



# ***VR Oculus MFPS PRO***



Tested in Oculus Quest 1-2



***V30.0***

IMPORTANT INFORMATION	
Author	michael.soler.beatty@gmail.com
Unity Version	2020.2.2.f1
Dependencies	URPL, PHOTON PUN 2, PHOTON VOICE 2
Tested targets	Oculus Quest 1 AND 2
<b>IMPORTANT</b>	<b>THIS A COMPLETE PROJECT. YOU MUST MAKE A BACKUP OF YOUR PROJECT BEFORE IMPROTING. USE "NEW"</b>
Discord	<a href="https://discord.gg/WqSxEU3">https://discord.gg/WqSxEU3</a>

## INDEX

1. DESCRIPTION OF THE ASSET .....	4
1.1. HOW DOES IT WORK? .....	6
1.2. WHAT CAN YOU ACHIEVE WITH THIS? .....	6
1.3. WHERE TO TEST IT FIRST? .....	7
2. IMPORTING DEPENDENCIES.....	7
3. INITIAL PACKAGE CHECK .....	7
3.1. tags and layers .....	7
3.2. physic matrix.....	8
3.3. TIME .....	9
3.4. FILE STRUCTURE .....	9
3.5. prepare urpl .....	9
3.6. prepare scenes.....	10
4. RESOURCES.....	12
4.1. IMPORTANT OBJECTS in photonprefabs folder.....	12
4.2. WEapon type rifle .....	15
4.3. weapon type shotgun .....	16
4.4. pistol .....	16
4.5. MAGAZINE .....	17
4.6. WEAPON MAGAZINE.....	17
4.7. tank.....	17
4.8. plane .....	18
4.9. CONFIGURE GRABBING OBJECTS .....	18
4.10. PLAYER AVATAR NEW CONFIGURATION .....	19
4.11. shooting and bullets.....	20
1. SCRIPTING .....	21
2. sCENES .....	22
2.1. 00-PersistentScene .....	22
2.2. 00-MenuScene .....	22
2.3. MULTIPLAYER SCENES.....	25
3. video-tutorials .....	26
3.1. PROMOTIONAL VIDEO.....	26
3.2. HOW TO CHANGE/ADD WEAPONS .....	26
3.3. HOW TO CHANGE/ADD avatarS .....	26
3.4. HOW TO CHANGE/ADD MAPS .....	26
3.5. HOW TO CHANGE VEHICLES .....	26

4. questions? .....	26
---------------------	----

## 1. DESCRIPTION OF THE ASSET

This is PRO version of the complete package for creating a multiplayer shooter game (MFPS) in virtual reality designed for the new XR system of Unity3D. This package is the result of a deep knowledge of VR and multiplayer interface.

We strongly recommend a maximum of "5x5" players for performance reasons. Players play in five different modes in two different configurable maps. The modes are teams, royale, drone, bomb and flag. The package is a complete and functional project ready to be deployed into the Oculus Quest 1 or Quest 2.

The game dynamics work as follow:

First, a popup menu is shown and the player can select his username (it is also auto-generated). In this same menu, the payer can choose between the active rooms which are displayed with its information in the canvas using advanced room properties. The menu allows the user to select the layout (weapons, abilities and scopes). Once the player logs into a room, a specific subscene is created in function of the selected parameters.

Movement:

Movement of the weapon and shooting is achieved with the Oculus quest controllers. Aiming is affected by the movement and shooting of the player of the player. The movement is divided into two consistent methods: room-scale of the XRrig and the movement of a character controller.

Weapons:

Weapons used int this package supports two-hand and multiple hand grabbing. This means that weapon interaction is totally realistic. Weapons need to be loaded using magazines and feeding it using the bolt.

This package includes v30:

1. Scripts that control the Photon server calls.
  - a. Player creation.
  - b. Lobby and Room management.
  - c. Player connection/disconnection.
  - d. Setting custom Photon properties for scoring.
2. Integration with oculus Quest for VR game.
3. Player movement, shooting and health.
4. Script that control the camera movement.
5. Scripts to change/shooting weapons.

6. Prefabs that can be customizable to change the avatars of characters.
7. Unique avatar sent across the photon server.
8. Scripts that control the "TAKE OVER" for common gameobjects in the scene.
9. Raycasting from player oculus right joystick using a laser effect.
10. All the textures, prefabs, gameobjects, materials shown in the video.
11. Almost one comment per 5 lines of code.
12. Customizable window for settings.
13. Scripts to debug OVR controller on Editor mode.
14. INVERSE kinematics to move the hands and head of the players.
15. Haptic feedback (vibration of the controllers) with custom noise feedback when shooting and being hit. Only compatible with version 15.0.
16. In-game voice.
- 17) Compatible with XR Unity3D system
- 18) New hand model (independent from Oculus)
- 19) Totally independent package with its own meshes and models.
- 20) New environment and dynamic doors
- 21) Post-processing and compatible with URP (universal renderer pipeline)
- 22) Avatar animation blending for walking in crouch mode and normal mode which is changed according to the height of the player.
- 23) Two vehicles with 3d movement using the controller sticks: a tank and a plane.
- 24) Turrets with heat control with two hand control. It needs to press both triggers to shoot.
- 25) Climbing ladders using hands.
- 26) Improved reloading routine as well as consistent grabbing effect for objects.
- 27) three weapons with different scopes and sights. Scopes use URPL compatible graph shaders.
- 28) New menu canvas with weapon selection, avatar selection and ability selection
- 29) four abilities to choose from. shield, health restoring, antitank and drone.
- 30) Consistent grabbing of other player's weapons.
- 40) Manage four game modes: teams, royale, drone, flag and bomb.
- 41) general scoring
- 42) Shop manager with in-game article shopping.

And there is much more and more to come!

This package needs the following FREE packages from Unity.

URPL

Photon PUN2

Photon Voice 2

SUPPORT

Discord

<https://discord.gg/WqSxEU3>

[michael.soler.beatty@gmail.com](mailto:michael.soler.beatty@gmail.com)

From cardboard buddies we want to give the best customer service, so we are available 24/7 for all doubts, errors in code development and potential modifications.

### 1.1. HOW DOES IT WORK?

The package can be divided in these steps:

#### **1- Connecting to the server.**

Photon pun 2 is used to connect the players and create the multiplayer interface.

#### **2- Configuring**

The player can change his/her name and weapon layout and abilities. When creating a room, they can also change the room game mode.

#### **3- Playing**

In function of the game mode or map selection a different scene is loaded.

### 1.2. WHAT CAN YOU ACHIEVE WITH THIS?

A complete MFPS game for Oculus Quest with vehicles, advanced hand interaction and vehicles.

### 1.3. WHERE TO TEST IT FIRST?

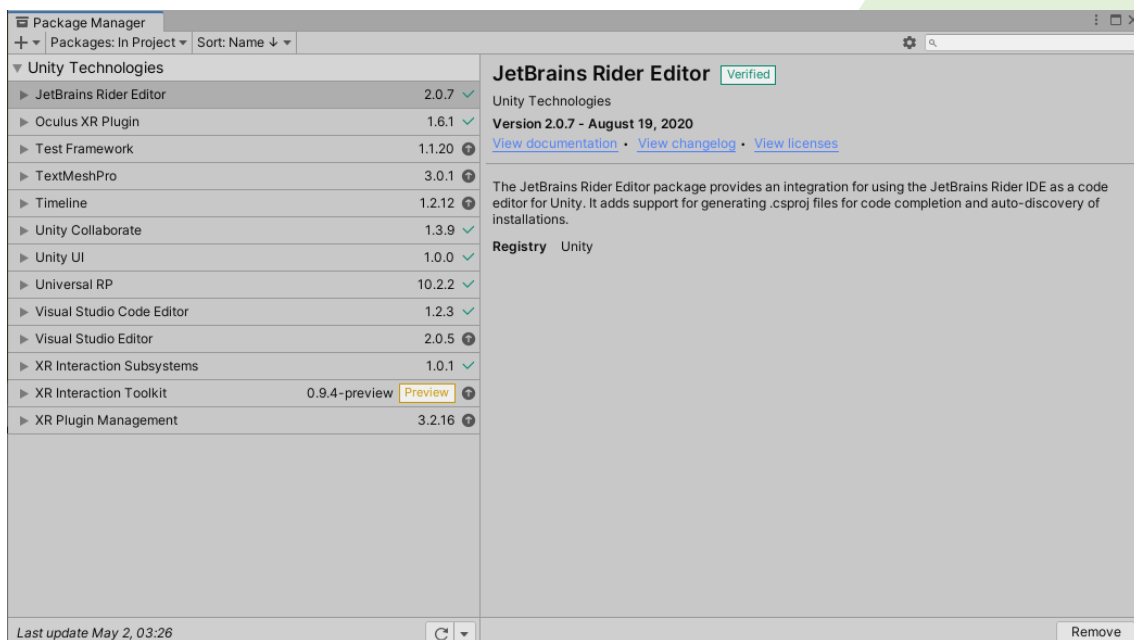
You can test the app from these links in google Drive and SideQuest.

