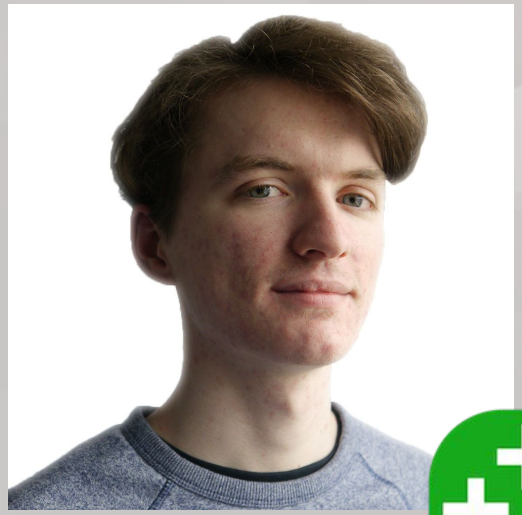
GameDev.tv

Unity 2020.1

Importing Mirror



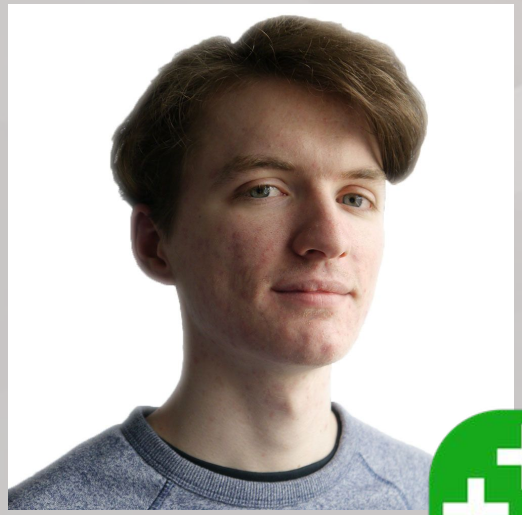
Import The Remaining Packages

- Cinemachine
- Input System
- ProBuilder
- ProGrids (enable preview packages)
- TextMeshPro
- Unity UI
- Universal RP
- Visual Studio Code Editor

