Multiplayer VR Template Fusion

Documentation

Thanks again for downloading the asset.

As **IRONHEAD Games**, we will do our best to help you set up the asset and provide the support needed in the future.

This asset uses Photon Fusion for Multiplayer and uses Unity

Also, this asset is compatible with at least Unity 2021.3 LTS. So, make sure not to use a lower version than Unity 2021.3 LTS!

The suggested Unity version is Unity 2021.3 LTS.

You are welcome to ask your questions to email address:

tevfikufuk@ironheadgamestudios.com

Also, here is the Asset's Discord server to get in touch better:

https://discord.gg/3yPnkxs

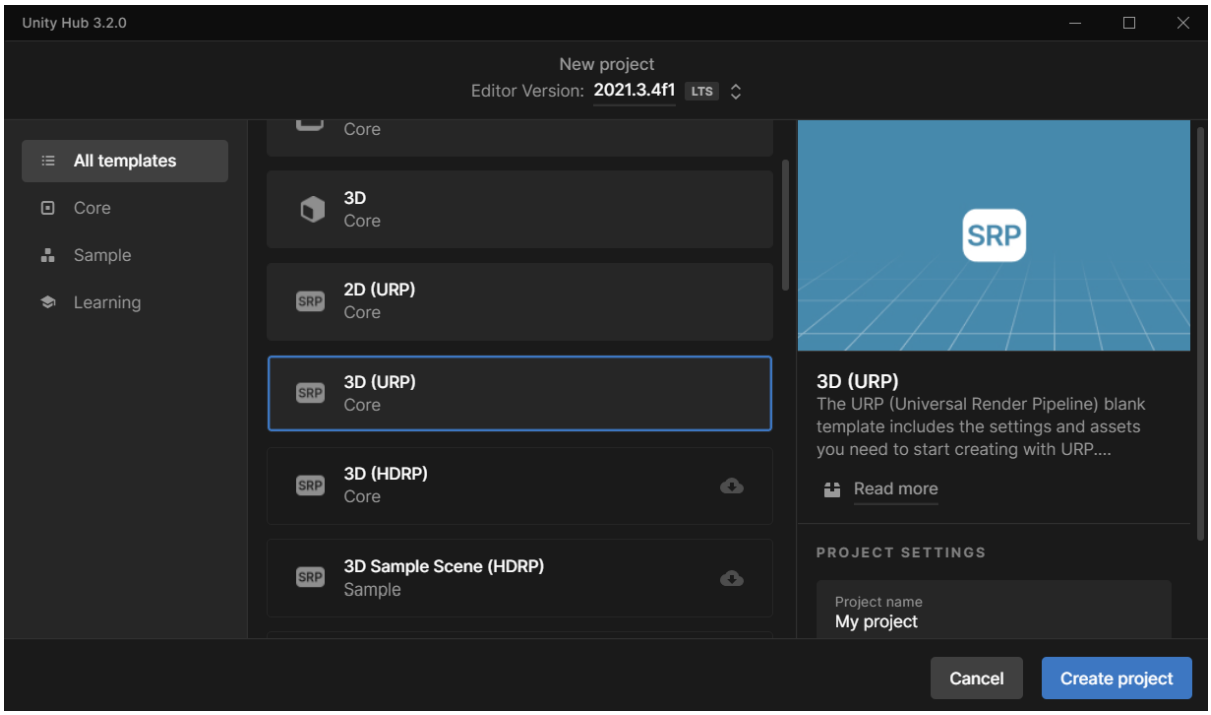
You can follow the below steps to set up the asset. Good luck!

How to Set Up the Asset

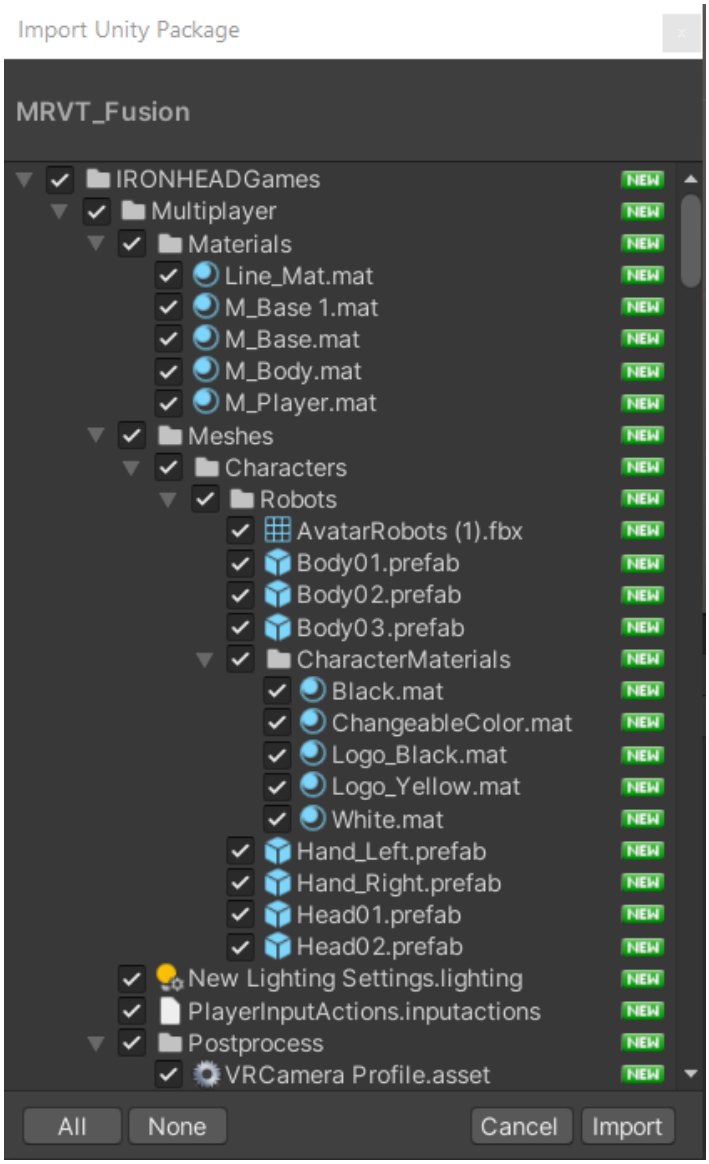
Importing the Asset and Configuring Photon Settings

1. Creating New Project

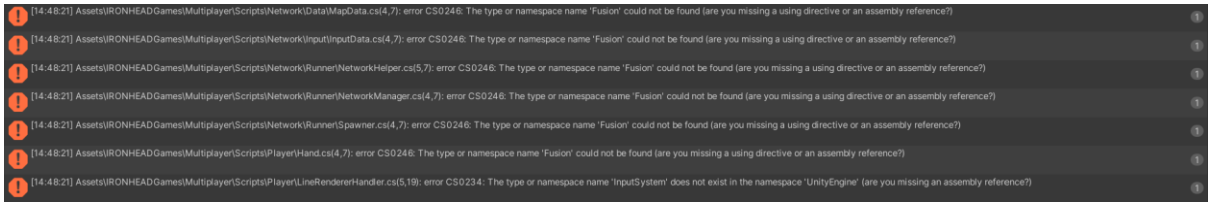
1.1. Firstly, create a new fresh Unity project with the default 3D(URP) core template.