<https://drive.google.com/file/d/1LkITch3eY8J6OaF2tniYrameOIUI7G35/view?usp=sharing>

<https://sidequestvr.com/app/2078/neonwars>

## 2. IMPORTING DEPENDENCIES

Use the package manager will ask you to import the following dependencies.



### 3. INITIAL PACKAGE CHECK

This is the process you need to follow in order to check if the **complete package** was imported properly. **IMPORT ALL DEPENDENCIES IN A NEW PROJECT.**

#### 3.1. TAGS AND LAYERS

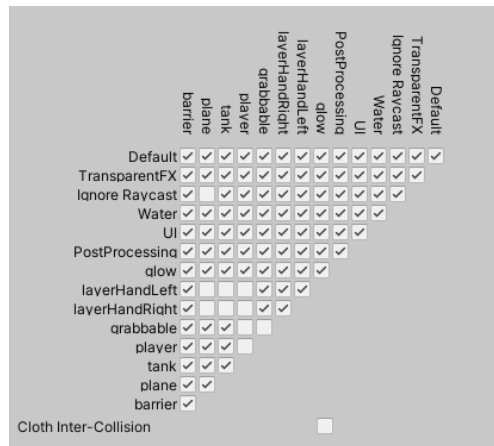
These are the tags and layers used in the project:

▼ Tags	
Tag 0	Bullet
Tag 1	magazine
Tag 2	rifleInsertMag
Tag 3	pistolInsertClip
Tag 4	scoreManager
Tag 5	handLeft
Tag 6	handRight
Tag 7	winner
Tag 8	time_hud
Tag 9	scoreBoardLine
Tag 10	debugText
Tag 11	bulletHUD
Tag 12	head
Tag 13	bullet
Tag 14	environmnet
Tag 15	roomInstance
Tag 16	hand
Tag 17	zombieHead
Tag 18	zombieBoddy
Tag 19	ovrCntroller
Tag 20	Avatar
Tag 21	grenadePos
Tag 22	menu
Tag 23	charBack
Tag 24	but
Tag 25	char
Tag 26	button
Tag 27	buttonKnob
Tag 28	bodyCollider
Tag 29	connectedIcon
Tag 30	disconnectedIcon
Tag 31	gameSetup
Tag 32	environment
Tag 33	spawnPoint
Tag 34	bulletContainer
Tag 35	ground
Tag 36	drone
Tag 37	XR
Tag 38	scopePositionRifle
Tag 39	handCollider
Tag 40	tank
Tag 41	plane
Tag 42	tankChasis
Tag 43	regen
Tag 44	maxLimit
Tag 45	minLimit
Tag 46	playerInstance
Tag 47	snapBomb
Tag 48	startTimerText
Tag 49	disableOnStart
Tag 50	bombManager
Tag 51	snapFlagAlpha
Tag 52	snapFlagBravo
Tag 53	ending
Tag 54	bomb

### 3.2. PHYSIC MATRIX

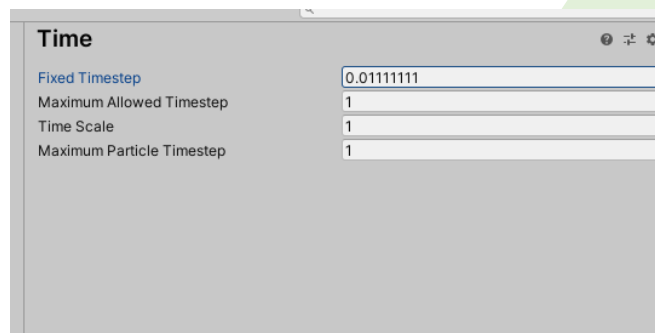
The Physic matrix is the one by default.





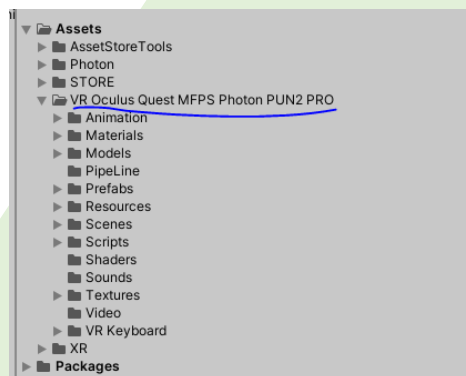
### 3.3. TIME

Time is reduced to work better for oculus quest.



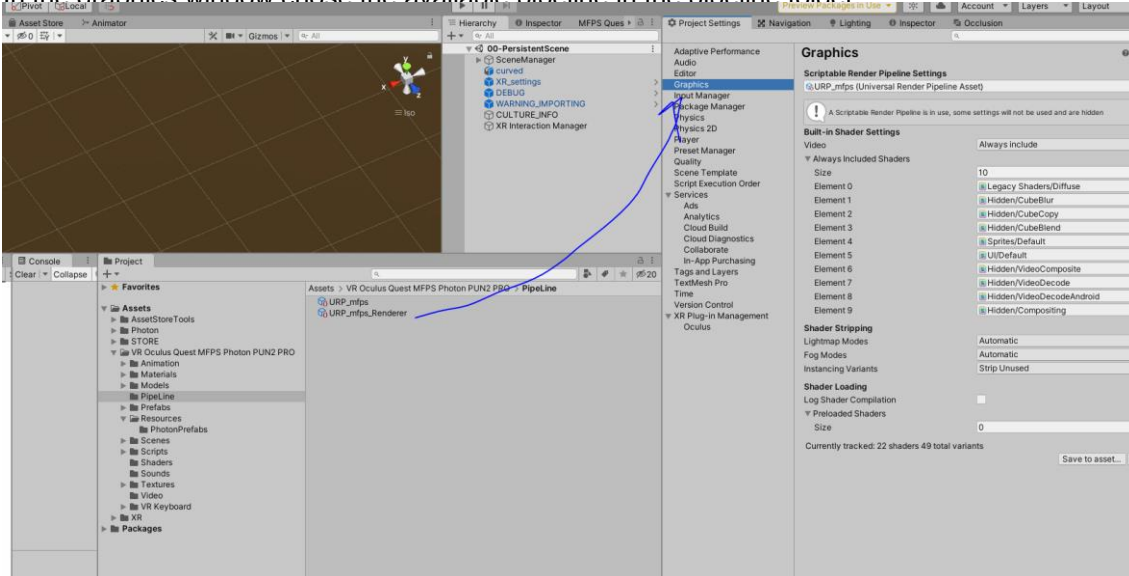
### 3.4. FILE STRUCTURE

The file structure you need to see in the inspector is:



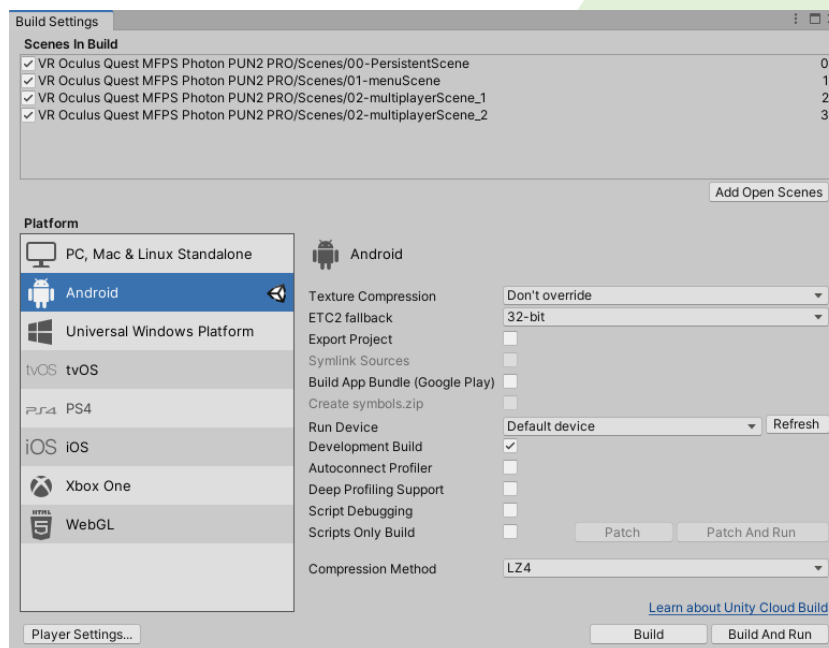
### 3.5. PREPARE URPL

In the graphics window chose the available pipeline in the pipeline folder:



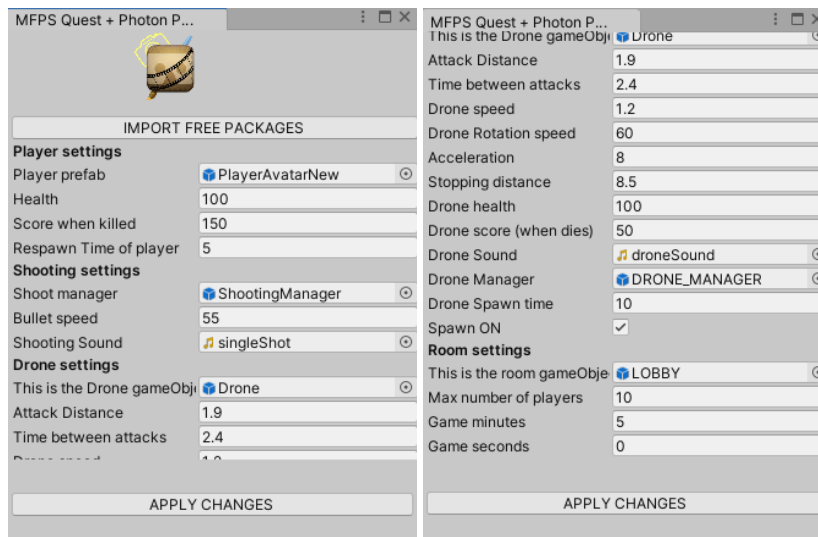
### 3.6. PREPARE SCENES

Add these scenes in order to the build settings:



### 3.7. ABOUT THE WINDOW "MFPS PRO"

This is the window that summarizes the most important variables of the asset. It always loads when playing and editing scripts. You can delete it if you do not want to use it.



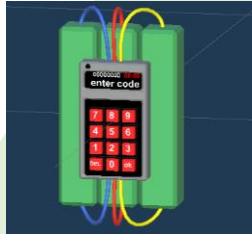
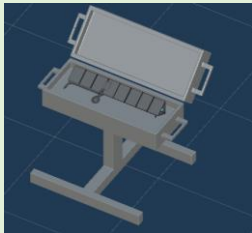

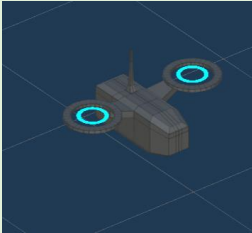

You can use the button to import the free packages of PHOTON used to work. If the values are different from those shown, you might have a problem while importing.


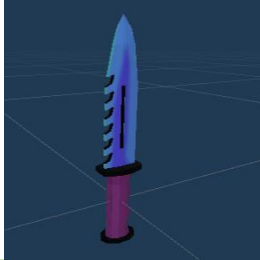
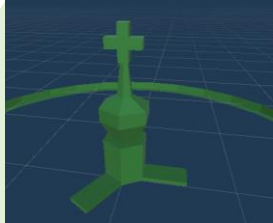

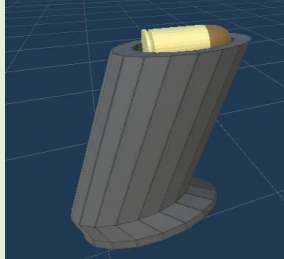
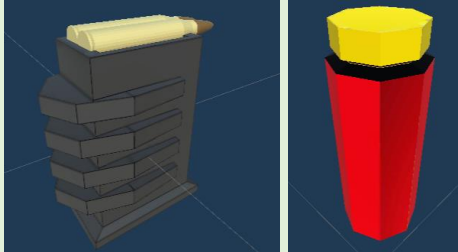
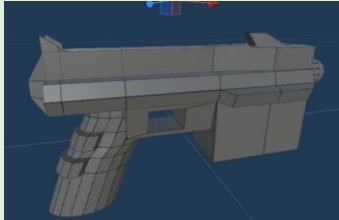
**If something is missing or you have errors at this point, please contact us at: [michael.soler.beatty@gmail.com](mailto:michael.soler.beatty@gmail.com). We usually give an answer in about 24-48 hours**

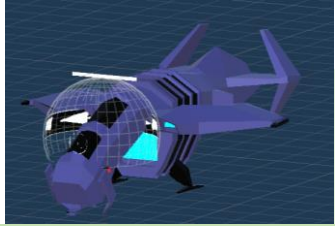
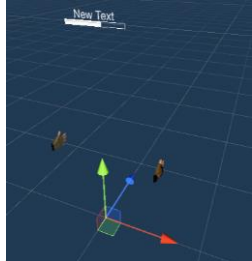
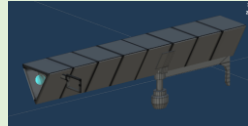
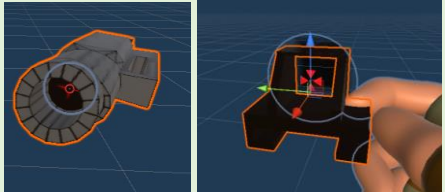
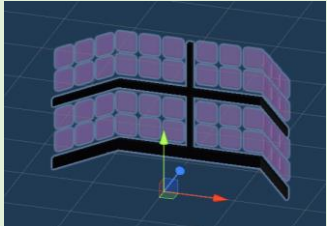
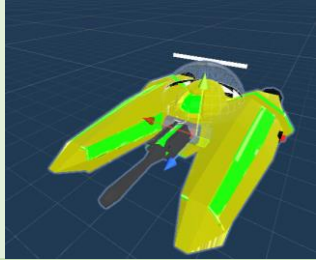


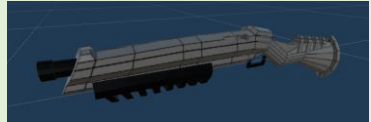
## 4. RESOURCES

### 4.1. IMPORTANT OBJECTS IN PHOTONPREFABS FOLDER

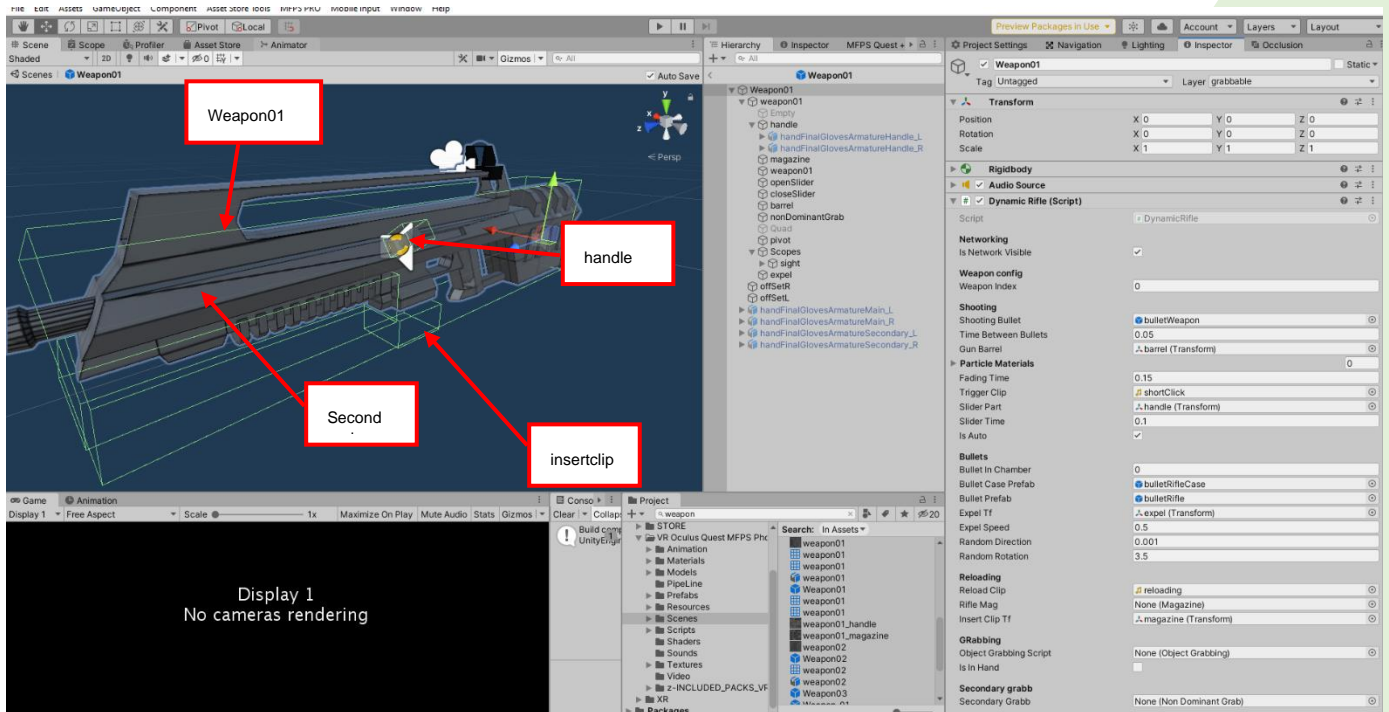
These elements have PhotonViews and multiple interaction scripts.

