Tab 1

From Blender To Unreal

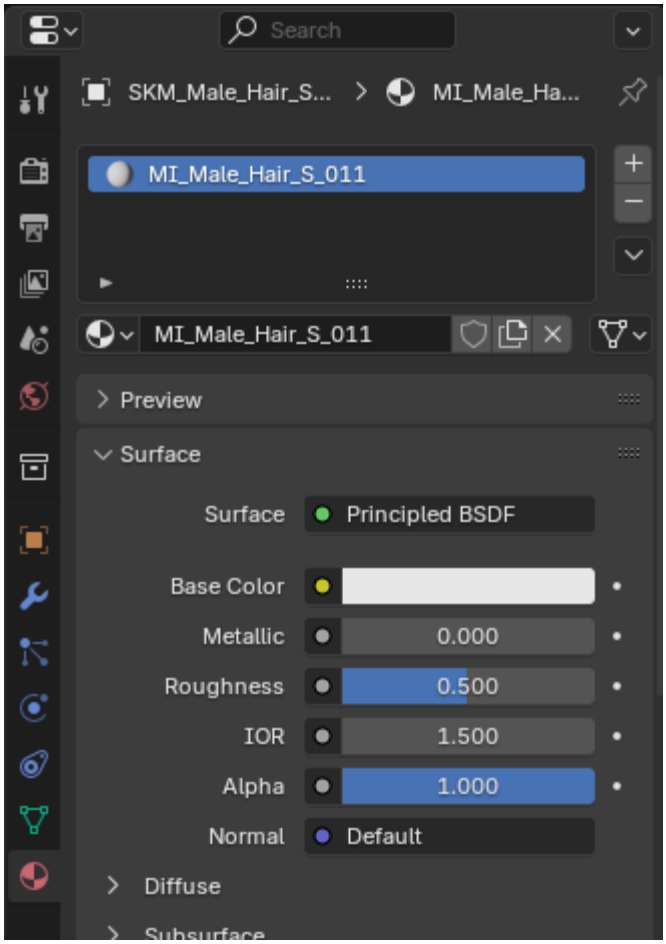
Prerequisites:

- Blender 4.2.2: Used to edit meshes. I recommend using v4.2 or older (to 4.0).
- PSK & PSA Blender Addon: Used to import & export the format that Fmodel gives you.
- UE 5.4.4 [InZOI]: Thank you to FrancisLouiss. Used to package assets for InZOI. **Need Epic Games Account & Github Access*
- InZOI Unreal Editor Project: Creation Tutorial Or Premade Project.
- A mesh you've already edited: This tutorial assumes you are ready to export a mesh from Blender to Unreal.

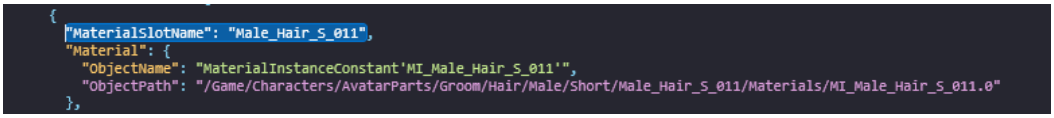
Notice: For the tutorial, I will be using a hair I've edited as an example. Naming and meshes will be different if you're working different assets. Name things according to your asset.*

Finding Asset Names:

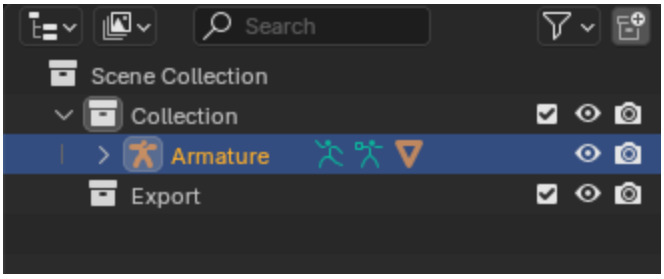
1. In Blender, make sure the mesh for your asset has all of it's materials
   They have to be named exactly the same. Since I am replacing *SKM_Male_Hair_S_011*, it should have imported with one material:



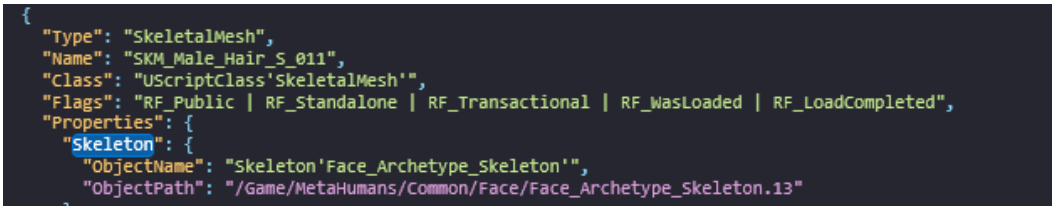
a. *MI_Male_Hair_S_011.* I will make sure these are named exactly the same as they are in FModel. If it's not, I can find the name of the material by going to the tab that opens for the mesh in FModel and using *CTRL + F* to search for *material*. I will look for *MaterialSlotName* and find *Male_Hair_S_011.* The names match the material I have in blender.



