Grid Builder

Index.
Quick Setup
Set up from scratch
Set up Auto Cell Blocking

Quick Setup

1.
Navigate to /Prefabs/WholeGrids, and drag in one of the premade grids and adjust the General settings such as height, width and cellSize.
Best to keep the cellSize at 1 = 1 metre. However everything will adjust if you decide to change it.
If you want the grid to be above some existing terrain, move the grid on the Y axis slightly above. A value of 0.015 works good.
You can use as many grids as you want, 1, 10 or 50, its up to you.
There is no 'Hard' limit on width or height, I have tested up to 90,000 cells with 300 height and 300 width which starts to slow down on my mid range machine.
Caution to increasing grids and cells quantity will certainly affect performance after certain thresholds dependent on your hardware.

2.
Navigate to /Prefabs and drag in a GridSelector, ObjectPlacer, and ObjectRemover.
You should only have one of each of these.
Assign the ObjectPlacer to the gridSelector slot.

3.
Back in /Prefabs, drag in a SelectObjectBtn and RemoveModeBtn.
This will create a canvas so you can move the buttons somewhere convenient.
By default the SelectObjectBtn has a demo prefab assigned but feel free to change it.
Please note there is no scaling done to prefabs,
so the models will have to match the grid size.

4.
That's it! Hit play and you should be able to move, click on an object and place it down on the grid.
You can also hit the removeMode button to highlight and delete objects.
Both placing and deleting objects is done with the left mouse button and cancelling either is done with either the right mouse button or excape.

5.

After this there is quite a bit to play around with, particularly the AutoCellBlock feature. Please read the documentation on how to set this up.
Its super simple and quick to set up once you know how it works.

You can create all of these prefabs from scratch if you like, creating an empty game object and adding the relevant scripts to them as components. You will certainly need to assign materials, and certain prefabs to them if you want to do it this way. All the folders are clearly marked so you should not have any trouble finding them.

Set up from scratch

1.
Create a new empty gameobject, and add the 'GridSquare' script.
If choosing this way, you will need to link up a gridCell prefab from /Prefabs/GridCells if in single cell mode,
or apply a 'Simple' material from /Materials/Grid if in simple mode.

2.
Feel free to play around the initial settings to align the grid, select height, width and cellsize.

3.
The grid mode is mostly game dependent on your style, complexity and your final output device.
Refer to full documentation for details. Best to start off with simple.

4.
If you are not using the autoCellBlock feature(removes cells depending on certain factors) please continue.
If you want to use autoCellBlock in your project, it will take seconds to set up when you understand how it works.
Please refer to the documentation provided to set this up.
If you choose to try this on your own, you may end up with no cells being created at runtime.

5.
Create a new empty gameobject, and add the 'GridSelector' script.
Again if choosing this way you will need to link up the mesh, found in /Meshes/GridSelectors > on the fbx and drag the mesh to meshfilter.
You will also need the selector materials, all found in /Materials/GridSelector.

6.
Now you should be able to select the grid, and move around, hazaah.
For placing objects, two components are required. One for creating the actual object,

and one for selecting what to build and passing the object to the gridSelector.
Or create an empty gameobject and place the 'ObjectPlacer' script on it, then drag this onto your gridSelector Object Placer. Easy peasy.
You can copy the buttons from my Demo scene, or simply create a new button, and add the 'SelectObject' script to it.
This script requires one thing, a prefab of an object you want to build on the grid.
All this does is passes an object to the gridselector, and then when clicking with the left mouse button,
passes the object to the ObjectPlacer to build the object.
Feel free to add any visual element under the button to represent what your building.
I was lazy and grabbed the thumbnail from the Asset folder =).

7. Hopefully now, you can click the button, and build on the grid, great!
For removing objects, create another button, this time add the 'RemoveMode' script on it.
Drag the ObjectRemover from /Prefabs into your scene.
Or create an empty gameObject, and place the 'ObjectRemover' script on it.
Add the invalidMaterial from /Material/GridSelector.

