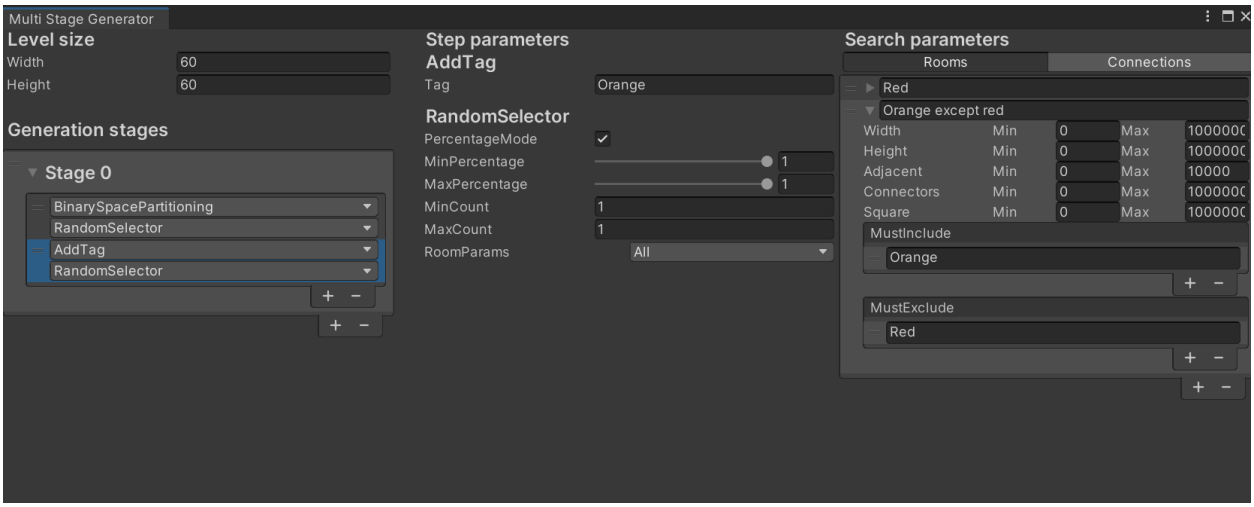
Multi Stage Generator

This generator allows you to set whole layout generation process manually and achieve more complicated dungeon structure.

Overview



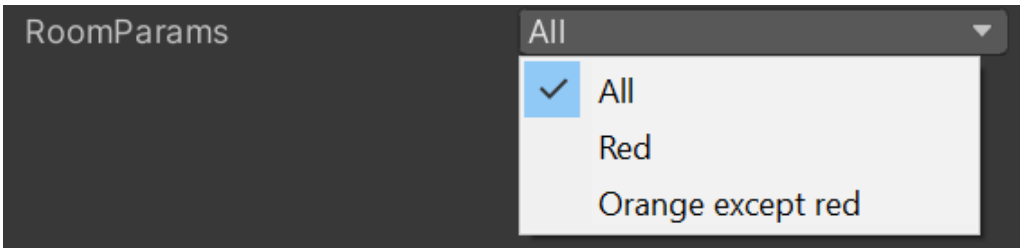
One the left side you can split generation process into multiple stages, and setup generation steps for each stage. Stages are needed only to make it easier to manage, there is no difference between using only one stage or more.

Each generation step has a modifier and selector (you can see it below modifier). To perform a step, generator uses selector to select rooms, and then passes them to modifier.

When you click on the area to the left of each step (blue on the picture) – its parameters appear in the middle.

On the right side you can see Search parameters. They are required as parameters in some modifiers and selectors. Their only purpose is to filter rooms and connectors. MustInclude list specifies which tags are allowed(if room has at least one tag from this list – it will be chosen), and MustExclude specifies tags that are not allowed. For example, on the picture above, “Orange except red”(names is not important, it is used only to make it easy to find) allows Orange rooms, that have no Red tag.