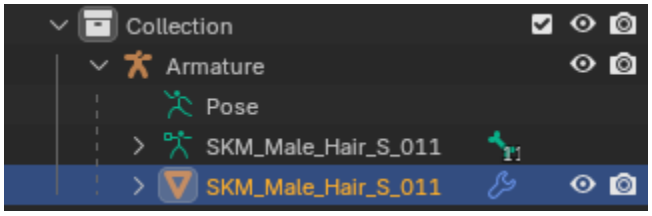
2. Rename the rig to *Armature*.



a. If you find yourself with more than one rig, you can export them one at a time. Name them appropriately when in Unreal. If they use the same skeleton, delete/force delete the additional one and assign the remaining skeleton to that mesh as well. If you are unsure what skeleton an asset uses, you can open the mesh in FModel and use CTRL + F to search for ‘skeleton’ within the tab that opens.

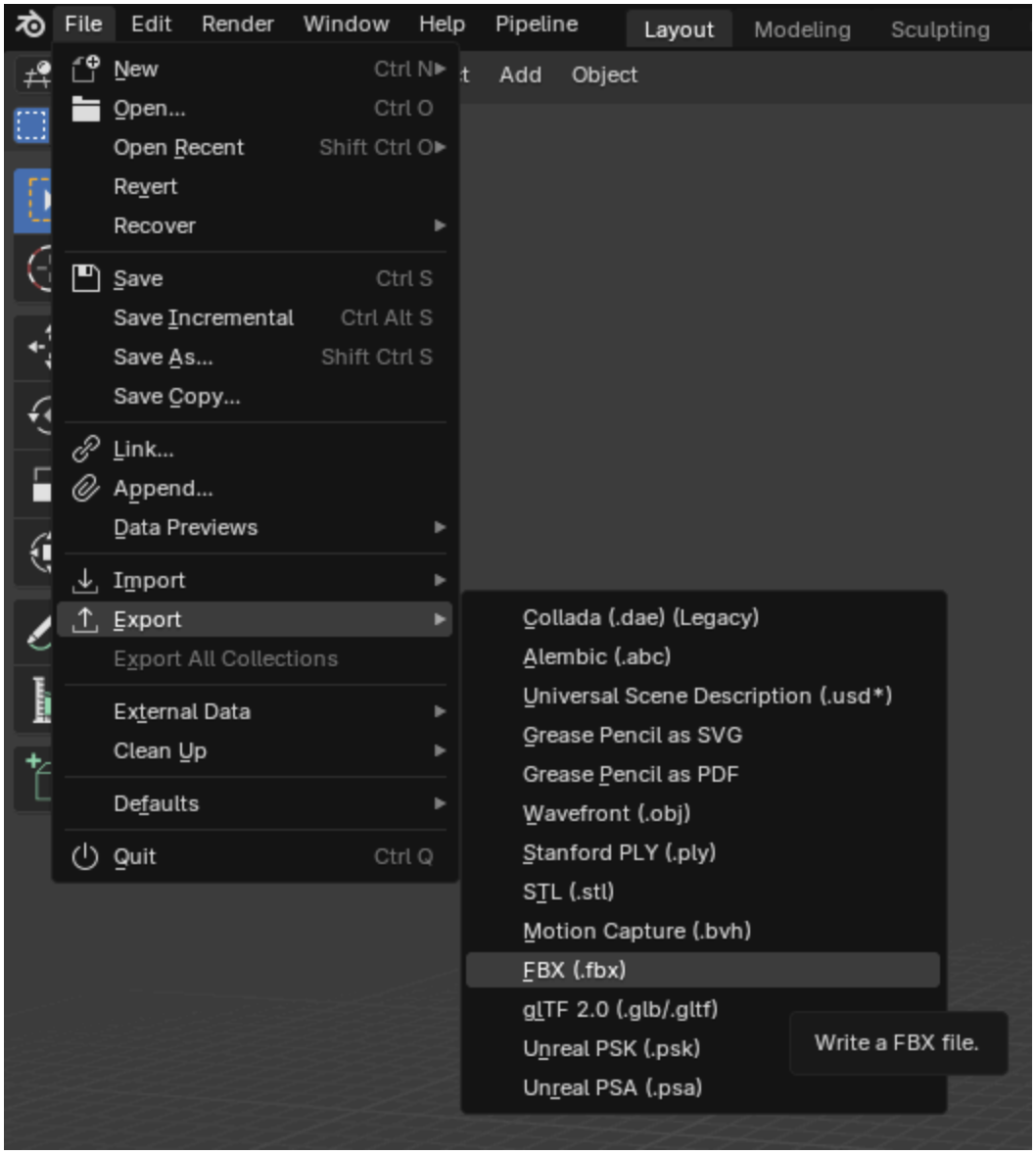


3. After naming the armature, I will click the arrow next to the name and in the drop down, I will find my mesh and rename it to *SKM_Male_Hair_S_011* if need be.

