



## ***MEDIEVAL FARMSTEAD BUILDER DOCUMENTATION***

Hello,

After a long production process, we are delighted to present the Medieval Farmstead Builder Asset package to you! This documentation has been prepared to provide you with all the details on the correct use of the package. If you cannot find answers to your questions through this documentation, please feel free to contact us. Assisting you in your game development process brings us joy.

Best regards, CropCraft Studios Team 😊

[cropcraftstudios@gmail.com](mailto:cropcraftstudios@gmail.com)

# User Guide

## Modular Structures

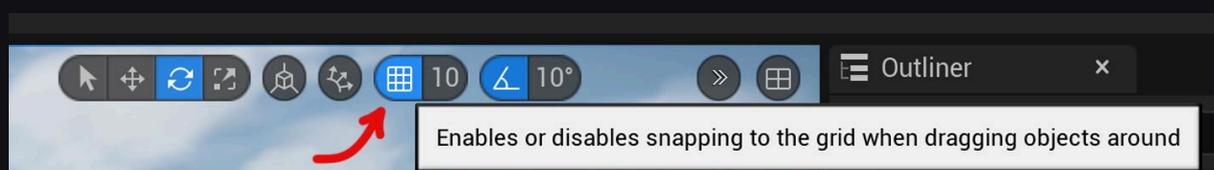
We took care to create a modular system that is as simple as possible. The parts are divided into 4 within themselves: Ground floor parts, 1st floor parts, attic floor parts and roof parts.

**Ground Floor Parts:** There are two different ground floor options, Stone and Plaster. These parts, which are used in the installation of the ground floor, are structurally divided into two: a corner wall and straight walls.

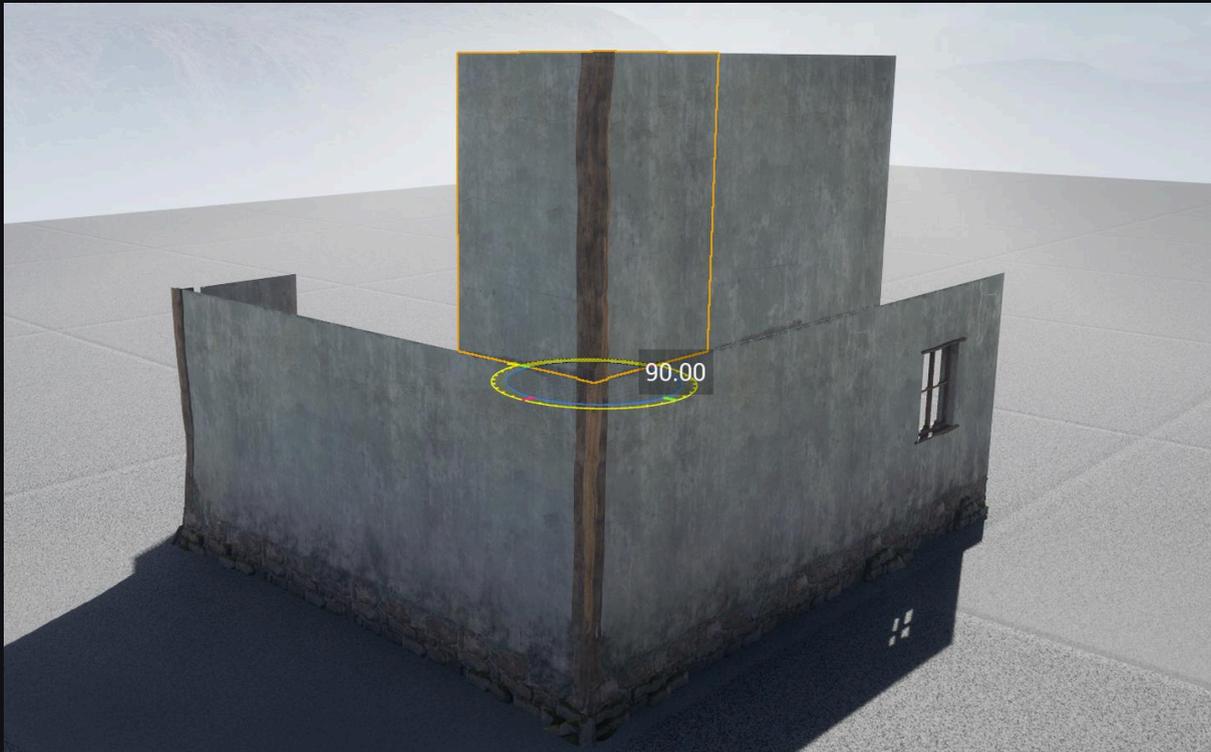


There are many variations such as windows, doors, gates among the straight walls. You can use the corner wall at any corner by rotating it.

