VR Office Template
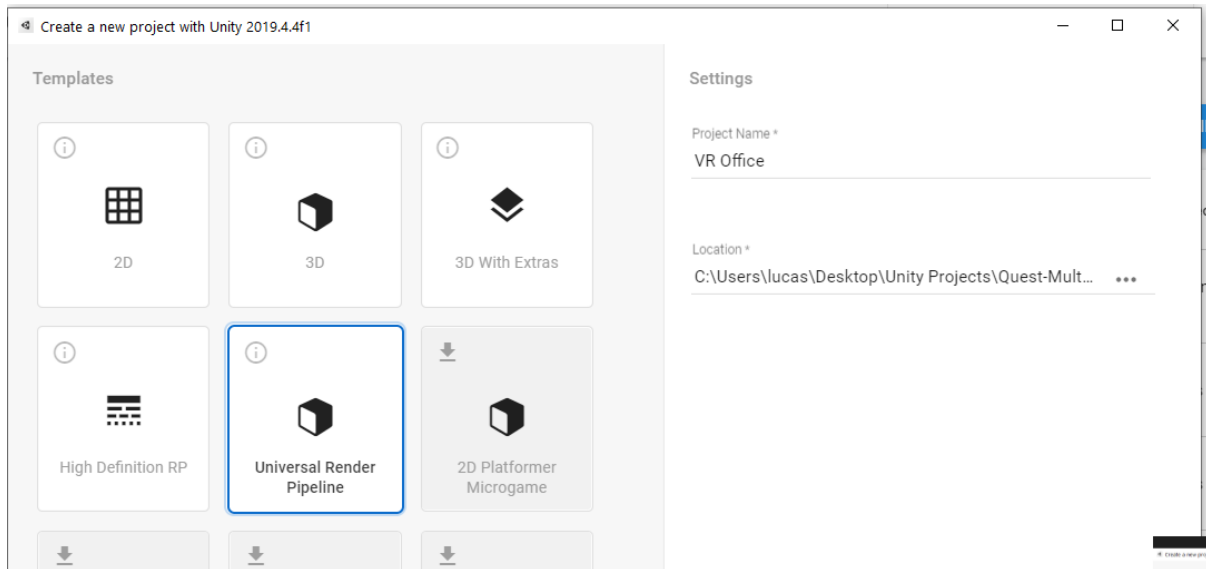
⇒ This document will help you with the setup of the project. The project is focused on building for Oculus Quest 1 & 2, but it should also work for other VR devices, as this project uses XR Interaction Toolkit.

⇒ I am using Unity 2020.3.(LTS) and the Universal Render Pipeline (URP) template, but it should also work for Unity 2020 with URP.
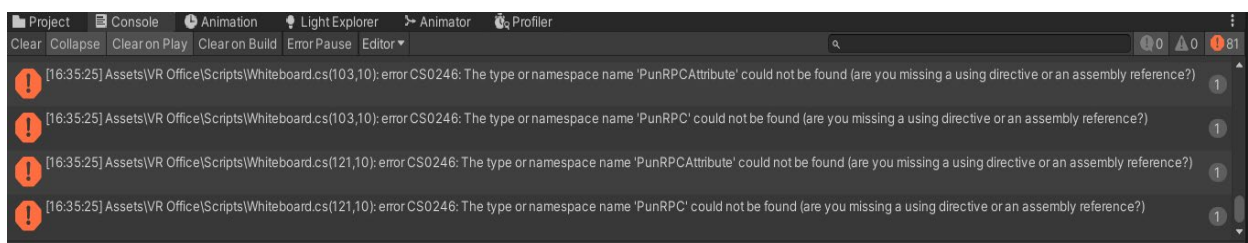

A) Creating a new project

Creating a new project using URP is important for performance and materials.

