[SCANS] Construction Site: Loader

DOCUMENTATION

Thank you for supporting **Scans Factory** by purchasing our pack, please check our other products on the UE Marketplace. If you have any problems or suggestions please write to us at: info@scansfactory.com

[SCANS] Construction Site: Loader pack features a fully functioning loader with its environment.

Unreal Engine 4

The Loader is based on an **Unreal Engine Vehicle System** with added physics of an arm and bucket. It interacts with the environment - it deforms the landscape, reacts to water and mud (wheels get dirty), has an extensive system of particles, and more - check video.

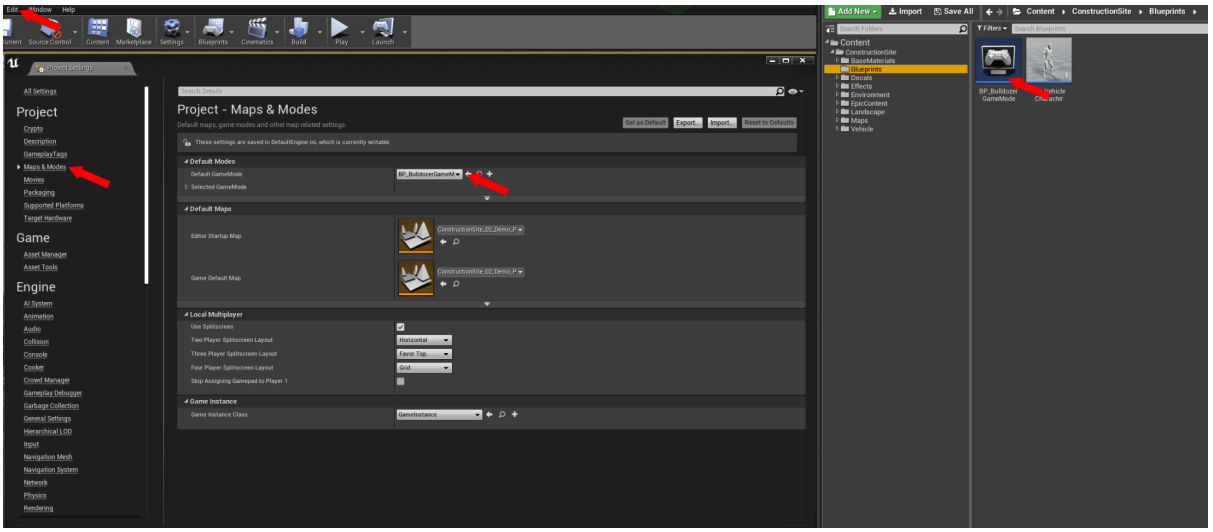
Unreal Engine 5

The Loader is based on **Unreal Engine Chaos Vehicles**. All functionalities have been adapted to the new version.

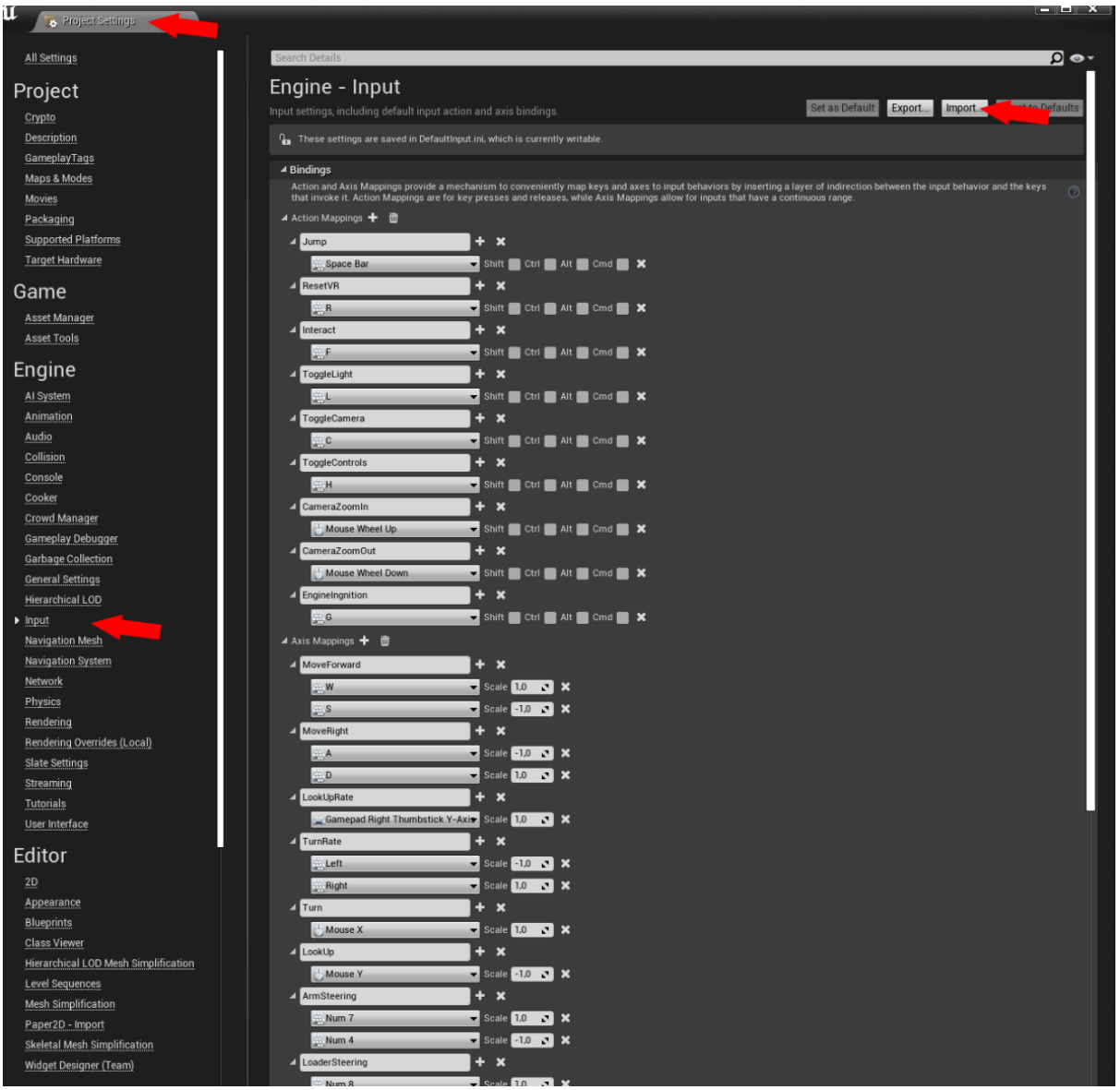
Unreal Enine 4

First Steps

The pack is based on a modified **Third Person template**, after adding it to your project you need to change **Default GameMode** in **Project Settings** to our **BP_BulldozerGameMode**.



If you add a pack to the Blank project you should definite Engine input or import it, simply download it from our drive - [Loader input]



The name of the demo scene is **ConstructionSite_02_Demo_P.**

Pawn

We add to the standard Third Person template pawn some extra functions:

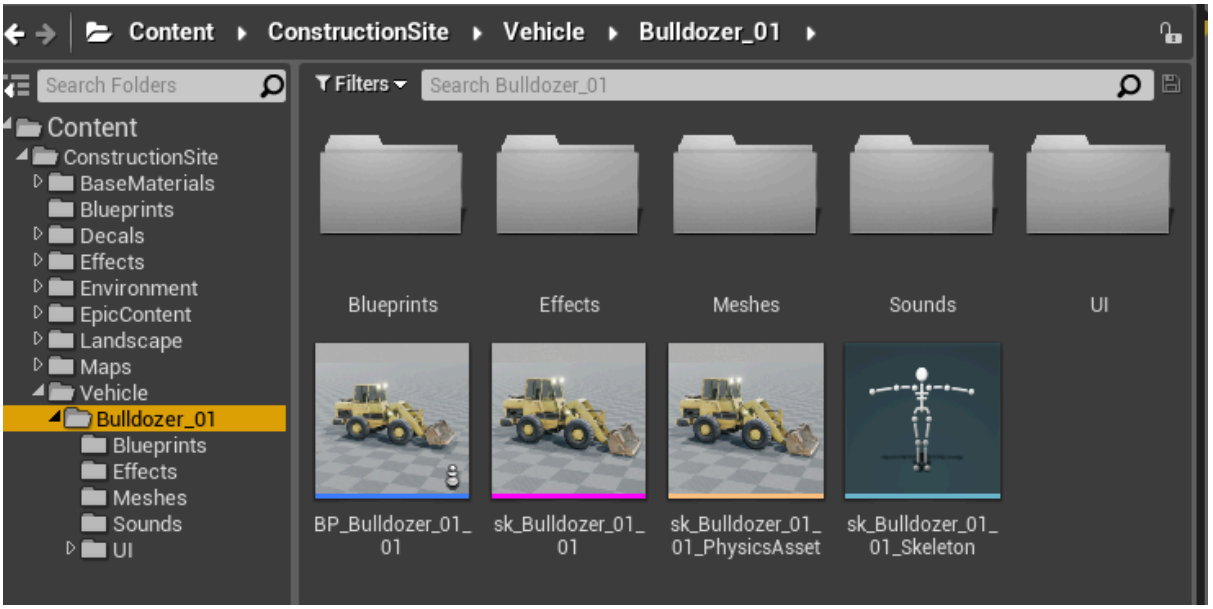
- switch the camera from **Third Person** to **First Person** by pressing **C** on the keyboard
- change the camera to **Top Third Person** by pressing **T** on the keyboard

