



Advanced Multiplayer Template

Thank you for purchasing the **Advanced Multiplayer Template!**

Within this **documentation**, you'll discover instructions and descriptions pertaining to the **Advanced Multiplayer Template**.

If you need **help** or have any **questions**, please feel free to **contact** me at the following **email address**: Contact@RedicionStudio.com

Table of contents

1. Setup.....	
2. Inventory.....	
3. Sorting, splitting, dropping and cooldowns of inventory items.....	
4. Item Shop.....	
5. Item rarity.....	
6. Player Outfit.....	
7. Companions.....	
8. Game World.....	
9. Build System.....	
10. Vehicles.....	
11. Adding a new account data entry.....	
12. Mobile Compatibility.....	
13. Master Server and Game Server.....	
14. Adding a new account data entry.....	
15. User Authentication.....	
16. Server Creation and List.....	
17. Run Master and Game Server on a VPS.....	
18. Hotkeys.....	

Setup

Below, you'll find several **setup videos** demonstrating how to set up the **Advanced Multiplayer Game Template** in Unity versions **2020.3.33**, **2021.x**, and **2022.x**.

2020.3.33



2021.x

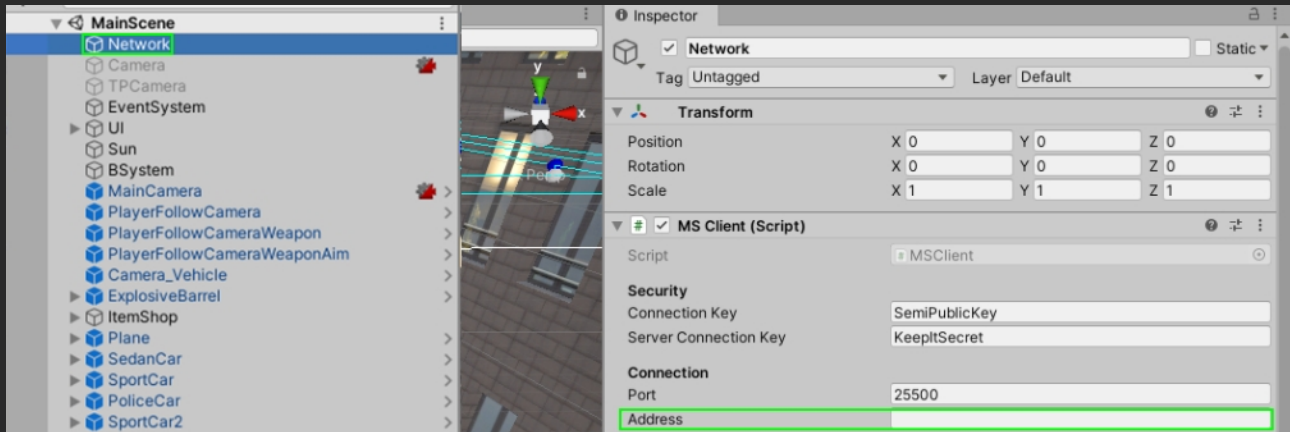


2022.x

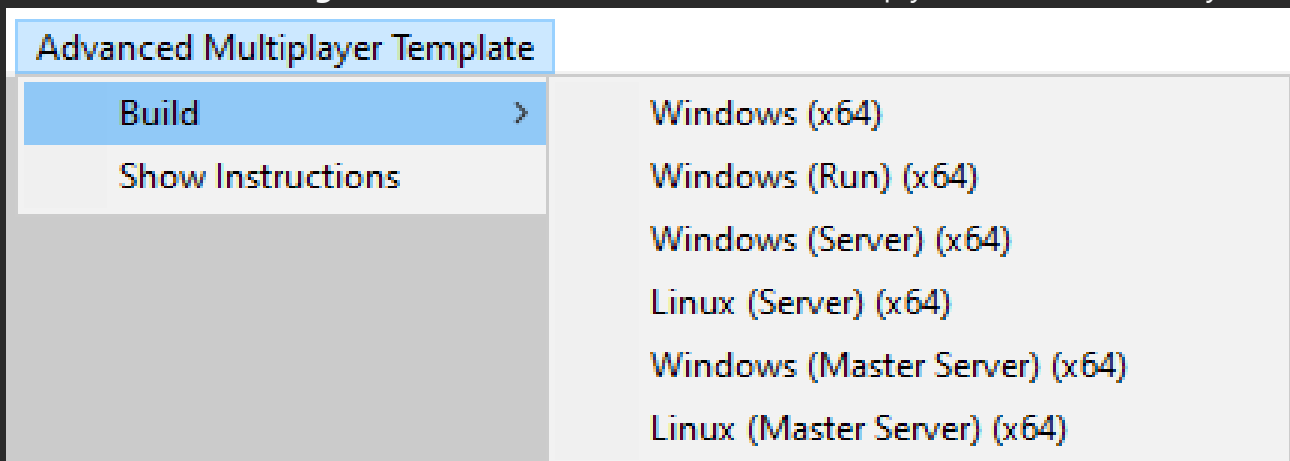


Setup and run the Master and Game Server

Open the "**MainScene.unity**" scene (*Assets/Advanced Multiplayer Template/Scenes/MainScene.unity*) and select the "**Network**" game object in the scene hierarchy. Insert your **IP address** into the "**Address**" variable of the "**MSClient.cs**" script and **save** the **scene**. For testing on your **local computer**, you can use the **localhost address** "**127.0.0.1**" or your **IPv4 address**.



Build the **master** and **game server** for **Windows** or **Linux** simply via the "**Build**" entry.