8.You should now be able to move around, create and delete objects!

There are a quite a few customisation options to choose from, feel free to have a play around with them or view the full documentation included in this package.

I have created a few example prefabs, cells, grids and textures which you are totally okay to use but the design and 'look' of the grid can be as unique as you want it to be.

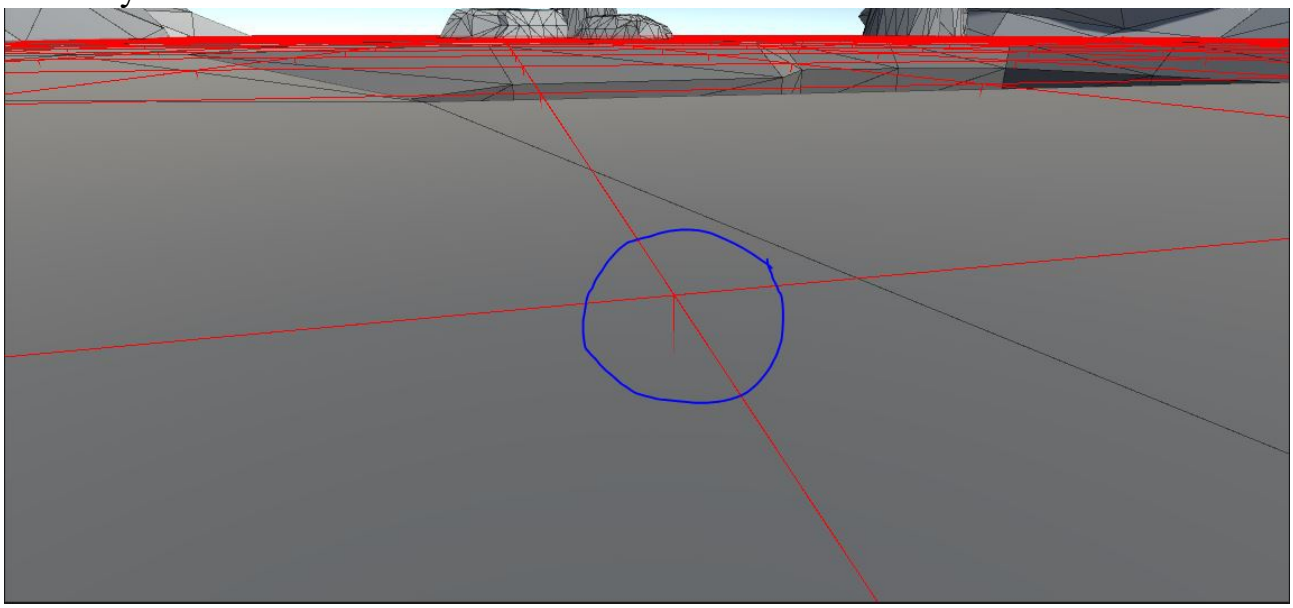
Set up Auto Cell Blocking

1.
Setup a new layer and call it ground or something similar.
Assign this layer to your physical ground object, making sure it has a collider or mesh collider on it. A mesh collider will work best for single objects that have dips or cliffs.

2.
Create a grid and sort out the rest of the settings, making sure that the grid type is set to either single cell or chequered.
Normally the grid will be hovered slightly above the terrain in many instances, so move the grid slightly above. Something like 0.015 above your ground object or however much above the ground is desirable for your game. Slightly above this value is the value needed in Ground Distance.