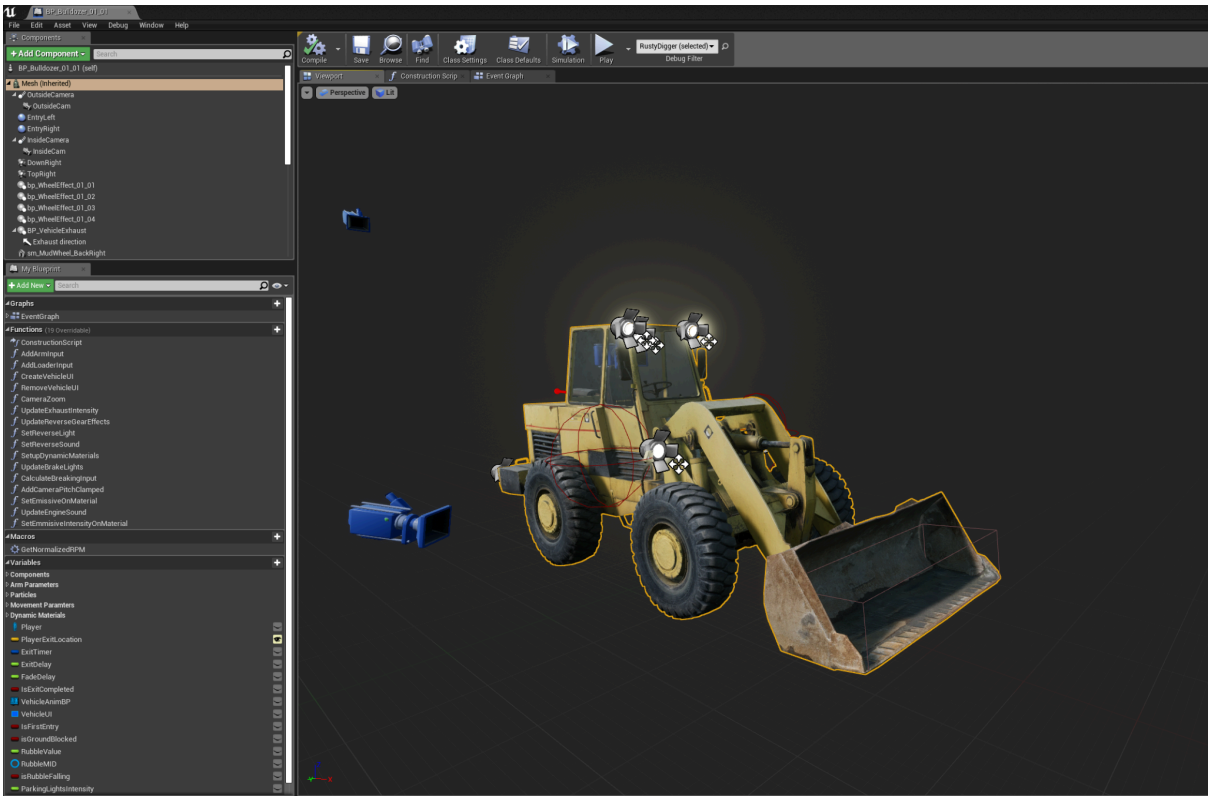
- pawn flashlight by pressing **L** on the keyboard
- functions for pawn footprints

- functions for pawn react with the landscape - water and mud

Bulldozer

Bulldozer BluePrint's name is **BP_Bulldozer_01_01**

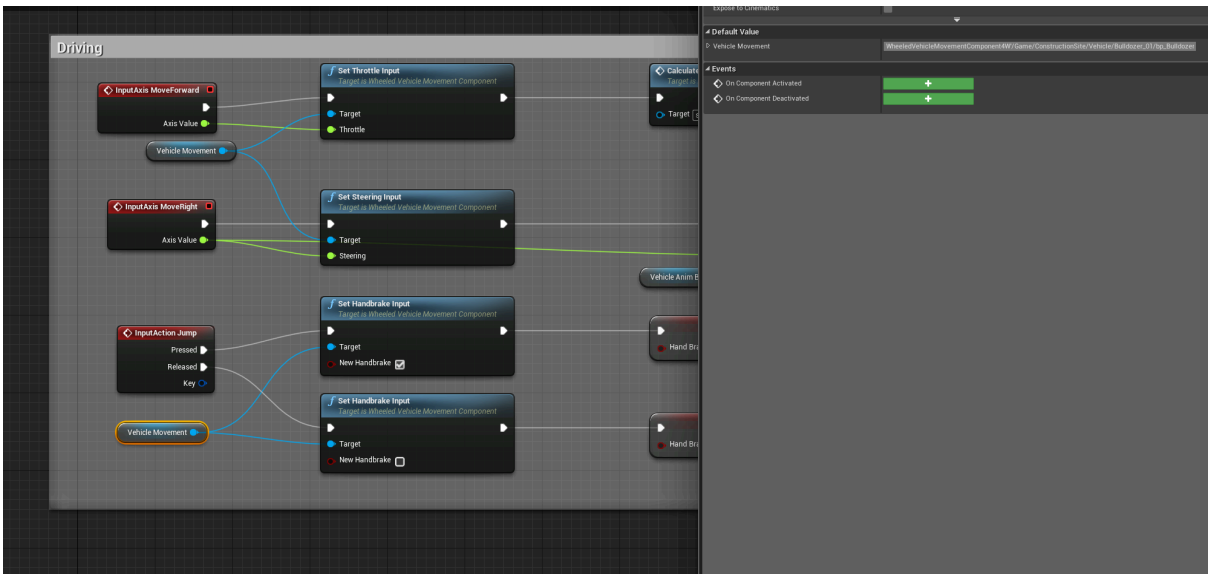




Bulldozer keyology

To get pawn into the bulldozer press **F** on the keyboard near the trigger area.

You can easily edit the driving characteristics of the bulldozer because it is based on the **Unreal Engine** [Vehicle System](#).



Animation of arm and bucket are based on **Animation BluePrint**.

Bulldozer features

- deforms the landscape

- reacts to water and mud

- wheels become dirty after defrosting landscape and clean after contact with water-based on layers system

- the bucket has interaction with objects possessing physics

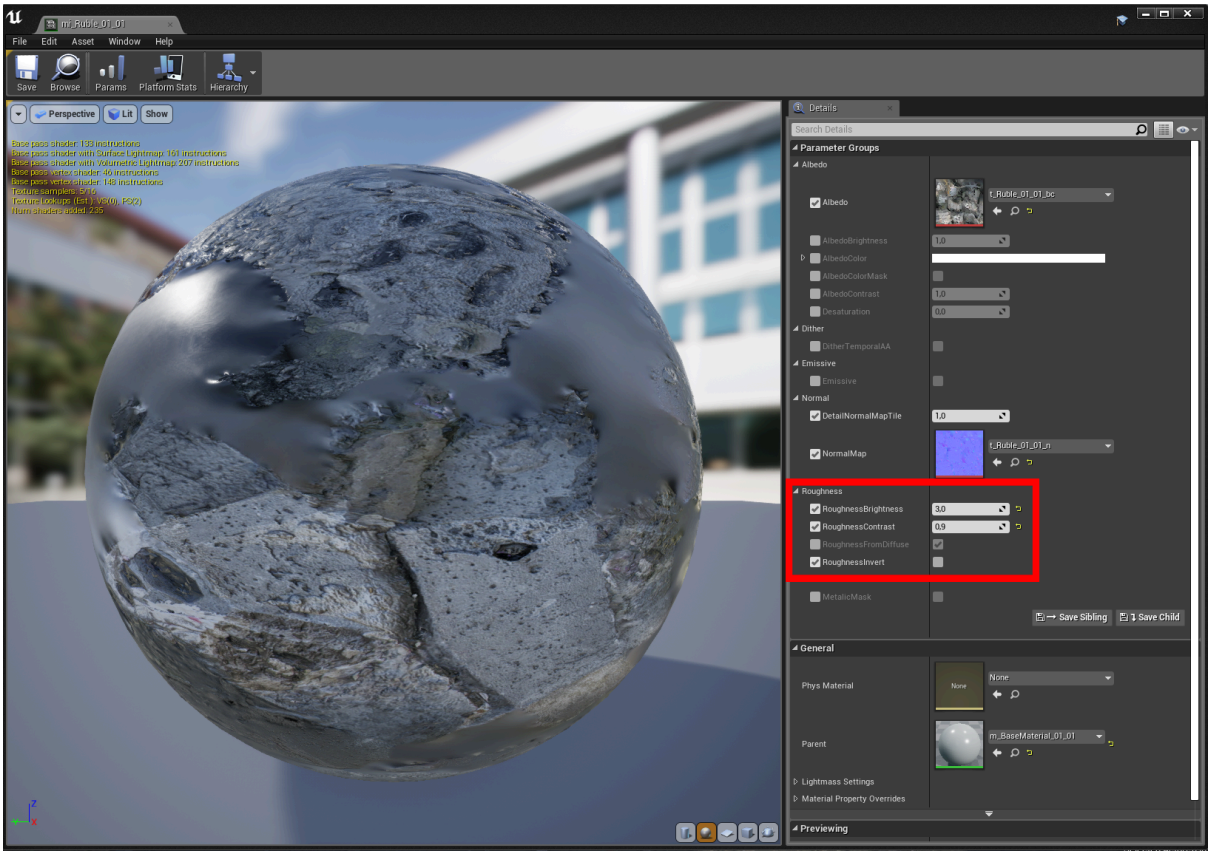
- In contact with the landscape, it flattens the area (heightmap) and the fake ground appears in the bucket, after being lifted it spills out the load - based on particles.

- and more like 3 definite cameras, lights, steering wheel, and lever animation.

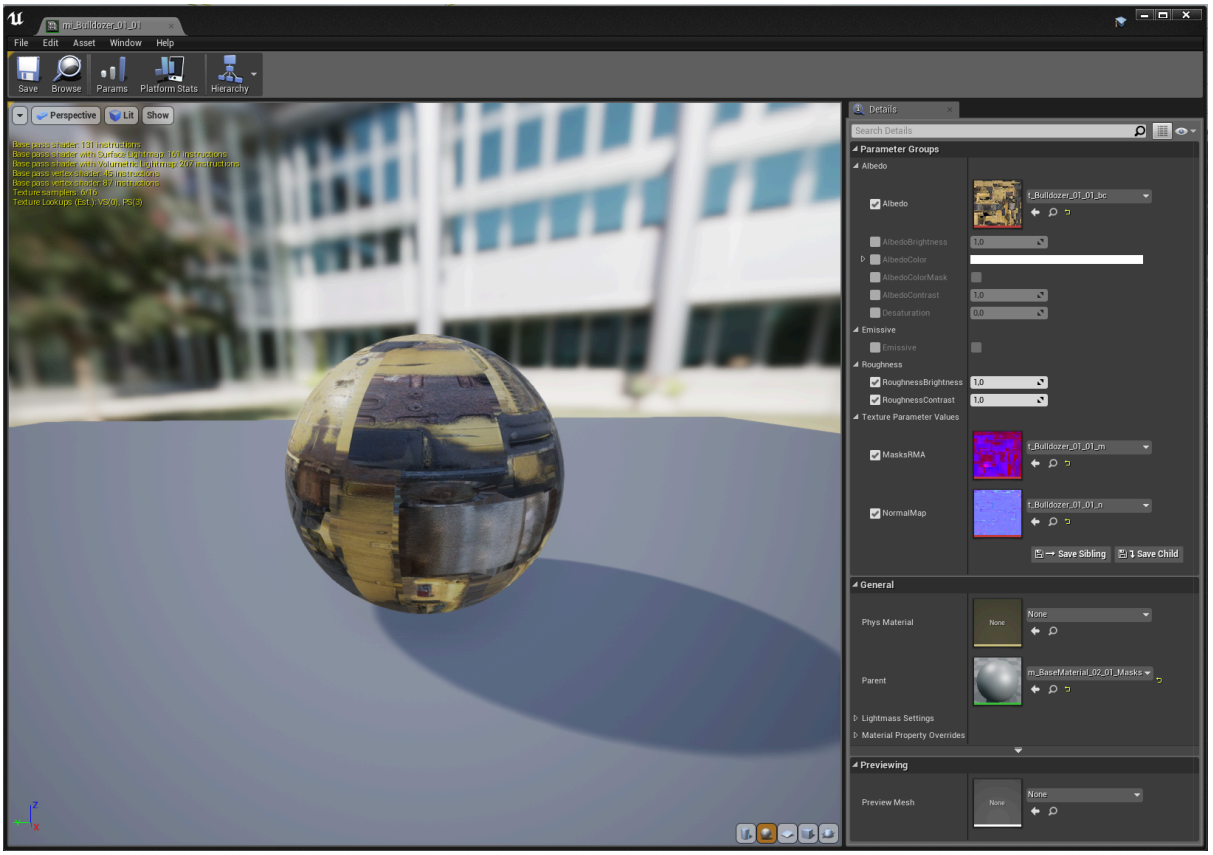
Materials

Two main materials are **m_BaseMaterial_01_01** and **m_BaseMaterial_02_01_Masks**.

