Cerebrum™

Level Gen. Designer Tool Guide

V 1.0.2

Procedure Summary

1. Use the Level Gen. Designer Tool to design the room and its contents
2. Export the Level Setup File
3. Create a scene in Unity
4. Disable any default lighting in the scene
5. Add a Game Manager to an empty game object
6. Add the RoomGenerator.cs script to an empty game object
7. Reference your Level Setup File with your Room Generator script
5. Run the level

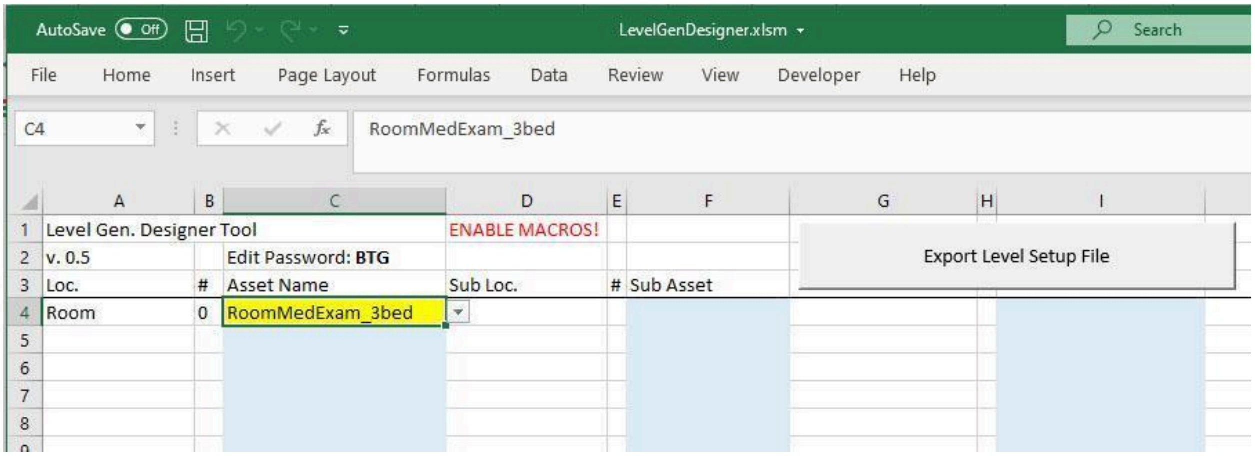
Level Gen. Designer Tool

Before you begin

The Level Generator system creates a **Room** for a specific scenario. There are several pre-made rooms to choose from, some empty and some furnished. The room contains some number of **Room Objects** (furniture, medical equipment, characters) which may in turn contain some number of **Sub-Objects** (documents, computer, infusion bag, patient on bed). There are specific **Object Locations** in each room where objects of specific types may spawn. Those objects, in turn, may have **Sub-Object Locations** where sub-objects may spawn as children of their parent objects.

Edit the Room Contents

a. Open the **LevelGenDesigner.xlsm** file in Excel.
b. Be sure to **ENABLE MACROS** if you haven't already done so. Otherwise, you will not be able to export the level setup file. Make sure Tools or Insert menu -> Add-Ins Analysis ToolPak or Analysis ToolPak - VBA is activated.



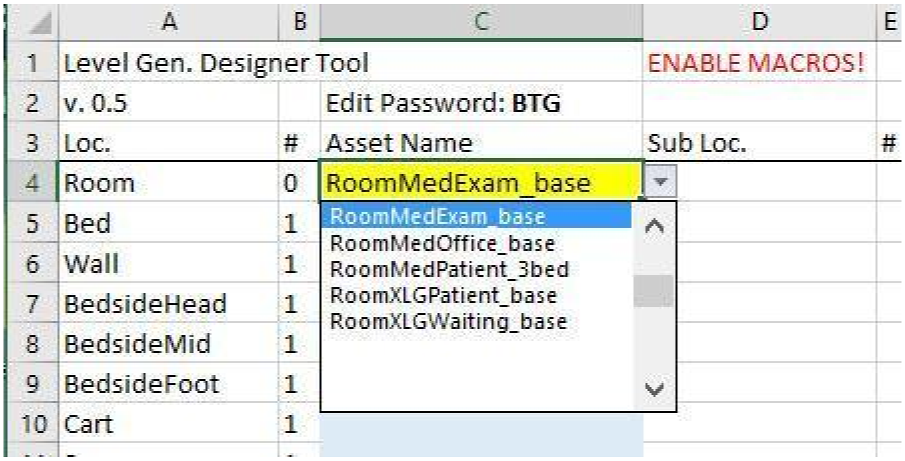
C4 | RoomMedExam_3bed

	A	B	C	D	E	F	G	H	I
1	Level Gen. Designer Tool			ENABLE MACROS!					
2	v. 0.5		Edit Password: **BTG**						
3	Loc.	#	Asset Name	Sub Loc.	#	Sub Asset			
4	Room	0	RoomMedExam_3bed						
5									
6									
7									
8									

