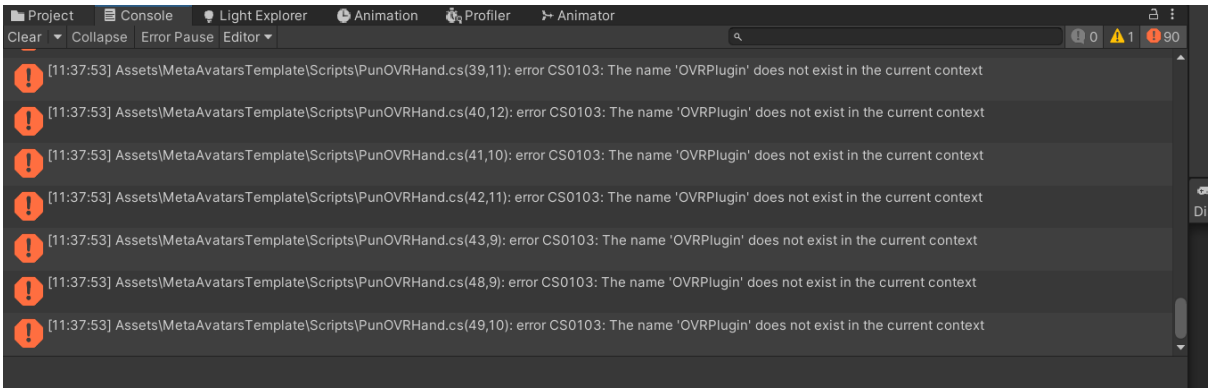
Meta Avatars Template Fusion setup guide.

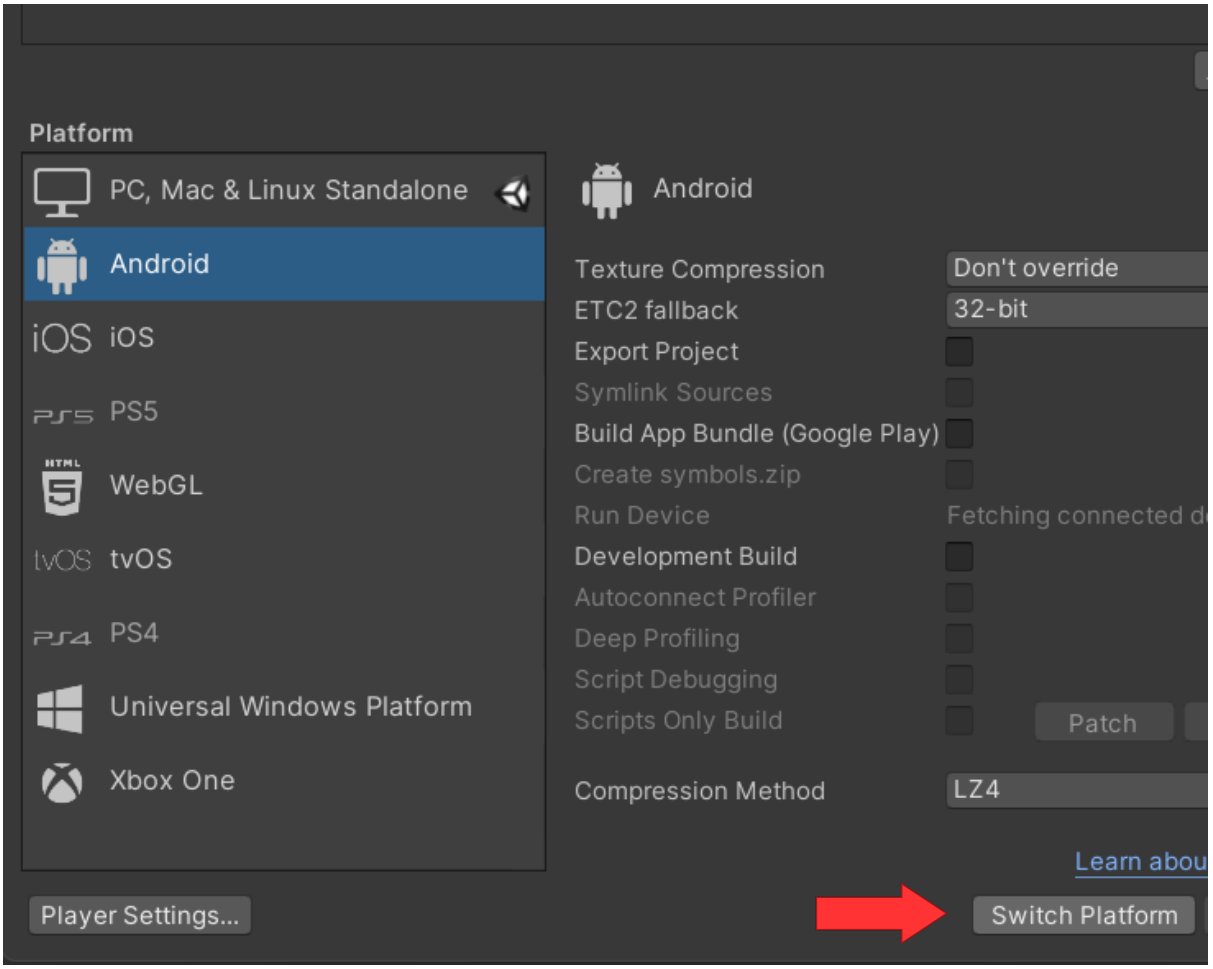
Thanks for purchasing! Join us in the discord server: https://discord.gg/gEhHu8Xydr

After importing the package to a fresh Unity project, you will see many errors. Don't panic! That’s because we still need to import some other packages for it to work.



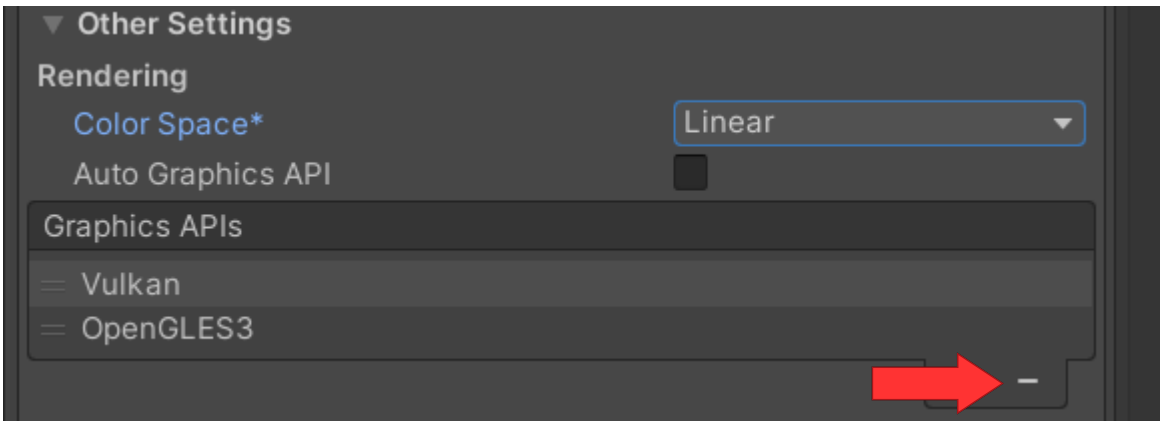
Let’s start setting up Project and Build Settings.

