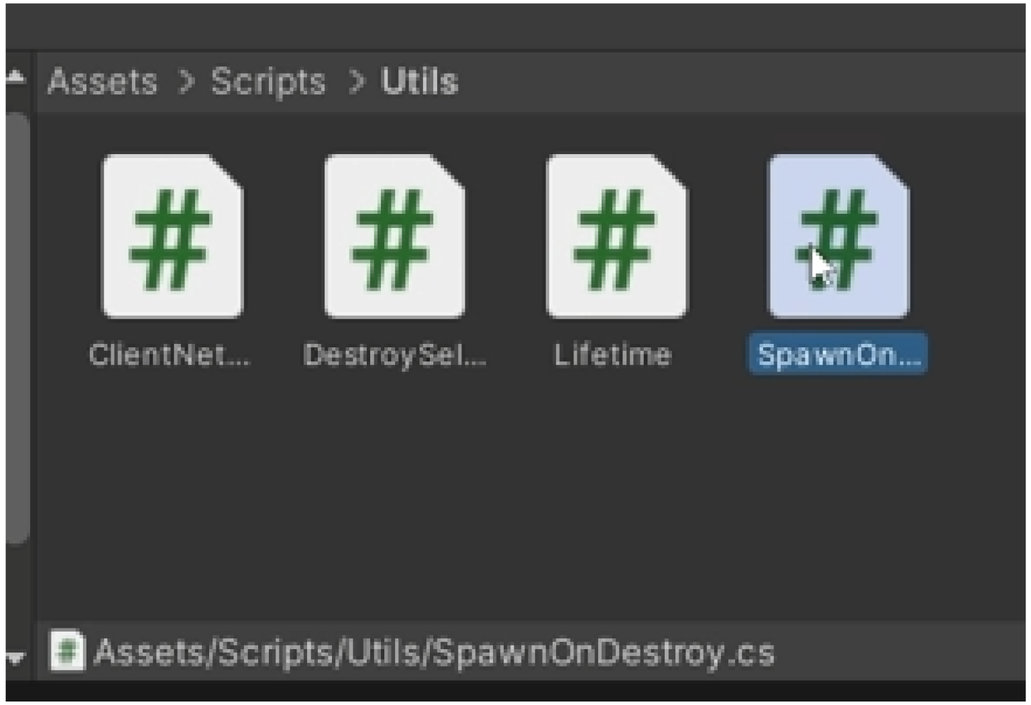
# Connecting Online 1

# Topic in this week

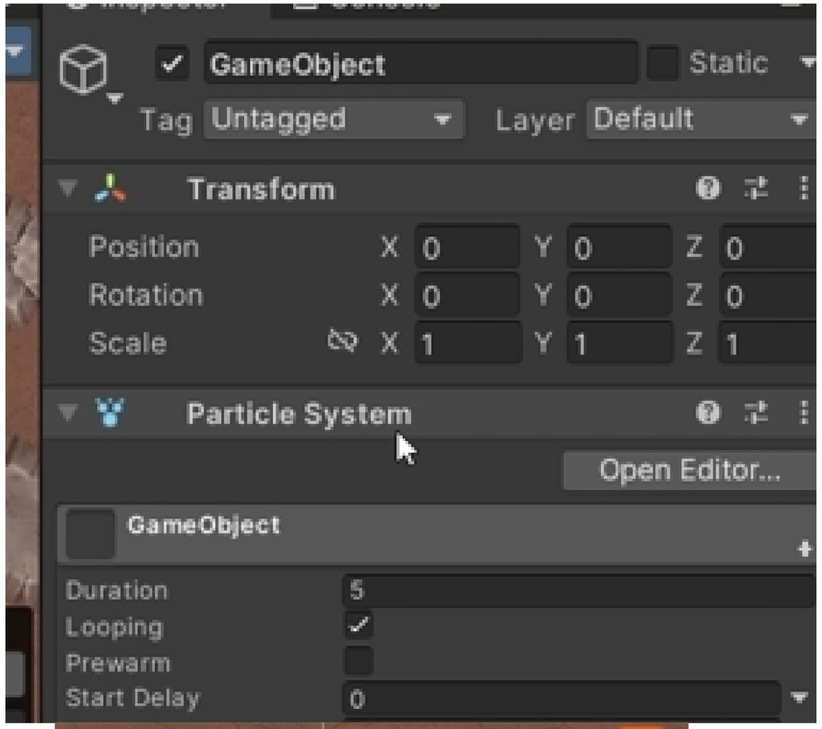
## Workshop

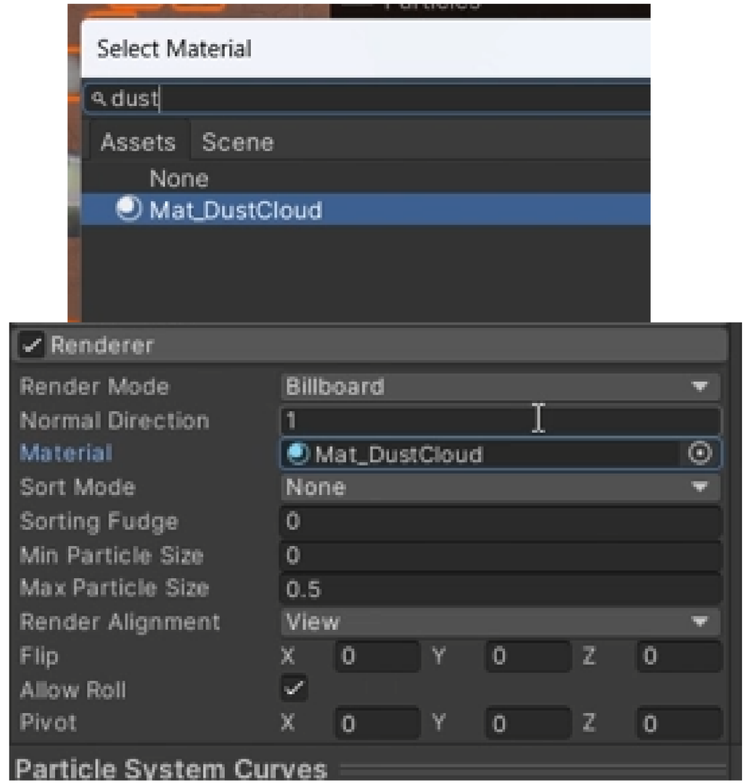
- Combat Polish
- Setup Main Menu
- Application Controller
- Authentication
- ~~Auth Improvements~~ (Next Week)

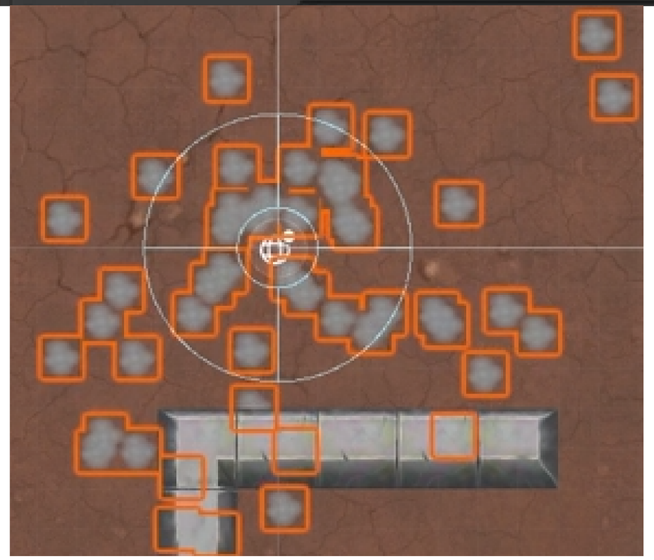
# Combat Polish

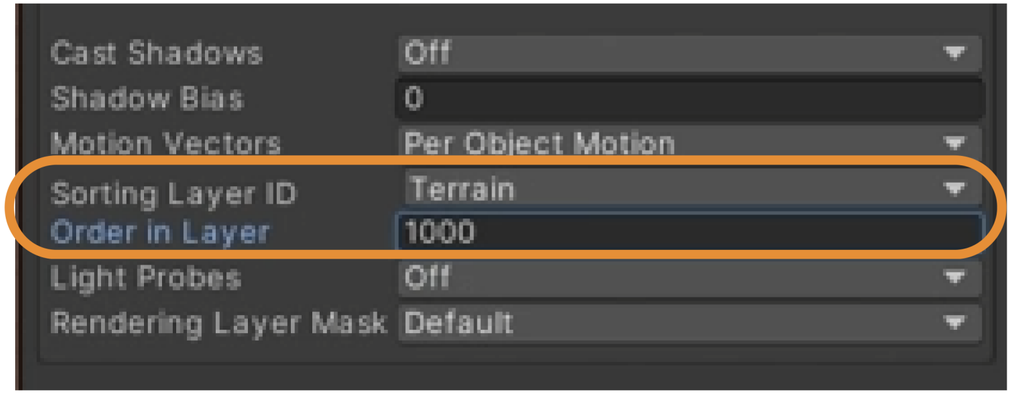

Assets > Scripts > Utils
ClientNet...
DestroySel...
Lifetime
SpawnOn...
Assets/Scripts/Utils/SpawnOnDestroy.cs

```
4
                0 references
5   public class SpawnOnDestroy : MonoBehaviour
6   {
                    1 reference
7       [SerializeField] private GameObject prefab;
8
                    0 references
9       private void OnDestroy()
10      {
11          Instantiate(prefab, transform.position, Quaternion.identity);
12      }
13  }
14
```

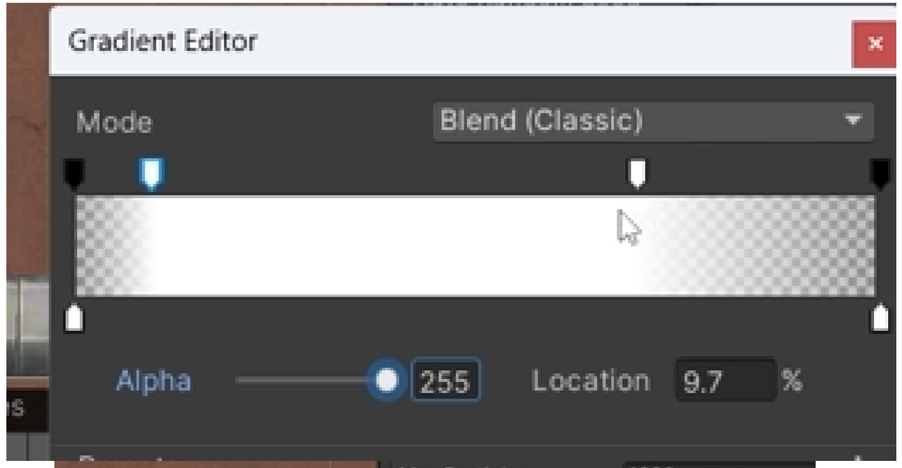

GameObject
Static
Tag Untagged
Layer Default
Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Particle System
Open Editor...
GameObject
Duration 5
Looping
Prewarm
Start Delay 0

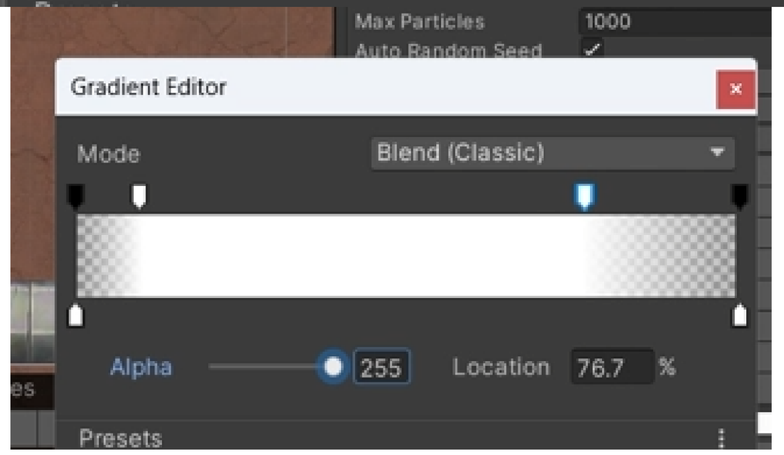

Select Material
dust
Assets Scene
None
Mat_DustCloud
Renderer
Render Mode Billboard
Normal Direction 1
Material Mat_DustCloud
Sort Mode None
Sorting Fudge 0
Min Particle Size 0
Max Particle Size 0.5
Render Alignment View
Flip X 0 Y 0 Z 0
Allow Roll
Pivot X 0 Y 0 Z 0
Particle System Curves

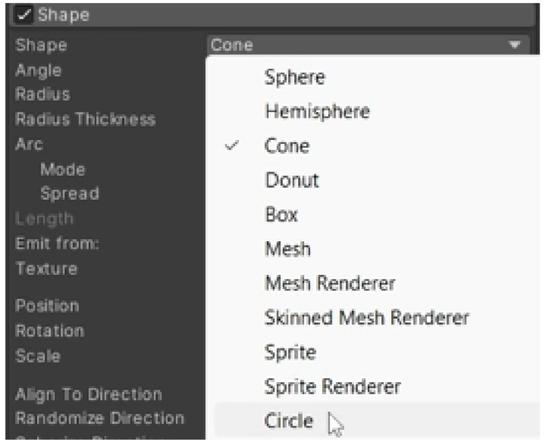


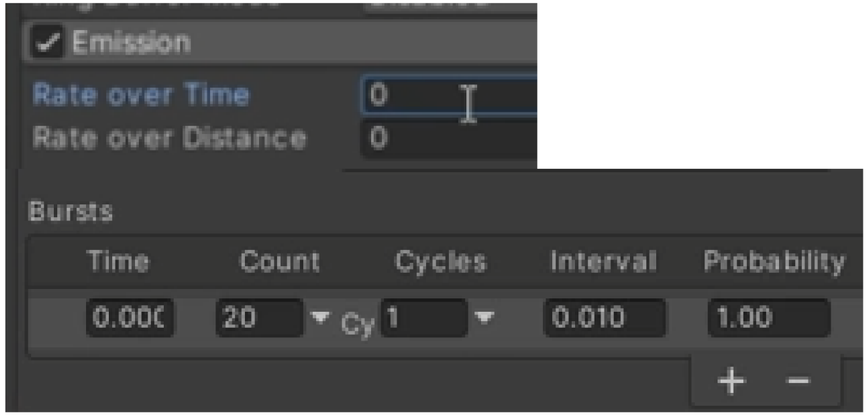

Cast Shadows Off
Shadow Bias 0
Motion Vectors Per Object Motion
Sorting Layer ID Terrain
Order in Layer 1000
Light Probes Off
Rendering Layer Mask Default

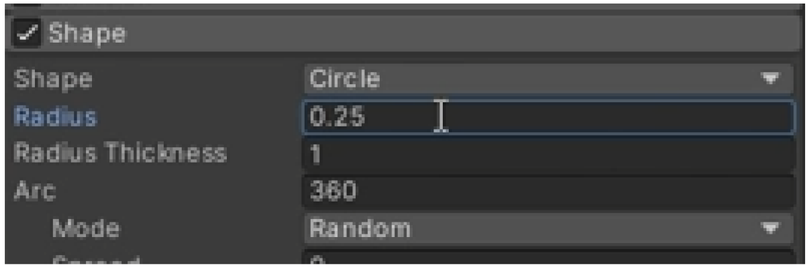

Color over Lifetime
Color
Color by Speed

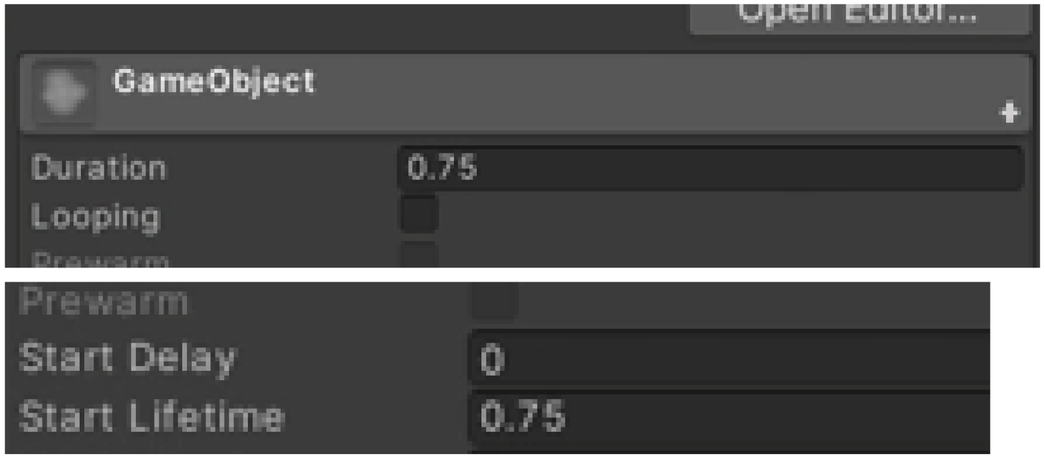

Gradient Editor
Mode
Blend (Classic)
Alpha
255
Location
9.7
%

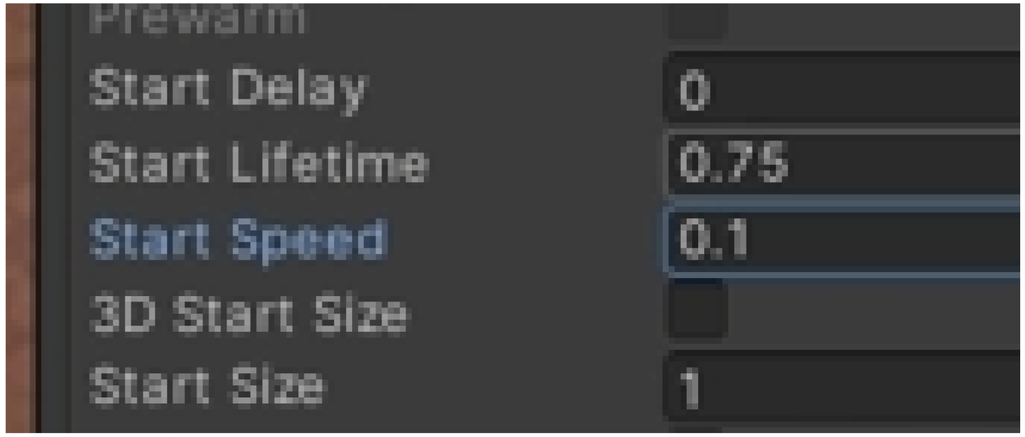

Max Particles
1000
Gradient Editor
Mode
Blend (Classic)
Alpha
255
Location
76.7
%
Presets


Shape
Shape
Cone
Angle
Radius
Radius Thickness
Arc
Mode
Spread
Length
Emit from:
Texture
Position
Rotation
Scale
Align To Direction
Randomize Direction
Sphere
Hemisphere
Cone
Donut
Box
Mesh
Mesh Renderer
Skinned Mesh Renderer
Sprite
Sprite Renderer
Circle

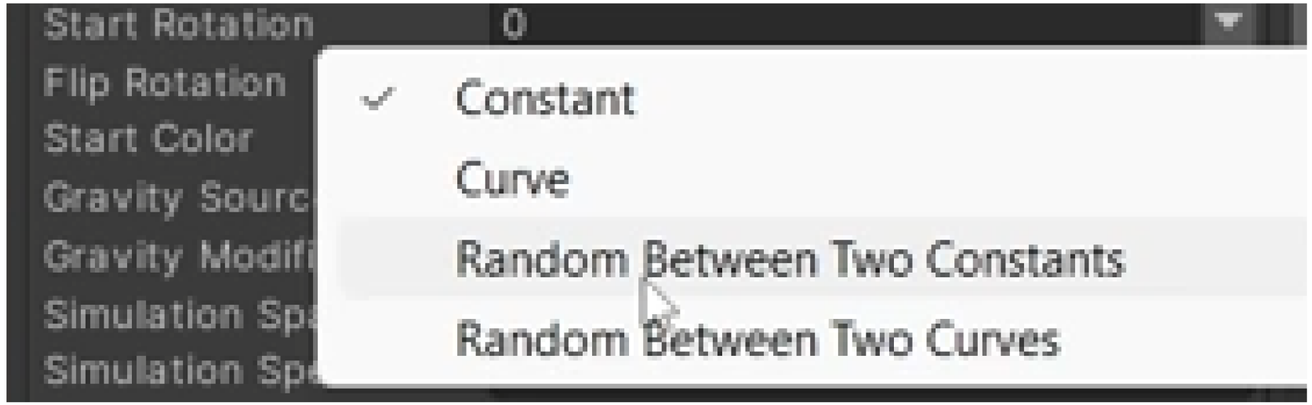

Emission
Rate over Time
0
Rate over Distance
0
Bursts
Time
Count
Cycles
Interval
Probability
0.000
20
Cy
1
0.010
1.00


Shape
Shape
Circle
Radius
0.25
Radius Thickness
1
Arc
360
Mode
Random


Open Editor...
GameObject
Duration
0.75
Looping
Prewarm
Start Delay
0
Start Lifetime
0.75


Prewarm
Start Delay
0
Start Lifetime
0.75
Start Speed
0.1
3D Start Size
Start Size
1

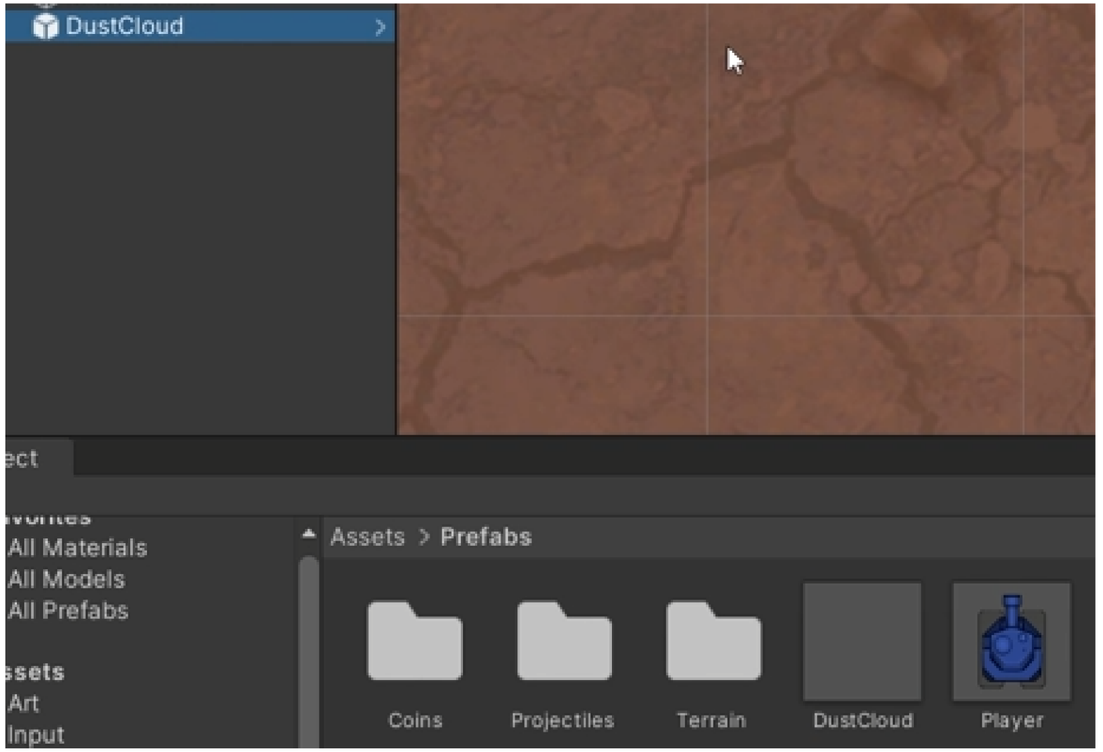

Particles
Pause
Restart
Stop
Playback Speed
1.00
Playback Time
0.68
Particles
20
Speed Range
0.0 - 0.0
Simulate Layers
Nothing
Resimulate
Duration
0.75
Looping
Prewarm
Start Delay
0
Start Lifetime
0.75
Start Speed
0
3D Start Size
Start Size
1
3D Start Rotation
Start Rotation
0
Flip Rotation
0
Start Color
Gravity Source
3D Physics
Gravity Modifier
0
Simulation Space
Local
Simulation Speed
1
Delta Time
Scaled
Scaling Mode
Local
Play On Awake*
Emitter Velocity Mode
Rigidbody
Max Particles
1000
Auto Random Seed
Stop Action
None