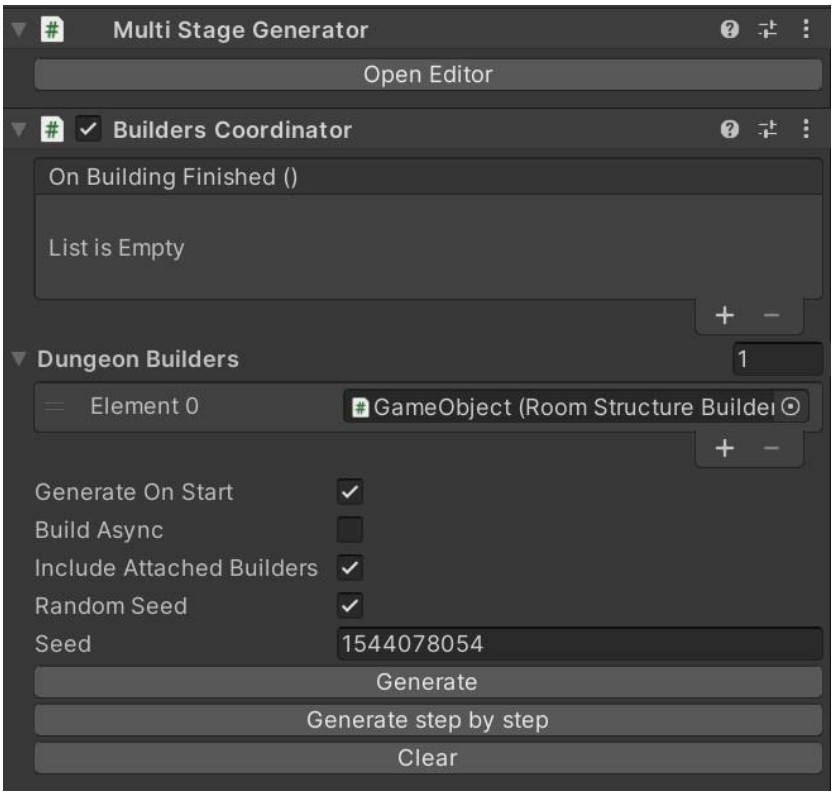
After you added some new search preset – you will be able to choose it from dropdown lists.



You can setup height, square, connectors count and adjacent count requirements as well.

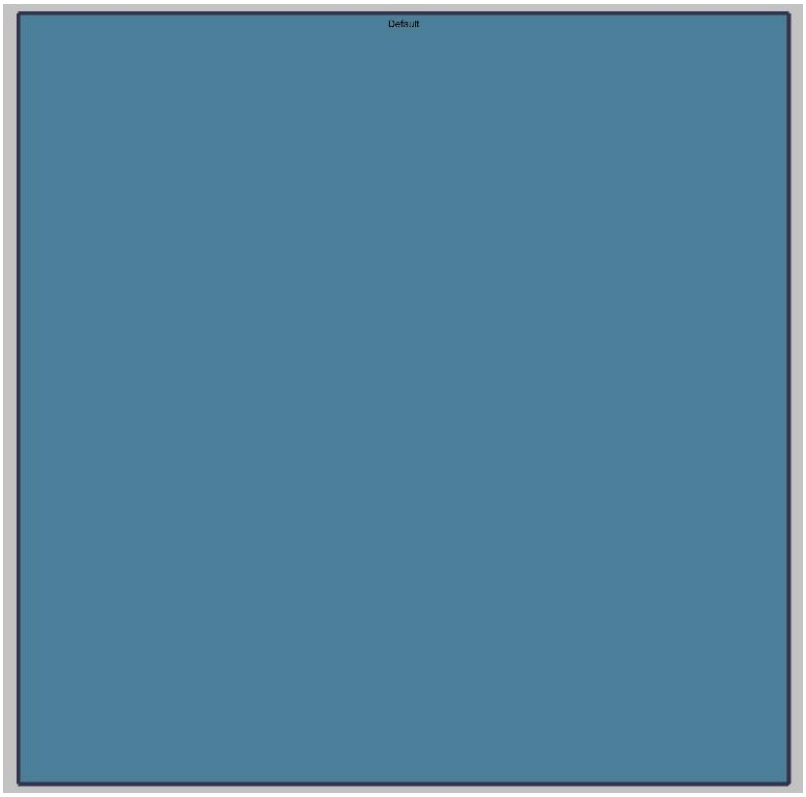
Quick guide

1. Add MultiStageGenerator and BuildersCoordinator components, and setup some RoomStructureBuilder.

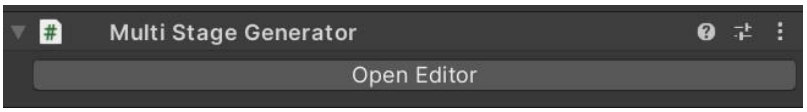


Note: ensure that Base tag is below others.

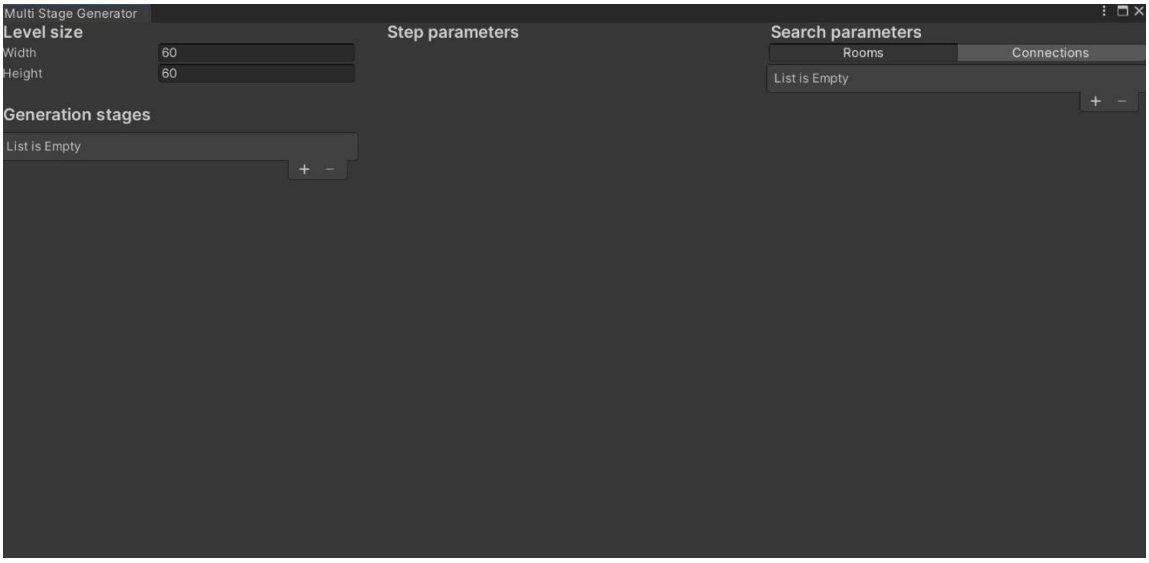
If you press Generate button - you will see big empty room.