NAME	DESCRIPTION	IMAGE
<b>bomb</b>	<p>It uses scripts to introduce a code and activate it in the case that the code is exactly the same as the password.</p> <p>It is used for the game mode called “bomb” and it also introduces advanced finger interaction.</p>	
<b>boxWeaponRocket</b>	<p>It is a gameobject that contains a special anti-vehicle weapon that needs to lock to shoot.</p> <p>It is spawned when the ability “missile” is selected.</p>	
<b>Drone</b>	<p>It is the drone object used for generating the game mode “drone”. It has basic AI to persecute the players and shooting in function of the distance to the connected users.</p>	
<b>dronePlayer</b>	<p>It is spawned when the ability “drone” is selected. It also has some AI to go after the players that are not in the same team as the player that has created it.</p>	
<b>flag_Alpha</b> <b>flag_Beta</b>	<p>These two objects are used in the “flag” game mode. It is used to score when players take the flags of their opponent and take it to their base.</p>	

<b>grenade</b>	This is interactable gameobject that can be activated and explodes after a certain time affecting the health of the players in an specific range.	
<b>knife</b>	An interactable object that allows to create melee attacks.	
<b>lifeGenerator</b>	It is another ability of the player that can be created in-game to re-generate the life of the users inside.	
<b>life</b>	Restores a part of the life of the user when pressing the inner button.	
<b>Magazine</b>	Used to load the pistol. It contains the number of bullets available. It is also dynamic in the sense that bullet meshes appears or disappears when the magazine is full or empty.	
<b>MagazineWeapon01</b> <b>MagazineWeapon02</b> <b>MagazineWeapon03</b> ...	These are the magazines used for the rifles (main guns). They have the same characteristics as the previous ones.	
<b>Pistol</b>	This is the secondary weapon available for the player. It needs to be loaded and you also need to pull the slider to feed the chamber. It also supports recoil.	

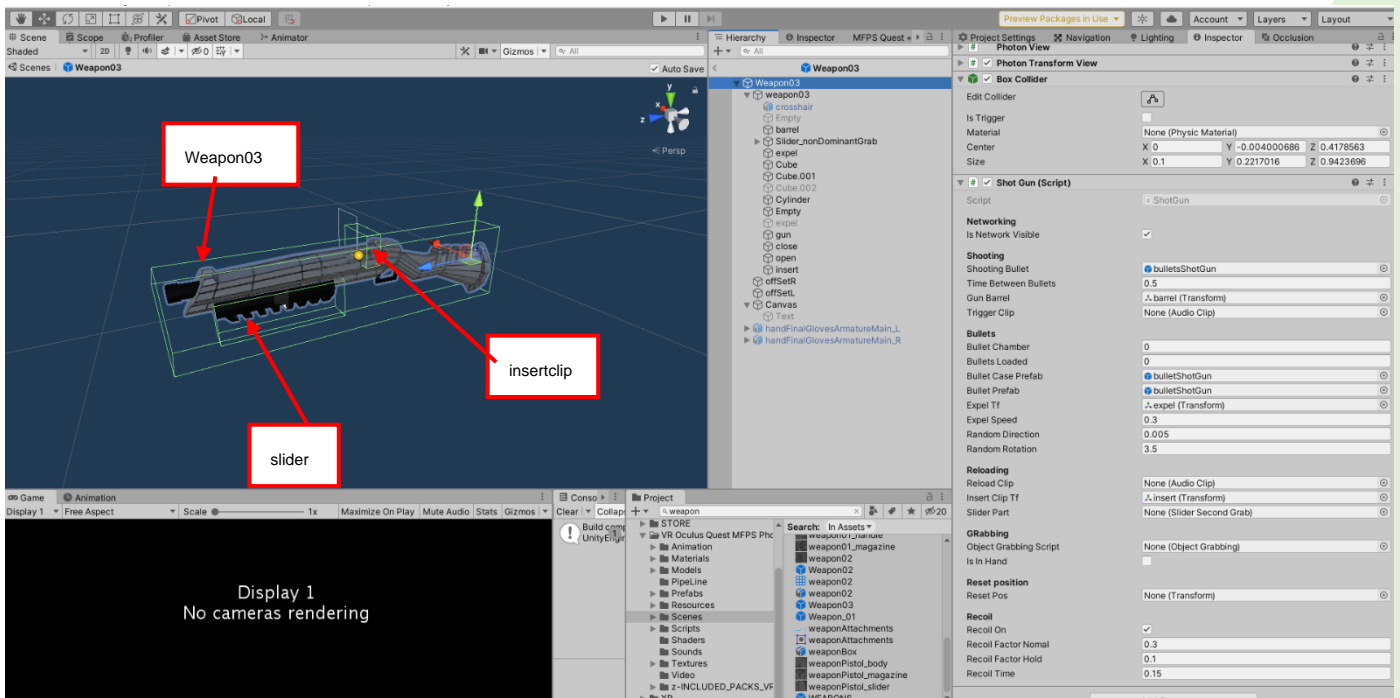
<b>Plane</b>	<p>This is the plane vehicle that allows the player to fly in the game scene. To enter the vehicle you need to press the red buttons.</p> <p>A specific Hud appears when the player is in the plane.</p>	
<b>PlayerAvatarNew</b>	<p>This is the main payer gameobject. It has the XR rig and the necessary scripts to manage weapons, photon animation, hands, avatar management, etc.</p>	
<b>rocketLauncher</b>	<p>This is an advanced weapon to destroy the vehicles. It needs to focus on the object and generates a missiles that follows the vehicle.</p>	
<b>Scope_W2_01</b> <b>Scope_W2_02</b> <b>sight_w1_01</b> ...	<p>These are the scopes and sights used in weapons.</p>	
<b>shield</b>	<p>A player ability to create a shield that protects the user from bullets</p>	
<b>tank</b>	<p>This is the tank vehicle that allows the player to move in the game scene. To enter the vehicle you need to press the red buttons.</p> <p>A specific Hud appears when the player is in the tank.</p>	
<b>Weapon01</b> <b>Weapon02</b> <b>Weapon03</b> ...	<p>These are the weapons used in the game. We have two different types, the rifle and the shotgun.</p> <p>Both allow two-hand grabbing, loading, and shooting with specific parameters.</p>	  