After you have built the **master** and the **game server**, run the **master server** first and then the **game server**. When both **servers** are fully started, run the client by entering the **play mode** in the editor when you are in the "**MainScene.unity**" scene or **build** and **run** the **client** via the "**Build**" entry.

After you have launched the **client**, you will need to **authenticate**, which means you must create an **account**. Please refer to the "**User Authentication**" section in this **guide** to learn how to **create an account**.

[Click here](#) to go to the "**User Authentication**" section.

Inventory

The **inventory** allows the player to **collect**, **move**, **split**, **swap**, **drop** and **equip** items.

Add a new weapon item



This video guide explains how to **add a new weapon** as an item so that it can be spawned by the **ItemSpawner**, collected and equipped by the player, and stored and retrieved from the **Inventory**.

Add a new ammo and consumable item

This video guide explains how to **add a new ammo and consumable item** so that it can be spawned by the **ItemSpawner**, collected, equipped and consumed by the player or other item, and stored and retrieved from the **Inventory**.



Sorting, splitting, dropping and cooldowns of inventory items

The **inventory** allows you to **move**, **swap**, **split** and **drop** items.

This video shows the **moving**, **swapping**, **splitting** and **dropping** of items. It also shows items that have a **cooldown**:



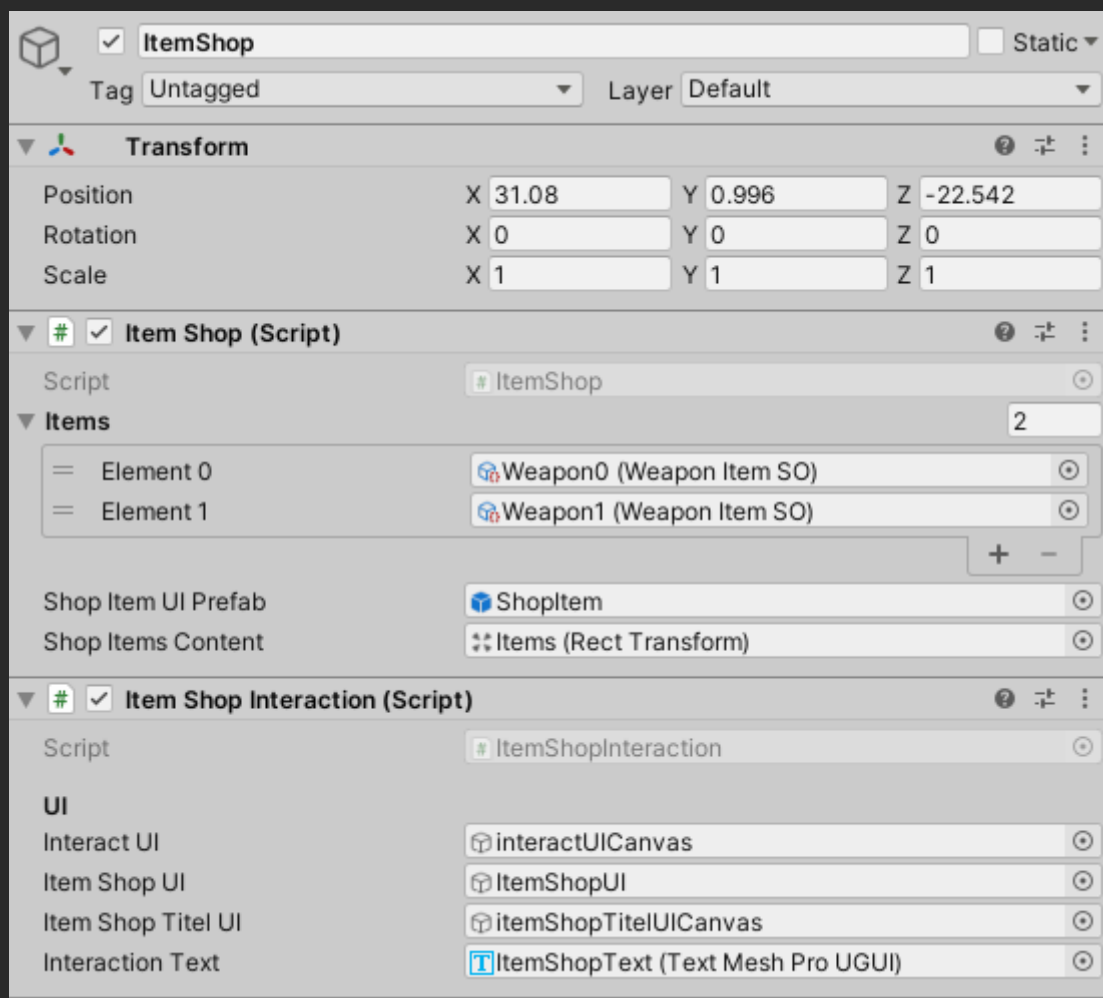
Item Shop

The **item shop** allows players to purchase registered items.

Add an item to the item shop

This guide explains how to **add an item** to the **item shop**.

Drag an **AmmolItemSO**, **ConsumableItemSO**, **WeaponItemSO**, **CompanionItemSO** or **OutfitItemSO** into the "Items" array shown in the picture below.