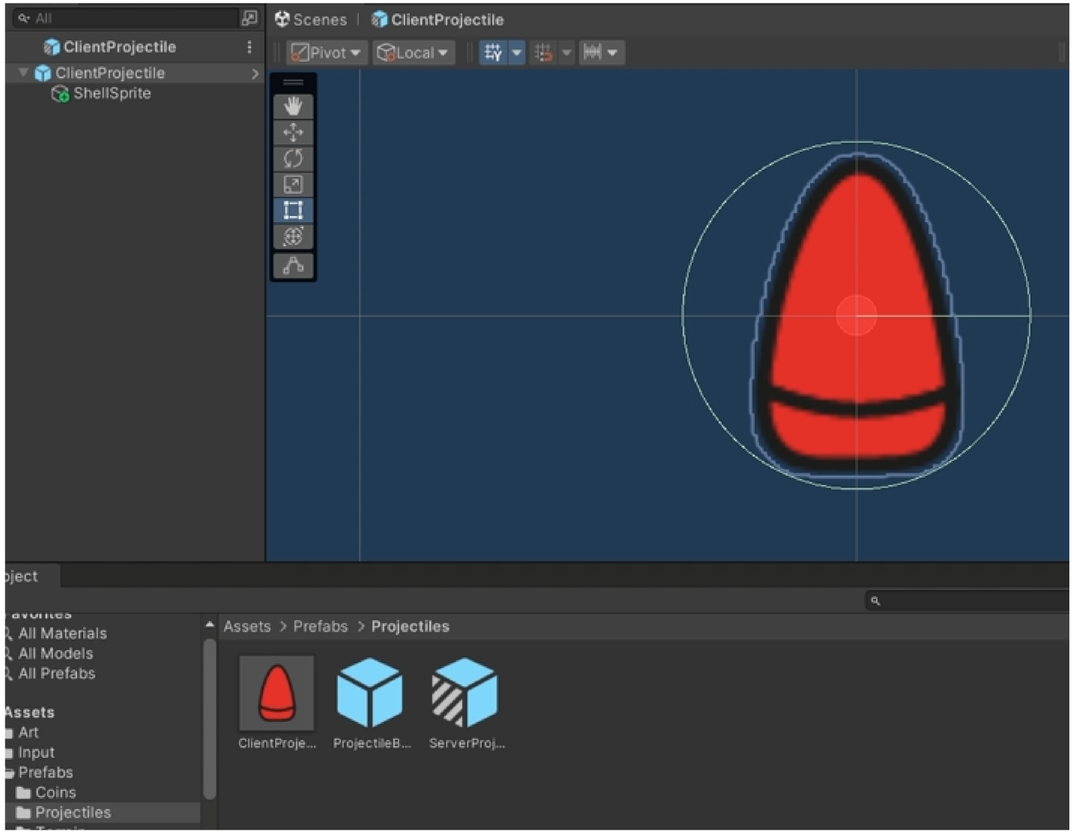

Start Rotation
0
Flip Rotation
Start Color
Gravity Sourc
Gravity Modifi
Simulation Spa
Simulation Spe
Constant
Curve
Random Between Two Constants
Random Between Two Curves

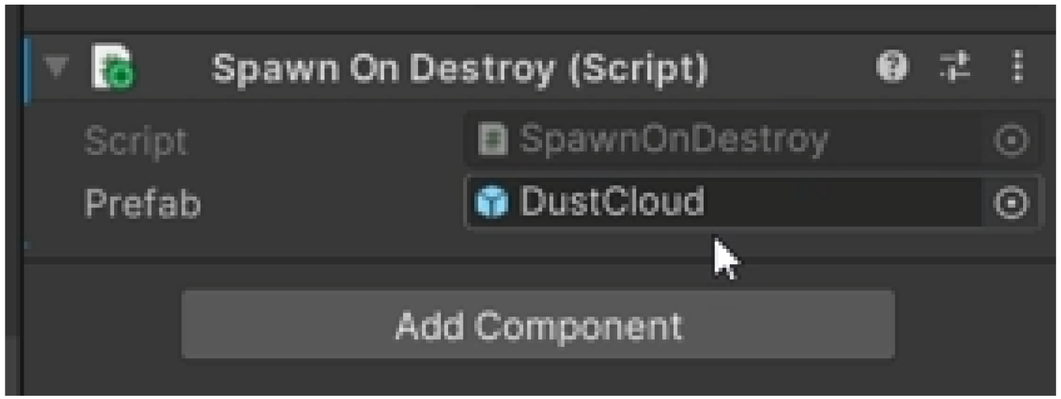

Start Rotation
-180
180


3D Start Size
Start Size
0.6


Stop Action
Destroy
Culling Mode
Automatic
Ring Buffer Mode
Disabled


DustCloud
All Materials
All Models
All Prefabs
Art
Input
Assets > Prefabs
Coins
Projectiles
Terrain
DustCloud
Player


ClientProjectile
ShellSprite
Scenes
ClientProjectile
Pivot
Local
All Materials
All Models
All Prefabs
Assets
Art
Input
Prefabs
Coins
Projectiles
Assets > Prefabs > Projectiles
ClientProje...
ProjectileB...
ServerProj...


Spawn On Destroy (Script)
Script
SpawnOnDestroy
Prefab
DustCloud
Add Component

```
0 references
7   public class ProjectileLauncher : NetworkBehaviour
```

```
0 references
23  private float timer;
2 references
```

```
57      if (Time.time < (1 / fireRate) + previousFireTime) { return; }
58
59      PrimaryFireServerRpc(projectileSpawnPoint.position, projectileSpawnPoint.up);
60
61      SpawnDummyProjectile(projectileSpawnPoint.position, projectileSpawnPoint.up);
62
63      previousFireTime = Time.time;
```

```
if (timer > 0) { return; }

PrimaryFireServerRpc(projectileSpawnPoint.position, projectileSpawnPoint.up);

SpawnDummyProjectile(projectileSpawnPoint.position, projectileSpawnPoint.up);

timer = 1 / fireRate;
```

```
[Header("Settings")]
[SerializeField] private float projectileSpeed;
[SerializeField] private float fireRate;
[SerializeField] private float muzzleFlashDurat
private bool shouldFire;
//private float previousFireTime;
private float timer;
private float muzzleFlashTimer;
```

```
private void Update()
```

```
        if(!IsOwner) { return; }
        if (timer > 0)
        {
            timer -= Time.deltaTime;
        }
        if(!shouldFire) { return; }
        if(timer>0) { return; }

        PrimaryFireServerRpc(projectileSpawnPoint.position, projectileSpawnPoint.up);
        SpawnDummyProjectile(projectileSpawnPoint.position, projectileSpawnPoint.up);

        timer = 1 / fireRate;

    }
```

# Spending Coins

- Create a **SpendCoins(int)** method in the **CoinWallet** script
- Have the client make sure that they have enough coins before firing
- Also have the server make the same check, then if they do, spend the coins

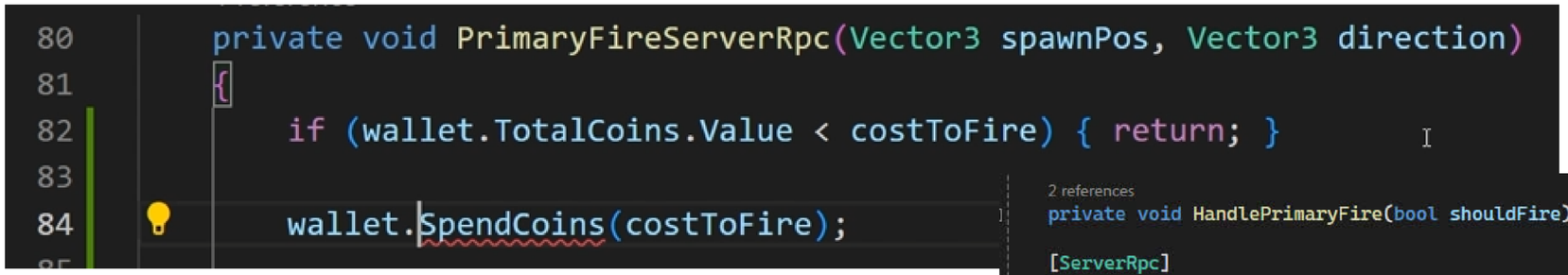
```
public class ProjectileLauncher : NetworkBehaviour
```

```
[Header("References")]
[SerializeField] private InputReader inputReader;
[SerializeField] private CoinWallet wallet;
[SerializeField] private Transform projectileSpawnPoint;
[SerializeField] private GameObject serverProjectilePrefab;
[SerializeField] private GameObject clientProjectilePrefab;
[SerializeField] private GameObject muzzleFlash;
[SerializeField] private Collider2D playerCollider;
```

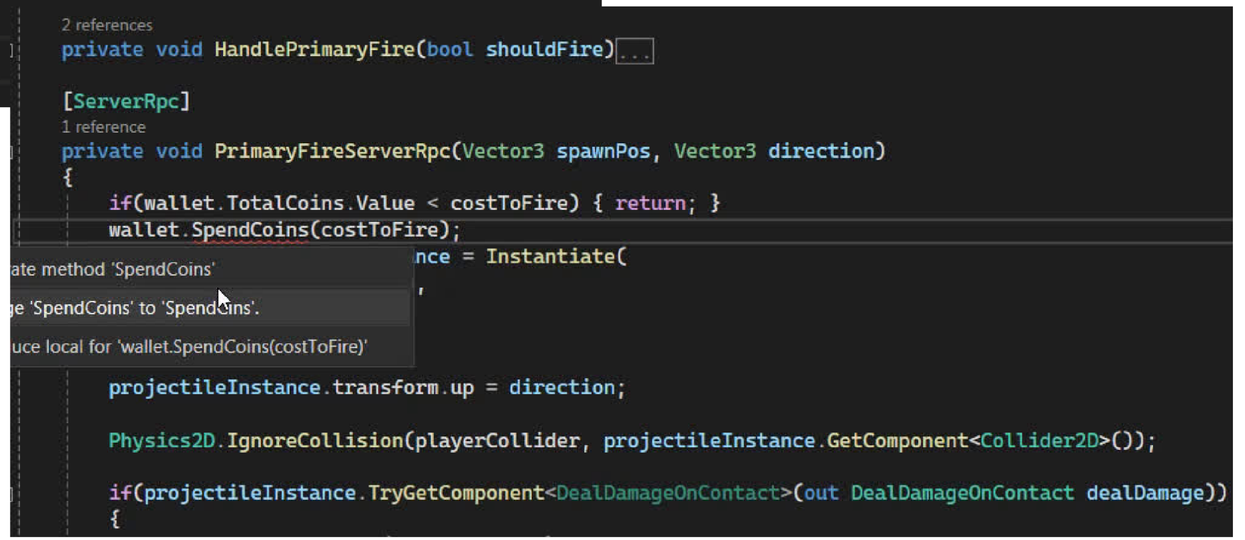
```
[Header("Settings")]
[SerializeField] private float projectileSpeed;
[SerializeField] private float fireRate;
[SerializeField] private float muzzleFlashDuration;
[SerializeField] private int costToFire;
```

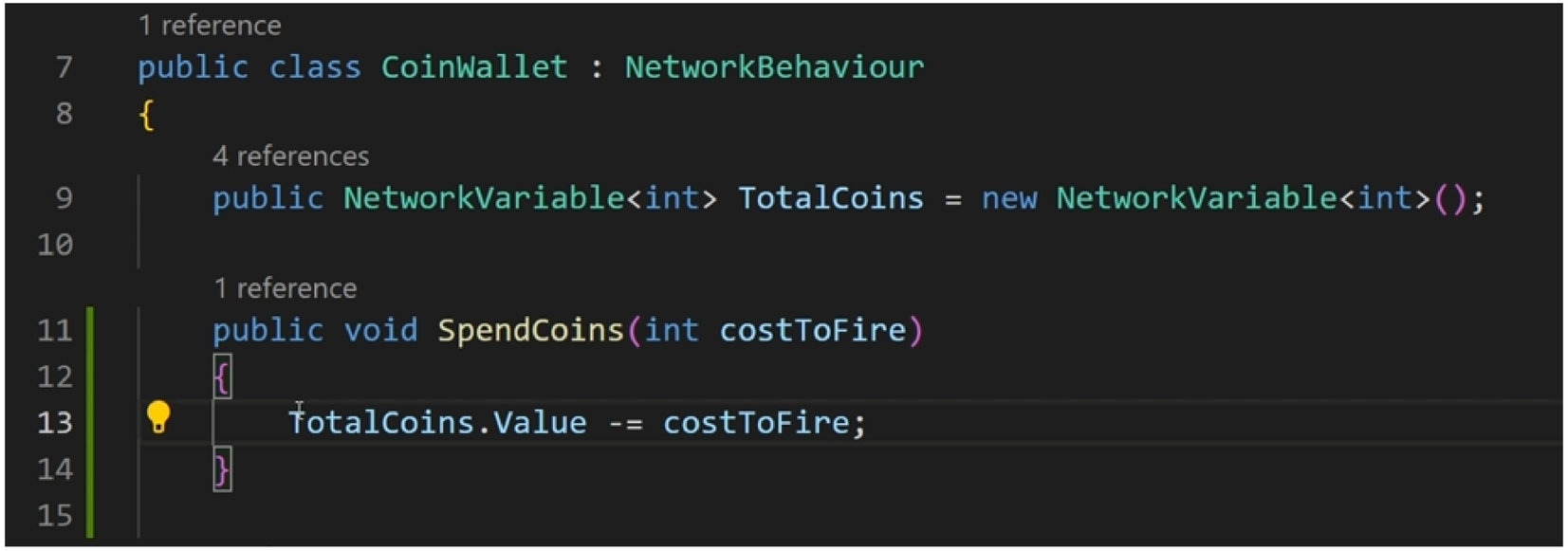
```
void Update()
63          if (timer > 0) { return; }
64
65          if (wallet.TotalCoins.Value < costToFire) { return; }
66
```

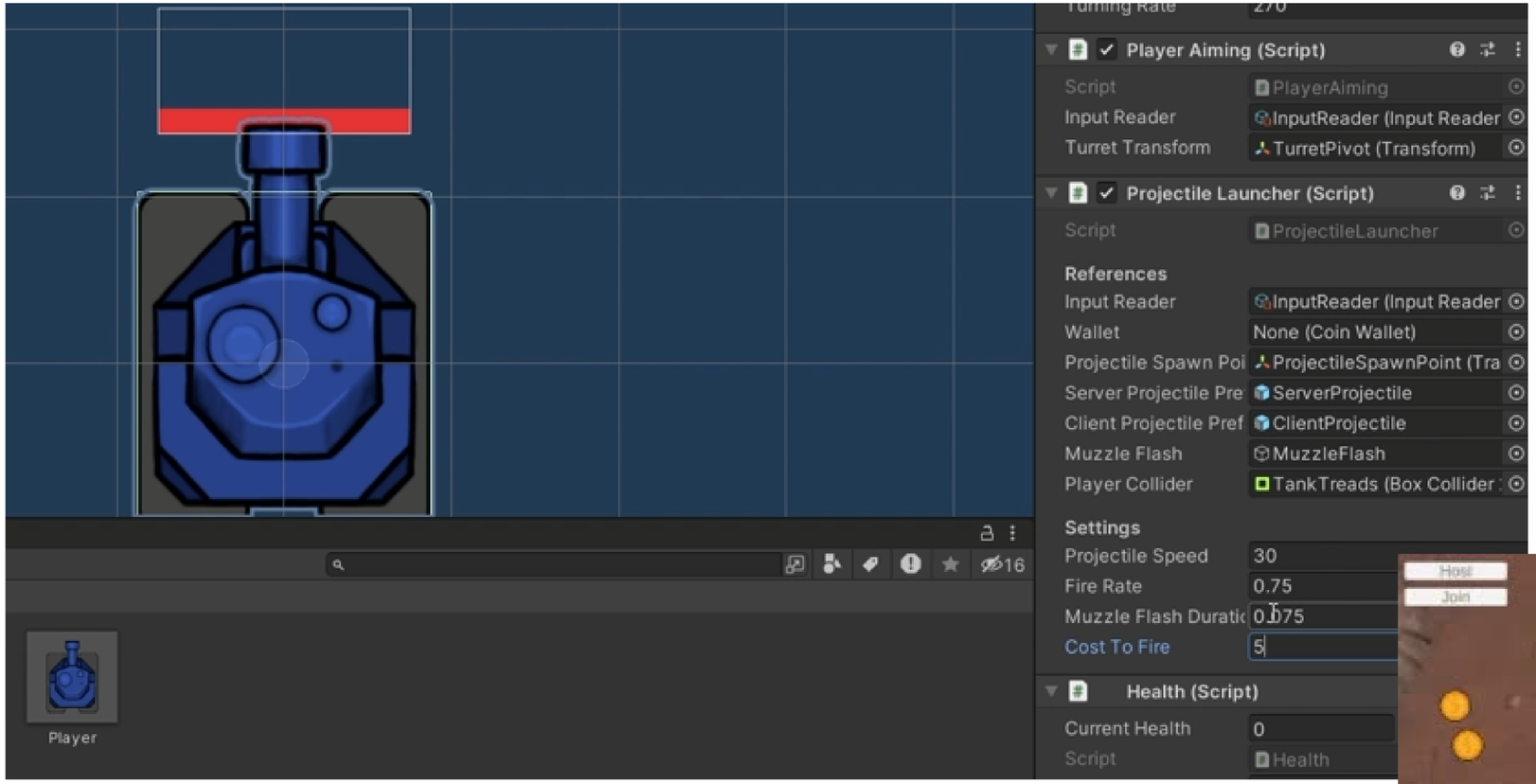
```
80      private void PrimaryFireServerRpc(Vector3 spawnPos, Vector3 direction)
81      {
82          if (wallet.TotalCoins.Value < costToFire) { return; }
```

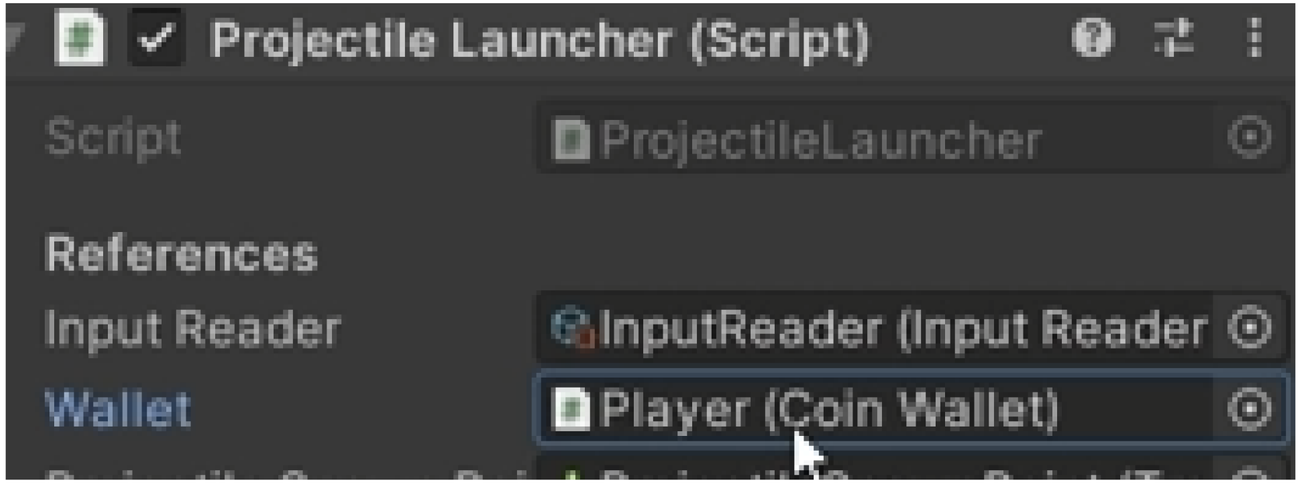

80 private void PrimaryFireServerRpc(Vector3 spawnPos, Vector3 direction)
81 {
82 if (wallet.TotalCoins.Value < costToFire) { return; }
83
84 wallet.SpendCoins(costToFire);


Quick Actions and Refactorings... Ctrl+.


2 references
private void HandlePrimaryFire(bool shouldFire)...
[ServerRpc]
1 reference
private void PrimaryFireServerRpc(Vector3 spawnPos, Vector3 direction)
{
if(wallet.TotalCoins.Value < costToFire) { return; }
wallet.SpendCoins(costToFire);
ate method 'SpendCoins'
e 'SpendCoins' to 'SpendCoins'.
uce local for 'wallet.SpendCoins(costToFire)'
projectileInstance.transform.up = direction;
Physics2D.IgnoreCollision(playerCollider, projectileInstance.GetComponent<Collider2D>());
if(projectileInstance.TryGetComponent<DealDamageOnContact>(out DealDamageOnContact dealDamage))
{

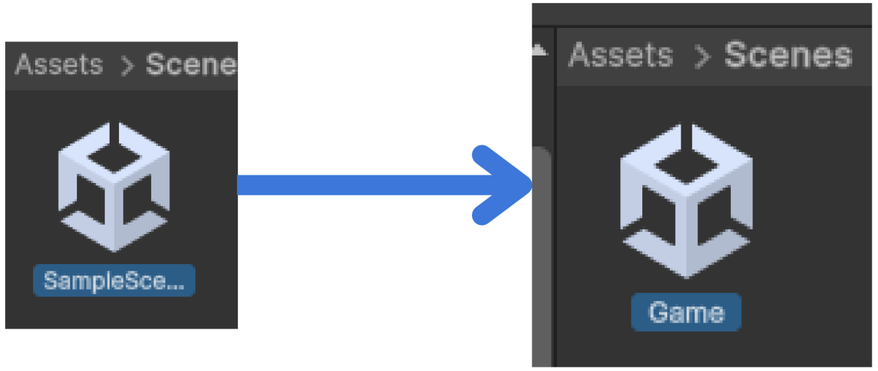

1 reference
public class CoinWallet : NetworkBehaviour
{
4 references
public NetworkVariable<int> TotalCoins = new NetworkVariable<int>();
1 reference
public void SpendCoins(int costToFire)
{
TotalCoins.Value -= costToFire;
}

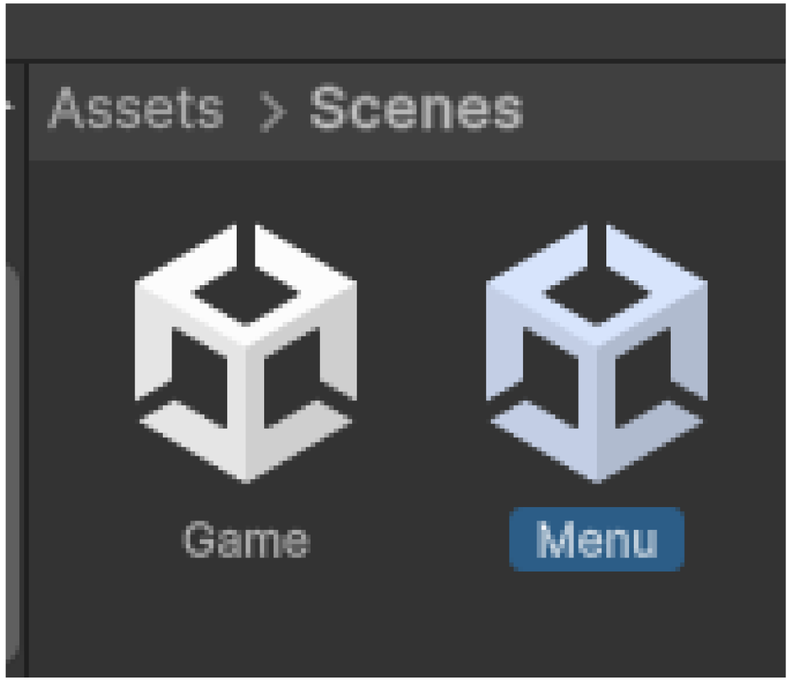

Player Aiming (Script)
Script
PlayerAiming
Input Reader
InputReader (Input Reader
Turret Transform
TurretPivot (Transform)
Projectile Launcher (Script)
Script
ProjectileLauncher
References
Input Reader
InputReader (Input Reader
Wallet
None (Coin Wallet)
Projectile Spawn Poi
ProjectileSpawnPoint (Tra
Server Projectile Pre
ServerProjectile
Client Projectile Pref
ClientProjectile
Muzzle Flash
MuzzleFlash
Player Collider
TankTreads (Box Collider
Settings
Projectile Speed
30
Fire Rate
0.75
Muzzle Flash Durati
0.075
Cost To Fire
5
Health (Script)
Current Health
0
Script
Health
Player