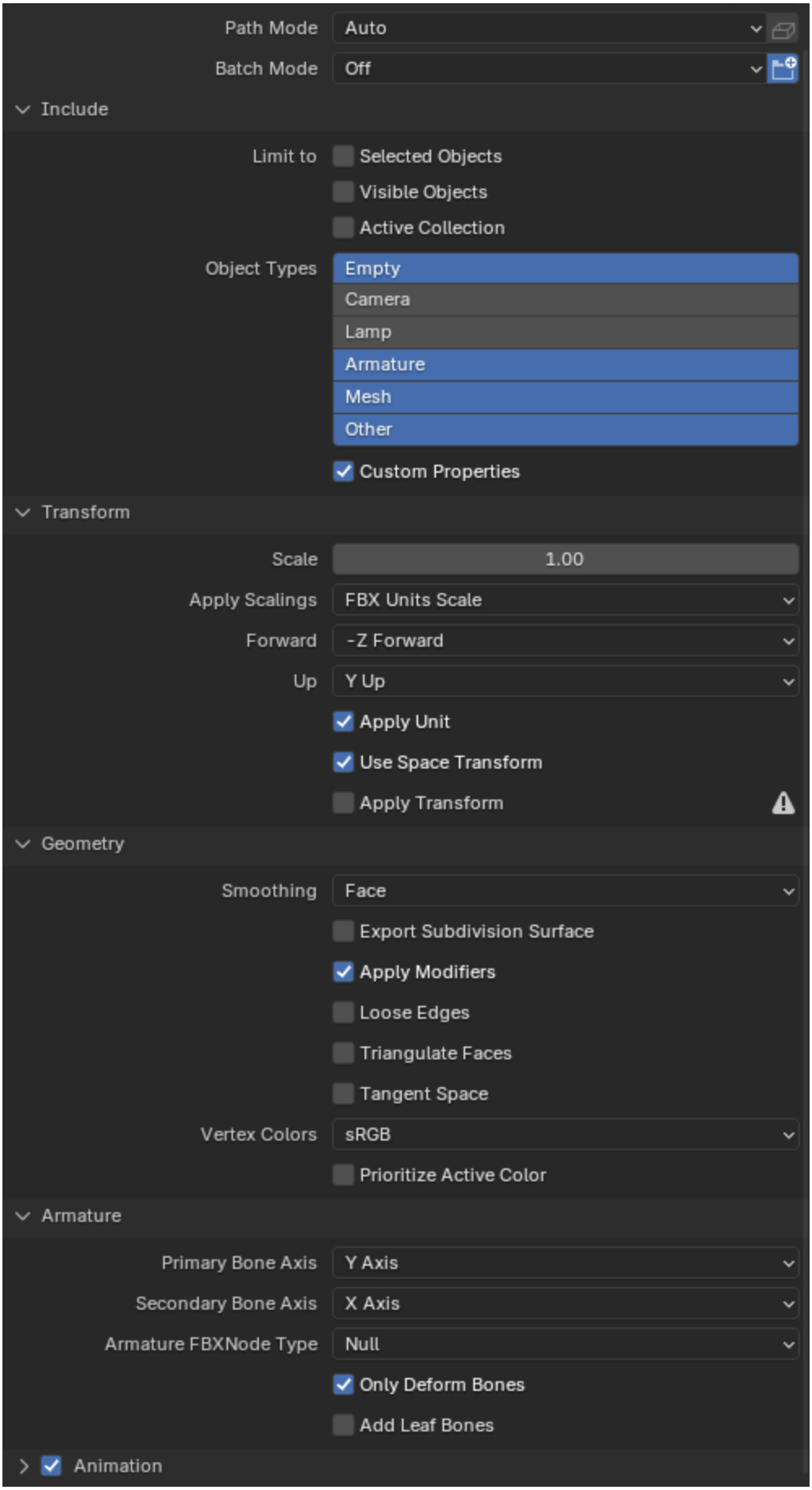


Exporting Assets To Unreal:

4. Now, we’ll export the mesh to Unreal. In the toolbar up top, find *File*. In that drop down, hover over *Export* and click *FBX.*