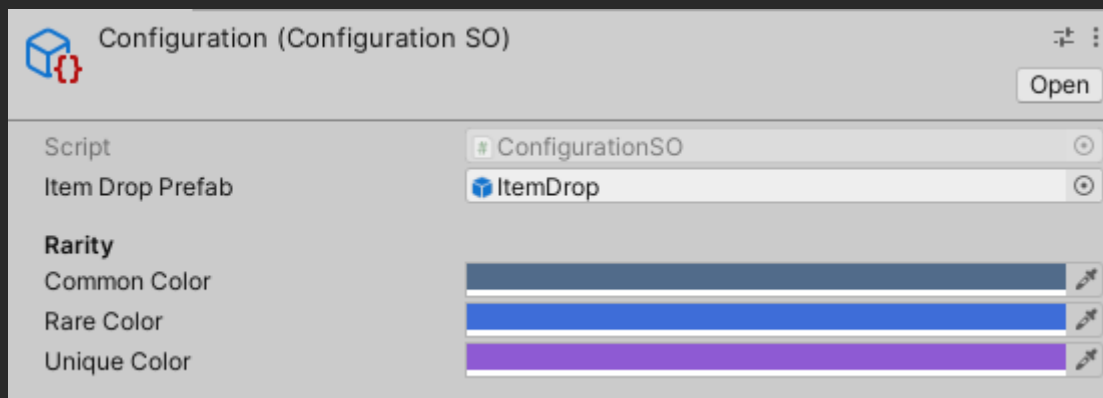
After you add an **itemSO** to the "Items" array, players can purchase this item.

Item rarity

Change the display color of the rarity levels

This guide explains how to **change the display color of the rarity levels**.

To change the display color of the respective rarity level, go to the "**Assets/InventorySystem/Resources/**" path and select the "**Configuration.asset**" and there you can change the colors.



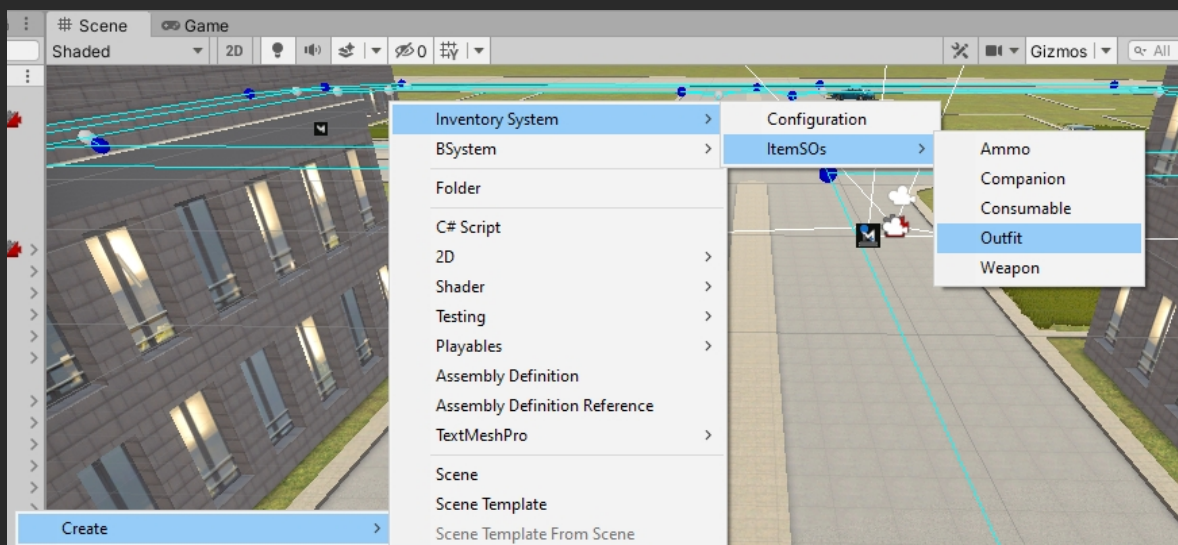
Player Outfits

Outfit items can be placed in the **outfit** slot in the **inventory** to change the appearance of the **player character**.

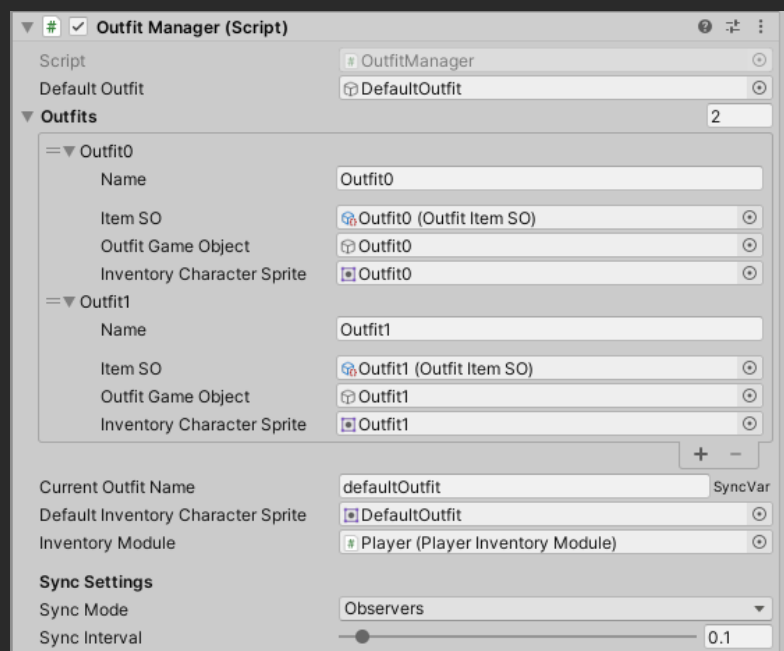
Add an outfit item

This guide explains how to **add an outfit item**.

Create an **OutfitItemSO**. Creating an **OutfitItemSO** is almost the same as creating any other **itemSO**.