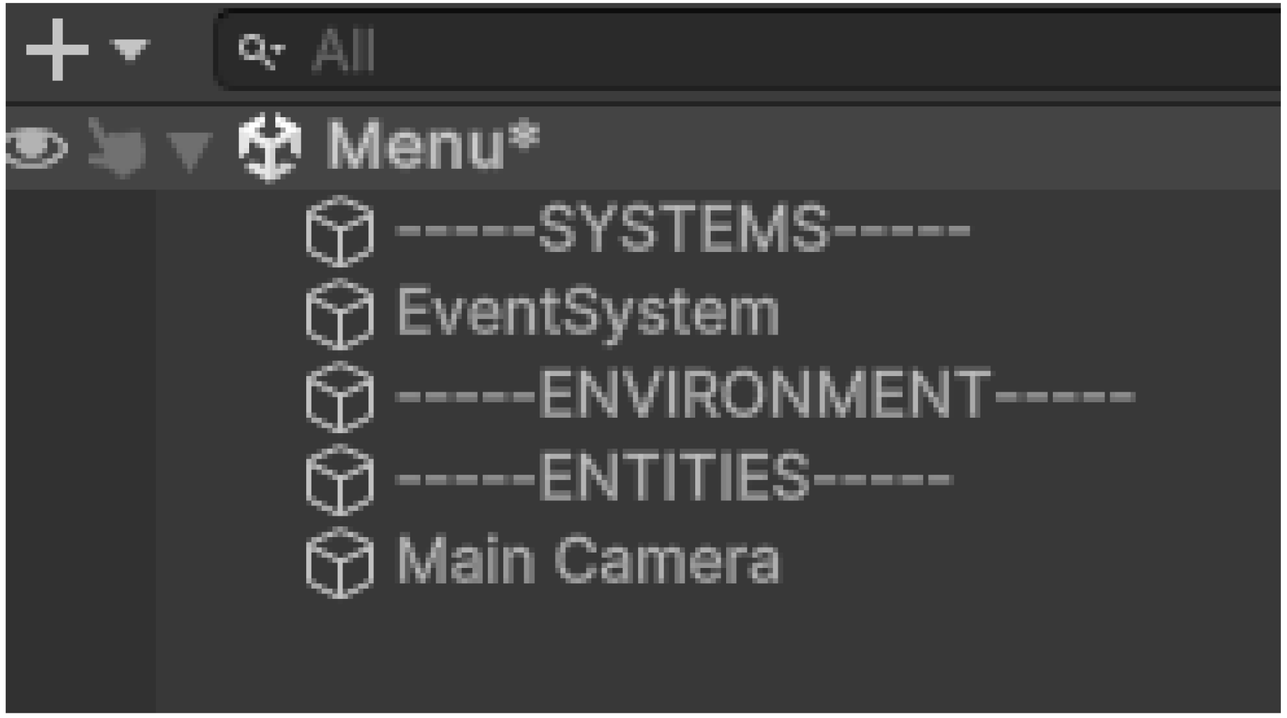

Projectile Launcher (Script)
Script
ProjectileLauncher
References
Input Reader
InputReader (Input Reader
Wallet
Player (Coin Wallet)

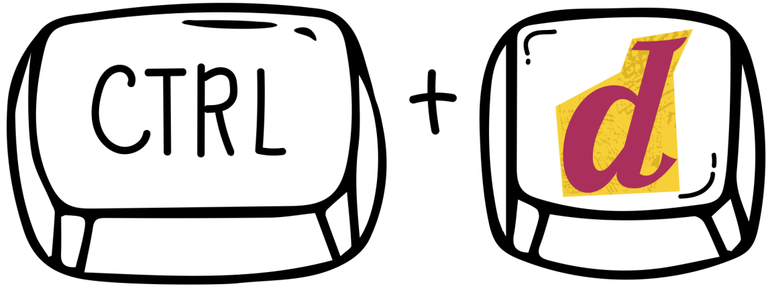

Host
Join

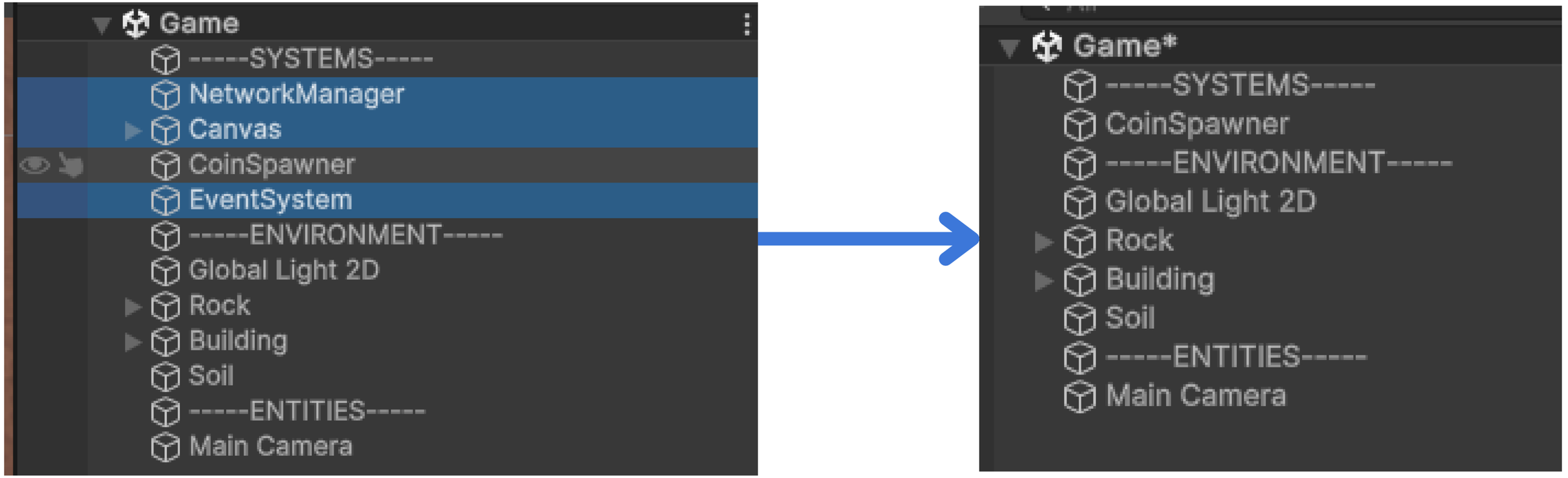
# Setup Main Menu

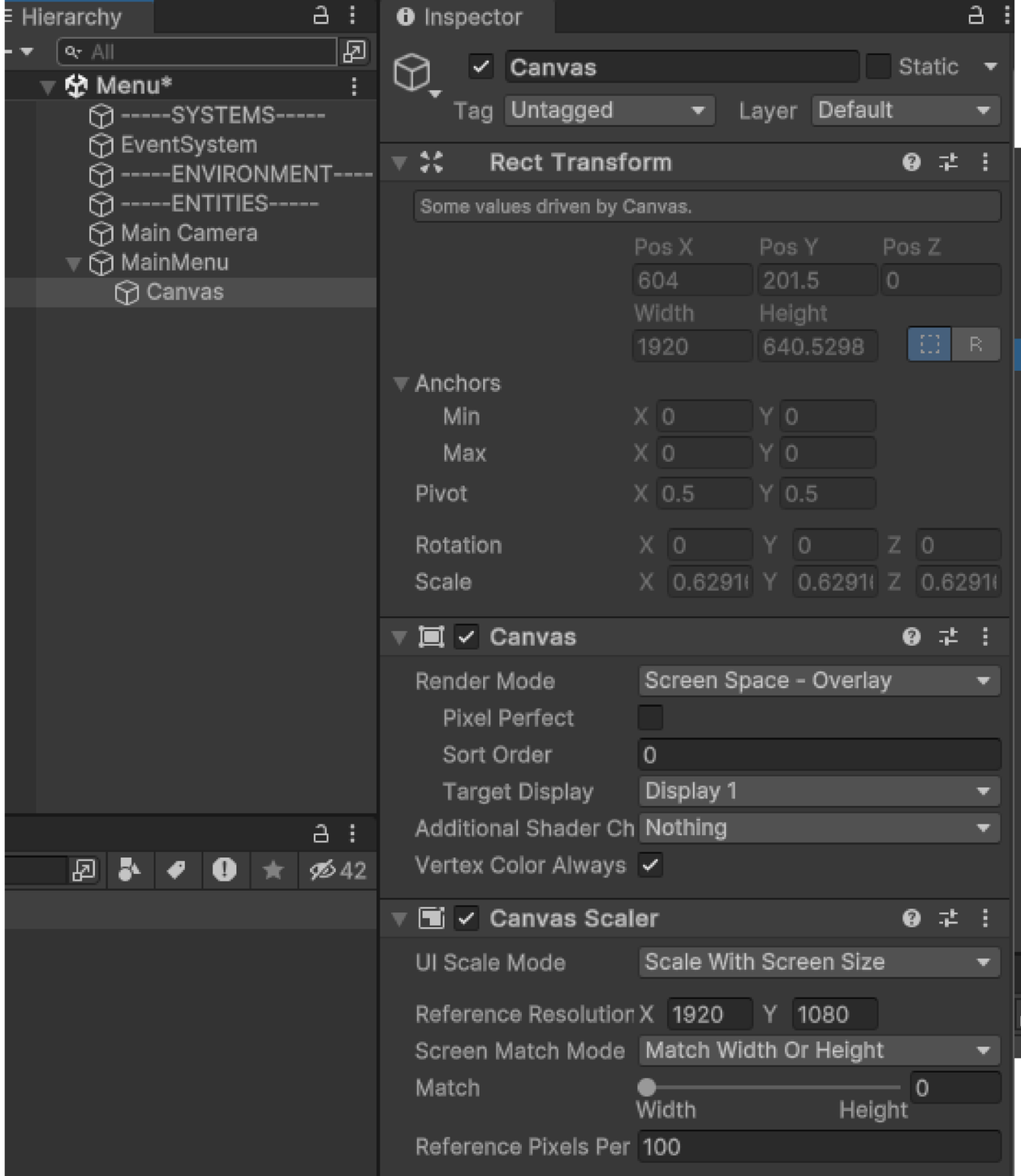

Assets > Scene
SampleSce...
Assets > Scenes
Game

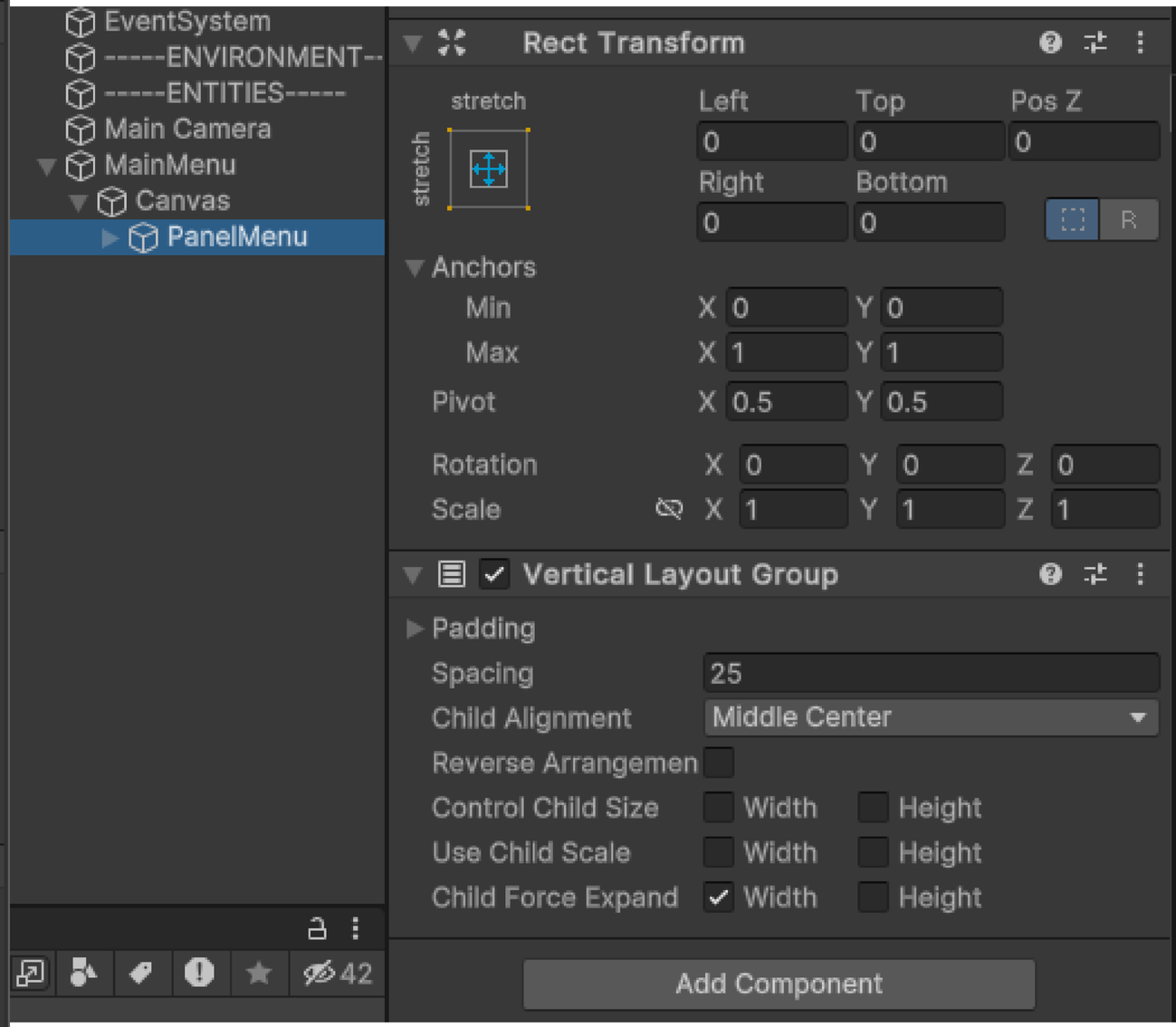

Assets > Scenes
Game
Menu

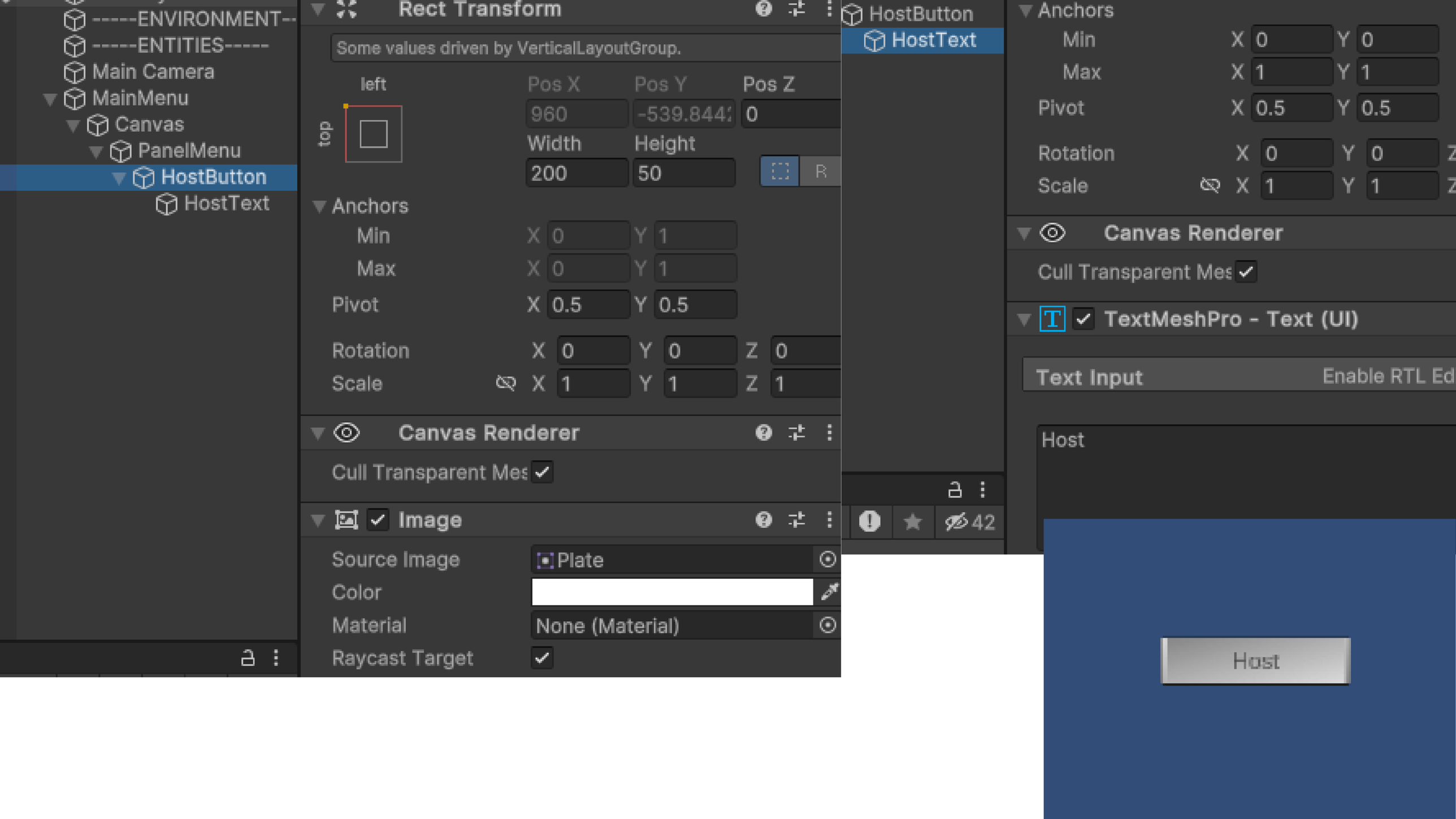

All
Menu*
-----SYSTEMS-----
EventSystem
-----ENVIRONMENT-----
-----ENTITIES-----
Main Camera

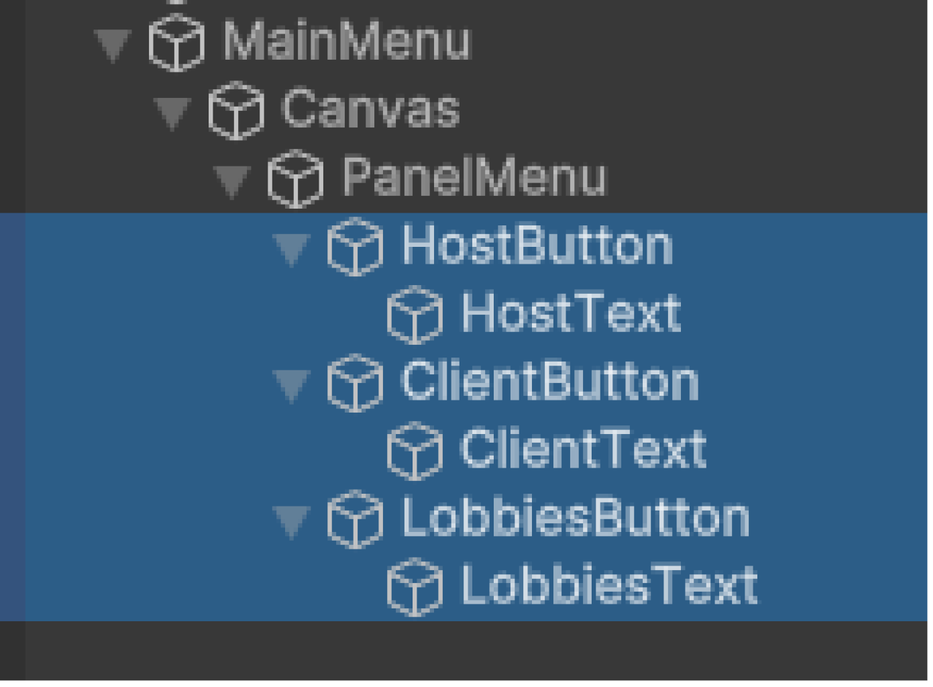

CTRL
+
d

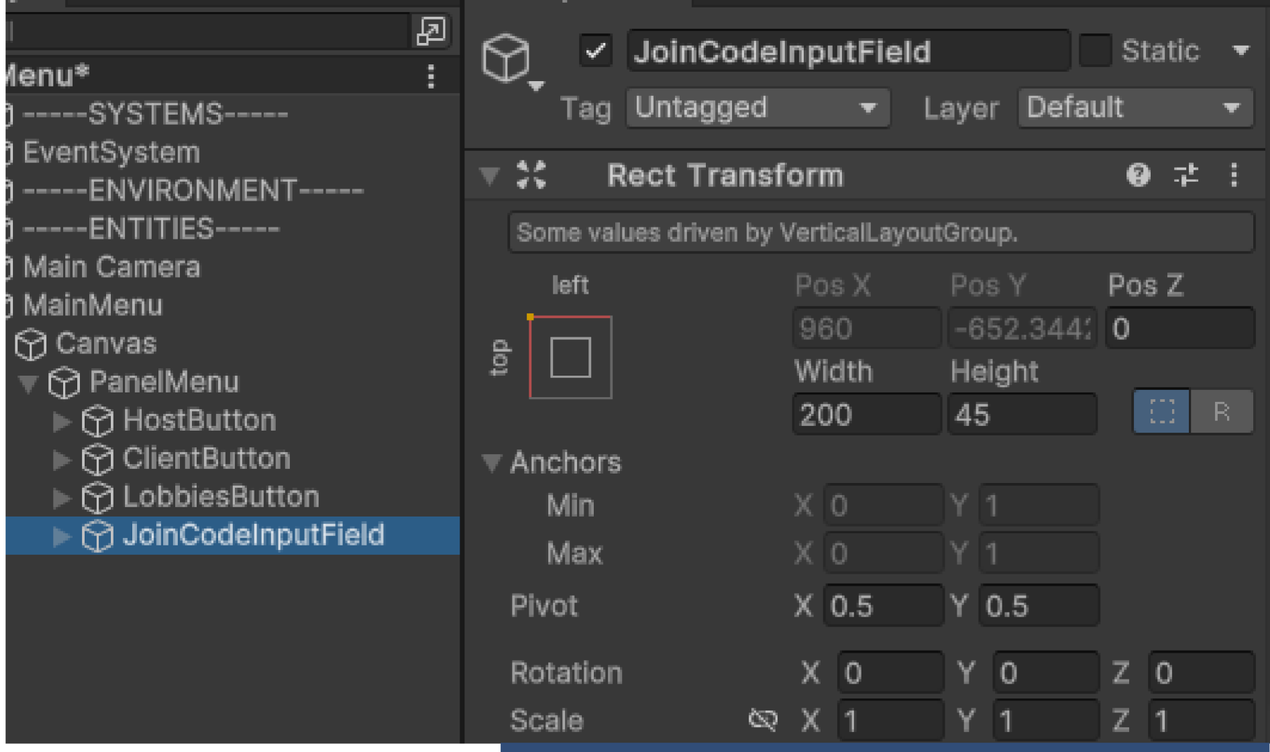

Game
-----SYSTEMS-----
NetworkManager
Canvas
CoinSpawner
EventSystem
-----ENVIRONMENT-----
Global Light 2D
Rock
Building
Soil
-----ENTITIES-----
Main Camera
Game*
-----SYSTEMS-----
CoinSpawner
-----ENVIRONMENT-----
Global Light 2D
Rock
Building
Soil
-----ENTITIES-----
Main Camera