## 4.2. WEAPON TYPE RIFLE



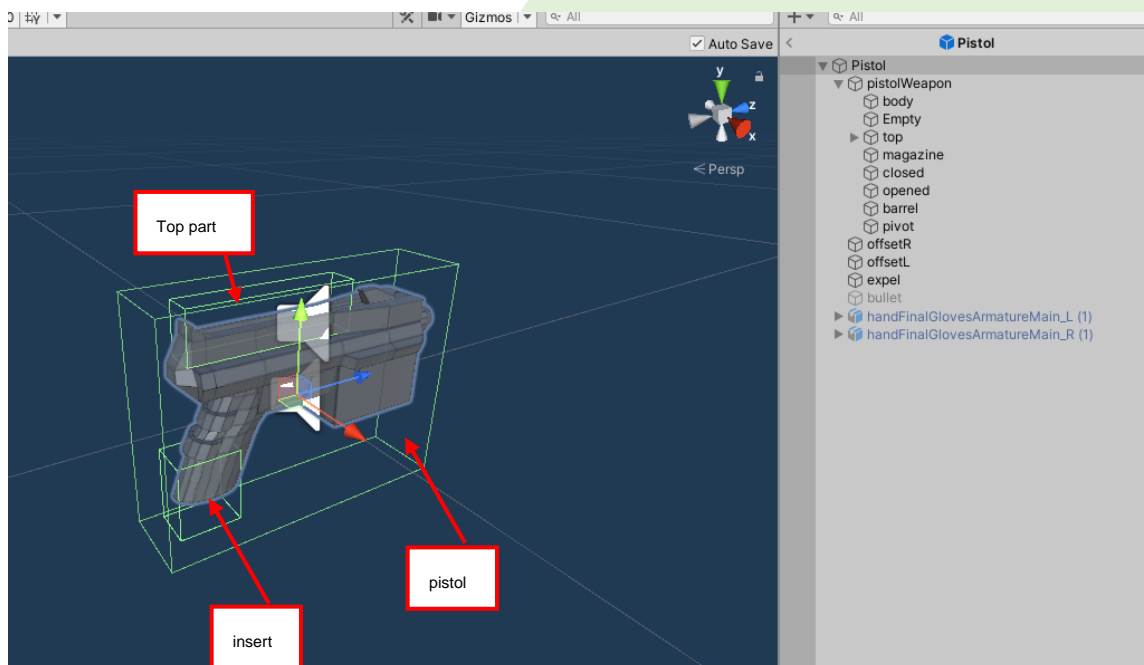
To work properly the weapon is composed by three elements. The main weapon mesh with the grabbing scripts and the rigid body. Then the secondary zone is created to allow two-hand grab. Finally the handle is the part of the weapon used for loading and feeding the chamber by moving the bolt. It is important to notice the insertClip collider, where magazines are introduced to increase the bullet number.

### 4.3. WEAPON TYPE SHOTGUN



The main change is that there is no handle, but a slider part that allows the player to load the bullets into the chamber. This last part is controlled by the “slidersecondarygrab”.

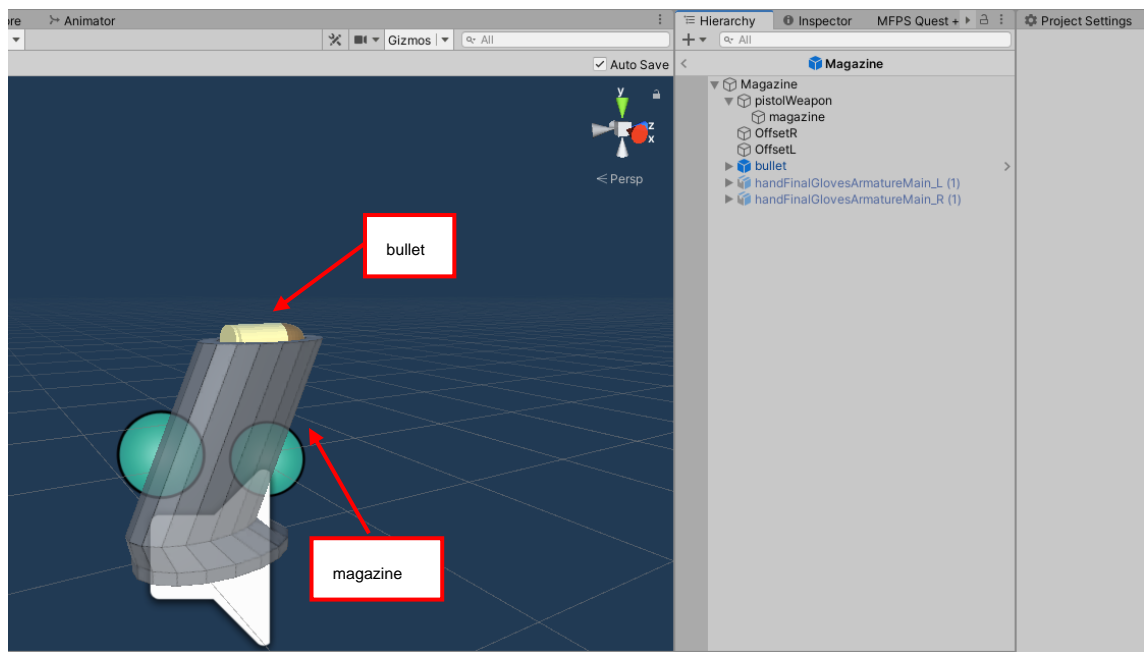
### 4.4. PISTOL



The pistol is composed of two main parts: the pistol body and the top part. It also has a collider for the magazine insertion.

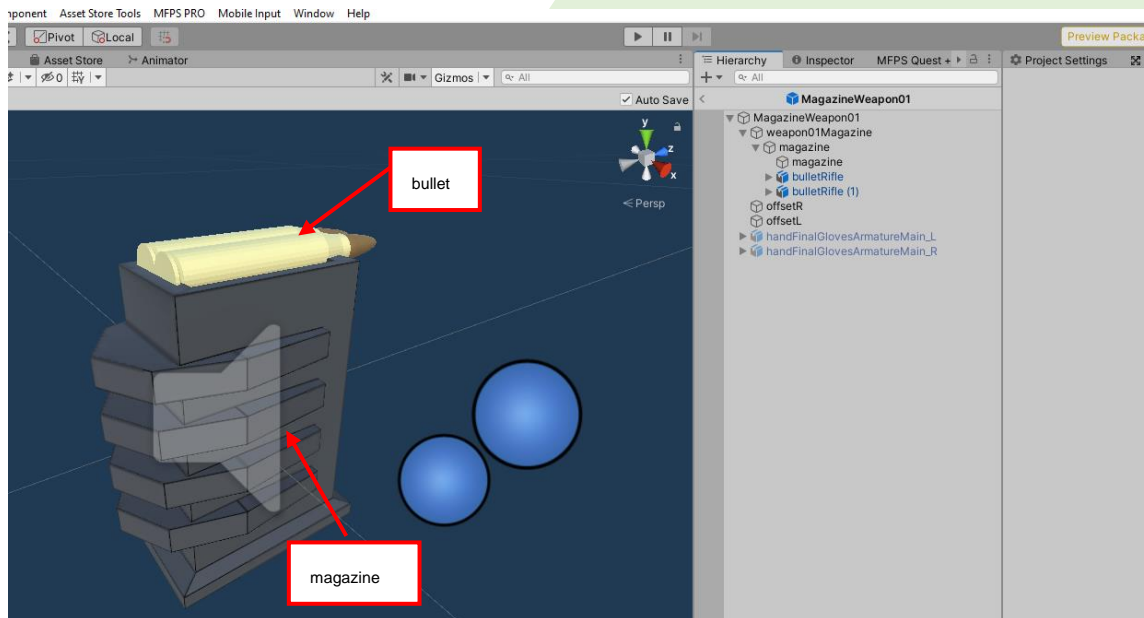


#### 4.5. MAGAZINE



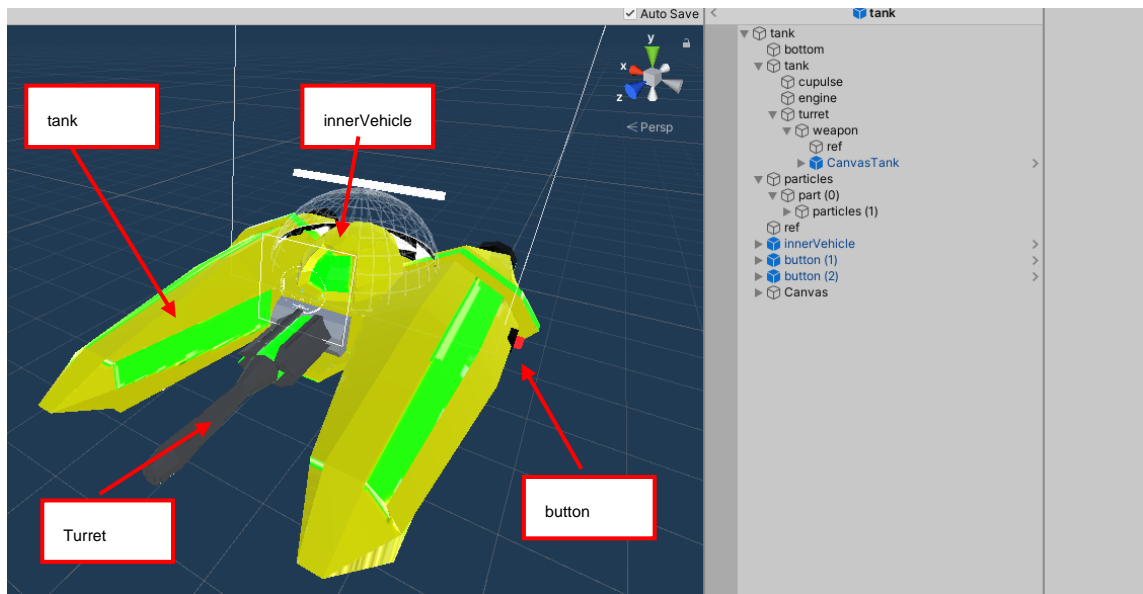
This is the magazine for the pistol. It has scripts to save the number of bullets and to be grabbed by the user.