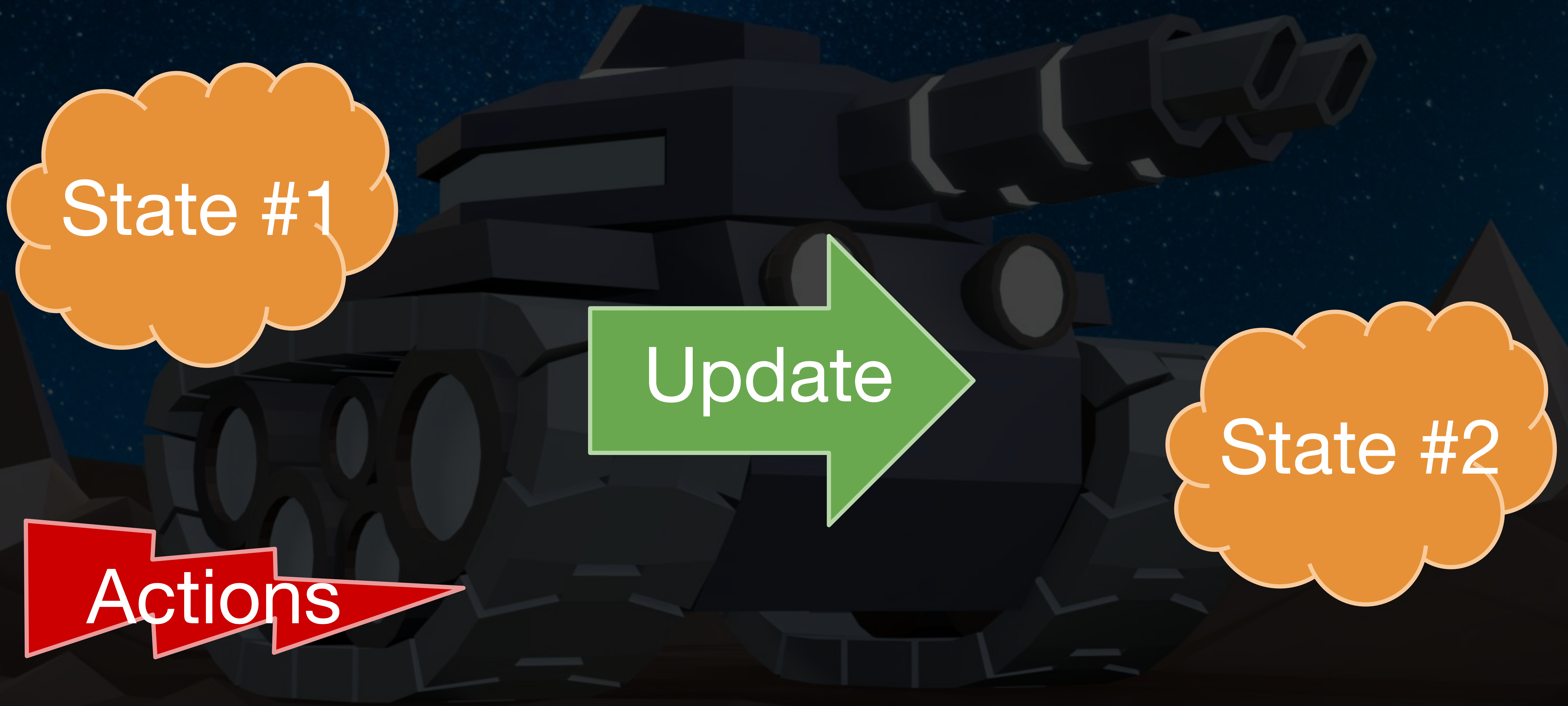
GameDev.tv

Unity 2020.1

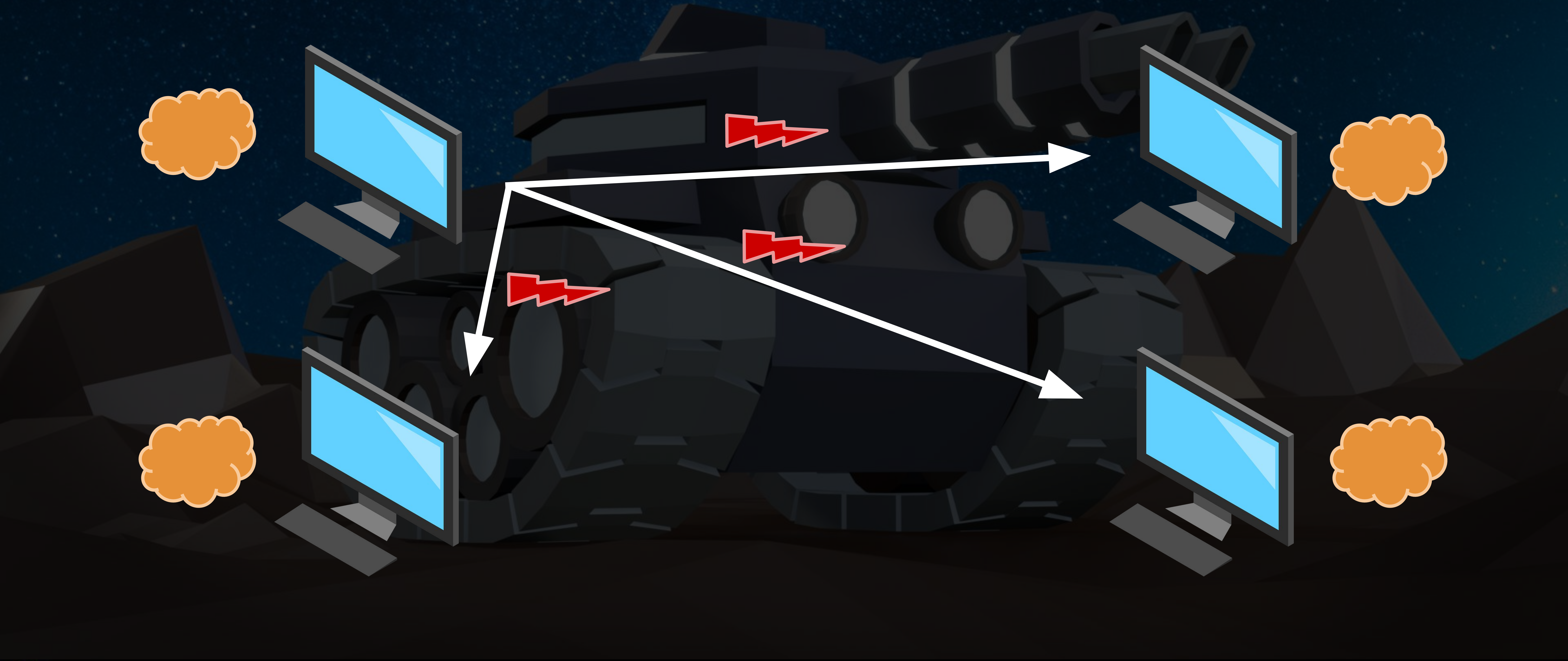
Client-Server Model



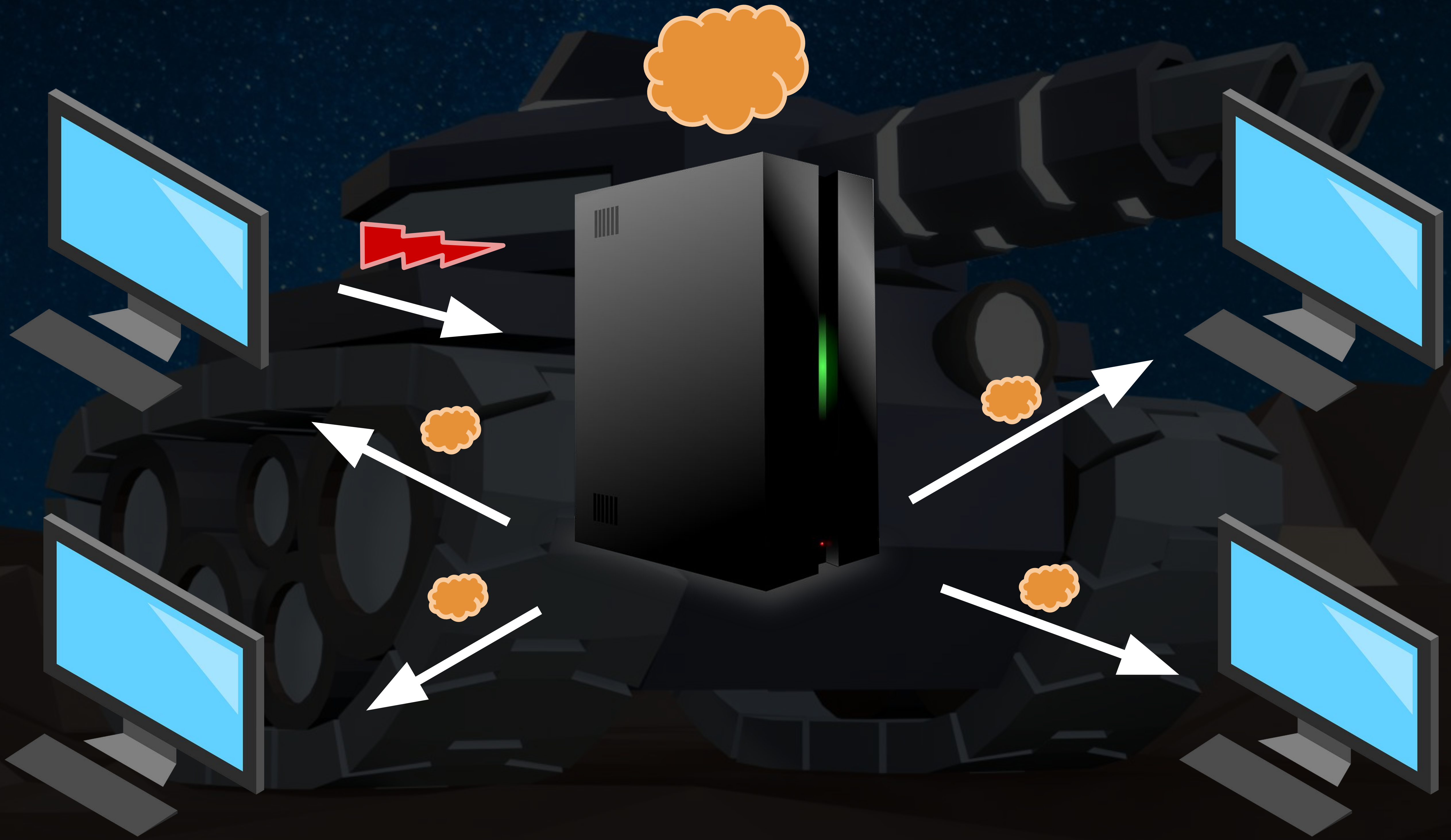
Input And State



Peer-To-Peer