1.2. Then, import the asset via Package Manager

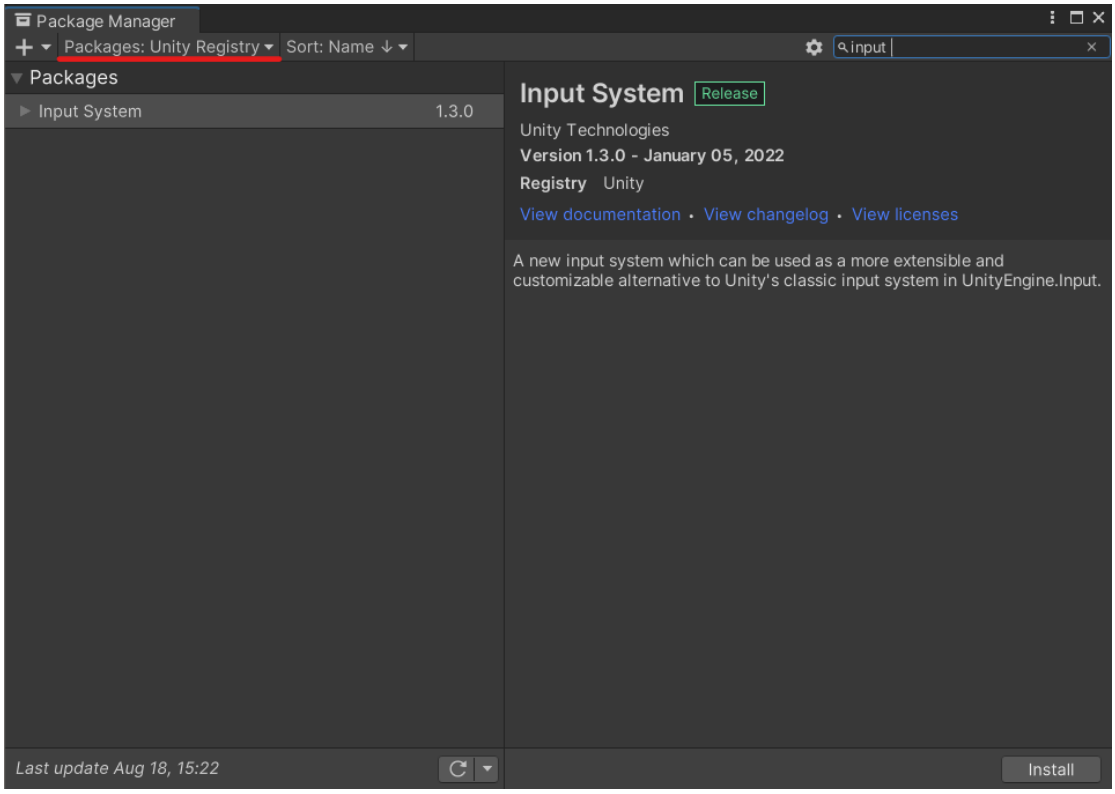


After importing the asset, you may see lots of errors on Unity Console. This is very normal, and we will fix it soon.

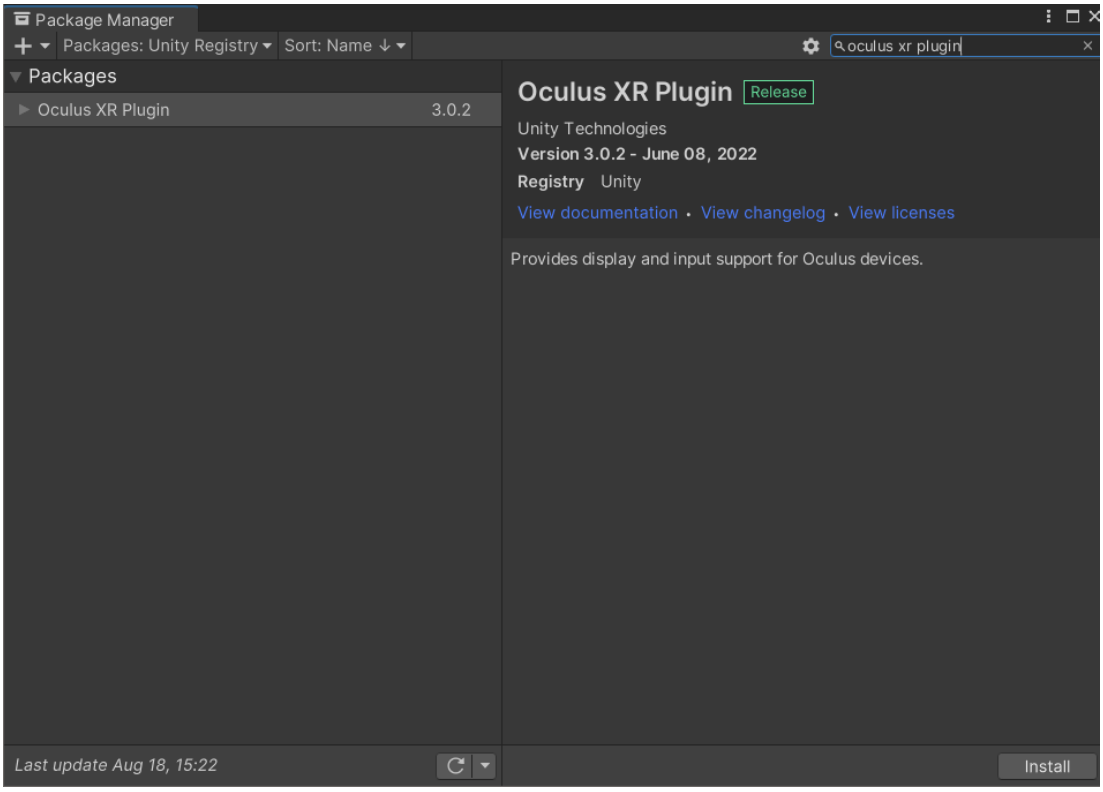


2. Importing Required Packages

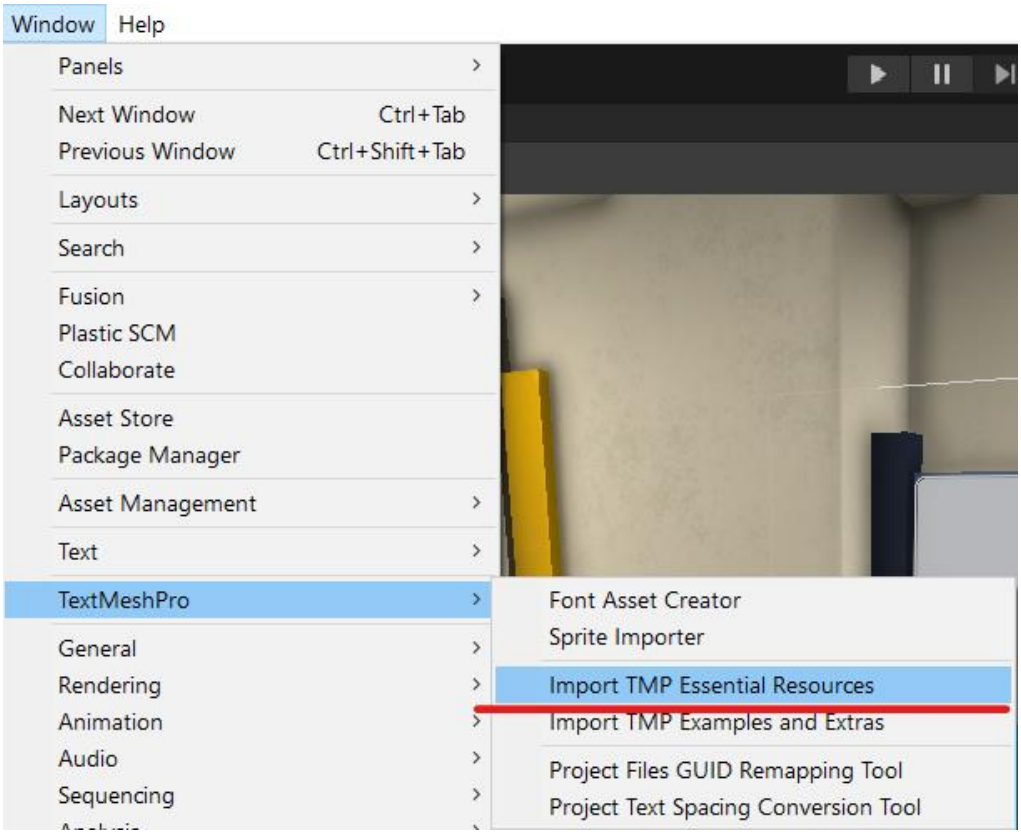
2.1. Input System: In this template, we use new unity input system. From **Window -> Package Manager** change packages from MyAssests to Unity Registry. Then, search “Input System”, import and install the package.