m_BaseMaterial_01_01 is special optimal material for models based on photogrammetry scans with many textures. We decide for optimization reasons, to resign from additional texture and generate roughness texture from base color. You can easily control a roughness by two parameters and easy invert it by switch:



m_BaseMaterial_02_01_Masks is based on PBR standard and use RMA mask (roughness, metalnes, ambient occlusion):

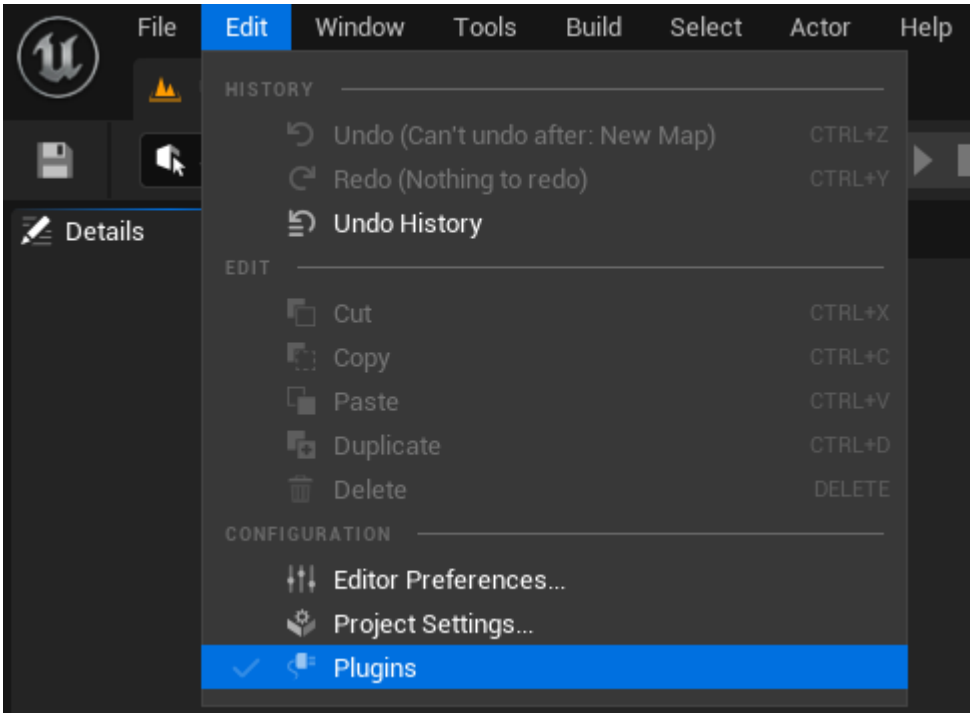


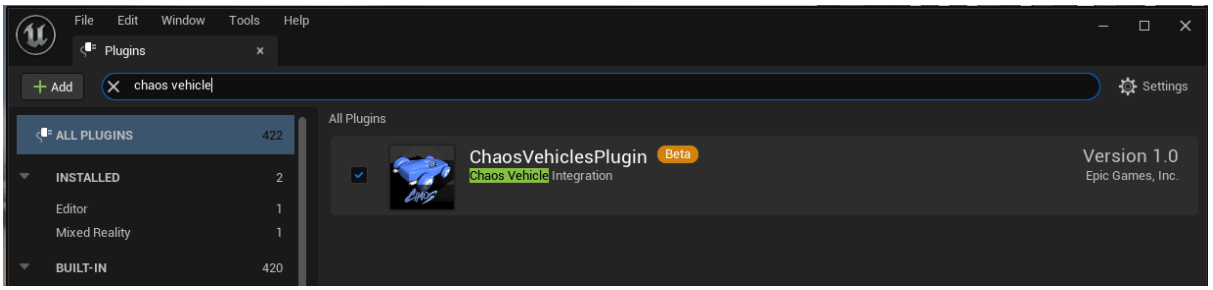
Unreal Enine 5.0 Update

The bulldozer is based on an [Unreal Engine Chaos Vehicles]. It interacts with the environment - it deforms the landscape, reacts to water and mud (wheels get dirty), has an extensive system of particles, and more - check video.

First Steps

You need to turn **ON** a **Chaos Vehicle Physics Plugin** **before importing this content**. Chaos Physics is set by default in Unreal Engine 5, but you still need to turn **ON** Chaos Vehicle Physics plugin.

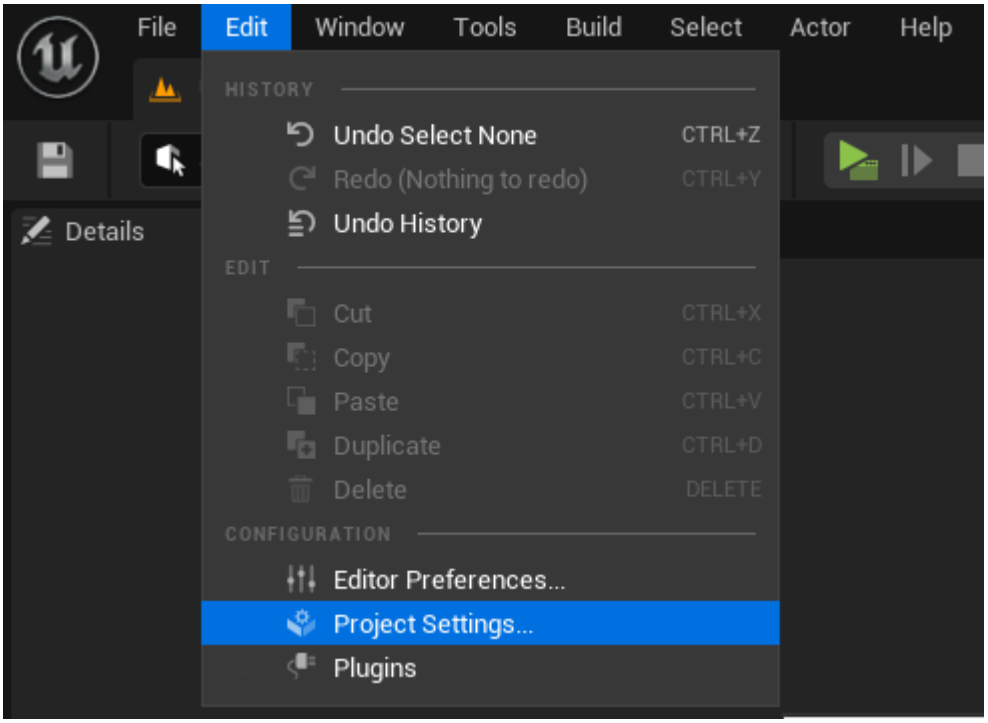


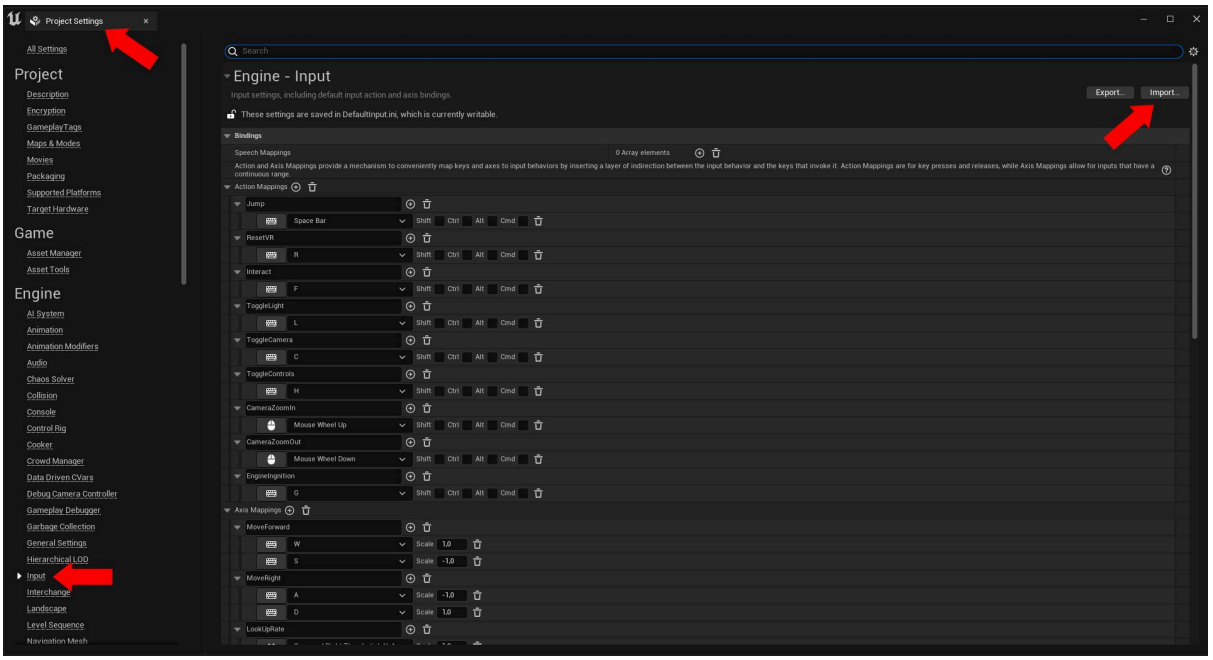


Main parts

Inputs

If you add a pack to the Blank project you should definite Engine input or import it, simply download it from our drive - [Loader input]

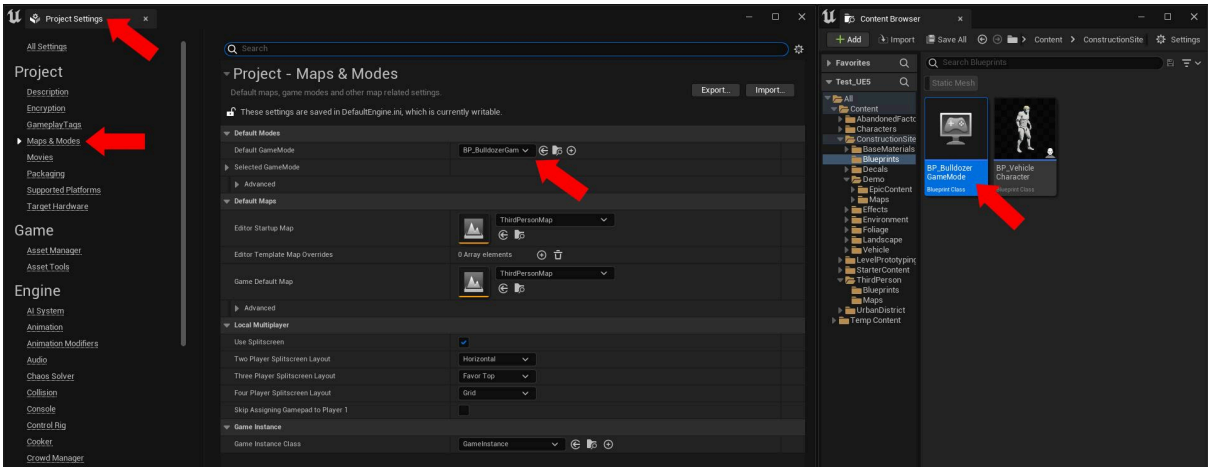




[Link to Import Input file](#).