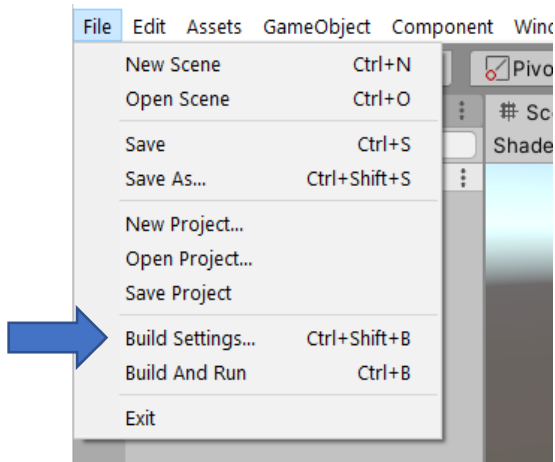After creating the project, you can import **VR Office Template** asset.
You will have lots of errors in the project at this point, but it is totally normal, as we still need to import some more stuff.
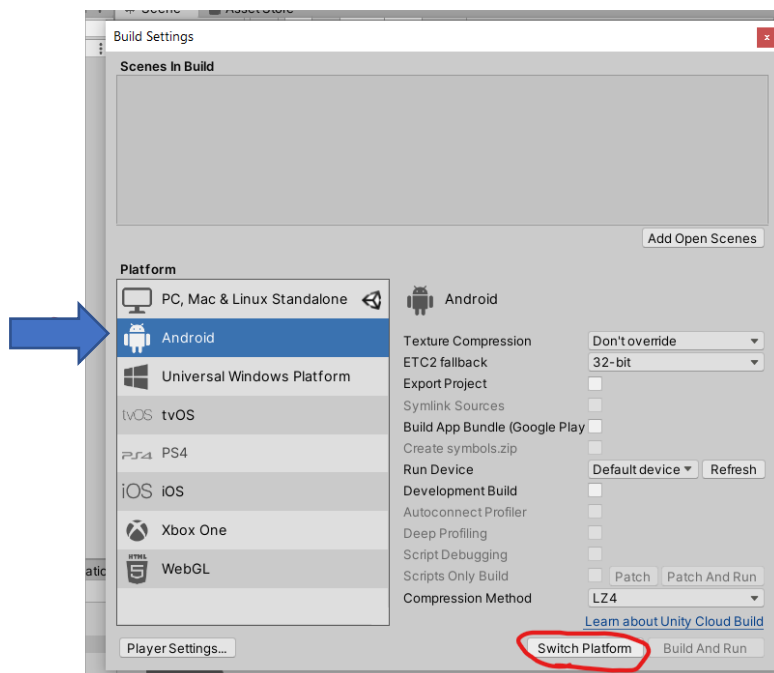

LOTS OF ERRORS! Don't panic.
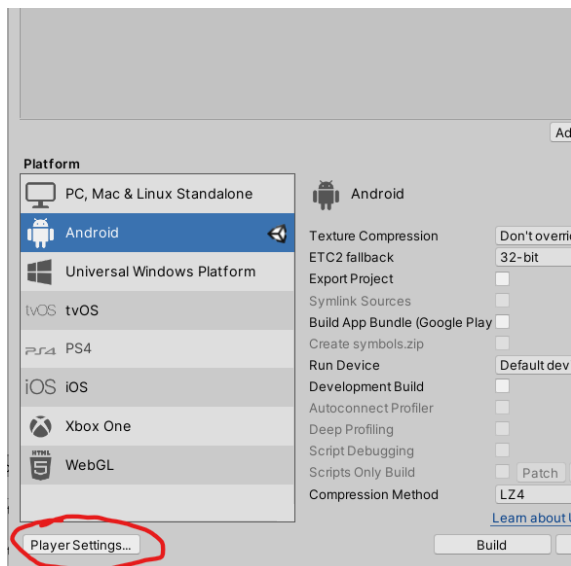
B) Setup for Quest: Let's begin!