2.2. Oculus Xr Plugin: To track hand and headset inputs we use Oculus XR Plugin. From **Window -> Package Manager** change packages from MyAssests to Unity Registry. Then, search “Oculus XR Plugin”, import and install the package.

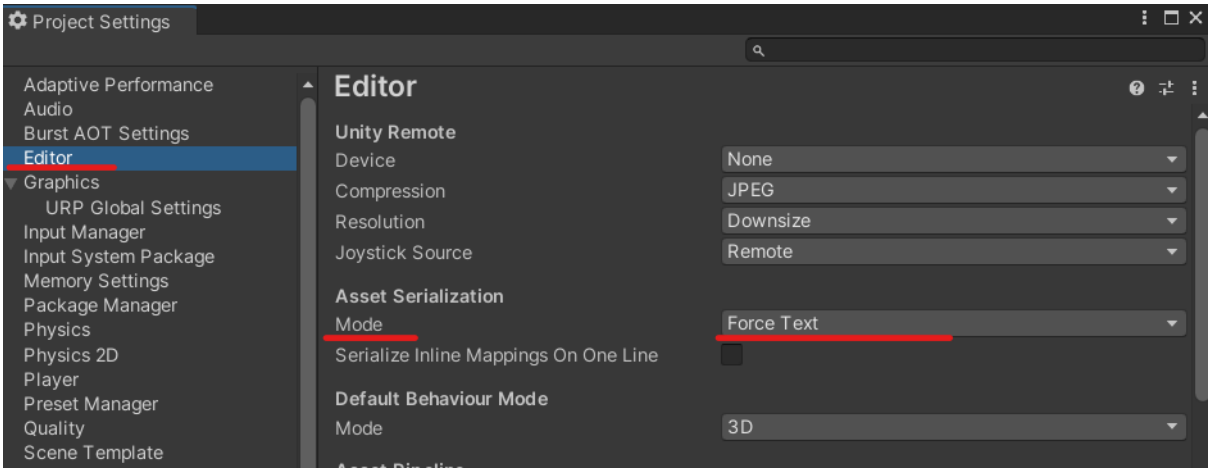


2.3. TMPro: Some of the Texts in the template may not be shown correctly, to fix this problem Navigate to **Window -> TextMeshPro** and click **Import TMP Essential Resources**.

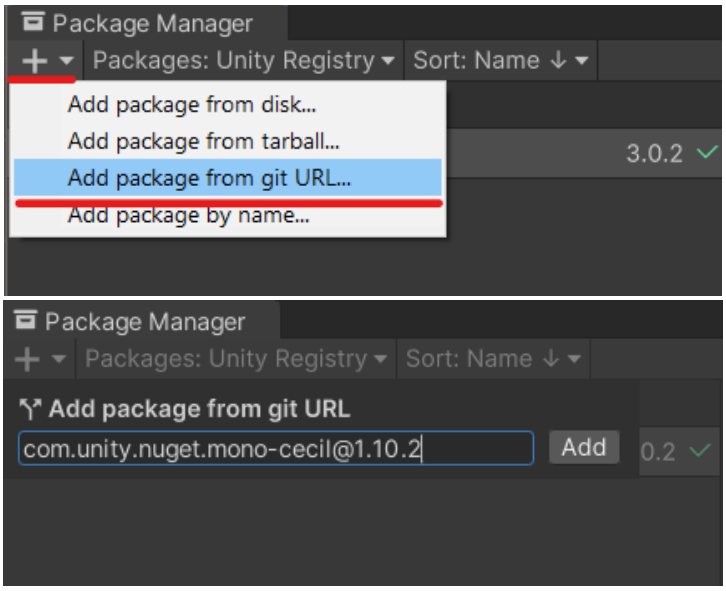


3. Preparing Project for Fusion

3.1. Asset Serialization: Some of the Fusion settings will be saved in ScriptableObject Assets. For these settings, we need to change the serialization mode to **Force Text.** Navigate to: **Edit -> Project Settings -> Editor -> AssetSerization -> Mode** to **Force Text.**



3.2. Mono Cecil: The Fusion IL Weaver generates low-level netcode based and injects it into the Assembly-CSharp.dll. To achieve this, the Mono Cecil package is used.
To import Mono Cecil, navigate to **Windows -> Package Manager -> "+" icon -> Add package from git URL** and add **com.unity.nuget.mono-cecil@1.10.2**

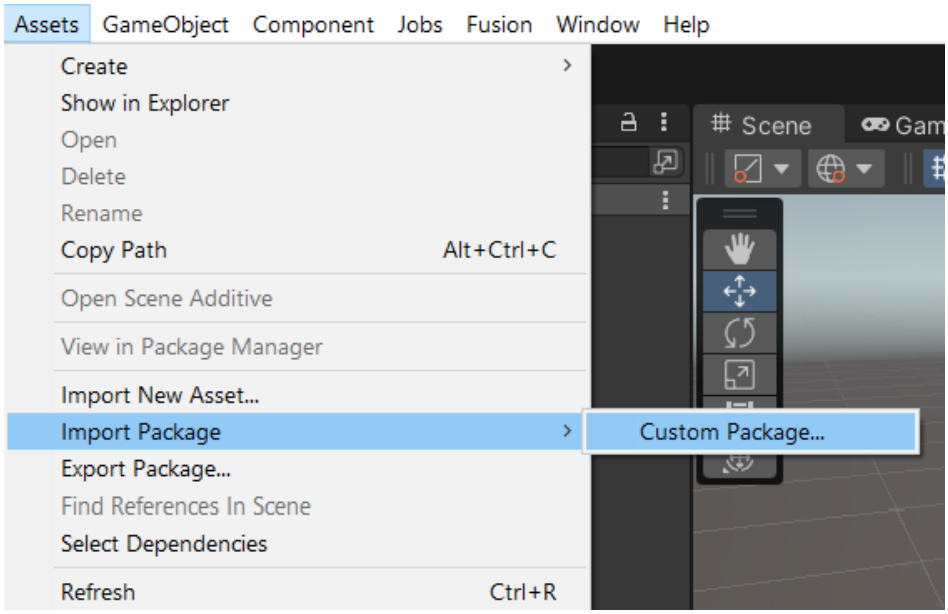


3.3. Photon Fusion SDK: With all steps done we are ready to install and import Fusion SDK.
To download Fusion SDK navigate to https://doc.photonengine.com/en-us/fusion/current/getting-started/sdk-download
We recommend using stable build.

