

Creating an escape room in Core

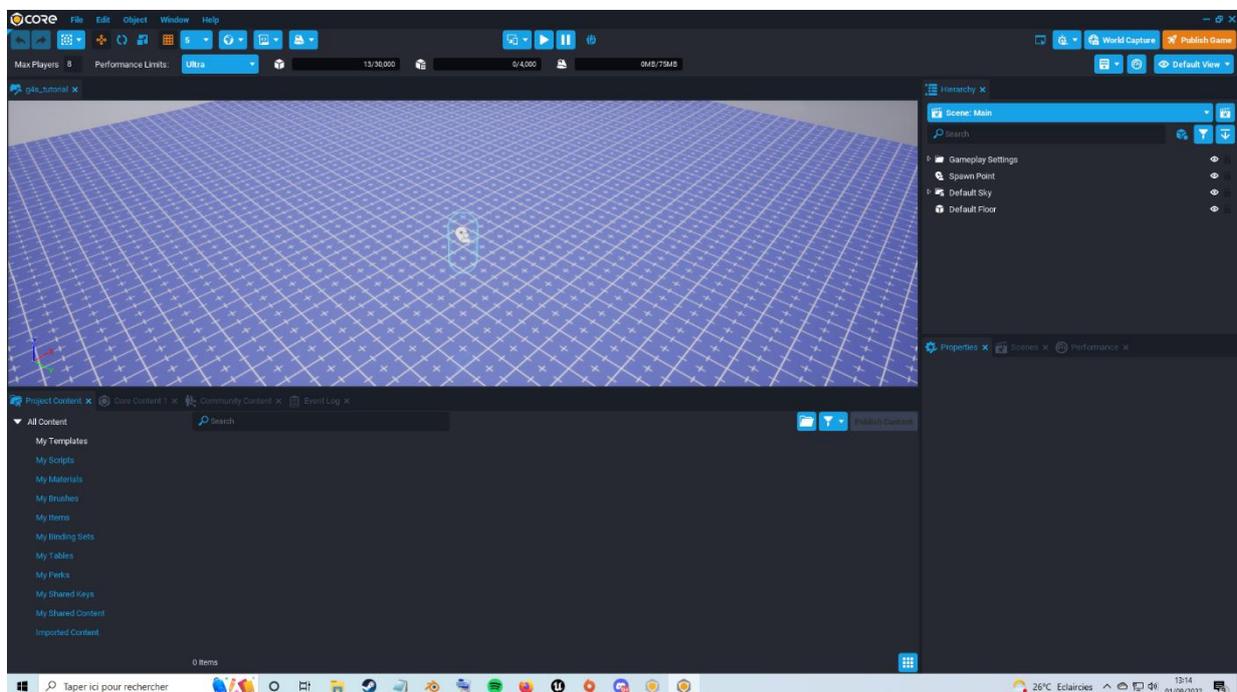
Previous compulsory steps / Prior students' knowledge	None
Learning objectives	Learning how game creation works.
Subjects	Arts, Problem-solving
Recommended Age	(15 - 18)
Material needed	<p>Minimum</p> <p>OS Windows 10 64-bit</p> <p>Processor Intel Core i5-7400 or AMD equivalent</p> <p>Memory 8 GB</p> <p>Graphics NVIDIA GeForce GTX 1050 Ti or AMD equivalent</p> <p>Requires Epic Games account</p> 
Sequence duration	60 (learning the basics) to 120 minutes (application)
Individual or group activity	Group activity or Individual
Tips to make the sequences more accessible or inclusive	<p>The game interface is in English no translations were available when we wrote this sequence. However, it is mostly “game development” vocabulary. What you can do is provide a translation sheet to your students beforehand. Go through the interface and this sequence to see the terms they will need.</p>

Skills developed

Digital literacy

Step by step: how to implement the sequence

- **Step 1: Interface overview (10 minutes)**



The creator interface in Core is composed of different areas. The toolbar at the top left allows you to manipulate a selected object (scale, rotation and position gizmos).

You also have a script generator and a map generator. In the center is the play button to test the current state of your game.

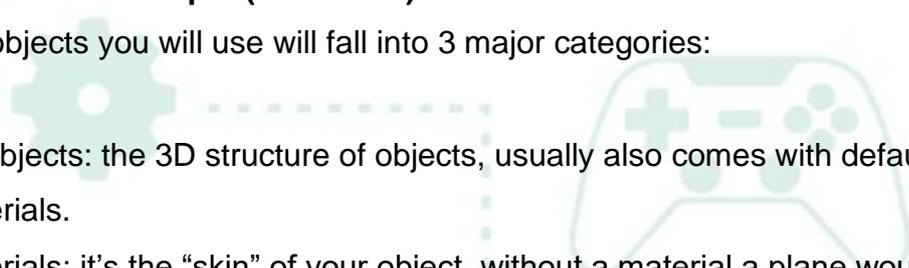
The right panel shows all the objects in your scene in a similar fashion to folders in a file explorer.

The bottom widget gives ways to access the imported and created templates you have in your project. It also lets you access the Community content to browse for objects created by the community. You can also access the basic objects provided by Core developers in Core Content.

When you click an object, it is highlighted in the hierarchy. In the bottom right corner, you will see the properties of the selected object.

- **Step 2: Basic concepts (5 minutes)**

The set of objects you will use will fall into 3 major categories:

- 
- The background for the list of categories features a large, faint green gear on the left and a game controller on the right, with a dashed line connecting them.
- **3D objects:** the 3D structure of objects, usually also comes with default materials.
 - **Materials:** it's the "skin" of your object, without a material a plane would be an abstract geometrical shape, but with a wood material on, it becomes a wall!
 - **Templates:** an object that gathers all the others to provide both visual and logic (gameplay) features.

Every template you create or import is a combination of other objects. You can dive inside a template to change their parts like a car mechanic.