c. All Level Generator options are on the first page of the workbook. User-editable fields are highlighted in light blue or yellow.

Selecting a **furnished** room asset (no object locations)

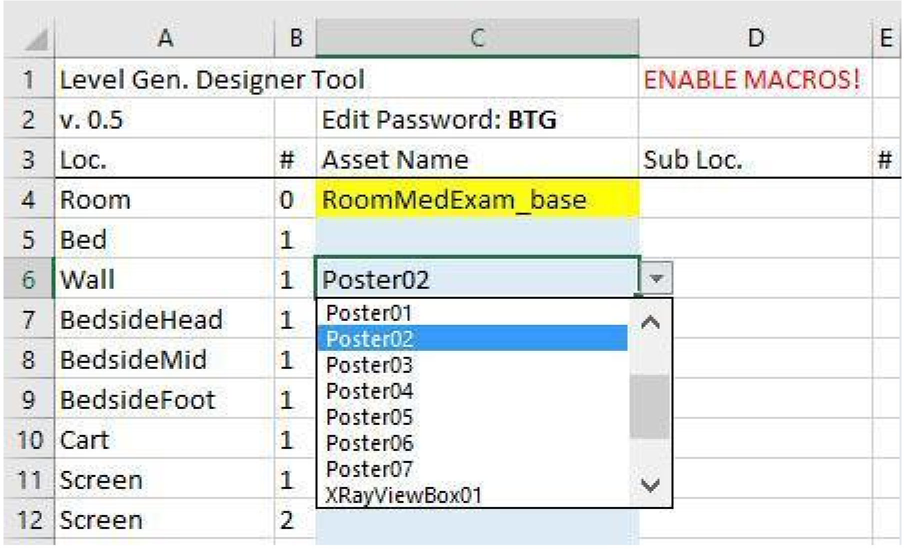
d. First, select a **Room** asset for the level to serve as the base setting for the scene. **Base** rooms are empty while others are pre-furnished. **You should be using BASE rooms as furnished rooms are not yet completely functional.** The name format is Room[Size][Type]_[descriptor]. There is always only one room asset and it is always in location zero.



	A	B	C	D	E
1	Level Gen. Designer Tool			ENABLE MACROS!	
2	v. 0.5		Edit Password: **BTG**		
3	Loc.	#	Asset Name	Sub Loc.	#
4	Room	0	RoomMedExam_base		
5	Bed	1			
6	Wall	1			
7	BedsideHead	1			
8	BedsideMid	1			
9	BedsideFoot	1			
10	Cart	1			

Selecting a **base** room asset (with object locations)

e. When a room is chosen, all available **Object Locations** will propagate through the workbook. Object locations are categorized by **Object Type** and **Number**. Usually, the object locations with the same number are grouped together (though, not always).