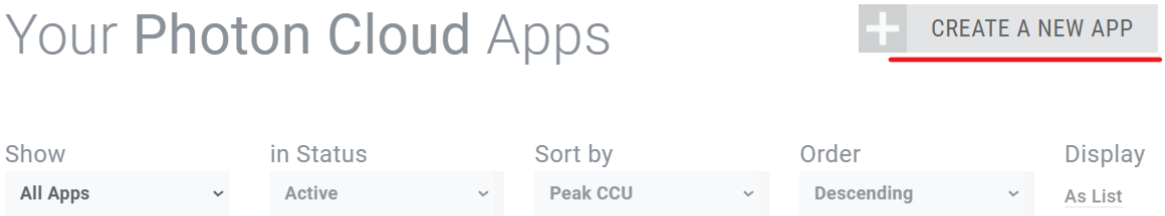
Nightly Builds

Version	Release Date	Download	
1.1.2 Nightly	Aug 17, 2022	Fusion SDK 1.1.2 Nightly Build 558	Release Notes

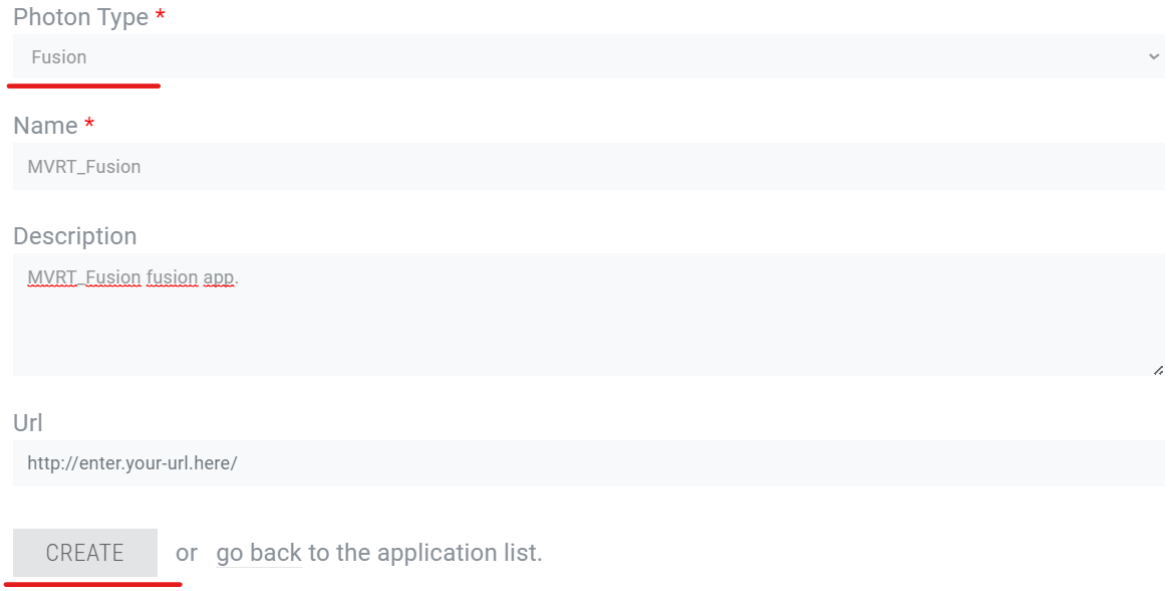
After downloading SDK, navigate to **Assets -> Import Package -> Custom Package.** Import package from where the SDK was downloaded.



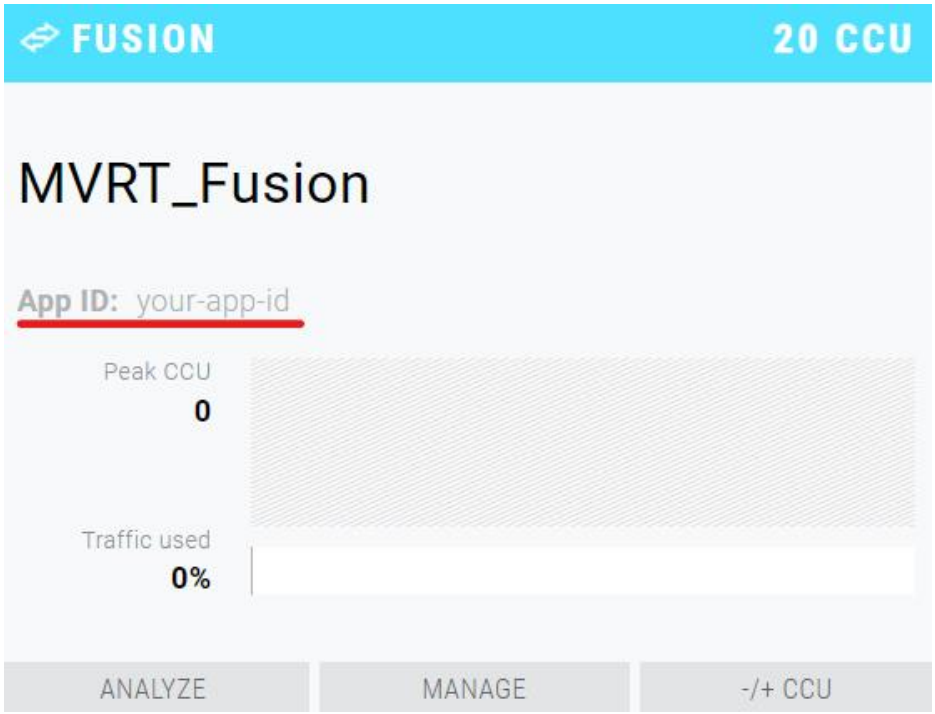
3.4. Create An App ID: Navigate to https://www.photonengine.com/en-US/Photon and sign in. After signing in navigate to **Dashboard** and click to **Create New App.**



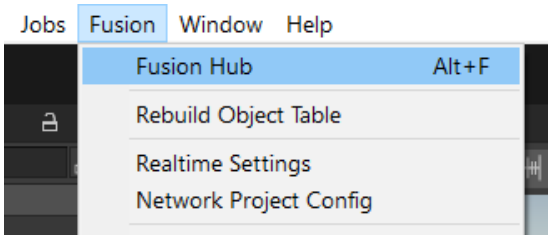
Set **Photon Type** to **Fusion** and write the name and description as you like. After all is set click **Create.**

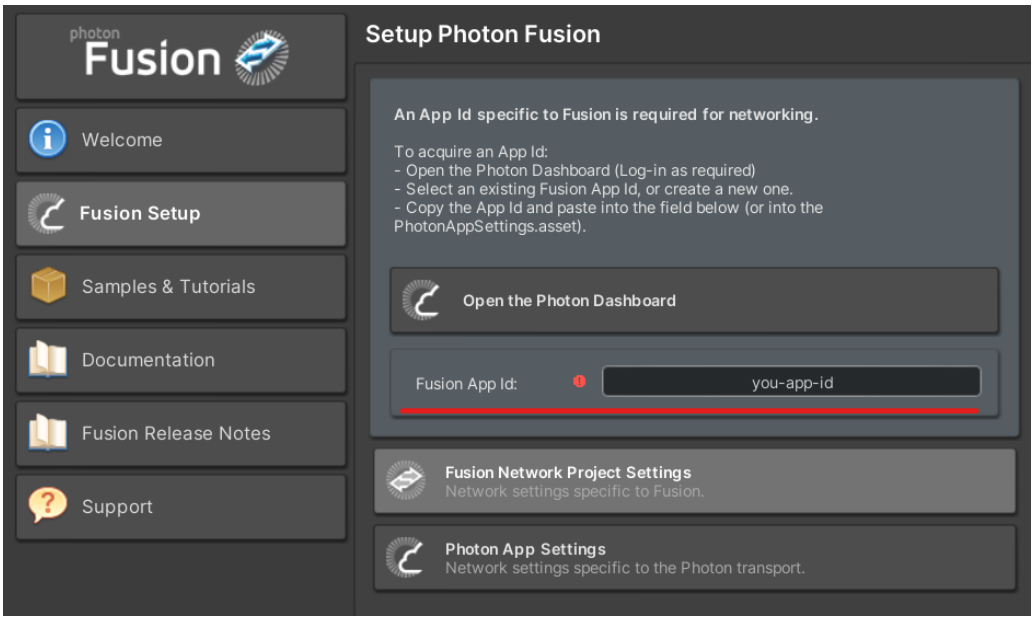


3.5. Using App ID: After creating new app we can use App ID in our Unity project. Navigate to Dashboard and copy App ID.



Now, we can return Unity and navigate to **Fusion -> Fusion Hub –> Fusion Setup** and paste your App ID.