Templates and objects also feature properties. The general properties are their location, rotation and scale, they give the object it's shape and position in the world. Other properties are the materials or variables used for the logic of the object.

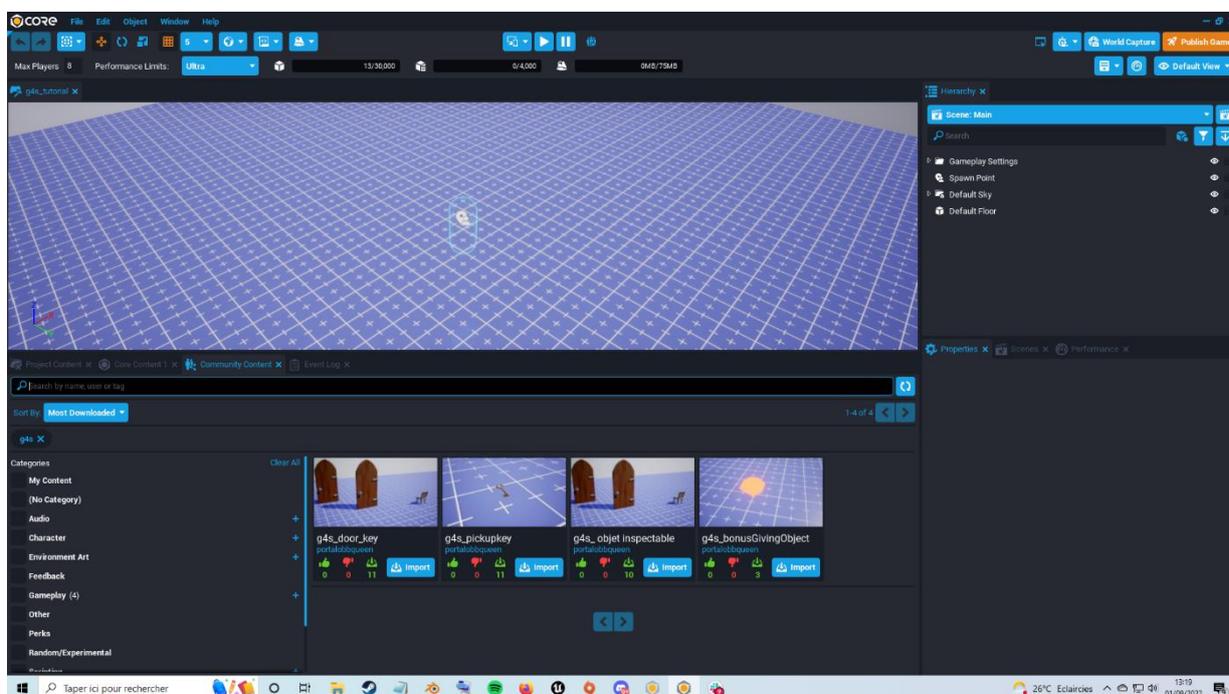
We can't cover everything so, for more details check [the official Core documentation](#).