After you have created the **OutfitItemSO**, specified the unique name, model prefab and sprite, open the "**Player.prefab**". After you have opened the "**Player.prefab**", go to the "**OutfitManager.cs**" attached to the root of the "**player.prefab**" and add a new entry in the "**Outfits**" array. It is important that the name of the entry is identical to the name you entered in the **outfit ItemSO**.



Next, specify the created **OutfitItemSO** in the „**ItemSO**“ variable of the array entry. The new player character outfit to be activated when equipping the **outfit item** must be specified in the „**Outfit Game Object**“ variable.

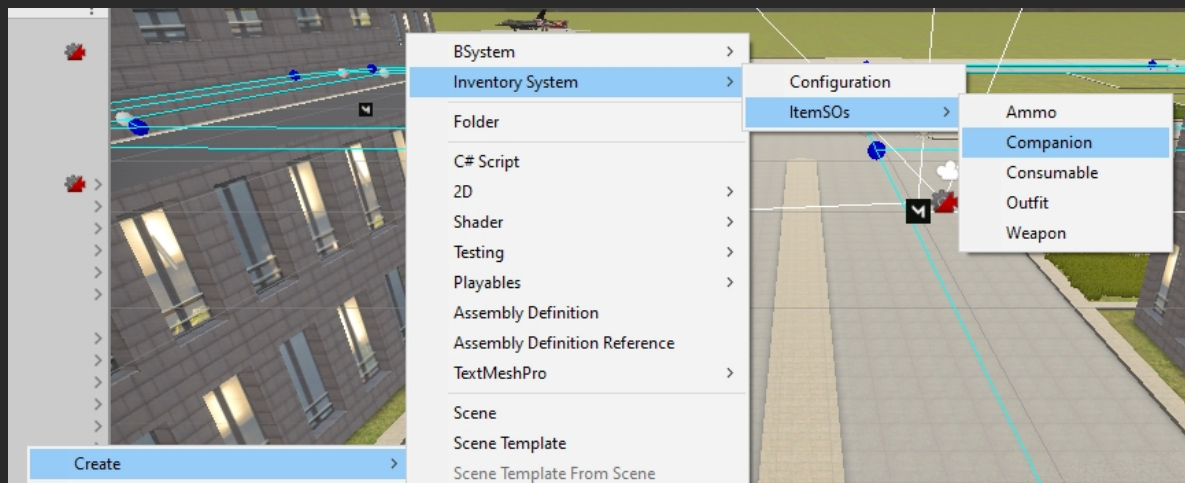
Companions

Companions can be placed in the **companion** slot in the **inventory**. After placing a **companion item** in the **companion** slot, it will appear and stay by the owner's side until the equipped **companion** item is removed from the **companion** slot.

Add an companion item

This guide explains how to **add an companion item**.

Create an **CompanionItemSO**. Creating an **CompanionItemSO** is almost the same as creating any other **itemSO**.



After you have created the **CompanionItemSO**, specified the unique name, the model prefab and the sprite, duplicate, rename and then open the "**Dog0Companion.prefab**". Then replace the **companion** model in the prefab with your new **companion** model and replace the animation controller component with the animation controller component from the new **companion** model. Then enter the idle, walk and run animation names of the new **companion** model. Select the "Network" game object located in the scene hierarchy of the "MainScene", navigate to the inspector window and add the created **companion** prefab to the "Registered Spawnable Prefabs" list of the CustomNetManager component. Finally, add a new entry to the "Companions" list of the "CompanionManager.cs" component at the player prefab, specifying the unique name, ItemSO and prefab of the companion.

Video guide: <https://www.youtube.com/watch?v=rVmp9OPgqw0>

Game World

To modify the **game world** or edit the objects, vehicles, and NPCs to be instantiated, open the "**Instance.prefab**" ("Assets/Advanced Multiplayer Template/Prefabs/Instance.prefab"). At the root of the **Instance.prefab** is an **Instance.cs** component that is responsible for instantiating world objects, vehicles, and NPCs.



Build System

The **build system** allows the players to purchase, place and sell placeable objects.

Add a new build object

This video guide explains how to add a new **build object** so that it can be selected, purchased, placed, stored and retrieved.



Place, rotate, edit and remove a build object

Open/Close Build Menu: Press the "X" button on your keyboard to open or close the build menu.

Place: Press the left mouse button to place a build object after you have selected it.

Rotate: Press the "R" key on your keyboard while selecting or editing a build object.

Edit: Focus on a build object and press the „E“ key on your keyboard to edit it.

Remove: Press the "ESC" key on your keyboard while editing a build object to remove it.

Vehicles

Cars and airplanes.

