Special topics lesson 2 [game making]

Professor: Dr. Rezapour

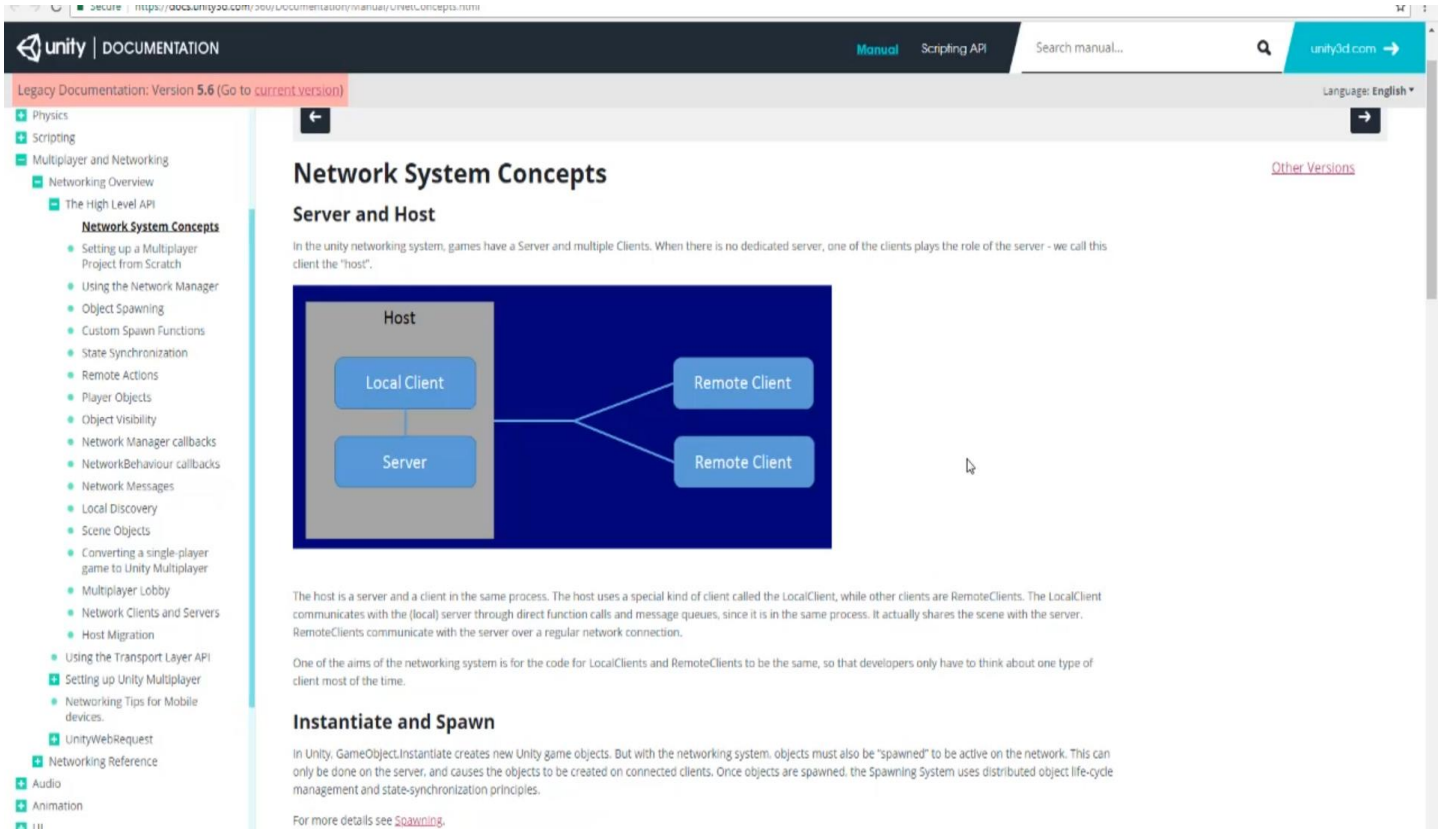
Network In Unity

Mohadese Baghbani - Mohammad Amin Kiani - Nastaran Sarkhosh

- Topics:

  - ✓ Unity Docs
  - ✓ Hints & Steps to set up
  - ✓ MNH Vs. Me!

Unity Docs


unity | DOCUMENTATION
Manual
Scripting API
Search manual...
unity3d.com
Legacy Documentation: Version 5.6 (Go to current version)
Language: English
Physics
Scripting
Multiplayer and Networking
Networking Overview
The High Level API
Network System Concepts
Setting up a Multiplayer Project from Scratch
Using the Network Manager
Object Spawning
Custom Spawn Functions
State Synchronization
Remote Actions
Player Objects
Object Visibility
Network Manager callbacks
NetworkBehaviour callbacks
Network Messages
Local Discovery
Scene Objects
Converting a single-player game to Unity Multiplayer
Multiplayer Lobby
Network Clients and Servers
Host Migration
Using the Transport Layer API
Setting up Unity Multiplayer
Networking Tips for Mobile devices.
UnityWebRequest
Networking Reference
Audio
Animation
Network System Concepts
Other Versions
Server and Host
In the unity networking system, games have a Server and multiple Clients. When there is no dedicated server, one of the clients plays the role of the server - we call this client the "host".
Host
Local Client
Server
Remote Client
Remote Client
The host is a server and a client in the same process. The host uses a special kind of client called the LocalClient, while other clients are RemoteClients. The LocalClient communicates with the (local) server through direct function calls and message queues, since it is in the same process. It actually shares the scene with the server. RemoteClients communicate with the server over a regular network connection.
One of the aims of the networking system is for the code for LocalClients and RemoteClients to be the same, so that developers only have to think about one type of client most of the time.
Instantiate and Spawn
In Unity, GameObject.Instantiate creates new Unity game objects. But with the networking system, objects must also be "spawned" to be active on the network. This can only be done on the server, and causes the objects to be created on connected clients. Once objects are spawned, the Spawning System uses distributed object life-cycle management and state-synchronization principles.
For more details see Spawning.

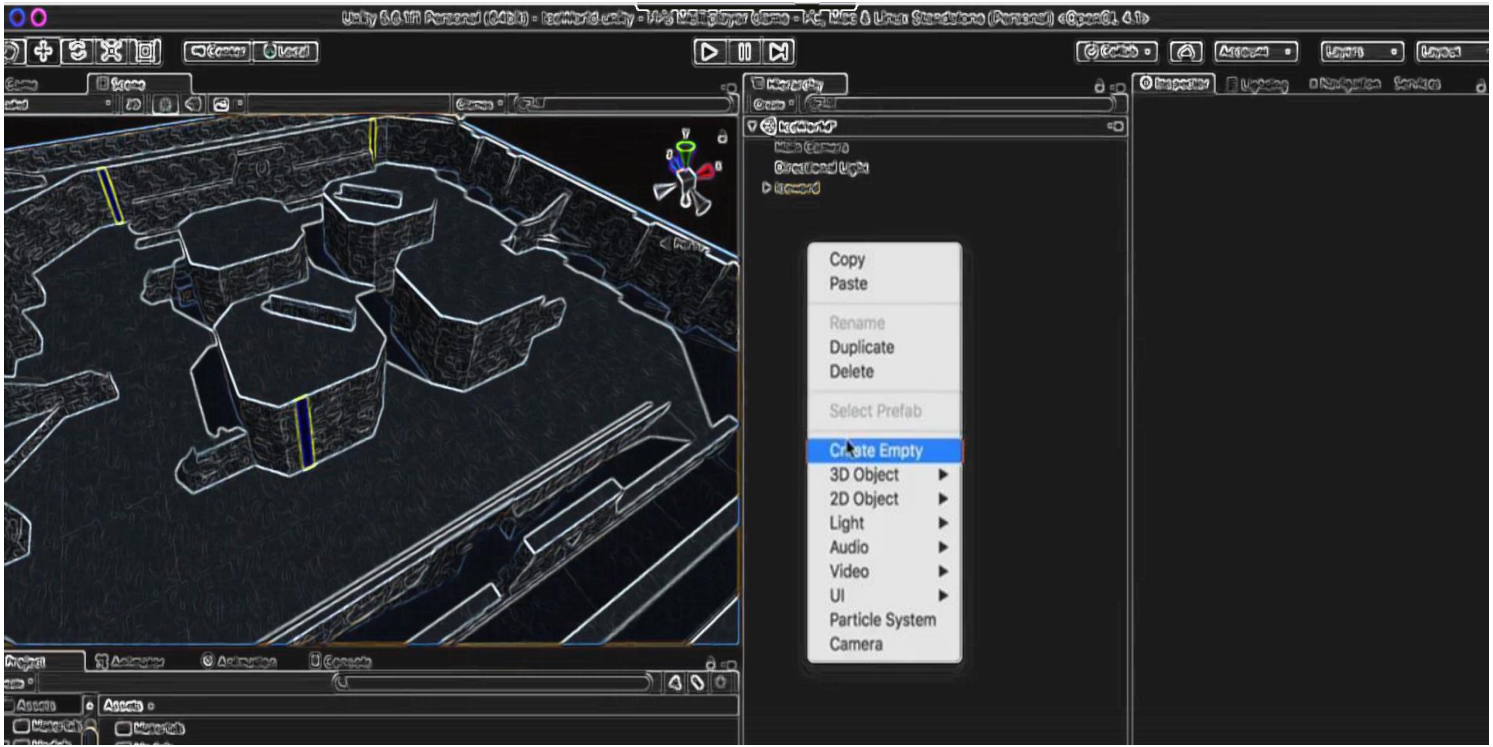
Danger!

