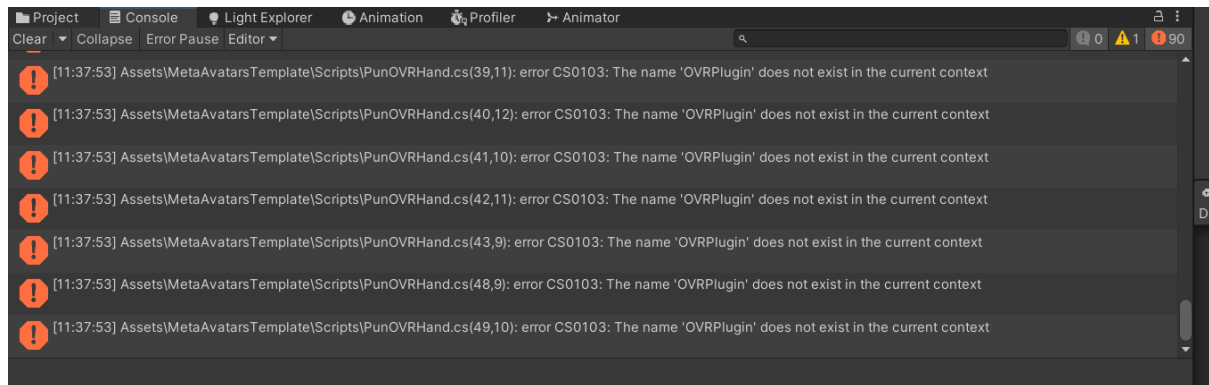


Multiplayer Mixed Reality Template (MMRT) setup guide.

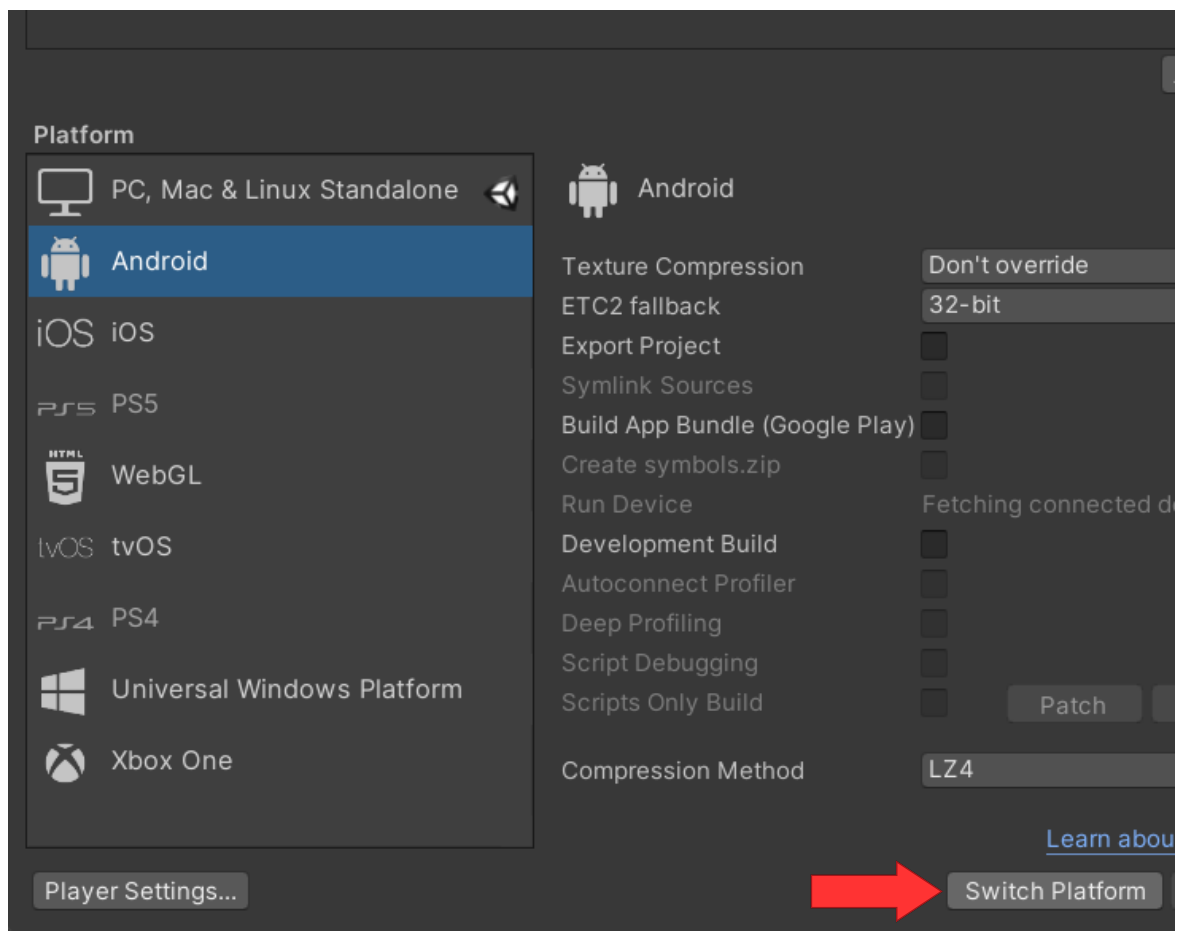
Thanks for purchasing! Join us on the discord server: <https://discord.gg/gEhHu8Xydr>

After importing the package to a fresh Unity 2020.3.X or 2021.3.X project, you will see many errors. Don't panic! That's because we still need to import other packages for it to work.