**Note:** All modular parts are compatible with the Grid turned on.



**1st Floor Parts:** There are two different 1st floor options, Plaster and Wood. Just like the ground floor, it consists of a corner wall and many straight walls. You can use the corner wall at any corner by rotating it. Also, if you



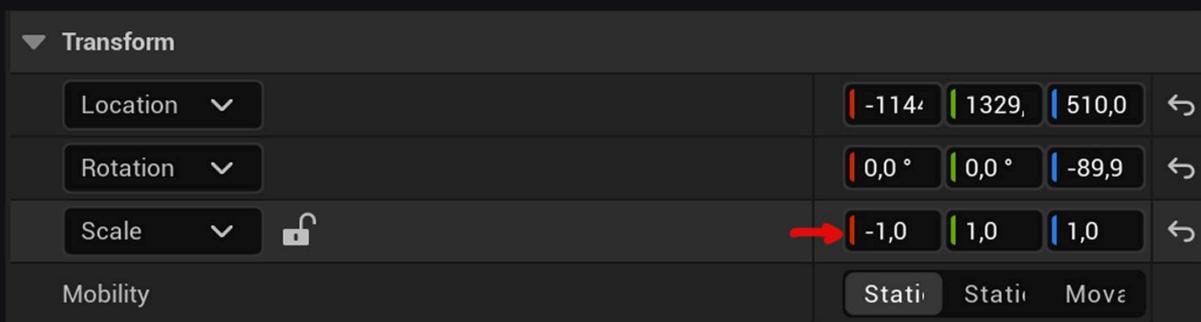
want to build a single-story building, you can skip using the parts of this floor and proceed directly to roof construction.

**Note:** Roof parts are one of the most complex and difficult to understand parts of the modular system.

**Attic Floor Parts and Roof Parts:** Although these two groups appear different from each other, they have to work together. They are designed as complementary to each other. In roof parts, "SM\_Roof\_Edge" is the piece to be placed at the corners when you start building the roof after the walls, and one corner of it is wall-shaped. After placing the "SM\_Roof\_Edge" piece at all corners of the building, the gaps in between are filled with attic floor walls.



To use the SM\_Roof\_Edge piece on opposite corners, we need to enter a value of -1 in the X direction in the Scale section of the Static Mesh Details panel. The reason for this is to optimize the instance count throughout the building by using the same model at all corners instead of using an additional different corner model.



Roof corners are the most numerous similar pieces. To avoid confusion, we need to divide these pieces into two main pieces. These are "Roof\_Edge" and "Roof\_Edge\_SecondFloor".

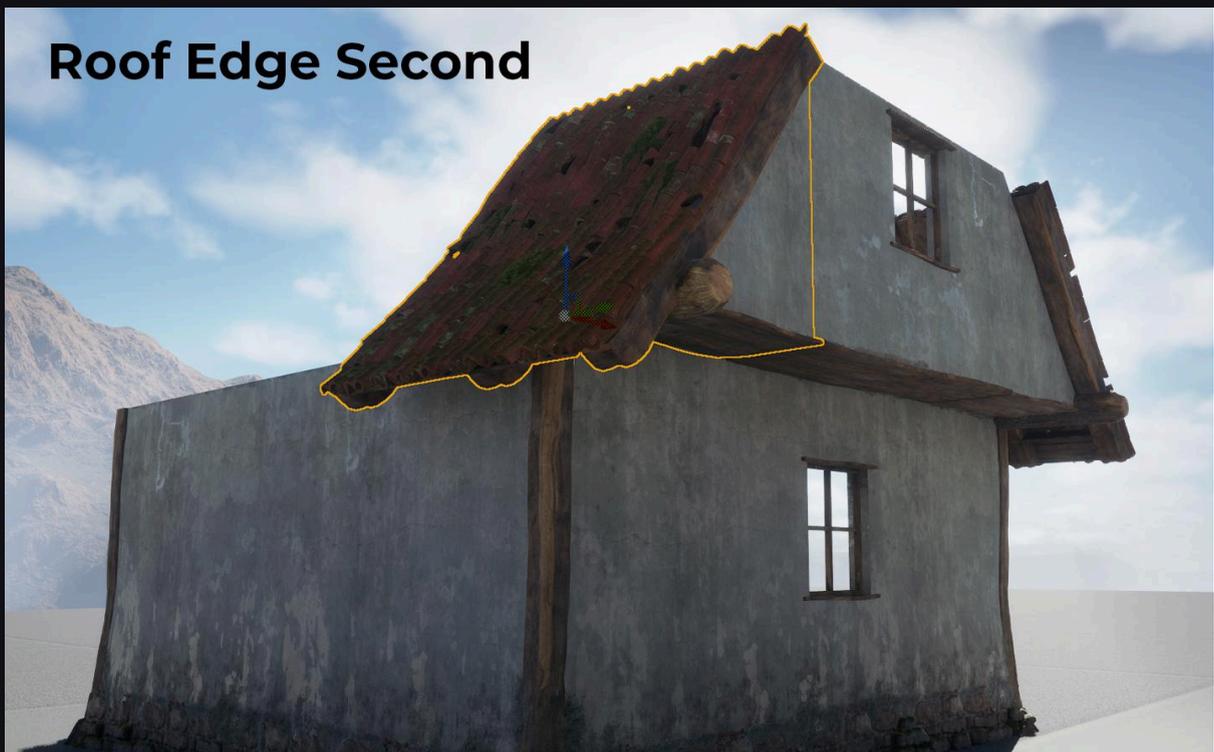
"Roof\_Edge" pieces are the first corner roof pieces that come after the walls. "Roof\_Edge\_SecondFloor", on the other hand, are the 2nd corner pieces added to complete the roof depending on the width of the building, and they come on top of the first corner pieces placed.