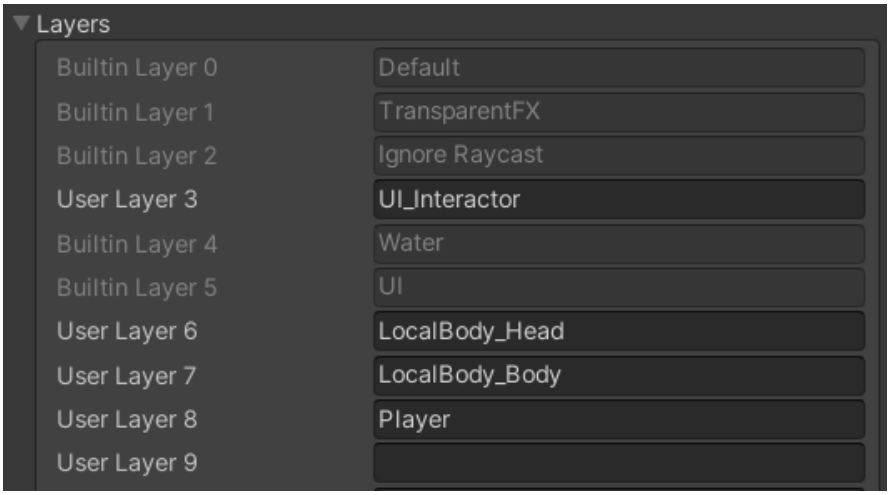




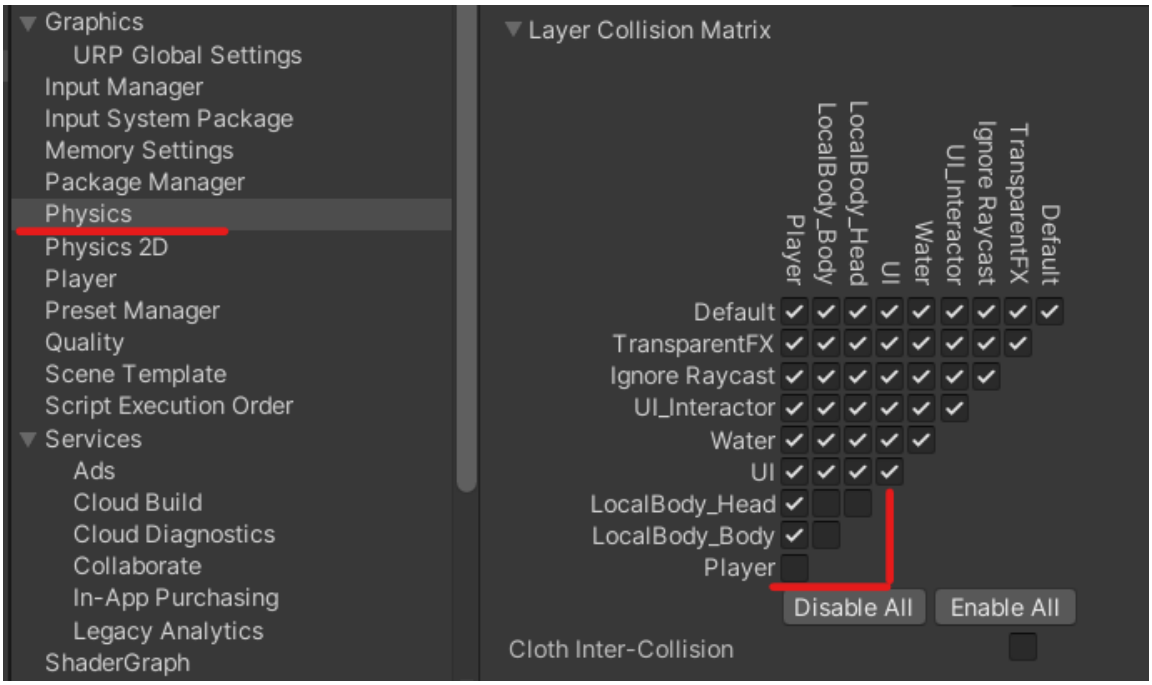
Now your project ready to use Photon Fusion.

4. Unity Template Settings

4.1. Adding Layers: Some of the game objects have custom layers to set. Add following layers to the project. Note that the order is important.



4.2. Setting Up Collision Layers: Navigate to **Edit -> Project Settings -> Physics -> Layer Collision Matrix** and disable the collision between the Player/Player, LocalBody_Head/ LocalBody_Head ,LocalBody_Body/ LocalBody_Body, LocalBody_Head/ LocalBody_Body and layer.

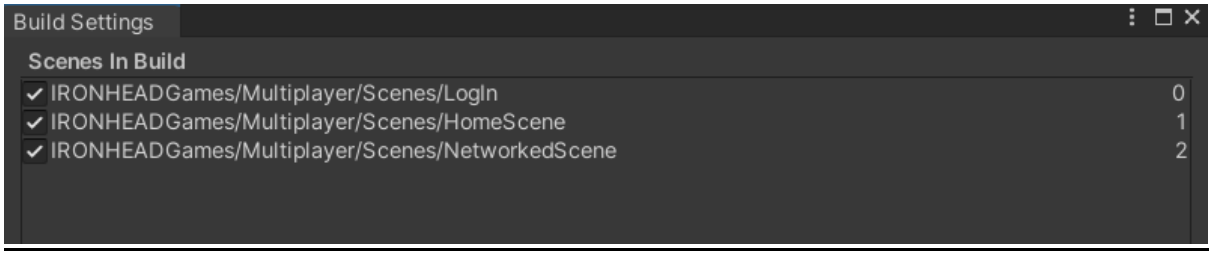


5. Configuring the Project for Oculus Quest2 Build

Now, we will prepare the project for Oculus Quest and Quest 2 build. And we will mainly follow the settings from here:

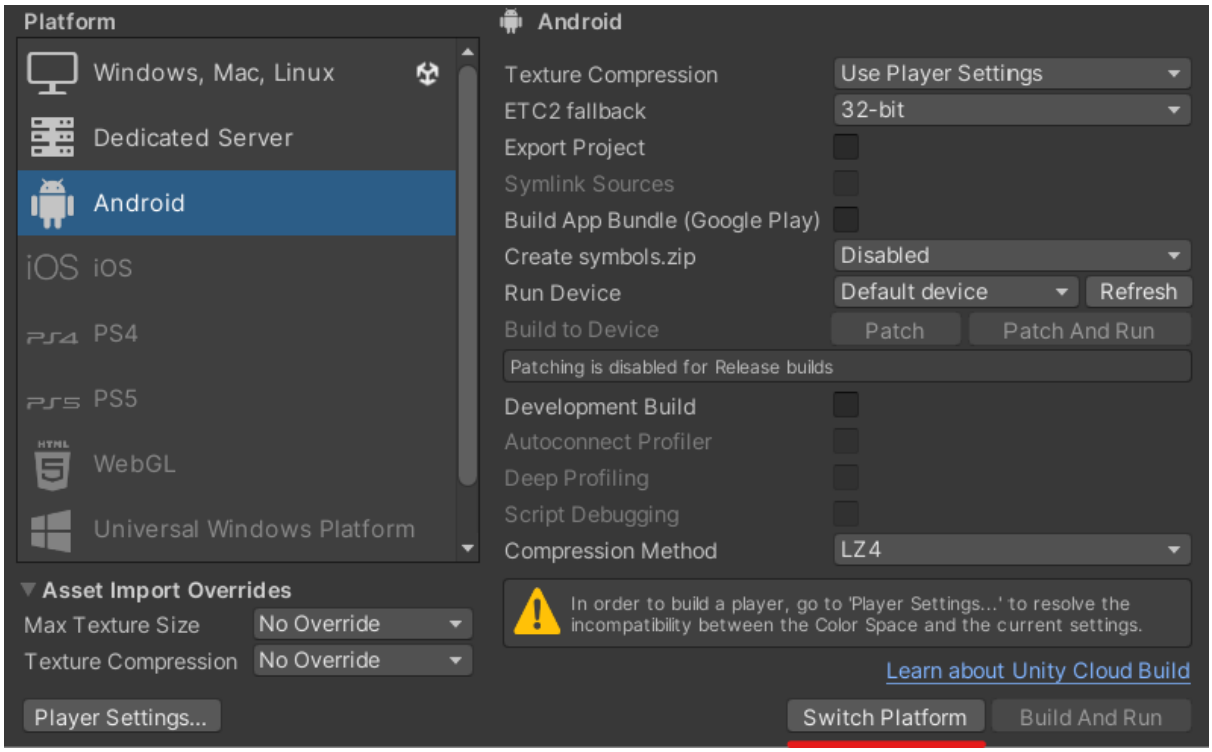
https://developer.oculus.com/documentation/unity/unity-conf-settings/

5.1. Adding Scenes to Build Settings: Navigate to **File -> Build** Settings and add the following scenes to Scenes in Build from **IRONHEADGames -> Multiplayer -> Scenes** folder.