- **Step 3: Obtaining game elements (5 minutes)**

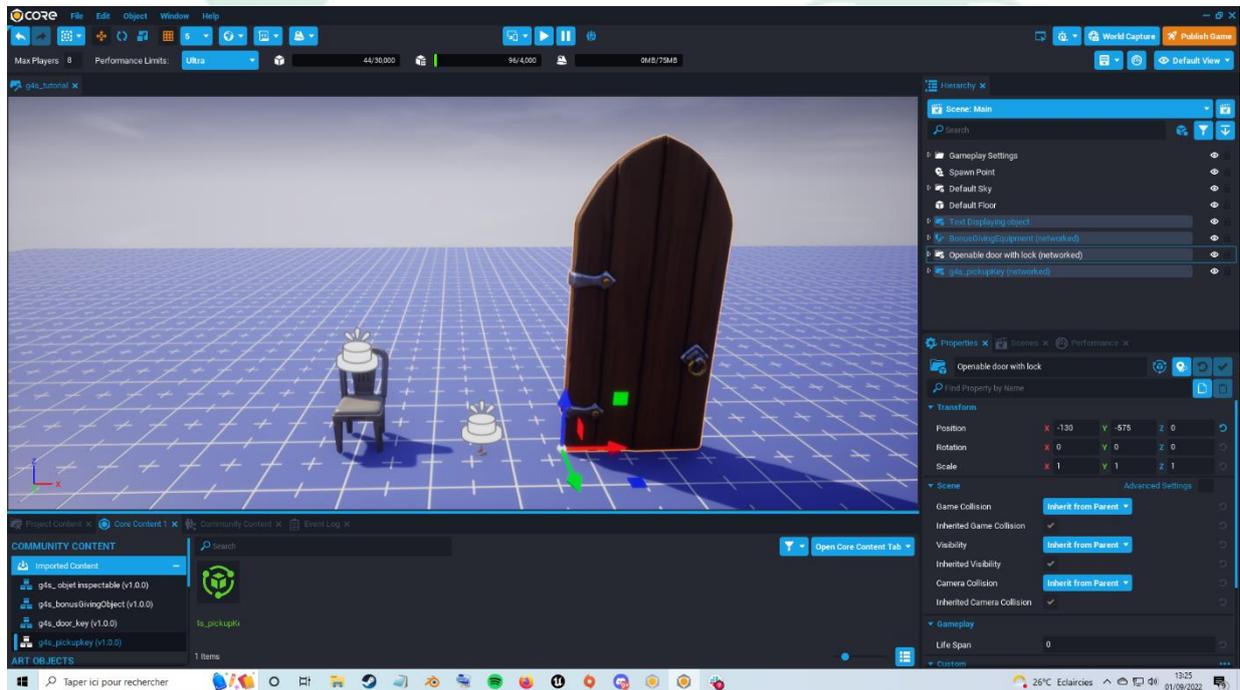
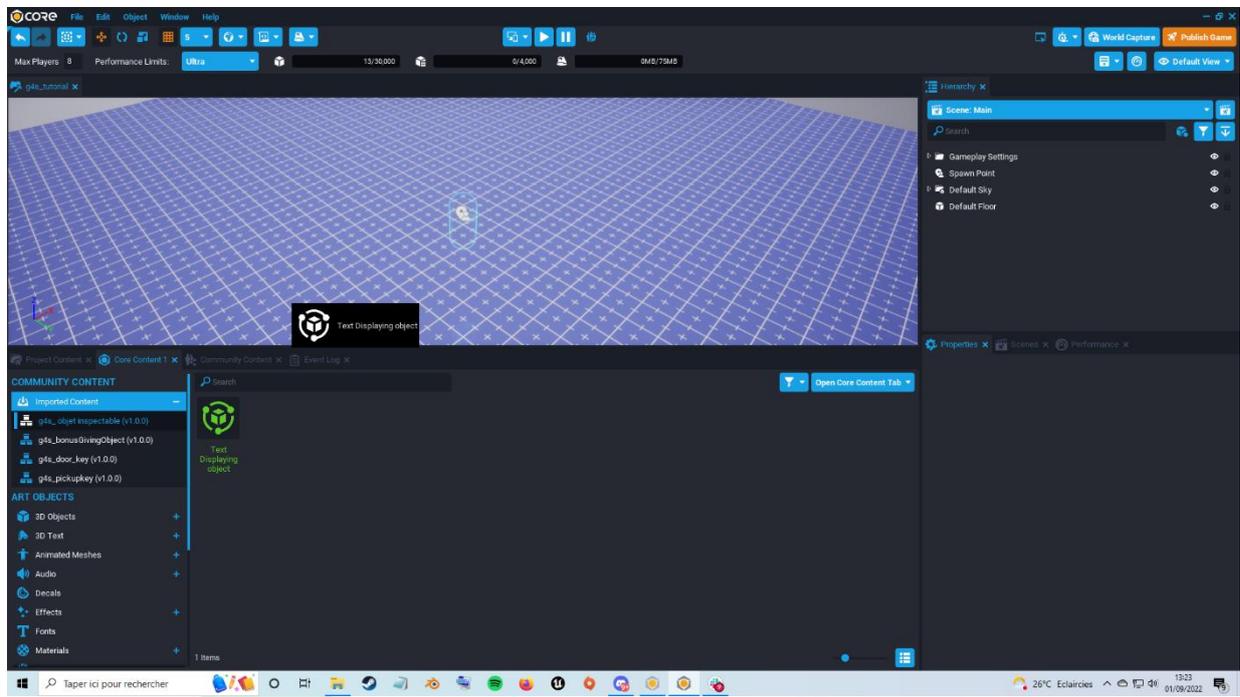
We published some basic gameplay elements that you can use.

Go to the bottom widget, in the “Community content” tab and search for “g4s”.

You'll see a set of templates starting by “g4s_”, import them into your project.



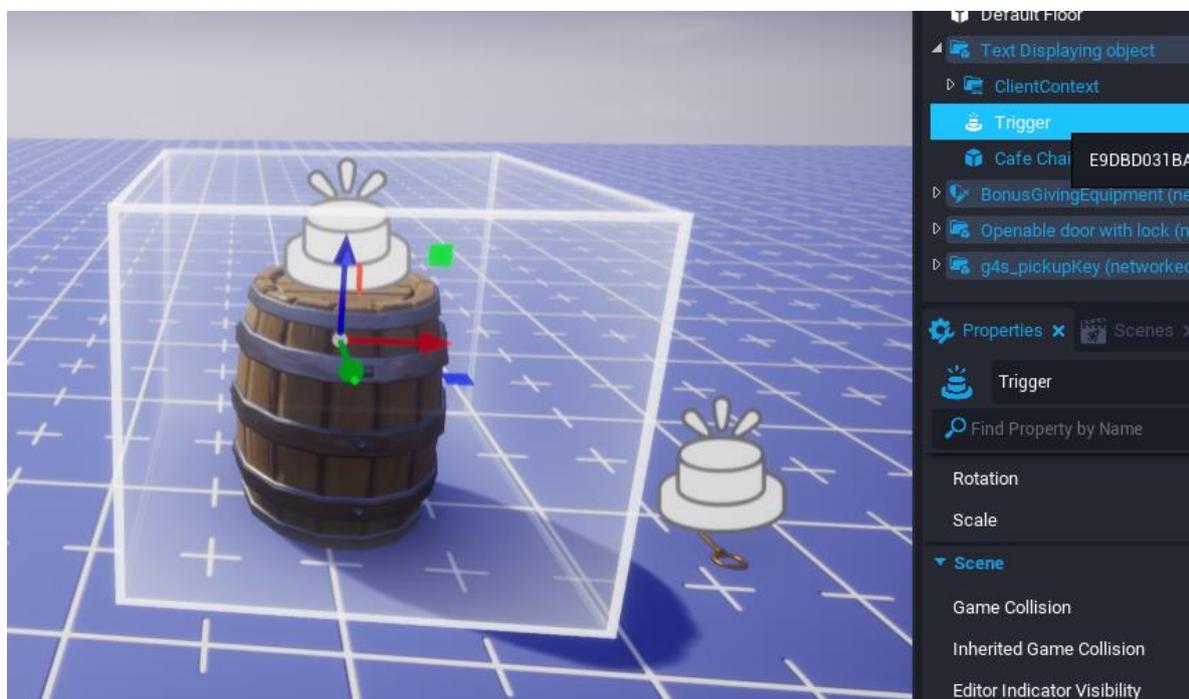
If you go to the “core content” tab, under “Imported content”, you'll see the imported templates. Drag and drop them in the viewport to add them to the scene.