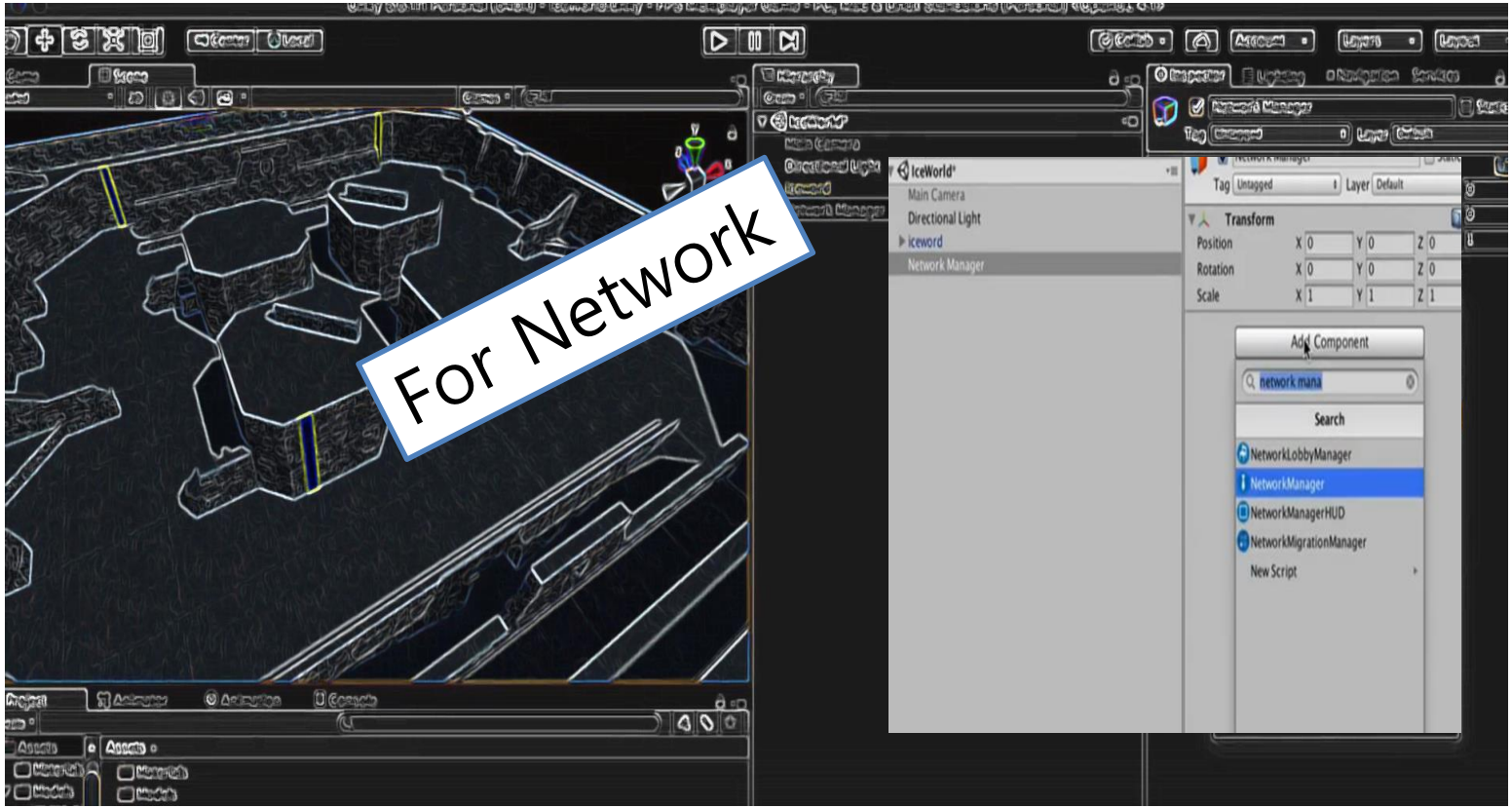
- Scripts in Unity are shared across the network

  - ✓ It means that if we shoot, other players will also shoot.
  - ✓ So, a condition should be made that if this code is executed on our computer, execute it, otherwise, do not do anything on other systems. Also, if someone else is running the code, it should not be run for us.
  - ✓ the local or non-locality of the system is very important in determining the control of the game!

How?