1. Change the build target platform, going to **File > Build Settings**.
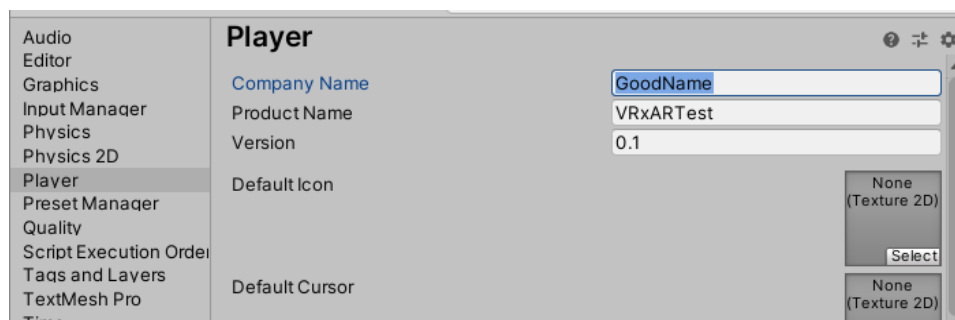

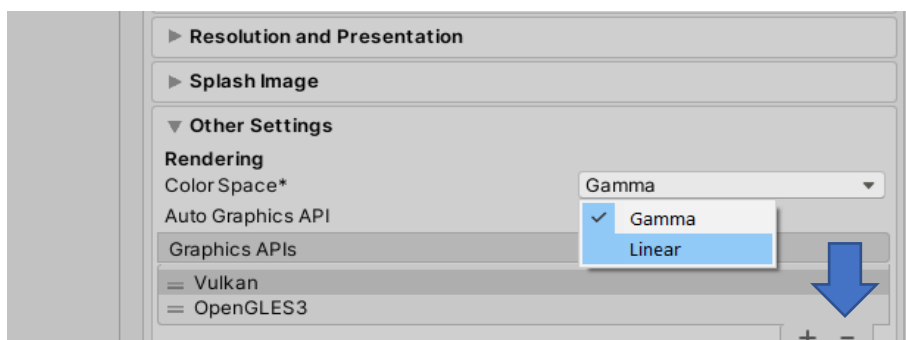2. Select **Android** and click **Switch Platform.**
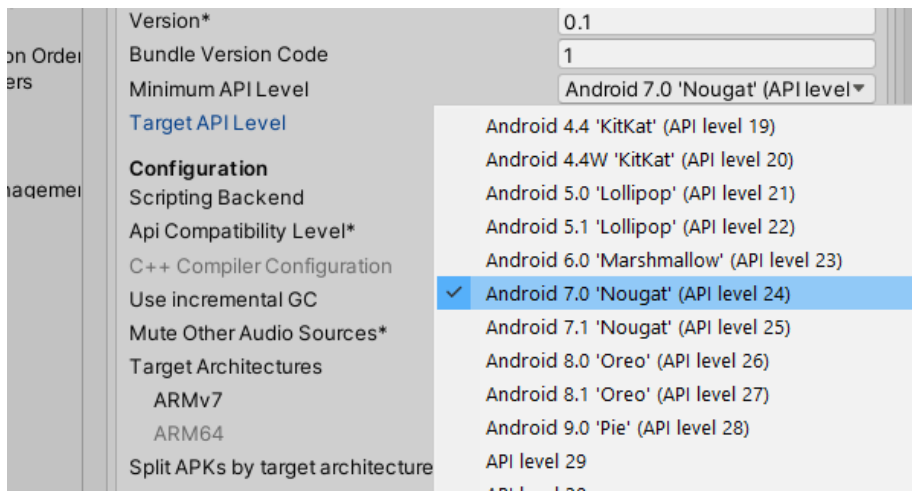
3. Open **Player Settings.**

4. Fill your great **Company Name.**

5. Go to "**Other Settings**" and select **Linear** space. Also select **Vulkan** and take it out of the list with the **"–"** sign, so only **"OpenGLES3"** is left.
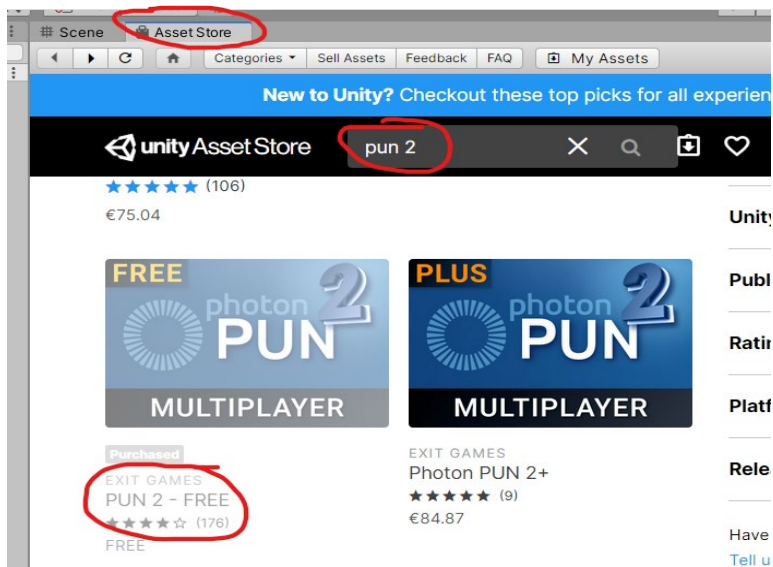
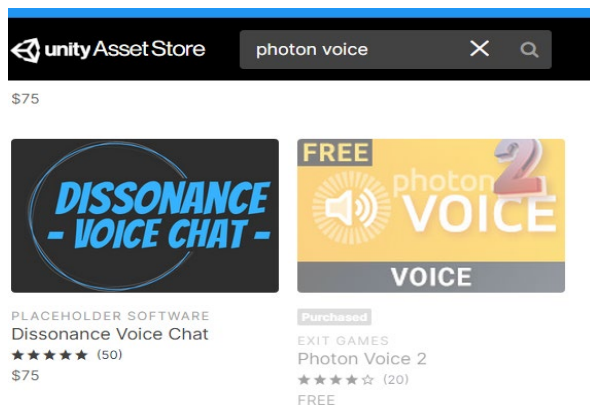6. Finally, in the **Minimum API Level**, chose **7.0/ lv 24.**