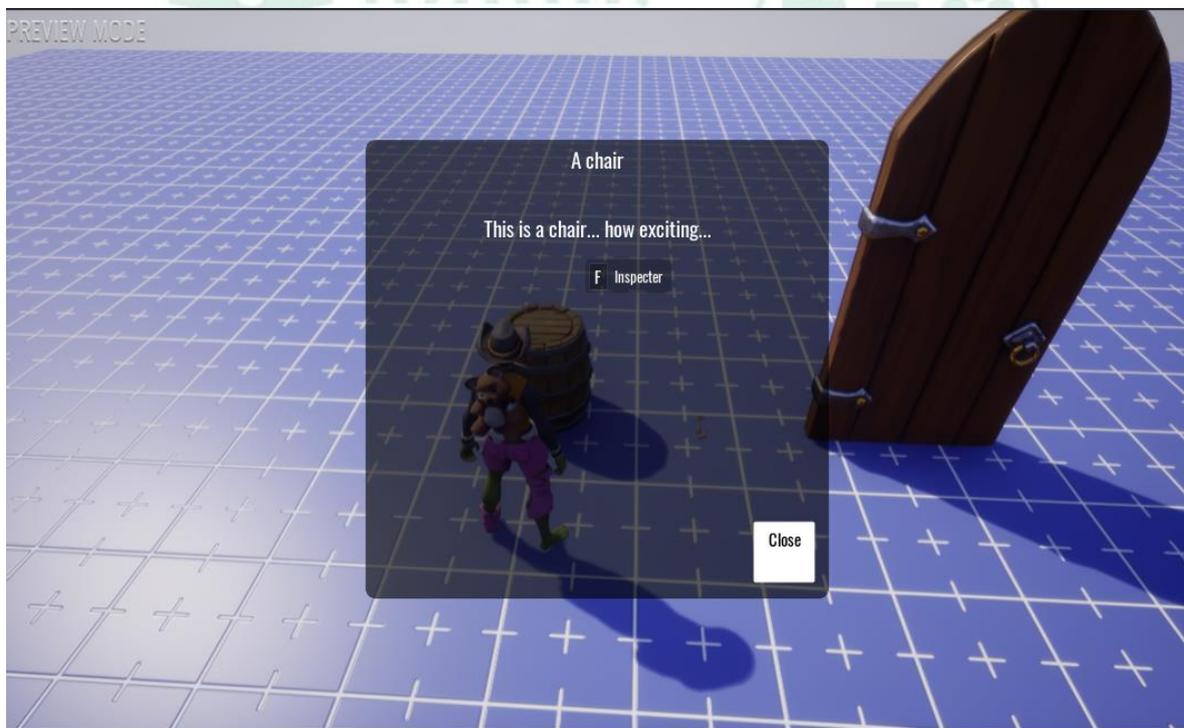


- **Step 4: Breakdown of game elements (5 minutes)**

The “text displaying object” is a template that provides a 3D model around which a trigger zone displays a command to the player: “F – Inspect”.

When the player clicks F while close to the object, it opens up a closable window with some text inside.





The door and keys work together.

When the player stands near the door, displays an interaction pop-up: it can be opened by pressing the F key.

If the player doesn't have the key, the door will not open.

If the player walks on the key, they obtain it. If they go back to the door and press F, this time it opens.

The door can require several keys to be opened. And it also works with a specific type of key.

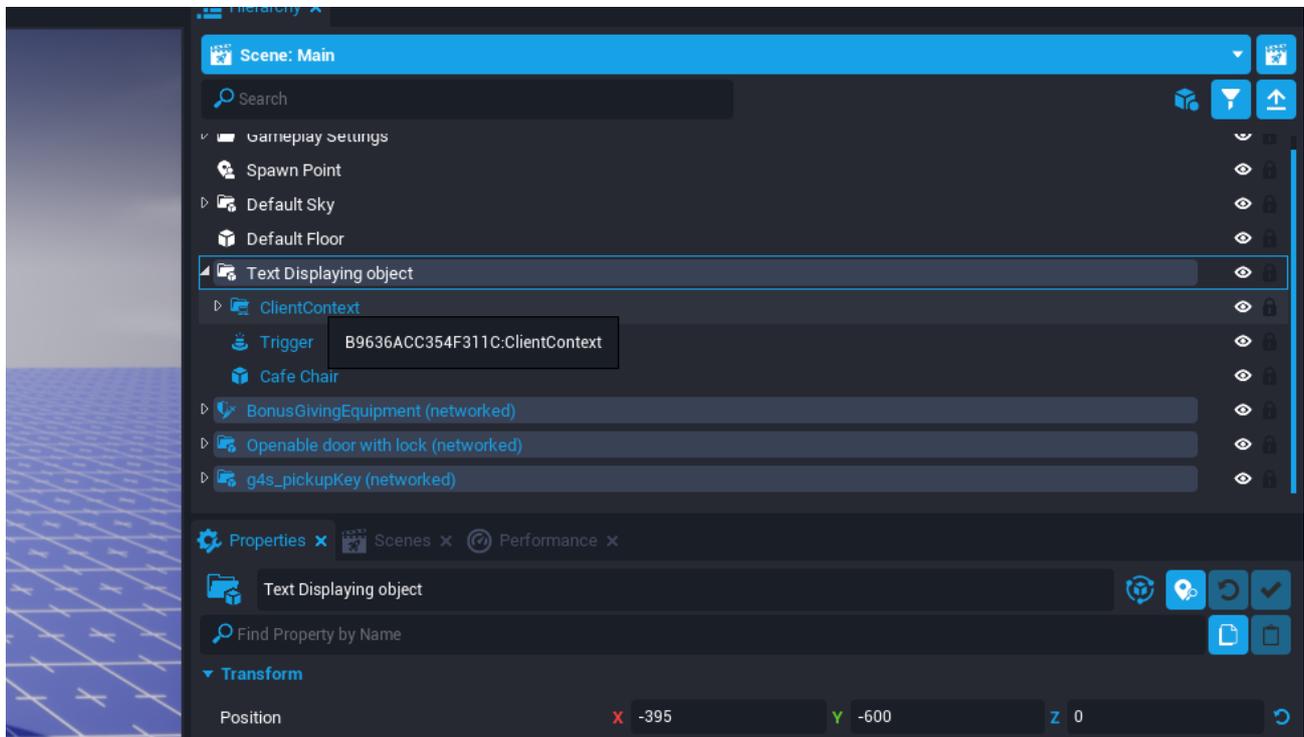
The properties of the door controlling this behavior are: "numberOfKeysRequired" and "KeyResourceName".