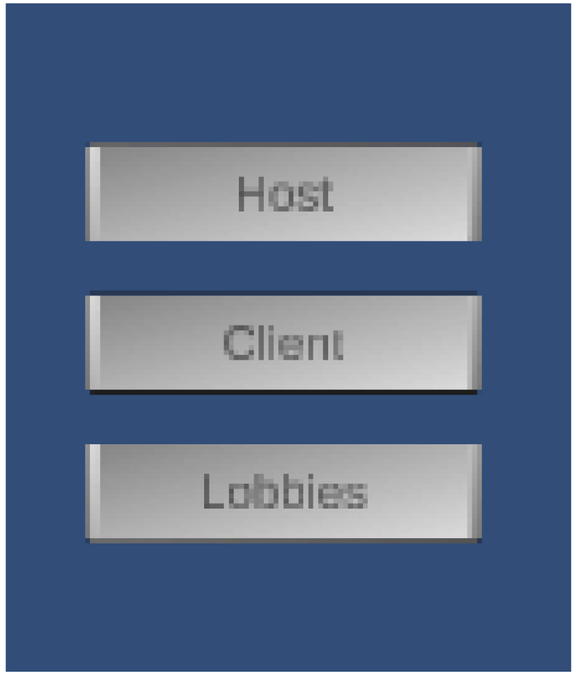

Hierarchy
All
Menu*
-----SYSTEMS-----
EventSystem
-----ENVIRONMENT----
-----ENTITIES-----
Main Camera
MainMenu
Canvas
42
Inspector
Canvas
Static
Tag
Untagged
Layer
Default
Rect Transform
Some values driven by Canvas.
Pos X
Pos Y
Pos Z
604
201.5
0
Width
Height
1920
640.5298
Anchors
Min
Max
Pivot
Rotation
Scale
Canvas
Render Mode
Screen Space - Overlay
Pixel Perfect
Sort Order
0
Target Display
Display 1
Additional Shader Ch
Nothing
Vertex Color Always
Canvas Scaler
UI Scale Mode
Scale With Screen Size
Reference Resolution
1920
1080
Screen Match Mode
Match Width Or Height
Match
Width
Height
Reference Pixels Per
100

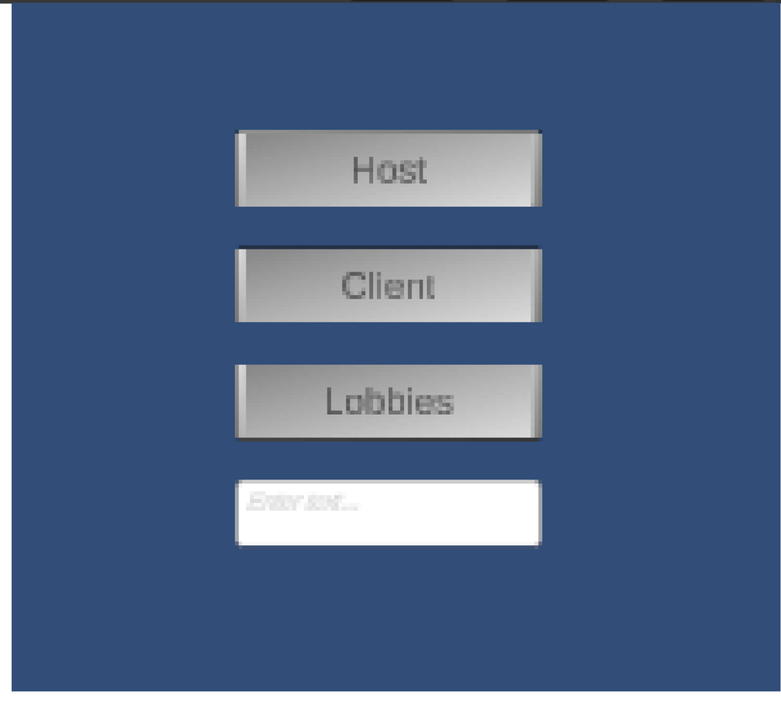

EventSystem
-----ENVIRONMENT--
-----ENTITIES-----
Main Camera
MainMenu
Canvas
PanelMenu
42
Rect Transform
stretch
stretch
Left
Top
Pos Z
0
0
0
Right
Bottom
0
0
Anchors
Min
Max
Pivot
Rotation
Scale
Vertical Layout Group
Padding
Spacing
25
Child Alignment
Middle Center
Reverse Arrangemen
Control Child Size
Width
Height
Use Child Scale
Width
Height
Child Force Expand
Width
Height
Add Component

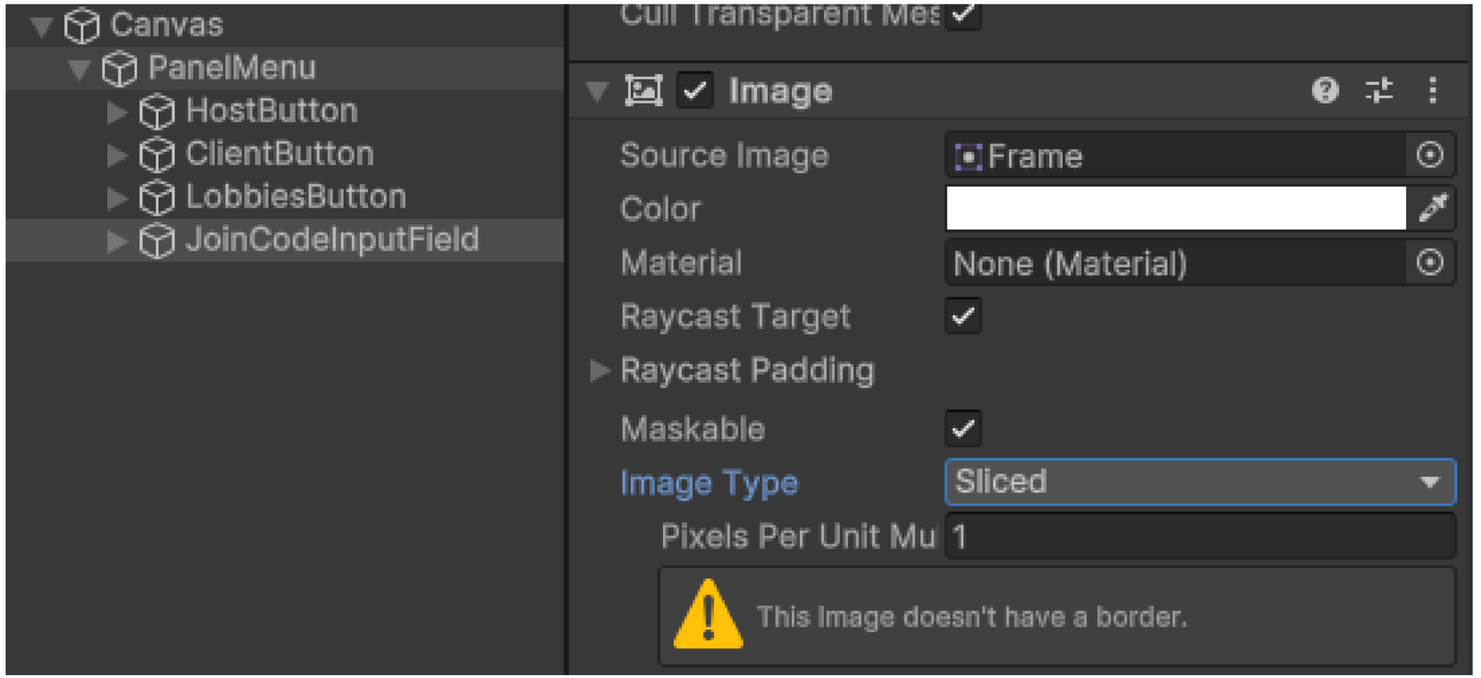

-----ENVIRONMENT--
-----ENTITIES-----
Main Camera
MainMenu
Canvas
PanelMenu
HostButton
HostText
Rect Transform
Some values driven by VerticalLayoutGroup.
left
top
Pos X
Pos Y
Pos Z
960
-539.844
0
Width
Height
200
50
R
Anchors
Min
X 0
Y 1
Max
X 0
Y 1
Pivot
X 0.5
Y 0.5
Rotation
X 0
Y 0
Z 0
Scale
X 1
Y 1
Z 1
Canvas Renderer
Cull Transparent Mes
Image
Source Image
Plate
Color
Material
None (Material)
Raycast Target
HostButton
HostText
42
Anchors
Min
X 0
Y 0
Max
X 1
Y 1
Pivot
X 0.5
Y 0.5
Rotation
X 0
Y 0
Scale
X 1
Y 1
Canvas Renderer
Cull Transparent Mes
TextMeshPro - Text (UI)
Text Input
Enable RTL Ed
Host
Host

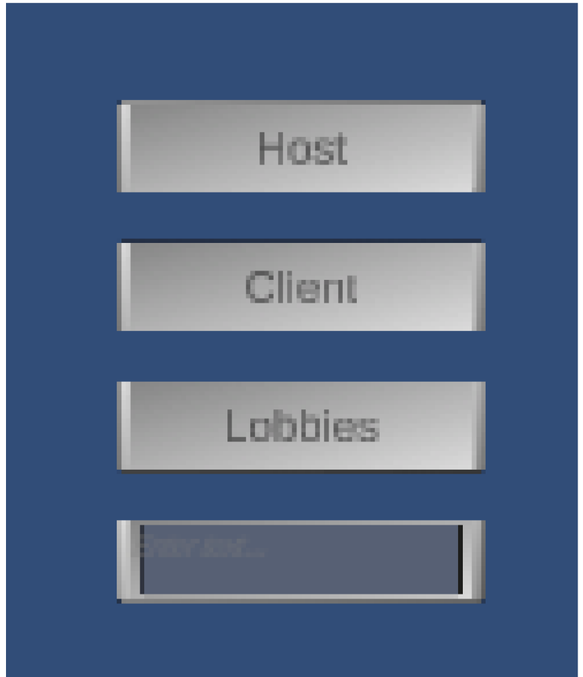

MainMenu
Canvas
PanelMenu
HostButton
HostText
ClientButton
ClientText
LobbiesButton
LobbiesText

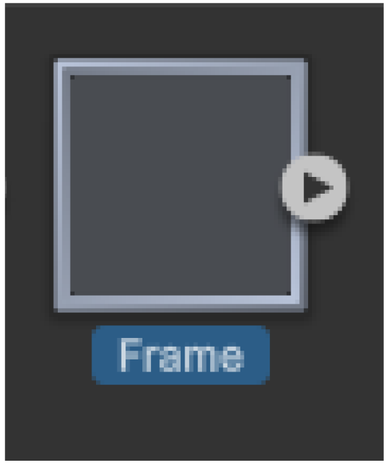

Menu*
-----SYSTEMS-----
EventSystem
-----ENVIRONMENT-----
-----ENTITIES-----
Main Camera
MainMenu
Canvas
PanelMenu
HostButton
ClientButton
LobbiesButton
JoinCodeInputField
JoinCodeInputField
Static
Tag
Untagged
Layer
Default
Rect Transform
Some values driven by VerticalLayoutGroup.
left
top
Pos X
Pos Y
Pos Z
960
-652.344
0
Width
Height
200
45
Anchors
Min
X 0
Y 1
Max
X 0
Y 1
Pivot
X 0.5
Y 0.5
Rotation
X 0
Y 0
Z 0
Scale
X 1
Y 1
Z 1


Host
Client
Lobbies


Host
Client
Lobbies


Canvas
PanelMenu
HostButton
ClientButton
LobbiesButton
JoinCodeInputField
Cull Transparent Mes
Image
Source Image
Frame
Color
Material
None (Material)
Raycast Target
Raycast Padding
Maskable
Image Type
Sliced
Pixels Per Unit Mu
1
This Image doesn't have a border.


Host
Client
Lobbies




Frame (Texture 2D) Import Settings

Open

Texture Type: Sprite (2D and UI)

Texture Shape: 2D

Sprite Mode: Single

Pixels Per Unit: 500

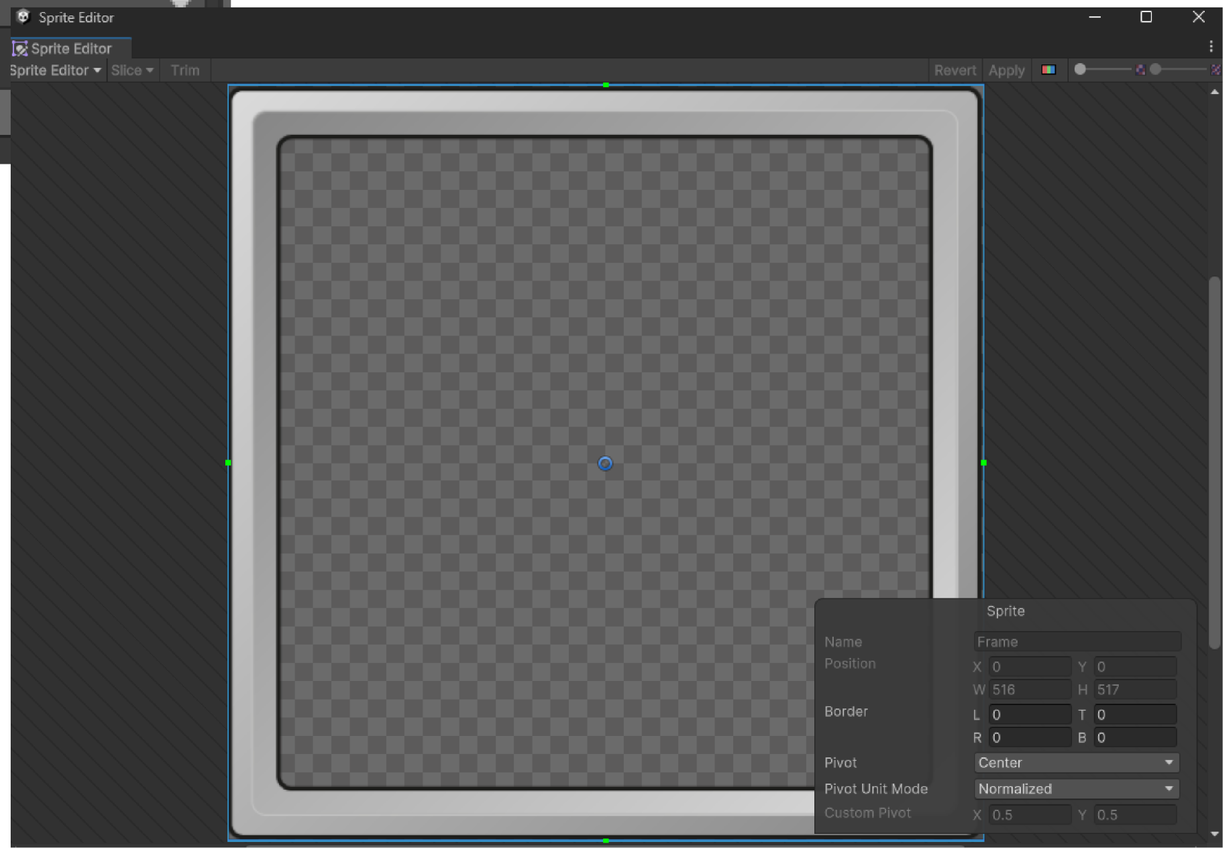
Mesh Type: Tight

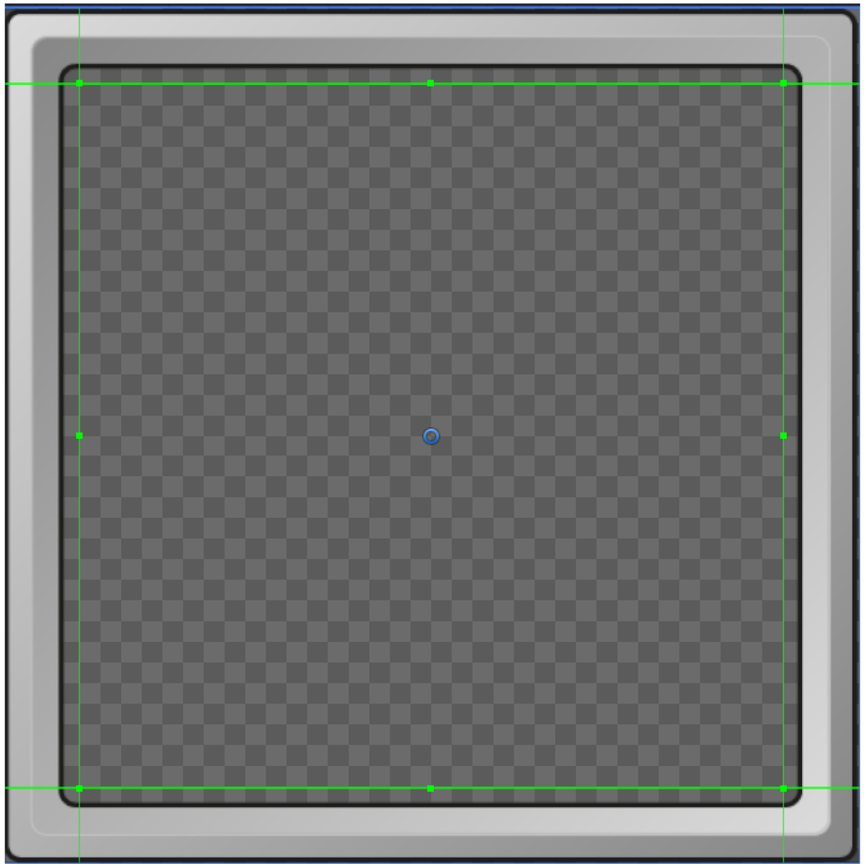
Extrude Edges: 1

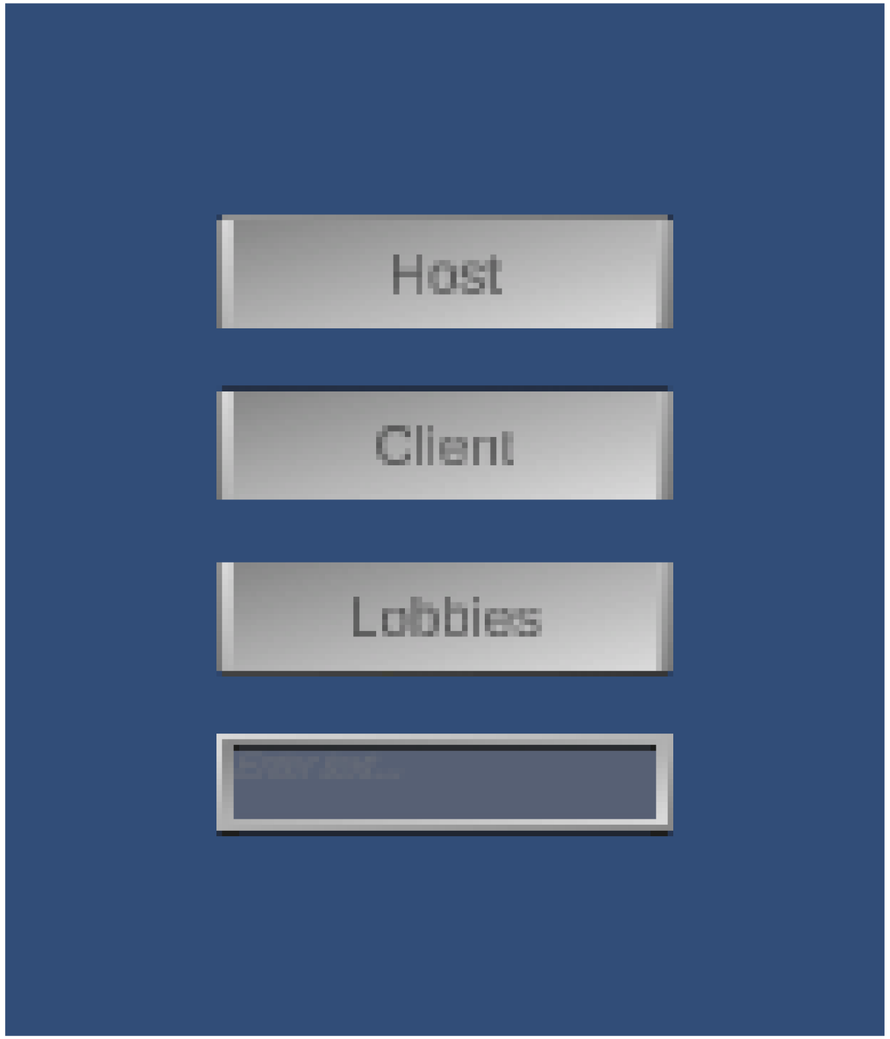
Pivot: Center

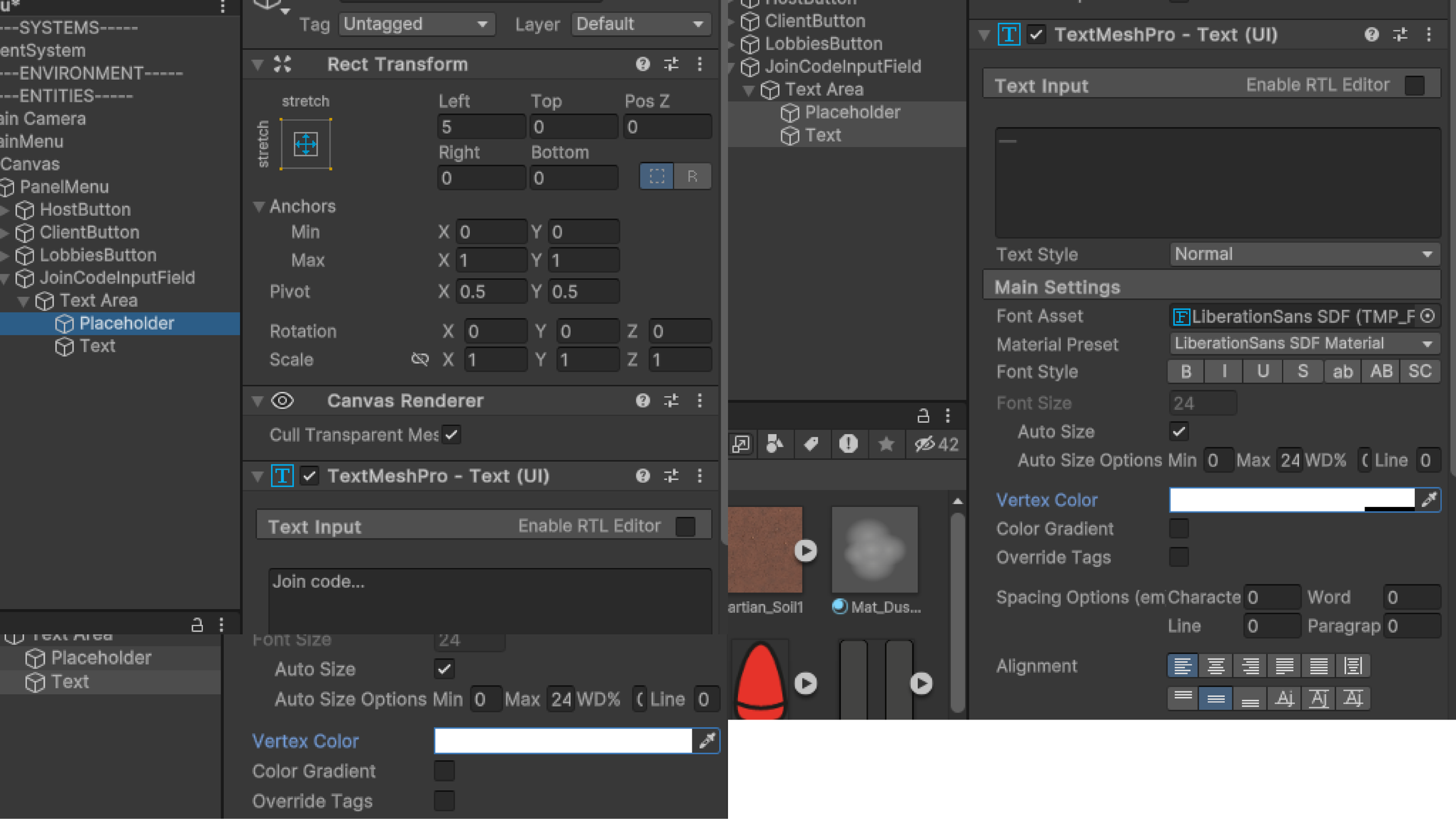
Generate Physics:

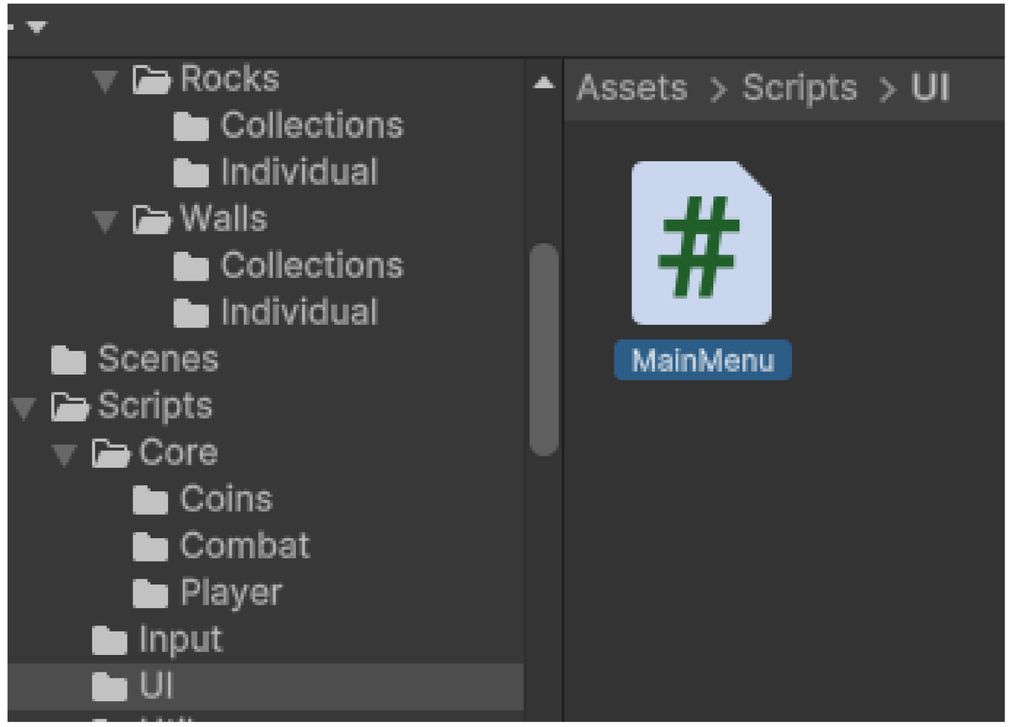
Open Sprite Editor

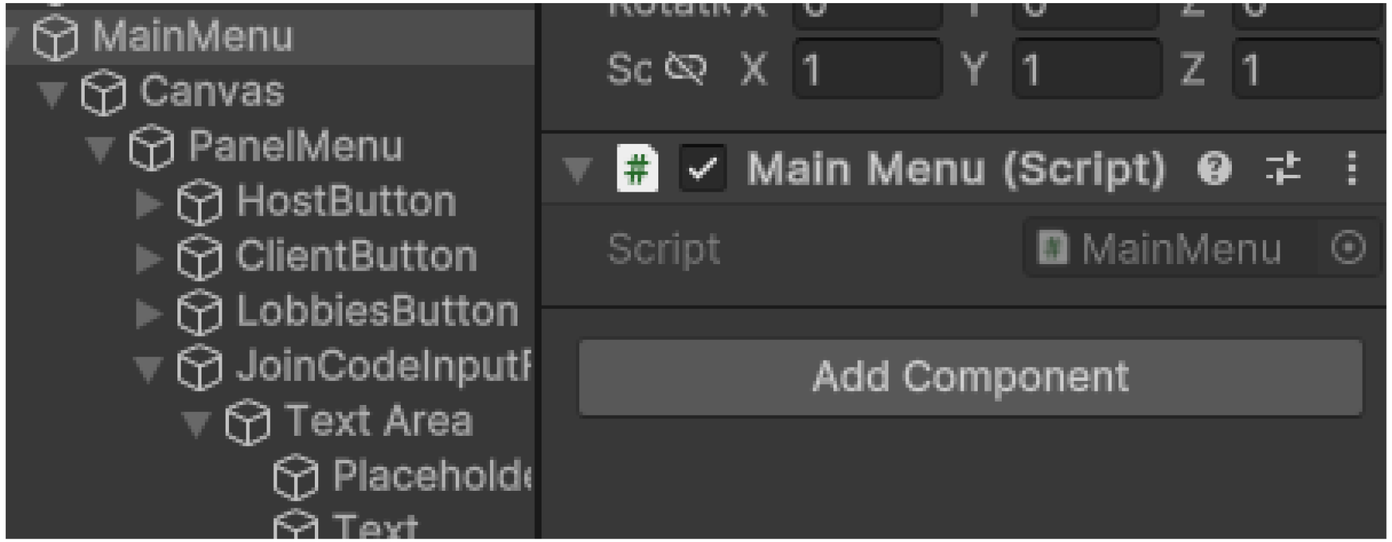


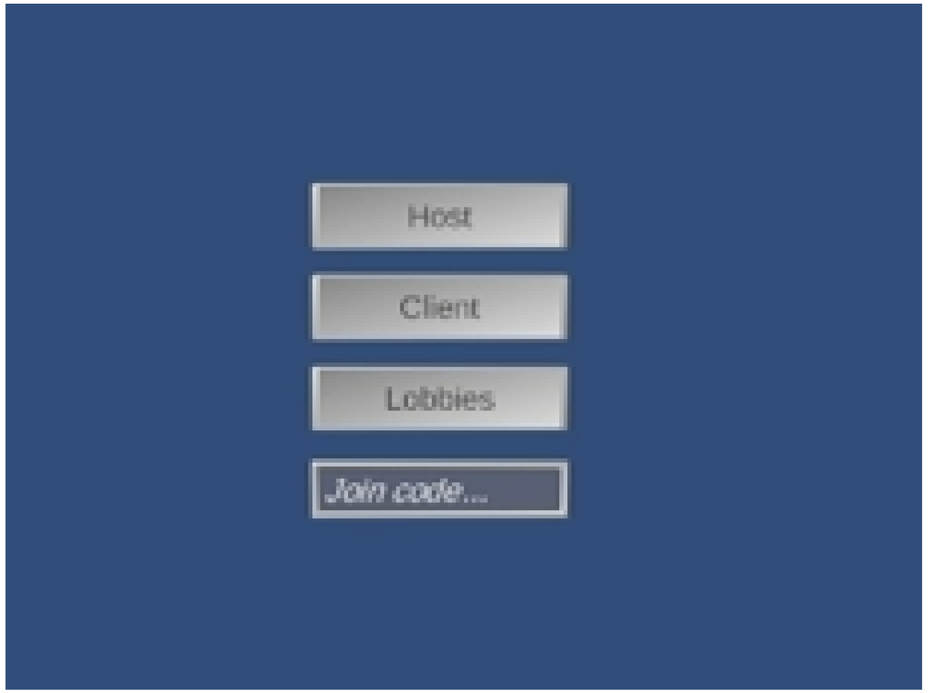



Host
Client
Lobbies


---SYSTEMS-----
entSystem
---ENVIRONMENT-----
---ENTITIES-----
ain Camera
ainMenu
Canvas
PanelMenu
HostButton
ClientButton
LobbiesButton
JoinCodeInputField
Text Area
Placeholder
Text
Tag Untagged
Layer Default
Rect Transform
stretch
Left 5
Top 0
Pos Z 0
Right 0
Bottom 0
Anchors
Min X 0 Y 0
Max X 1 Y 1
Pivot X 0.5 Y 0.5
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Canvas Renderer
Cull Transparent Mes
TextMeshPro - Text (UI)
Text Input
Enable RTL Editor
Join code...
Font Size 24
Auto Size
Auto Size Options Min 0 Max 24 WD% Line 0
Vertex Color
Color Gradient
Override Tags
ClientButton
LobbiesButton
JoinCodeInputField
Text Area
Placeholder
Text
42
artian_Soil1
Mat_Dus...
TextMeshPro - Text (UI)
Text Input
Enable RTL Editor
Text Style Normal
Main Settings
Font Asset LiberationSans SDF (TMP_F
Material Preset LiberationSans SDF Material
Font Style B I U S ab AB SC
Font Size 24
Auto Size
Auto Size Options Min 0 Max 24 WD% Line 0
Vertex Color
Color Gradient
Override Tags
Spacing Options (em Characte 0 Word 0
Line 0 Paragrap 0
Alignment

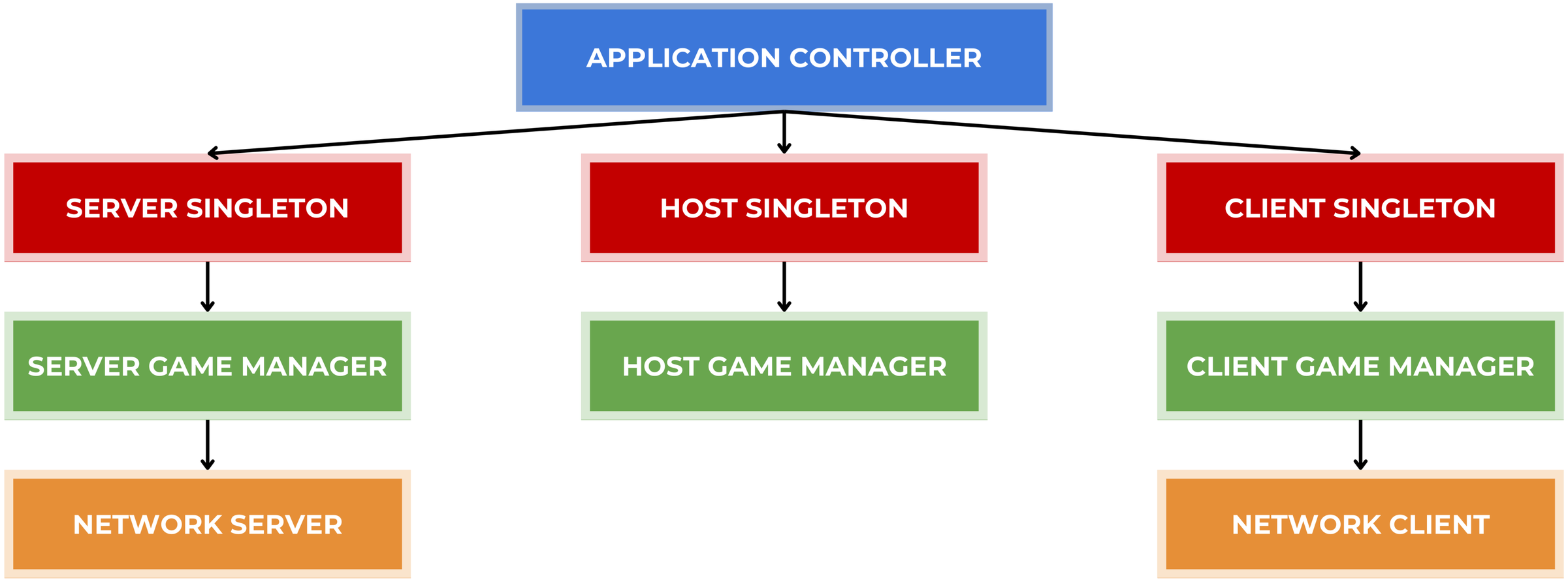

Rocks
Collections
Individual
Walls
Collections
Individual
Scenes
Scripts
Core
Coins
Combat
Player
Input
UI
Assets > Scripts > UI
MainMenu

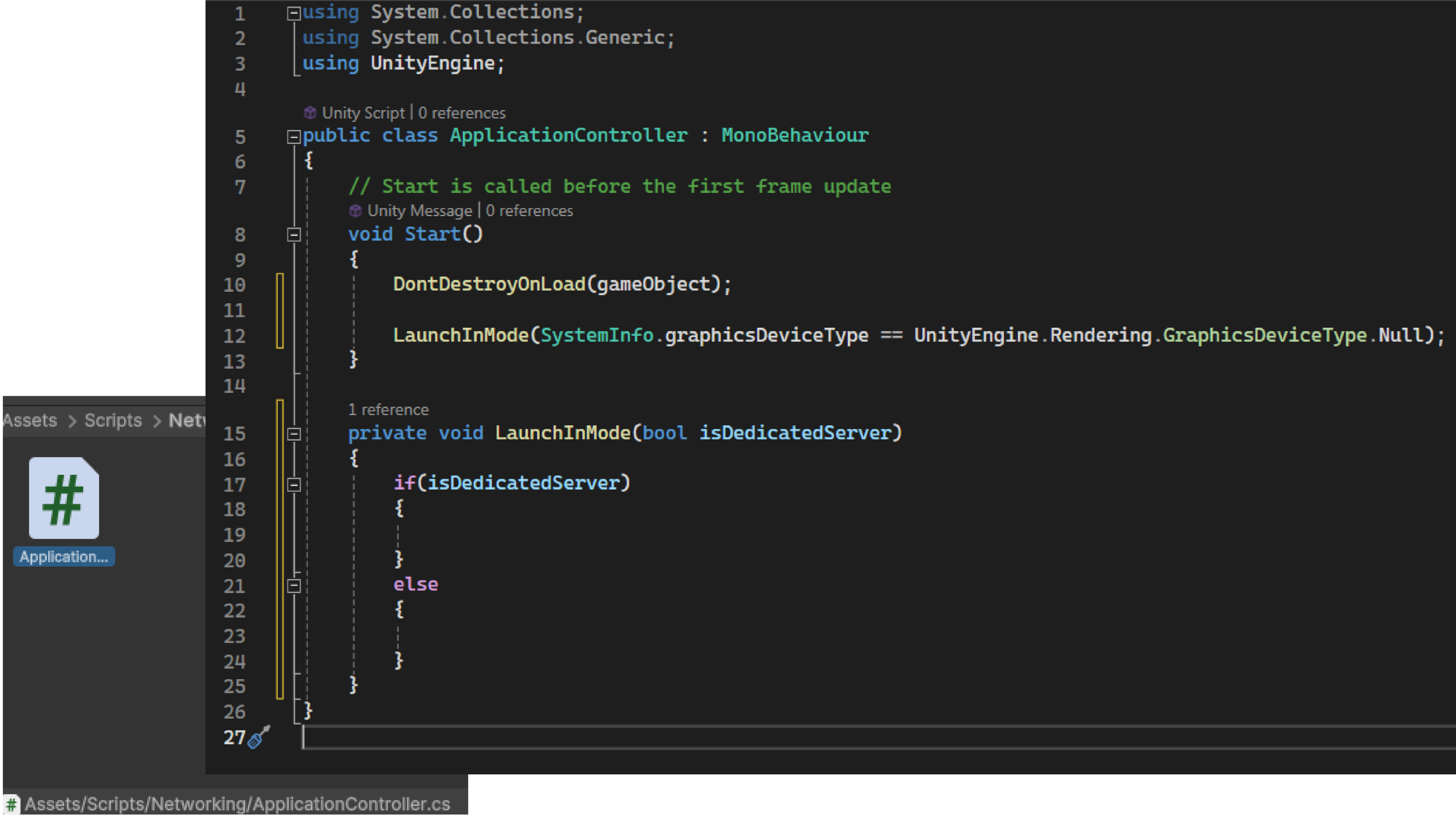

MainMenu
Canvas
PanelMenu
HostButton
ClientButton
LobbiesButton
Text Area
Sc
X 1
Y 1
Z 1
Main Menu (Script)
Script
MainMenu
Add Component


Host
Client
Lobbies
Join code...

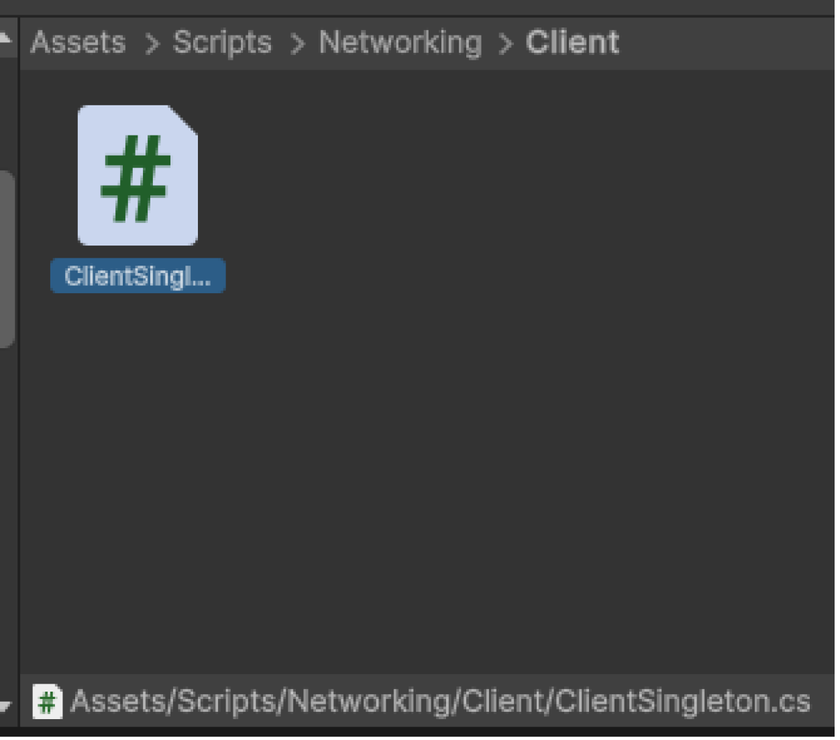
# Application Controller

# Application Structure

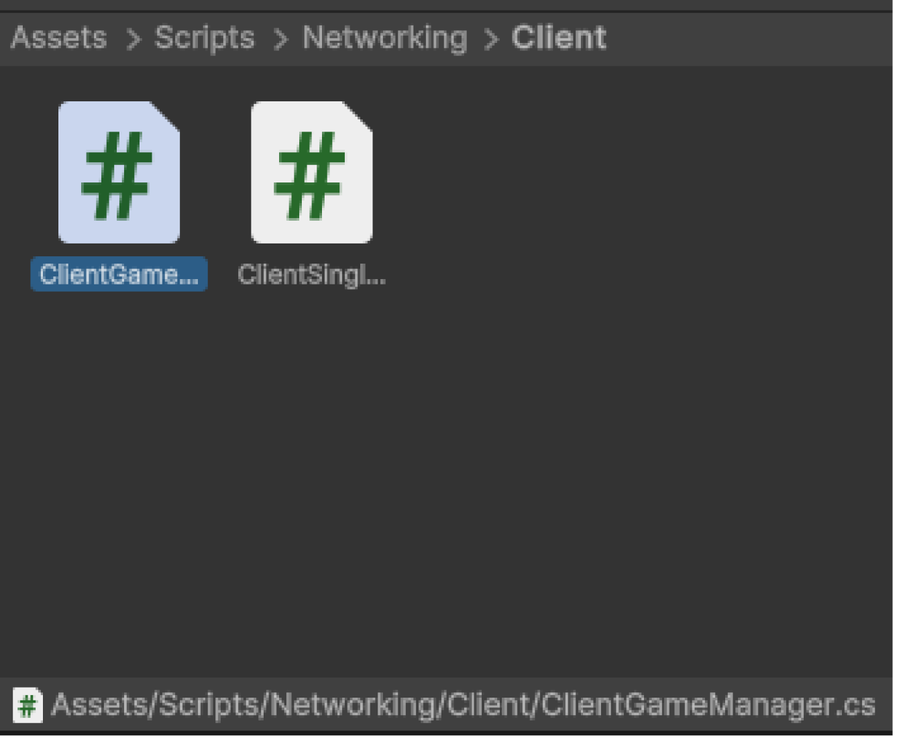





```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

Unity Script | 0 references
public class ApplicationController : MonoBehaviour
{
    // Start is called before the first frame update
    Unity Message | 0 references
    void Start()
    {
        DontDestroyOnLoad(gameObject);

        LaunchInMode(SystemInfo.graphicsDeviceType == UnityEngine.Rendering.GraphicsDeviceType.Null);
    }

    1 reference
    private void LaunchInMode(bool isDedicatedServer)
    {
        if(isDedicatedServer)
        {

        }
        else
        {

        }
    }
}
```




```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
    Unity Script | 3 references
5   public class ClientSingleton : MonoBehaviour
6   {
7       private static ClientSingleton instance;
8
        0 references
9       public static ClientSingleton Instance
10      {
11          get
12          {
13              if(instance == null) { return instance; }
14              instance = FindFirstObjectByType<ClientSingleton>();
15
16              if (instance == null)
17              {
18                  Debug.LogError("No ClientSingleton in the scene!");
19                  return null;
20              }
21              return instance;
22          }
23      }
24
25      // Start is called before the first frame update
        Unity Message | 0 references
26      void Start()
27      {
28          DontDestroyOnLoad(gameObject);
29      }
30
31  }
```




ClientGameManager.cs*

Assembly-CSharp

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
        0 references
6   public class ClientGameManager
7   {
            0 references
8       public async Task InitAsync()
9       {
10          // Authemticate player
11      }
12  }
13
```

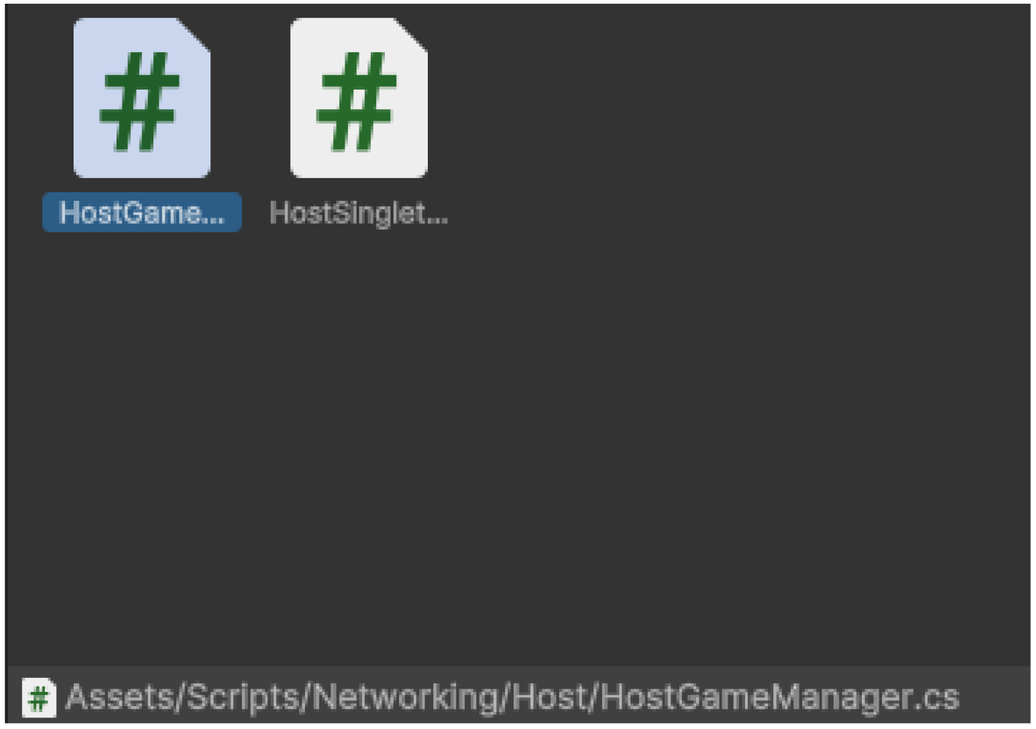
```
public class ClientSingleton : MonoBehaviour
{
    private static ClientSingleton instance;

    private ClientGameManager gameManager;
    0 references
```

```
Unity Message | 0 references
void Start()
{
    DontDestroyOnLoad(gameObject);
}