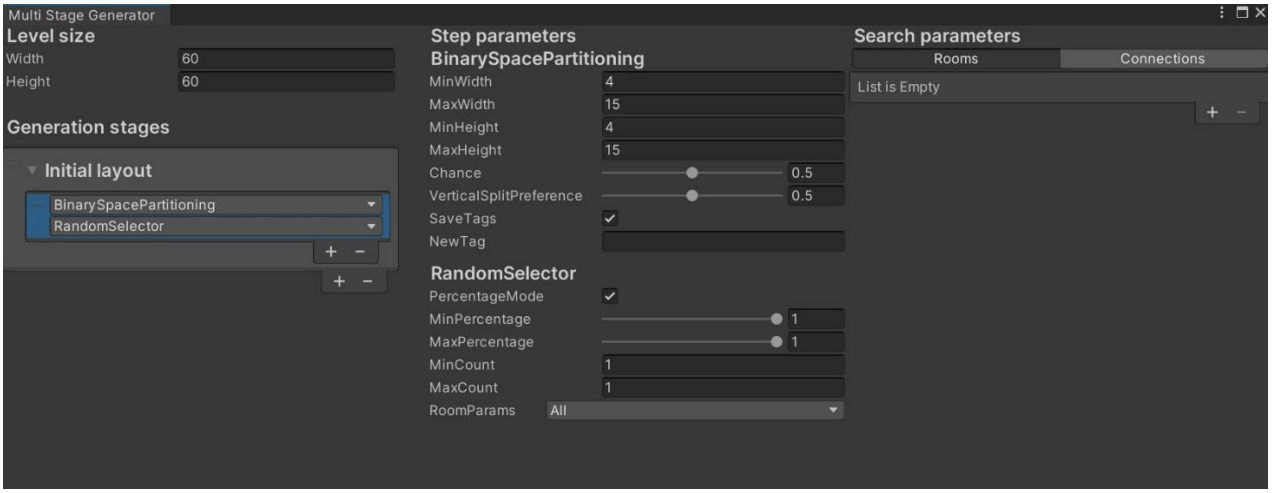
2. Open MultiStageGenerator editor.



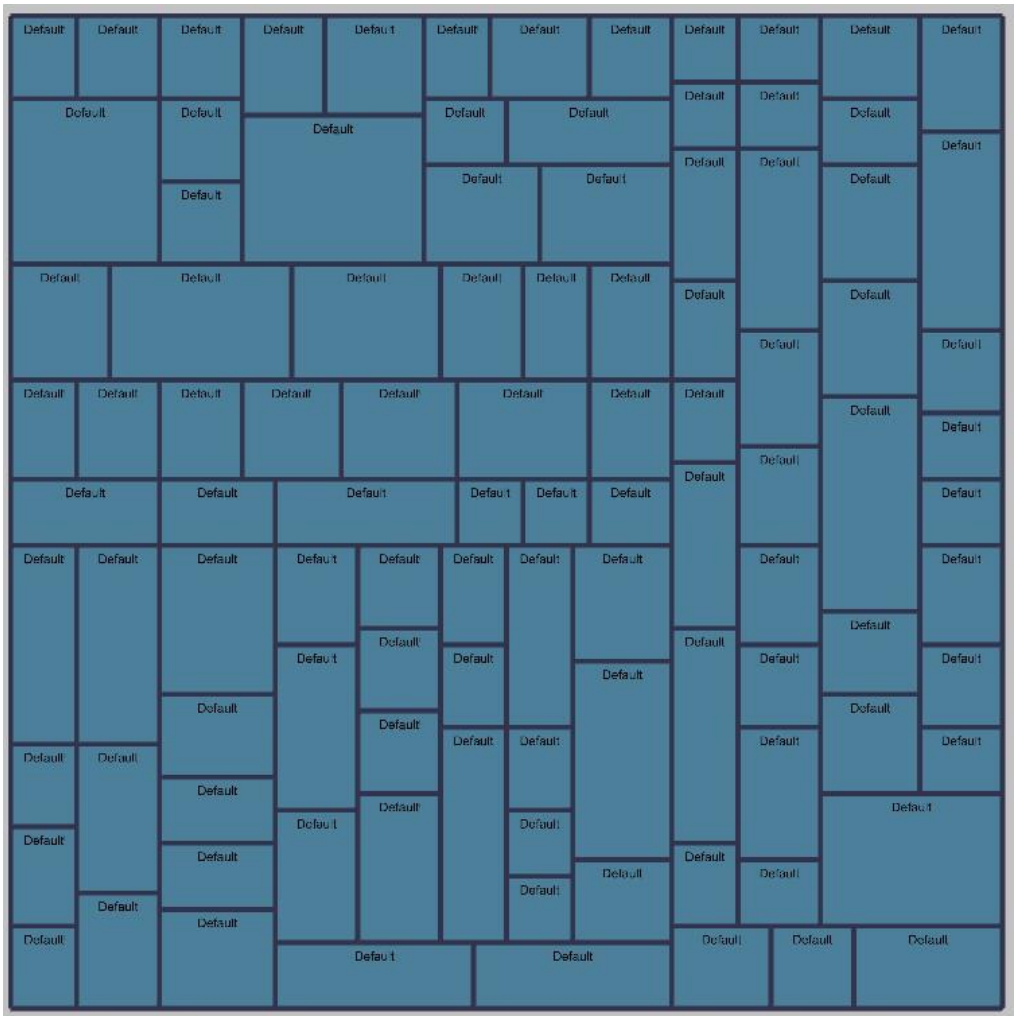
It will open editor window.