1. Go to build settings and **switch the platform to Android.**



2. **Let's go to Project Settings > Player > Other Settings:**

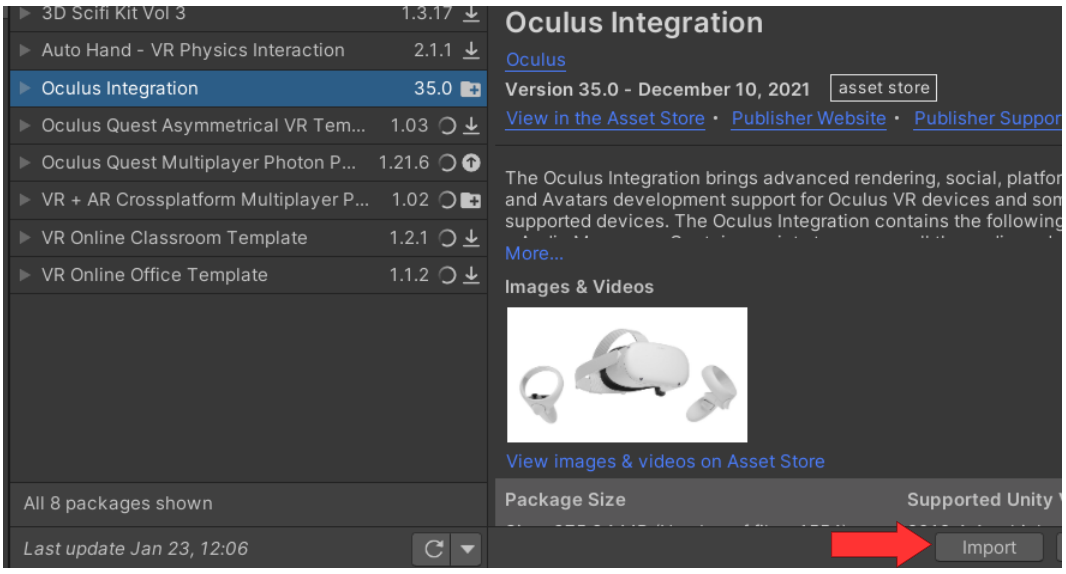
   Change Color Space to "Linear", and remove Vulkan from the Graphics API list:



3. **Set the minimum API Level to 6.0 (API level 23),** switch the scripting backend to IL2CPP, and set the target architecture to 64 bits.

4. **Now, let's import some packages:**

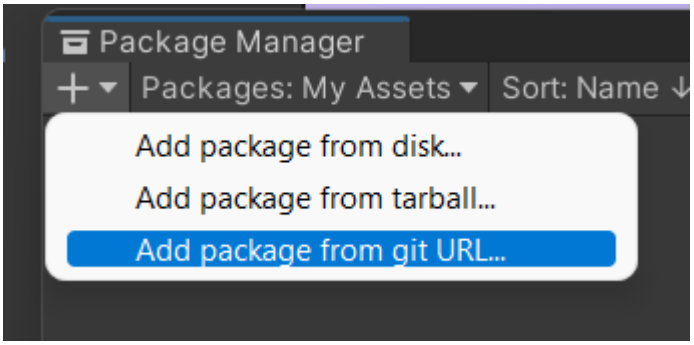
- Oculus Integration v35.0 or higher:

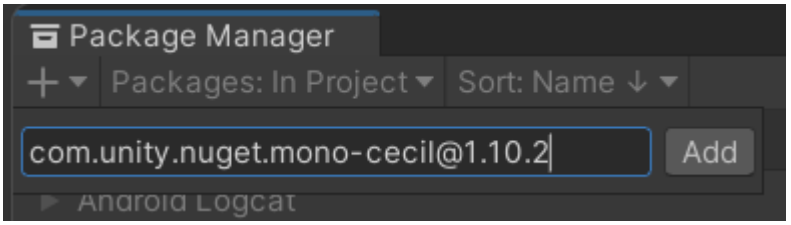


- **Photon Fusion**: Go to https://doc.photonengine.com/en-us/fusion/current/getting-started/sdk-download and download their latest **stable build,** and import it to the project.

  Also, read the requirements below, you might need to install the Mono.Cecil package in the project: *"Mono.Cecil (the com.unity.nuget.mono-cecil@1.10.2 package can be added manually via the package manager if it is missing from the project)."*

This is done by going to the package manager, and clicking on the “+” sign and selecting “Add package from git URL”, and pasting **com.unity.nuget.mono-cecil@1.10.2**





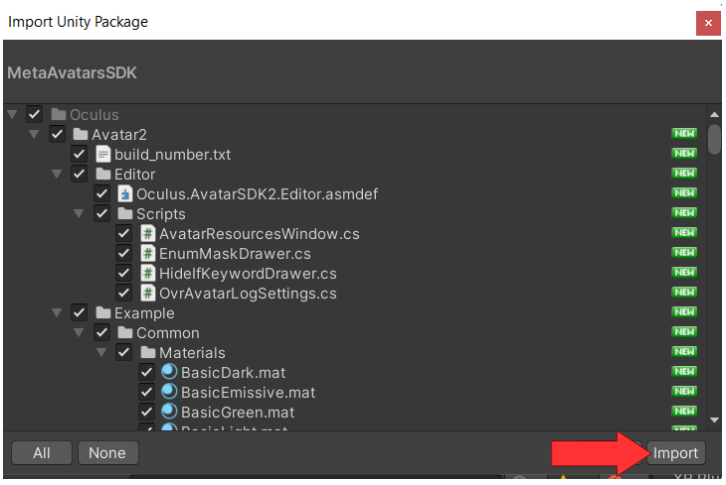
- Photon Voice 2

Import Photon Voice 2 package.

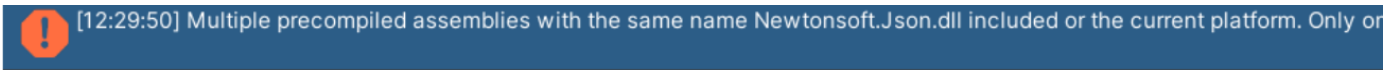
- Go to https://developer.oculus.com/downloads/package/meta-avatars-sdk/ and download the latest version of Meta Avatars SDK

-

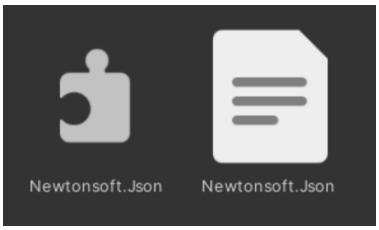
- **Import the MetaAvatarSDK** into your Unity Project.