0 references
public async Task CreateClient()
{
    gameManager = new ClientGameManager();

    await gameManager.InitAsync();
}
```

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
5   public class ApplicationController : MonoBehaviour
6   {
7       [SerializeField] private ClientSingleton clientPrefab;
9
10
        Unity Message | 0 references
11      private async void Start()
12      {
13          DontDestroyOnLoad(gameObject);
14
15          await LaunchInMode(SystemInfo.graphicsDeviceType == UnityEngine.Rendering.GraphicsDeviceType.Null);
16      }
17
        1 reference
18      private async Task LaunchInMode(bool isDedicatedServer)
19      {
20          if(isDedicatedServer)
21          {
22
23          }
24          else
25          {
26              ClientSingleton clientSingleton = Instantiate(clientPrefab);
27
28              await clientSingleton.CreateClient();
29
30              // Go to main menu
31          }
32      }
33  }
```

# Host Singleton & Game Manager

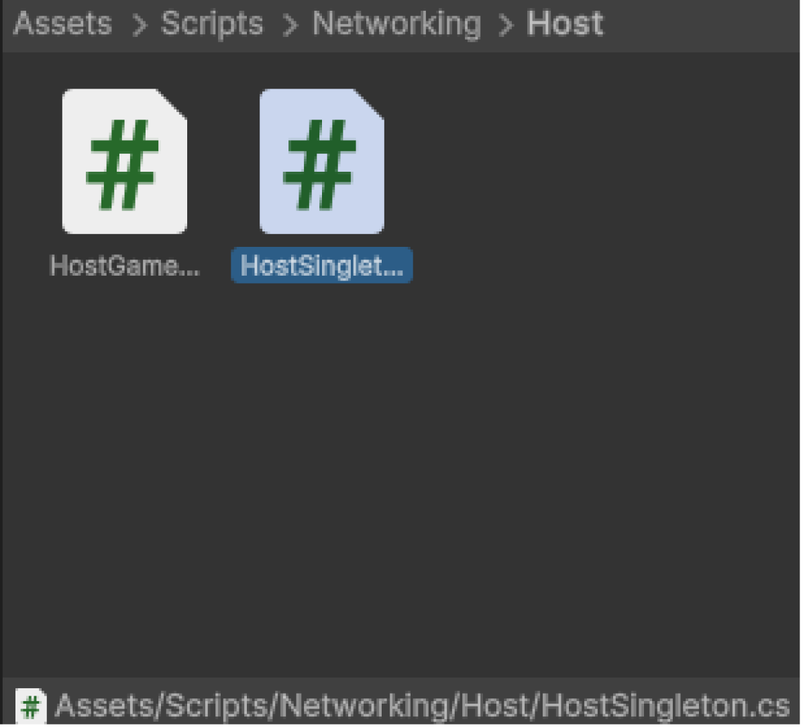
- Duplicate our **ClientSingleton** & **ClientGameManager** scripts
- Place them in their own Host folders following the same structure we used for clients
- Rename any instances of **Client** to **Host**




HostGameManager.cs | ApplicationController.cs | ClientSingleton.cs

Assembly-CSharp

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
    0 references
6   public class HostGameManager
7   {
8
9   }
```




HostSingleton.cs

Assembly-CSharp | HostSingleton

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
    Unity Script | 5 references
6   public class HostSingleton : MonoBehaviour
7   {
8       private static HostSingleton instance;
9
10      private HostGameManager gameManager;
        0 references
11      public static HostSingleton Instance
12      {
13          get
14          {
15              if (instance == null) { return instance; }
16              instance = FindFirstObjectByType<HostSingleton>();
17
18              if (instance == null)
19              {
20                  Debug.LogError("No HostSingleton in the scene!");
21                  return null;
22              }
23              return instance;
24          }
25      }
26
        Unity Message | 0 references
27      void Start()
28      {
29          DontDestroyOnLoad(gameObject);
30      }
31
        1 reference
32      public void CreateHost()
33      {
34          gameManager = new HostGameManager();
35      }
36  }
```

```
3   using System.Threading.Tasks;
4   using UnityEngine;
5
    public class ApplicationController : MonoBehaviour
    {
        [SerializeField] private ClientSingleton clientPrefab;
        [SerializeField] private HostSingleton hostPrefab;

        Unity Message | 0 references
11      private async void Start()
12      {
13          DontDestroyOnLoad(gameObject);
14          await LaunchInMode(SystemInfo.graphicsDeviceType == UnityEngine.Rendering.GraphicsDeviceType.Null);
15      }
16
        1 reference
17      private async Task LaunchInMode(bool isDedicatedServer)
18      {
19          if(isDedicatedServer)
20          {
21
22          }
23          else
24          {
25              ClientSingleton clientSingleton = Instantiate(clientPrefab);
26              await clientSingleton.CreateClient();
27
28              HostSingleton hostSingleton = Instantiate(hostPrefab);
29              hostSingleton.CreateHost();
30
31              // Go to main menu
32          }
33      }
34  }
35
```

# Application Controller (Complete) : Script ApplicationController

```
using System.Collections;
using System.Collections.Generic;
using System.Threading.Tasks;
using UnityEngine;

Unity Script | 0 references
public class ApplicationController : MonoBehaviour
{
    [SerializeField] private ClientSingleton clientPrefab;
    [SerializeField] private HostSingleton hostPrefab;

    Unity Message | 0 references
    private async void Start()
    {
        DontDestroyOnLoad(gameObject);
        await LaunchInMode(SystemInfo.graphicsDeviceType == UnityEngine.Rendering.GraphicsDeviceType.Null);
    }

    1 reference
    private async Task LaunchInMode(bool isDedicatedServer)
    {
        if(isDedicatedServer)
        {

        }
        else
        {
            ClientSingleton clientSingleton = Instantiate(clientPrefab);
            await clientSingleton.CreateClient();

            HostSingleton hostSingleton = Instantiate(hostPrefab);
            hostSingleton.CreateHost();

            // Go to main menu
        }
    }
}
```

Code Recheck

# Application Controller (Complete) : Script **ClientGameManager**

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
    Unity Script | 5 references
6   public class ClientSingleton : MonoBehaviour
7   {
8       private static ClientSingleton instance;
9       private ClientGameManager gameManager;
        0 references
10      public static ClientSingleton Instance
11      {
12          get
13          {
14              if(instance == null) { return instance; }
15              instance = FindFirstObjectByType<ClientSingleton>();
16
17              if (instance == null)
18              {
19                  Debug.LogError("No ClientSingleton in the scene!");
20                  return null;
21              }
22              return instance;
23          }
24      }
25
        Unity Message | 0 references
26      void Start()
27      {
28          DontDestroyOnLoad(gameObject);
29      }
30
        1 reference
31      public async Task CreateClient()
32      {
33          gameManager = new ClientGameManager();
34
35          await gameManager.InitAsync();
36      }
```

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
        2 references
6   public class ClientGameManager
7   {
        1 reference
8       public async Task InitAsync()
9       {
10          // Authemticate player
11      }
12  }
```

Script **ClientSingleton**

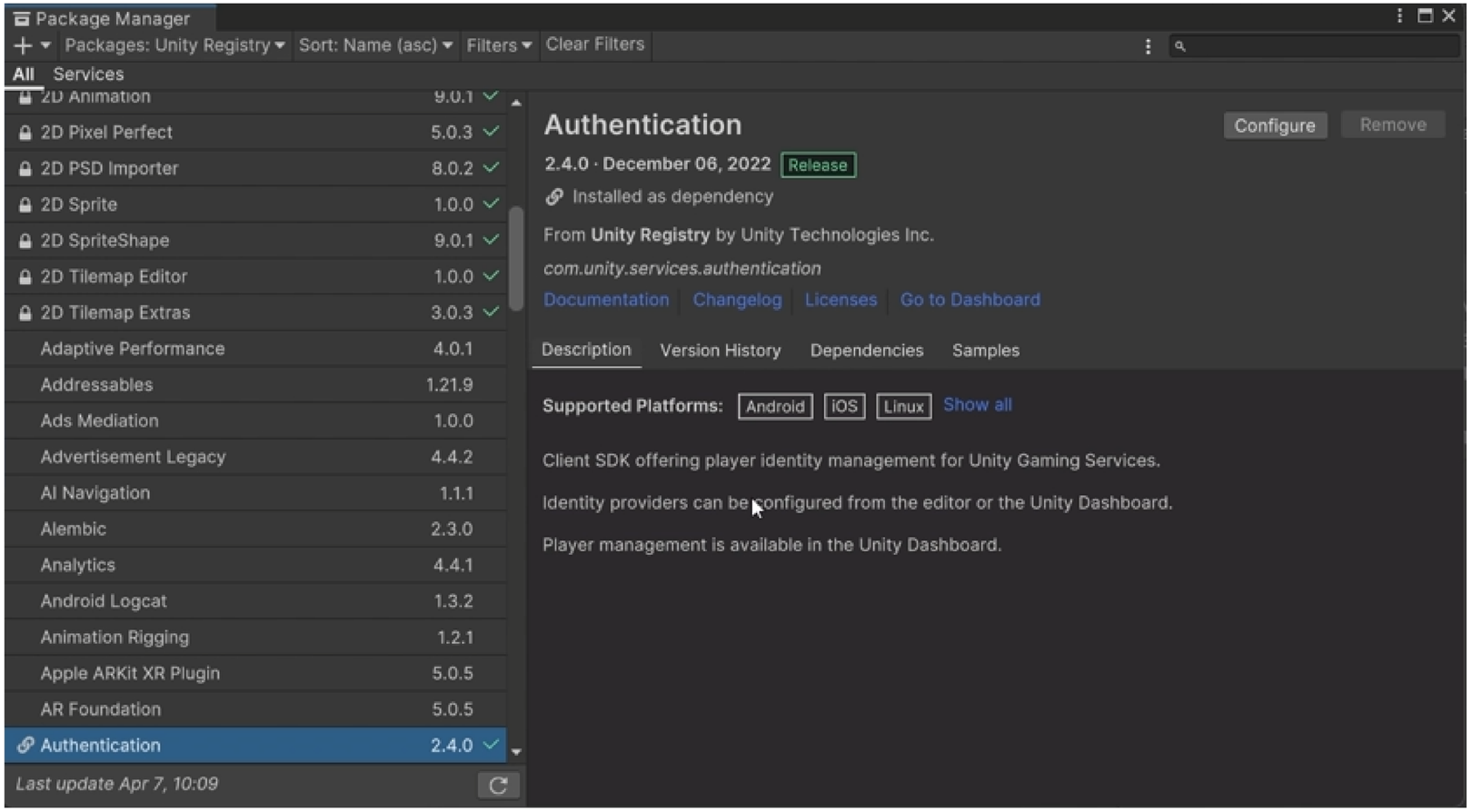
# Application Controller (Complete) : Script **HostGameManager**

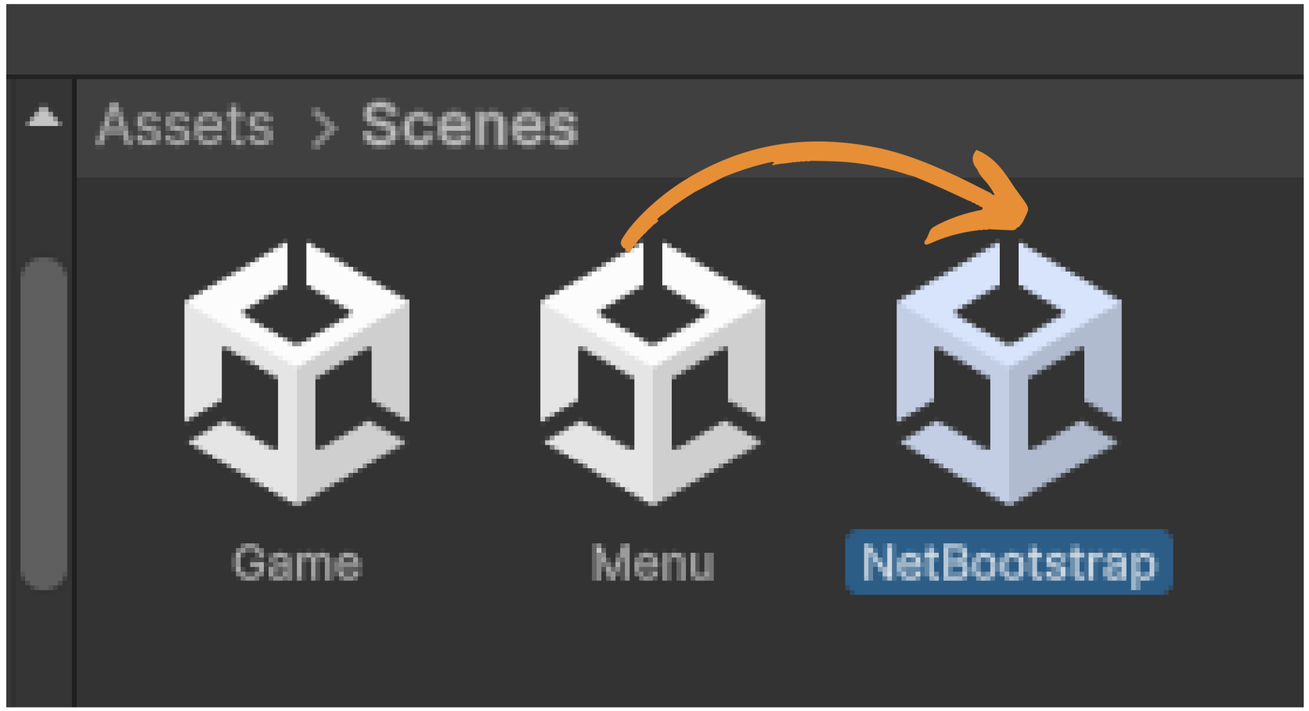
```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
    Unity Script | 5 references
6   public class HostSingleton : MonoBehaviour
7   {
8       private static HostSingleton instance;
9
10      private HostGameManager gameManager;
        0 references
11      public static HostSingleton Instance
12      {
13          get
14          {
15              if (instance == null) { return instance; }
16              instance = FindFirstObjectByType<HostSingleton>();
17
18              if (instance == null)
19              {
20                  Debug.LogError("No HostSingleton in the scene!");
21                  return null;
22              }
23              return instance;
24          }
25      }
26
        Unity Message | 0 references
27      void Start()
28      {
29          DontDestroyOnLoad(gameObject);
30      }
31
        1 reference
32      public void CreateHost()
33      {
34          gameManager = new HostGameManager();
35      }
36  }
```

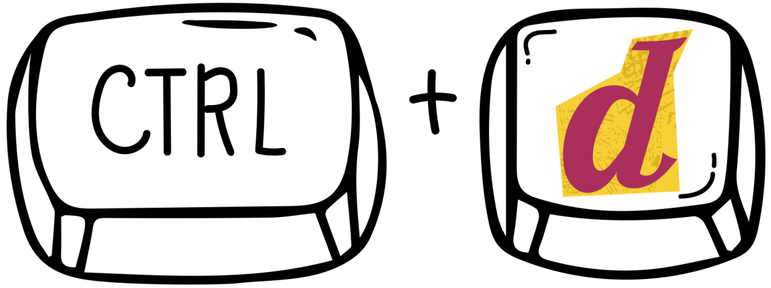
```
1   using System.Collections;
2   using System.Collections.Generic;
3   using System.Threading.Tasks;
4   using UnityEngine;
5
        2 references
6   public class HostGameManager
7   {
8
9   }
```