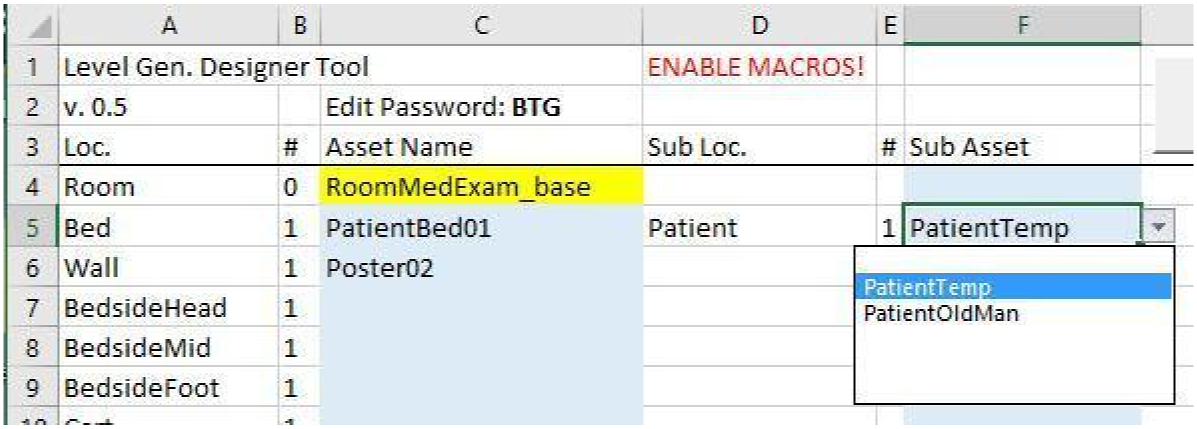


	A	B	C	D	E
1	Level Gen. Designer Tool			ENABLE MACROS!	
2	v. 0.5		Edit Password: **BTG**		
3	Loc.	#	Asset Name	Sub Loc.	#
4	Room	0	RoomMedExam_base		
5	Bed	1			
6	Wall	1	Poster02		
7	BedsideHead	1			
8	BedsideMid	1			
9	BedsideFoot	1			
10	Cart	1			
11	Screen	1			
12	Screen	2			

Poster01
Poster02
Poster03
Poster04
Poster05
Poster06
Poster07
XRayViewBox01

Selecting a wall object asset at object location **Wall1**

f. Click the drop-down arrow in the cell just to the right of a location number. The drop-down list contains all available **Room Object** assets (by Addressable name) that match the location's object type. Select one room object to spawn in this location. **Note:** you may need to scroll up or down to see all options.



	A	B	C	D	E	F
1	Level Gen. Designer Tool			ENABLE MACROS!		
2	v. 0.5		Edit Password: **BTG**			
3	Loc.	#	Asset Name	Sub Loc.	#	Sub Asset
4	Room	0	RoomMedExam_base			
5	Bed	1	PatientBed01	Patient	1	PatientTemp
6	Wall	1	Poster02			
7	BedsideHead	1				
8	BedsideMid	1				
9	BedsideFoot	1				

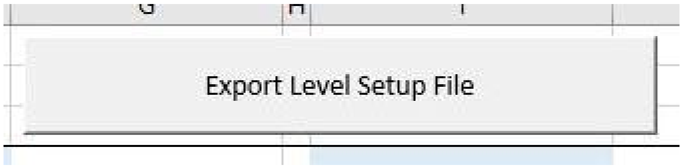
PatientTemp
PatientOldMan

Selecting a patient object asset at sub-object location **Bed1/Patient1**

g. If applicable, the chosen room object will propagate all available **Sub-Object Locations** categorized by **Sub-Object Type** and **Number**.

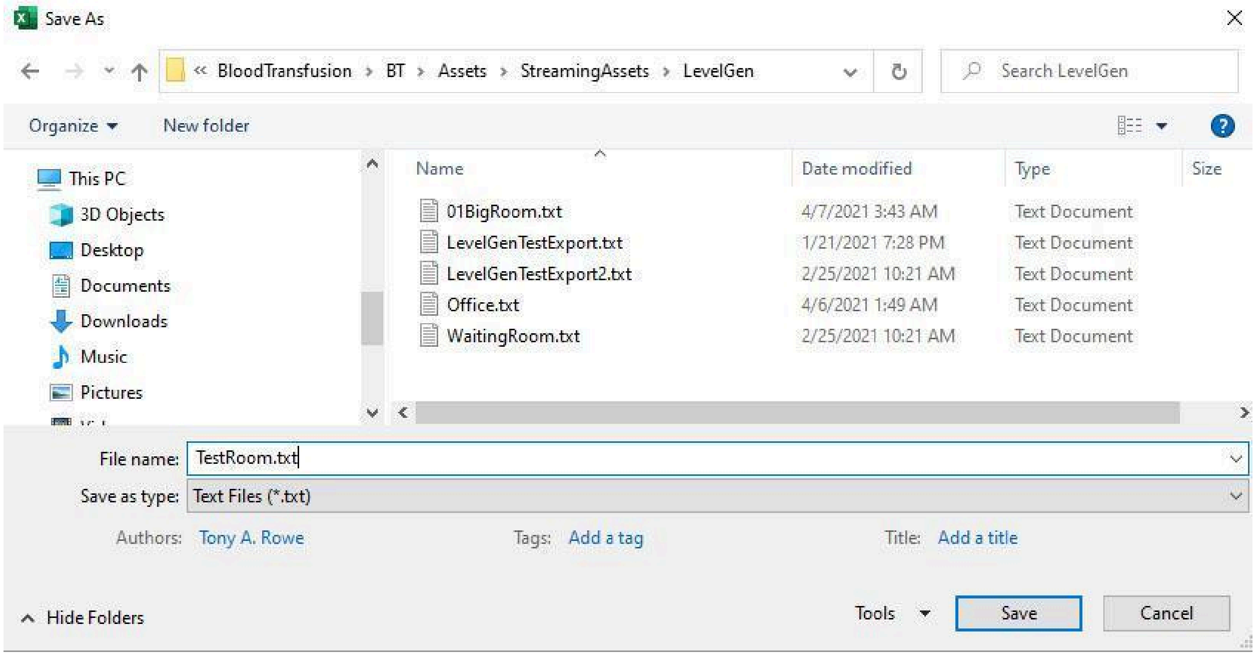
h. Click the drop-down arrow in the cell just to the right of a sub-object location number. The drop-down list contains all available **Sub-Object** assets that match the location's sub-object type. Select one sub-object to spawn at this location. The sub-object will spawn as a child of the parent room object and may be referred to in script by [parent object location][number]/[sub-object location][number].