- If you get this error:

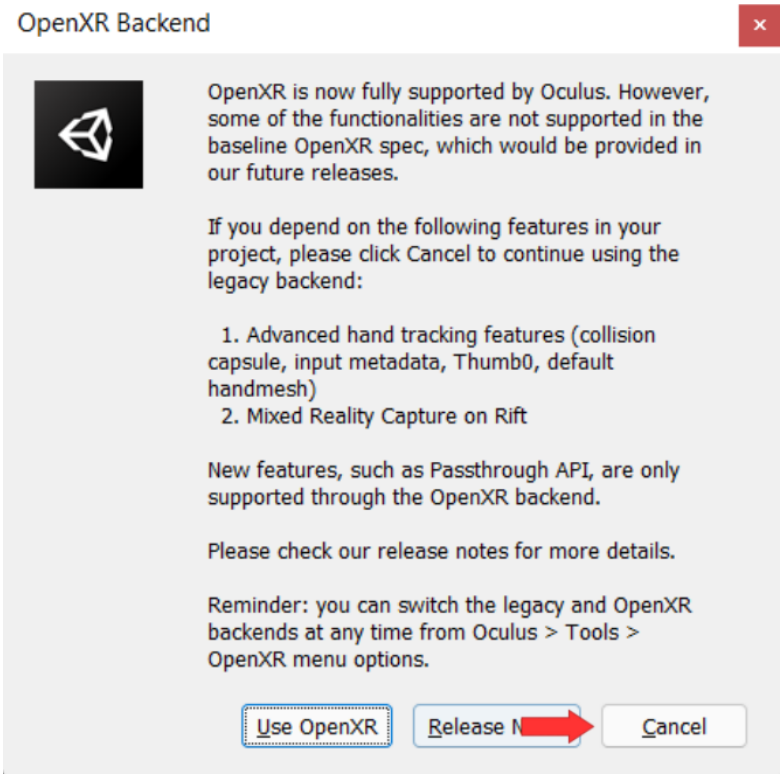


Go to Assets > Oculus > Avatar2 > Scripts > AvatarEdtorDeepLink, and delete these 2 files (they are duplicated).



- Now, the project should compile, and start asking for some updates. Accept all the updates requested.

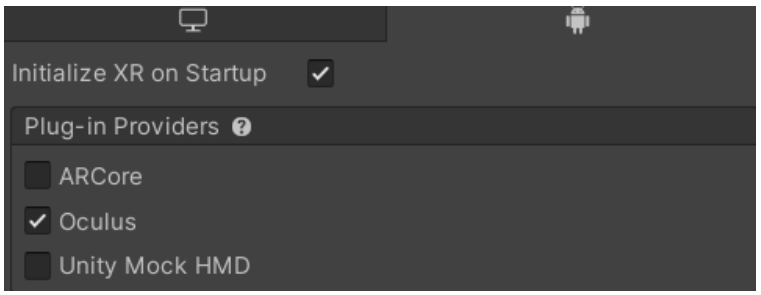
5. **Open XR** backend is optional. For hand tracking, it is recommended to stay with the legacy backend (clicking cancel). **You can freely switch later.**



- Unity will want to restart afterward, click yes.

6. **Installing XR Plugin Management**
   Go to Project Settings > XR Plugin Management and install the plugin.

   Add the Oculus platform for both Windows and Android.

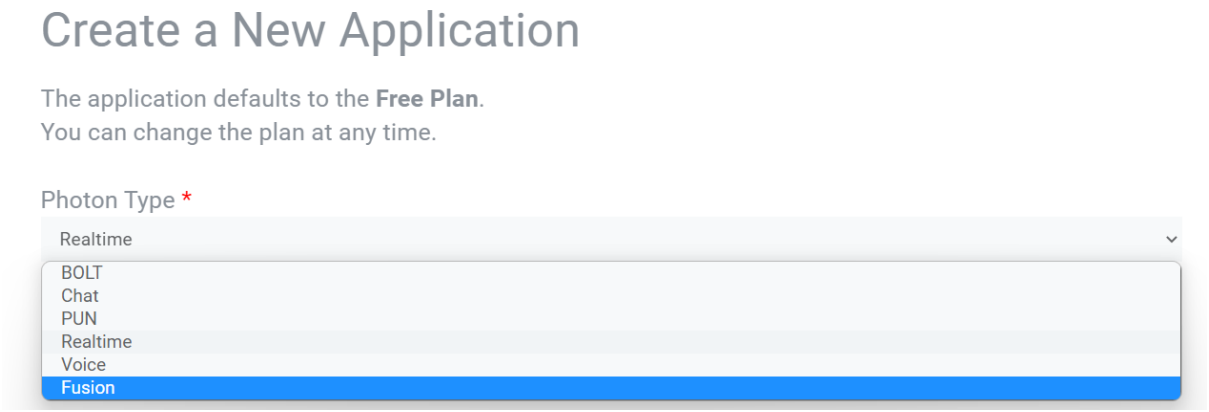


7. **Configuring Photon Fusion**

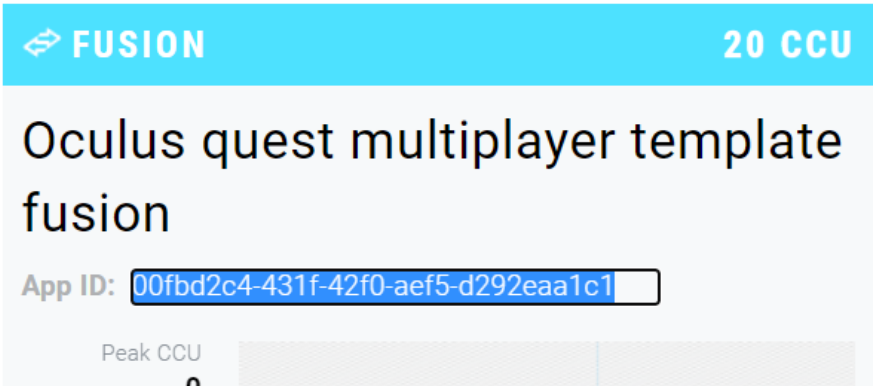
- Go to https://dashboard.photonengine.com/en-US/ and create an account if you don't have one.

- Create a new **Fusion App** (you can create as much apps as you like)