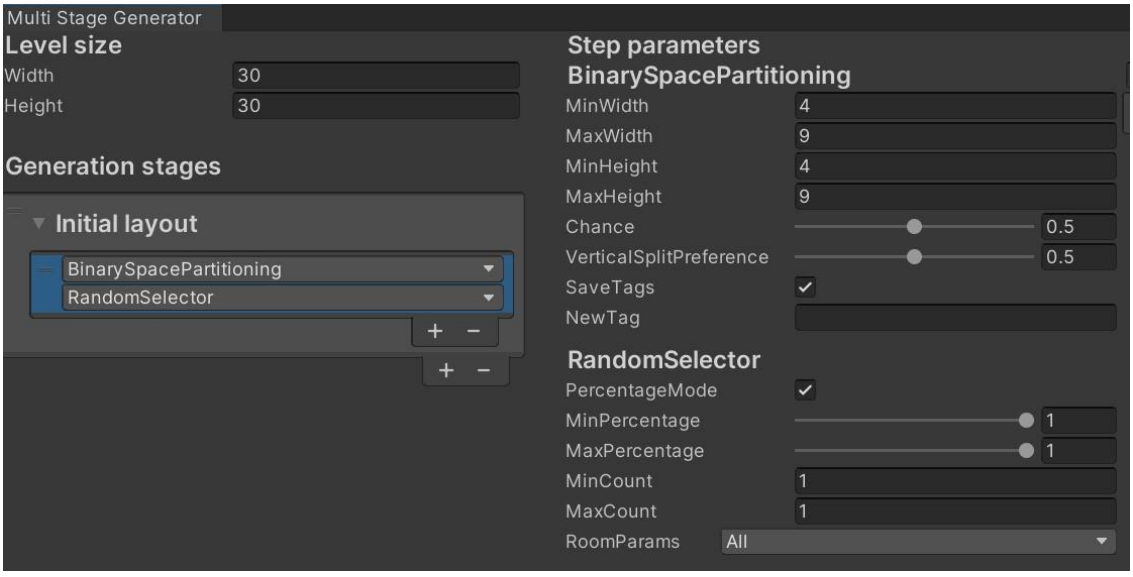
3. Add a new Generation Stage in the left part by clicking "+" button, set it's name to "Inital Layout" or something like this.
After that unfold the stage, and add it's first step by clicking "+" button.



Without changing any parameters, click Generate button - now dungeon has more smaller rooms, which were generated inside inital room.