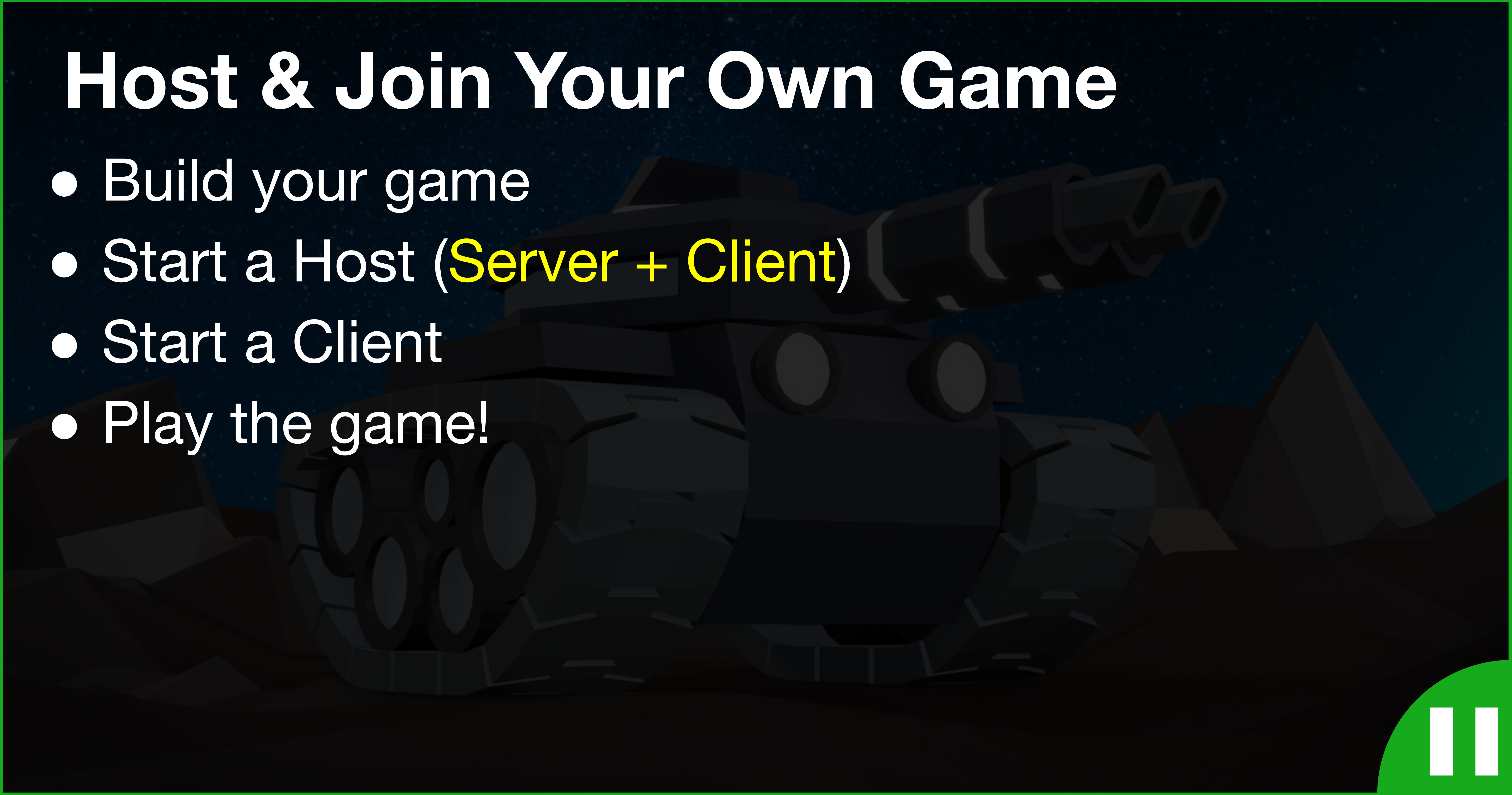


Client-Server



Host & Join Your Own Game

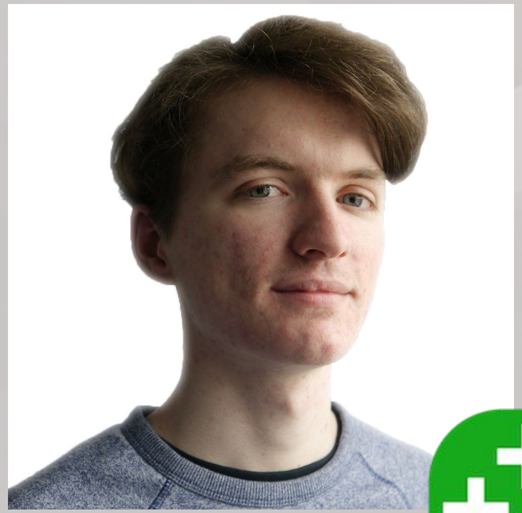
- Build your game
- Start a Host (Server + Client)
- Start a Client
- Play the game!