GameMode

The pack is based on a modified **Third Person template**, after adding it to your project you need to change **Default GameMode** in **Project Settings** to our **BP_BulldozerGameMode**.



.../ConstructionSite/Blueprints/**BP_BulldozerGameMode**
DefaultPawnClass is set to BP_VehicleCharacter

Character

.../ConstructionSite/Blueprints/**BP_VehicleCharacter**
This is the default character that is used with Bulldozer vehicle. Mainly it handles:

- *entering vehicle,*
- *possesing/unpossesing player (with fade),*

and additionally it implements functionalities such as:

- *player flashlight,*
- *player camera change TPP/FPP,*
- *movement input,*
- *toggling HUD (help display).*

Bulldozer Blueprint

.../ConstructionSite/Blueprints/Vehicle/Bulldozer_01/**BP_Bulldozer_01_01**
This is the main vehicle that player can drive around.
*This blueprint derives from **WheeledVehiclePawn** which is a base class for vehicles in Chaos physics engine.Bulldozer driving behavior is done just by tweaking values in the **VehicleMovementComponent**, e.g. in the **MechanicalSetup** or **VehicleSetup** sections (in Details panel).*

*Player inputs are handled in the **Event Graph**:*

- *steering,*
- *accelerating,*
- *breaking,*
- *reversing,*
- *handbrake,*
- *manipulating loader arm,*
- *manipulating loader bucket.*

Additional vehicle functionalities:

- *toggle lights,*
- *wheel FX (mud),*
- *loader bucket debris (rubble) FX,*

- *handle fade on player enter/exit,*
- *and other minor FX effects.*

*In a Bulldozer blueprint there is also a **CameraSwitcher** component. It handles vehicle camera changes. On **BeginPlay** three cameras are set and when the player presses the **ToggleCamera** input button, the camera is changed.*
*Camera rotation and zoom is handled in **Camera rotation** and **Camera zoom in/out** sections in blueprint Event Graph. **Wheel Camera** is static that is why it is not present in these sections.*

Wheels

.../ConstructionSite/Blueprints/Vehicle/Bulldozer_01/Blueprints/**BP_BulldozerFrontWheel_01_01**
.../ConstructionSite/Blueprints/Vehicle/Bulldozer_01/Blueprints/**BP_BulldozerRearWheel_01_01**

Basic wheel setup for the vehicle. Most important values:

- *wheel radius/width,*
- *brake torque/hand brake torque*

AnimationBlueprint

.../ConstructionSite/Blueprints/Vehicle/Bulldozer_01/Blueprints/**AB_Bulldozer_01_01**
Base animation blueprint for bulldozer vehicle. Additionally to wheel setup this blueprint handles animations:

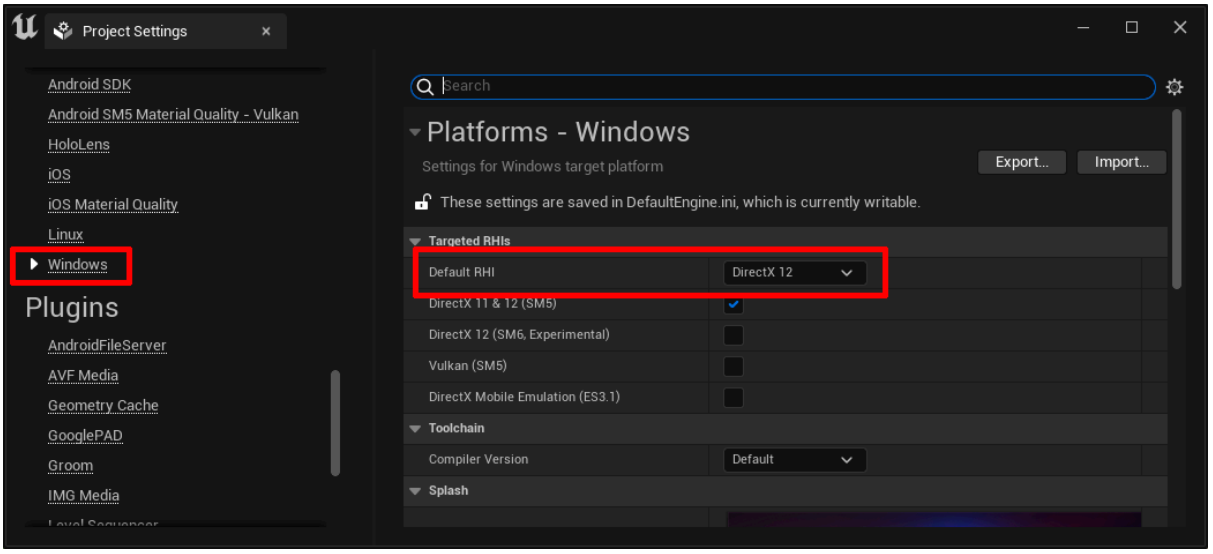
- *hydraulics setup for loader arm and bucket,*
- *secondary animations:*
    - *steering wheel,*
    - *speed/RPM gauges,*
    - *loader manipulation levers.*

Landscape Tessellation with Virtual Heightfield Mesh

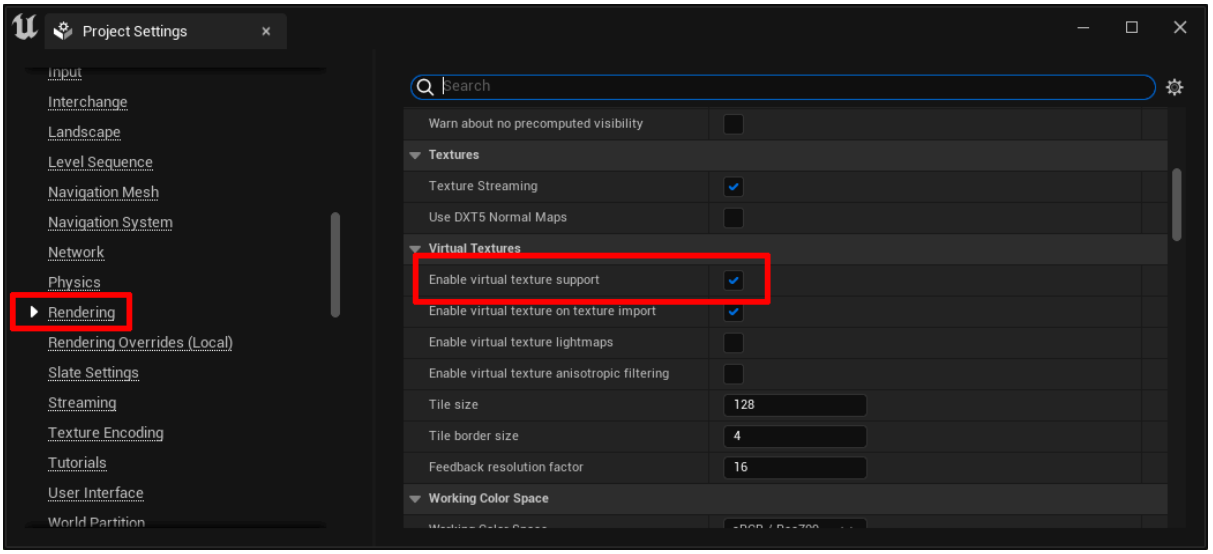
Unreal Engine 5 has no tessellation option. To use landscape with height map detail, Virtual Heightfield Mesh should be used.

Follow these steps for **each** Landscape actor at level:

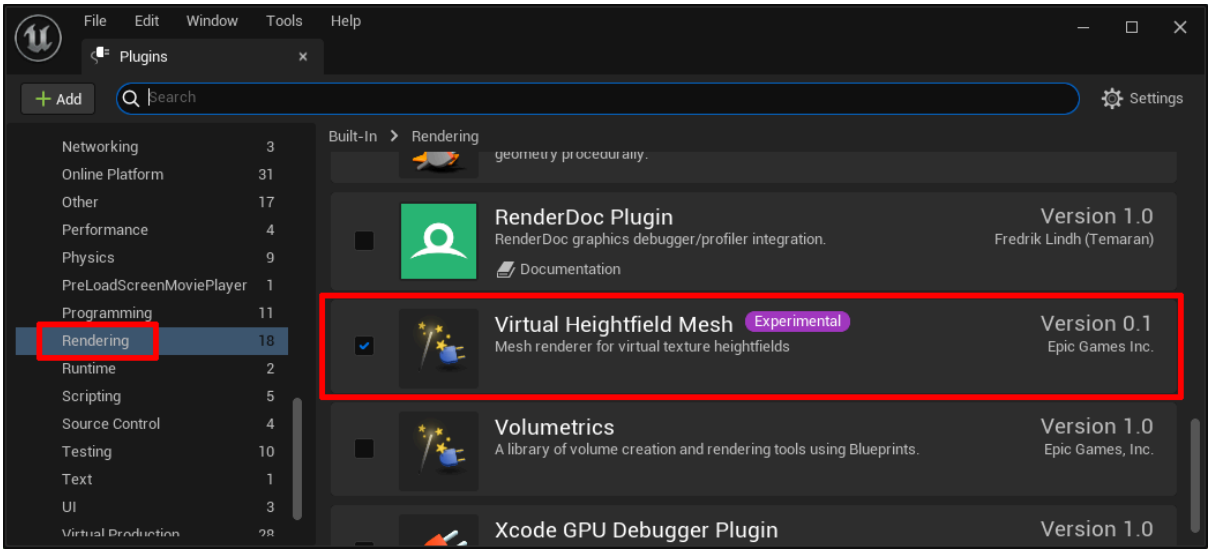
1. Make sure you have **DirectX 12 set as default RHI** in your project settings (Windows tab). New projects created in Unreal Engine 5 use DirectX 12 by default.