In addition, both "Roof\_Edge" and "Roof\_Edge\_SecondFloor" pieces have alternative pieces named (Second). These pieces allow for a slight expansion in the attic compared to the others. These pieces can be identified by adding "Second" to the normal names. For example: "Roof\_Edge\_Second" or "Roof\_Edge\_SecondFloor\_Second".

## Roof Edge SecondFloor



## Roof Edge Second



After completing the corner pieces, there are "Roof 3m" pieces for the intermediate sections. These pieces are simpler than the corners. "Roof\_3m\_Edge" pieces are used for the first floor, while "Roof\_3m\_SecondFloor" is used to complete the second floors.



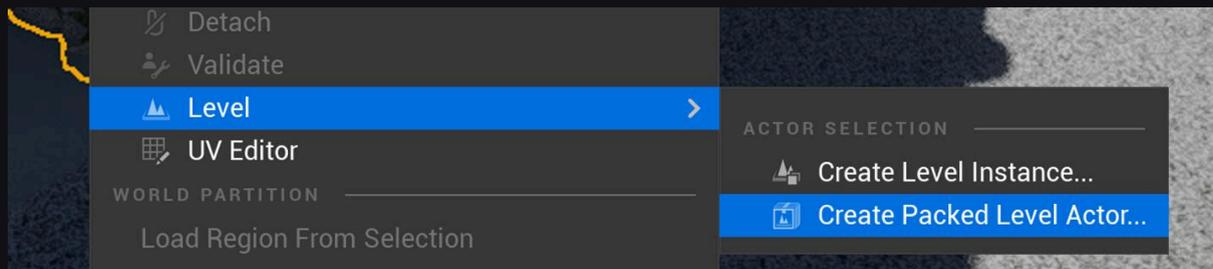
**Roof Top:** These pieces are the simplest roof completion pieces. They allow you to make the final assembly of the roof.

**Floor and Ceiling:** Floors and ceilings are common parts. You can easily create the floors of buildings by joining them side by side, and you can create the ceiling and the floor of the upper floor by copying the same parts. (**Note:** This applies when the "SM\_GroundWood" mesh is used. "GroundHay" and "GroundStone" are only for creating the ground floor.)

If this explanation did not help you or if you have any points where you are stuck, please open the level files of the buildings by following the path "Blueprints/Buildings" in the package and examine all the buildings.

## Packed Level Actor

Packed Level Actor: You can create your buildings on any map. The only thing you need to pay attention to is to select all the parts you used after making sure you have finished the building, and right-click and follow these steps in order: Level/Create Level Packed Actor. Then, in the screen that appears, simply select "Pivot Type: Center Min Z". In the subsequent save screen, you can save the Level and Blueprint (BPP) actor wherever you want.



You can later modify all the packed actors you have saved through their levels, and add indoor or outdoor extensions. This allows you to make your buildings more dynamic and customizable. For example, you can add new rooms to the interior of a building, change its exterior, or add extensions such as a garden around it.

**Note:** In some cases, the pivot point of the Packed Actor may not be set correctly. In such cases, you can open the level of the packed actor, select all the parts, and move them to the zero point. This change will also cause the pivot point to change.

## Procedural Content Generation (PCG)

There are different PCGs for all level design actors in the package, and each PCG has a Blueprint (BP) file. Usage is done through BP files.