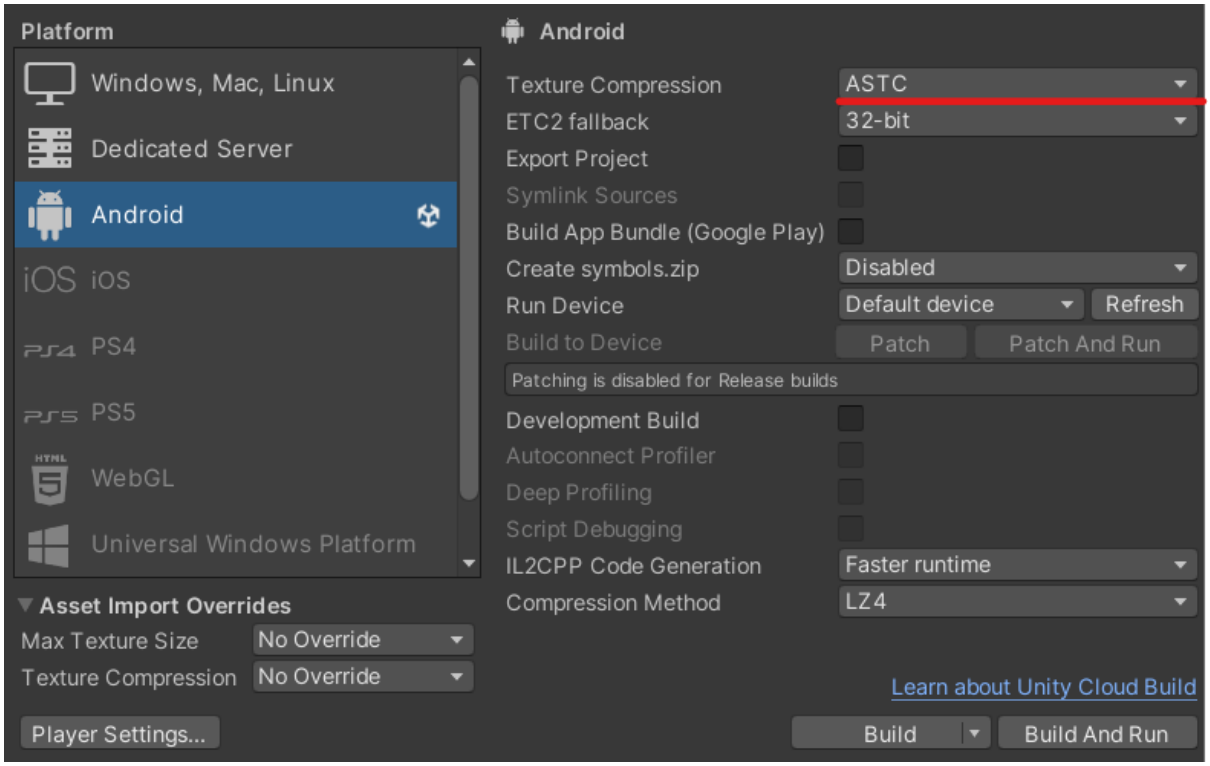


Make sure to follow the order such that the **LogIn** scene must be the first scene.

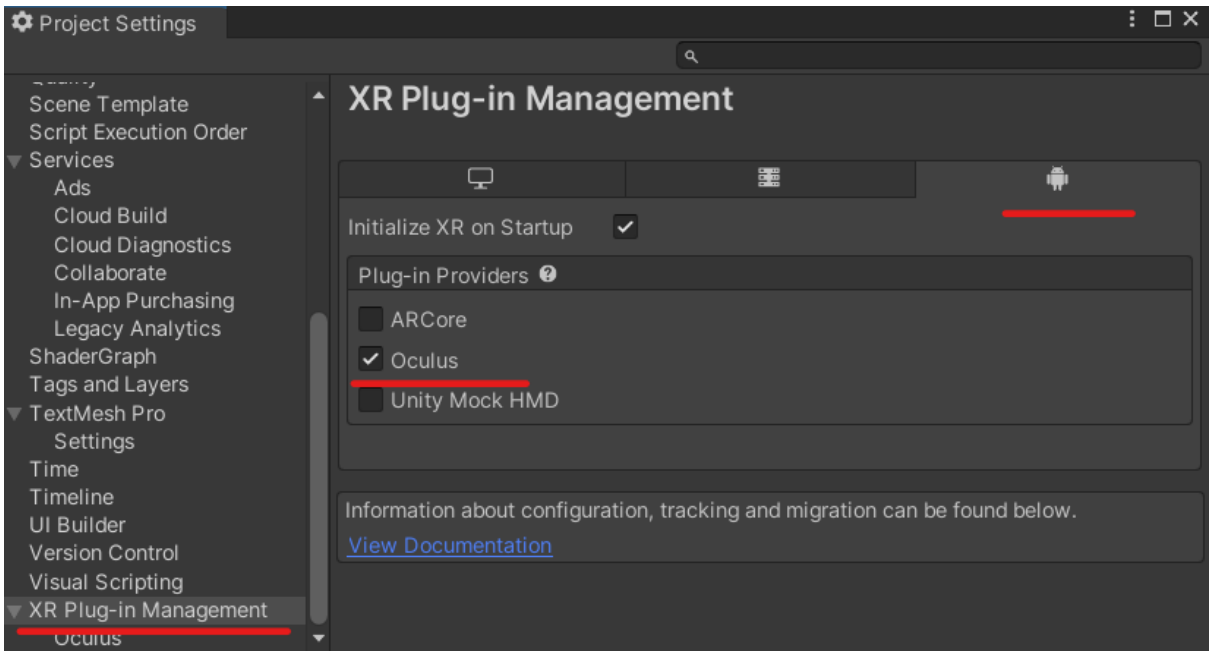
5.2. Switching Platform: Oculus Quest 2 is an Android device. So, the build file needs to be .apk file. Navigate to **File -> Build**, change platform to Android, and **click Switch Platform**.