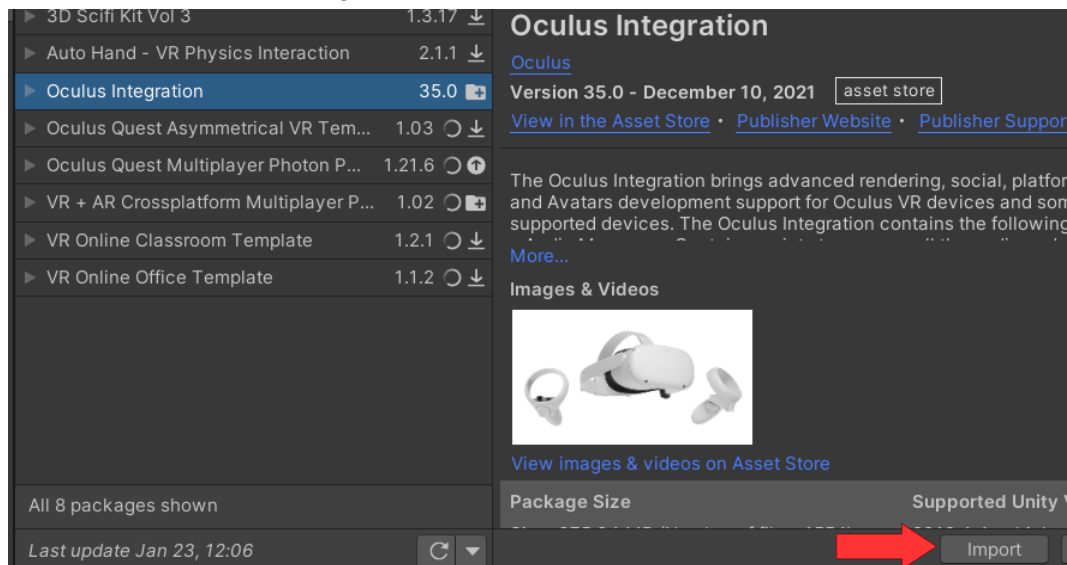
Let's start setting up the project!

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



2. Now, let's import some packages:

- Latest Oculus Integration:



- Photon Pun 2 - FREE



- Photon Voice 2



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

Meta Avatars SDK

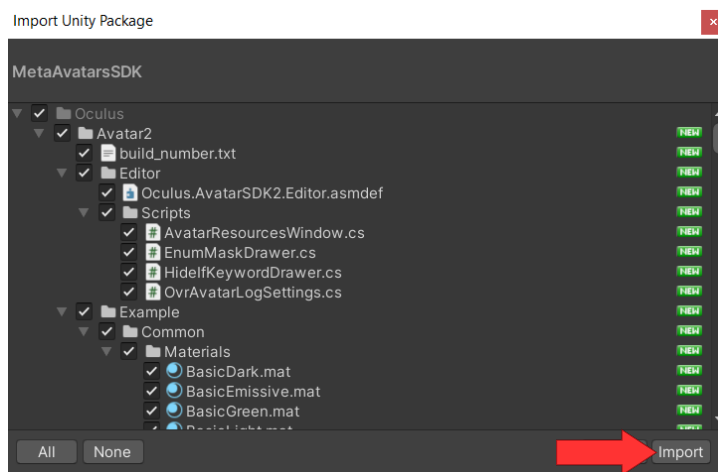
Published: Jan 11, 2022

ODH Unity

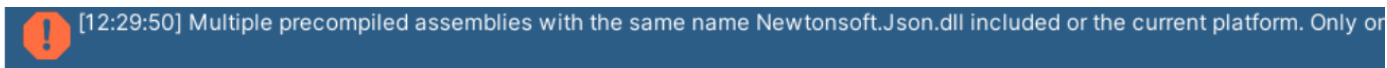
VERSION
9.1

DOWNLOAD

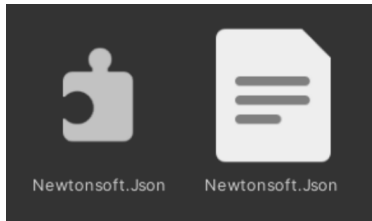
- Import the MetaAvatarSDK into your Unity Project.



- If you get this error:

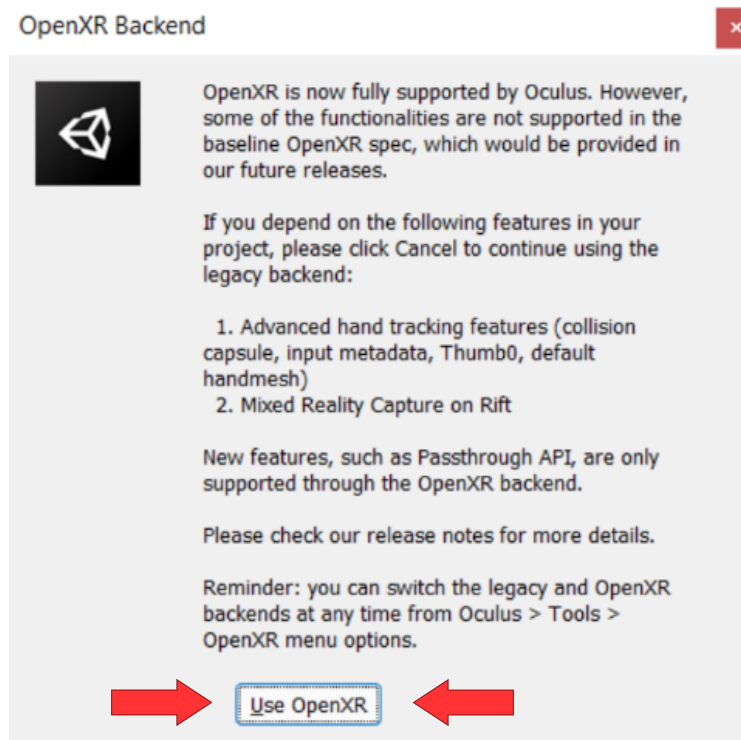


Go to Assets > Oculus > Avatar2 > Scripts > AvatarEditorDeepLink, and delete these 2 files (they are duplicated).