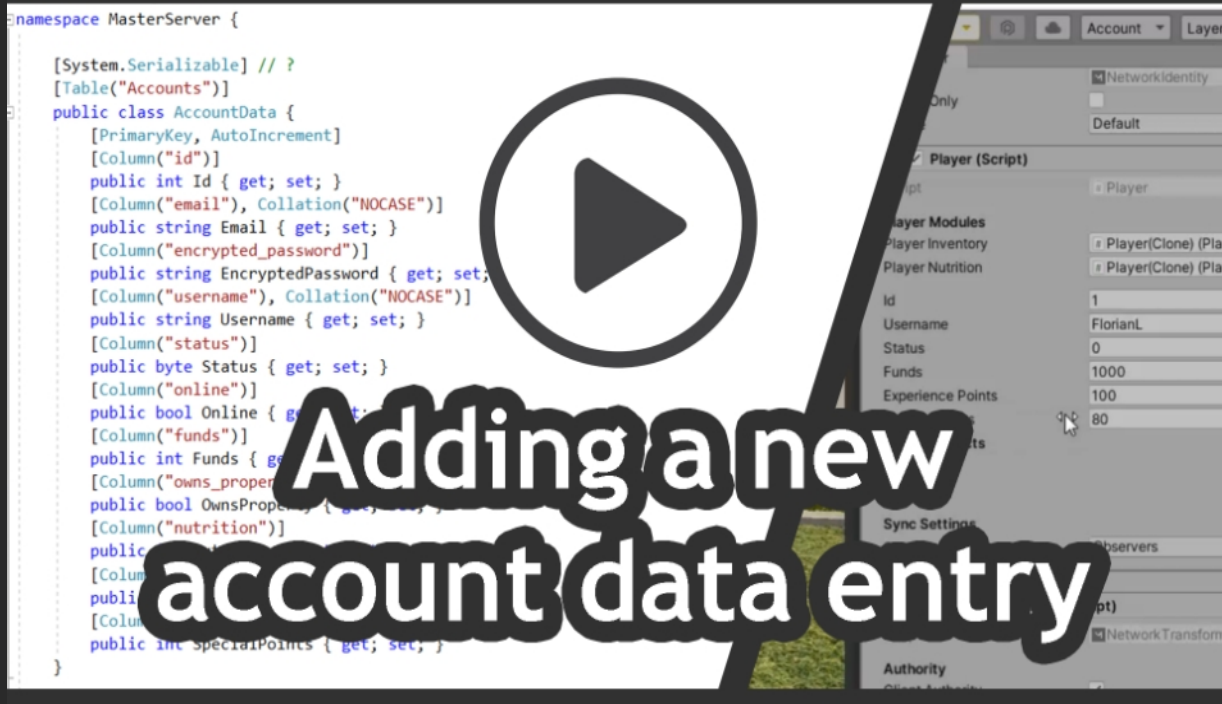
Add a new car

This video guide explains how to **add a new car** so that players can interact with it.



Adding a new account data entry

This video guide explains how to add a new account data entry.



The video thumbnail features a large play button icon in the center. On the left, a code editor displays the following C# code:

```
namespace MasterServer {  
  
[System.Serializable] // ?  
[Table("Accounts")]  
public class AccountData {  
    [PrimaryKey, AutoIncrement]  
    [Column("id")]  
    public int Id { get; set; }  
    [Column("email"), Collation("NOCASE")]  
    public string Email { get; set; }  
    [Column("encrypted_password")]  
    public string EncryptedPassword { get; set; }  
    [Column("username"), Collation("NOCASE")]  
    public string Username { get; set; }  
    [Column("status")]  
    public byte Status { get; set; }  
    [Column("online")]  
    public bool Online { get; set; }  
    [Column("funds")]  
    public int Funds { get; set; }  
    [Column("owns_property"), Collation("NOCASE")]  
    public bool OwnsProperty { get; set; }  
    [Column("nutrition")]  
    public int Nutrition { get; set; }  
    [Column("special_points")]  
    public int SpecialPoints { get; set; }  
}
```

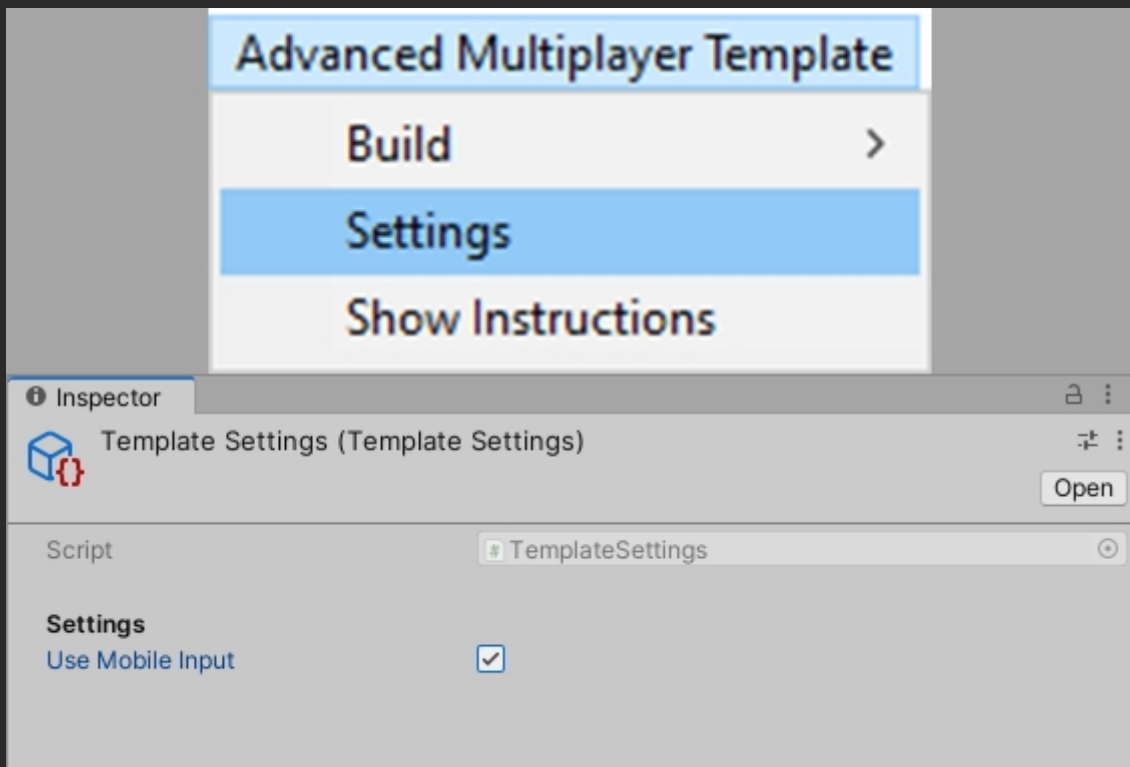
On the right, a game interface is shown with a dropdown menu set to 'Account' and a 'Layer' dropdown. Below these, a 'Player (Script)' section displays a table of player data:

Player	Id	Username	Status	Funds	Experience Points
Player (Clone) (Pla	1	FlorianL	0	1000	100
Player (Clone) (Pla					80

The title 'Adding a new account data entry' is overlaid in large, bold, white text with a black outline.