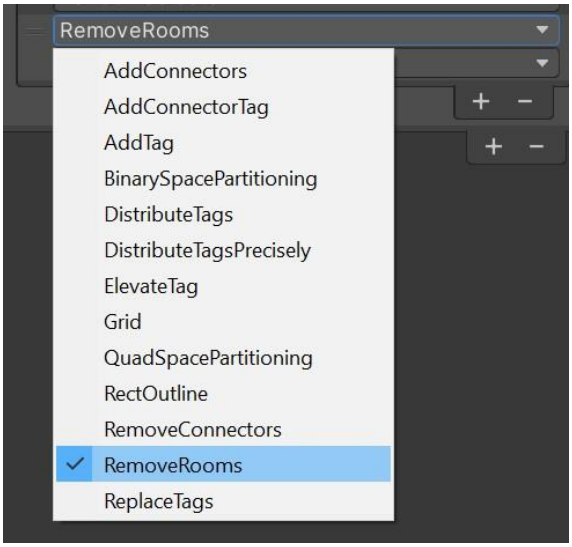


4. Change parameters as following. To open step parameters - click at it's left part.

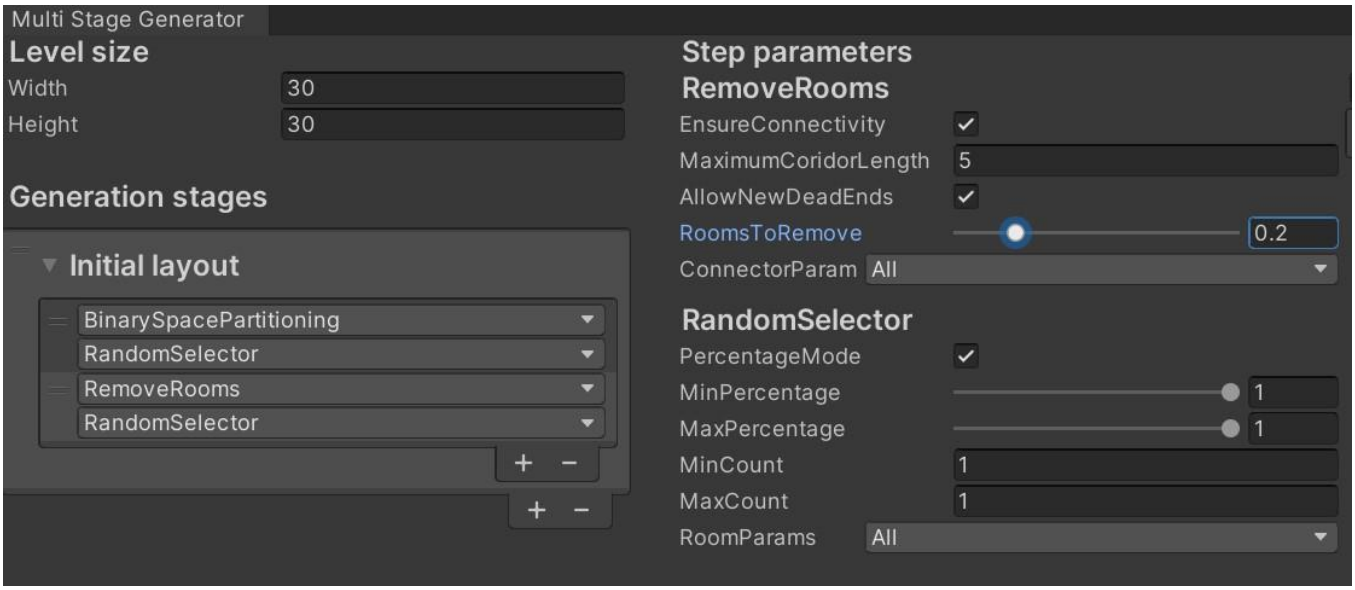


Each step contains modifier(here - BinarySpacePartitioning) and selector(RandomSelector)
Selector chooses room from layout, and then modifier takes these rooms and applies it's modification.

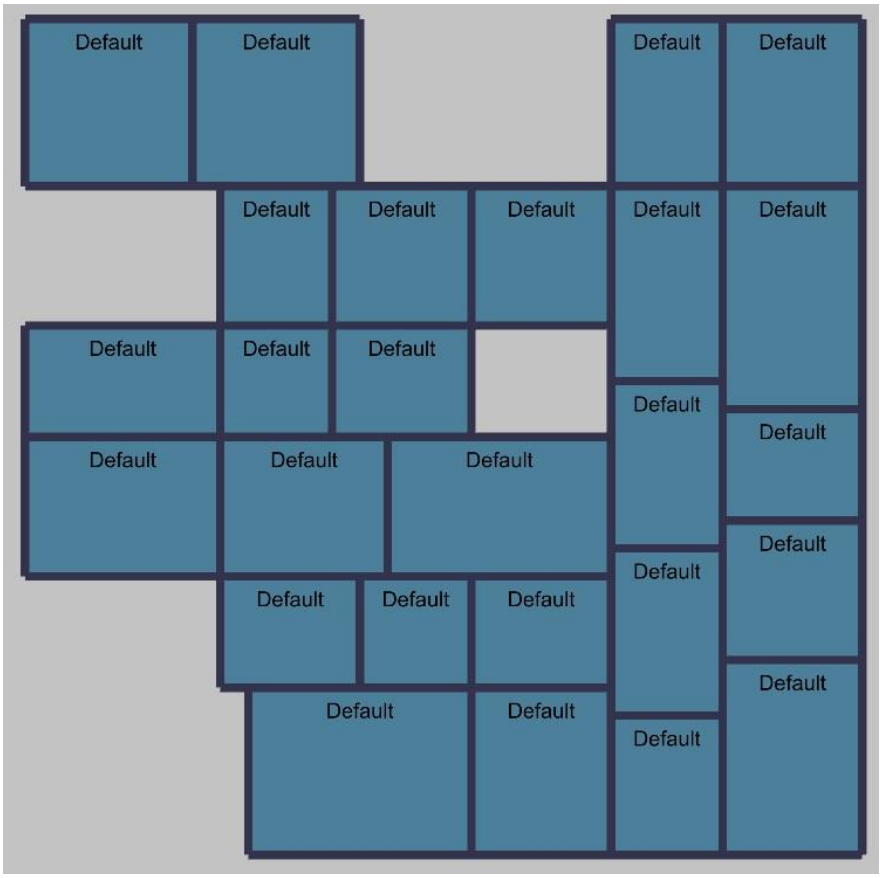
Add another step, and change it's type to RemoveRooms:



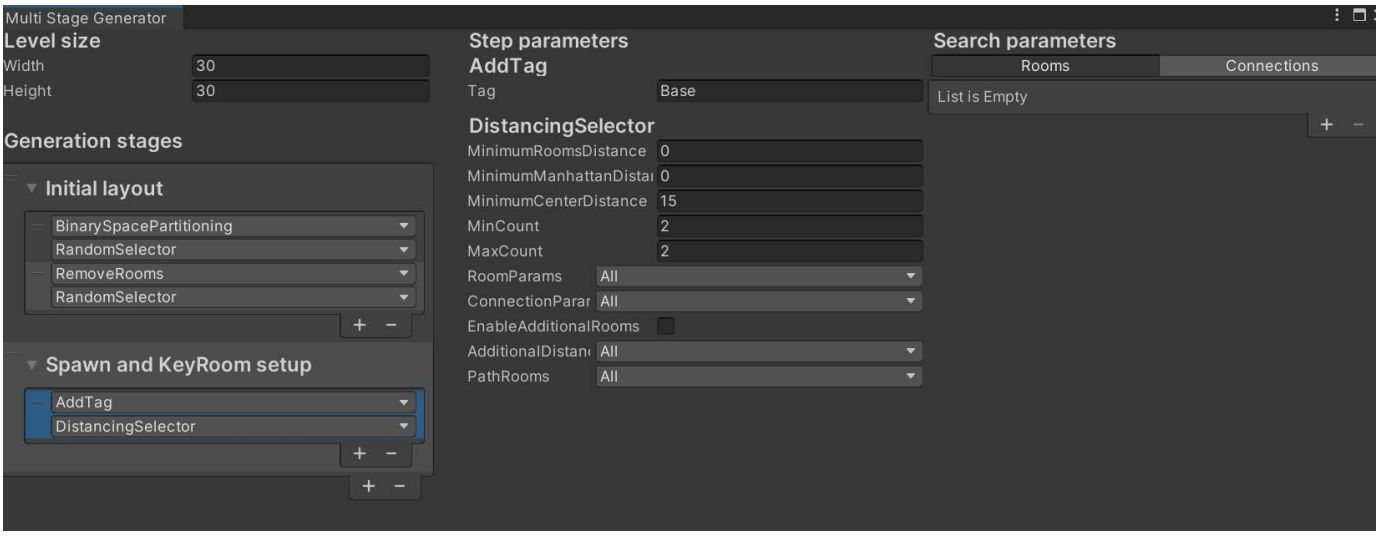
Set it's parameres as following:



Generation result will look like this:

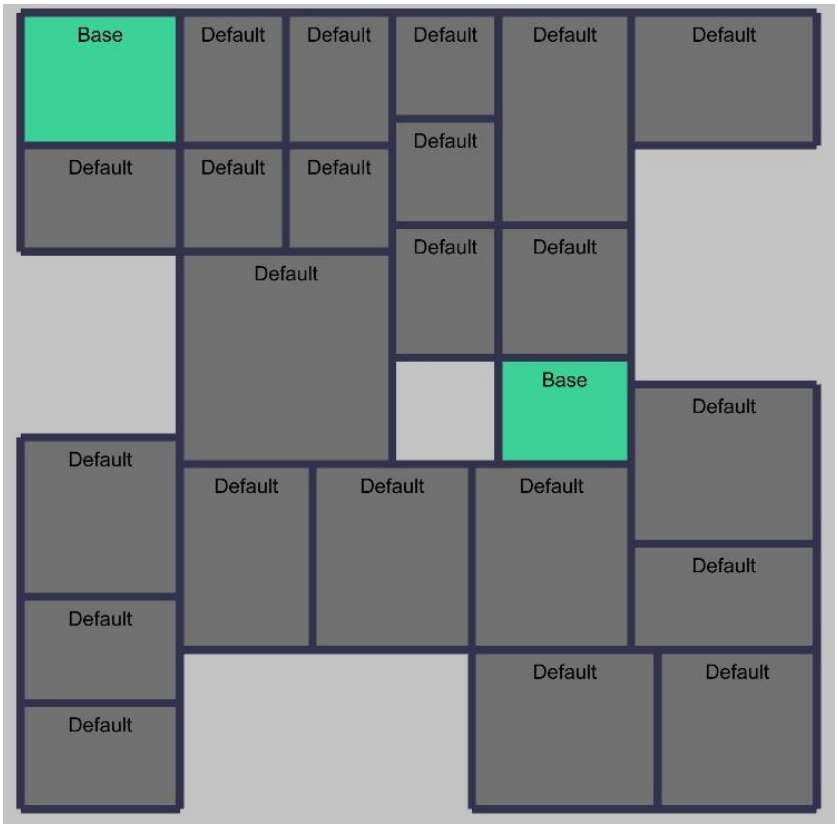


5. Add another generation stage, and first step with AddTag modifier, and DistancingSelector selector.
Set parameters as following.