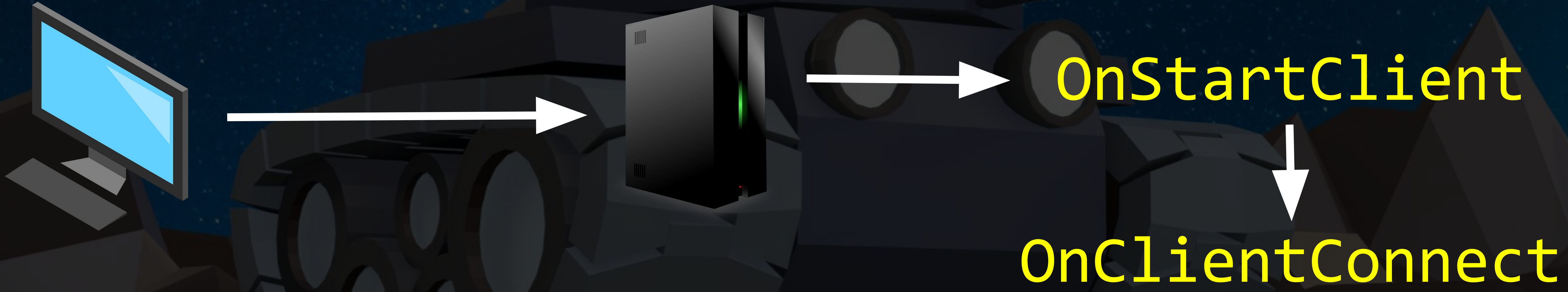
Unity 2020.1

z

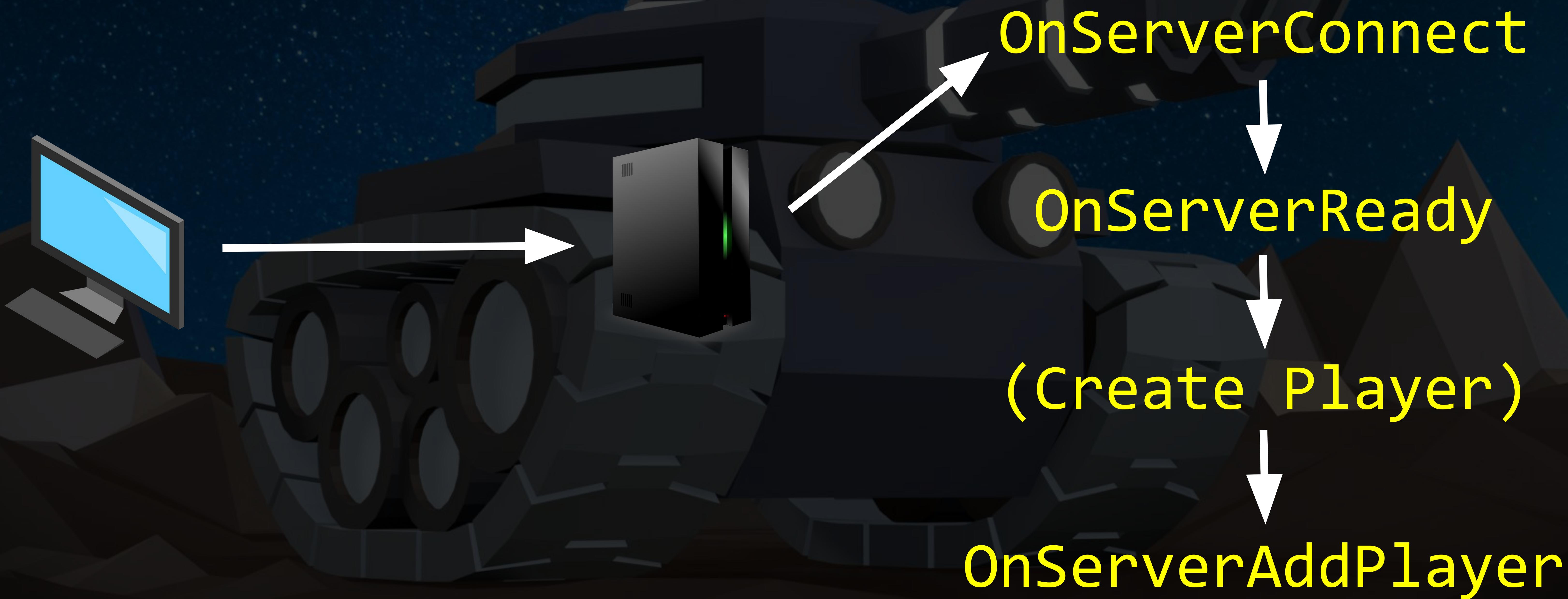
Network Manager



When A Client Connects (Client)



When A Client Connects (Server)



Log Whenever A Player Is Added

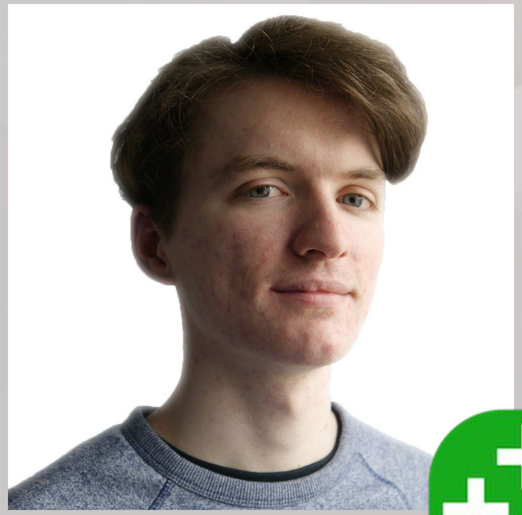
- Override the `OnServerAddPlayer` method
- Call the base method
- Use the `Debug.Log` method to write a message to the console
- You can use the `numPlayers` property to get the number of active connections to the server