Mobile Compatibility

To import the necessary components to make the template playable on **mobile devices**, you must import the contents of the "**Mobile compatibility.unitypackage**" ("Assets/Advanced Multiplayer Template/Mobile/Mobile compatibility.unitypackage") into your project. After you have imported the content into your project, open the "**TemplateSettings.asset**" via the "**Advanced Multiplayer Template/Settings**" entry and select whether you want to use the **mobile input** or not. If you want to use **mobile input**, check the "**Use Mobile Input**" checkbox.



With the following **link** you can watch a **youtube video** showing how the **controllers** and **functions** can be **controlled** and **executed** via **Ui elements** after importing the components necessary for **mobile use**: <https://www.youtube.com/watch?v=GDJFv0Mn9d8>

Run Master and Game Server on a VPS

This video guide demonstrates how to **build**, **upload**, and **launch** the **master** and **game server** on a **Linux VPS**.



If your project is using the **HDRP (High-Definition Render Pipeline)**, make sure to **configure** the **settings** shown in the **video**, which you can access via the following **URL**. Otherwise, problems may arise when building the game server for Linux:

<https://www.youtube.com/watch?v=WVC1fOdRclo>

Master Server and Game Server

Powerfull systems that make it possible to have a stable **master server** and **game servers**.

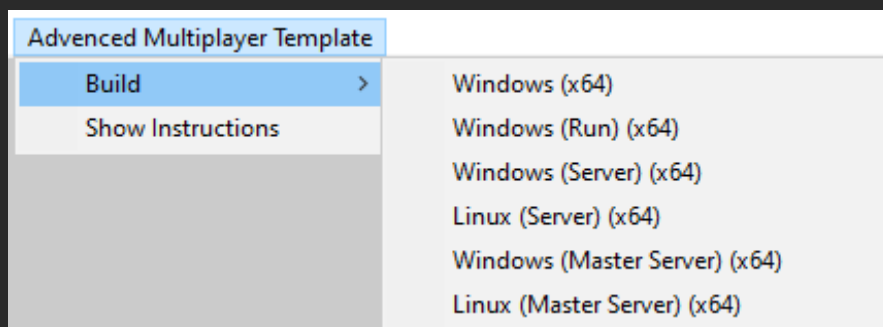
The **master server** and **game server** can be **built directly in unity** and not depends on something third party like mysql server. We tried to automate every process. You don't need to enter a bunch of information when creating a new **game server**, just run it and the **master server** itself will send the right information to the **clients**. The more servers you run, the more players will be able to create **subservers** in the menu.

1 server - 16 subservers of **16 players** (**256** in total, can be **increased** to **512**)

Questions and answers regarding the master and game server:

Q: How can I **build** the **master** and **game server**?

A: **Build** the **Master** and **Game Server** for **Windows** and **Linux** simply via the "**Build**" entry (Shown in every **video guide**.):



Q: Where are player **account data**, **inventory items**, **placed items**, current **money amount** and so on **saved/stored**?

A: All **data** is **saved/stored** in the "**db.sqlite**" file located in the **data folder** of the **master server**.

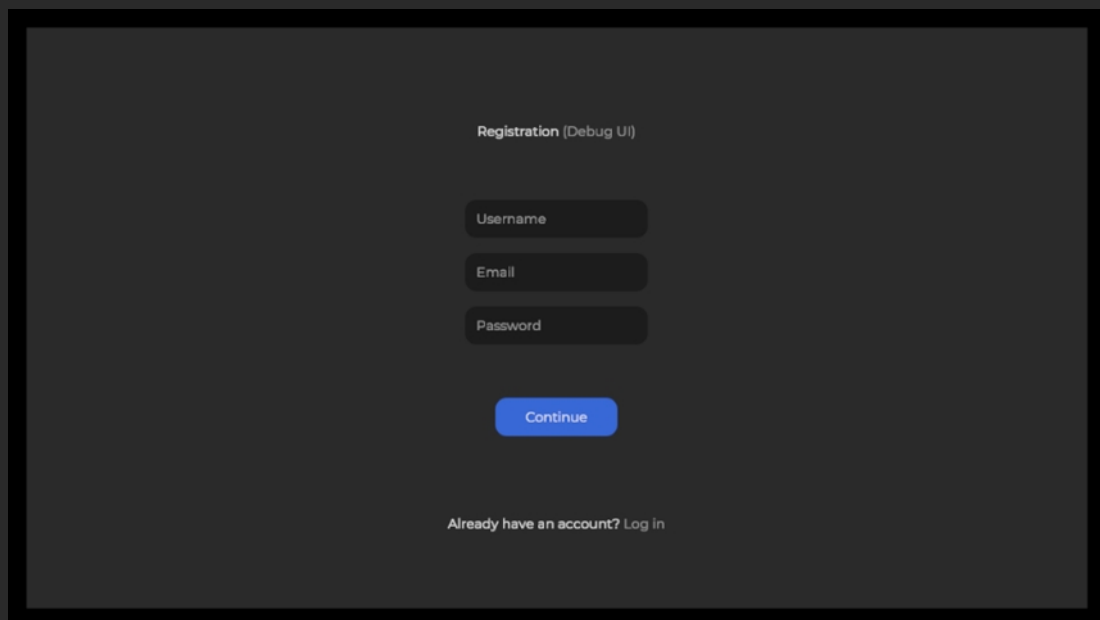
User Authentication

The user can **create** an **account** by **registering** and **logging** in with it.