i. Continue selecting room objects assets and sub-object assets until the room is complete.



Export Level Setup File button

j. Press the Export Level Setup File button near the top of the workbook. **MACROS MUST BE ENABLED.**



Name	Date modified	Type	Size
01BigRoom.txt	4/7/2021 3:43 AM	Text Document	
LevelGenTestExport.txt	1/21/2021 7:28 PM	Text Document	
LevelGenTestExport2.txt	2/25/2021 10:21 AM	Text Document	
Office.txt	4/6/2021 1:49 AM	Text Document	
WaitingRoom.txt	2/25/2021 10:21 AM	Text Document	

File name: TestRoom.txt

Save as type: Text Files (*.txt)

Authors: Tony A. Rowe Tags: Add a tag Title: Add a title

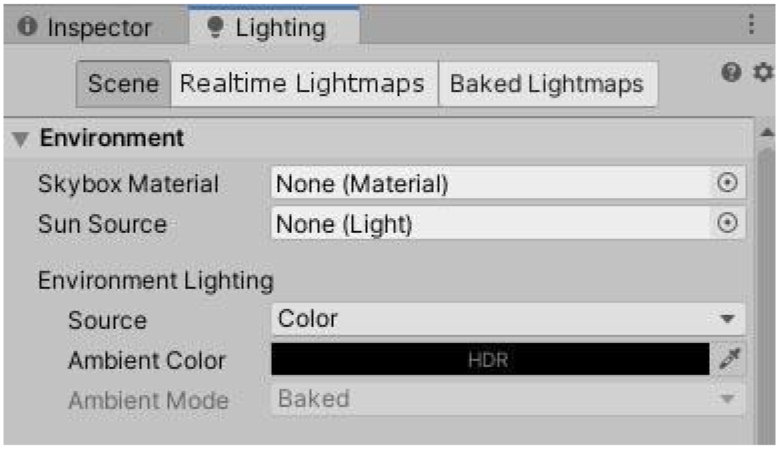
Level setup file save dialog

k. Name your Level Generator Scene file and export it to the project at **Assets/StreamingAssets/LevelGen/**.

l. You may close Excel if you wish. You do not need to save your changes to the Excel file.

In the Unity Editor

a. Create a new **Scene** file in the Unity editor.