GameDev.tv

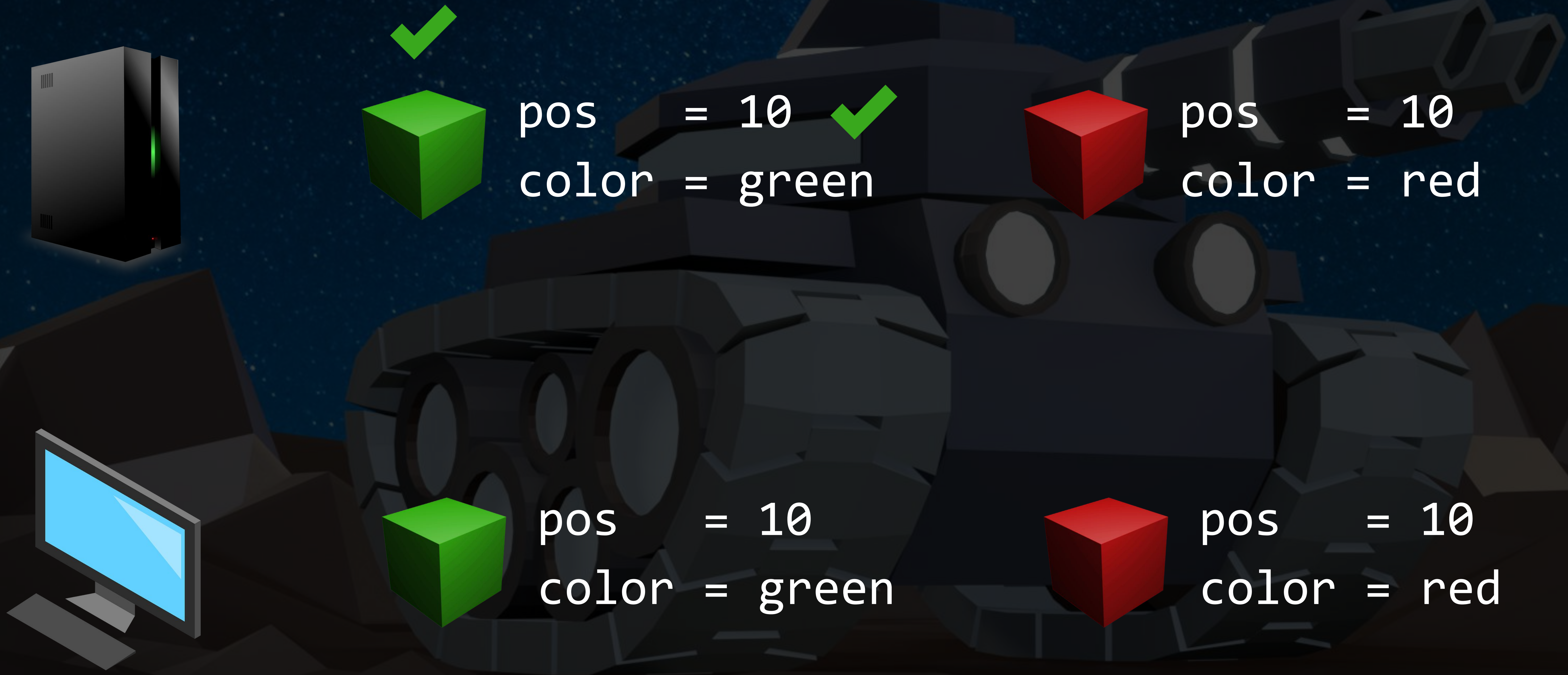
Unity 2020.1

z

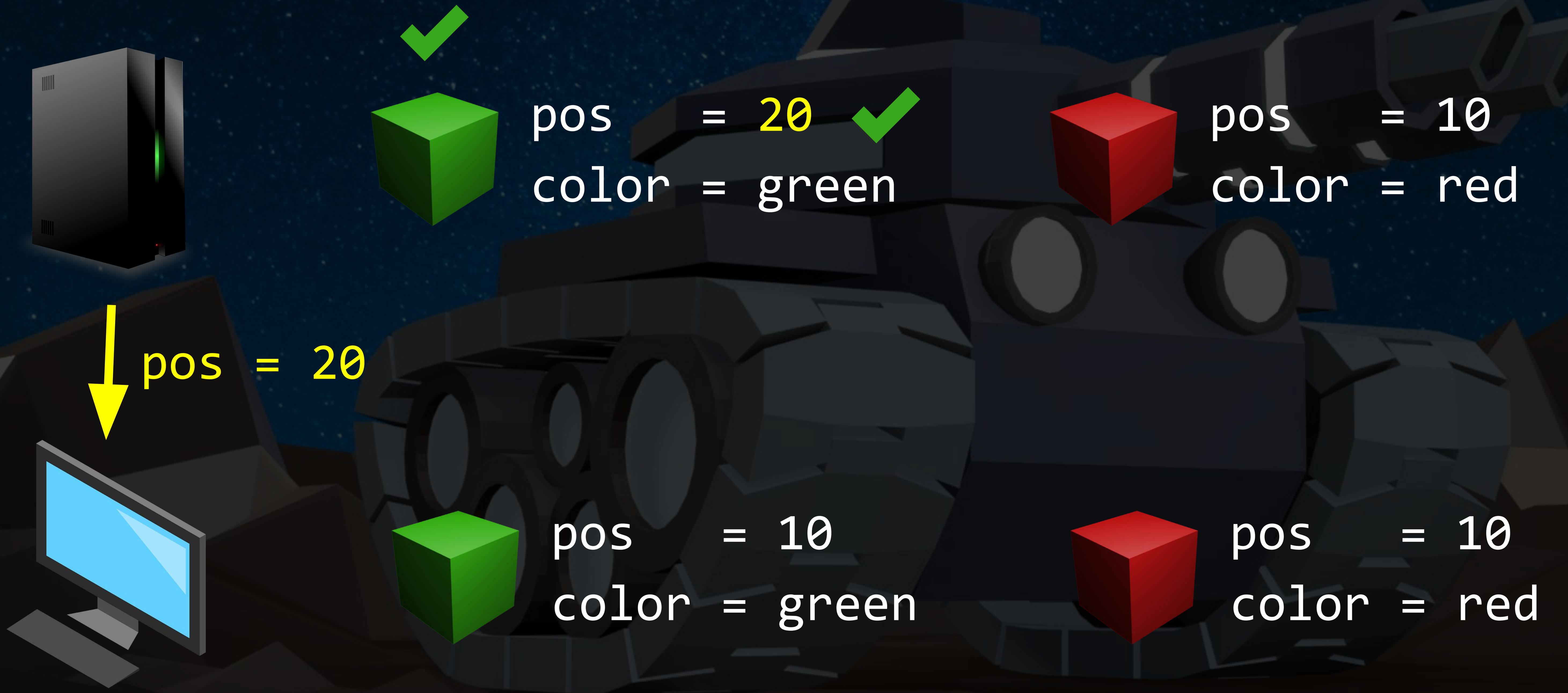
Syncing Variables



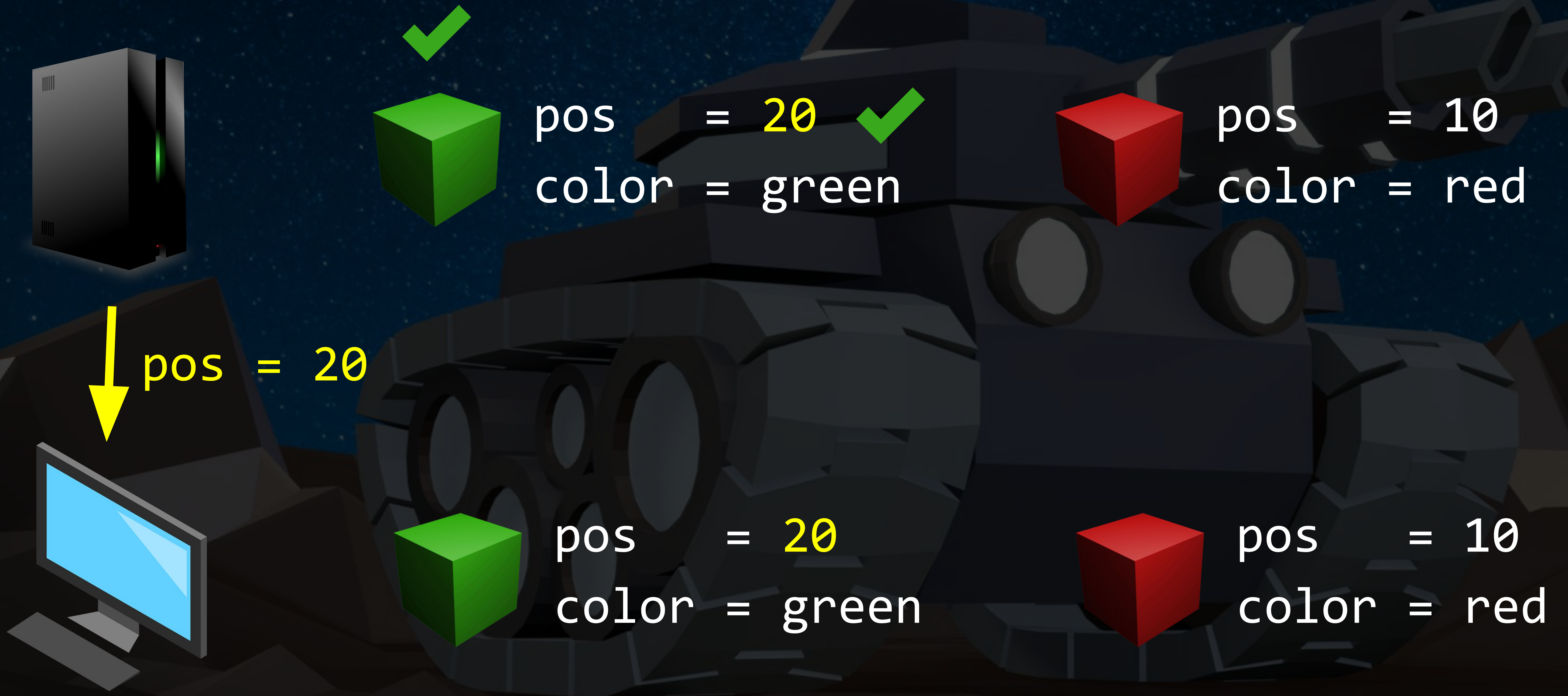
Network Behaviours & SyncVars



Network Behaviours & SyncVars



Network Behaviours & SyncVars



Assign & Sync The Player’s Colour

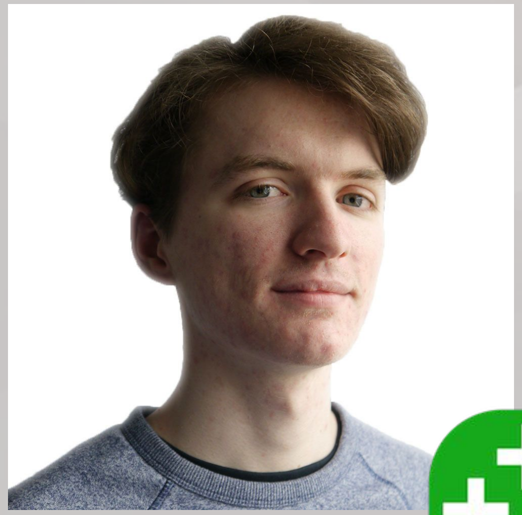
- Create the colour variable using the `Color` type
- Add the `[SyncVar]` attribute
- Assign each player a random colour on the server

- HINT: `new Color(`randomly generate each of the R, G, B values between 0 and 1`);`
- HINT: `Random.Range(0f, 1f);`