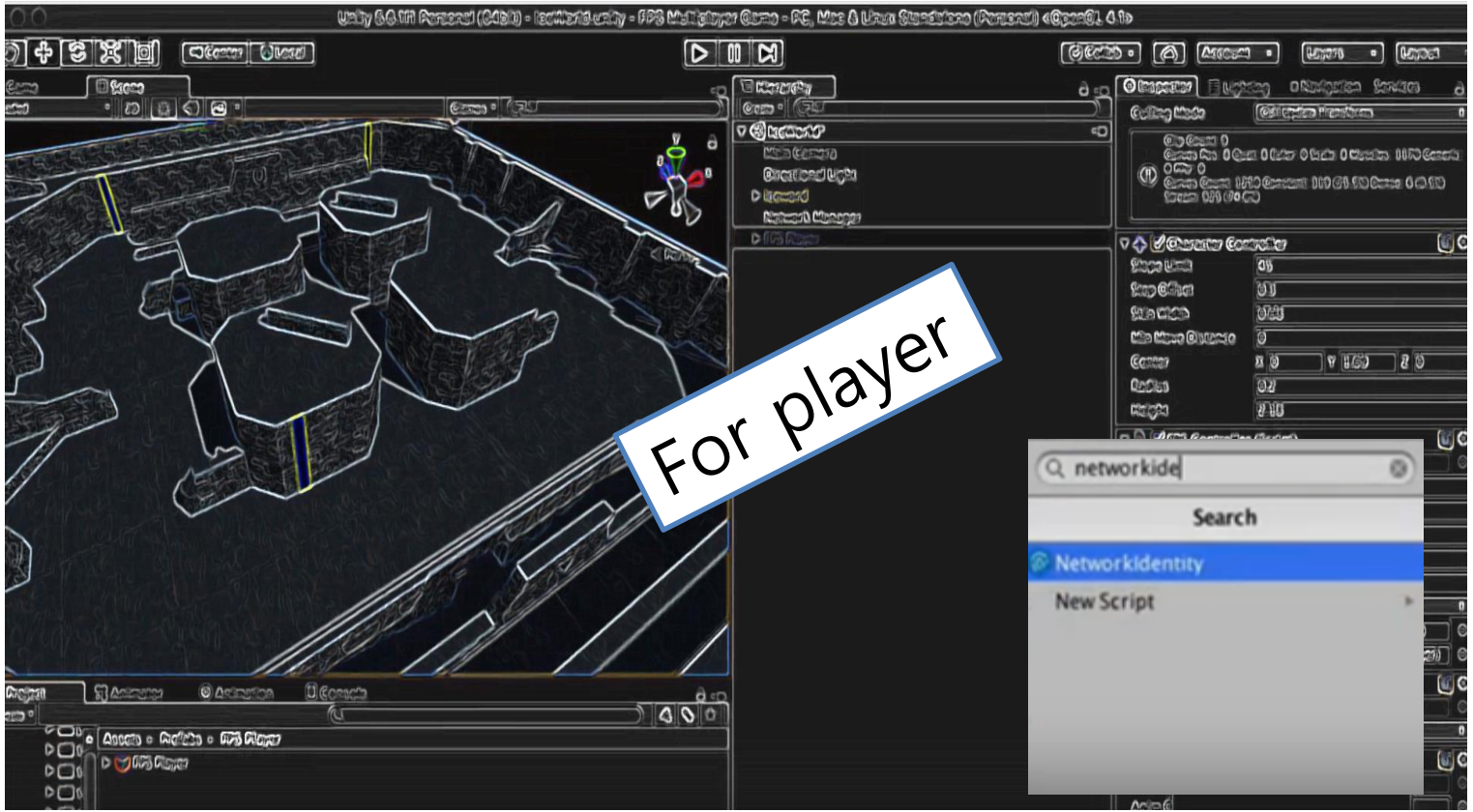
Networking Steps

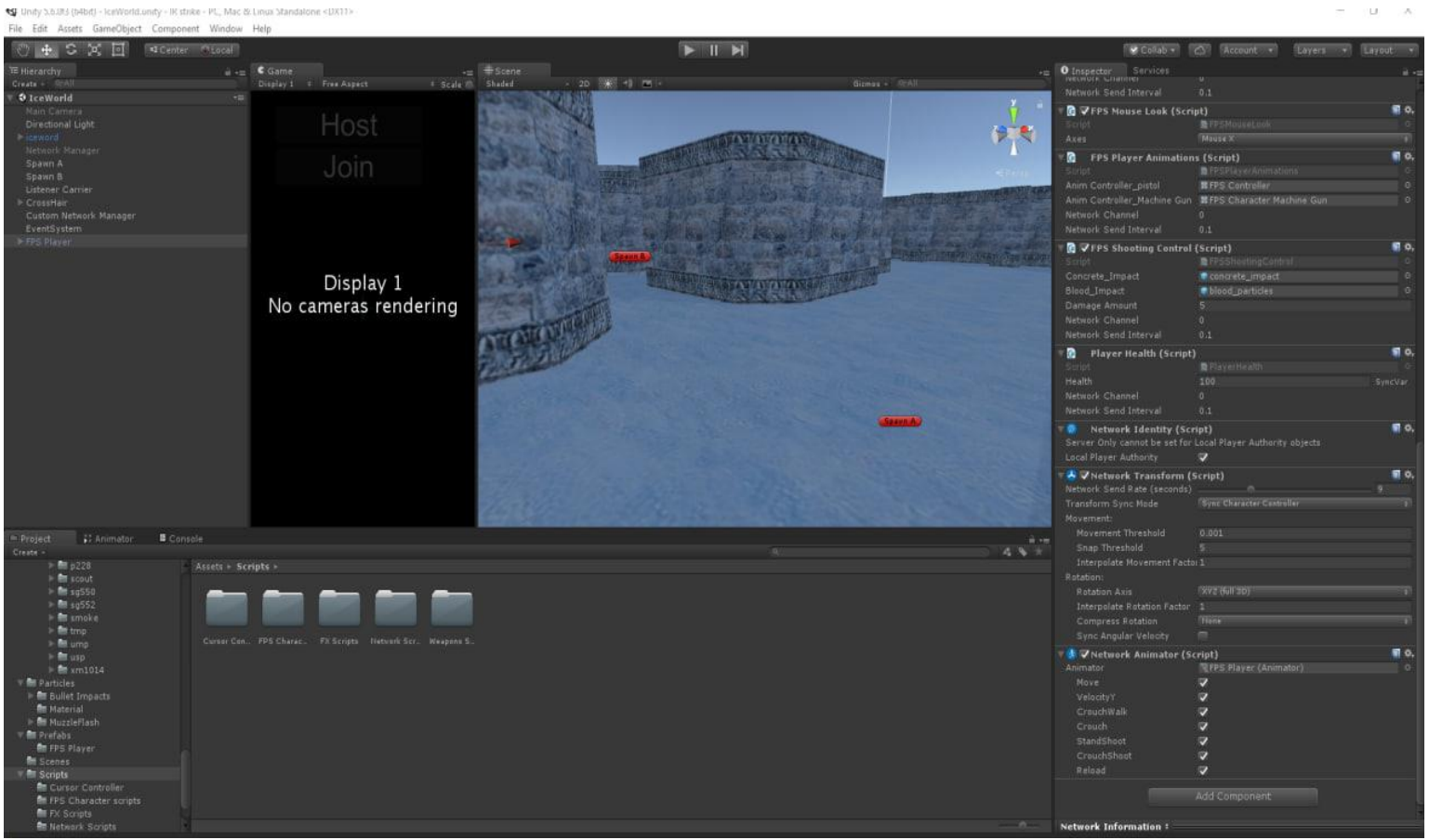


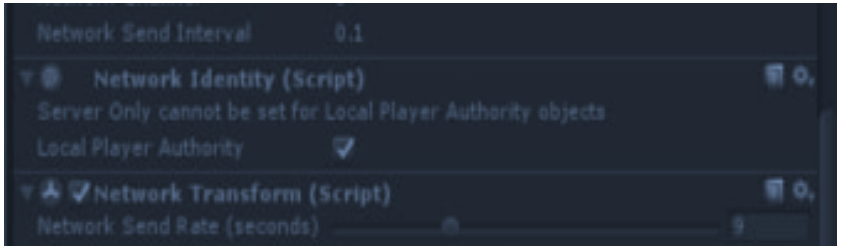

For Network
IceWorld*
Main Camera
Directional Light
iceword
Network Manager
Transform
Position
Rotation
Scale
Add Component
network mana
Search
NetworkLobbyManager
NetworkManager
NetworkManagerHUD
NetworkMigrationManager
New Script

We cannot add the player to the network. Why?

Because it is normally treated like other objects such as a tree and has no identity to determine the controller.


For player
networkide
Search
Networkidentity
New Script


File Edit Assets GameObject Component Window Help
Hierarchy
IceWorld
Main Camera
Directional Light
Network Manager
Spawn A
Spawn B
Listener Carrier
CrossHair
Custom Network Manager
EventSystem
FPS Player
Game
Host
Join
Display 1
No cameras rendering
Scene
Inspector
FPS Mouse Look (Script)
FPS Player Animations (Script)
FPS Shooting Control (Script)
Player Health (Script)
Network Identity (Script)
Network Transform (Script)
Network Animator (Script)
Add Component
Project
Animator
Console
Assets ▸ Scripts ▸
Particles
Bullet Impacts
Material
MuzzleFlash
Prefabs
FPS Player
Scenes
Scripts
Cursor Controller
FPS Character scripts
FX Scripts
Network Scripts