#### 4.6. WEAPON MAGAZINE



Very similar to the pistol magazine.

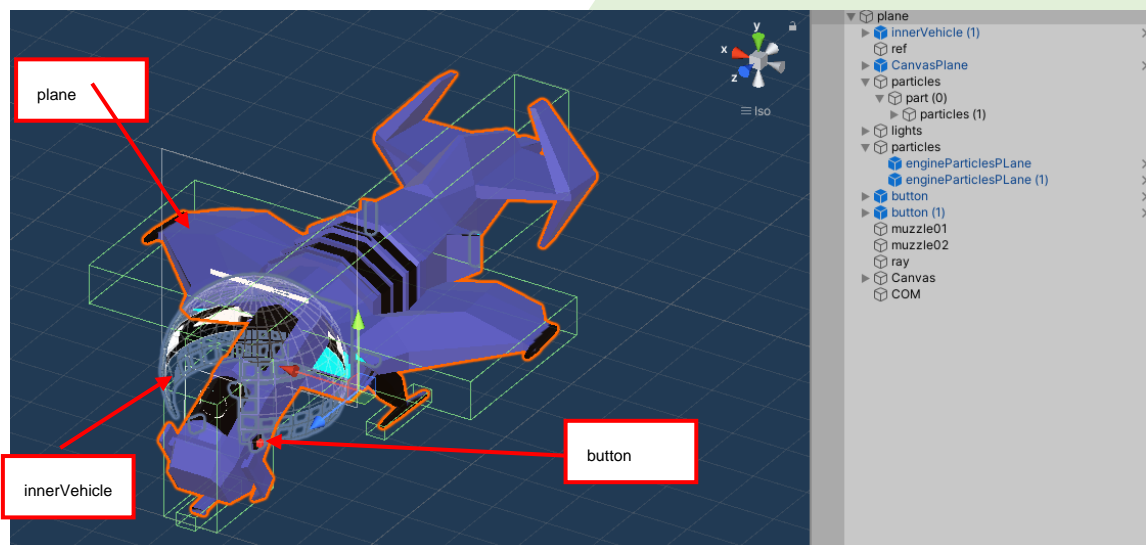
#### 4.7. TANK



Buttons are used to enter and quit the tank. The “innerVehicle” gets activated or deactivated in function if the player is inside or not.

Movement is achieved with the left stick and turret’s rotation changes with the right stick.

#### 4.8. PLANE

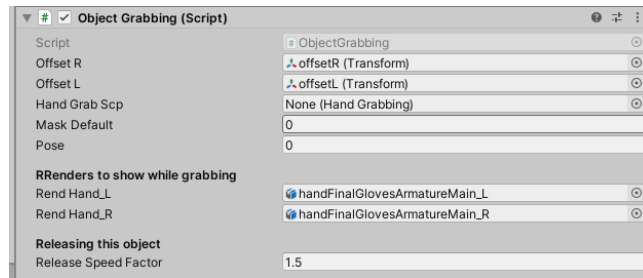


Vertical position and Y rotation is controlled by the right stick, while the inclination forward and to the sides is controlled with the left pad.

#### 4.9. CONFIGURE GRABBING OBJECTS

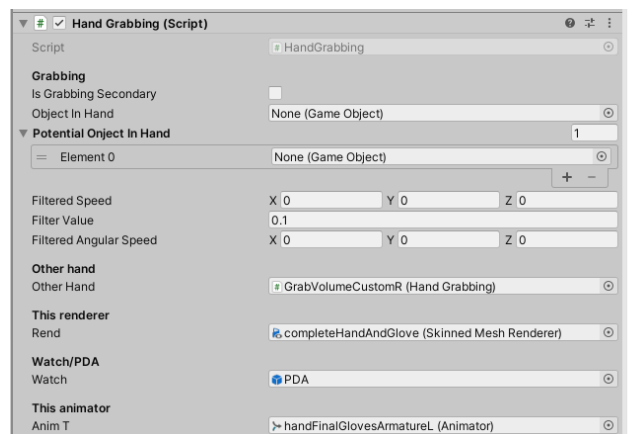
Grabbing is a combination of two scripts, one placed on the object an another on the hands.

**Script in the object to grab**



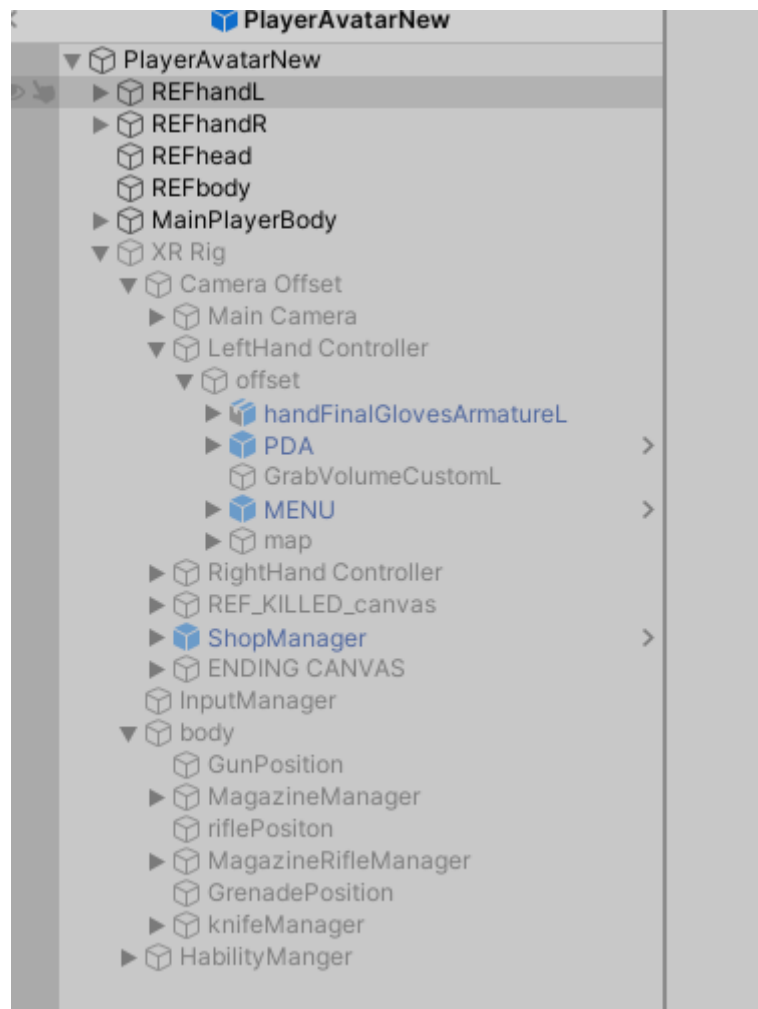
The offsets are the values obtained experimentally to adapt the position of the object when being grabbed. The renders are the hands used when grabbing. Notice that the real hands are disabled at this point, and the hand renders of the object are shown.