Registration

If the user want to **register**, the user will be taken to this window shown below, where the user must enter the following information:

- **Username** (The username is displayed in the game above the player character and in the chat.)
- **E-Mail**
- **Password**



Registration (Debug UI)

Username

Email

Password

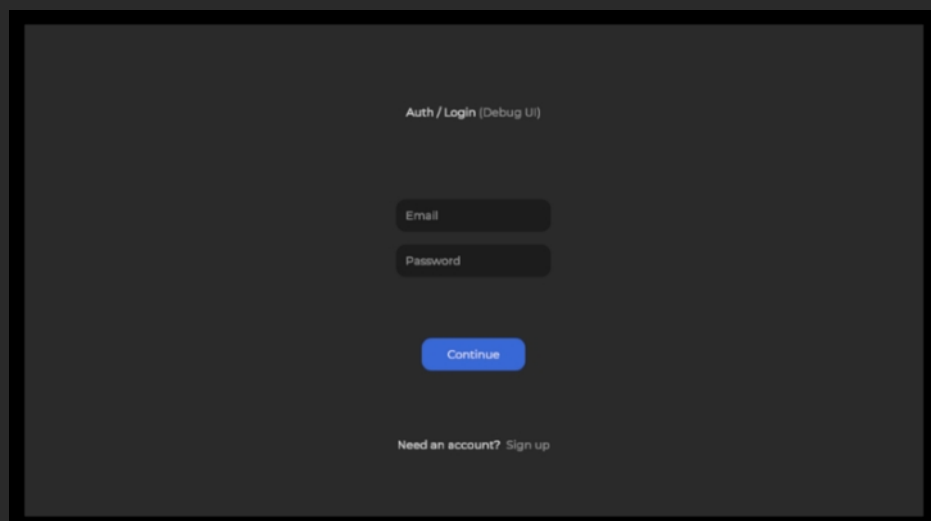
Continue

Already have an account? Log in

The image shows a registration form with a dark background. At the top, it says "Registration (Debug UI)". Below this are three input fields labeled "Username", "Email", and "Password". Under the input fields is a blue button labeled "Continue". At the bottom, there is a link that says "Already have an account? Log in".

Login

After the user has **registered**, the next time the user opens the game, the user will get to the window shown below and will have to enter his **e-mail** and **password**:



Auth / Login (Debug UI)

Email

Password

Continue

Need an account? Sign up

The image shows a login form with a dark background. At the top, it says "Auth / Login (Debug UI)". Below this are two input fields labeled "Email" and "Password". Under the input fields is a blue button labeled "Continue". At the bottom, there is a link that says "Need an account? Sign up".

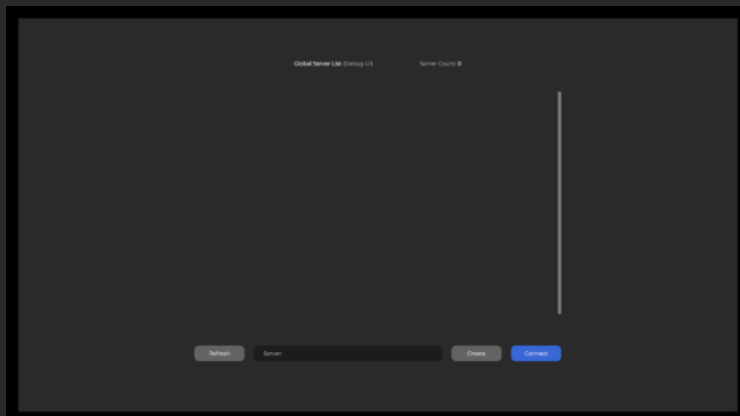
Server Creation and List

The user can **create** a **server** and display it in the **server list** so that other players can join the **server**.

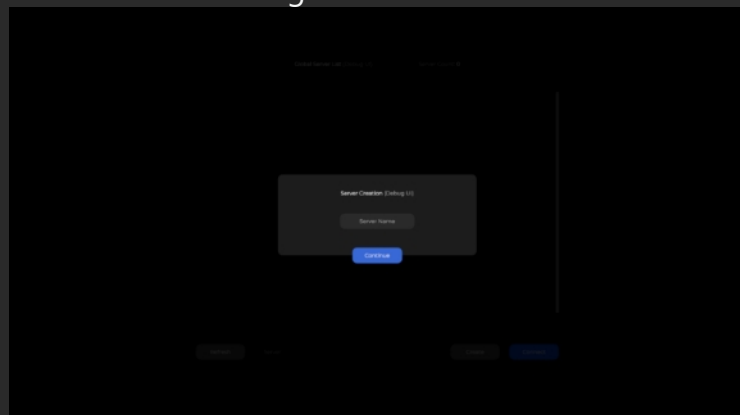
Server Creation

To **create** a **server**, the user must click on the

Create

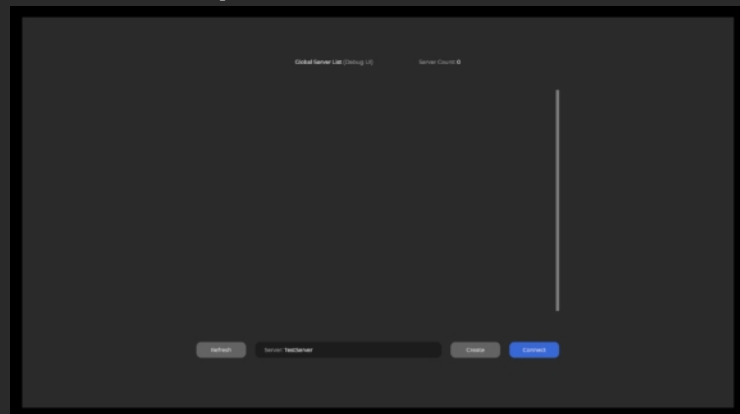


Now the user must give the **server** a **name**:



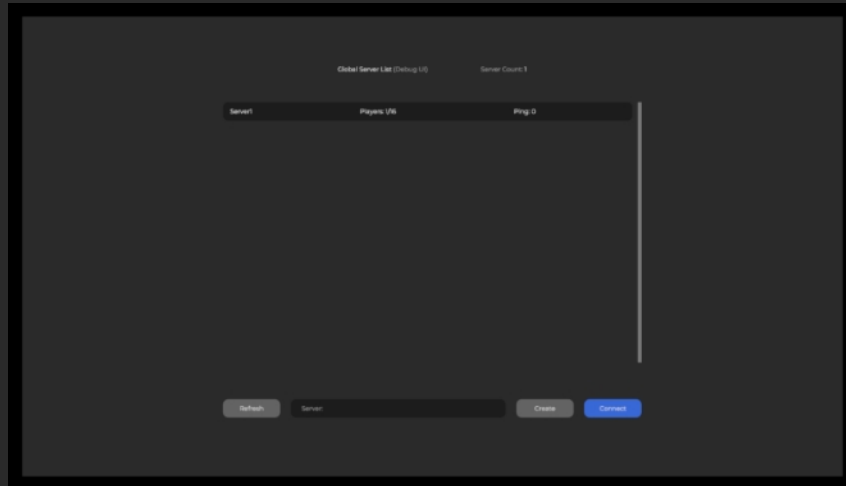
Finally, the user must click on the

Connect

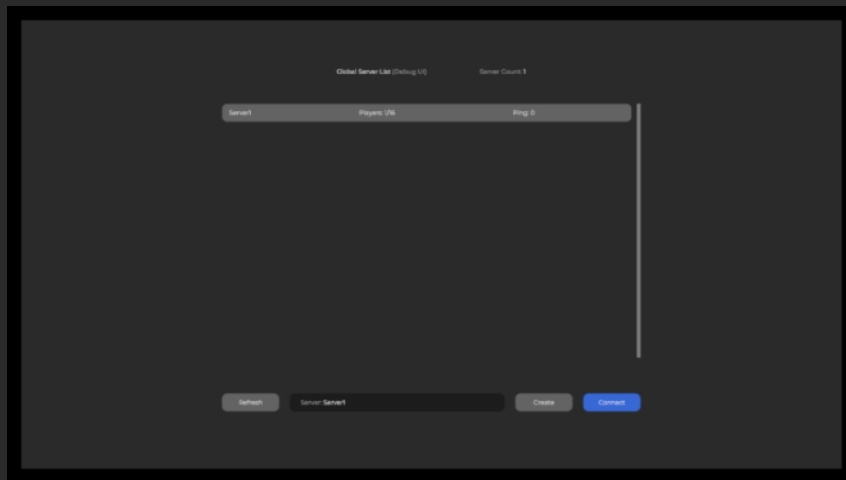


Server List

When a user has **created** a **server**, it appears in the **server list**:



A **server** that is displayed in the **server list** can be joined by any **logged-in** user if the **server** is not full:

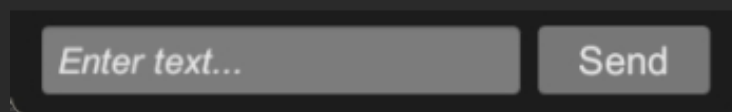
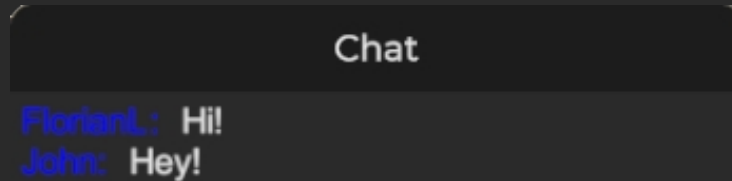


Chat

The **chat** allows players to **communicate** with each other in game.

Use of the chat

To open the **chat**, the player must click the "**T**" key on his keyboard. The user can automatically enter a text in the input filed and with the button "**Send**" the player can **send** the text and all players on the **server** will see this text.



To open the **chat**, the player must click the "**T**" button on his keyboard. The

Hotkeys

- Inventory

To open the inventory, press the "**Tab**" key on your keyboard.

To split stackable items, hold the "**Shift**" key on your keyboard

- Build System

To open the build system, press the "**X**" key on your keyboard.

Press the left mouse button to place a build item after selecting it.

Press the "**R**" key on your keyboard while selecting or editing a build object to rotate it.

Focus on a build object and press the "**E**" key on your keyboard to edit it.

While editing a build object, press the "**ESC**" key on your keyboard to remove it.

- Emote Wheel

To open the emote wheel, press the "**M**" key on your keyboard.

- Vehicles

To get into a vehicle, press the "**E**" key on your keyboard.