**BP\_Main:** All major controls are gathered in this BP. You can access Wind, Village, and Forest settings from here.

- Wind: You can determine the direction and strength of the wind through the parameters here. When setting the Direction, numbers between 0 and 1 determine the 360-degree direction.
- Forest: You can turn the forest on and off from this section. (It is recommended to keep it closed while working on level design for performance reasons.) You can also set the Tree Density. The closer the

number you set in the density setting is to 1, the lower the density. You can also adjust the Rock and Bushes group densities from here.

- Village Settings:
  - House Density setting; while doing it, the closer the number is to 10,000, the fewer houses there will be.
  - Village Roads Loop: Checking this box disables the loop feature of the road surrounding the Spline in the Village PCCG.
  - Village Inside Fields On/Off: This option allows you to turn the fields inside the village on and off.
  - Houses On/Off: This option allows you to turn village buildings on and off. (Note: Due to a bug in Unreal Engine 5.5, buildings can sometimes be positioned at a different point from their location. You can fix this by turning the buildings on and off.)
  - Change Houses Location: This option allows houses, fences surrounding houses, and fences around the road to be positioned on the outer circle of the spline. (Note: In Village structures that you will determine by drawing a spline via the Modeling tool, this option may sometimes work in reverse.)

**BP\_Village:** There is only a section on it where you can add the Static Mesh you want to use in the existing field. Apart from that, you can use it by giving the shape you want with the spline. In other words, with this Blueprint, you can customize the appearance of the fields in your village and determine the overall shape of the village using spline tools.

**BP\_Fields:** It is the same as BP\_Village. You can add the crop you want and give it the shape you want with the spline. It is the first in the order of priorities and deletes everything else at the point you put it.

**BP\_Road:** With the settings on this Blueprint, you can turn the fences on the sides of the road on and off, turn the foliage that appears on the side of the road on and off, and choose whether the road is dirt or stone. This allows you to customize the appearance and properties of your roads.

**BP\_River and BP\_Market:** There are no settings on these two blueprints. You can only use them as you wish with the spline. This means that the shape and size of rivers and market places are completely under your control. You can create customized rivers and market places in your game world using these blueprints.

**Location Volumes:** Location Volumes are at the top of the priority list. All objects at the points where you use these volumes are deleted (except Spline Roads). This provides great convenience for you during level design.

**Data Assets:** There are two different Data Assets in the package: DA\_MarketStalls and DA\_VillageHouses. You can easily add the buildings or market stalls you created to the PCGs using these data assets.

**Landscape Material and PCG Relationship:** Some of the PCGs work together with Landscape Layers. For example:

- **Forest:** Forest materials (trees, bushes, etc.) appear in the area where the "Forest" Layer is painted. In other words, when you apply the "Forest" Layer to the terrain, the PCG system automatically places forest elements in that area.
- **Village:** The "Grass" Layer that you will apply in the area within the BP\_Village Spline area allows Alder trees to appear sparsely in some areas. This helps the village to have a natural look.

**Important Note:** If you do not want anything else (trees, vegetation, etc.) to appear in the areas where buildings or market stalls are located, you can use the "Fields" or "Mud" Layers on these grounds. These layers instruct the PCG system not to create any additional items in these areas.

## Technical Details & Recommended Usage

- Suitable for creating realistic medieval-themed games.
- The package content is designed to provide all the content needed for cozy games.
- Creating large forests, village areas, and fields is very easy.
- It is recommended to use it together with the "Ultimate Farming" series.
- Interior and exterior designs are completely shaped by your imagination. In addition, the package includes ready-to-use buildings, market stalls, and various prop collections.

Note: The recommended usage can be subject to change according to your own creativity and objectives during the game development process.

## Triangle counts

- **Lowest triangle count: 2 (SM\_WindowGlass)**
- **Highest triangle count: 2.464,012 (SM\_Roof\_Window\_Thatched)**
- **Average triangle count: 16000**

Note: All objects in the package utilize Nanite, When using this package with Unreal Engine 5.2 and above, it is strongly recommended to utilize the Nanite and Lumen systems.

<https://docs.unrealengine.com/5.0/en-US/lumen-global-illumination-and-reflections-in-unreal-engine/>

Important Note: To ensure the package works correctly, be sure to download the recommended "Project Settings" and enable the "Procedural Content Generation" plugin.

## Performance

System : Amd Ryzen 7 5800x - 16gb Ram - Nvidia GeForce RTX 3080 - Epic Settings

PCG\_Village Map 55 - 60 fps

Showcase Map 65 - 70 fps

to full screen Editor Mode

System : Amd Ryzen 9 7950x - 32gb Ram - Nvidia GeForce RTX 4090 - Epic Settings

Old\_Town Map 70 - 90 fps

Showcase Map 85-120 fps

to full screen Editor Mode

Note: The FPS rates mentioned above are based on the maps prepared for the purpose of accurately presenting the package to you