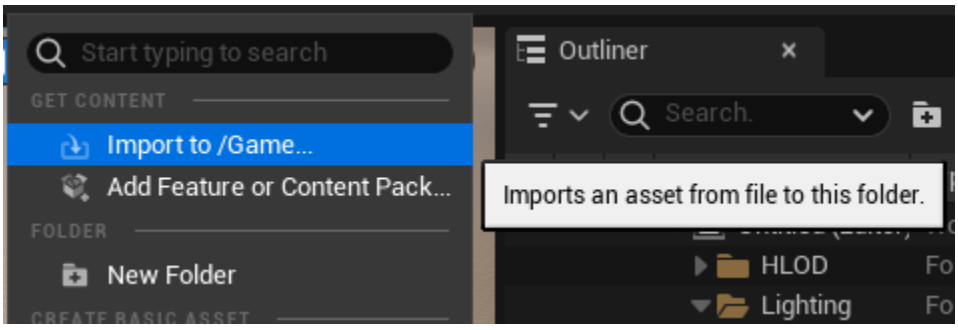


5. In the export box that pops up, change the settings to match these:

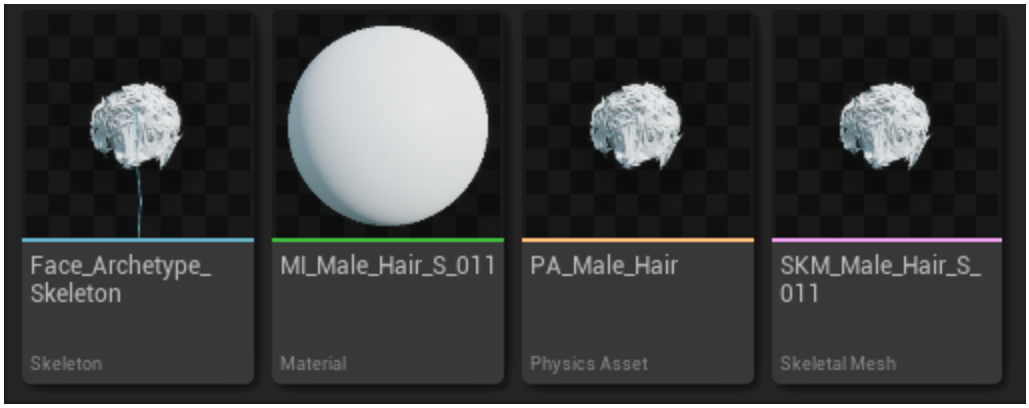

Path Mode
Auto
Batch Mode
Off
Include
Limit to
Selected Objects
Visible Objects
Active Collection
Object Types
Empty
Camera
Lamp
Armature
Mesh
Other
Custom Properties
Transform
Scale
1.00
Apply Scalings
FBX Units Scale
Forward
-Z Forward
Up
Y Up
Apply Unit
Use Space Transform
Apply Transform
Geometry
Smoothing
Face
Export Subdivision Surface
Apply Modifiers
Loose Edges
Triangulate Faces
Tangent Space
Vertex Colors
sRGB
Prioritize Active Color
Armature
Primary Bone Axis
Y Axis
Secondary Bone Axis
X Axis
Armature FBXNode Type
Null
Only Deform Bones
Add Leaf Bones
Animation

a. *Include:*
    i. *Object Types: Empty, Armature, Mesh, Other.*
    ii. Tick/Enable: *Custom Properties*
b. *Transforms:*
    i. *Scale: 1.00*
    ii. *Apply Scalings: FBX Units Scale*
    iii. *Forward: -Z Forward*
    iv. *Up: Y Up*
    v. Tick/Enable: *Apply Unity*
    vi. Tick/Enable: *Use Space Transform*
c. *Geometry:*
    i. *Smoothing: Face*
    ii. Tick/Enable: *Apply Modifiers (IF you have modifiers other than Armature but you will lose shape keys)*
d. *Armature:*
    i. Tick/Enable: *Only Deform Bones*
    ii. Untick/Disable*: Add Leaf Bones*

6. Export the FBX to a location you can remember. When you've done that, return to Unreal Editor and right click on an empty space in the Content Browser. In the menu that pops up, click
Import to /Game…



Find your FBX file and import it into Unreal. The settings in this box do not need to be changed. Click import.



7. You'll now have your edited assets within the Editor.

Packaging a Mod

Prerequisites:

- UE 5.4.4 [InZOI] Thank you to FrancisLouiss. Used to package assets for InZOI. **Need Epic Games Account & Github Access*
- InZOI Unreal Editor Project: Creation Tutorial Or Premade Project.

If you'd like, you can follow the packaging video provided by Aethel Hexem!

