

Modeling Mode

Contents

1	Core Concept: The In-Engine Workflow	1
2	Primary Toolset: Cube Grid	1
2.1	Selection and Manipulation	1
2.2	Mathematical Precision	2
3	Constructing the "Room"	2
3.1	Wall and Window Formation	2
3.2	Roof Geometry (PolyEdit)	2
4	Staircase Design and Refinement	3
4.1	The Staggered Method	3
4.2	PolyGroup Optimization	3
5	Technical Summary Table	3
6	Workflow Best Practices	4
7	Collision Generation and Setup	4
7.1	Standard Collision Methods	4
7.2	Optimization: The "Collision" Tool in Modeling Mode	4
7.3	Physics Settings Table	4

1 Core Concept: The In-Engine Workflow

We can "Model from Scratch" workflow using the **Modeling Tools Editor Mode**. This allows for a non-destructive, rapid iteration process within the viewport, eliminating the need to export/import FBX files from external DCCs (Digital Content Creation) like Blender.

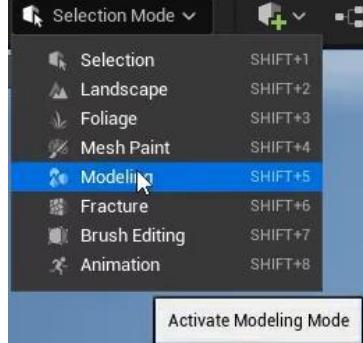


Figure 1: Modeling Mode – Shift+5

2 Primary Toolset: Cube Grid

The **Cube Grid** is the foundational tool for blocking out architectural volumes. It operates on a 2D selection grid that projected into 3D space.

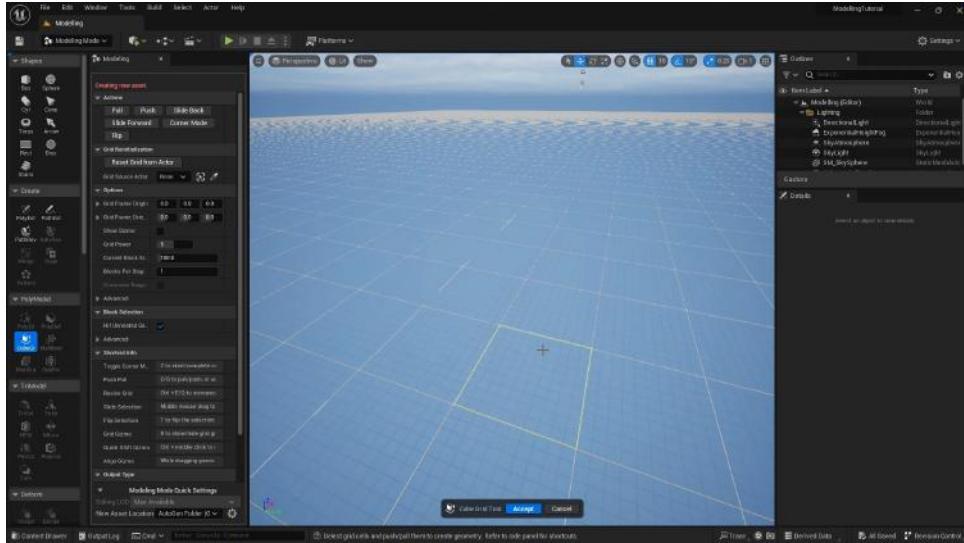


Figure 2: Cube Grid Tool

2.1 Selection and Manipulation

- **Grid Selection:** Click and drag on the grid to define a face.
- **Extrusion (E):** Moves the selected face outward by one grid increment.
- **Depression/Subtraction (Q):** Pushes the face inward, effectively deleting geometry or creating voids.
- **Corner Mode:** Allows for the selection of individual corners of the grid to create slanted or diagonal surfaces (trapezoidal volumes).