Server only

Control and execution by network

NPCs

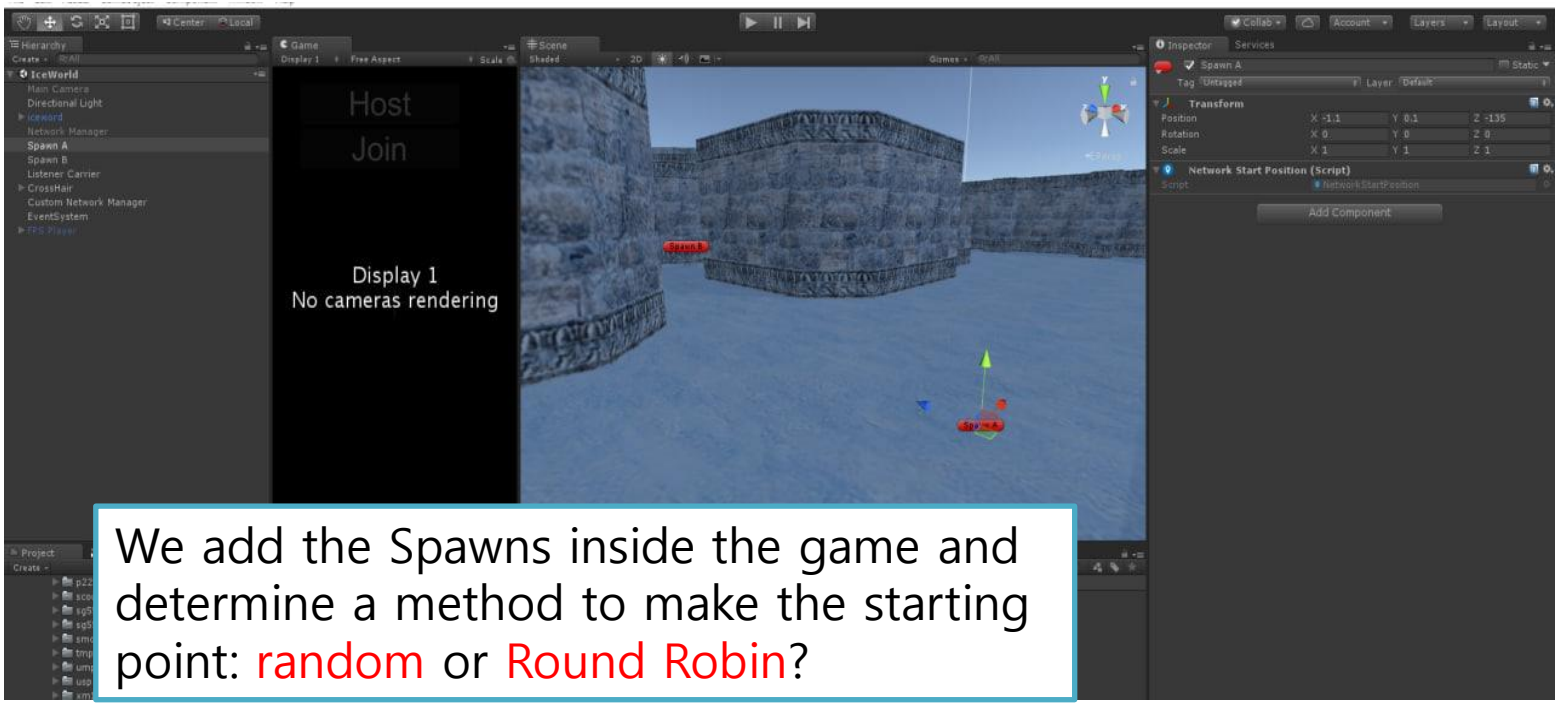
Local player

Object control by us

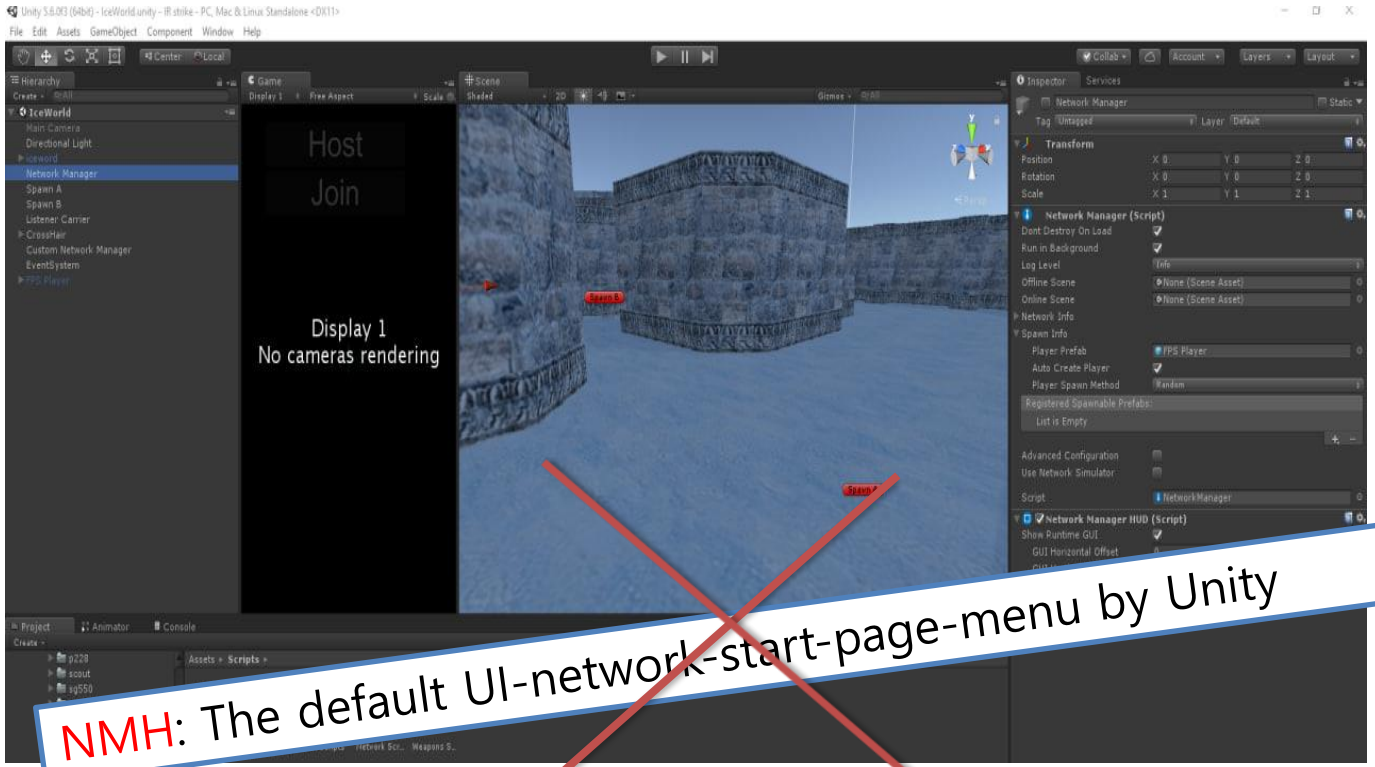
Real player

Spawns

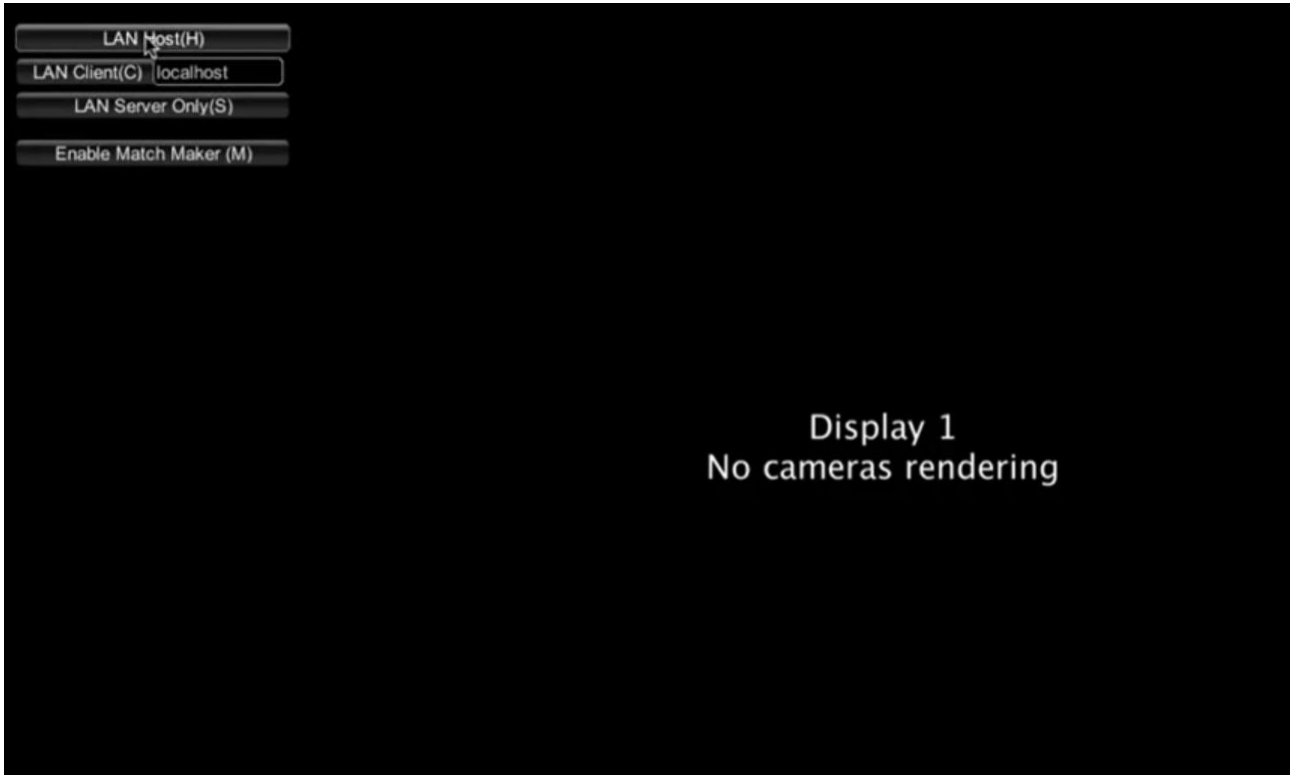
- ✓ It means that brings the player to life inside the game.
- ✓ They are starting points!
- ✓ Without them ➔ Throwing the player anywhere on the screen means the same bugs that we sometimes see!
- ✓ Starting players from a point has two methods:
  - ✓ **Random** ➔ Every time, any one can enter any spawn
  - ✓ **Round Robin** ➔ Each time, their place is changed in a certain order