Renaming Assets and Setting Up Folder Hierarchy:

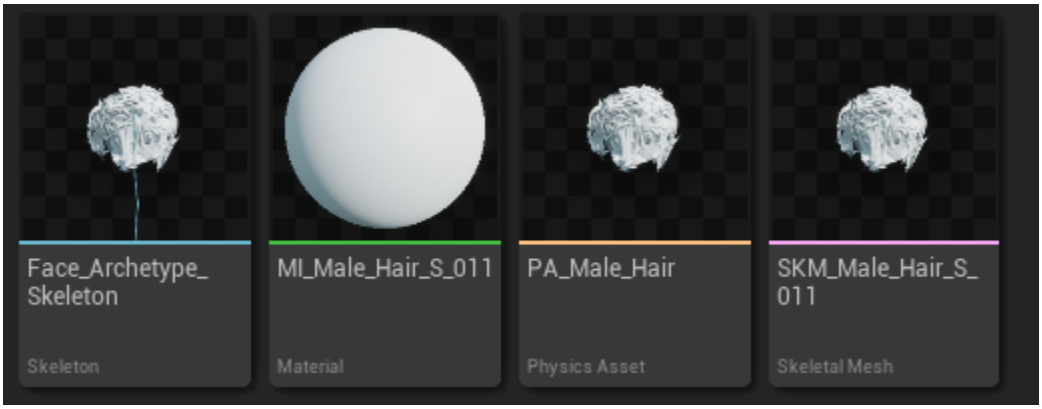
1. Open it. Rename everything to match exactly with the assets you will be replacing. Since I am replacing *SKM_Male_Hair_S_011*, I will name everything related to it accordingly. You should do the same for the assets you're replacing.

2. If you are unsure about the name of something, you can find the name of the material, skeleton & physics asset by going to the tab that opens for the mesh in FModel and using *CTRL + F* to search

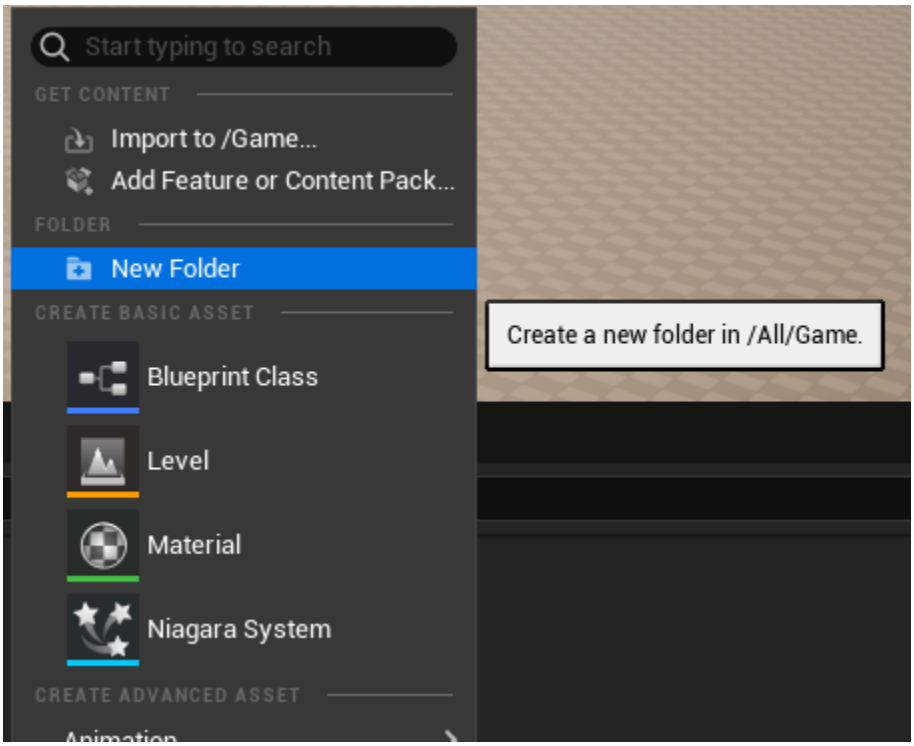
Notice: It may look like it's named correctly but sometimes exports have .mo or .ao at the end of them. Make sure to remove them.*

    a. Rename the mesh to *SKM_Male_Hair_S_011*
    b. Rename the Physics Asset to *PA_Male_Hair*