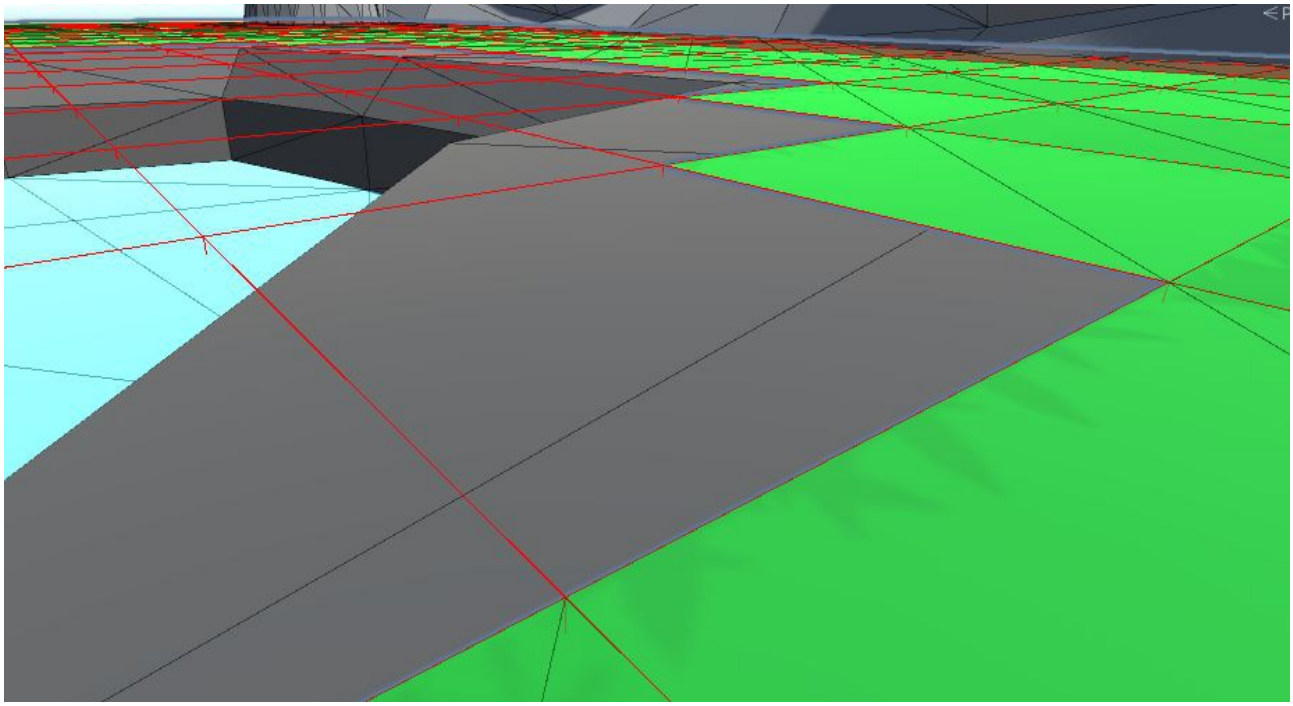
3.
Turn on Auto Cell Block and set the type to block both.
Set the ground layer to the layer your ground object is in.

4.
Turn on Show Below Rays to see the Ground Distance value.
You may need to zoom in a little to see them.



You can see in the picture where the ray hits the ground object and is going through it. This ray should only just be penetrating or going through the ground, so adjust the Ground Distance value to replicate this. If all four corners of a cell touch the ground, a cell will be created, otherwise, it will be blocked.

If your having trouble with this, turn on Check Ground Hits to see your most common distance from the grid to your ground. The most repeated debug.log will most likely be how far your grid is from the ground. A value slightly larger than this in Ground Distance will do the trick. If you have no debug.logs increase your Ground Distance value further.



We can see here there is one corner of a cell overhanging and not touching anything, thus a cell is NOT created.

5.
You can also play around with the Check Box Size and Height creating different spaces around objects like the image below.

NOTE. This takes a little getting used to but it really is quite easy after the first time or two.

Have a look at the sample scene provided if you get stuck to see it working or feel free to contact me on the email below.
If you have any questions or are having trouble setting up please do not hesitate to contact me at support@golemitegames.com.