2. **Enable Virtual Texture support** in the Rendering tab in your project settings.



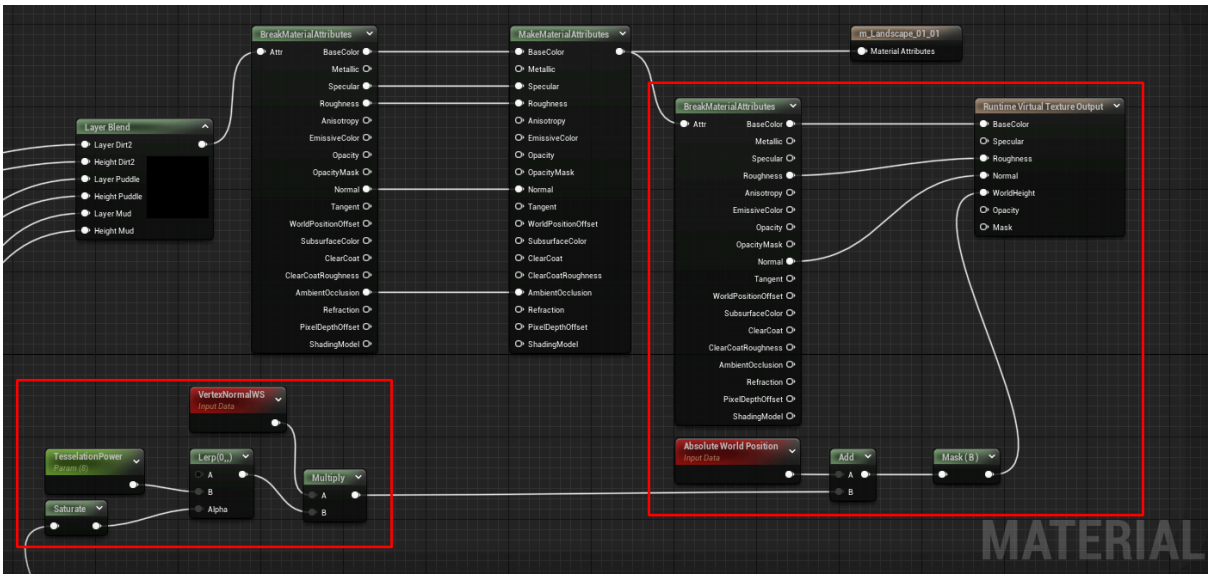
3. Enable the **Virtual Heightfield Mesh plugin**.



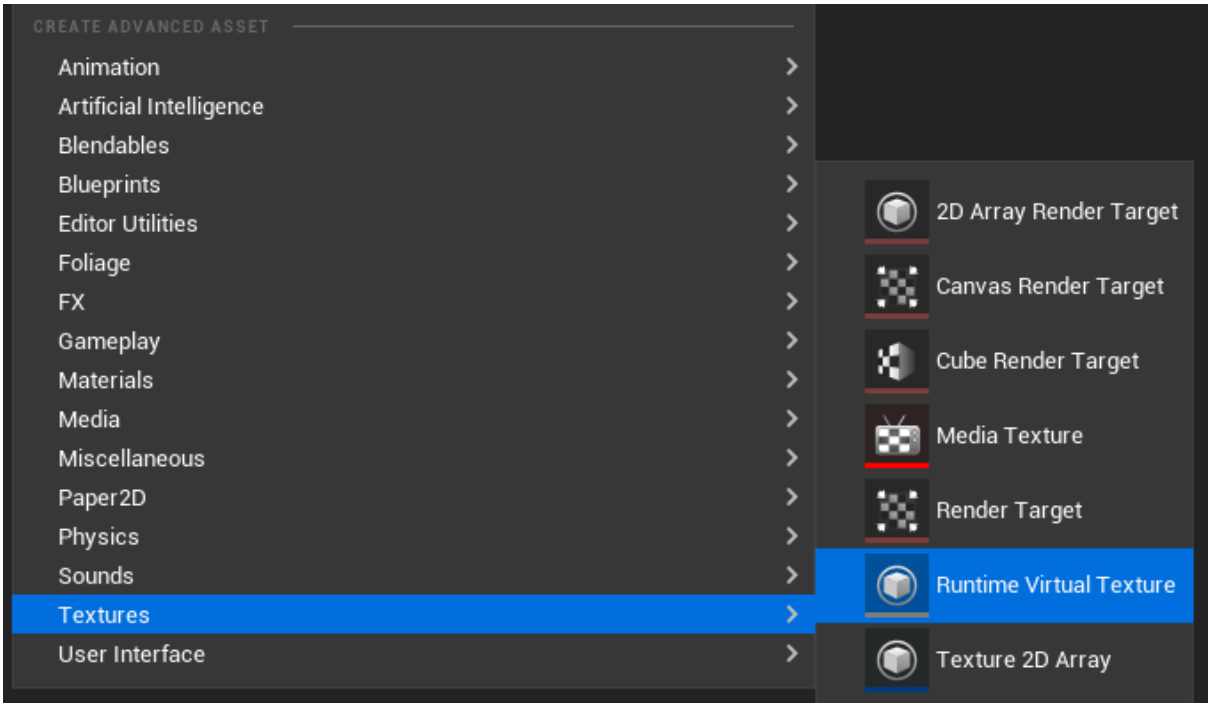
4. Restart editor.

After that demo map **ConstructionSite_02_Demo_P** will be run properly. If you want this functionality on your own map follow next steps:

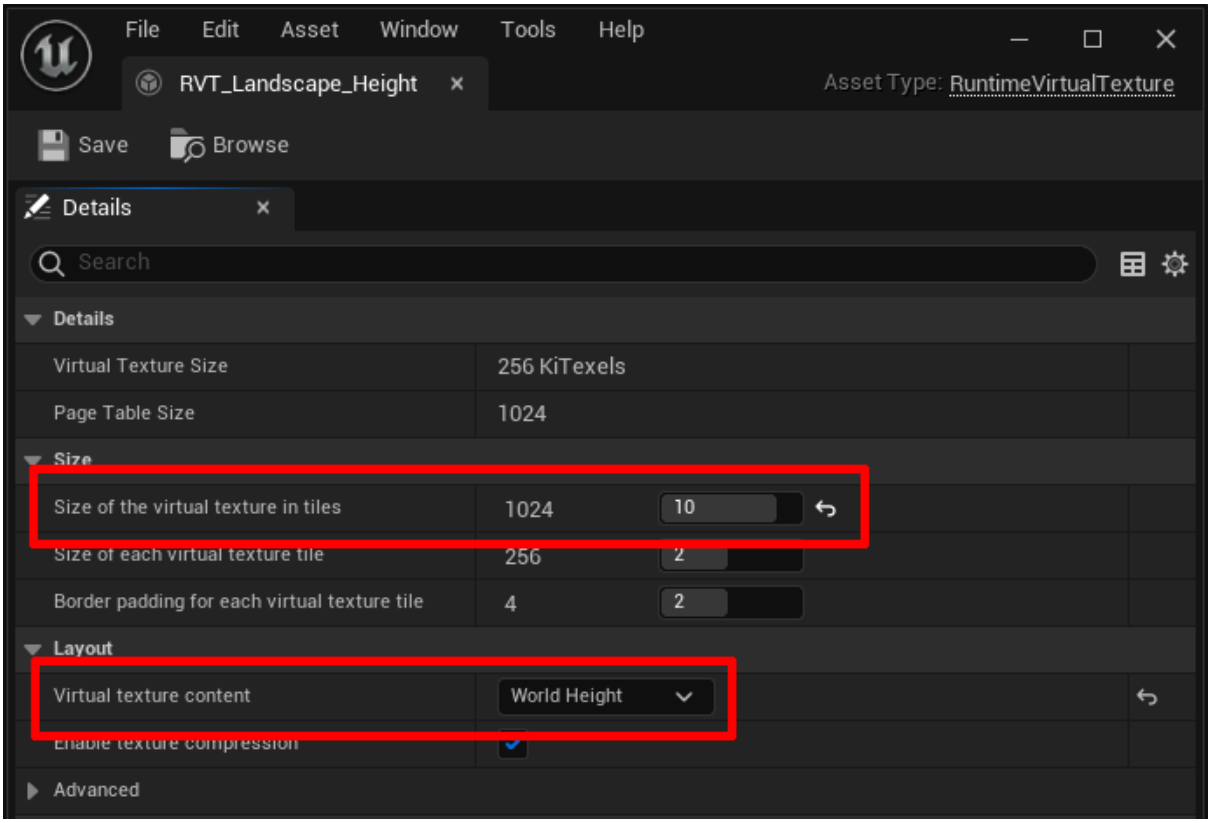
5. Open landscape material **m_Landscape_01_01**, add **Runtime Virtual Texture Output node,** and setup material like in the picture.

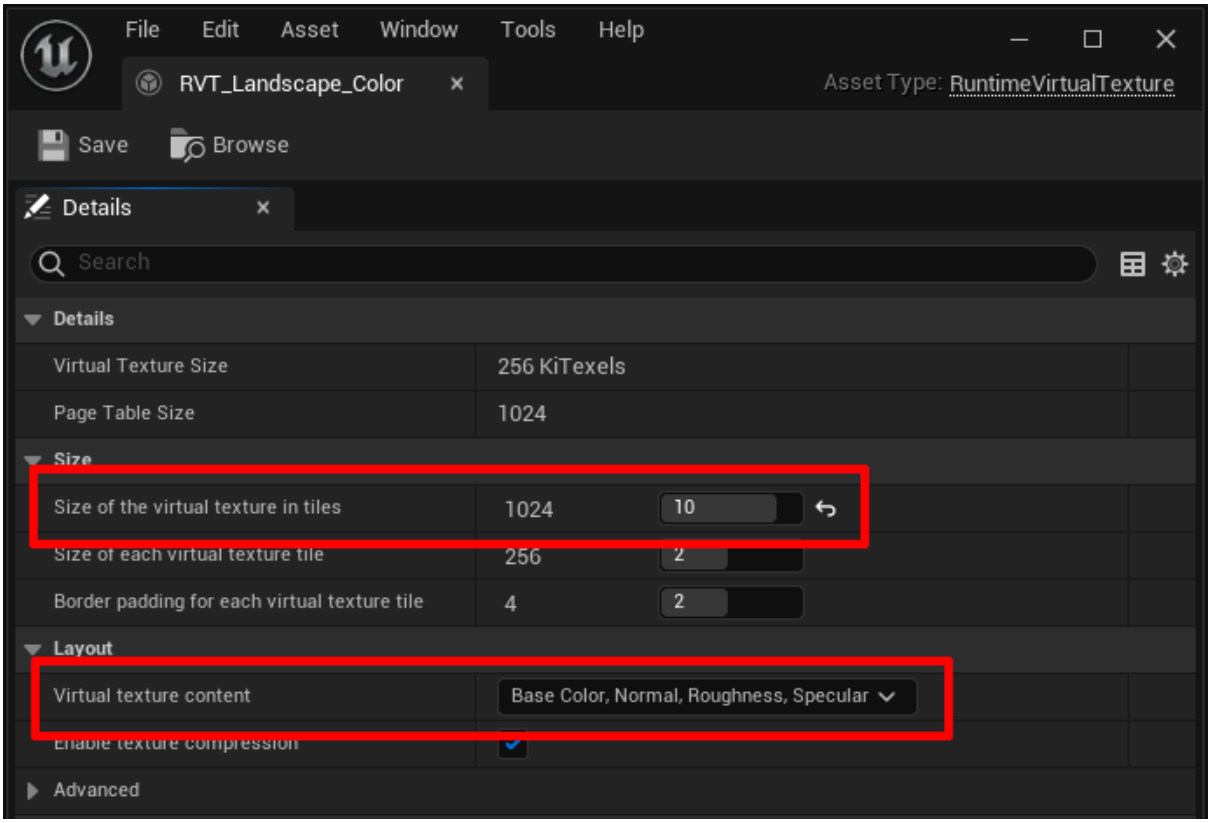


6. In Content Browser create two **Runtime Virtual Texture**. Name the first texture **RVT_Landscape_Height** and name the second **RVT_Landscape_Color**.