Unity 2020.1

z

SyncVar Callbacks

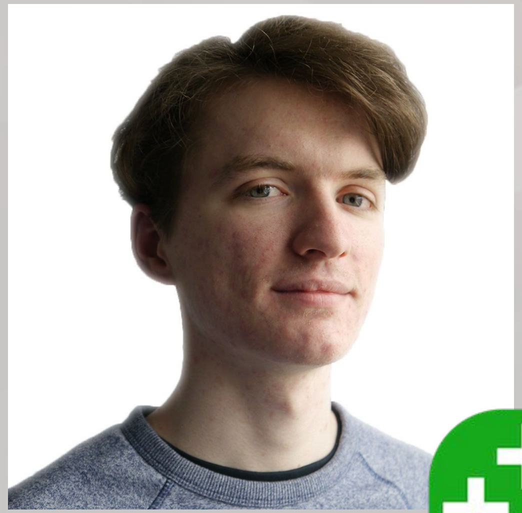


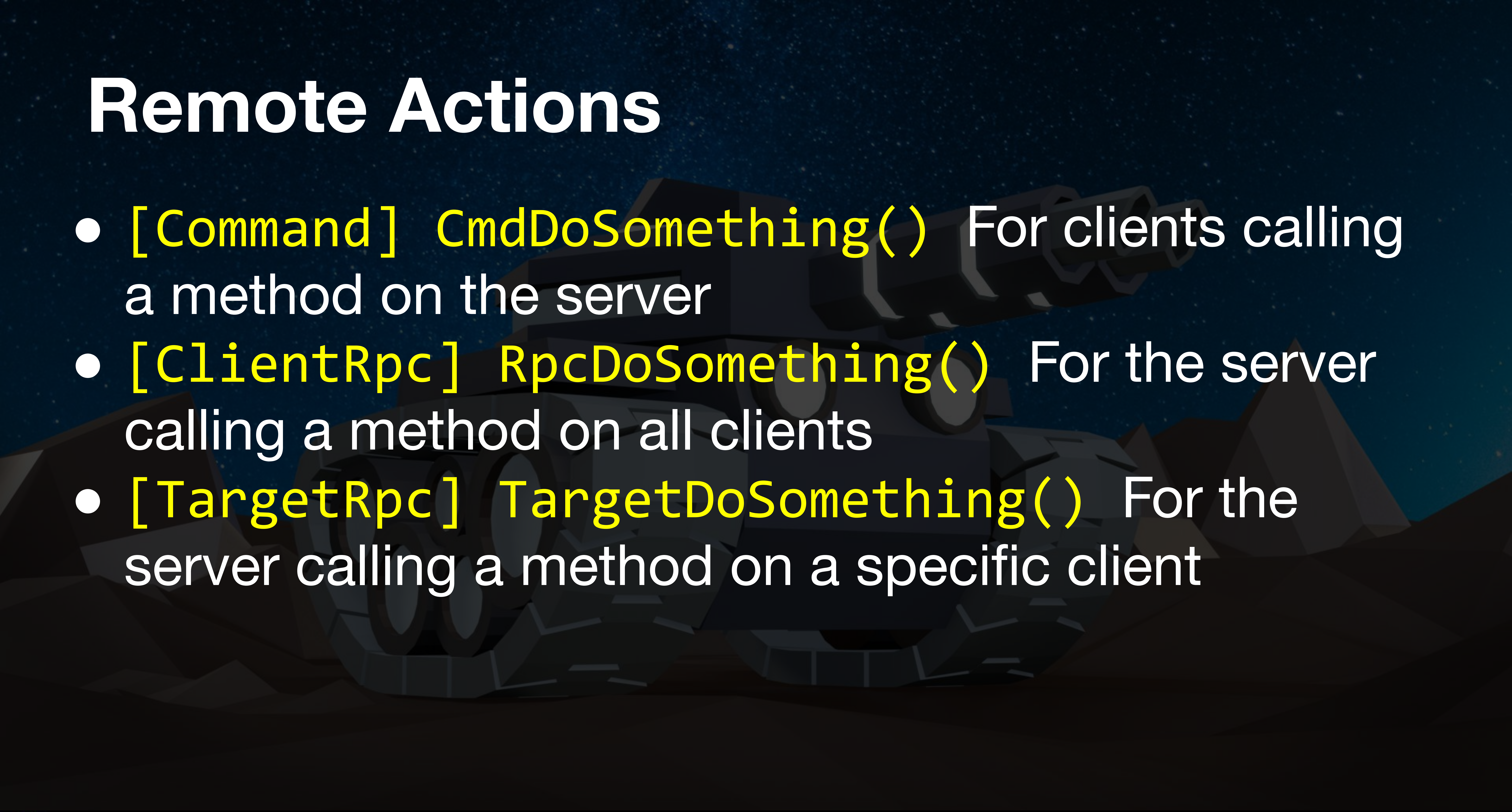
Update The Player’s Name UI

- Create a method to be used as the callback
- Add a hook, `[SyncVar(hook=nameof(Method))]` to that method
- Make the method change the player’s UI text `(displayNameText.text)` to the new name that was just sent from the server

Unity 2020.1

Remote Actions





Remote Actions

- `[Command] CmdDoSomething()` For clients calling a method on the server
- `[ClientRpc] RpcDoSomething()` For the server calling a method on all clients
- `[TargetRpc] TargetDoSomething()` For the server calling a method on a specific client