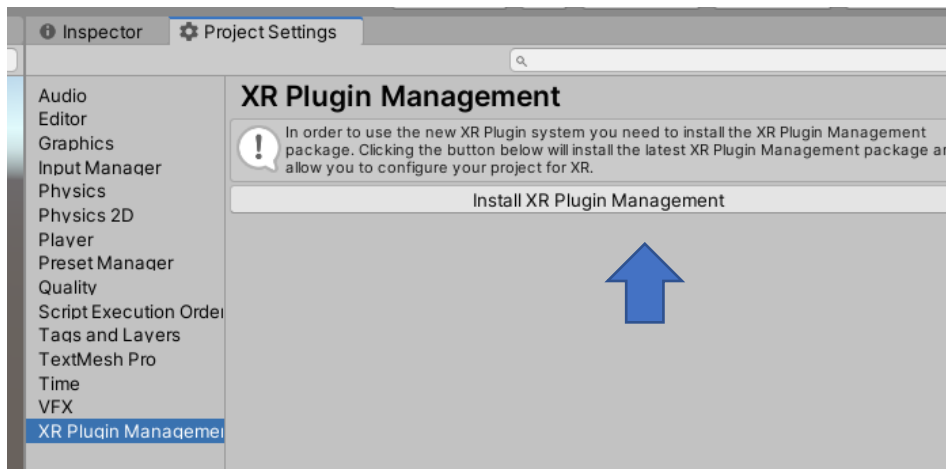## C) Let's Import stuff! (and get rid of errors)

1. Go to the Asset Store tab (or webpage if on 2020), search for **"Pun 2"** and import the free version.
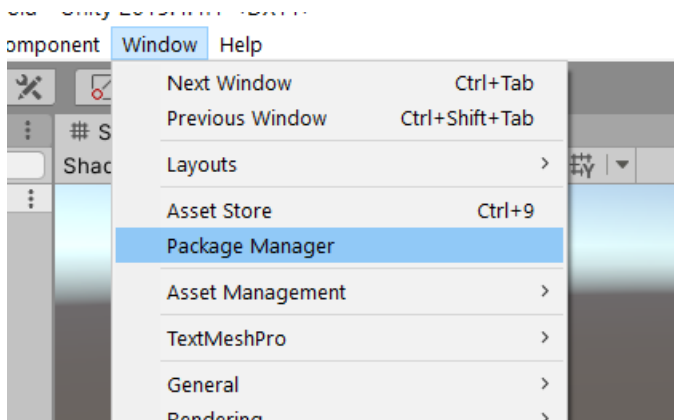

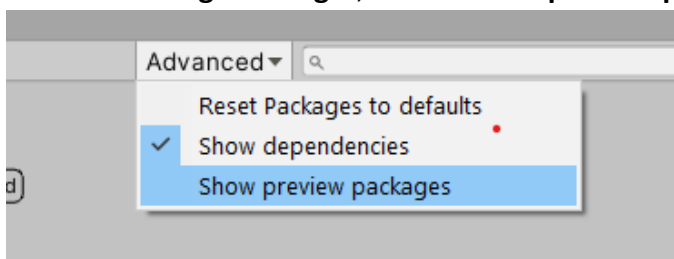2. After that, import **Photon Voice 2** (free).

3. Go again to the Project's settings tab (**File>Build Settings>Player Settings**) go to **XR Plugin management** and **Install XR Plugin management**, yes, click that button!
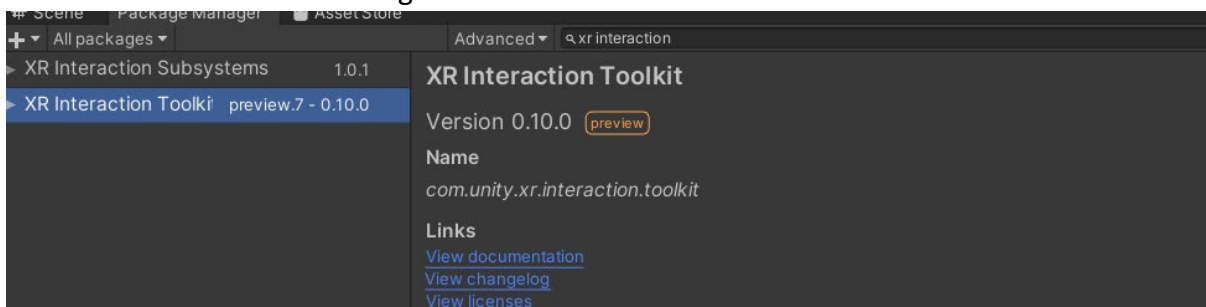