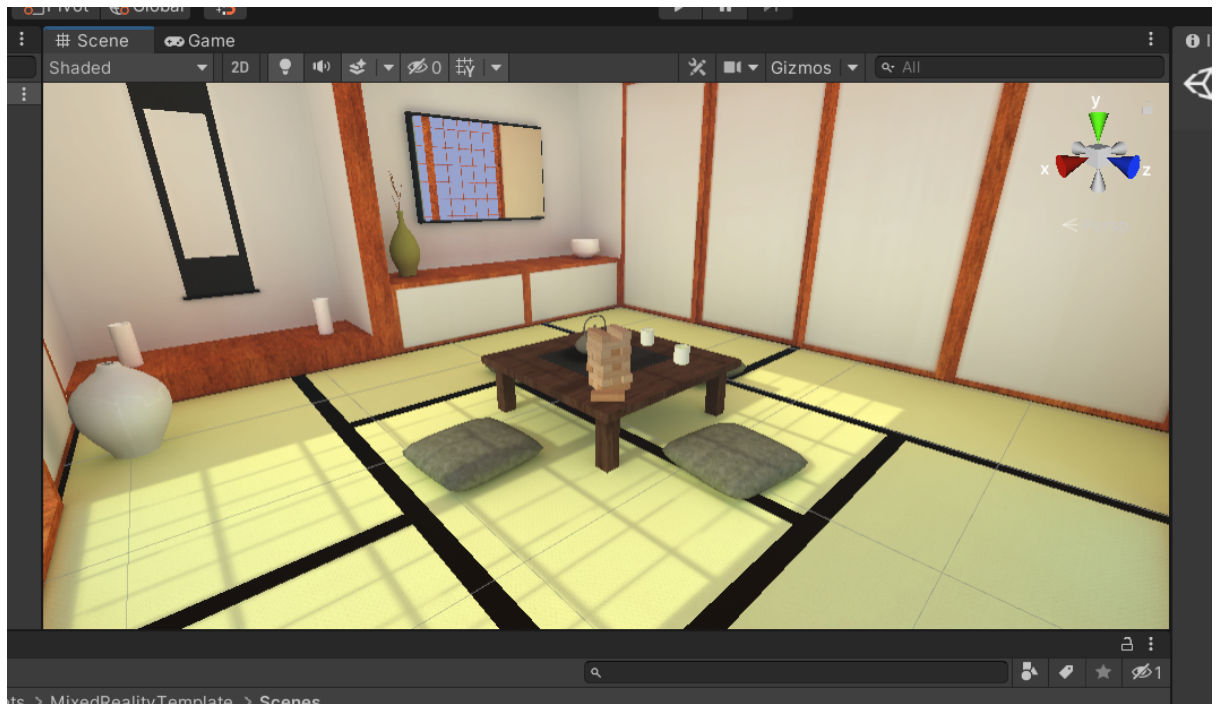
- Now, the project should compile, and start asking for some updates. Accept all the updates requested.

3. **Enable Open XR** This is mandatory for Passthrough to work.



- Unity will want to restart afterward, click yes.

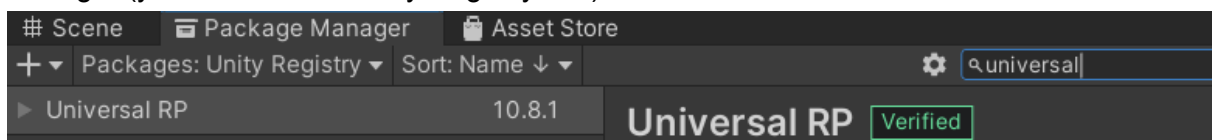
4. You can now open the main scene in
Assets>MultiplayerMixedRealityTemplate>Scenes



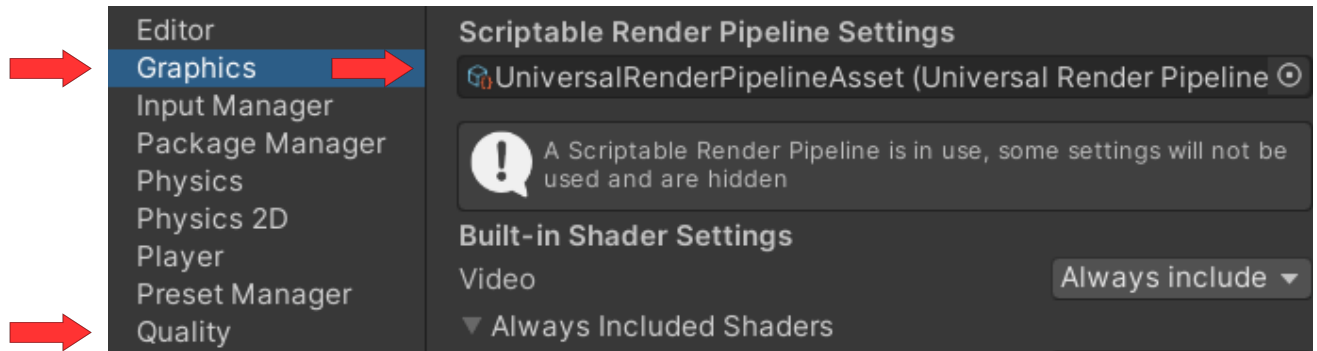
OPTIONAL: if you see everything pink like in this screenshot,