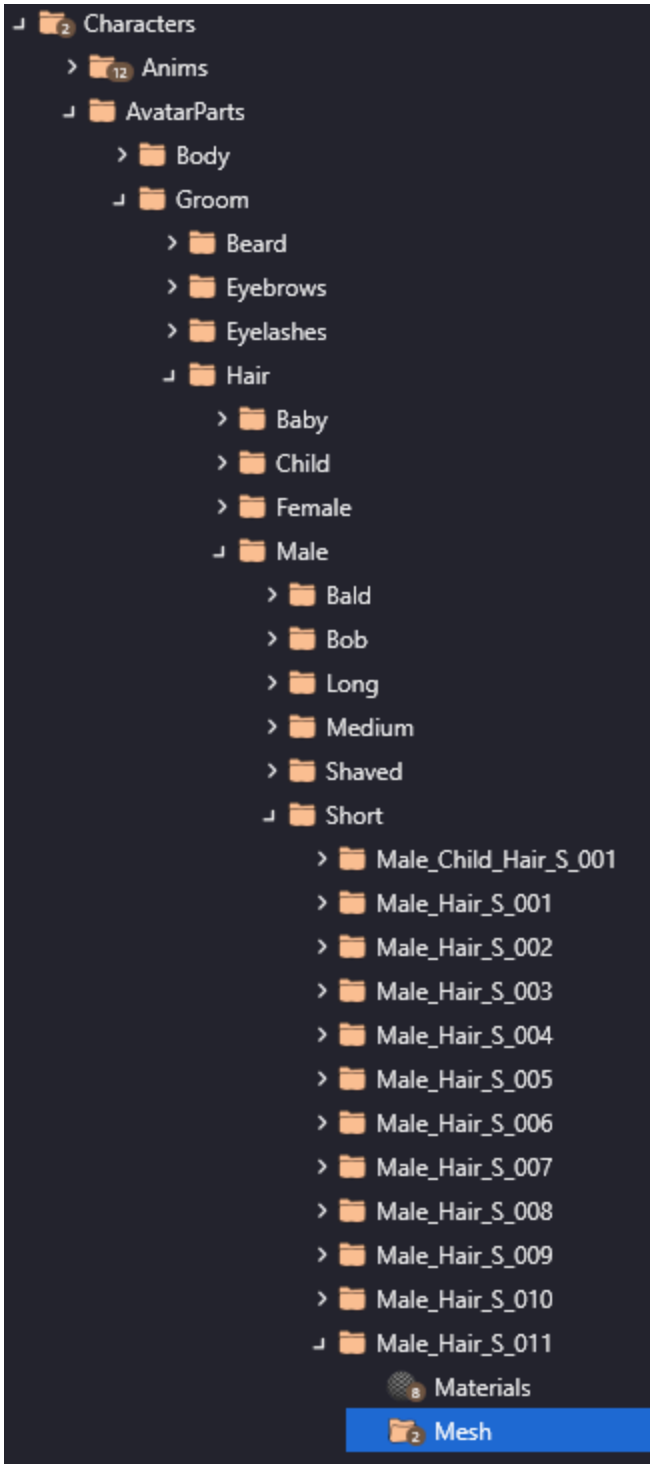
c. Rename the Skeleton to *Face_Archetype_Skeleton*



3. Now, within Unreal. you will set up the folder hierarchy for the mesh you plan to replace. Right click on an empty space in the Content Browser window and in the drop down, create a new folder.



a. Everything must have the exactly the same name and structure folder wise. This can be found in FModel.



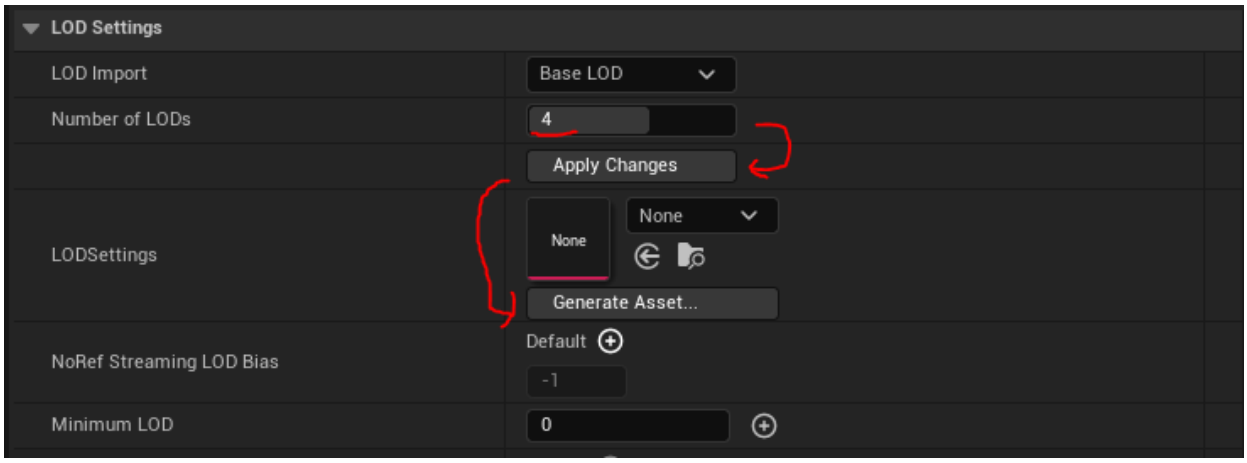
Each asset should go in its respective folder.

a. Hair Mesh: *Content/Characters/AvatarParts/Groom/Hair/Male/Short/Male_Hair_S_011/Mesh*
b. Materials: *Content/Characters/AvatarParts/Groom/Hair/Male/Short/Male_Hair_S_011/Materials*
c. Skeleton: *Content/MetaHumans/Common/Face*
d. Physics Asset: */Content/Characters/AvatarParts/Body/Male/Mesh*