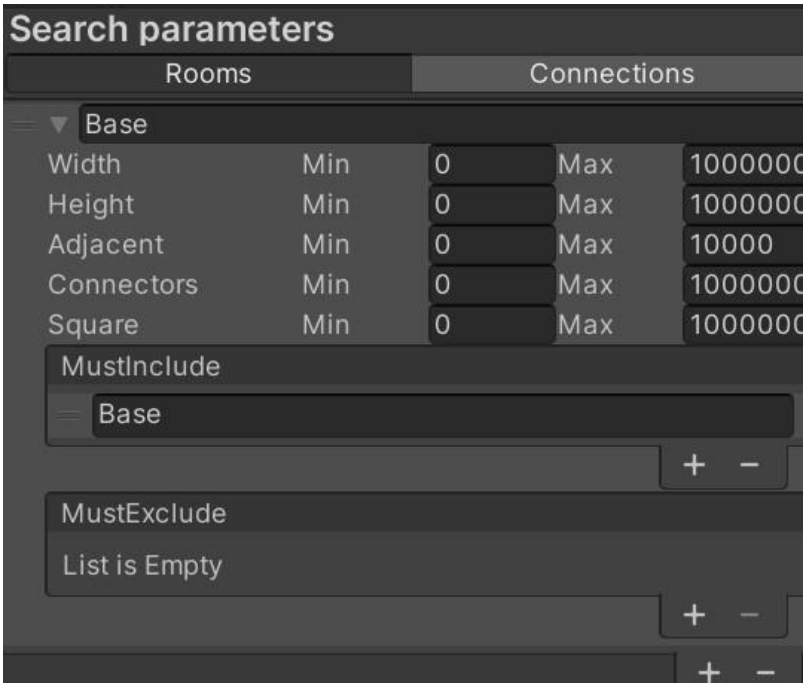


This will select 2 random rooms, with distance at least 15 between their centers,

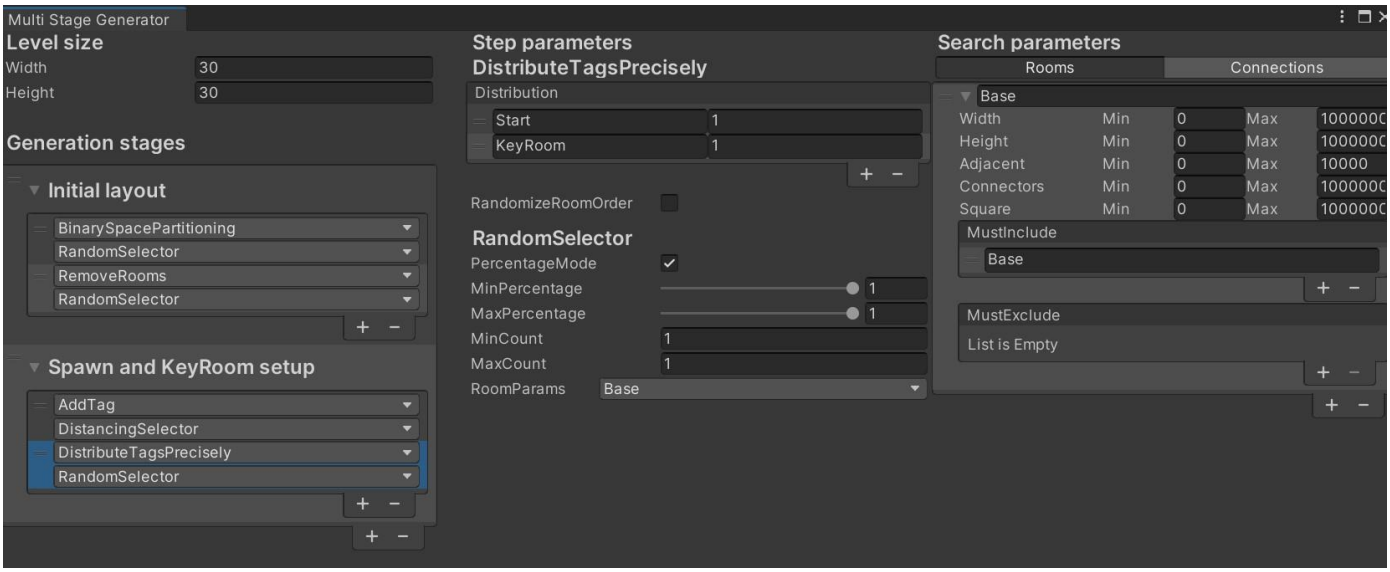
and after that modifier will add tag Base to each of them.