4. Go to **Window>Package Manager**
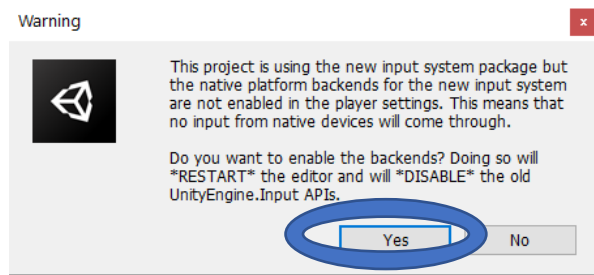

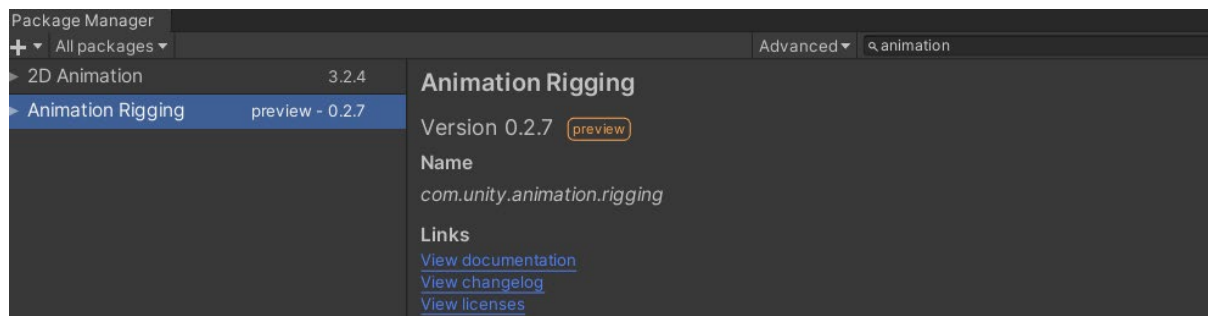5. In the **Package Manager**, check **"Show preview packages"** under **Advanced**.

6. And now Search for "**XR interaction**" and install the **XR Interaction Toolkit** with the "install" button at the lower right corner.

7. Finally import the **"Animation Rigging"** package, that will handle the avatar IK.
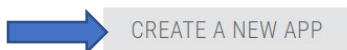

At this moment you should not have errors in the project. Cool!

D) Setting up Photon

Photon is a free (until a certain big limit) networking solution for Unity and, in my personal opinion, the best one around.

1. Go to www.photonengine.com and create an account if you don't have one already.

2. Click on "**Create New App**".

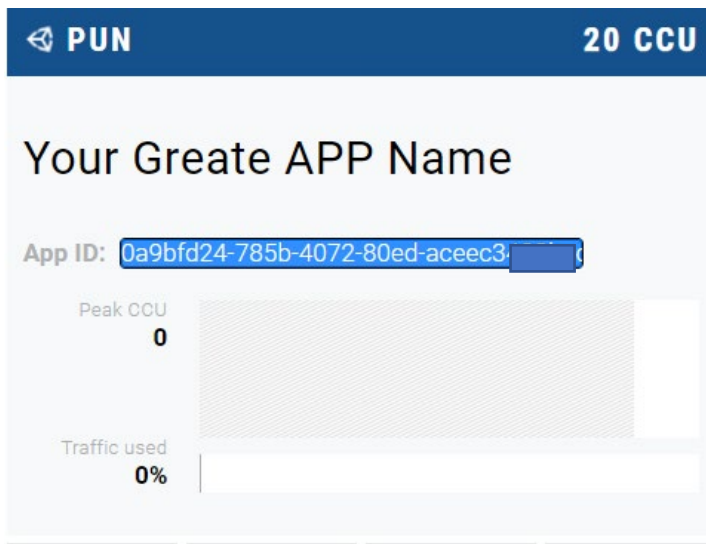3. Set the type to **PUN** and 4. Put a name to your app and hit Create.

Photon Type *

Photon PUN
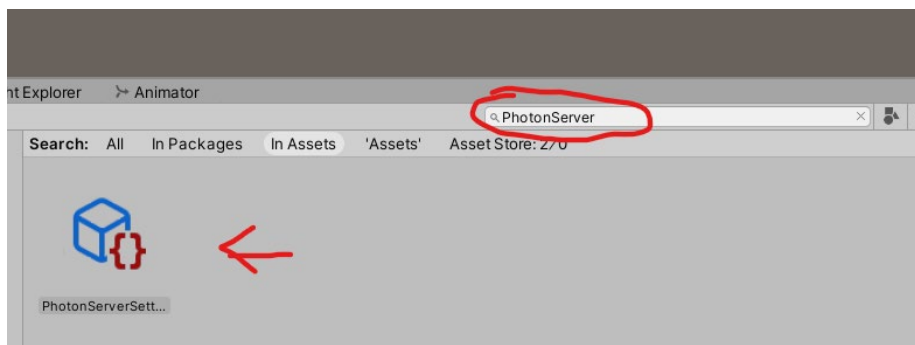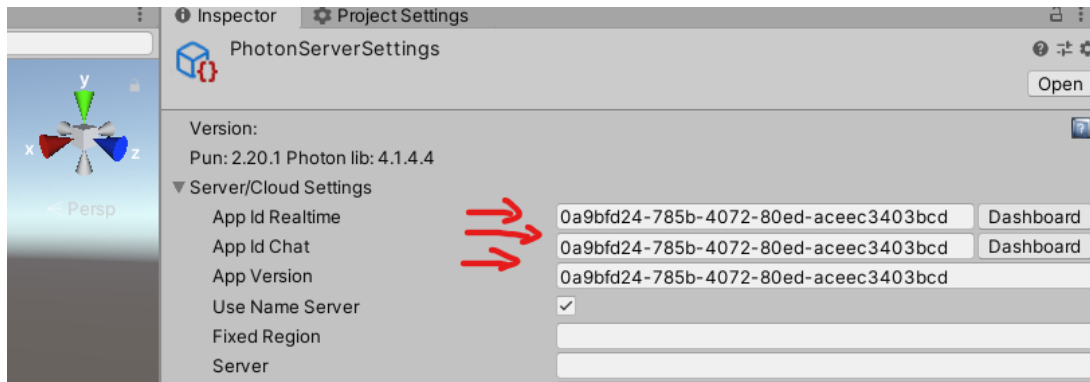
Name *