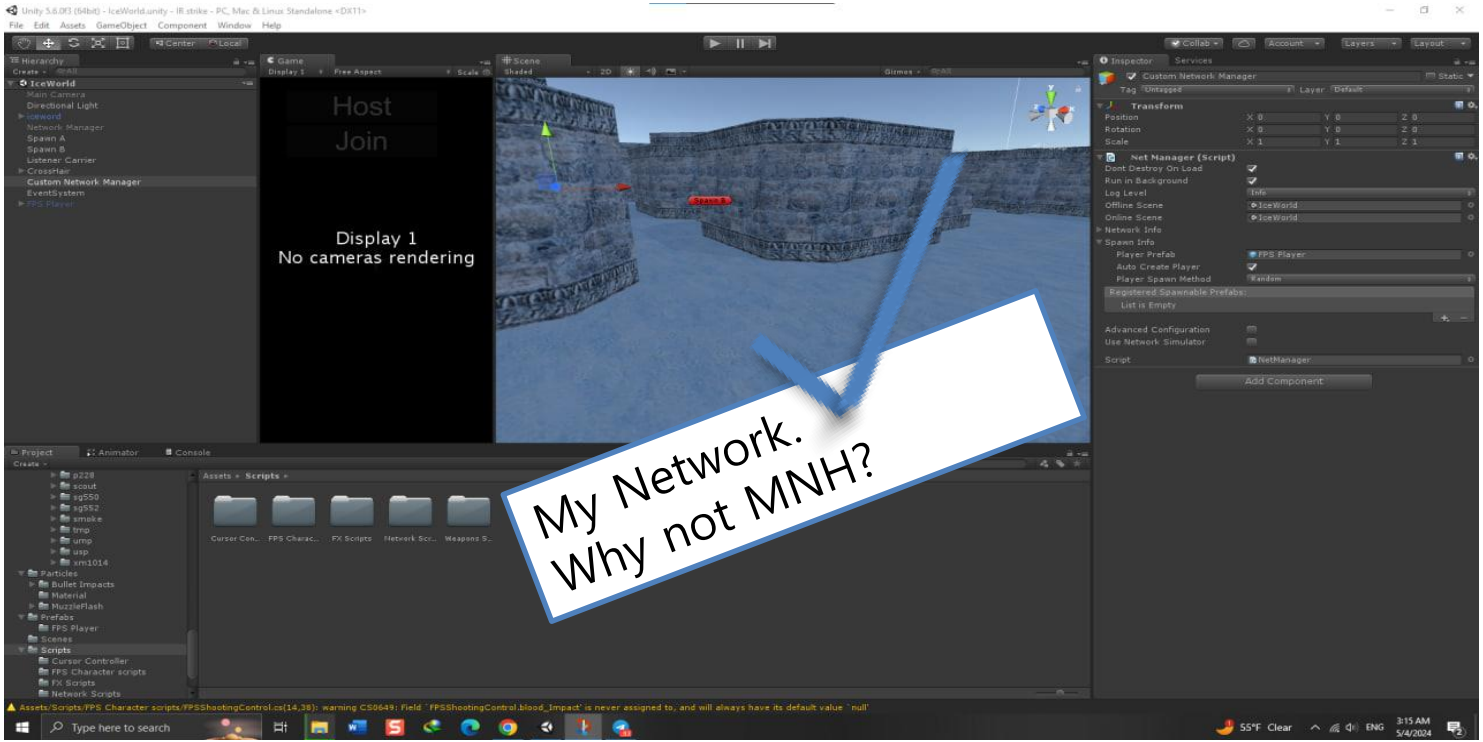


We add the Spawns inside the game and determine a method to make the starting point: random or Round Robin?

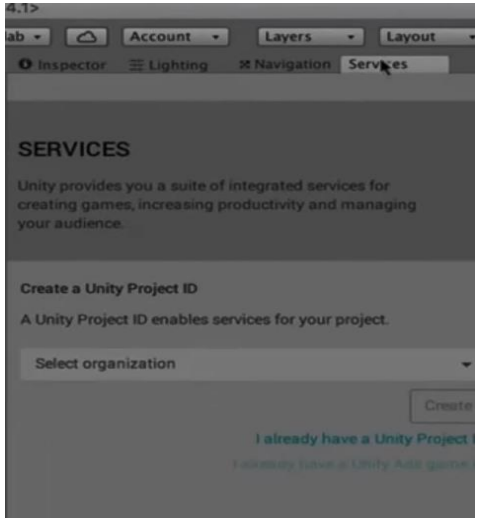

Host
Join
Display 1
No cameras rendering
NMH: The default UI-network-start-page-menu by Unity
Not Good!

• Network Manager Hub




Host
Join
Display 1
No cameras rendering
My Network.
Why not MNH?

- ✓ Due to being banned and unable to have a valid global account for international connection and building a global network.

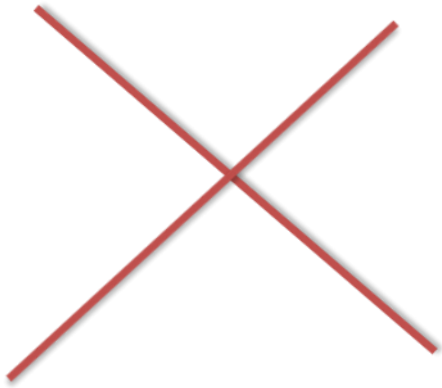


My Account
Projects
FPS Multiplayer Game
Ads
Analytics
Cloud Build
Collaborate
IAP
Performance Reporting
Multiplayer
FPS Multiplayer Game
Fahir Mehovic UPID 658459d2-9ad7-4c21-82c4-c8fed7ccbd33
Configuration
New Multiplayer Configuration: To enable multiplayer for this project please set the room size for each instance.
Max players per room