- **Step 5: Modifying game elements (15 minutes)**

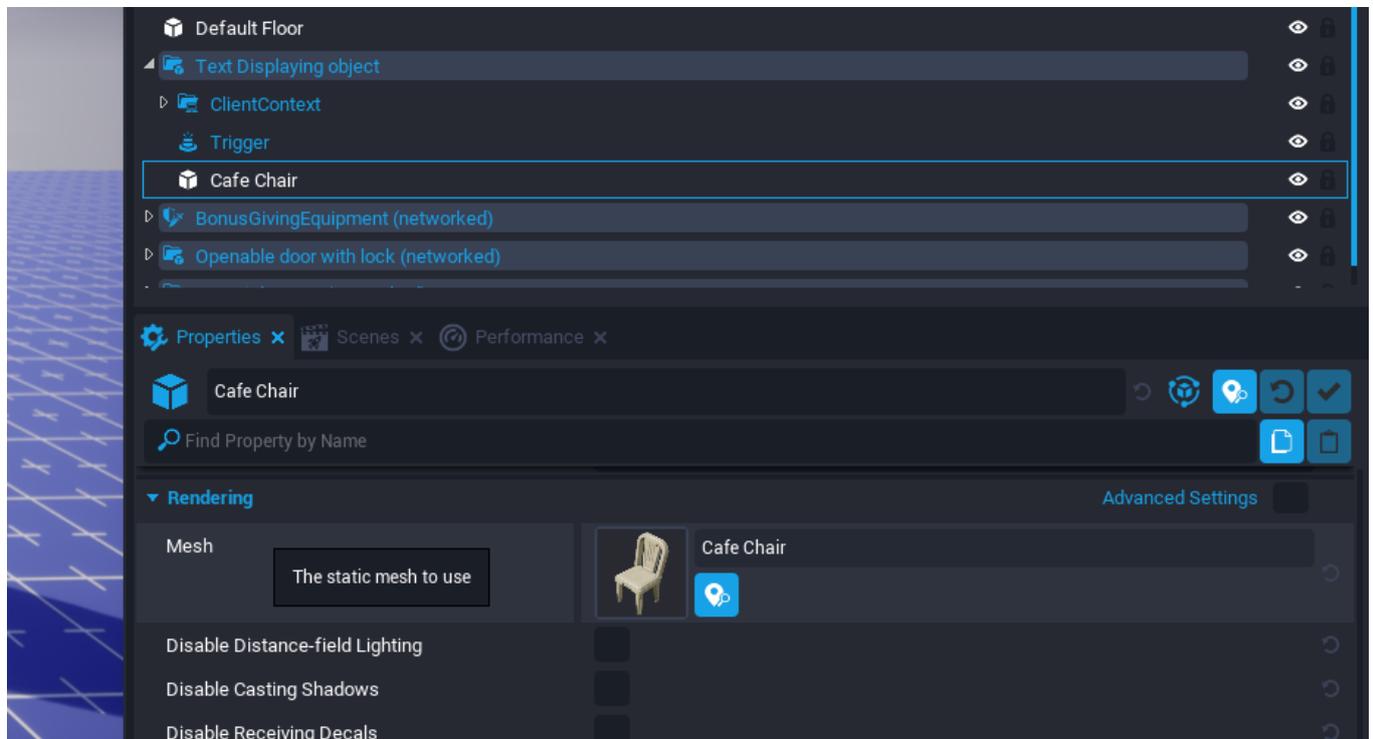
These elements are objects that have different components. If you click them, they will be highlighted in the Hierarchy in the right widget.

You can explore the content of an object just like you would with a folder in a file explorer.

- Let's open the "Text Displaying Object" folder.



The cube named "café chair" corresponds to the visual part of the object, what we see in the game. Once click, search the object properties tab for the "mesh" property:



You can double click it to choose another object's visual (3D model).

A menu pops up, use the filter on the left to search for the 3D model you would like.



We have changed the object's appearance but it works the exact same way. The logic hasn't changed.