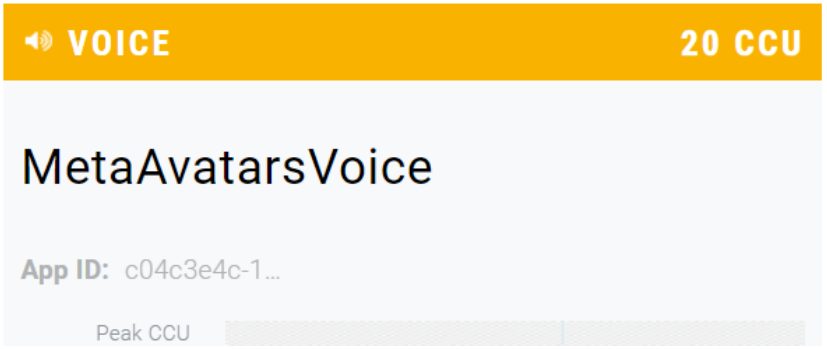
- **Make sure it's Type Fusion**



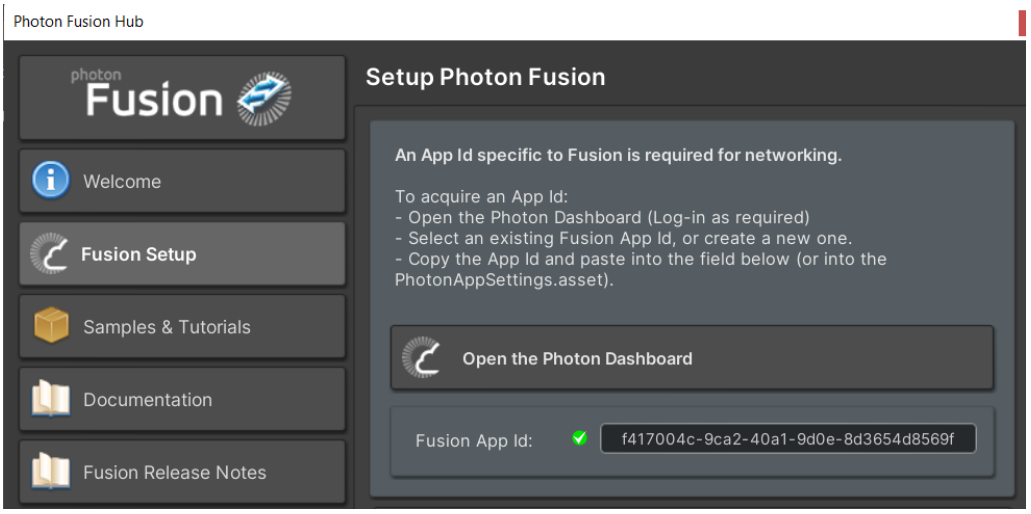
- Search for your created App in the dashboard, and copy the App ID



- Also, create a **Voice App**

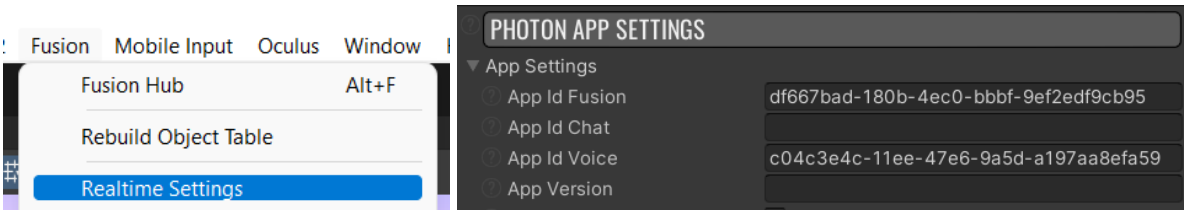


-
- Go back to Unity and open the Fusion Hub (Fusion>Fusion hub, or alt+f)

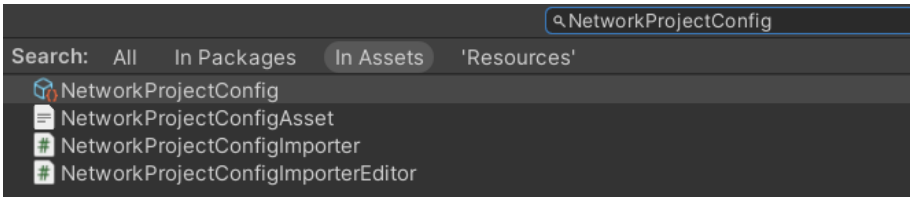


- Paste the AppID in the Fusion App id.

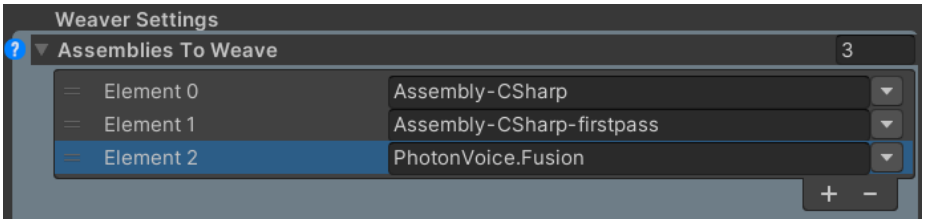
Now, you will need to go to Fusion>Realtime settings, and also add the App and Voice IDs there:



Now search for “NetworkProjectConfig” and select it:



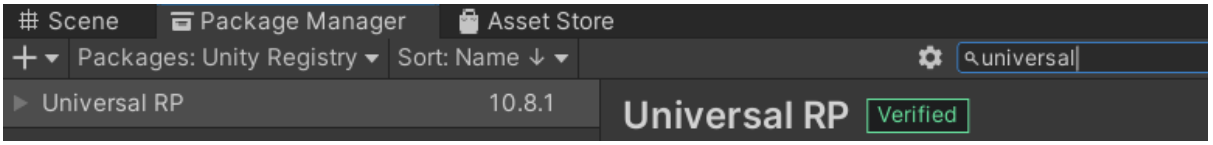
Go to Config>Assembles to Weave and add “PhotonVoice.Fusion”.



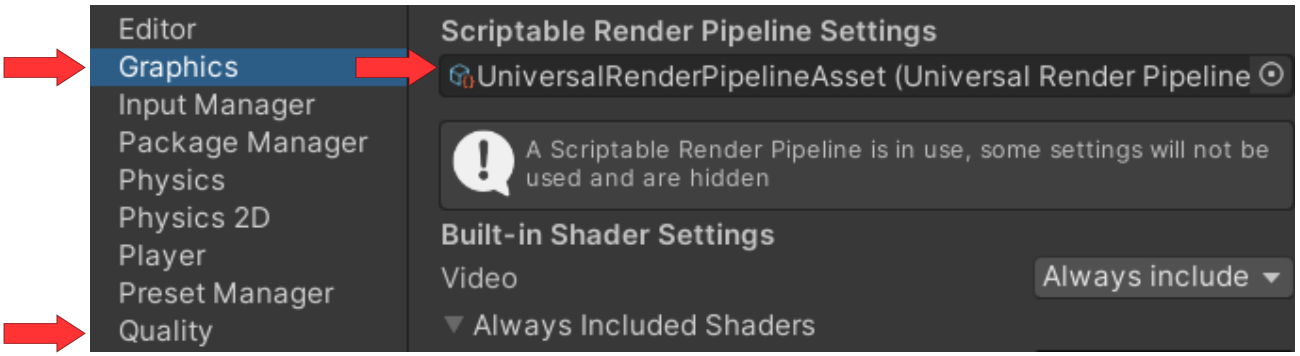
Hit “Apply”.

OPTIONAL: if you see everything pink like in this screenshot,

you need to install the URP (Universal render pipeline) package from the package manager (you find it under the Unity Registry Tab).

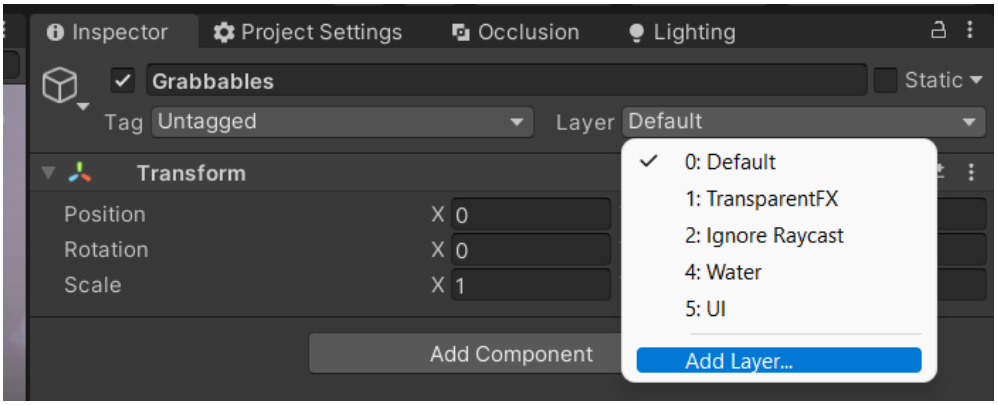


After importing the URP, go to Project Settings > Quality and Project Settings > Graphics and assign the UniversalRenderPipelineAsset into the fields.