## Scripts in the hands



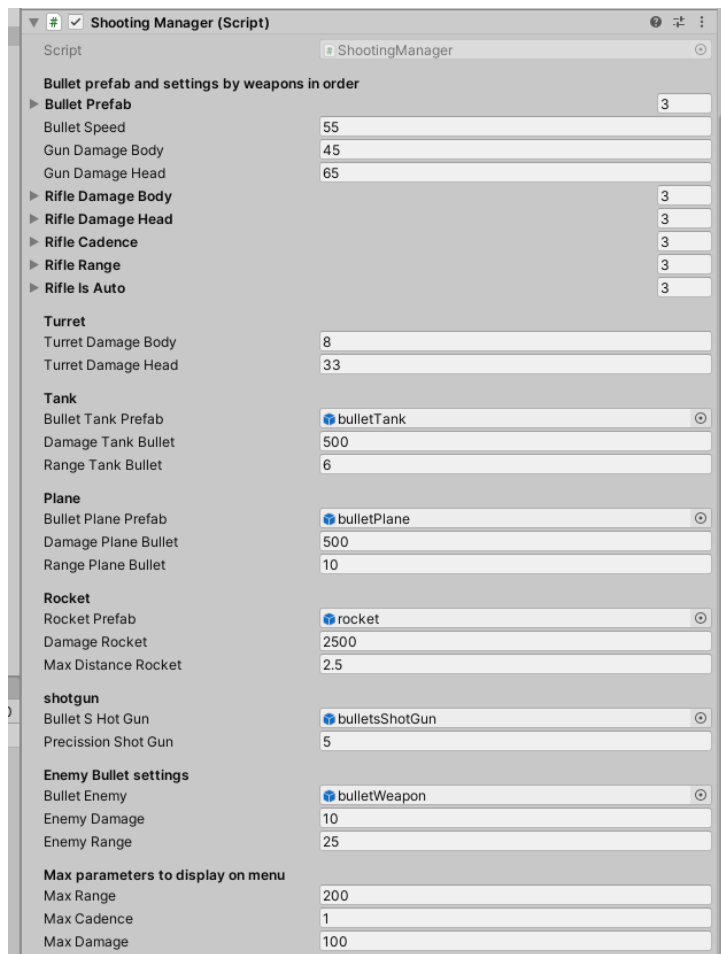
Detects and stores all the objects where the hand is close to. Then it also manages the speed for realizing the values and the animator of the hand to change the grabbing and pointing states.

## 4.10. PLAYER AVATAR NEW CONFIGURATION



This is the original configuration of the avatar. The XR rig allows to manage the camera in VR. The other elements called managers, are used to control all the needed functions to spawn and place weapons, control the shop states, etc.

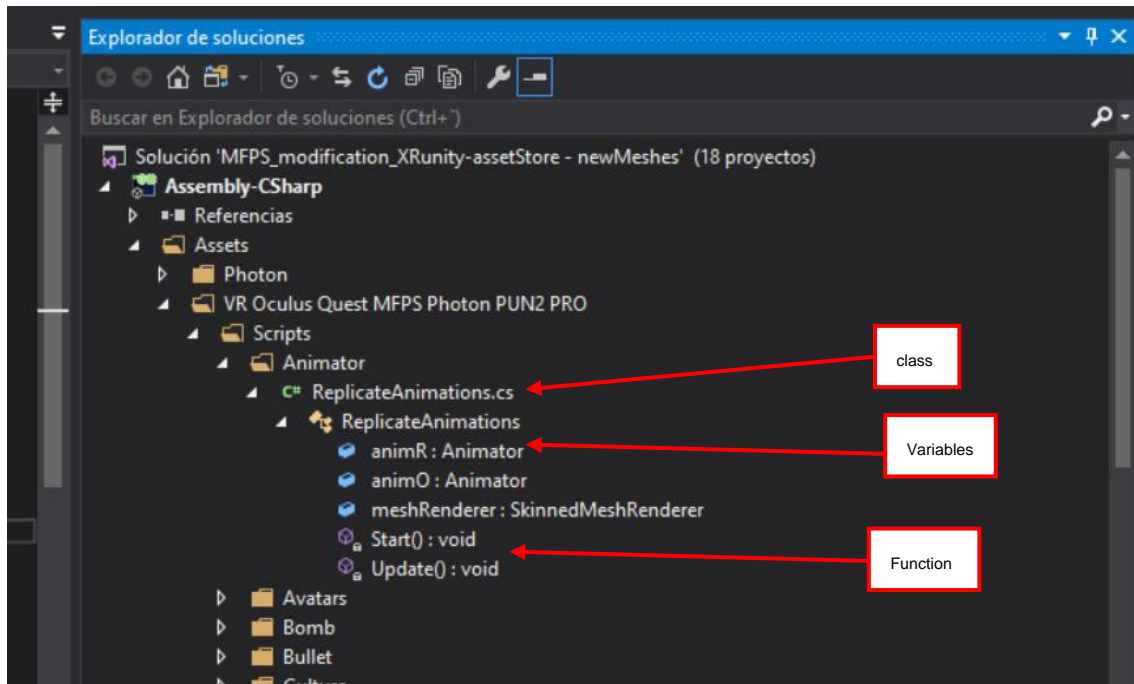
#### 4.11. SHOOTING AND BULLETS



The shooting interface is centralized in the previous script.

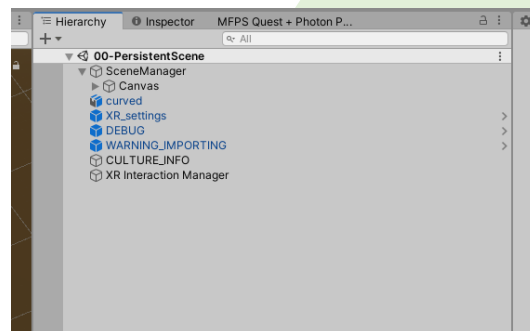
## 1. SCRIPTING

The package has more than 35 scripts fully commented using also regions. Now we explain how to use VISUAL STUDIO to check the different variables and functions. This is useful to know how scripts works in a general way. In this documentation we will only focus on how the scripts are displayed in the editor.



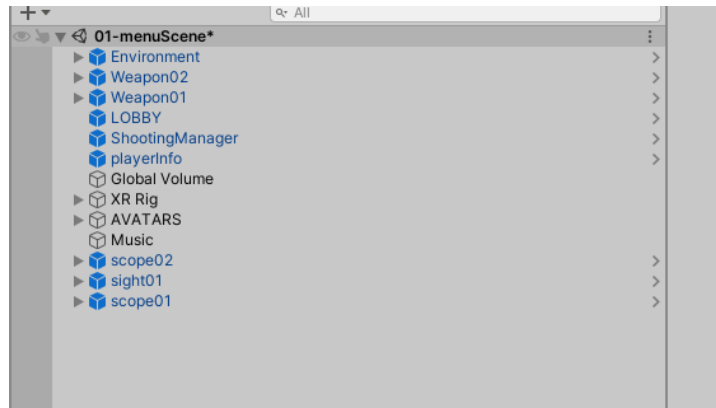
## 2. SCENES

### 2.1. 00-PERSISTENTSCENE



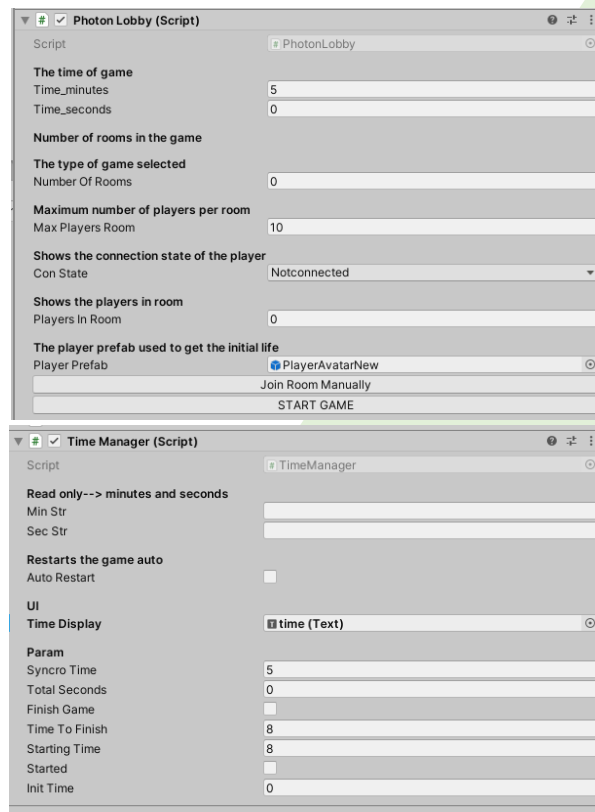
It is the basic scene in which the developer can show its logo and performs an initial check as well as setting the culture info.