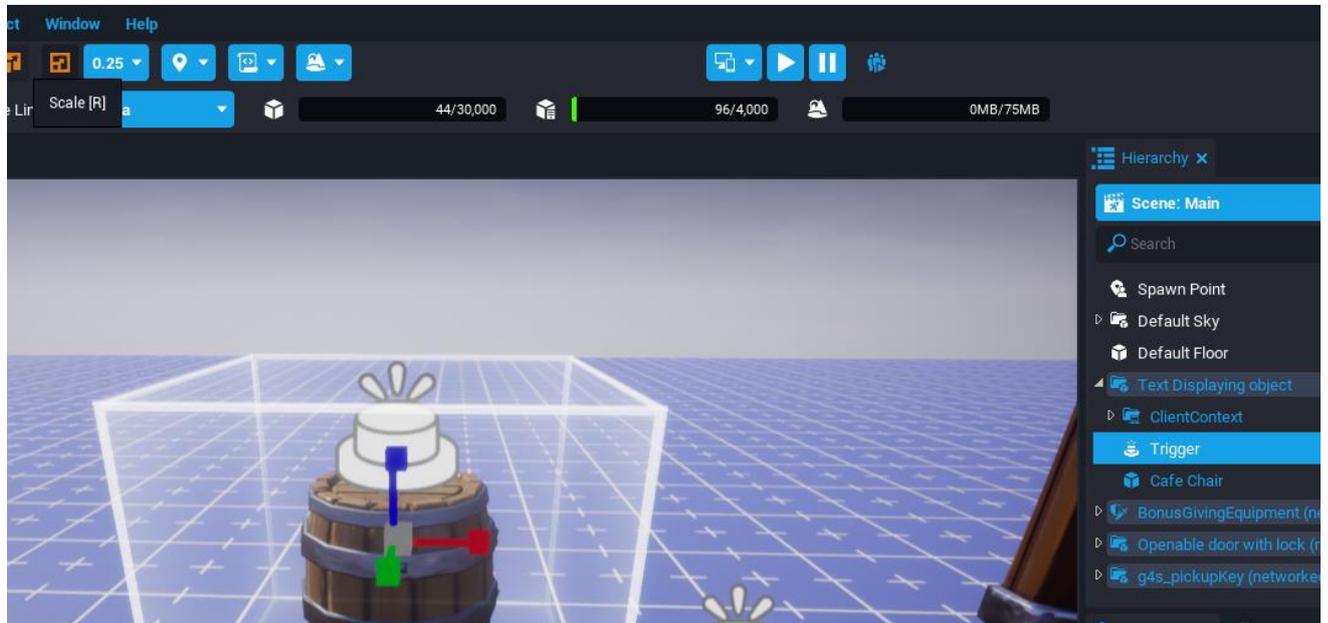
The logic works as follows:

The trigger detects if a player is in its area. If they are, they launch a script ("UIInteraction", under "UI Container" component).

You can change the interaction label under the trigger properties, it will change the displayed text when player comes near the object.

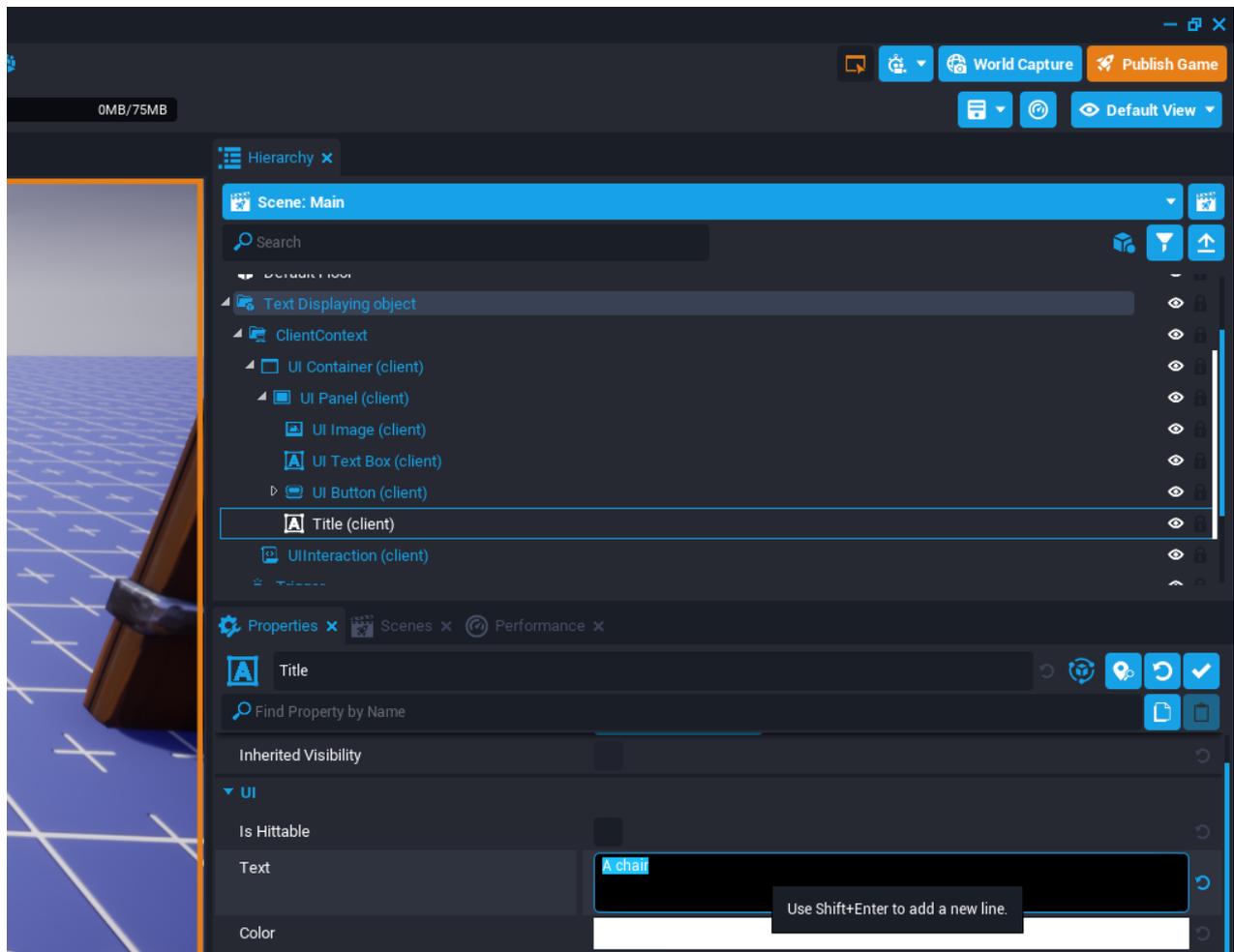
If you change the object 3D model, you might end up with a trigger zone that doesn't fit anymore.