Then, the scene should be rendered appropriately.

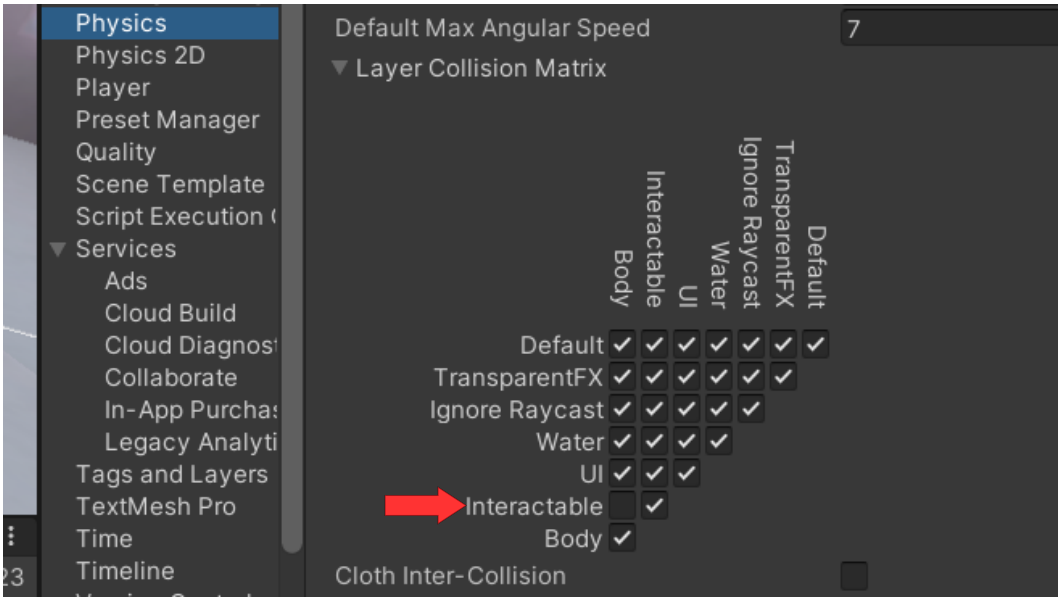
8. **Setting up layers:** the template needs a custom layer setup to avoid the body capsule colliding with interactables. To set up this, select any object in the scene and go to layer>add layer:



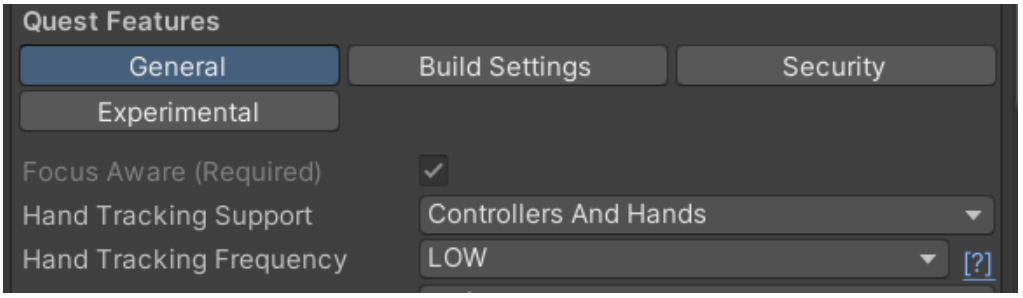
Then, click on the “presets” button in the top right:

Select the preset included in the template.

Now, go to Project Settings>Physics, and make sure that the layers “Body” and “Interactable” doesn’t collide between them, unclicking it:

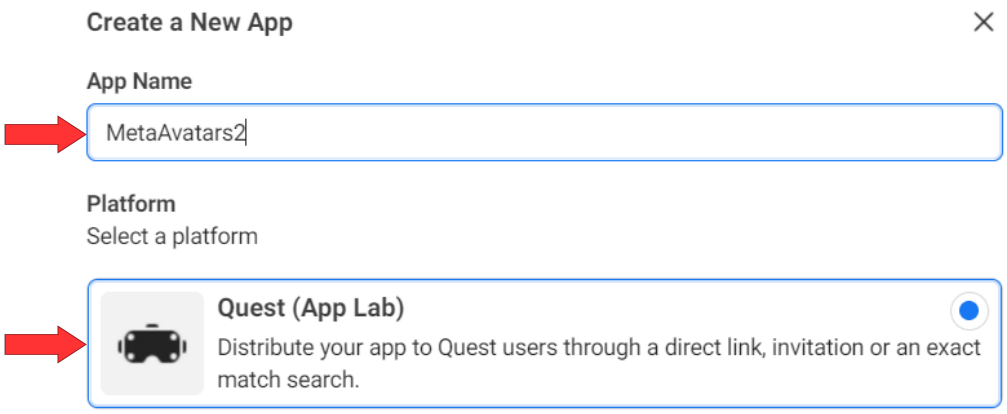


9. To enable **Hand Tracking,** select the OVRNetworkCameraRig from the scene, and in the OVRMangaer component, select Controllers and Hands in the Hand Tracking Support dropdown:

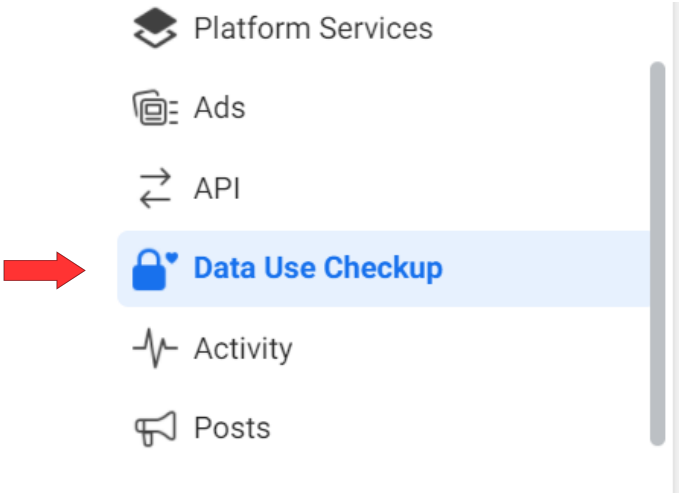


10. **Creating an Oculus App and enabling avatars.**
    In order to use the avatars, you need to enable them in the Oculus developer portal. Please go to https://developer.oculus.com/manage/ and create a new app:

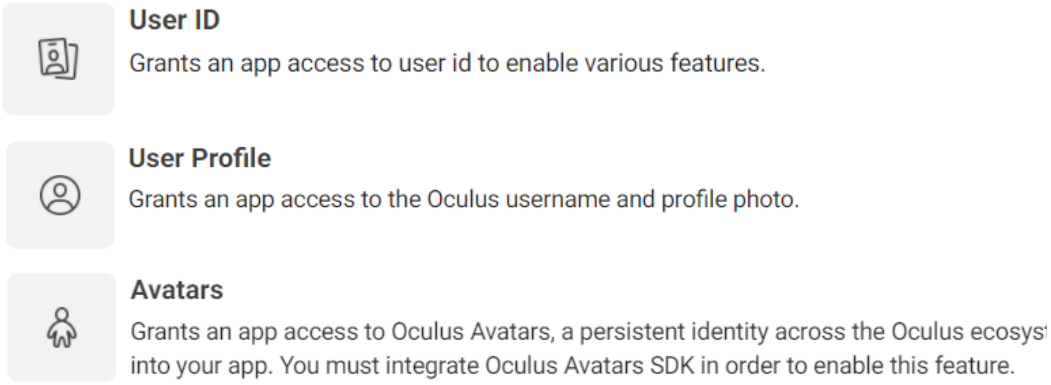
- Choose a name and select the quest platform



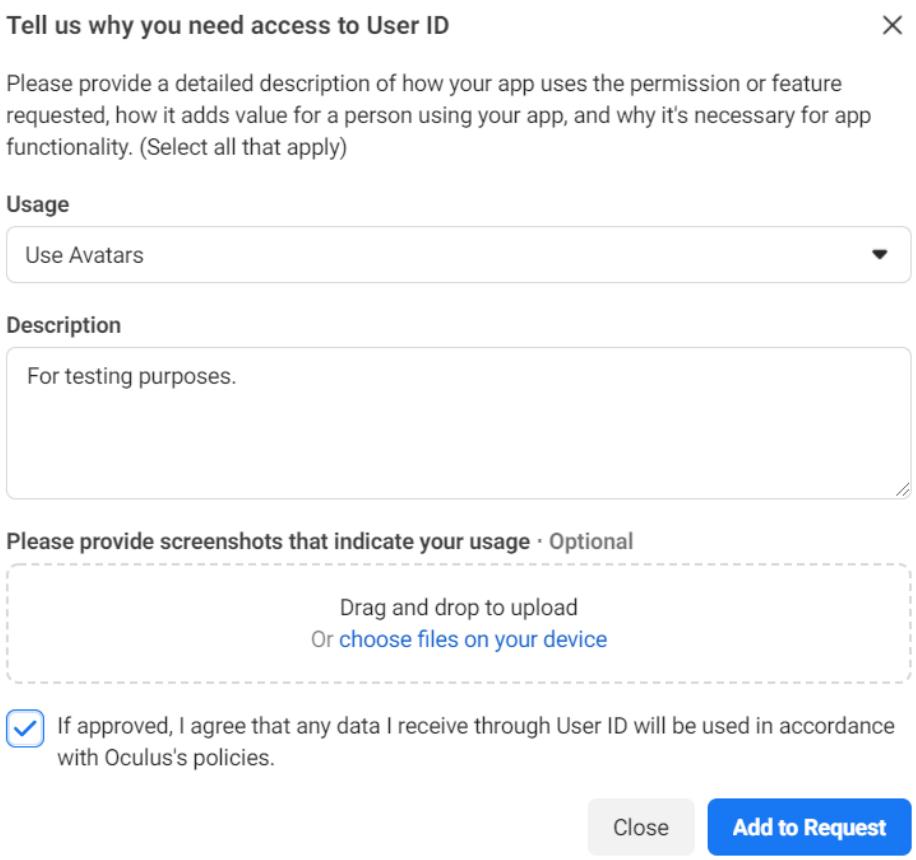
- Go to Data Use Checkup in the left menu.



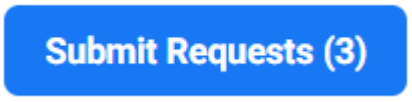
- Request access to the following features: UserID, User Profile and Avatars.



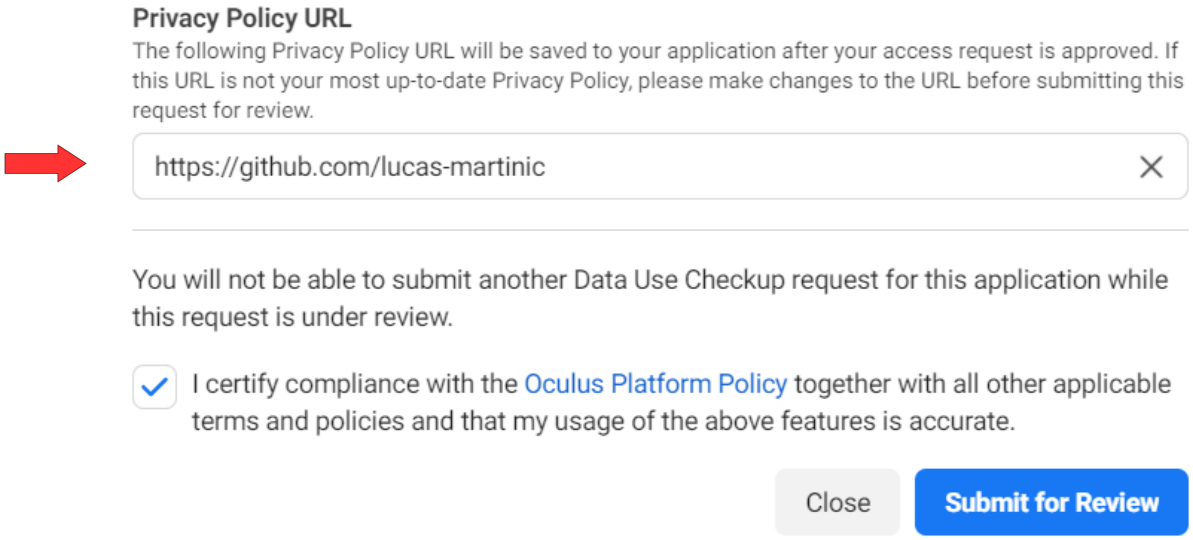
- Add the 3 requests like in this example, you can describe your personal purpose/use case.



- After adding the 3 requests, click on “Submit Requests (3)”

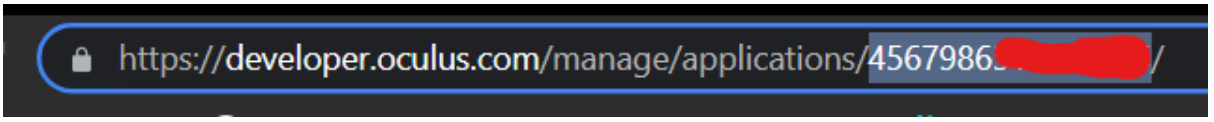


- You will be asked to provide a privacy policy, but as we will not yet be sending the app to the Oculus Store, you can provide a placeholder link, for example, your GitHub account.