2.2 Mathematical Precision

The **Grid Size** parameter is calculated as per:

$$\text{Step Size} = 2^n \times 10 \text{ units} \quad (1)$$

Common increments used:

- **50 cm:** Used for thick structural walls and floors.
- **100 cm:** Standard unit for large-scale blocking.
- **10 cm:** Detail work for window frames or stair steps.

3 Constructing the "Room"

Let's create a room:

3.1 Wall and Window Formation

The Footprint: Laying a flat plane using Cube Grid.

Wall Extrusion: Selecting the outer perimeter of the floor and using *E* to pull up walls.

Boolean-style Windows: By selecting a 1×2 grid area on a vertical wall and using *Q*, a hole is punched through. This is "manifold-safe," meaning Unreal automatically handles the side-faces (caps) of the hole.

3.2 Roof Geometry (PolyEdit)

To transition from a flat ceiling to a pitched roof:

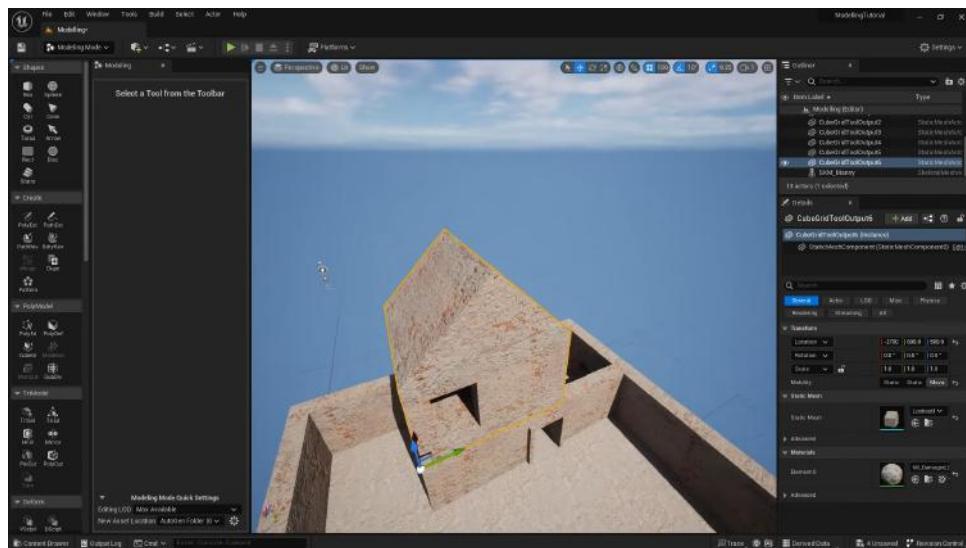


Figure 3: Roof

1. Switch to the **Corner Mode** tool in the Cube Grid.
2. We can select the corner vertices using this tool.
3. Select the ridge line (the center edge) of the roof mesh.
4. After selecting just press *Q* to make sloped rooftop.

4 Staircase Design and Refinement

We can custom stairs to ensure character "playability."