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

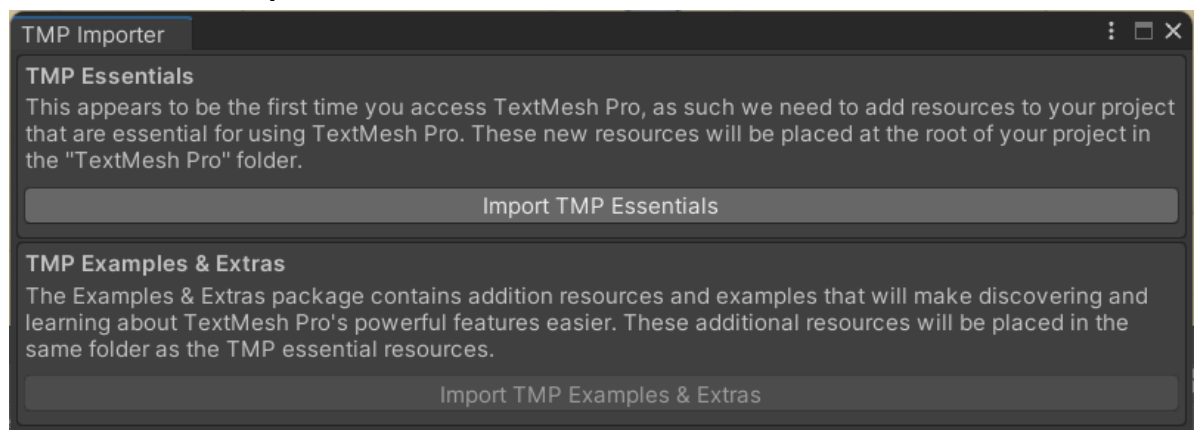


After importing the URP, go to Project Settings > Quality and Project Settings > Graphics and assign the UniversalRenderPipelineAsset into the fields.



Then, the scene should render appropriately.

5. You will need to import Text Mesh Pro:

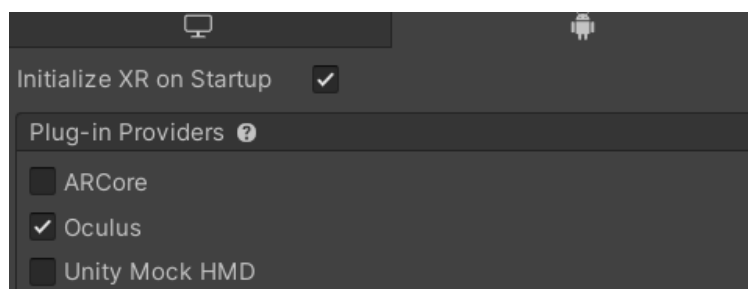


6. Install XR Plugin Management

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

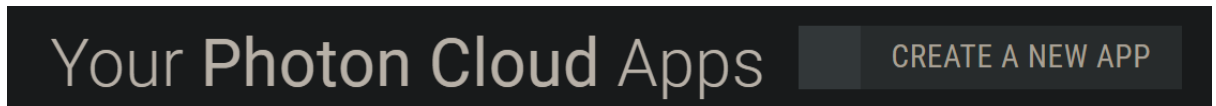


7. Add the Oculus platform for both windows and android.

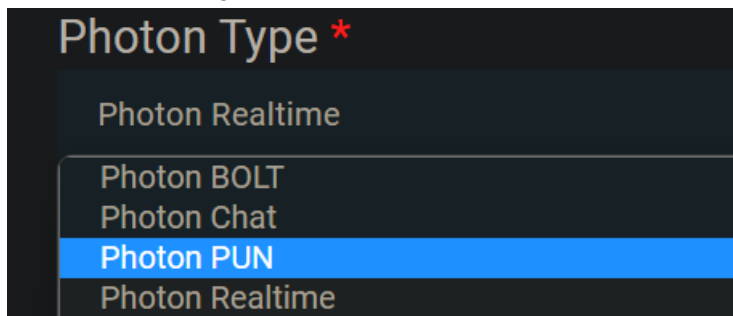


8. Configure Photon Pun2

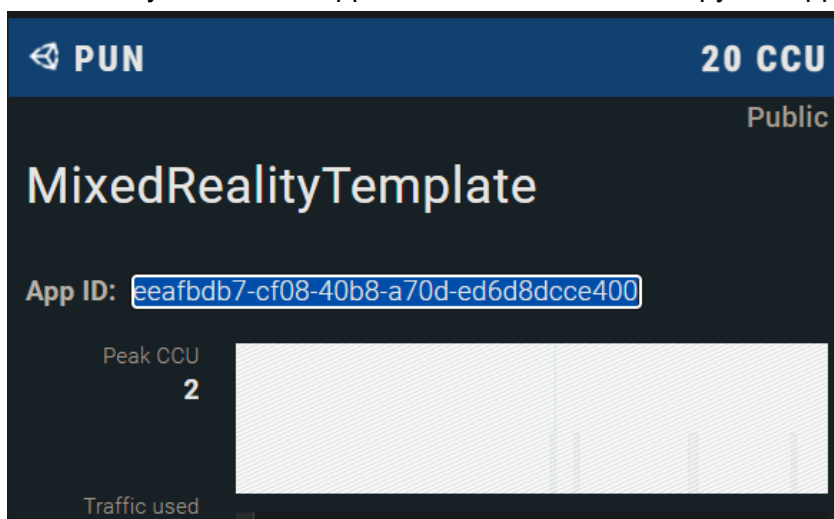
- Go to <https://dashboard.photonengine.com/en-US/> and create an account if you don't have one.
- Create a new Pun App (you can create as many apps as you like)