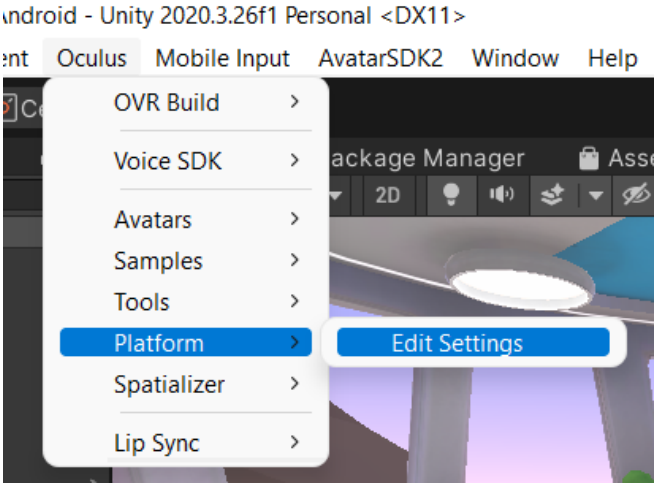


These features will be approved right away. Remember we need to complete this step because the Meta Avatars require an Oculus ID to work, as they use the user's avatar that is linked to their account.

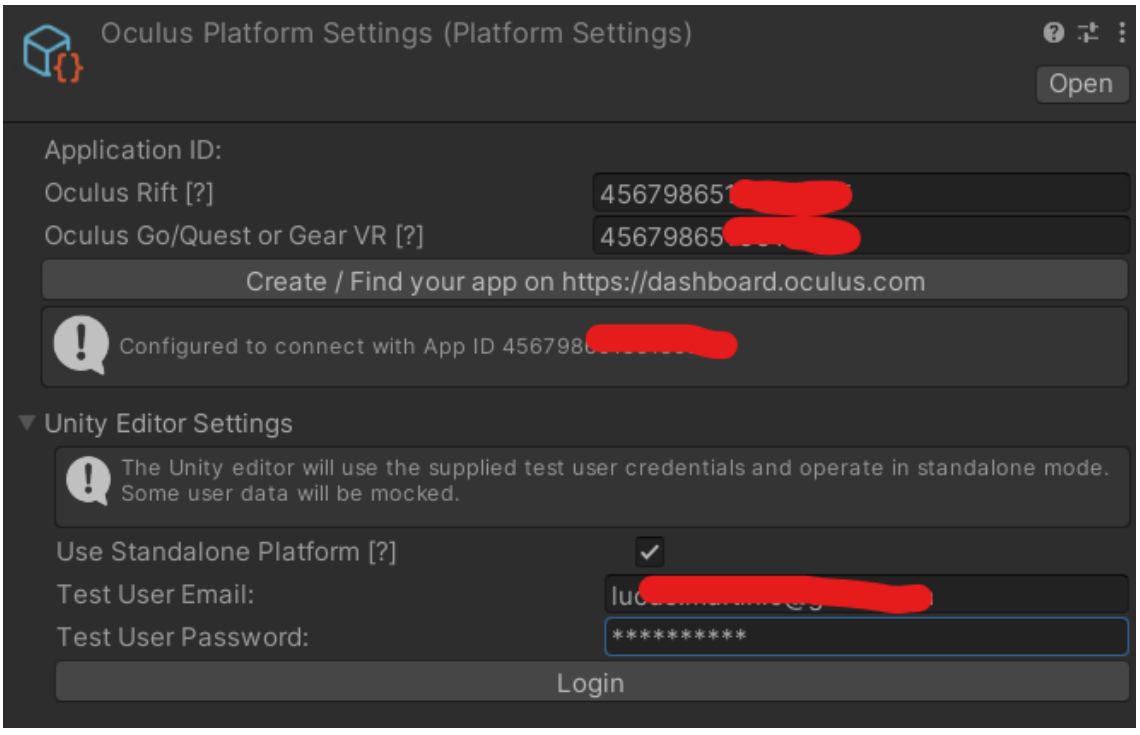
Now, copy your **Oculus App ID** from the top of your browser:



- Go back to Unity, and paste it into Oculus > Platform > Edit Settings in both the Oculus Rift and Oculus Quest fields.

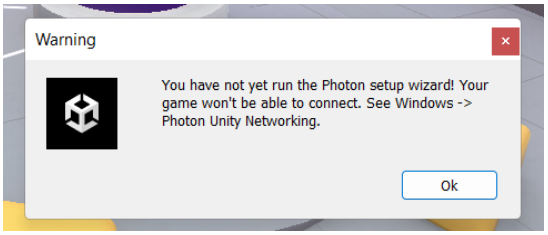


Also, select "Use Standalone Platform" and enter your Oculus/Meta credentials.



Now, you should be able to click on “Play” and the app should load your Avatar linked to your Oculus account!

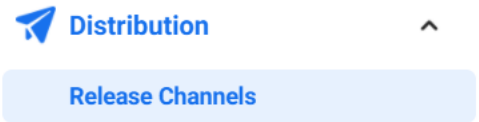
(PD: You might get this warning because Photon Voice 2 is still looking for Pun2, but you can just ignore it by clicking “OK”).



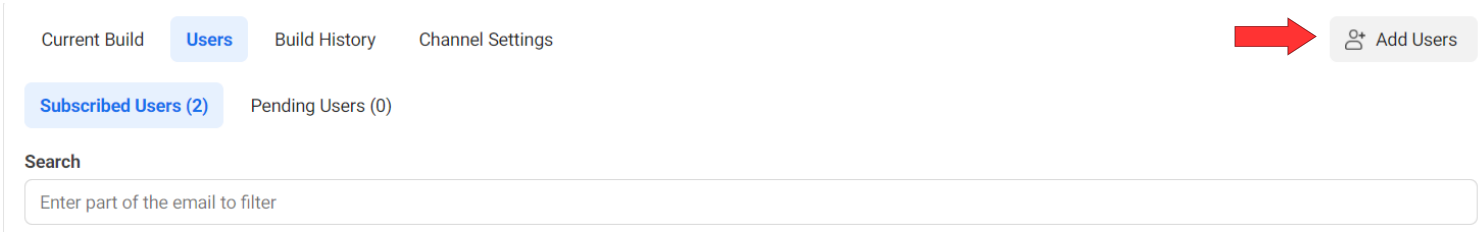
> Where’s the head?!? Don’t worry, this is a first-person avatar (the one you see for yourself inside VR), so this is normal. Other avatars from other users will look normal, with a head :)

IMPORTANT

For an external player to join your app/game with their own Avatars, they must pass the “entitlement check” (normally, to own the game). You can go to the Oculus Developer Portal, and upload it to an Alpha/Beta channel. **You must follow their instructions to upload your .apk into the release channel.**

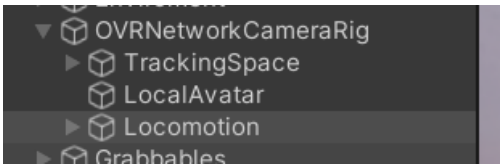


After that, you will be able to invite other accounts with their emails. Then they will be able to join and correctly load their avatars.