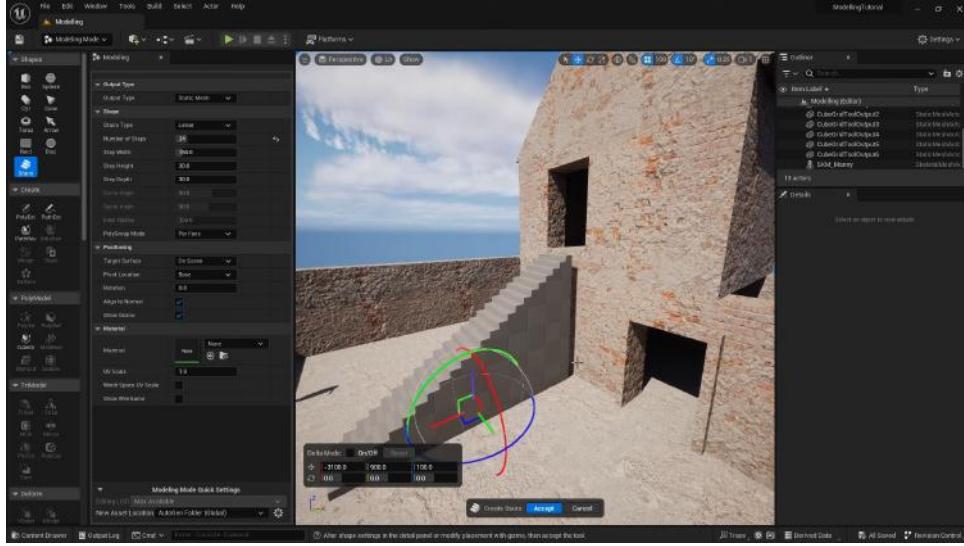


Figure 4: Stairs are present in the Basic Shapes Component.

4.1 The Staggered Method

Instead of a single slope, the Cube Grid is used to "draw" the steps:

- **Rise:** 20 cm (Vertical movement).
- **Run:** 30–40 cm (Horizontal movement).

4.2 PolyGroup Optimization

After the stairs are blocked out, the **PolyGroup Edit** tool is used to clean the geometry.

- **Merge Faces:** Multiple small faces on the side of the stairs are merged into a single PolyGroup.
- **Beveling:** Selecting edges to add a chamfer, making the structure look less "digitally sharp."

5 Technical Summary Table

Tool	Primary Use Case	Key Setting
Cube Grid	Initial Blocking, Walls, Floors	Grid Power (Z key)
PolyEdit	Moving Vertices/Edges, Slanted Roofs	Selection Filter
PolyGroup Edit	Cleaning up "Blocky" geometry	Group Topology
Bake Transform	Freezing scale/rotation after modeling	Pivot Location

Table 1: Overview of Tool Applications in UE5 Modeling Mode

6 Workflow Best Practices

- **Face Orientation:** Always check that normals are facing outward (Front-facing) to avoid rendering artifacts in Lumen.
- **Pivot Alignment:** Use the *Edit Pivot* tool after modeling to ensure the pivot is at the bottom corner (0,0,0 of the mesh) for easier placement later.
- **Material IDs:** Assign different PolyGroups to different sections (e.g., Roof vs. Walls) to allow for multiple material slots on a single mesh.

7 Collision Generation and Setup

Once a mesh is modeled "from scratch," it exists as a **Dynamic Mesh** or **Static Asset** that requires a collision hull for character interaction.

7.1 Standard Collision Methods

Ways to handle collision within the **Attributes** or **Physics** category of the Modeling Toolset:

1. **Collision Tool (Simple):** Used for basic shapes (Boxes, Spheres, Capsules).
 - *Best for:* Simple pillars or flat floor slabs.
 - *Pros:* Extremely cheap for the physics engine to calculate.
2. **Convex Hull (Hulls):** Generates a simplified "shrink-wrap" around the mesh.
 - *Setting:* `Max Hulls` and `Max Vertices` per hull.
 - *Best for:* Complex organic shapes or irregular walls.
3. **Mesh-Based Collision (Complex):** Uses the actual polygons of the model as the collision surface.
 - *Setting:* Set `Collision Complexity` to "Use Complex Collision as Simple".
 - *Best for:* Staircases and window cutouts where the player needs to pass through specific holes.

7.2 Optimization: The "Collision" Tool in Modeling Mode

Within the **Modeling Mode** (`Shift + 5`), there is a dedicated **Collision** tool.

- **Input:** Select the mesh in the viewport and click *Collision*.
- **Method - V-HACD:** This is the "Volumetric Hierarchical Approximate Convex Decomposition." It breaks the mesh into multiple small convex parts.
- **Visualization:** Enable *Show Collision* (`Alt + C`) in the viewport to see the hidden green wireframe representing the physical bounds.

7.3 Physics Settings Table

Setting Name	Function
Can Character Step Up On	Allows the player to walk up the stairs without jumping.
Collision Presets	Usually set to <i>BlockAll</i> for structural modeling.
Project Default	Uses the global engine settings for physics.