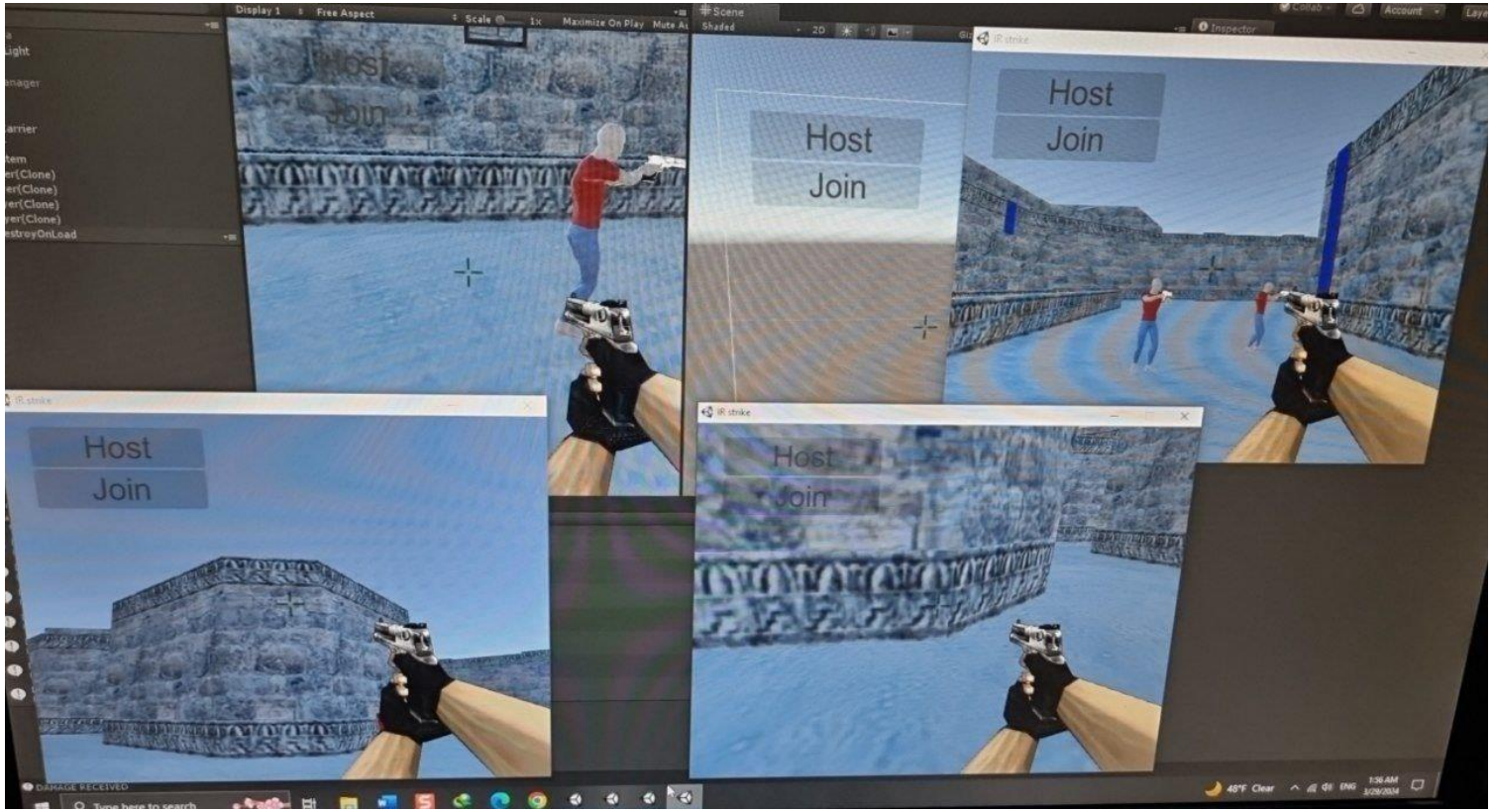
Royalty Free	✓	✓	✓
Splash Screen	MWU Splash Screen	Custom Animation or None	Custom Animation or None
Revenue Capacity	$100k	$200k	Unlimited
Unity Cloud Build	Standard Queue	Priority Queue	Concurrent Builds
Unity Analytics	Personal Analytics	Plus Analytics	Pro Analytics
Unity Multiplayer	20 Concurrent Users	50 Concurrent Users	200 Concurrent Users
Unity In-App Purchase		✓	✓
Unity Ads		✓	✓
Beta Access		✓	✓
Pro Editor UI Skin		✓	✓
Performance Reporting		✓	✓
Flexible Seat Management		✓	✓

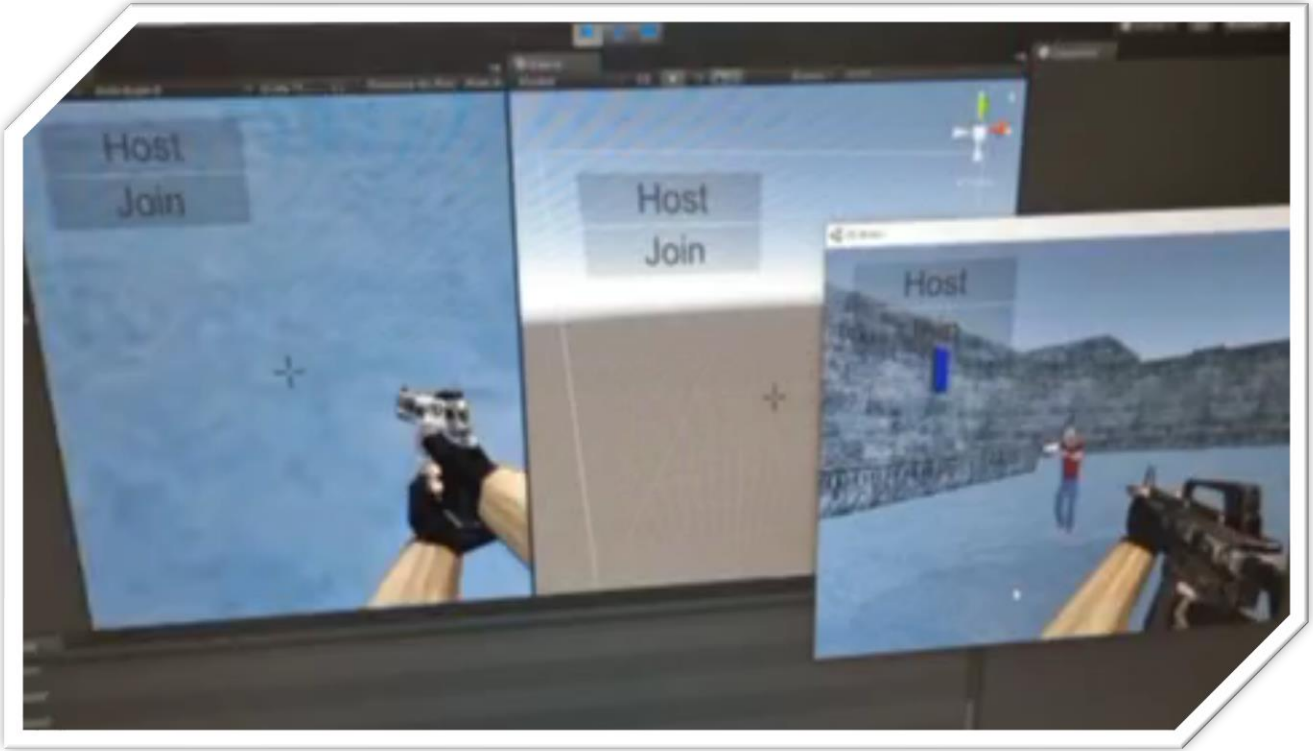
Host up to 20 simultaneous players with Unity. Great for prototyping and testing your game. Need more? Go Live for $0.49 per GB.