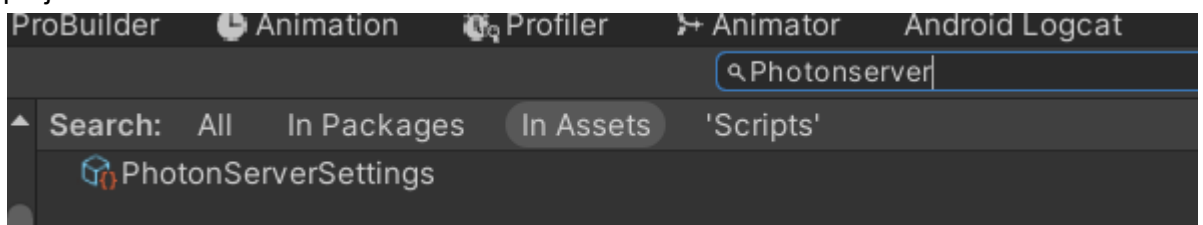
- **Make sure it's Type PUN**



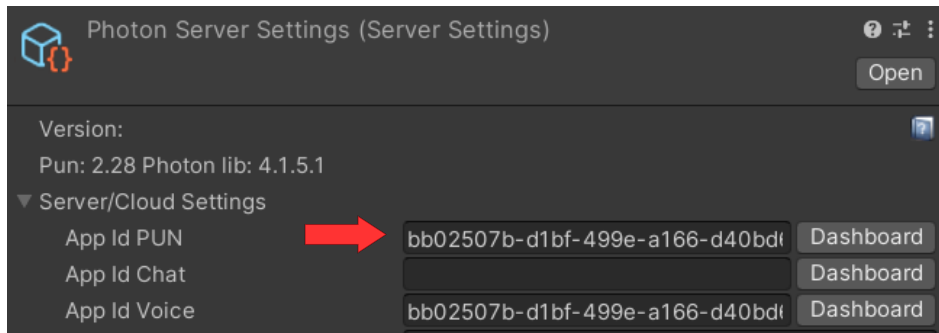
- Search for your created App in the dashboard, and copy the App ID



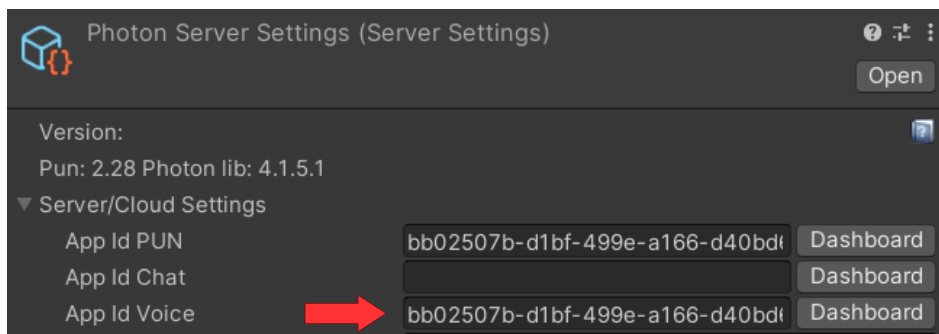
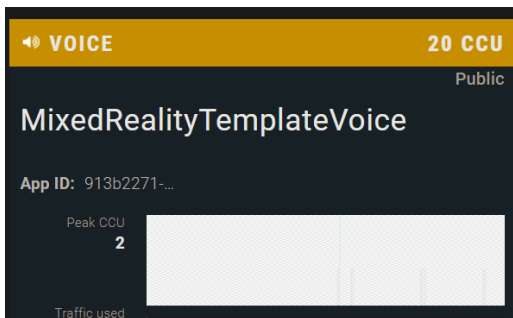
- Go back to Unity and look for the PhotonServerSettings configuration file in the project window.



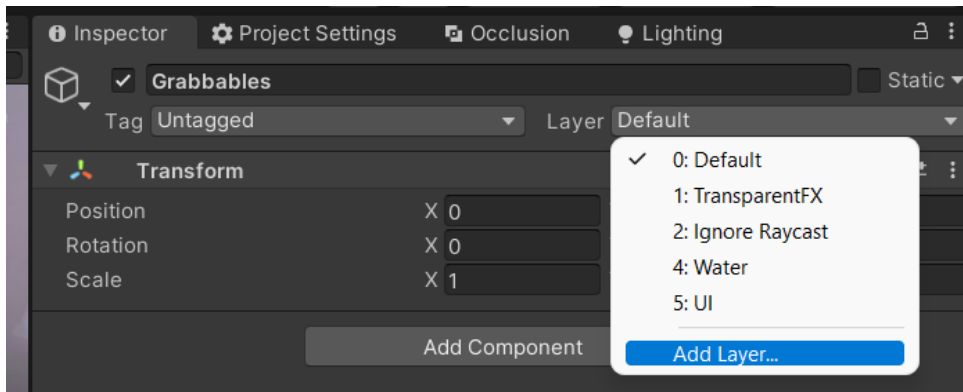
- In the PhotonServerSettings configuration, paste the AppID.



- Go back to the Photon Dashboard and also create a **Voice** app and paste the ID on the App id Voice field in Unity:



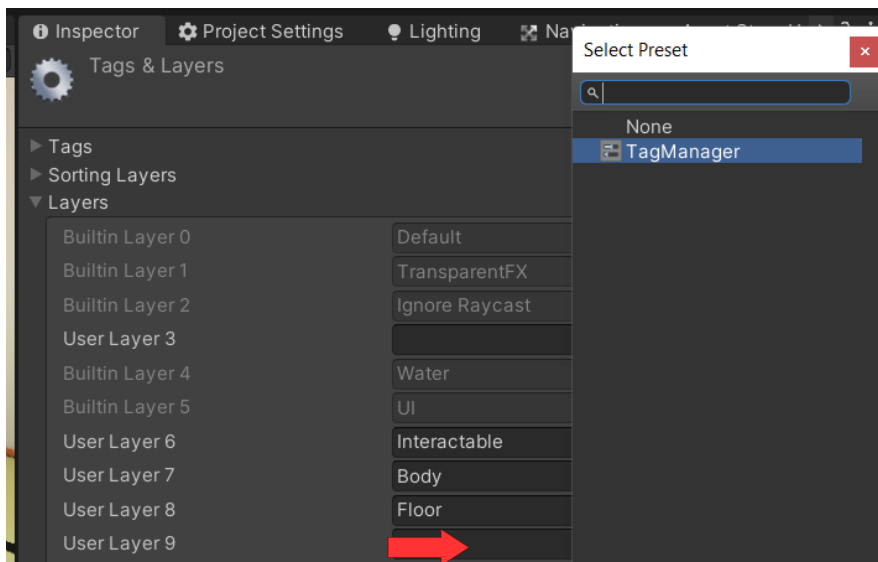
9. **Setting up layers:** the template needs a custom layer setup to avoid unwanted collisions. To set up this, select any object in the scene and go to Layer>Add Layer:



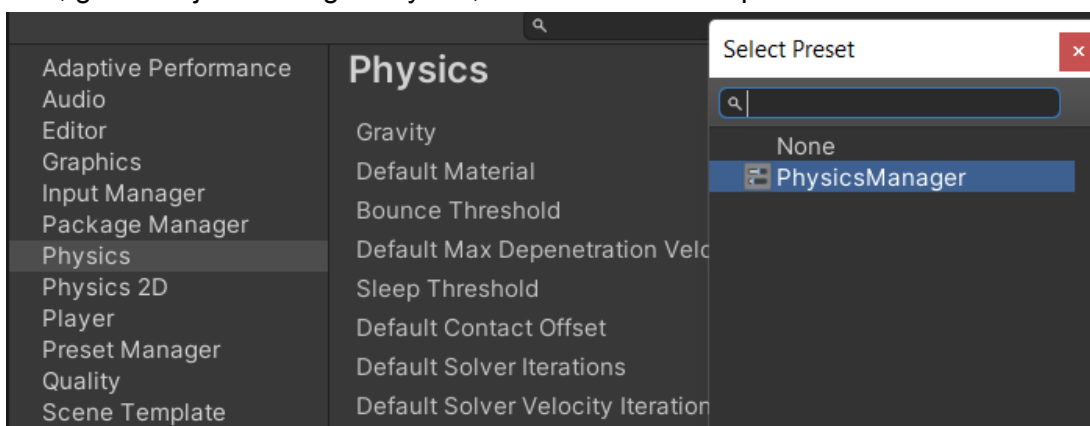
Then, click on the “presets” button in the top right:



And select the preset included in the template.



Now, go to Project Settings>Physics, and also select the preset included with the asset:



10. Enabling hand tracking and passthrough

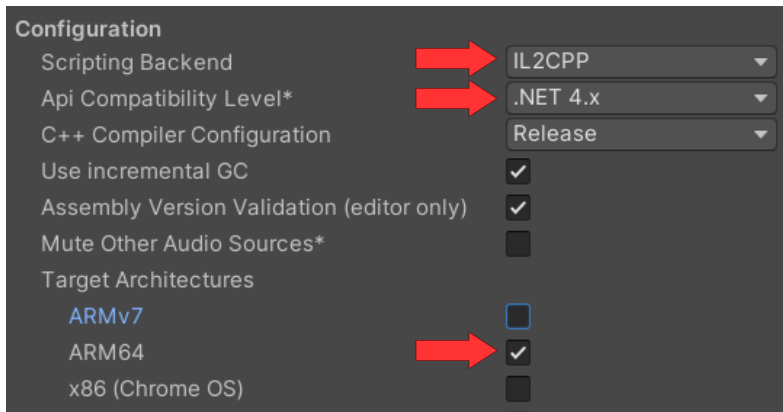
- Go to the OVRNetworkCameraRig, and to the Quest Features section, and enable hand tracking and passthrough:

The image shows the 'Quest Features' settings panel. It has three tabs: 'General', 'Build Settings', and 'Security'. The 'Experimental' sub-tab is selected. The settings are as follows:

Setting	Value
Focus Aware (Required)	<input checked="" type="checkbox"/>
Hand Tracking Support	Hands Only
Hand Tracking Frequency	HIGH
Hand Tracking Version	V2
Render Model Support	Disabled
Requires System Keyboard	<input type="checkbox"/>
Tracked Keyboard Support	None
Anchor Support	Disabled
Body Tracking Support	None
Face Tracking Support	None
Eye Tracking Support	None
System Splash Screen	None (Texture 2D)
Allow Optional 3DoF Head Tracking	<input type="checkbox"/>
Passthrough Capability Enable	<input checked="" type="checkbox"/>
Mixed Reality Capture for Quest	Disabled
Mixed Reality Capture	[?]
Insight Passthrough	[?]
Enable Passthrough	<input checked="" type="checkbox"/>

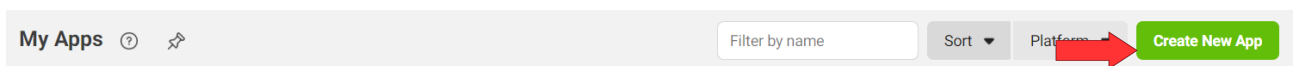
Red arrows point to the following settings: 'Hands Only', 'HIGH', 'V2', 'Passthrough Capability Enable', and 'Enable Passthrough'.