5.3. Texture Compression: In same panel (Build Settings) change **Texture Compression** to the **ASTC**.

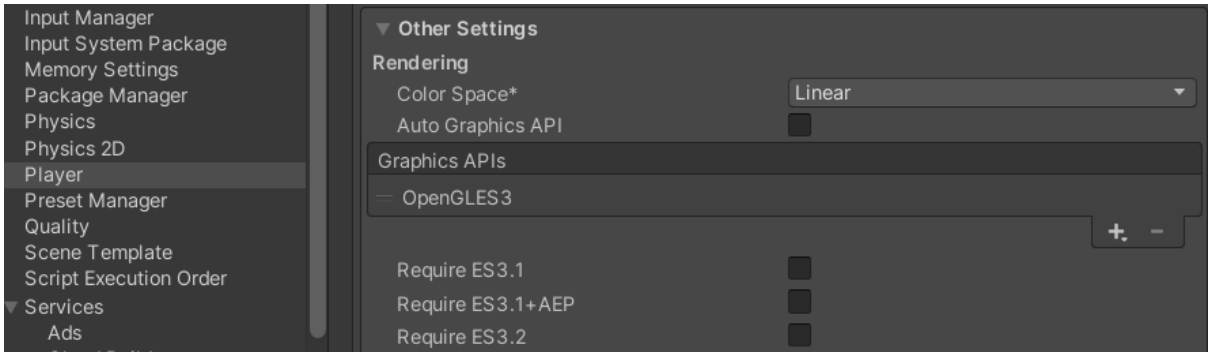


5.4. XR Plugin Management: Navigate to **Edit -> Project Settings -> XR Plug-in Management -> Android Tab** and select Oculus.



5.5. Player Settings: Navigate to **Project Settings -> Other Settings.** In this part we will change several options.

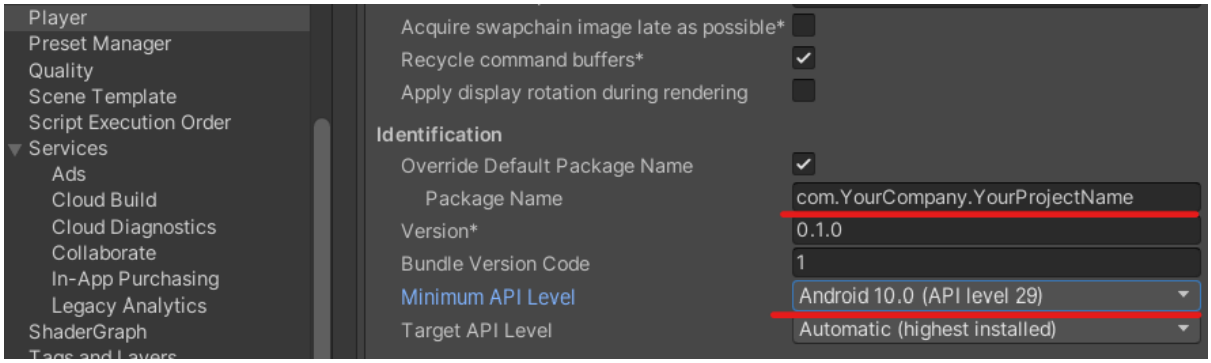
- Set Color Space to **Linear.**
- **Deselect** Auto Graphics API.
- Remove **OpenGLES2** from Graphics API. In this part only OpenGLES3 active needs to be active.



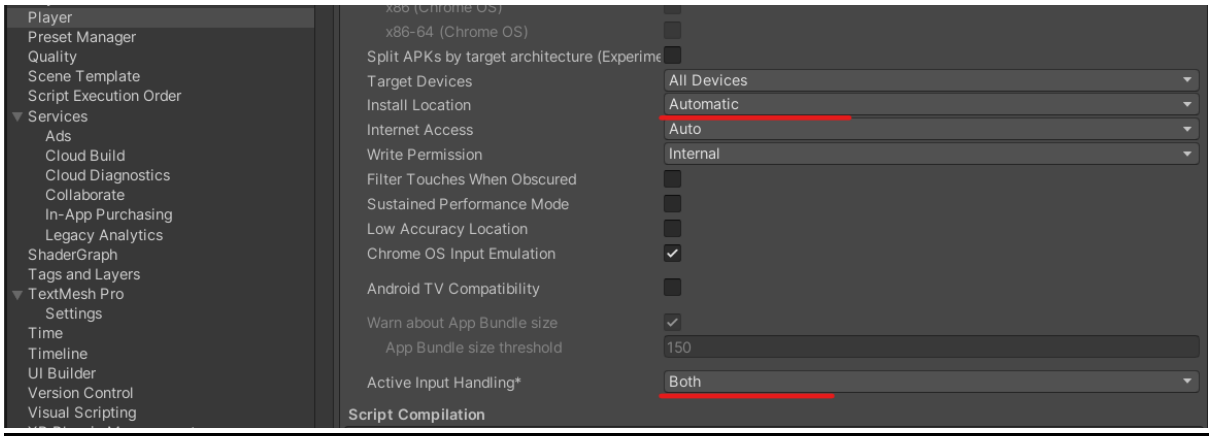
See more info:

https://developer.oculus.com/documentation/unity/unity-conf-settings/

- Under the identification part check Override Default Package Name and enter your unique package name.
- Change Minimum API Level to **Android 10.0 (API level 29)**.

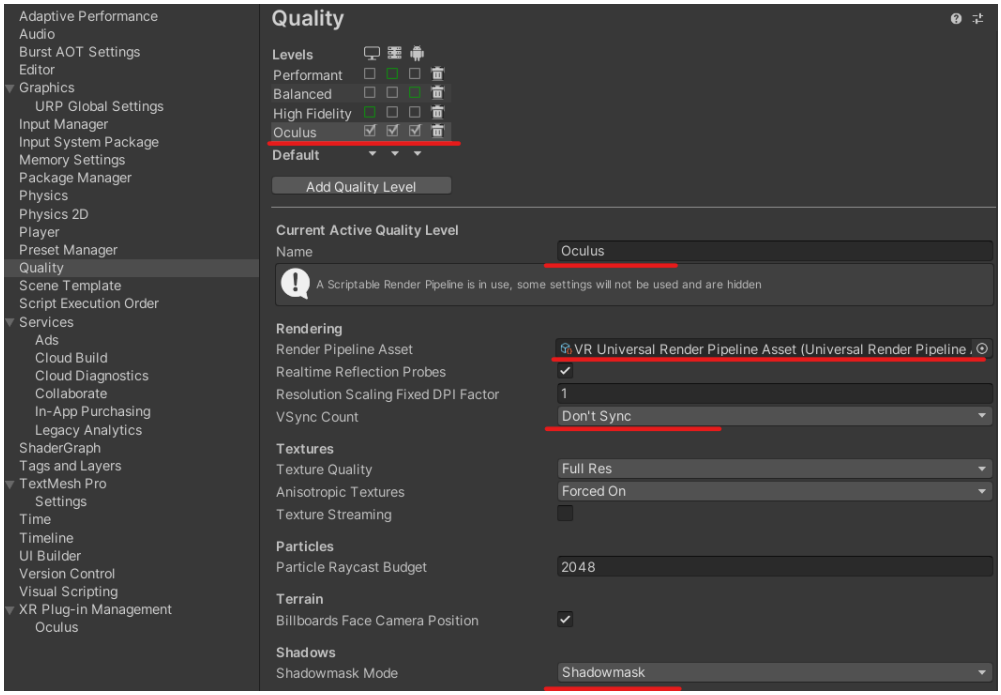


- Under the Configuration part change the Install Location to **Automatic**.
- Change Active Input Handling to **Both**.