Create Internet Match
LAN Client(C)
Room Name:
default
LAN Server Only(S)
Find Internet Match
Change MM server
MM Uri: https://mm.unet.unity3d.com/
Disable Match Maker

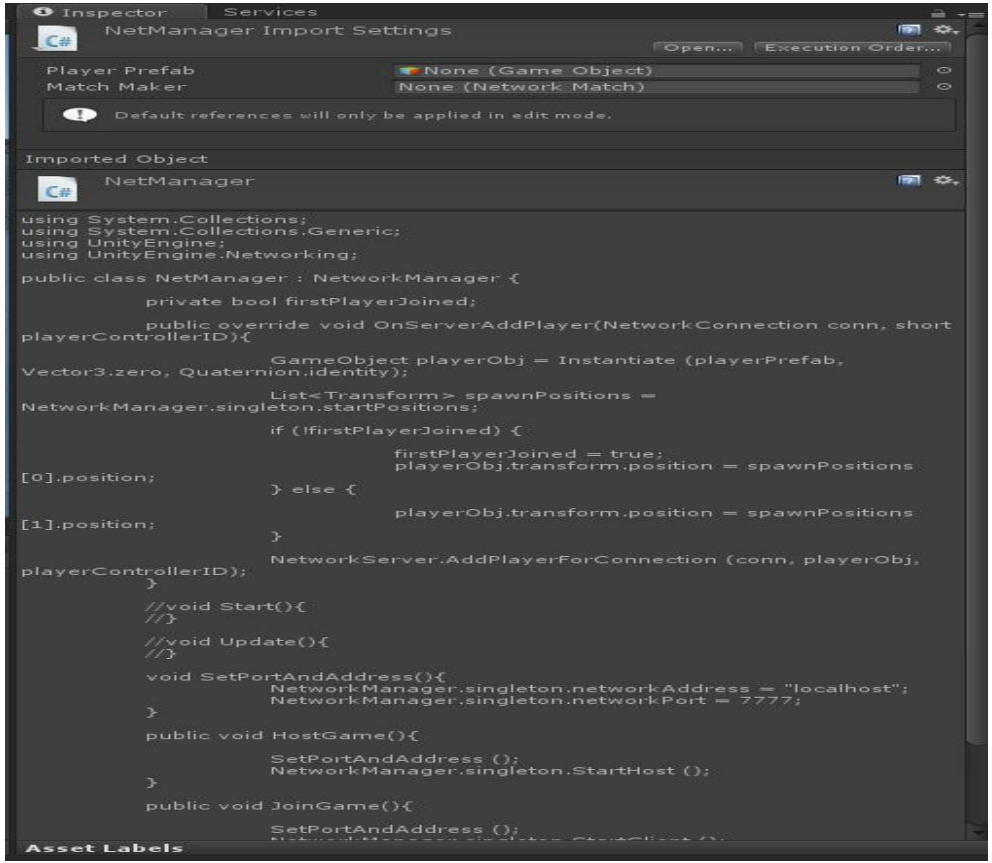


- My Network



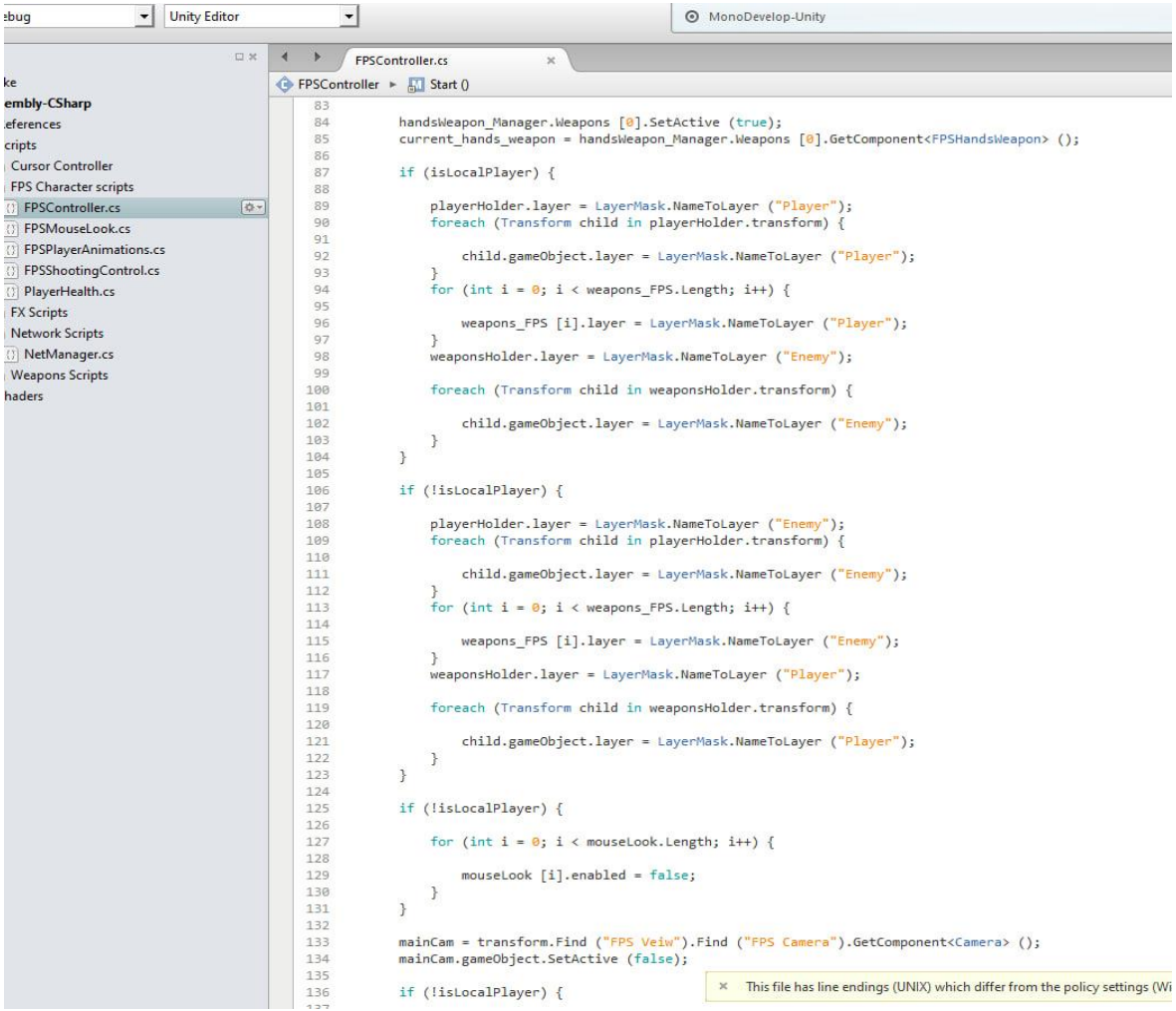

Host
Join
Host
Join
Host

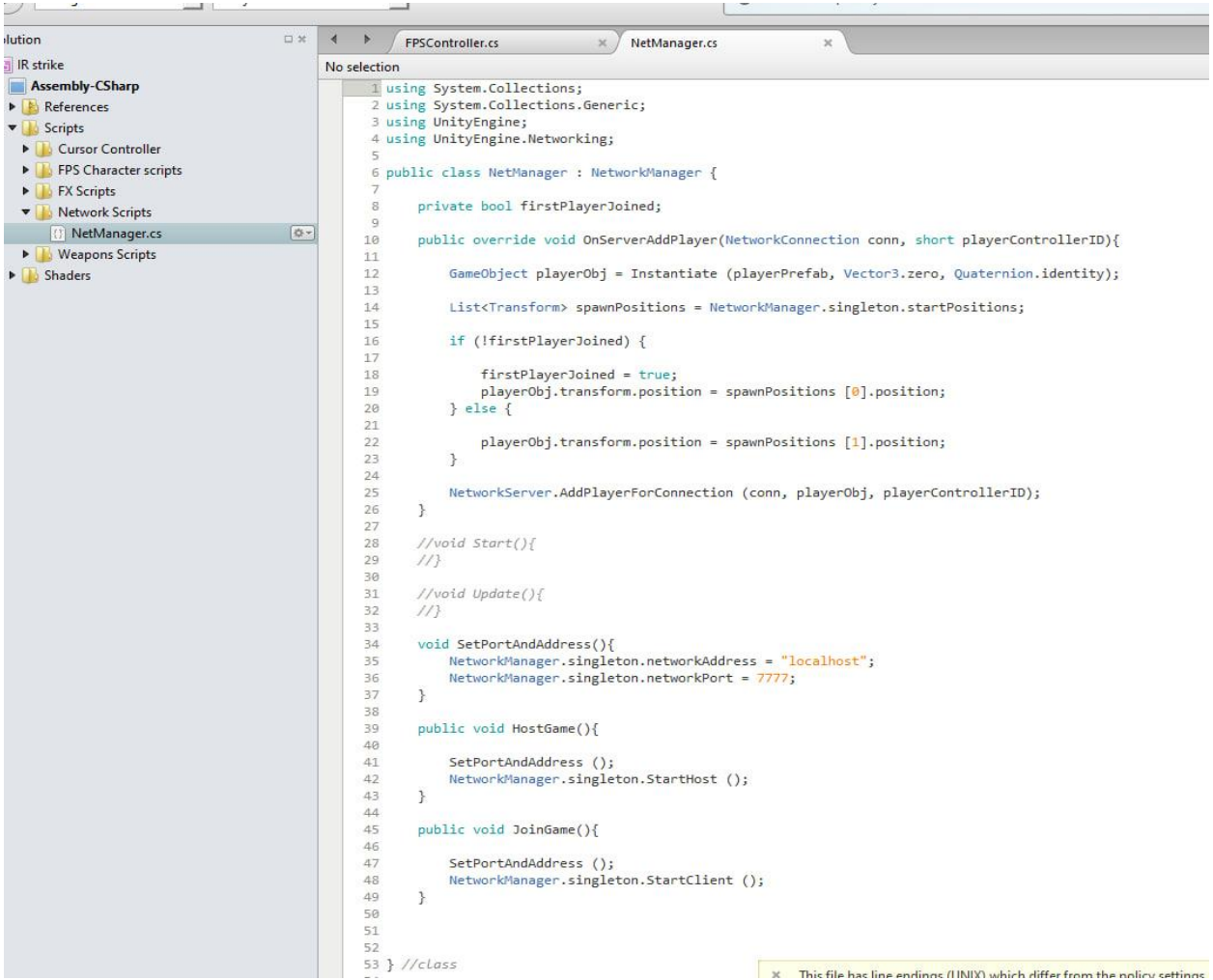
- Codes


Inspector
Services
NetManager Import Settings
Open...
Execution Order...
Player Prefab
None (Game Object)
Match Maker
None (Network Match)
Default references will only be applied in edit mode.
Imported Object
NetManager
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Networking;
public class NetManager : NetworkManager {
private bool firstPlayerJoined;
public override void OnServerAddPlayer(NetworkConnection conn, short playerControllerID){
GameObject playerObj = Instantiate (playerPrefab, Vector3.zero, Quaternion.identity);
List<Transform> spawnPositions = NetworkManager.singleton.startPositions;
if (!firstPlayerJoined) {
firstPlayerJoined = true;
playerObj.transform.position = spawnPositions [0].position;
} else {
playerObj.transform.position = spawnPositions [1].position;
}
NetworkServer.AddPlayerForConnection (conn, playerObj, playerControllerID);
}
//void Start(){
//}
//void Update(){
//}
void SetPortAndAddress(){
NetworkManager.singleton.networkAddress = "localhost";
NetworkManager.singleton.networkPort = 7777;
}
public void HostGame(){
SetPortAndAddress ();
NetworkManager.singleton.StartHost ();
}
public void JoinGame(){
SetPortAndAddress ();
Asset Labels

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

public class NetManager : NetworkManager {

		private bool firstPlayerJoined;

		public override void OnServerAddPlayer(NetworkConnection conn, short
playerControllerID){

				GameObject playerObj = Instantiate (playerPrefab,
Vector3.zero, Quaternion.identity);

				List<Transform> spawnPositions =
NetworkManager.singleton.startPositions;

				if (!firstPlayerJoined) {

						firstPlayerJoined = true;
						playerObj.transform.position = spawnPositions
[0].position;
				} else {

						playerObj.transform.position = spawnPositions
[1].position;
				}

				NetworkServer.AddPlayerForConnection (conn, playerObj,
playerControllerID);
		}

		//void Start(){
		//}

		//void Update(){
		//}

		void SetPortAndAddress(){
				NetworkManager.singleton.networkAddress = "localhost";
				NetworkManager.singleton.networkPort = 7777;
		}

		public void HostGame(){

				SetPortAndAddress ();
				NetworkManager.singleton.StartHost ();
		}

		public void JoinGame(){

				SetPortAndAddress ();
				NetworkManager.singleton.StartClient ();
		}


} //class
```

Asset Labels


Unity Editor
MonoDevelop-Unity
FPSController.cs
FPSController ▸ Start ()
embly-CSharp
eferences
cripts
Cursor Controller
FPS Character scripts
FPSController.cs
FPSMouseLook.cs
FPSPlayerAnimations.cs
FPSShootingControl.cs
PlayerHealth.cs
FX Scripts
Network Scripts
NetManager.cs
Weapons Scripts
haders
83 handsWeapon_Manager.Weapons [0].SetActive (true);