- **11. Go to Project Settings>Player Settings**, switch the scripting backend to ILC2PP, and set the project to 64bit

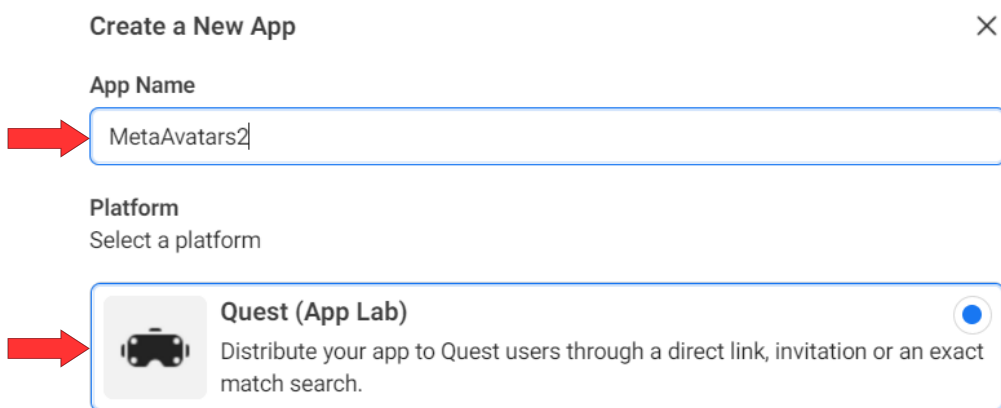


12. Creating an Oculus App and enabling avatars.

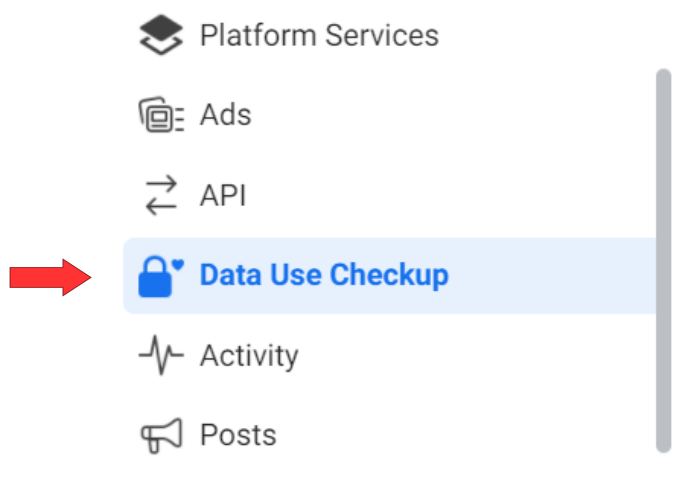
In order to use the avatars, you need to enable them in the oculus developer portal. Please go to <https://developer.oculus.com/manage/> and create a new app:




- Choose a name and select quest platform





- Go to Data Use Checkup in the left menu.



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


User ID
 Grants an app access to user id to enable various features.


User Profile
 Grants an app access to the Oculus username and profile photo.


Avatars
 Grants an app access to Oculus Avatars, a persistent identity across the Oculus ecosystem into your app. You must integrate Oculus Avatars SDK in order to enable this feature.

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

Tell us why you need access to User ID

×

Please provide a detailed description of how your app uses the permission or feature requested, how it adds value for a person using your app, and why it's necessary for app functionality. (Select all that apply)

Usage

Use Avatars

Description

For testing purposes.

Please provide screenshots that indicate your usage · Optional

Drag and drop to upload
 Or [choose files on your device](#)

☒ If approved, I agree that any data I receive through User ID will be used in accordance with Oculus's policies.

Close

Add to Request

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

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.

Privacy Policy URL

The following Privacy Policy URL will be saved to your application after your access request is approved. If this URL is not your most up-to-date Privacy Policy, please make changes to the URL before submitting this request for review.



<https://github.com/lucas-martinic>



You will not be able to submit another Data Use Checkup request for this application while this request is under review.



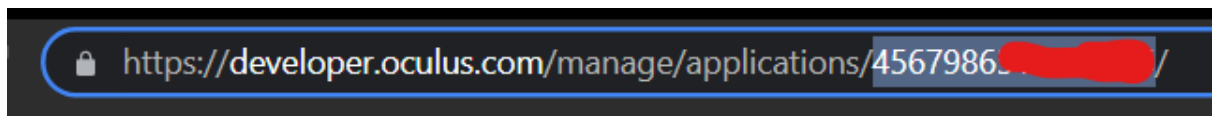
I certify compliance with the [Oculus Platform Policy](#) together with all other applicable terms and policies and that my usage of the above features is accurate.

Close

Submit for Review

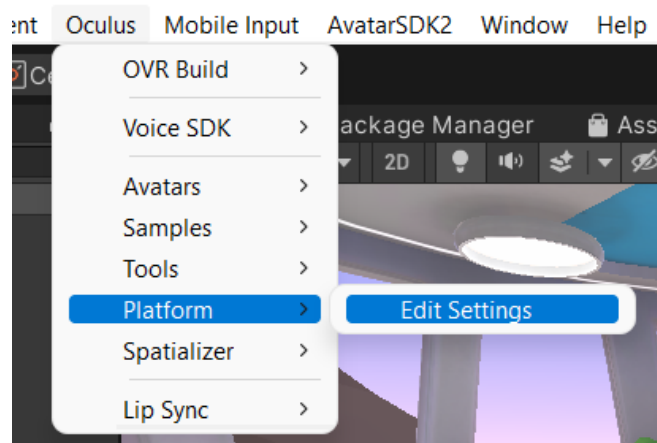
These features will be approved right away. Remember we need to complete this step because the Meta Avatars require an Oculus ID to work, as they use the user's personal avatar that is linked to their account.