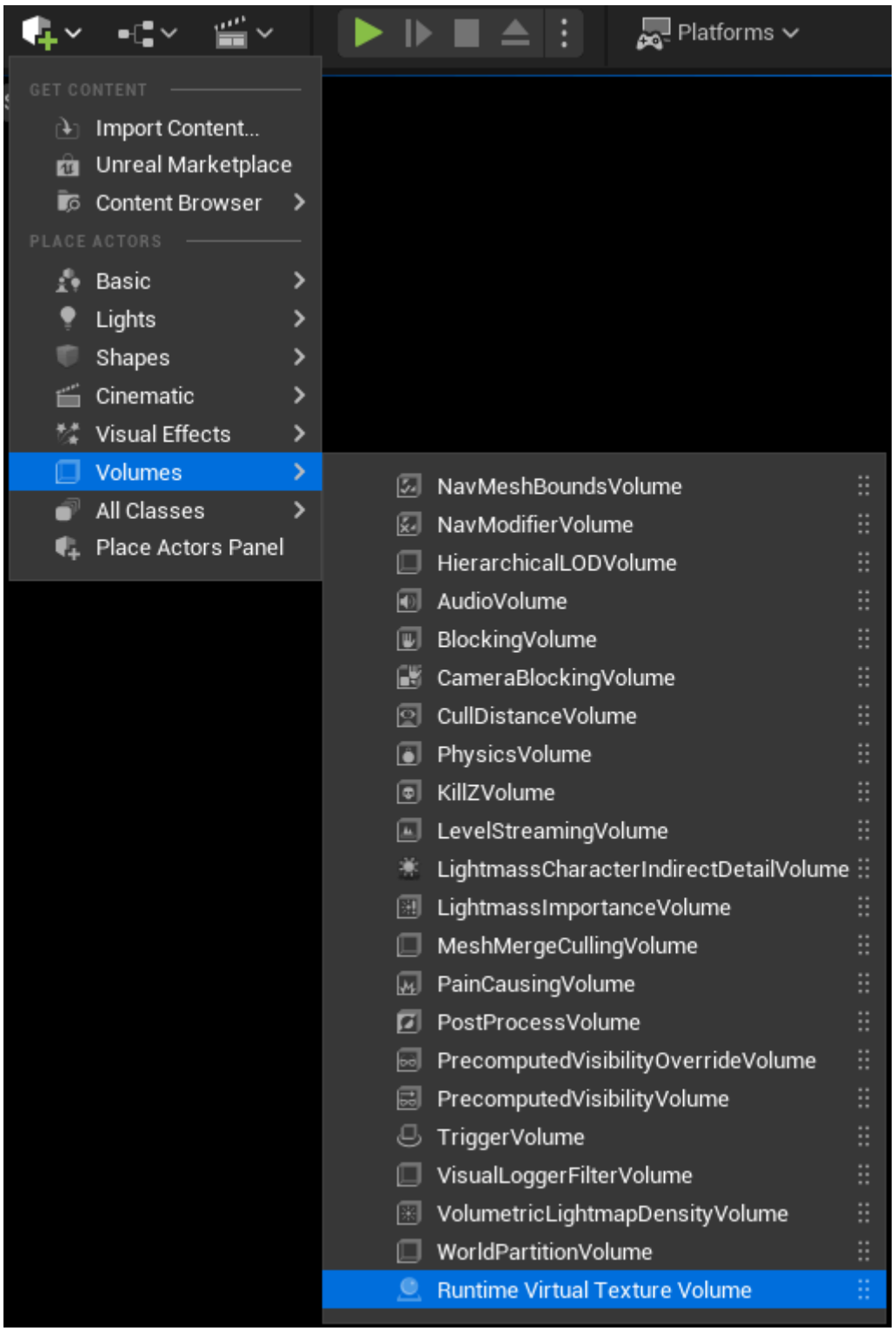


7. Open **RVT_Landscape_Height/RVT_Landscape_Color** and set them as shown in the pictures.





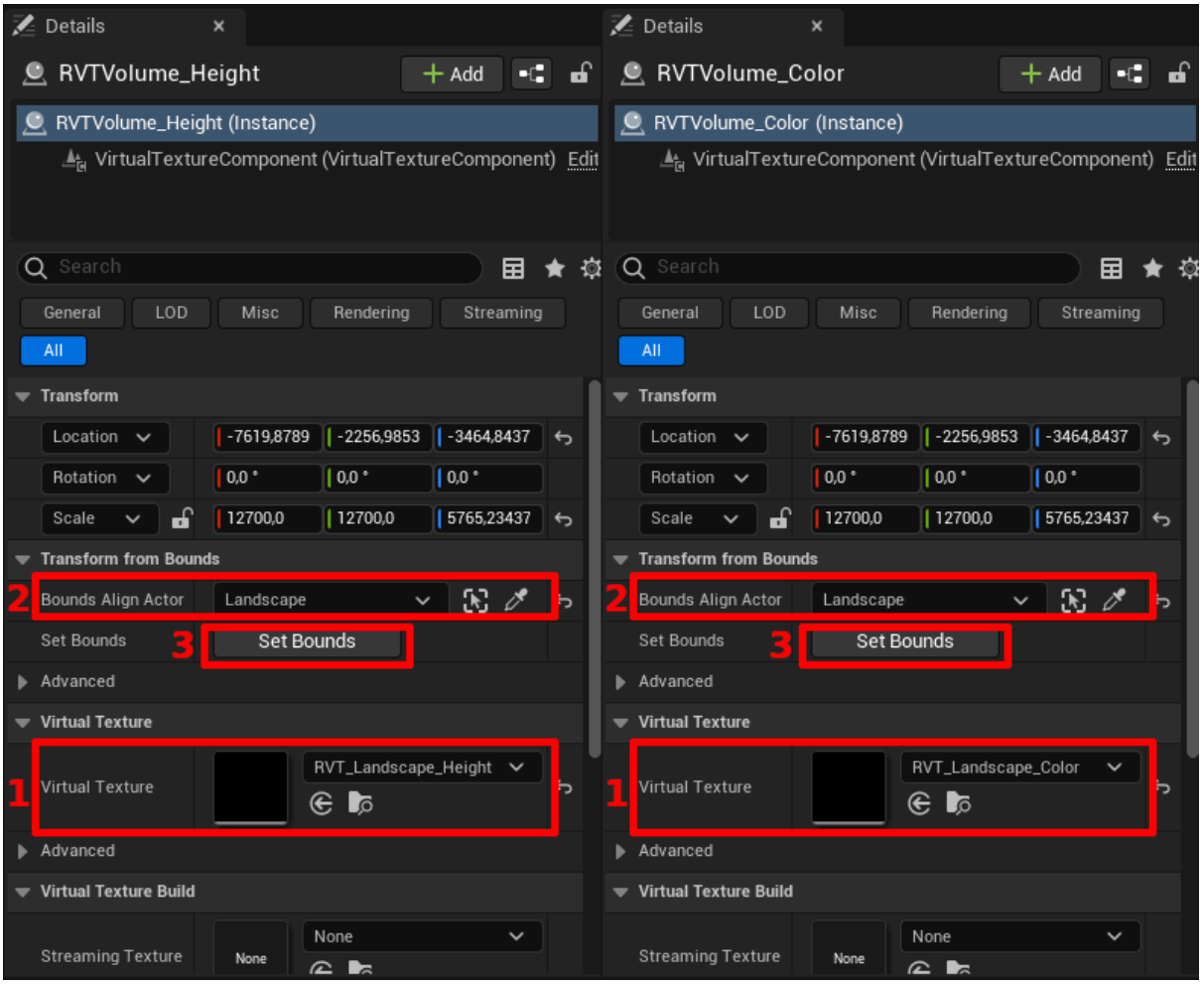
8. Add to level two **Runtime Virtual Texture Volume** from Volumes menu. Name the first volume **RVTVolume_Height** and name the second **RVTVolume_Color**.



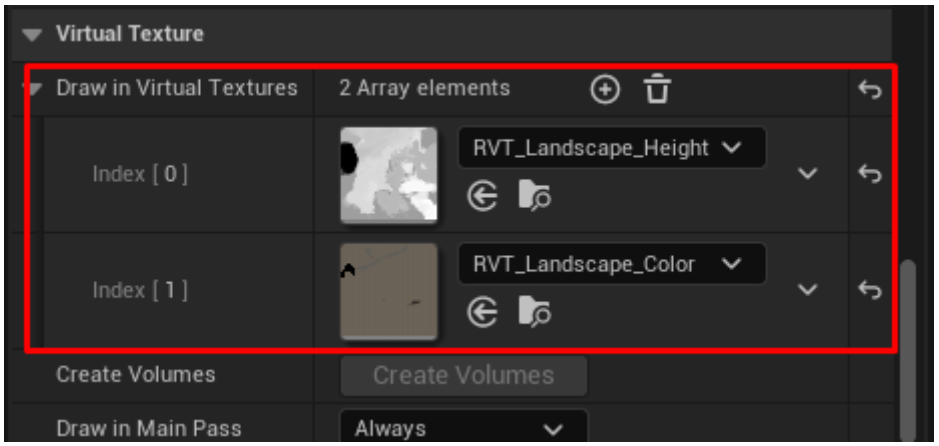
9. Select the **RVTVolume_Height**, assign the **RVT_Landscape_Height** texture to **Virtual Texture** property (1), assign the landscape actor from level to **Bounds Align Actor** (2) and click **Set Bounds** button (3).