84 current_hands_weapon = handsWeapon_Manager.Weapons [0].GetComponent<FPSHandsWeapon> ();
85
86 if (isLocalPlayer) {
87
88 playerHolder.layer = LayerMask.NameToLayer ("Player");
89 foreach (Transform child in playerHolder.transform) {
90
91 child.gameObject.layer = LayerMask.NameToLayer ("Player");
92 }
93 for (int i = 0; i < weapons_FPS.Length; i++) {
94
95 weapons_FPS [i].layer = LayerMask.NameToLayer ("Player");
96 }
97 weaponsHolder.layer = LayerMask.NameToLayer ("Enemy");
98
99 foreach (Transform child in weaponsHolder.transform) {
100
101 child.gameObject.layer = LayerMask.NameToLayer ("Enemy");
102 }
103 }
104
105 if (!isLocalPlayer) {
106
107 playerHolder.layer = LayerMask.NameToLayer ("Enemy");
108 foreach (Transform child in playerHolder.transform) {
109
110 child.gameObject.layer = LayerMask.NameToLayer ("Enemy");
111 }
112 for (int i = 0; i < weapons_FPS.Length; i++) {
113
114 weapons_FPS [i].layer = LayerMask.NameToLayer ("Enemy");
115 }
116 weaponsHolder.layer = LayerMask.NameToLayer ("Player");
117
118 foreach (Transform child in weaponsHolder.transform) {
119
120 child.gameObject.layer = LayerMask.NameToLayer ("Player");
121 }
122 }
123
124 if (!isLocalPlayer) {
125
126 for (int i = 0; i < mouseLook.Length; i++) {
127
128 mouseLook [i].enabled = false;
129 }
130 }
131
132 mainCam = transform.Find ("FPS Veiw").Find ("FPS Camera").GetComponent<Camera> ();
133 mainCam.gameObject.SetActive (false);
134
135 if (!isLocalPlayer) {
136
This file has line endings (UNIX) which differ from the policy settings (Wi


lution
IR strike
Assembly-CSharp
References
Scripts
Cursor Controller
FPS Character scripts
FX Scripts
Network Scripts
NetManager.cs
Weapons Scripts
Shaders
FPSController.cs
NetManager.cs
No selection
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.Networking;
5
6 public class NetManager : NetworkManager {
7
8 private bool firstPlayerJoined;
9
10 public override void OnServerAddPlayer(NetworkConnection conn, short playerControllerID){
11
12 GameObject playerObj = Instantiate (playerPrefab, Vector3.zero, Quaternion.identity);
13
14 List<Transform> spawnPositions = NetworkManager.singleton.startPositions;
15
16 if (!firstPlayerJoined) {
17
18 firstPlayerJoined = true;
19 playerObj.transform.position = spawnPositions [0].position;
20 } else {
21
22 playerObj.transform.position = spawnPositions [1].position;
23 }
24
25 NetworkServer.AddPlayerForConnection (conn, playerObj, playerControllerID);
26 }
27
28 //void Start(){
29 //}
30
31 //void Update(){
32 //}
33
34 void SetPortAndAddress(){
35 NetworkManager.singleton.networkAddress = "localhost";
36 NetworkManager.singleton.networkPort = 7777;
37 }
38
39 public void HostGame(){
40
41 SetPortAndAddress ();
42 NetworkManager.singleton.StartHost ();
43 }
44
45 public void JoinGame(){
46
47 SetPortAndAddress ();
48 NetworkManager.singleton.StartClient ();
49 }
50
51
52
53 } //class
This file has line endings (UNIX) which differ from the policy settings

Thanks for your attention!