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

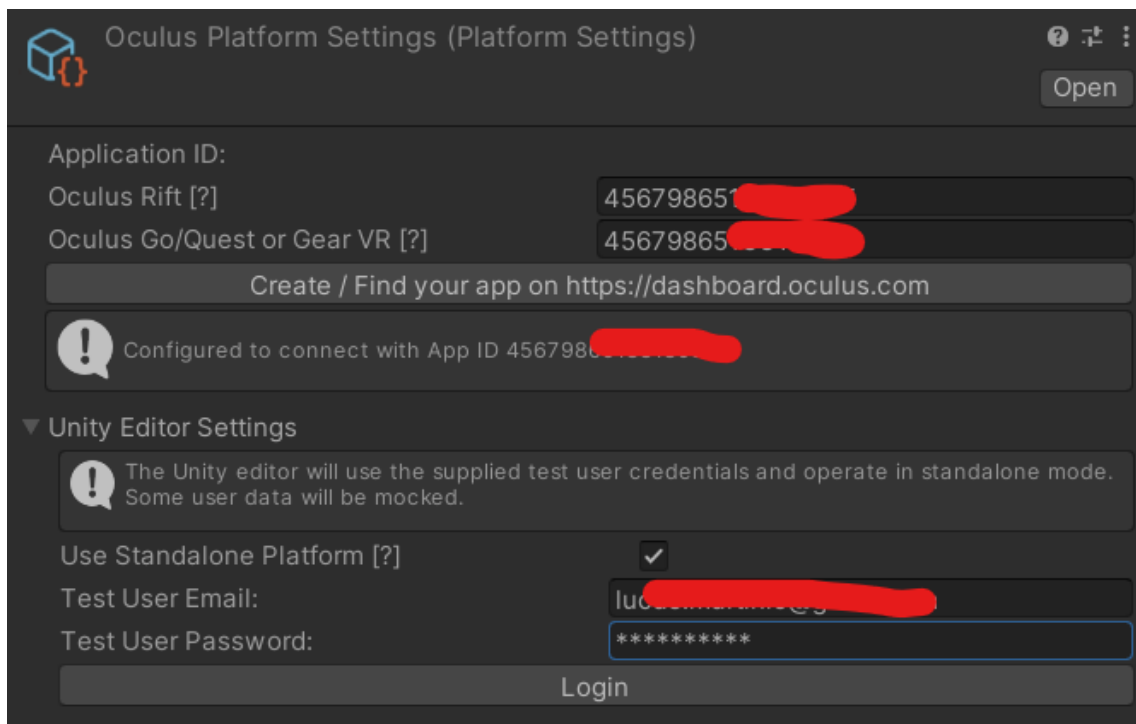


- And go back to Unity, and paste it into Oculus > Platform > Edit Settings in both Oculus Rift and Oculus Quest fields.

android - Unity 2020.3.26f1 Personal <DX11>



Also, select “Use Standalone Platform” and enter your oculus/meta credentials.



Oculus Platform Settings (Platform Settings) Open

Application ID:
Oculus Rift [?] 45679865
Oculus Go/Quest or Gear VR [?] 45679865

Create / Find your app on <https://dashboard.oculus.com>

Configured to connect with App ID 45679865

▼ Unity Editor Settings

The Unity editor will use the supplied test user credentials and operate in standalone mode. Some user data will be mocked.

Use Standalone Platform [?] ☒

Test User Email: lucas@lucas.com

Test User Password: *****

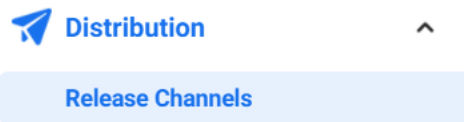
Login

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

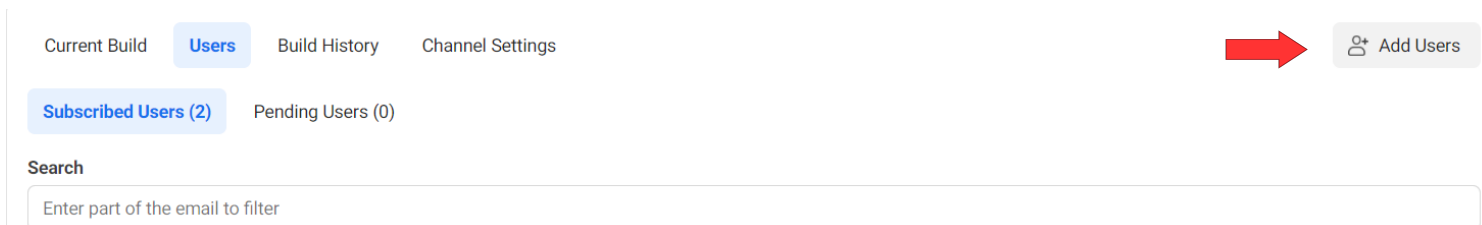


IMPORTANT

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 need to follow their instructions to upload your apk into the release channel.**



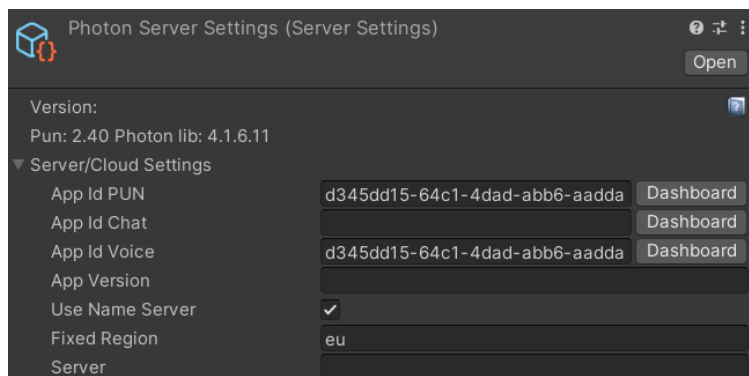
After that, you will be able to invite other accounts with their emails. Then they will be able to join and correctly load their avatars.



Common Issues

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

Please try fixing the Photon Region in the PhotonServer settings file, you can try with something like eu or us to force both clients to connect to the same Photon Server.



More info on Photon Regions [here](#).