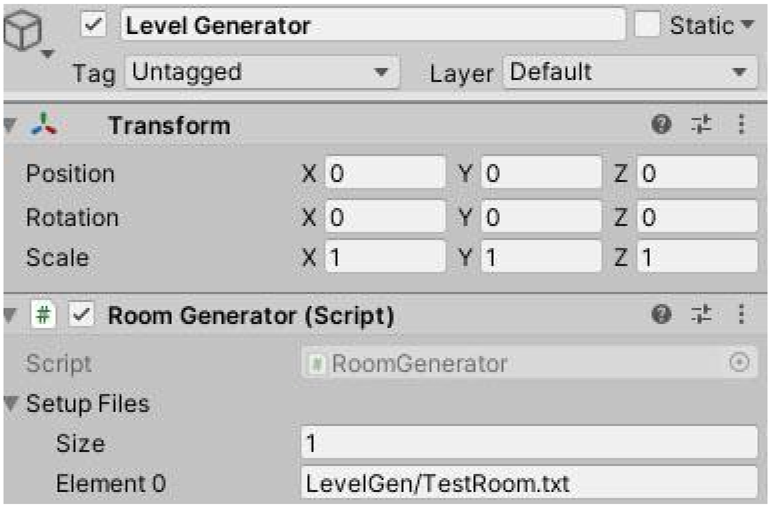
Skybox Material	None (Material)
Sun Source	None (Light)
Environment Lighting	
Source	Color
Ambient Color	HDR
Ambient Mode	Baked

Lighting Window (Scene): Set the Skybox Material to None and set Ambient Color to Black

b. The scene's lighting information will be imported by the level generation system. **Delete the default directional light added to the scene by Unity**. Open **Window -> Rendering -> Lighting Settings**. Set the **Skybox Material** to **None** (click the roundel at the right of the text box, then select the None setting at the start of the Material Explorer). Set the **Ambient Color** to **Black**. That should get rid of all the excess lighting information that can affect the scene.

c. The scene should only have a Main Camera object at this time. Add either an **Object Message Handler** (Paul's messaging system) or a **Message Handler Camera** (Tony's messaging system), depending on which system you are using.

d. The scene should also have a game manager. Add an empty game object with either **My Game Manager** (Paul's system) or **Game Manager** (Tony's system). You don't need a scenario file yet if you are just testing out the level generation.



☑ Level Generator □ Static

Tag Untagged Layer Default

Transform

Position	X 0	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

☑ Room Generator (Script)

Script RoomGenerator

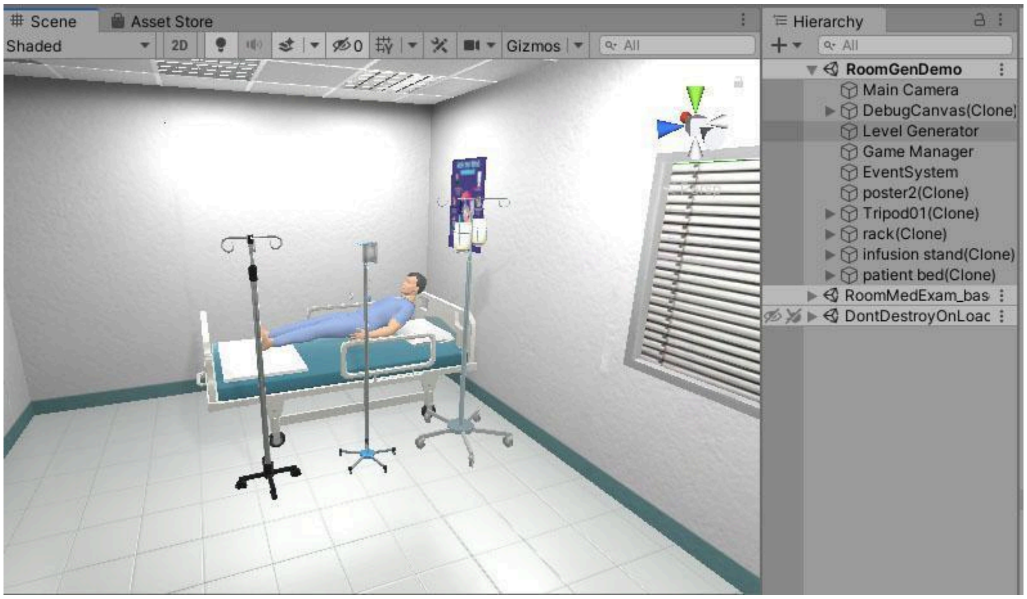
Setup Files

Size 1

Element 0 LevelGen/TestRoom.txt

Room Generator script settings

e. Finally, add an empty game object and add the **Room Generator** script. **Setup Files** should have a size of **1**. The first element needs to be the name of your level generator scene file and its subdirectory in the streaming assets folder. In the example above, I used: **LevelGen/TestRoom.txt**



A simple scene loaded with a silly number of infusion racks

f. Run the scene. You should see your room set with all of the requested objects spawned into the scene within about 1-2 seconds. We will need to add a fade in when a scene starts so the user does not see objects pop into view.