   Then select the **RVTVolume_Color**, assign the **RVT_Landscape_Color** texture to

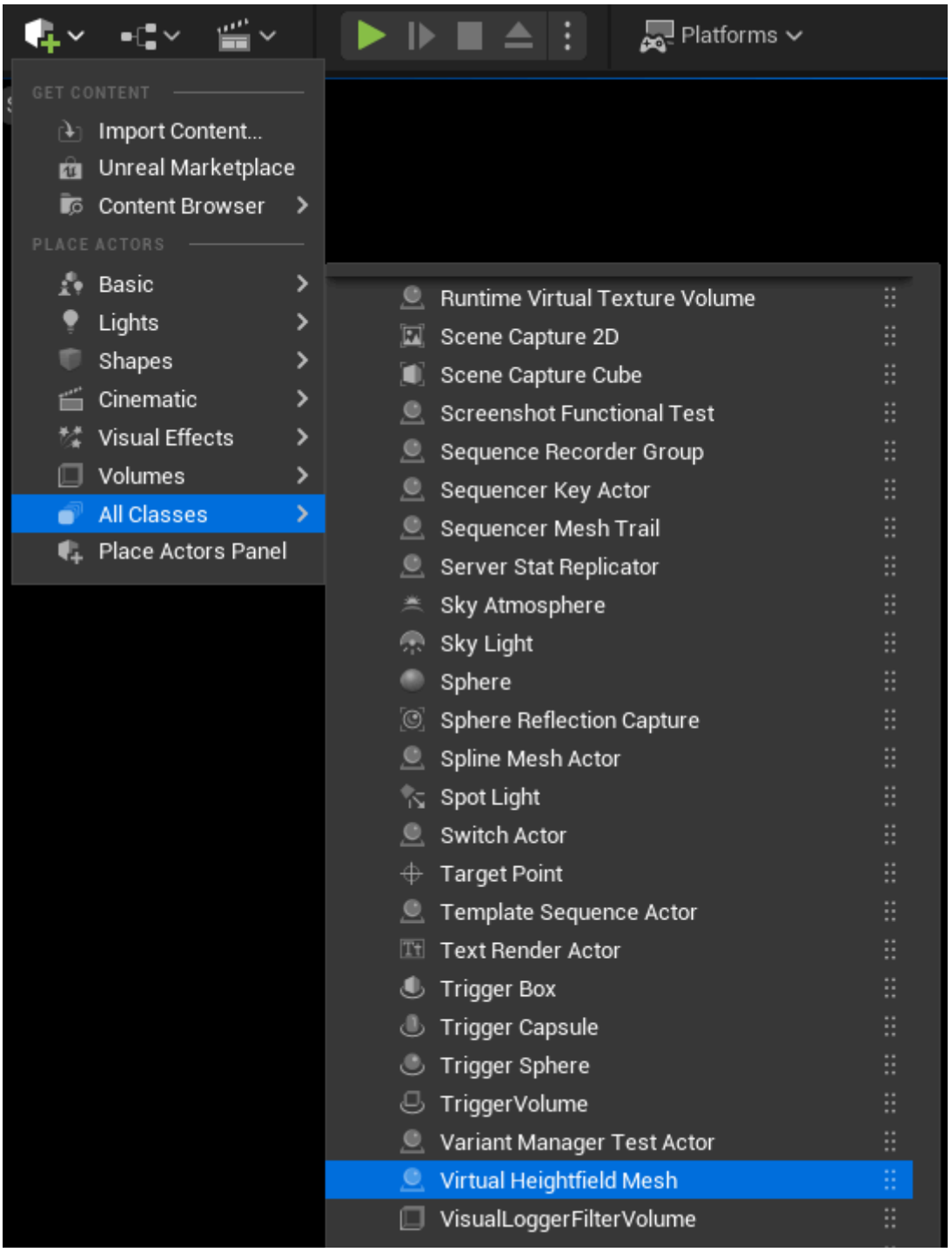
Virtual Texture property (1), assign the landscape actor from level to **Bounds Align Actor** (2) and click the **Set Bounds** button (3).



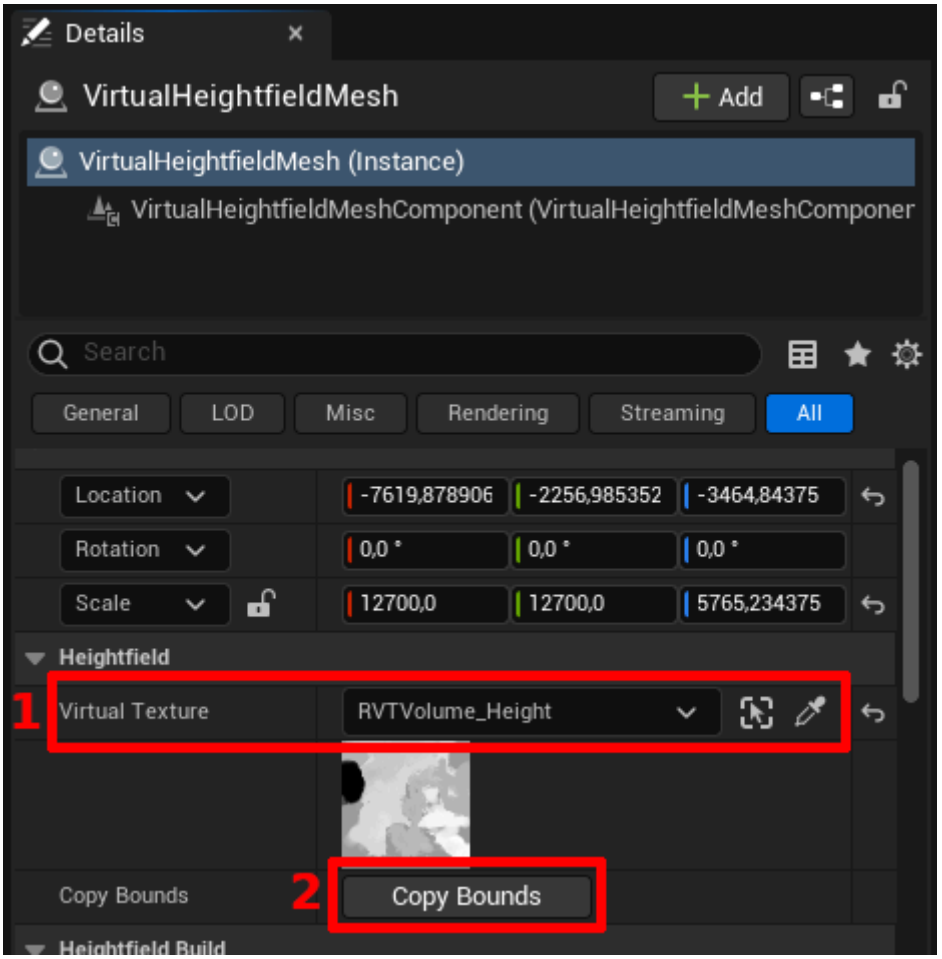
10. Select the landscape actor at level and assign **VT_Landscape_Height** and **RVT_Landscape_Color** textures to **Draw in Virtual Textures** under the Virtual Texture tab like in the picture.



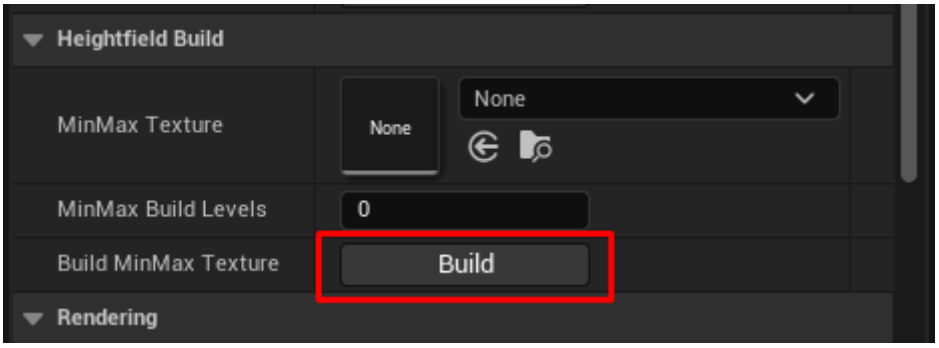
11. Add to level the **Virtual Heightfield Mesh** from All Classes.



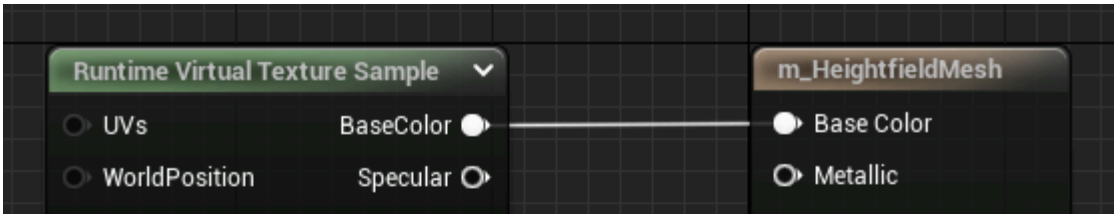
12. Select the **Virtual Heightfield Mesh**, assign **RVTVolume_Height** volume to **Virtual Texture** (1) and click the **Copy Bounds** button (2).

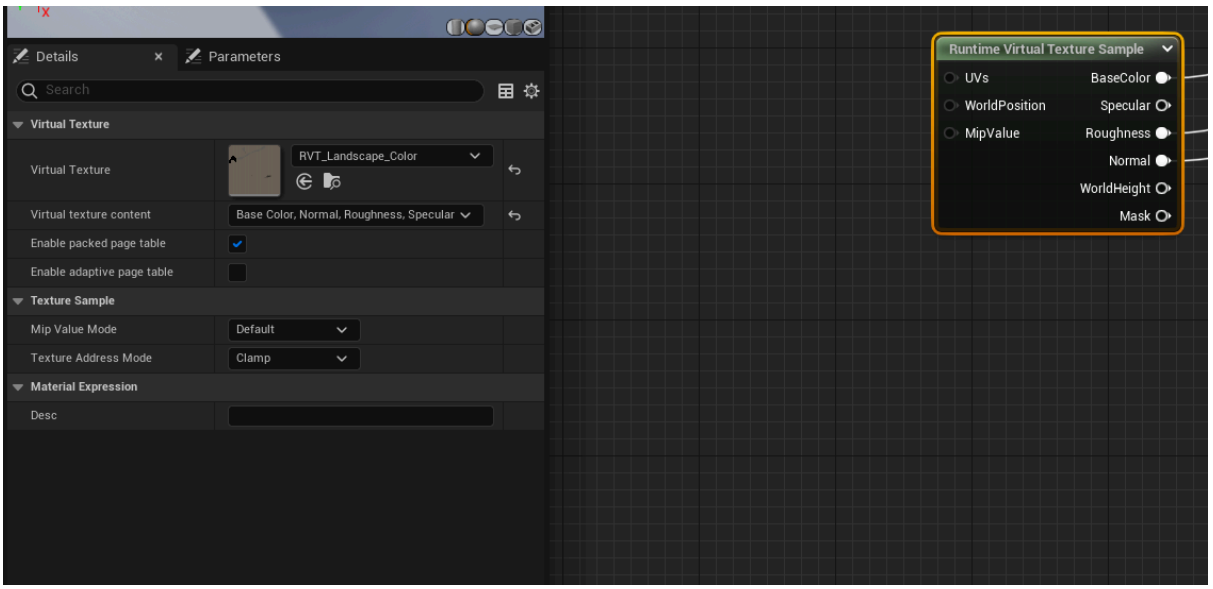


13. Build MinMax Texture by clicking the **Build** button under Heightfield Build. Name and save this texture in the project. This may take some time up to 30 minutes. When the build is over, don't forget to save the texture.