Notice: All of the materials and the skeleton dummy assets. Meaning, we are only using them to help our modified asset find the right directories where its textures and skeletons are within the game's existing files. That way, we don't have to rebuild anything from the ground up. We will not be shipping our mod with any of the dummy assets.*

Creating LODs:

4. Now, we'll make the LODs (Level of Detail) for our mesh so it displays properly when the game is on lower graphical settings
    a. Double click on your mesh asset in the *Content Browser* and in the *Asset Details* tab, scroll down until you find *Lod Settings*. Change the *Number of LODs* to 4 then click *Apply Changes.* After that, click *Generate Asset* and save it in the same place as your mesh asset.

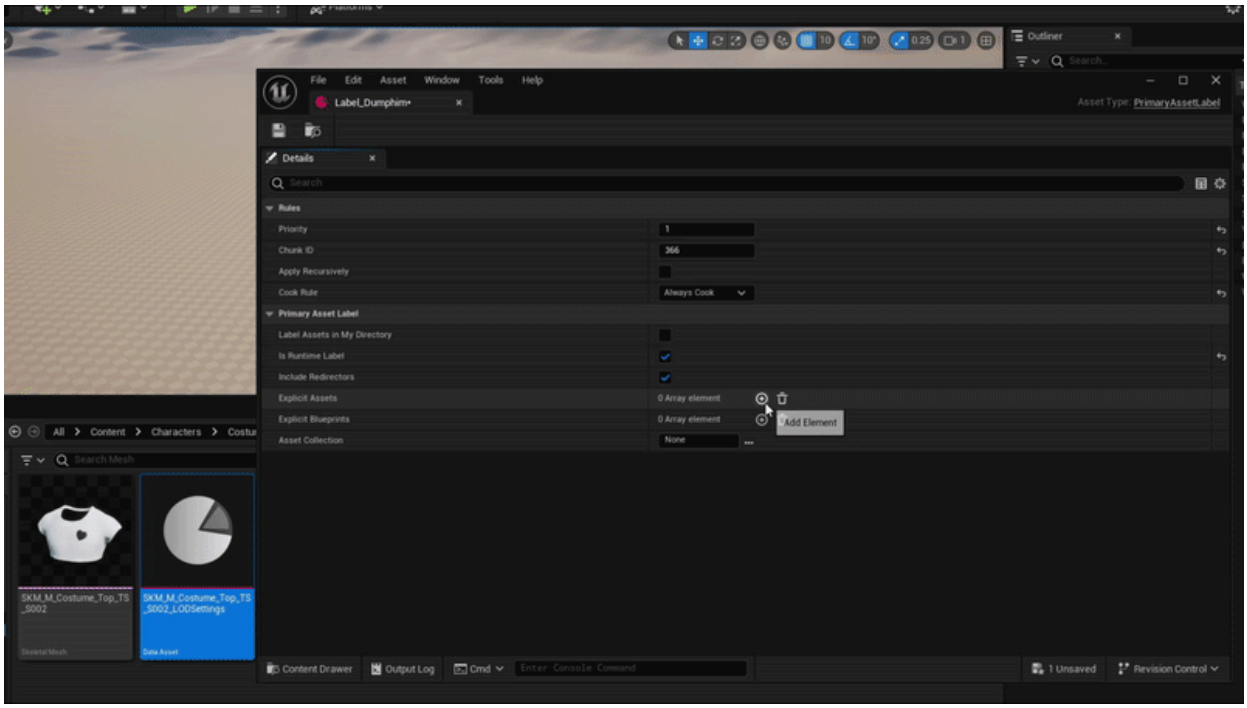


Save then close this window.

Notice: Remember to regenerate LODs if you change and reimport the mesh.*

Adding Your Assets To Your Chunk:

5. With everything in order, we can package our mod! However, we need to assign the mesh for our hair and its LODs to the chunk we made so only those assets are packaged.
    a. Open your primary asset label and underneath *Explicit Assets,* click the plus button twice to create two new entries.
    b. Now, navigate to the folder your mesh and LOD data are in.
    c. Selecting one at a time, drag and drop each asset into the blank entries.