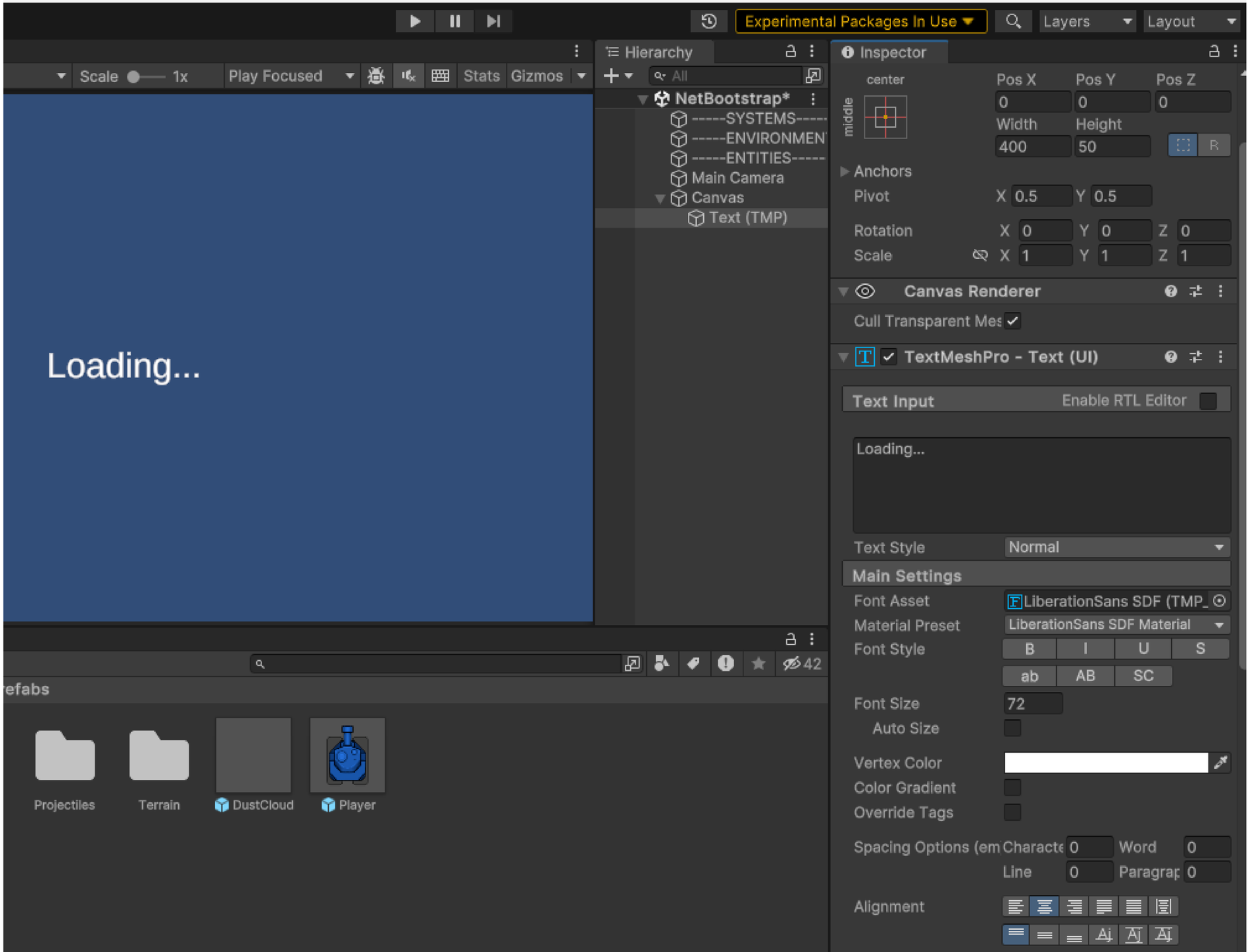
Script **HostSingleton**

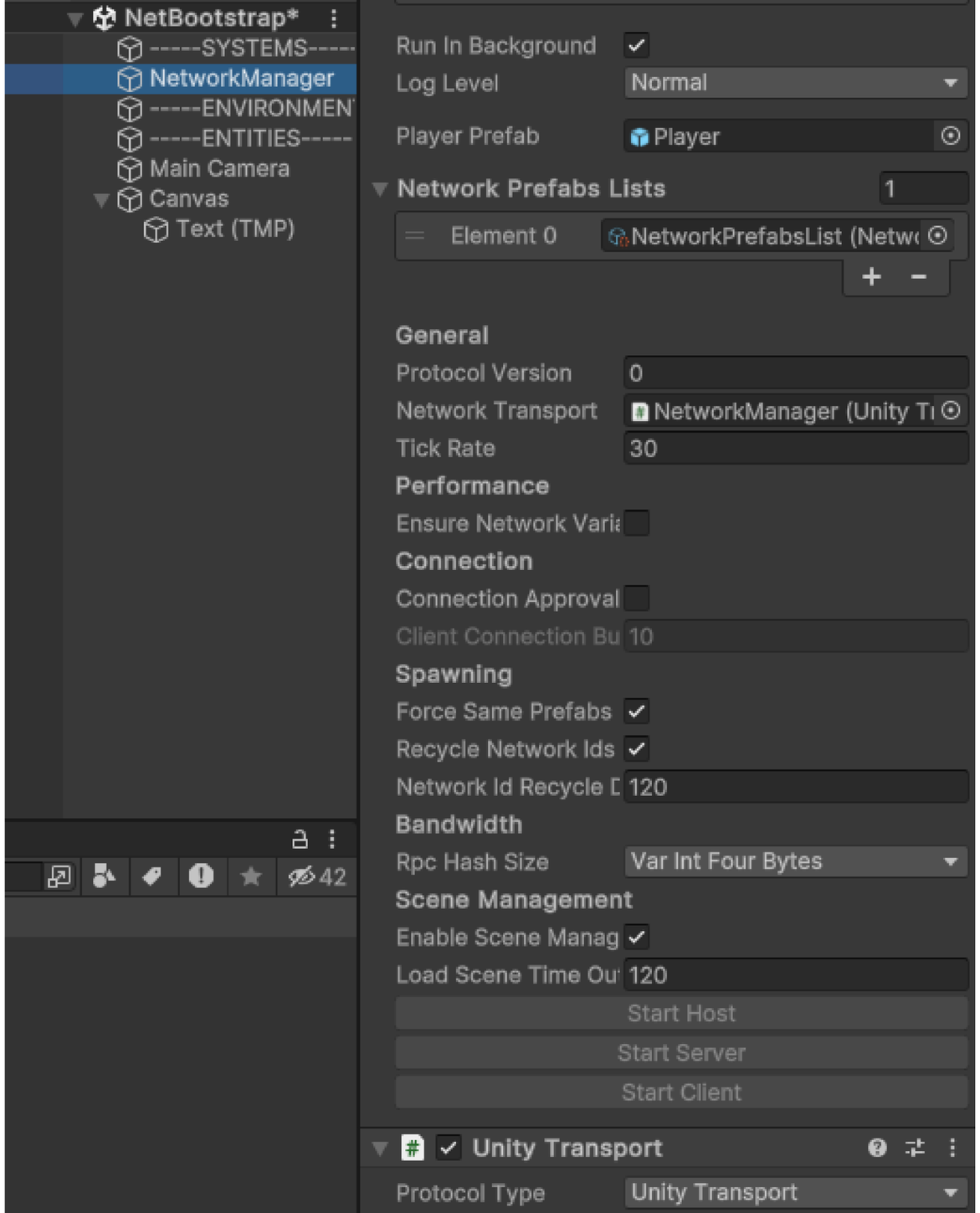
# Authentication

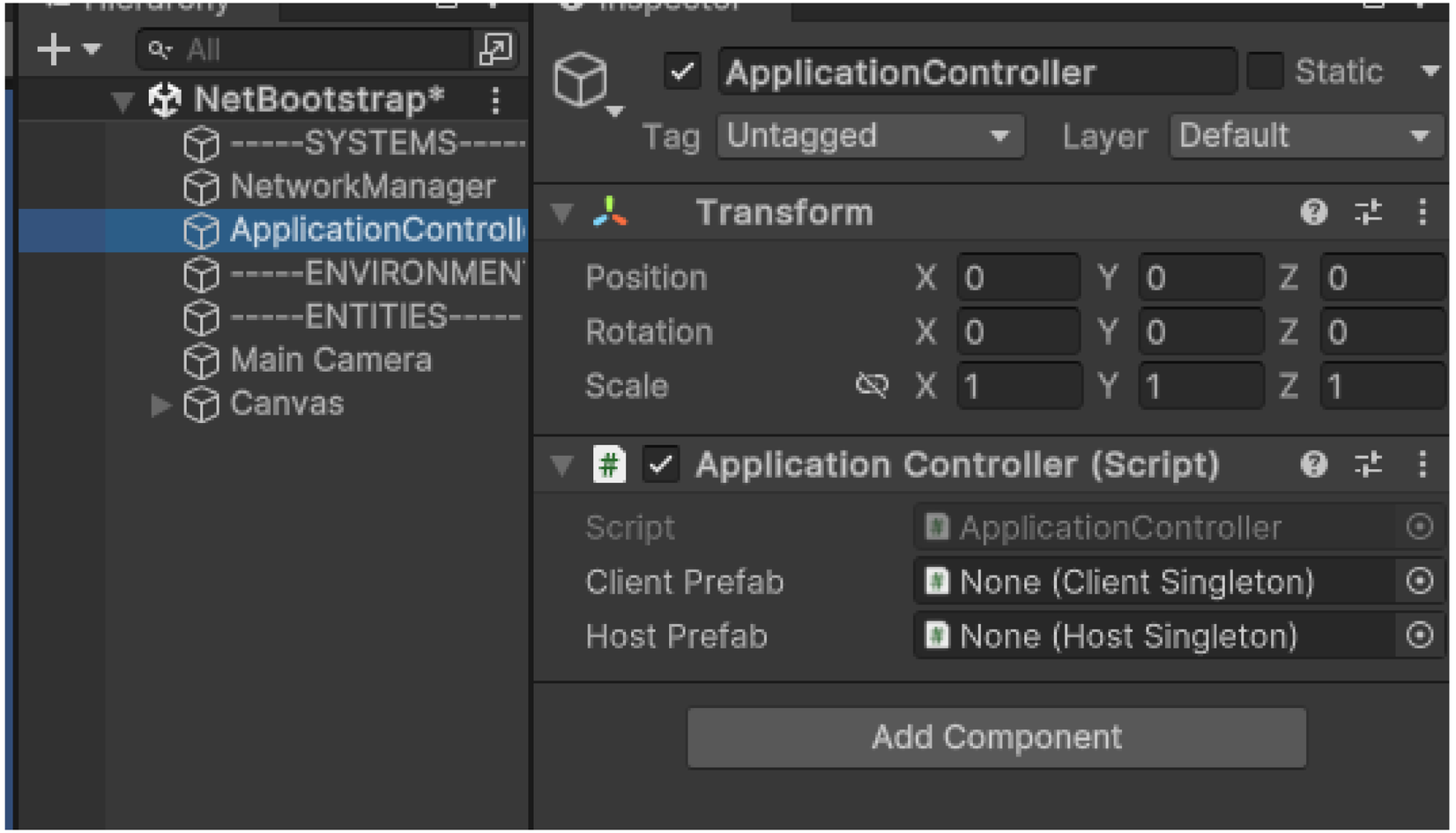

Package Manager
Packages: Unity Registry
Sort: Name (asc)
Filters
Clear Filters
All
Services
2D Animation 9.0.1
2D Pixel Perfect 5.0.3
2D PSD Importer 8.0.2
2D Sprite 1.0.0
2D SpriteShape 9.0.1
2D Tilemap Editor 1.0.0
2D Tilemap Extras 3.0.3
Adaptive Performance 4.0.1
Addressables 1.21.9
Ads Mediation 1.0.0
Advertisement Legacy 4.4.2
AI Navigation 1.1.1
Alembic 2.3.0
Analytics 4.4.1
Android Logcat 1.3.2
Animation Rigging 1.2.1
Apple ARKit XR Plugin 5.0.5
AR Foundation 5.0.5
Authentication 2.4.0
Last update Apr 7, 10:09
Authentication
Configure
Remove
2.4.0 · December 06, 2022 Release
Installed as dependency
From Unity Registry by Unity Technologies Inc.
com.unity.services.authentication
Documentation
Changelog
Licenses
Go to Dashboard
Description
Version History
Dependencies
Samples
Supported Platforms: Android iOS Linux Show all
Client SDK offering player identity management for Unity Gaming Services.
Identity providers can be configured from the editor or the Unity Dashboard.
Player management is available in the Unity Dashboard.

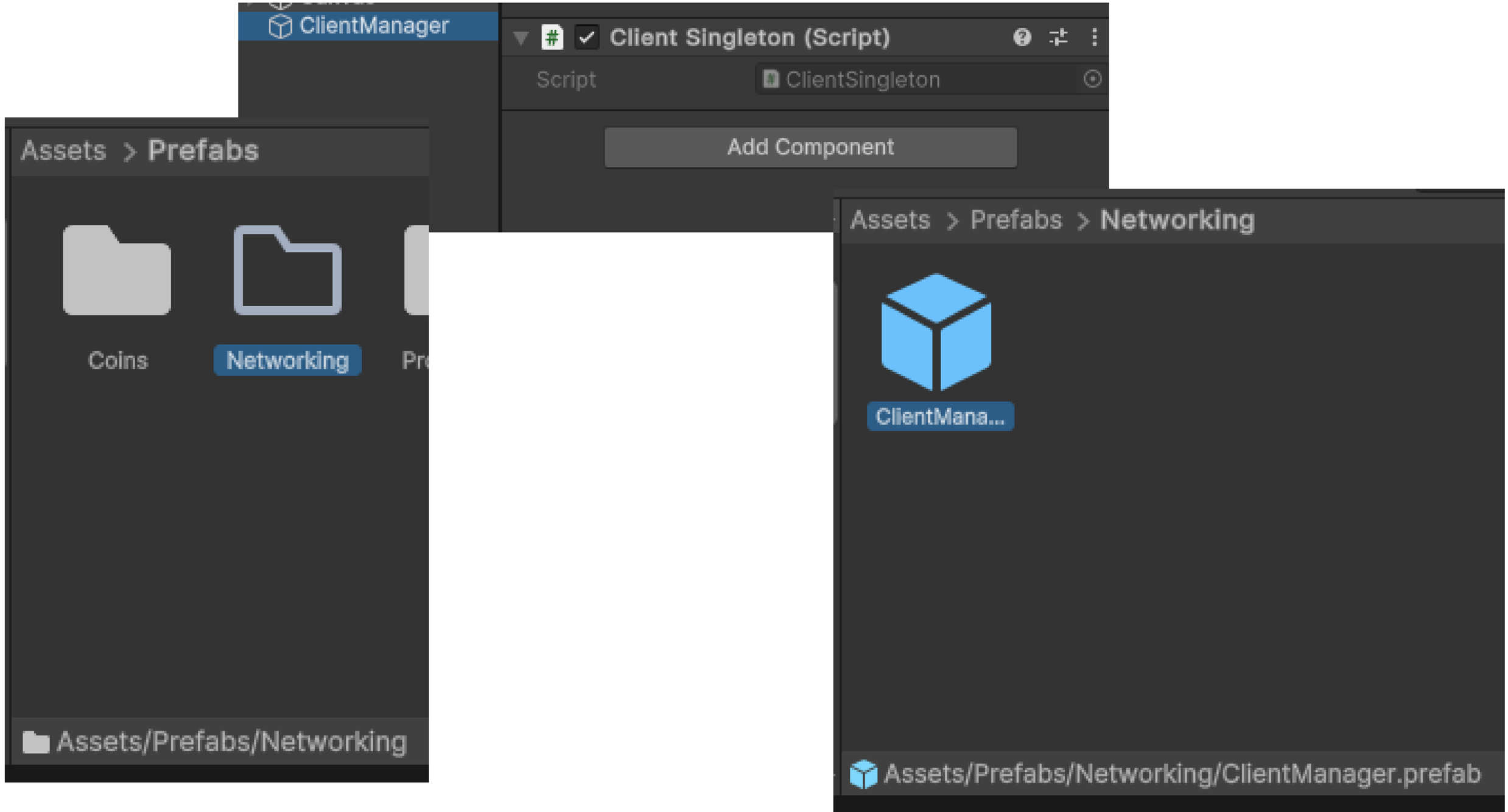

Assets > Scenes
Game
Menu
NetBootstrap

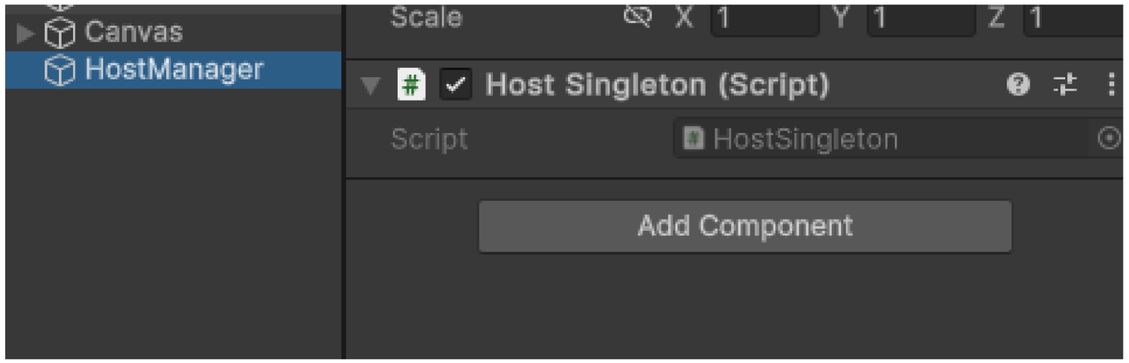

CTRL
+
d

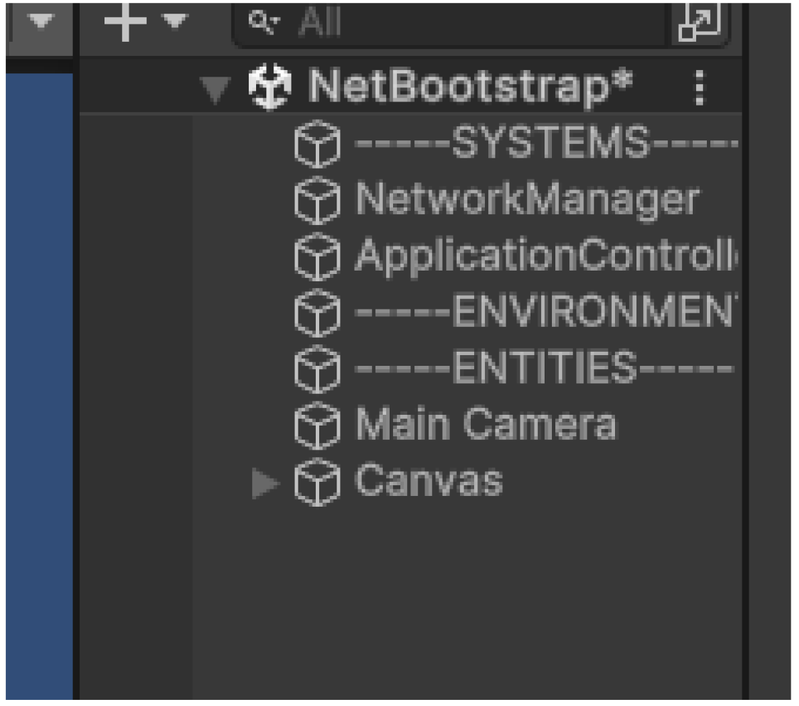

Experimental Packages In Use
Layers
Layout
Scale
1x
Play Focused
Stats
Gizmos
Loading...
Hierarchy
All
NetBootstrap*
-----SYSTEMS-----
-----ENVIRONMEN
-----ENTITIES-----
Main Camera
Canvas
Text (TMP)
Inspector
center
middle
Pos X
Pos Y
Pos Z
Width
Height
Anchors
Pivot
Rotation
Scale
Canvas Renderer
Cull Transparent Mes
TextMeshPro - Text (UI)
Text Input
Enable RTL Editor
Loading...
Text Style
Normal
Main Settings
Font Asset
LiberationSans SDF (TMP_
Material Preset
LiberationSans SDF Material
Font Style
Font Size
Auto Size
Vertex Color
Color Gradient
Override Tags
Spacing Options (em
Line
Alignment
efabs
Projectiles
Terrain
DustCloud
Player

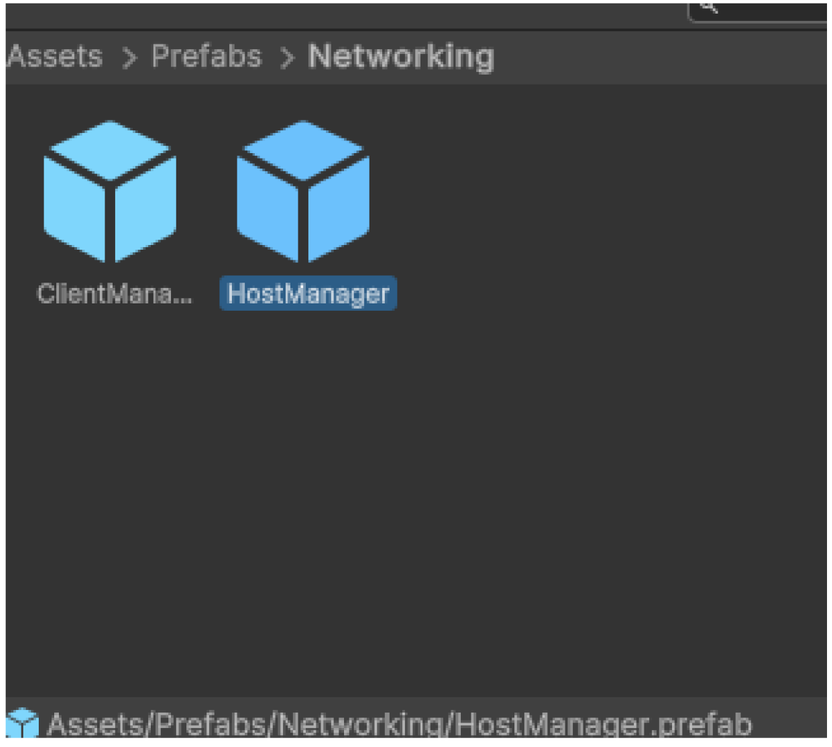

NetBootstrap*
-----SYSTEMS-----
NetworkManager
-----ENVIRONMEN
-----ENTITIES-----
Main Camera
Canvas
Text (TMP)
42
Run In Background
Log Level
Normal
Player Prefab
Player
Network Prefabs Lists
1
Element 0
NetworkPrefabsList (Netw
General
Protocol Version
0
Network Transport
NetworkManager (Unity Tı
Tick Rate
30
Performance
Ensure Network Varia
Connection
Connection Approval
Client Connection Bu
10
Spawning
Force Same Prefabs
Recycle Network Ids
Network Id Recycle D
120
Bandwidth
Rpc Hash Size
Var Int Four Bytes
Scene Management
Enable Scene Manag
Load Scene Time Ou
120
Start Host
Start Server
Start Client
Unity Transport
Protocol Type
Unity Transport

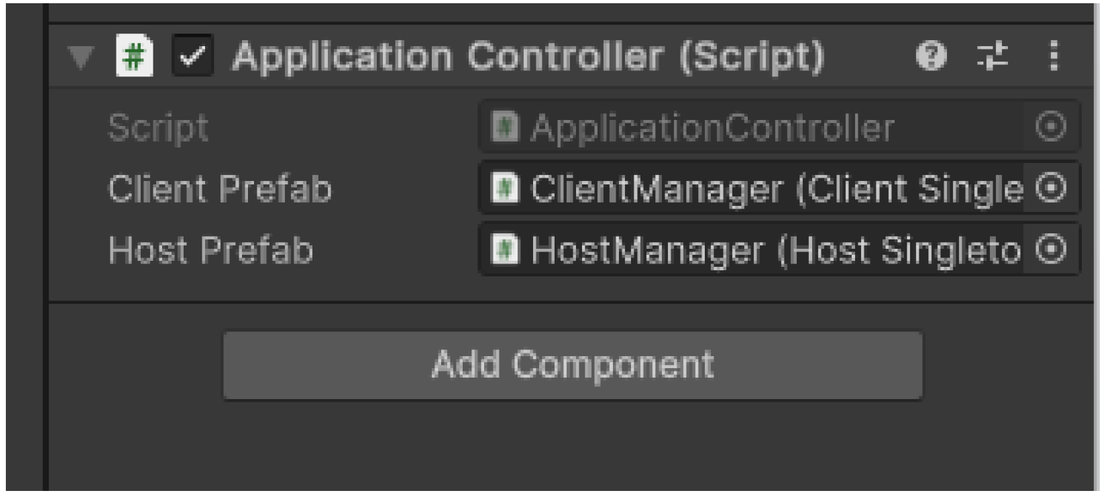

All
NetBootstrap*
-----SYSTEMS-----
NetworkManager
ApplicationControll
-----ENVIRONMEN
-----ENTITIES-----
Main Camera
Canvas
ApplicationController
Static
Tag
Untagged
Layer
Default
Transform
Position
X 0
Y 0
Z 0
Rotation
X 0
Y 0
Z 0
Scale
X 1
Y 1
Z 1
Application Controller (Script)
Script
ApplicationController
Client Prefab
None (Client Singleton)
Host Prefab
None (Host Singleton)
Add Component

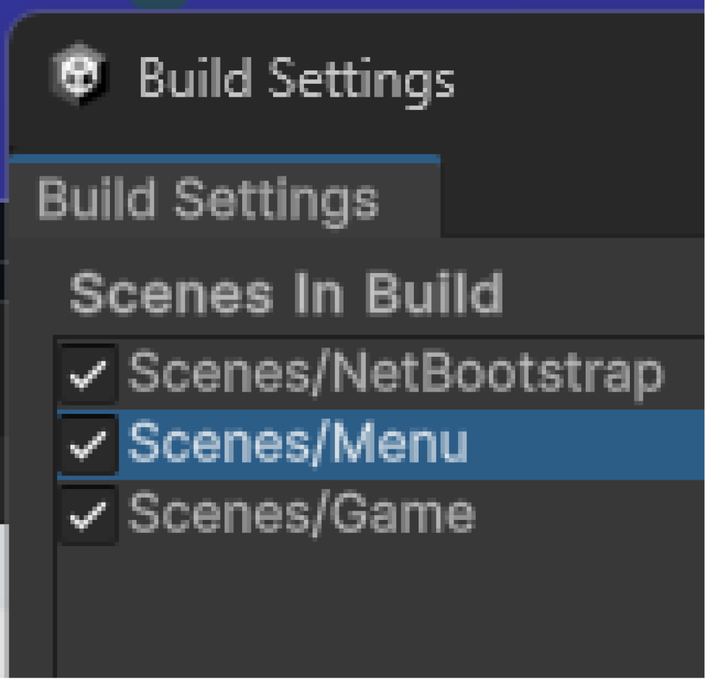

ClientManager
Client Singleton (Script)
Script
ClientSingleton
Add Component
Assets > Prefabs
Coins
Networking
Assets/Prefabs/Networking
Assets > Prefabs > Networking
ClientMana...
Assets/Prefabs/Networking/ClientManager.prefab

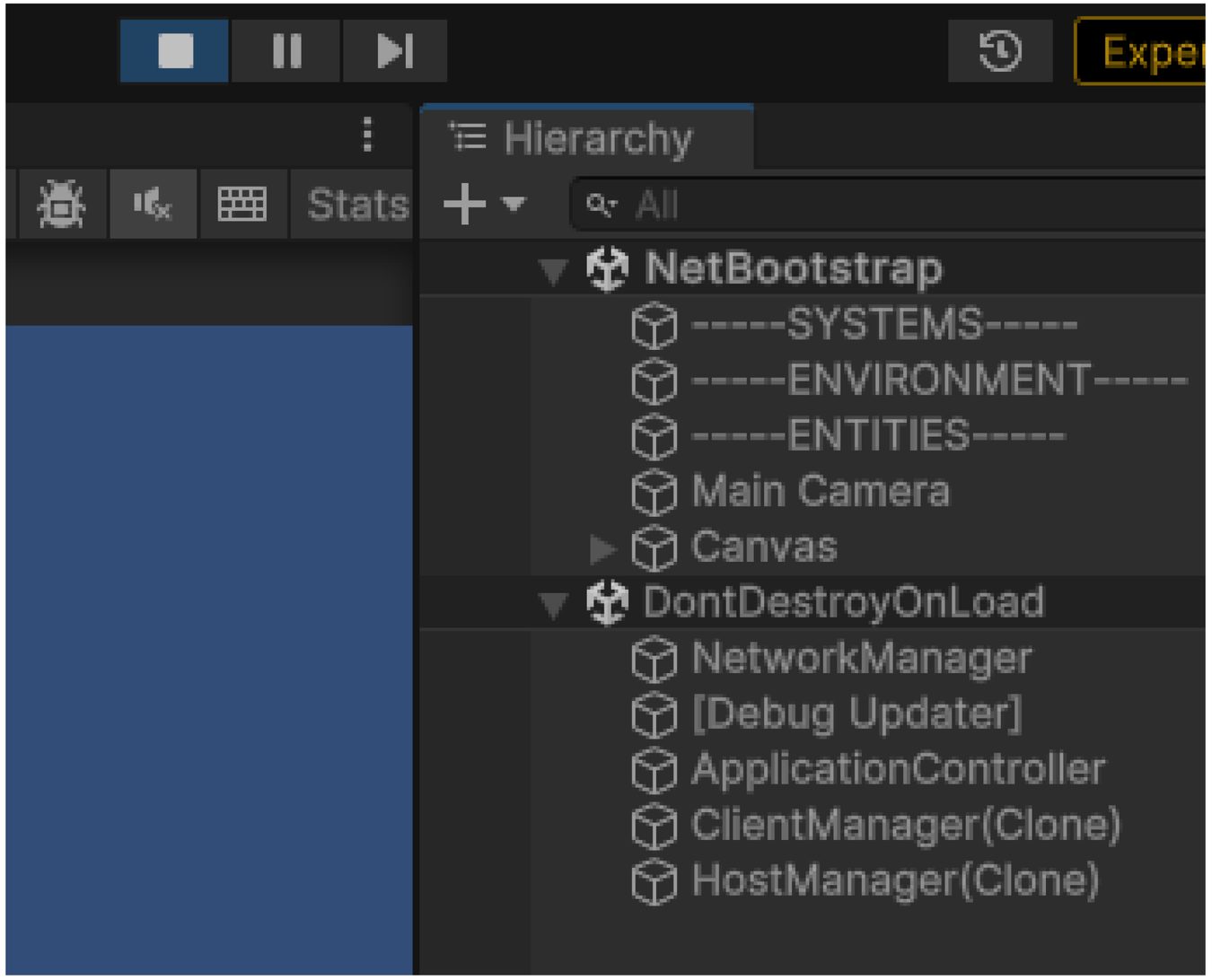

Canvas
HostManager
Scale
X 1
Y 1
Z 1
Host Singleton (Script)
Script
HostSingleton
Add Component

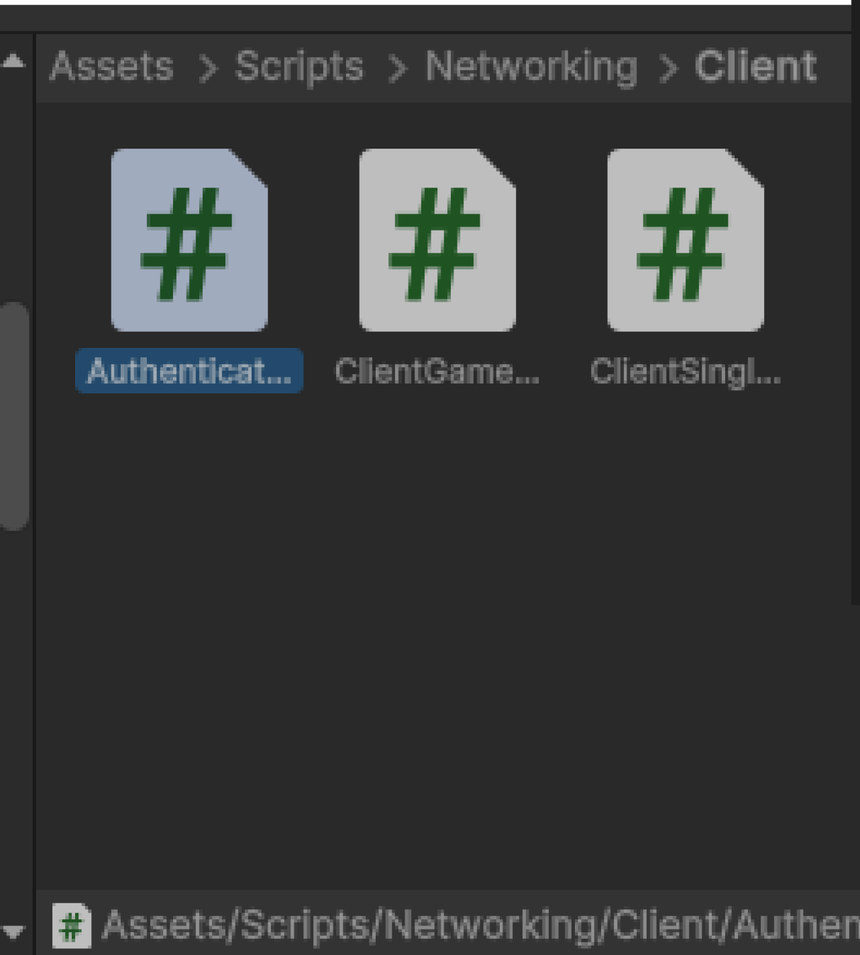

All
NetBootstrap*
-----SYSTEMS-----
NetworkManager
ApplicationControll
-----ENVIRONMEN
-----ENTITIES-----
Main Camera
Canvas