To resize the trigger, click it in the hierarchy and in the toolbar select “scale”:

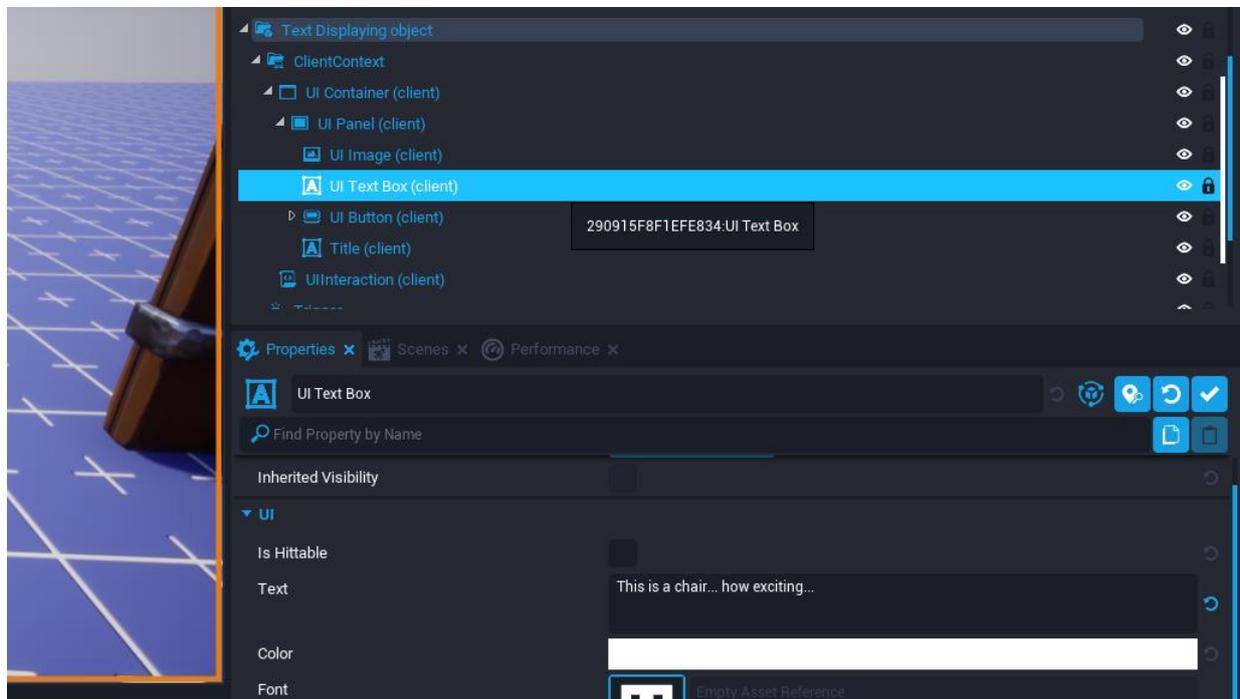


Use the gizmo to modify the trigger area.

To modify the title of the displayed window, in the object hierarchy go to ClientContext>UI Container> UI Panel> Title
Change the “Text” property.



To modify the text of the displayed window, in the object hierarchy go to ClientContext>UI Container> UI Panel> UI Text Box
Change the “Text” property.



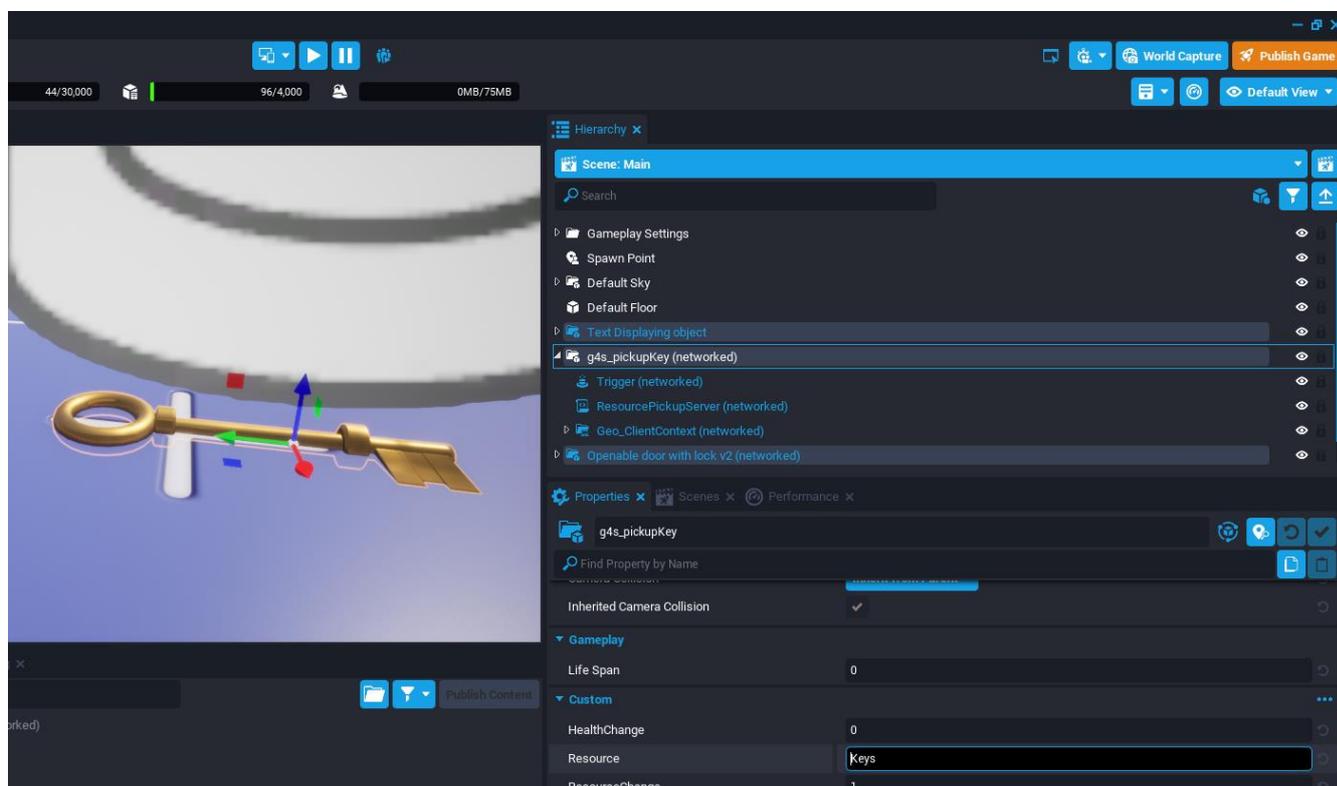
For the door you can modify its appearance just like the “text displaying object”.
To create gameplay with the door you need to modify