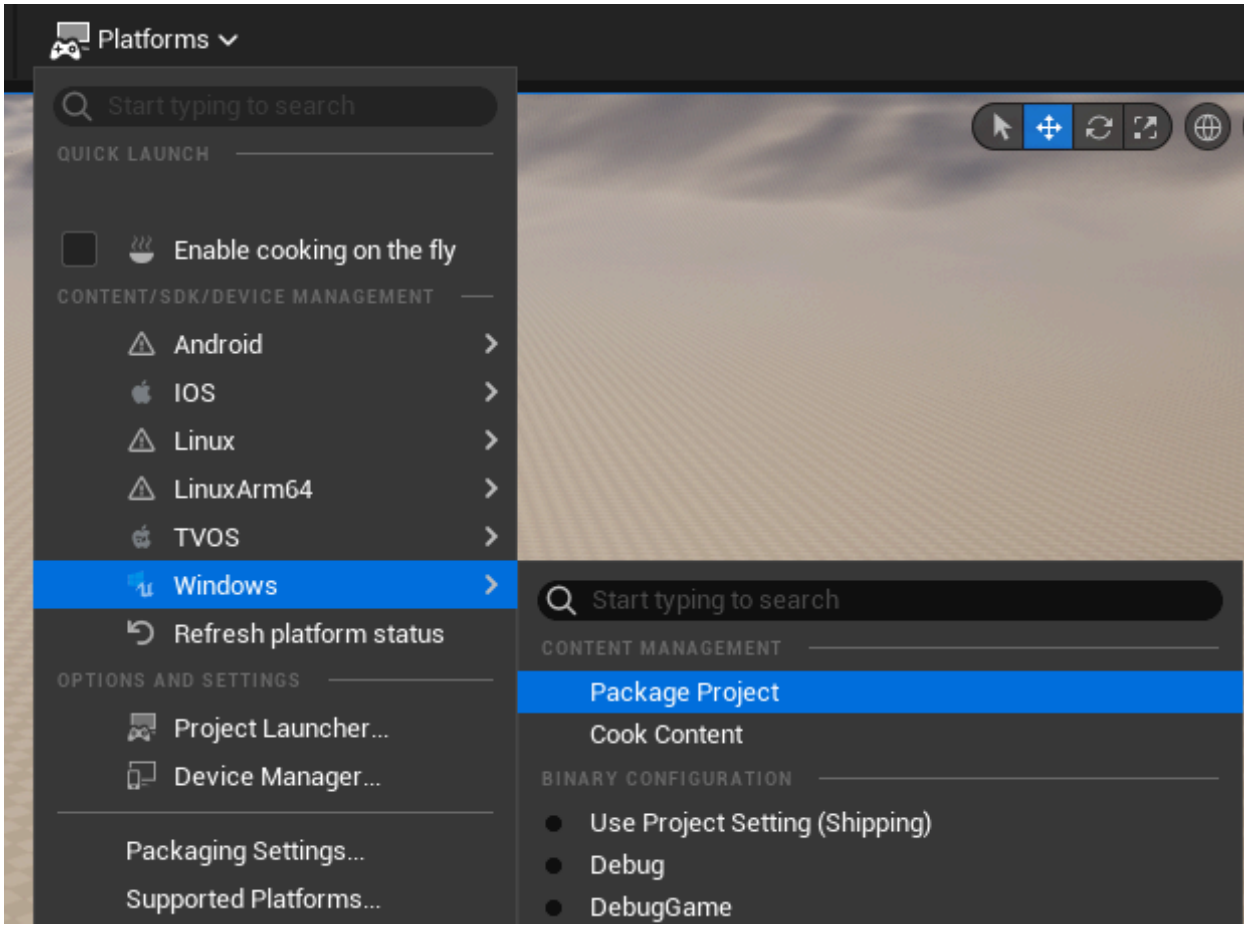
Notice: Assign **ONLY** the hair mesh and its LODs data to your chunk. That way, when we drop our mod into the game, it won't modify the skeleton or hair materials since that is not what we want to do.*

6. Make sure to save your work using the floppy disk icon at the bottom of the window

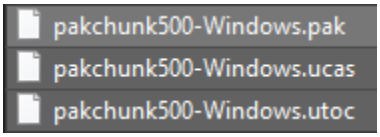
Packaging Your Mod:

7. You can now package your mod. Click on *Platforms* and in *Binary Configuration*, make sure it's set to *Shipping.* Then, under *Content Management* click *Package Project*. It will take a moment or two depending on the specs of your system.

Notice: If this is your first time cooking a project, it will take a while. All cooks you do after, will be much quicker.*

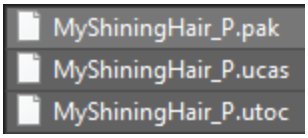


8. When it's done, find your files here:
    a. *[Your Unreal Project Directory]\Windows\BlueClient\Content\Paks*



Notice: Use only the files with the chunk number you assigned. If you use any of the chunk-0 or global files, it will cause your inZOI to crash.*

9. You can rename your mod files however you'd like, just make sure to add a *_P* at the end. When replacing assets, the P gives it priority in the load order. Without it, the normal assets will load before ours and our replacement won't work. (*i.e. MyShiningHair_P*).



10. Drop it in the *~mods* or *Logic Mods* folder which should be here:
    inZOI Demo\BlueClient\Content\Paks

Your asset replacement should now show in game!