Assets > Prefabs > Networking
ClientMana...
HostManager
Assets/Prefabs/Networking/HostManager.prefab


Application Controller (Script)
Script
ApplicationController
Client Prefab
ClientManager (Client Single
Host Prefab
HostManager (Host Singleto
Add Component


Build Settings
Build Settings
Scenes In Build
Scenes/NetBootstrap
Scenes/Menu
Scenes/Game

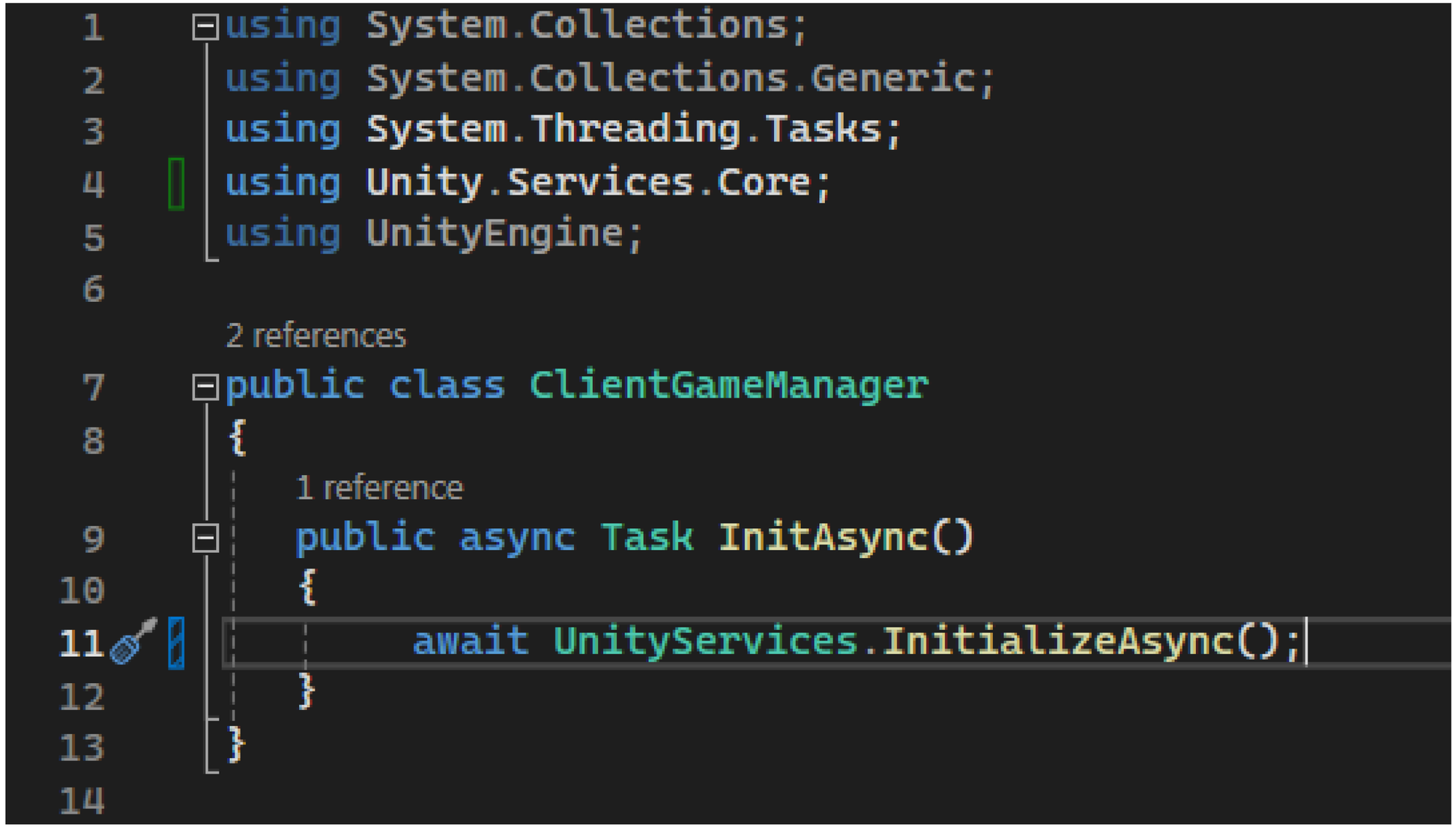

Stats
Hierarchy
All
NetBootstrap
-----SYSTEMS-----
-----ENVIRONMENT-----
-----ENTITIES-----
Main Camera
Canvas
DontDestroyOnLoad
NetworkManager
[Debug Updater]
ApplicationController
ClientManager(Clone)
HostManager(Clone)



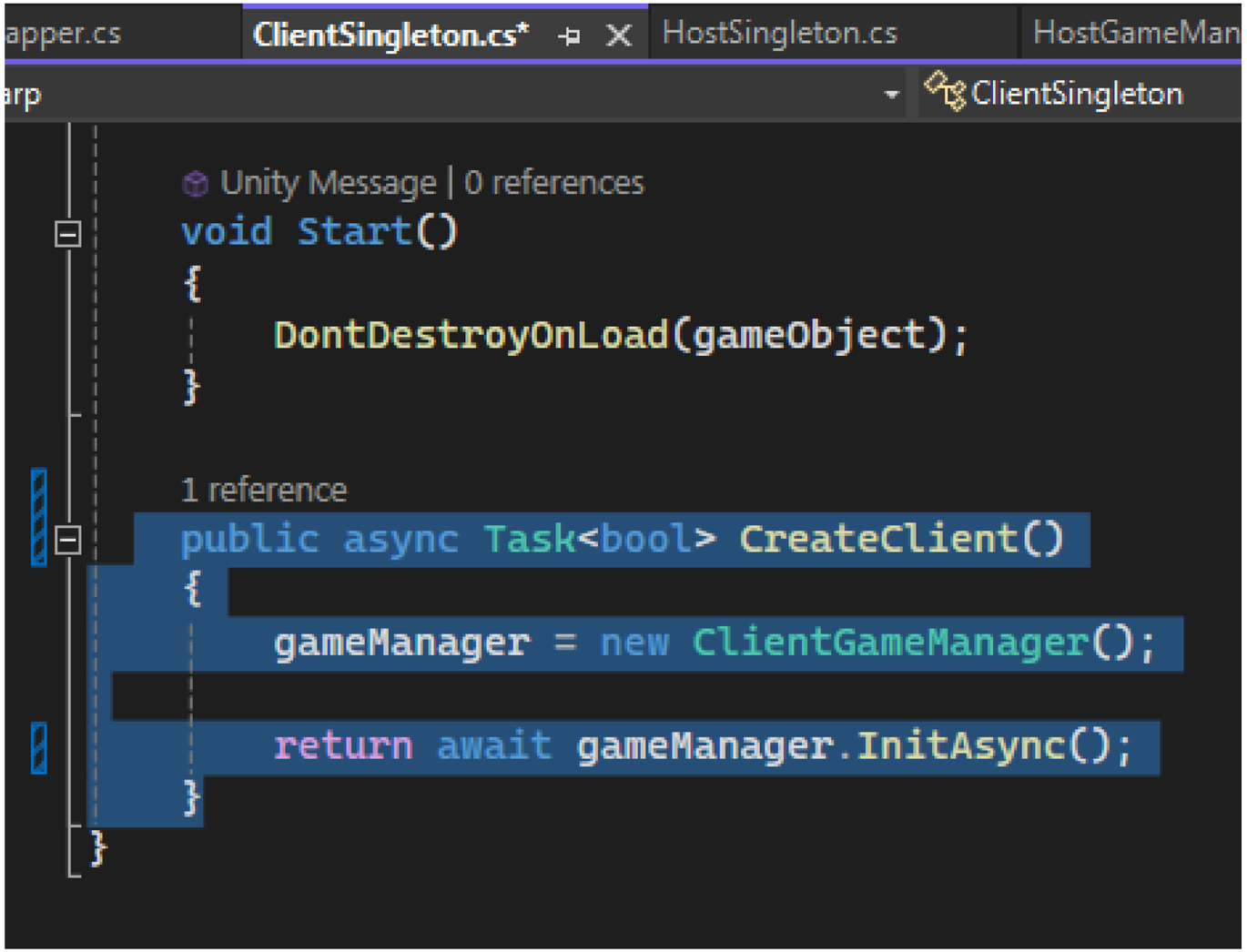

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
    0 references
5   public static class AuthenticationWrapper
6   {
        0 references
7       public static AuthState AuthState { get; private set; }
8   }
9
    1 reference
10  public enum AuthState
11  {
12      NotAuthenticated,
13      Authenticating,
14      Authenticated,
15      Error,
16      TimeOut
17  }
```

```
0 references
public static class AuthenticationWrapper
{
    3 references
    public static AuthState AuthState { get; private set; } = AuthState.NotAuthenticated;

    public static async Task<AuthState> DoAuth(int maxRetries = 5)
    {
        if(AuthState == AuthState.Authenticated)
        {
            return AuthState;
        }

        AuthState = AuthState.Authenticating;

        int retries = 0;
        while(AuthState == AuthState.Authenticating &&retries < maxRetries)
        {
            await AuthenticationService.Instance.SignInAnonymouslyAsync();
        }
    }
}

5 references
public enum AuthState
{
    NotAuthenticated,
    Authenticating,
    Authenticated,
    Error,
    TimeOut
}
```

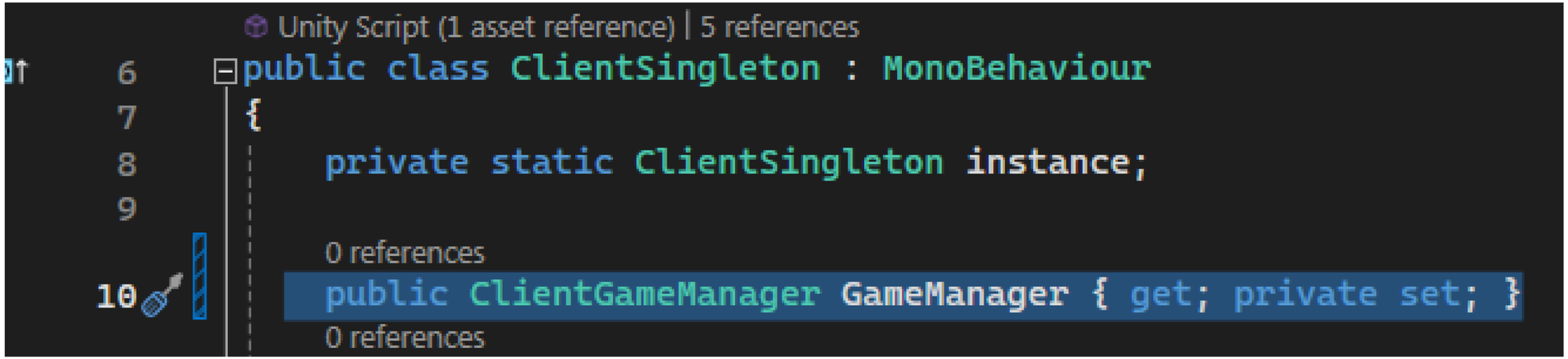
```
11      public static async Task<AuthState> DoAuth(int maxRetries = 5)
12      {
13          if(AuthState == AuthState.Authenticated)
14          {
15              return AuthState;
16          }
17
18          AuthState = AuthState.Authenticating;
19
20          int retries = 0;
21          while(AuthState == AuthState.Authenticating &&retries < maxRetries)
22          {
23              await AuthenticationService.Instance.SignInAnonymouslyAsync();
24
25              if(AuthenticationService.Instance.IsSignedIn && AuthenticationService.Instance.IsAuthorized)
26              {
27                  AuthState = AuthState.Authenticated;
28                  break;
29              }
30
31              retries++;
32              await Task.Delay(1000);
33          }
34          return AuthState;
35      }
36  }
```


using System.Collections;
using System.Collections.Generic;
using System.Threading.Tasks;
using Unity.Services.Core;
using UnityEngine;
2 references
public class ClientGameManager
{
1 reference
public async Task InitAsync()
{
await UnityServices.InitializeAsync();
}
}

```
using System.Collections;
using System.Collections.Generic;
using System.Threading.Tasks;
using Unity.Services.Core;
using UnityEngine;

2 references
public class ClientGameManager
{
    1 reference
    public async Task<bool> InitAsync()
    {
        await UnityServices.InitializeAsync();

        AuthState authState = await AuthenticationWrapper.DoAuth()

        if(authState == AuthState.Authenticated)
        {
            return true;
        }

        return false;
    }
}
```

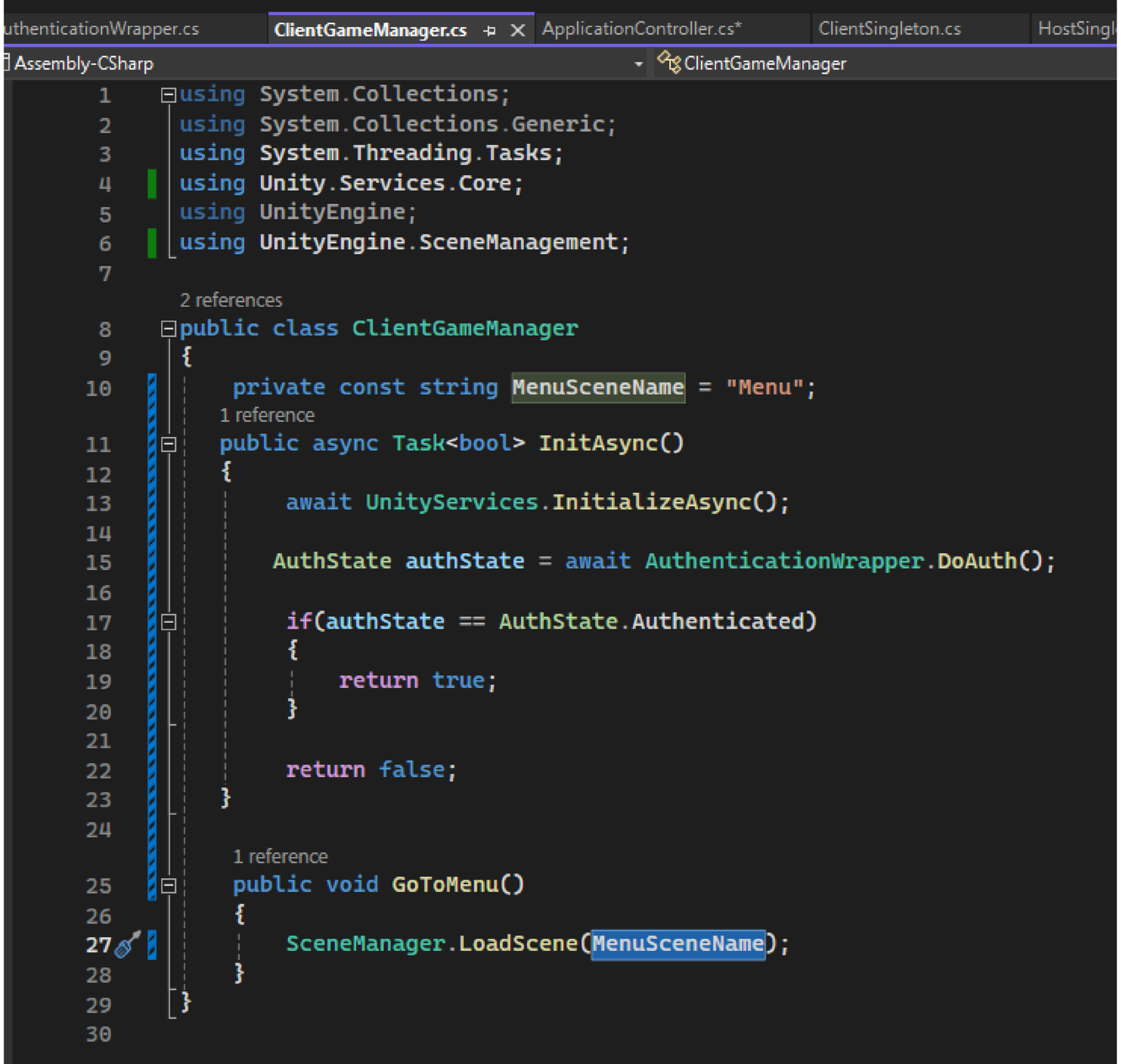

apper.cs
ClientSingleton.cs*
HostSingleton.cs
HostGameMan
arp
ClientSingleton
Unity Message | 0 references
void Start()
{
DontDestroyOnLoad(gameObject);
}
1 reference
public async Task<bool> CreateClient()
{
gameManager = new ClientGameManager();
return await gameManager.InitAsync();
}
}

```
ApplicationController
public class ApplicationController : MonoBehaviour
{
    [SerializeField] private ClientSingleton clientPrefab;
    [SerializeField] private HostSingleton hostPrefab;

    Unity Message | 0 references
    private async void Start()
    {
        DontDestroyOnLoad(gameObject);
        await LaunchInMode(SystemInfo.graphicsDeviceType == UnityEngine.Rendering.

    }

    1 reference
    private async Task LaunchInMode(bool isDedicatedServer)
    {
        if(isDedicatedServer)
        {

        }
        else
        {
            ClientSingleton clientSingleton = Instantiate(clientPrefab);
            bool authenticated = await clientSingleton.CreateClient();

            HostSingleton hostSingleton = Instantiate(hostPrefab);
            hostSingleton.CreateHost();

            if (authenticated)
            {

            }
        }
    }
}
```




```
1 reference
public async Task<bool> CreateClient()
{
    GameManager = new ClientGameManager();

    return await GameManager.InitAsync();
}
```

```
5
                Unity Script (1 asset reference) | 0 references
6   public class ApplicationController : MonoBehaviour
7   {
8       [SerializeField] private ClientSingleton clientPrefab;
9       [SerializeField] private HostSingleton hostPrefab;
10
                Unity Message | 0 references
11      private async void Start()
12      {
13          DontDestroyOnLoad(gameObject);
14          await LaunchInMode(SystemInfo.graphicsDeviceType == UnityEngine.Re
15      }
16
                1 reference
17      private async Task LaunchInMode(bool isDedicatedServer)
18      {
19          if(isDedicatedServer)
20          {
21
22          }
23          else
24          {
25              ClientSingleton clientSingleton = Instantiate(clientPrefab);
26              bool authenticated = await clientSingleton.CreateClient();
27
28              HostSingleton hostSingleton = Instantiate(hostPrefab);
29              hostSingleton.CreateHost();
30
31              if (authenticated)
32              {
33                  clientSingleton.GameManager.GoToMenu();
34              }
35          }
36      }
37  }
```


uthenticationWrapper.cs
ClientGameManager.cs
ApplicationController.cs*
ClientSingleton.cs
HostSingl
Assembly-CSharp
ClientGameManager
using System.Collections;
using System.Collections.Generic;
using System.Threading.Tasks;
using Unity.Services.Core;
using UnityEngine;
using UnityEngine.SceneManagement;
2 references
public class ClientGameManager
{
private const string MenuSceneName = "Menu";
1 reference
public async Task<bool> InitAsync()
{
await UnityServices.InitializeAsync();
AuthState authState = await AuthenticationWrapper.DoAuth();
if(authState == AuthState.Authenticated)
{
return true;
}
return false;
}
1 reference
public void GoToMenu()
{
SceneManager.LoadScene(MenuSceneName);
}
}

Code Recheck

# Authentication (Complete) :

Script **AuthenticationWrapper**

```
using System.Threading.Tasks;
using Unity.Services.Authentication;
using UnityEngine;

1 reference
public static class AuthenticationWrapper
{
    6 references
    public static AuthState AuthState { get; private set; } = AuthState.NotAuthenticated;

    1 reference
    public static async Task<AuthState> DoAuth(int maxRetries = 5)
    {
        if(AuthState == AuthState.Authenticated)
        {
            return AuthState;
        }

        AuthState = AuthState.Authenticating;

        int retries = 0;
        while(AuthState == AuthState.Authenticating &&retries < maxRetries)
        {
            await AuthenticationService.Instance.SignInAnonymouslyAsync();

            if(AuthenticationService.Instance.IsSignedIn && AuthenticationService.Instance.IsAuthorized)
            {
                AuthState = AuthState.Authenticated;
                break;
            }

            retries++;
            await Task.Delay(1000);
        }
        return AuthState;
    }
}

9 references
public enum AuthState
{
    NotAuthenticated,
    Authenticating,
    Authenticated,
    Error,
    TimeOut
}
```

# Authentication (Complete) :

Script **ClientSingleton**

```
using System.Collections;
using System.Collections.Generic;
using System.Threading.Tasks;
using UnityEngine;

Unity Script (1 asset reference) | 5 references
public class ClientSingleton : MonoBehaviour
{
    private static ClientSingleton instance;

    3 references
    public ClientGameManager GameManager { get; private set; }
    0 references
    public static ClientSingleton Instance
    {
        get
        {
            if(instance == null) { return instance; }
            instance = FindFirstObjectByType<ClientSingleton>();

            if (instance == null)
            {
                Debug.LogError("No ClientSingleton in the scene!");
                return null;
            }
            return instance;
        }
    }

    Unity Message | 0 references
    void Start()
    {
        DontDestroyOnLoad(gameObject);
    }

    1 reference
    public async Task<bool> CreateClient()
    {
        GameManager = new ClientGameManager();

        return await GameManager.InitAsync();
    }
}
```

Code Recheck

# Authentication (Complete) :

Script **ClientGameManager**

```
using System.Collections;
using System.Collections.Generic;
using System.Threading.Tasks;
using Unity.Services.Core;
using UnityEngine;
using UnityEngine.SceneManagement;

2 references
public class ClientGameManager
{
    private const string MenuSceneName = "Menu";

    1 reference
    public async Task<bool> InitAsync()
    {
        await UnityServices.InitializeAsync();

        AuthState authState = await AuthenticationWrapper.DoAuth();

        if(authState == AuthState.Authenticated)
        {
            return true;
        }

        return false;
    }

    1 reference
    public void GoToMenu()
    {
        SceneManager.LoadScene(MenuSceneName);
    }
}
```

# Auth Improvements

(เนื้อหาประมาณ 10 นาที)

โหวตจะให้เป็นวีดีโอ หรือ ไว้อยู่ในเนื้อหาสัปดาห์หน้า

# Separate Sign-In Logic

- Create a new method: **private static async Task SignInAnonymouslyAsync(int)**
- Move our sign-in logic into that method
- Call this method, by awaiting, in the **DoAuth** method

# Assignment

ให้ทำการถ่ายคลิปผลลัพธ์ของการทำ Workshop ตามเนื้อหา Workshop ในแต่ละหัวข้อนี้พร้อมอธิบายประกอบ

- Combat Polish
- Setup Main Menu
- Application Controller
- Authentication

https://discord.gg/24xmUFHfzR

# Topic next week

## Workshop

- Auth Improvements
- Relay
  - Service Setup
  - Allocating A Relay
  - Joining A Relay
- Lobbies
  - Lobbies UI
  - Creating Lobbies
  - Joining Lobbies

WOLVEDEN ACADEMY

# THANK YOU

facebook.com/GameDevMew