### 2.2. 00-MENUSCENE



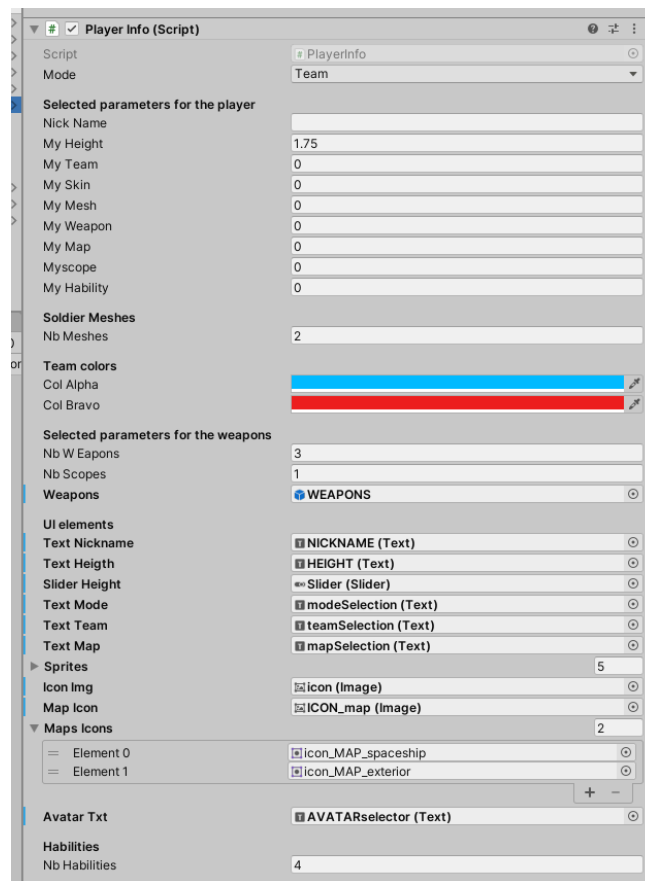
Where the main photon scripts are as well as the starting menu that allows selecting the parameters.

“LOBBY” gameobject contains the photon login scripts as well as the time\_manager.

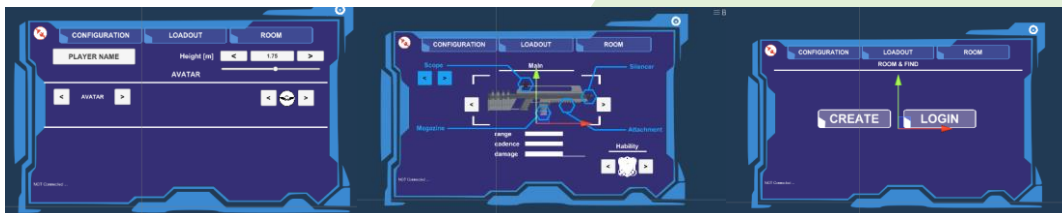


The login process is separated into joining a room and starting the game. This allows the users to join an open room that is controlled by the host.

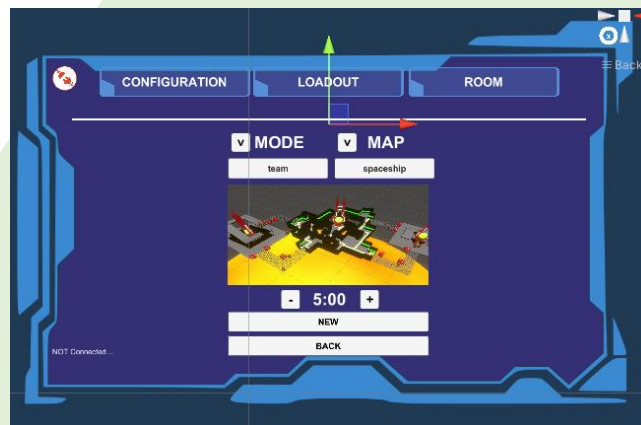
The “playerInfo” contains the selected information of the user. The “my[...]” values are the ones that the user selects in the menu.



If we talk about the menu, it is aranged in these parts:



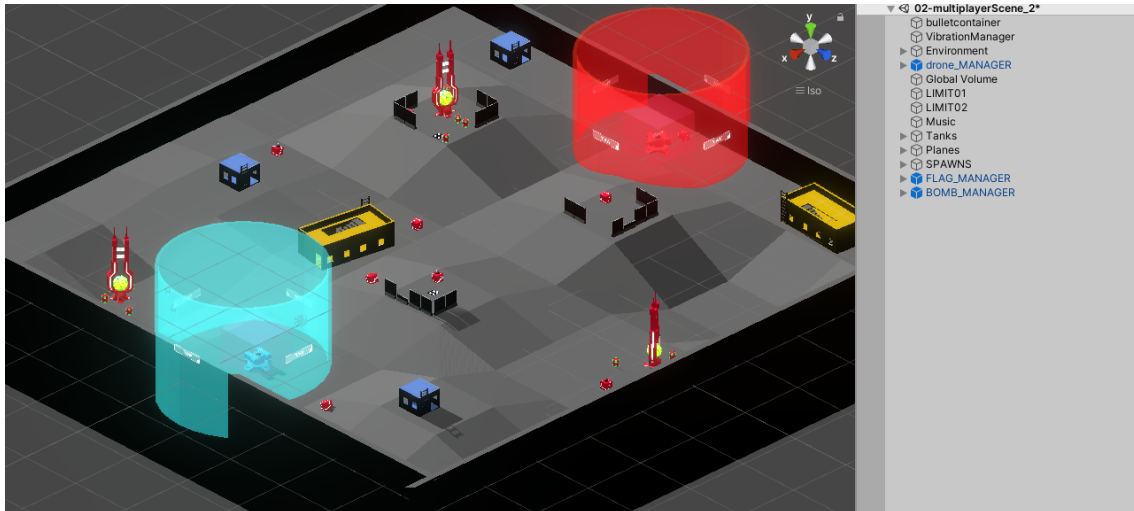
The first sceen is used to select the name, height and avatar of the player. The second, it is used to change the weapon parameters. The last screen is used to connect to a existing room or to create a new room. The new room menu is:





The player can select the mode, the map and the game time.

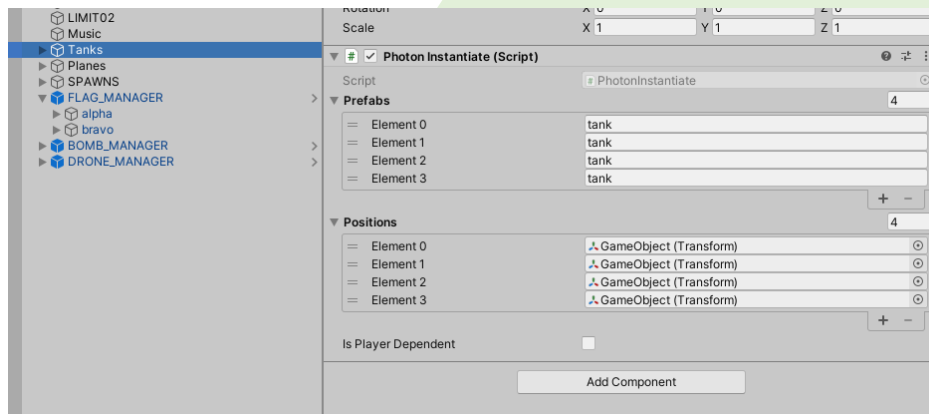
### 2.3. MULTIPLAYER SCENES



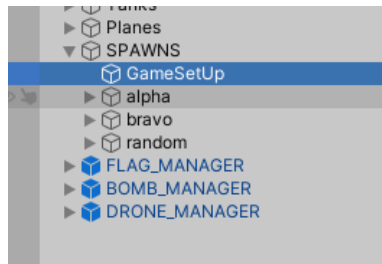
There are the actual gaming scene. They have a similar structure. The “drone\_manager” creates the enemy drones in the game.

The flag manager creates the flags and prepares the scene if the gamemode flag is selected. The bomb manager prepares the scene if the bomb gamemode is selected.

Tanks and planes creates the “scene objects” for the tanks and planes. This is managed using a “photonInstantiate” script. For this script you need to introduce the name of the photon prefab and its spawn position.



“SPAWNS” is used for selecting where the players will be spawned in function of the game mode.



The “gamesetup” is used to manage the re-start and the spawns at the same time.

### 3. VIDEO-TUTORIALS

Here you will find:

#### 3.1. PROMOTIONAL VIDEO

[https://youtu.be/VVMoCw3P\\_DY](https://youtu.be/VVMoCw3P_DY)

#### 3.2. HOW TO CHANGE/ADD WEAPONS

**Primary weapon**

<https://youtu.be/MxK16P5vxMY>

**Secondary weapon**

<https://youtu.be/hu5vjLYh23I>

#### 3.3. HOW TO CHANGE/ADD AVATARS

<https://youtu.be/ZOUAT7jmHIY>

#### 3.4. HOW TO CHANGE/ADD MAPS

[https://youtu.be/9\\_-goYkFpko](https://youtu.be/9_-goYkFpko)

#### 3.5. HOW TO CHANGE VEHICLES

<https://youtu.be/Th1th2M7bew>

### 4. QUESTIONS?

Discord: <https://discord.gg/WqSxEU3>

Contact us at [michael.soler.beatty@gmail.com](mailto:michael.soler.beatty@gmail.com)