Modifying the door

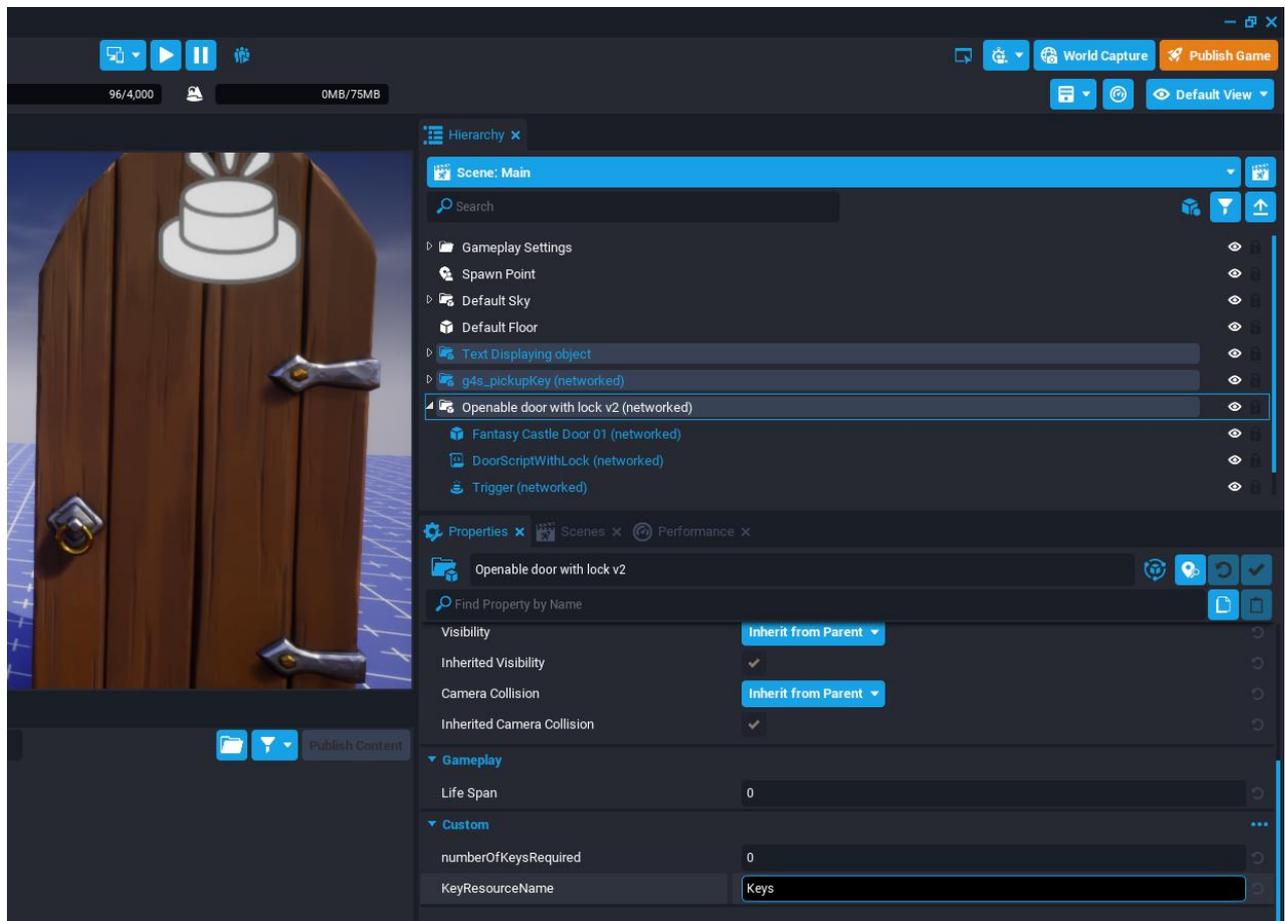
The basic door template: “g4s_doorWithLock_en” works with 1 “g4s_door_key”.

You can have different keys working for different doors.

To match a key with a door, change the key’s “Resource” property:



And type the same value in the door's "KeyResourceName" property:



You can also modify the “numberOfKeysRequired” property to make the door need several keys of a specific type to be opened.

- **Step 6: Level design (20 minutes or more)**

You now have all the tools you need to make a simple escape game!

You can create a level using community content and add the 3 gameplay elements presented previously to make a door openable only if the player finds certain keys in a certain amount and some objects in the room give clues to find them.

What you can do is place a key in a spot a little hard to reach/find and create a simple enigma that indicates it. Go to the community content and use the search bar. You can also filter for Props> interior.

References:

All images are in-house screenshot from Core (Manticore Games, 2021).

Official documentation: <https://learn.coregames.com/>