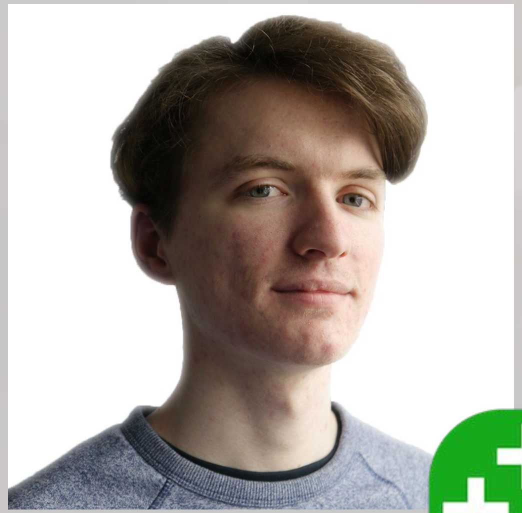
Call A Method On Each Client

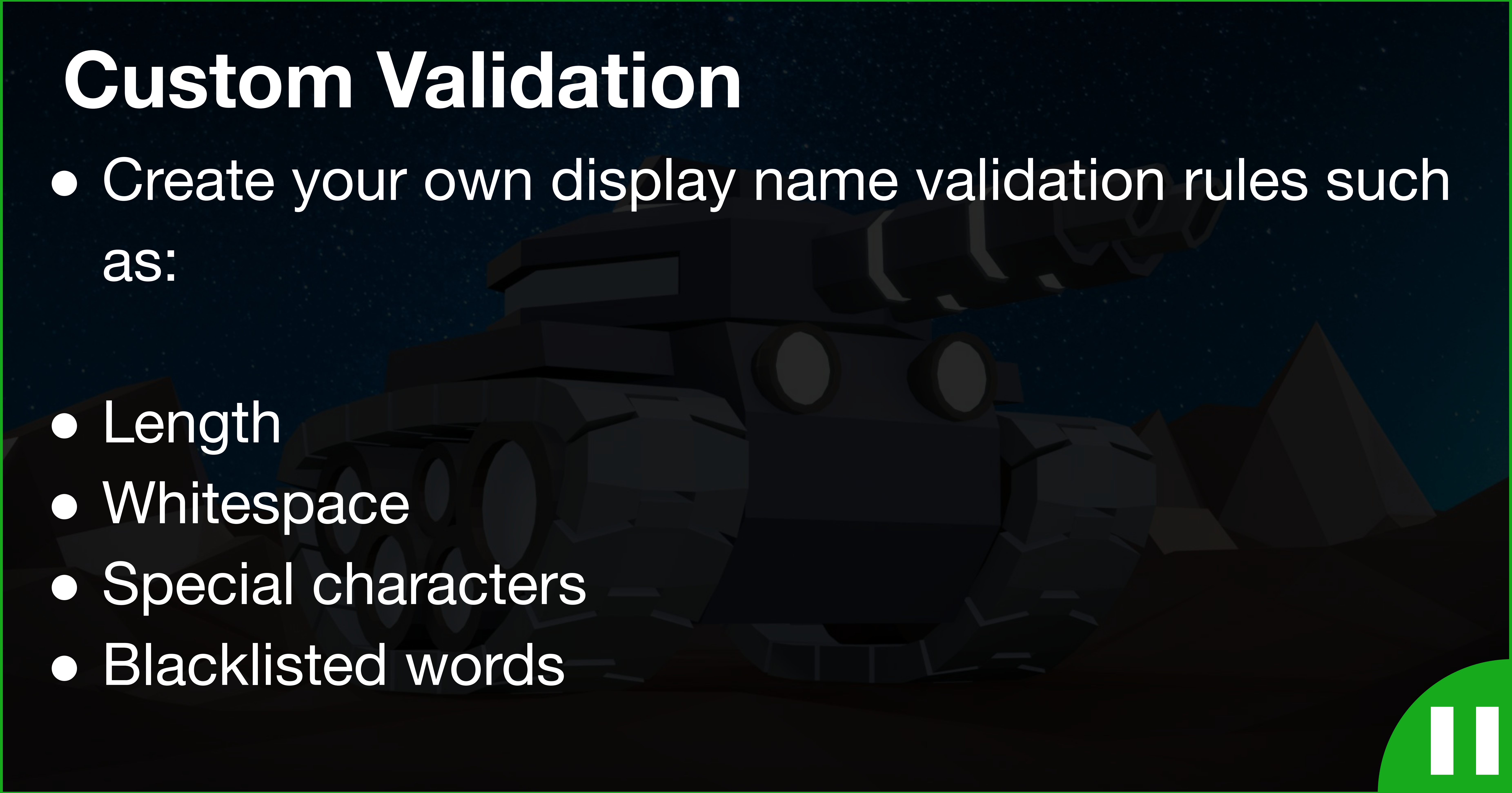
- Create a method beginning with `Rpc` that takes in a `string` as a parameter for the new name
- Log that new name to their console
- Use the `[ClientRpc]` attribute
- Call the method from the server when setting a name

Unity 2020.1

z

Server Authority





Custom Validation

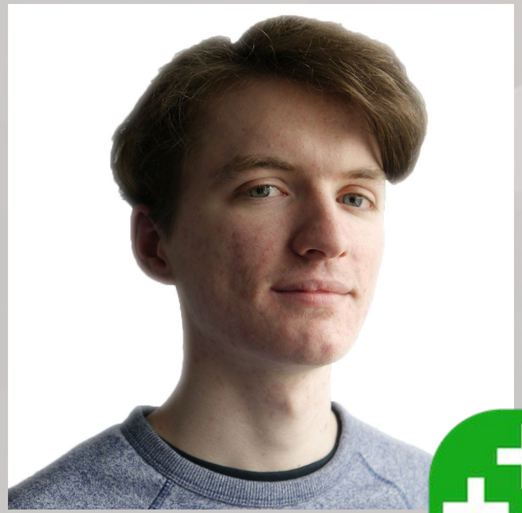
- Create your own display name validation rules such as:

- Length
- Whitespace
- Special characters
- Blacklisted words

GameDev.tv

Unity 2020.1

Network Transform



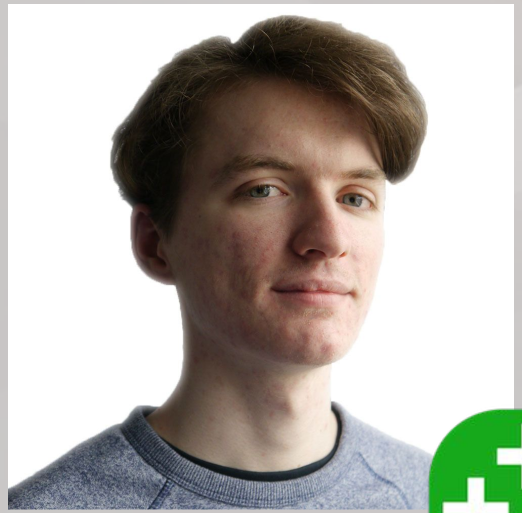
Make The Level More Interesting

- Add some boxes
- Add some walls
- Set up the spawn points using the `NetworkStartPosition` component

GameDev.tv

Unity 2020.1

Moving Our Player



Set Up The Player Movement Script

- Create a new C# script
- Make sure it inherits from `NetworkBehaviour`
- Create an empty Move `[Command]` method

- `CmdMove()`