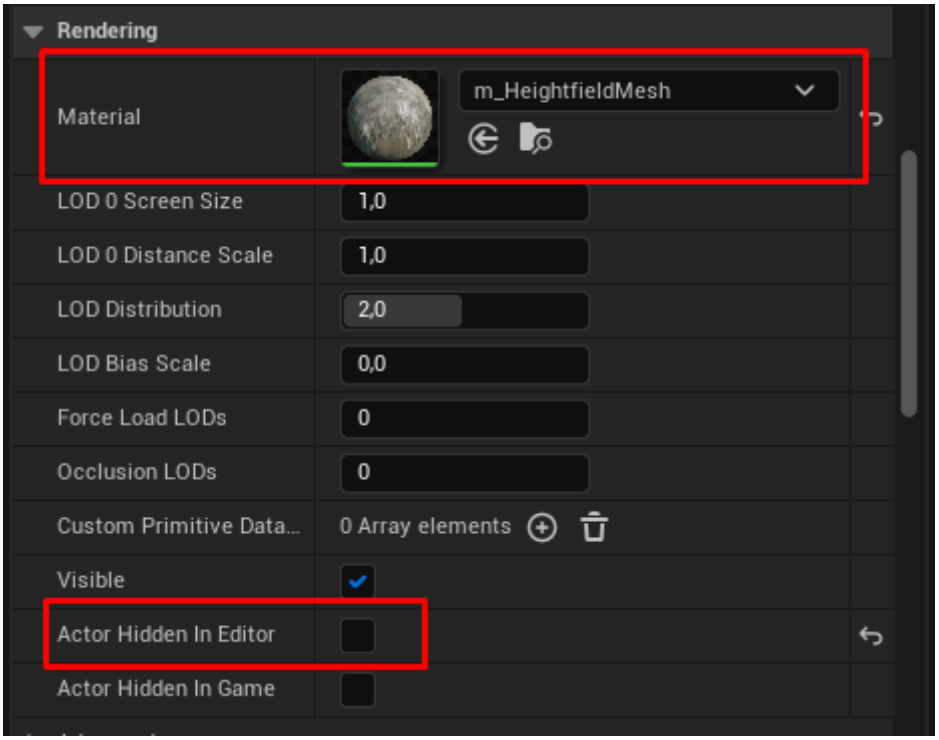


14. Create a new material for the **Virtual Heightfield Mesh** and setup like in the picture. In the **Runtime Virtual Texture Sample** node assign the **RVT_Landscape_Color** texture from the content.





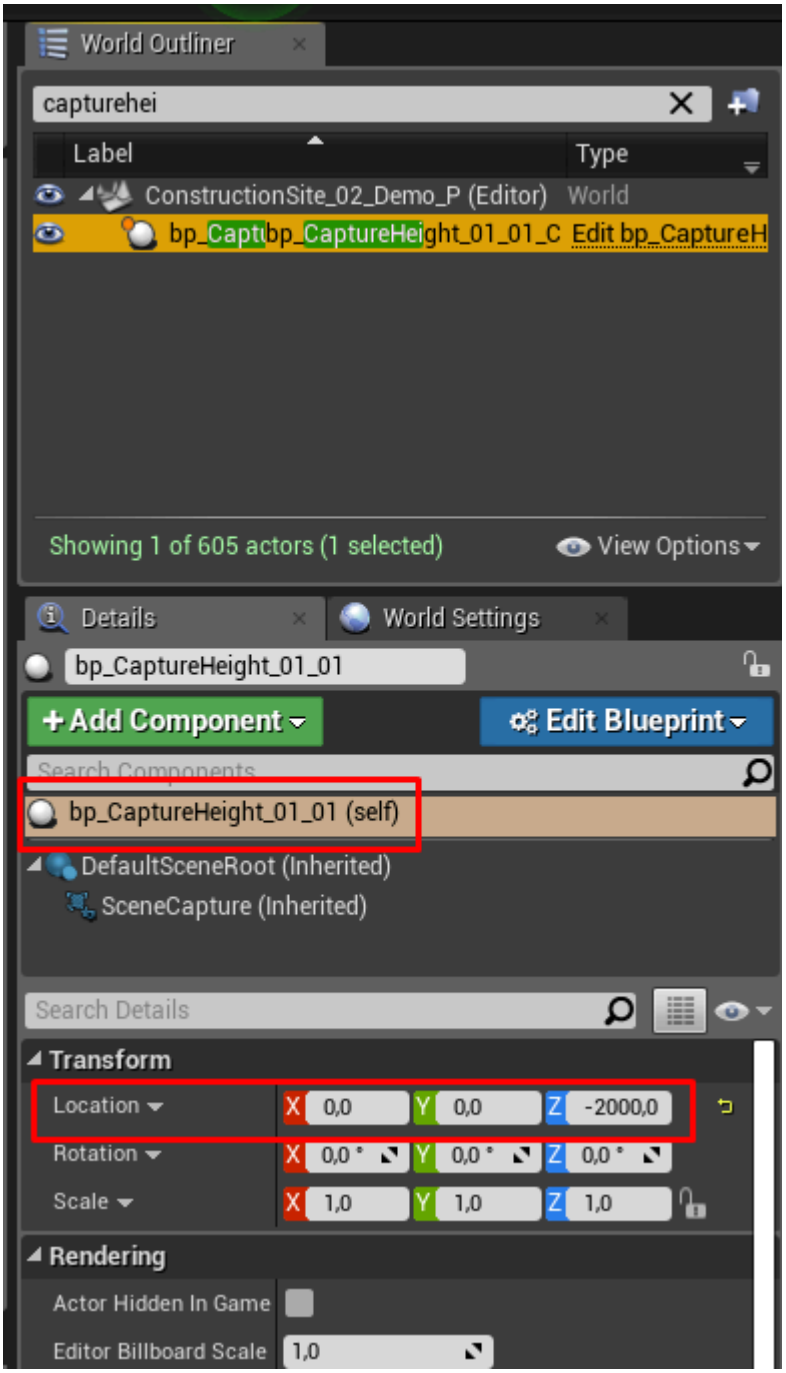
15. Select again the **Virtual Heightfield Mesh** and assign created new material. Then disabled **Actor Hidden in Editor**. And hide original landscape.



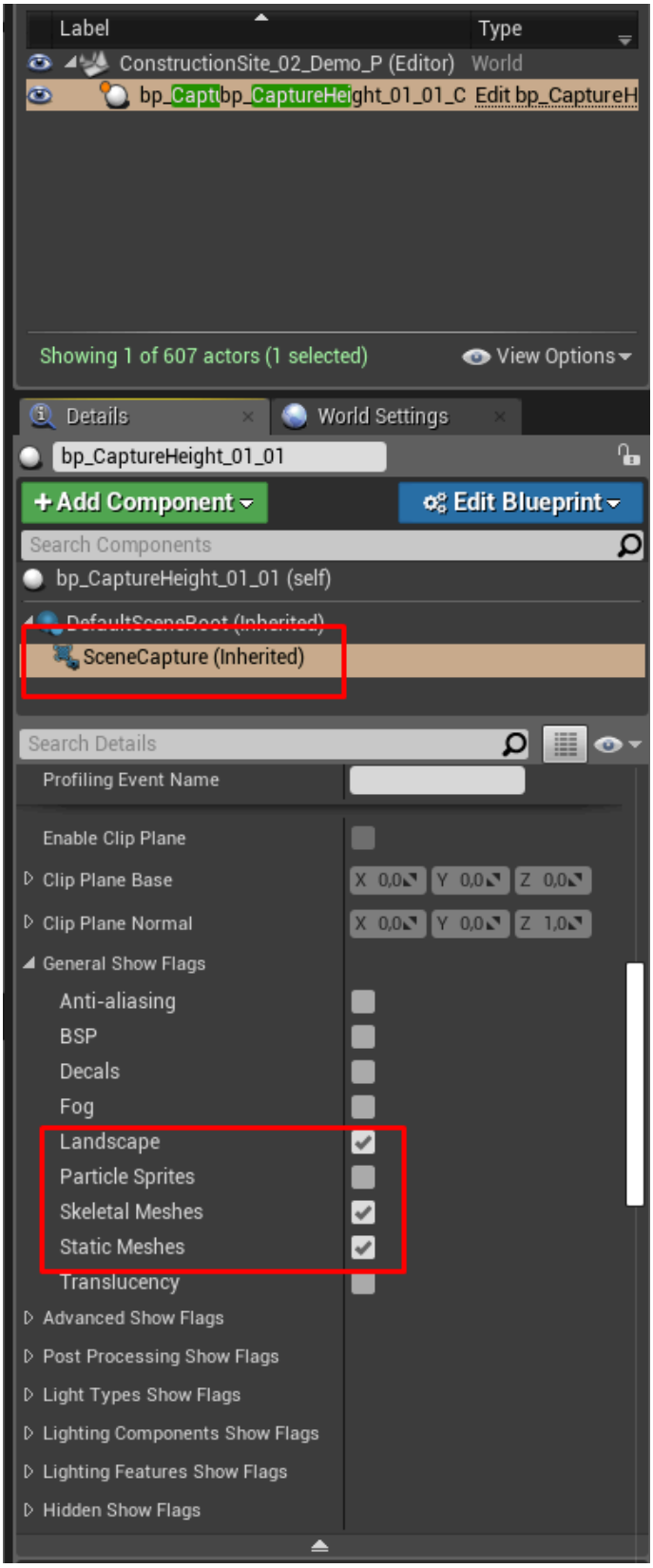
16. Now you have a high detail landscape in the Unreal Engine 5. Repeat these steps for each landscape actor on the level.

Landscape Deformation with Virtual Heightfield Mesh

1. Landscape needs to use Virtual Heighfield Mesh
2. Put on level actor bp_CaptureHeigh_01_01 (under the landscape for example -2000 Z (like in the Demo scene)



3. Be sure Scene Capture component in this blueprint have proper flag turn on (Lanscape, Skeletal Meshes and/or static meshes



4. On every mesh which you want ot react with landscape you should turn on CustomDepth Pass(rendering section)