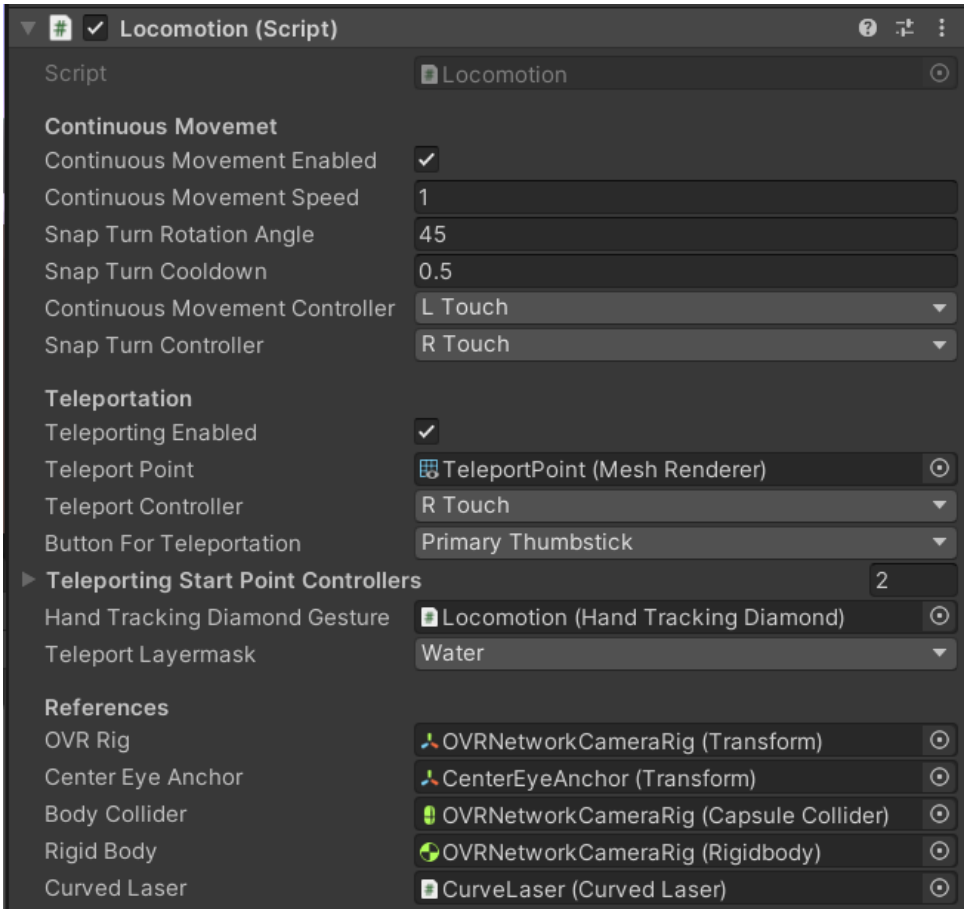


Locomotion System:

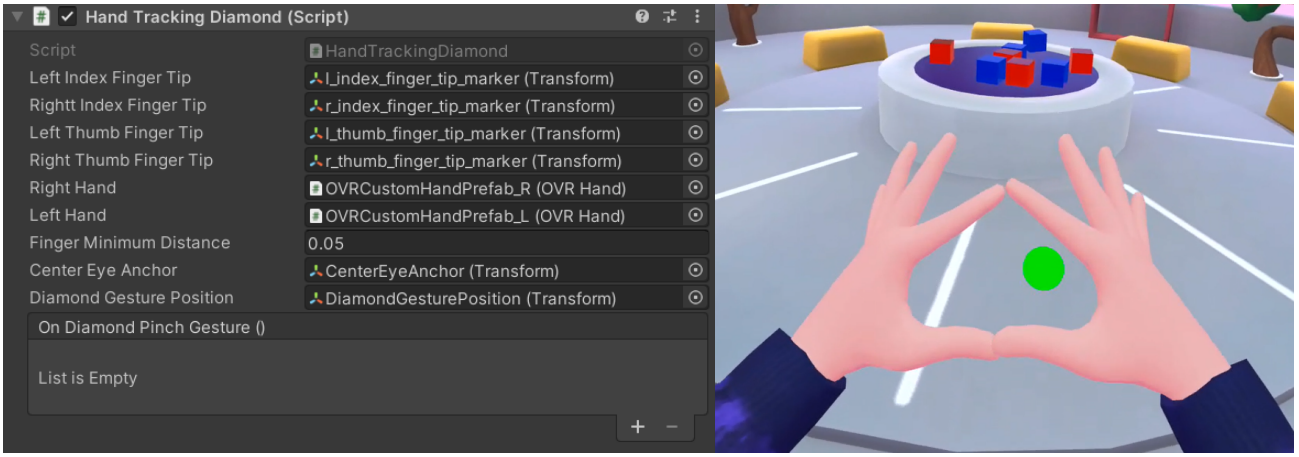
Since version 0.5, there's a custom Locomotion system included in the package, that will correctly work in a network.



Plenty of customization can be achieved through the Locomotion component:



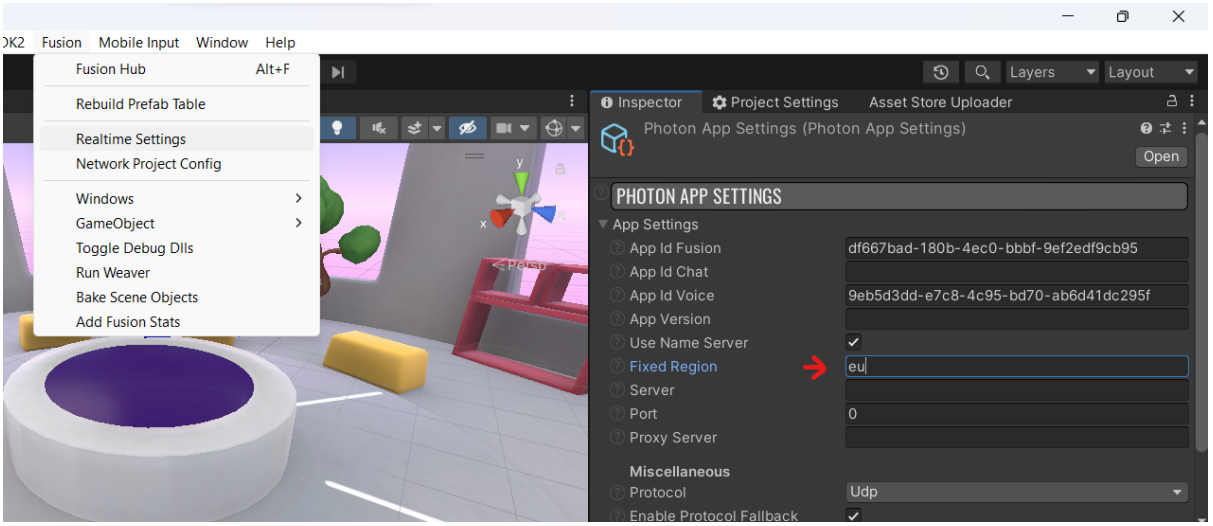
There is also a **HandTrackingDiamond** component which detects a gesture inspired by https://youtu.be/OkxyWVT0hoY?t=109 which is used as default for hand-tracking teleportation. You can subscribe to the pinch event for custom behaviors.



Common Issues

1. “When I tested with my colleague, I was able to see my avatar and he is able to see his avatar, but we were not able to see each other's avatars.”

Please try fixing the Photon Region by going to **Fusion>Realtime settings**



More info on Photon Regions here.