Your Greate APP Name

5. Find your app in the list and double click on the **App ID and copy it**. We need to fill this ID in Unity.

PUN   20 CCU

Your Greate APP Name

App ID: 0a9bfd24-785b-4072-80ed-aceec3

Peak CCU
        0

Traffic used
        0%

6. Go back to Unity and search for **PhotonServer** then click on it.

PhotonServerSett...

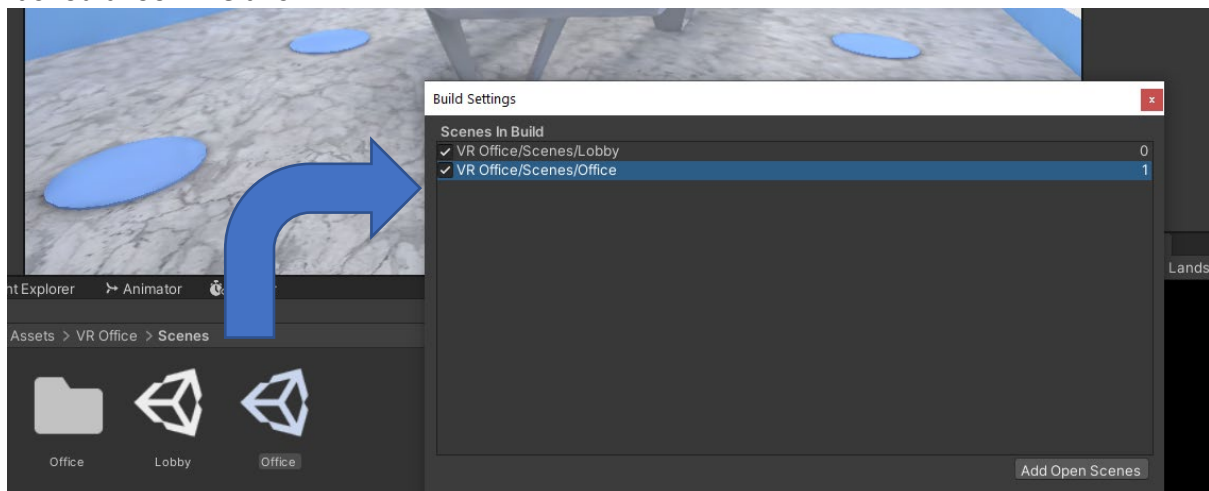7. Paste your App ID in these three places and then save your project.

Photon's Setup Ready!!

E) Setting the scenes and XR Settings: we are almost there!

1. Search for the 2 scenes included in the package by going to **VR Office>Scenes** and drag them to the build settings in the correspondent order.
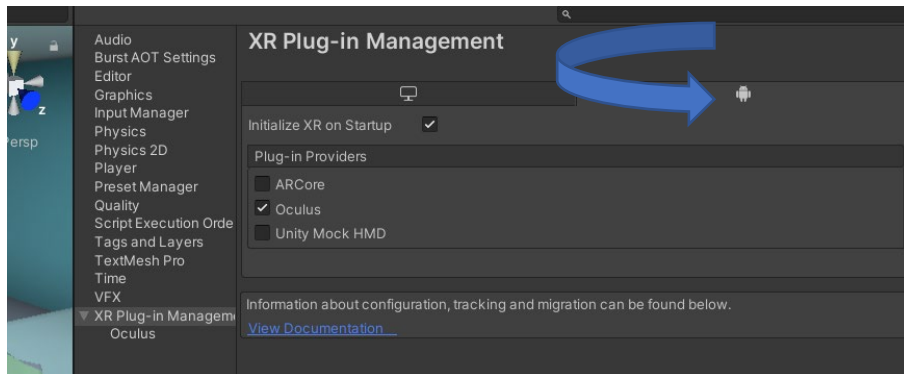
It should look like this:

2. Open the Office scene, and you should see something like this:

3. Now go to **Project Settings** again (**Edit>Project Settings**) and in the **XR Plugin Management** check your desired platform, in our case, Oculus (in **Android tab for Quest**).
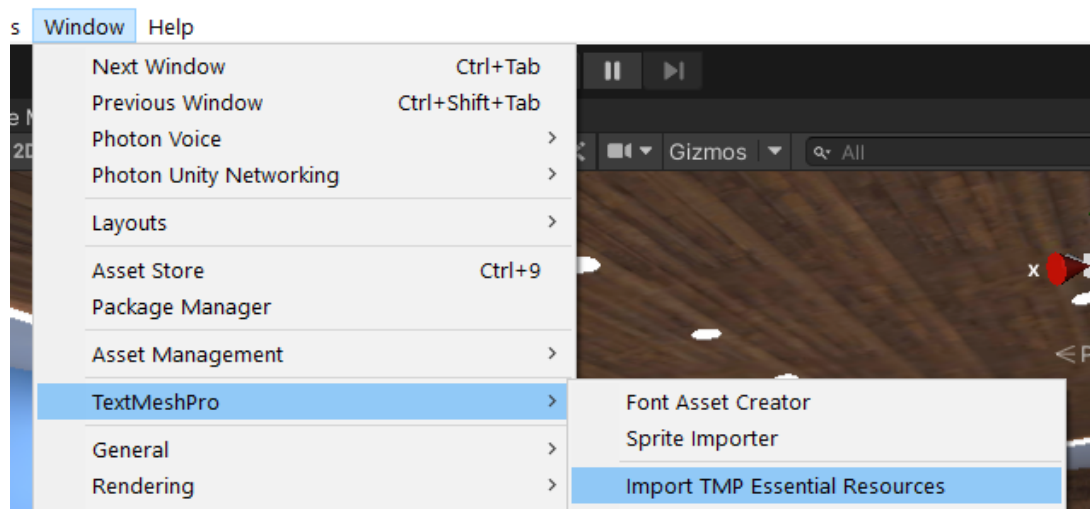

4. One more thing: we need to load a **TagManager** preset file, so the project works correctly. For this go to **Project Settings>Tag and Layers** and click on the presets icon:
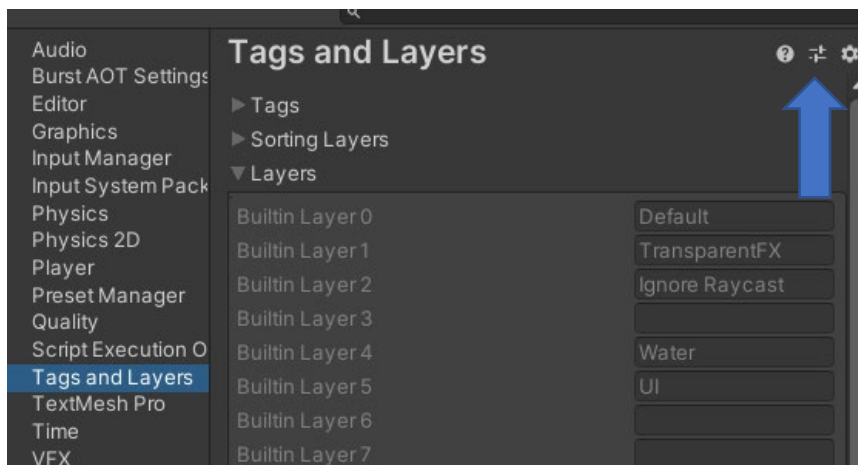
And then double click in the only preset:
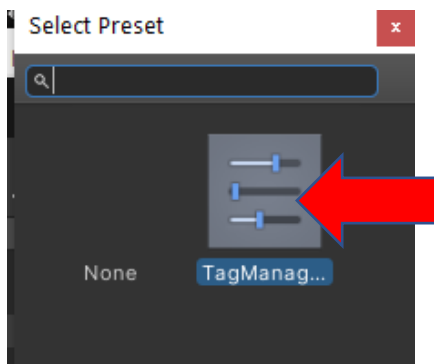
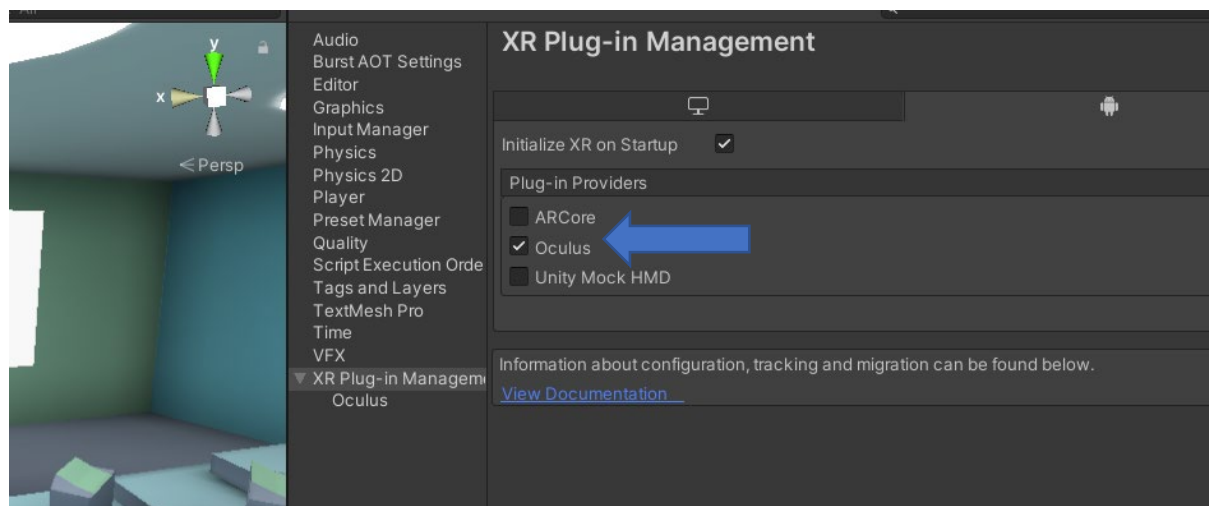7. Finally, let's import the **Text Mesh Pro (TMPro)** essentials:

We have everything needed in our project at this point, we can now build our apk!

F) VR build (Oculus Quest)

1. Select the **Manager Game Object** in the scene and make sure it is in **VR mode** (other modes are in progress).

2. In **Project Settings>XR Plug-in Management** make sure you have Oculus checked in the **Android tab.**

3. Connect your Oculus Quest headset and Build! (If a "Text mesh pro window" appears, cancel the build and **"Import TMPro essentials",** then start building again.)
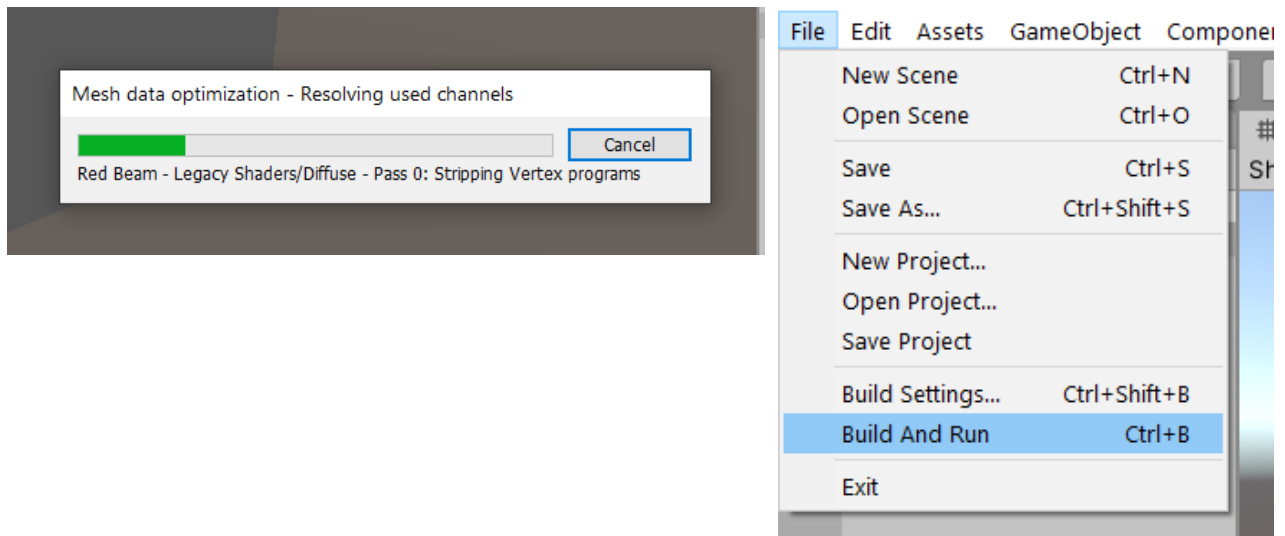
****We are in VR!!****

If you have any other questions/issues, don't hesitate to write to chiligamesco@gmail.com

UPDATE 1.1:

The project is now compatible with unity 2020.3 and the new input system/action based system, which is compatible with OpenXR. You will find 2 different VrRigs in the scene, you can drag the one you want to use (device or action based) to the Manager's VrRig field.