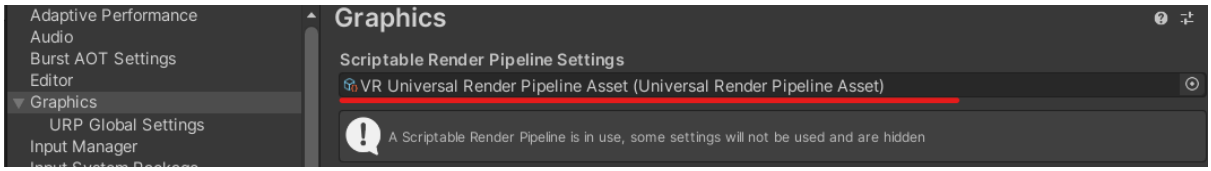


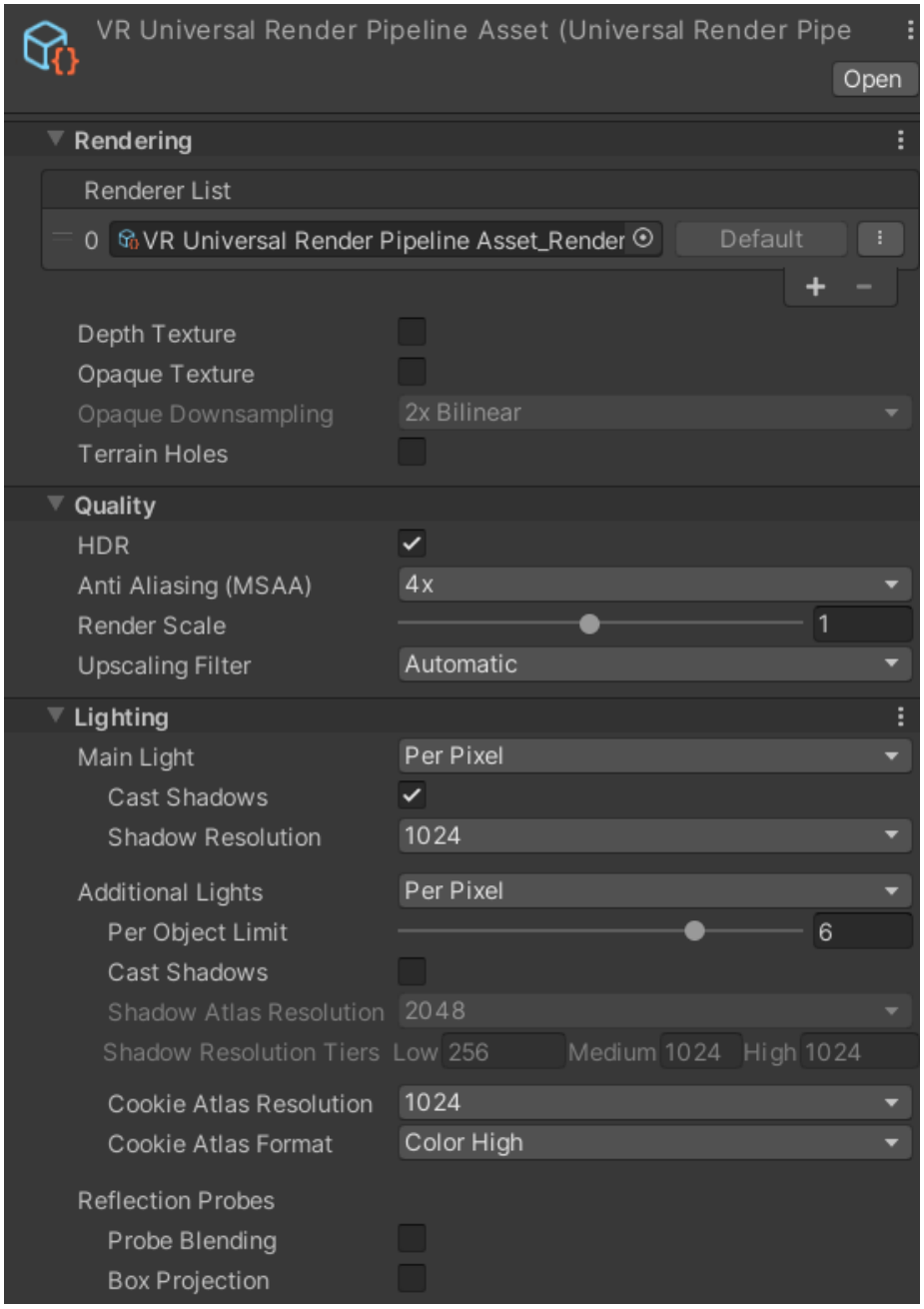
5.6. Quality Settings: Navigate to **Edit -> Project Settings -> Quality**.

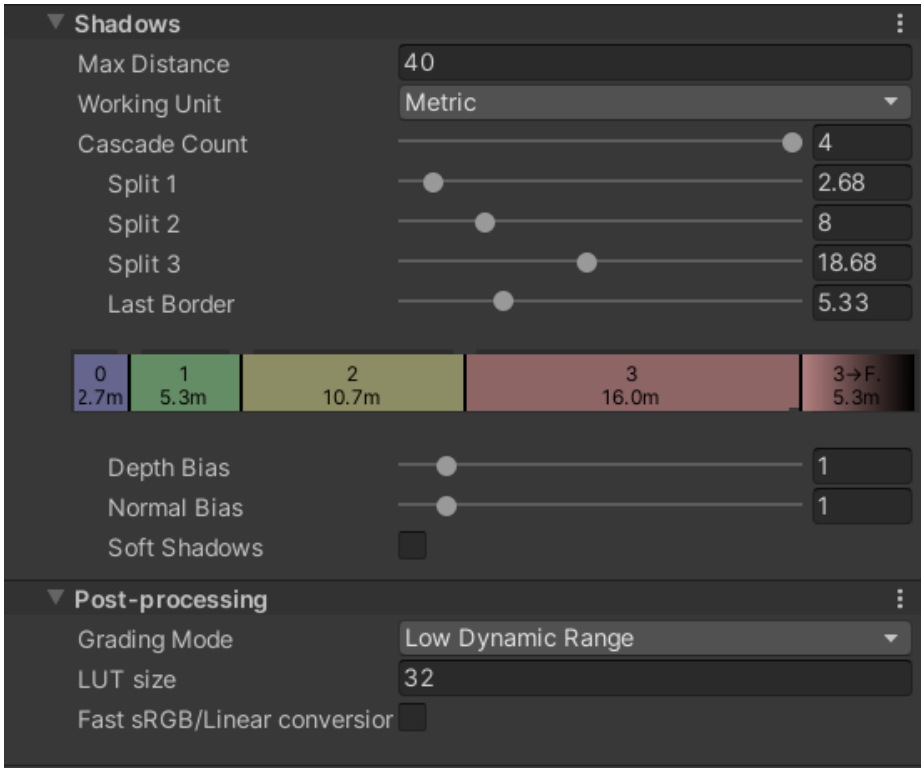
- Click Add Quality Level and rename it as “Oculus”.
- After creating new quality level deselect all other quality levels from the Android column.
- Change Render Pipeline Asset to **VR Universal Render Pipeline Asset**.
- Change VSync Count to **Don’t Sync**.
- Change Shadowmask Mode to the **Shadowmask**.



5.7. Graphic Settings: Navigate to Edit -> Project Settings -> Graphics and change Scriptable Render Pipeline Settings to **VR Universal Render Pipeline Asset**. This asset contains graphic options for VR headset.




VR Universal Render Pipeline Asset (Universal Render Pipe
Open
Rendering
Renderer List
0
VR Universal Render Pipeline Asset_Render
Default
Depth Texture
Opaque Texture
Opaque Downsampling
2x Bilinear
Terrain Holes
Quality
HDR
Anti Aliasing (MSAA)
4x
Render Scale
1
Upscaling Filter
Automatic
Lighting
Main Light
Per Pixel
Cast Shadows
Shadow Resolution
1024
Additional Lights
Per Pixel
Per Object Limit
6
Cast Shadows
Shadow Atlas Resolution
2048
Shadow Resolution Tiers
Low
256
Medium
1024
High
1024
Cookie Atlas Resolution
1024
Cookie Atlas Format
Color High
Reflection Probes
Probe Blending
Box Projection


Shadows
Max Distance
40
Working Unit
Metric
Cascade Count
4
Split 1
2.68
Split 2
8
Split 3
18.68
Last Border
5.33
0
2.7m
1
5.3m
2
10.7m
3
16.0m
3→F.
5.3m
Depth Bias
1
Normal Bias
1
Soft Shadows
Post-processing
Grading Mode
Low Dynamic Range
LUT size
32
Fast sRGB/Linear conversion