6. In the Search Parameters tab add a new Room, unfold it, and add "Base" to MustInclude tags:



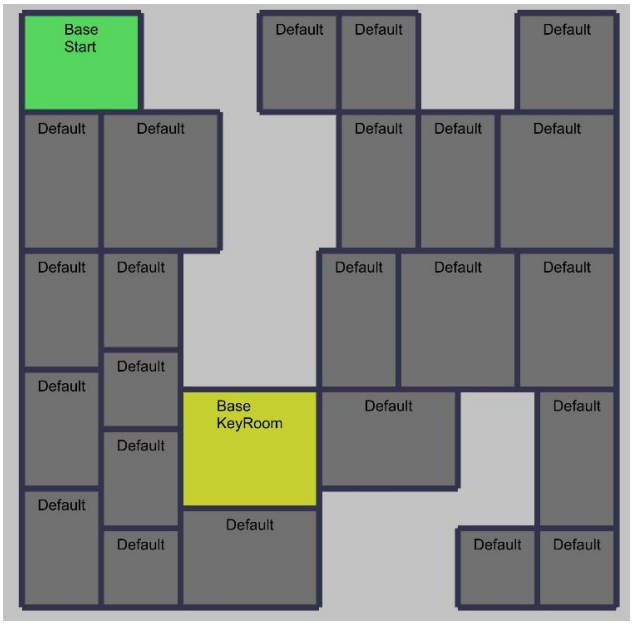
After that add a new step, with DistributeTagsPrecisely and RandomSelector, set their parameters as follows:



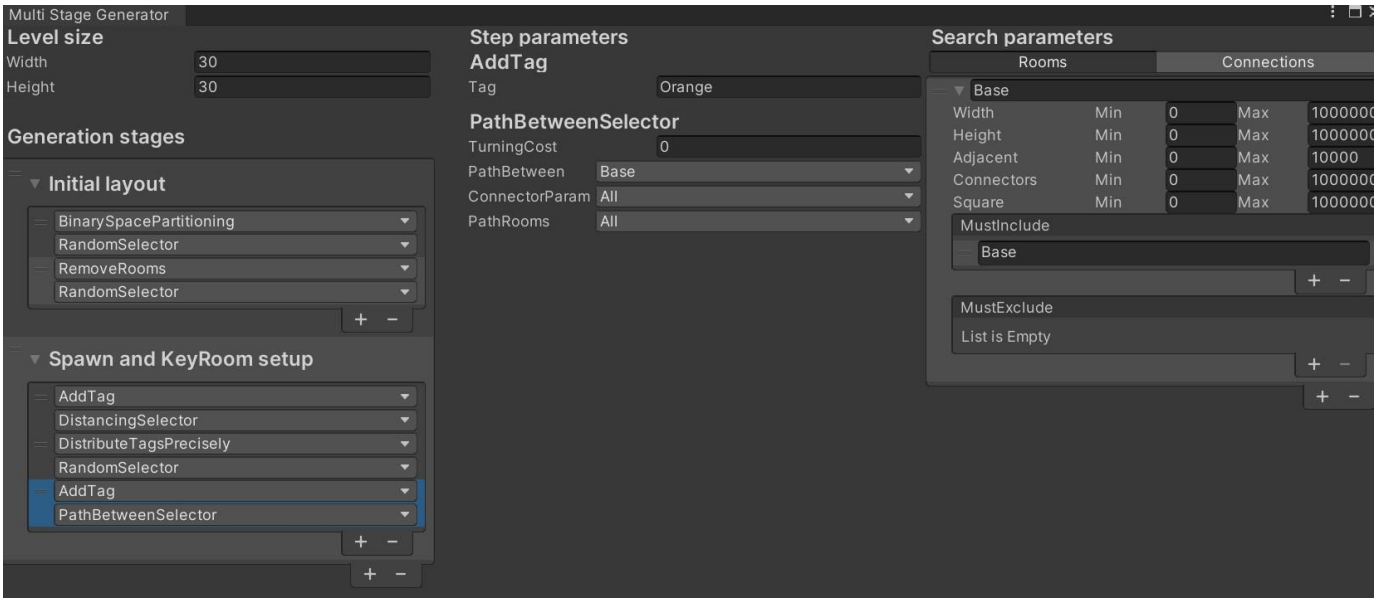
RoomParams of RandomSelector determines the parameters of rooms, which can be selected. In this case we filter all the rooms, and include only rooms with Base tag, and all 2 of them are selected. **MustInclude list means that room must contain at least 1 of the given tags.**

DistributeTagsPrecisely adds tags to the given set according with given values - 1 KeyRoom, 1 Start, total 2 rooms.

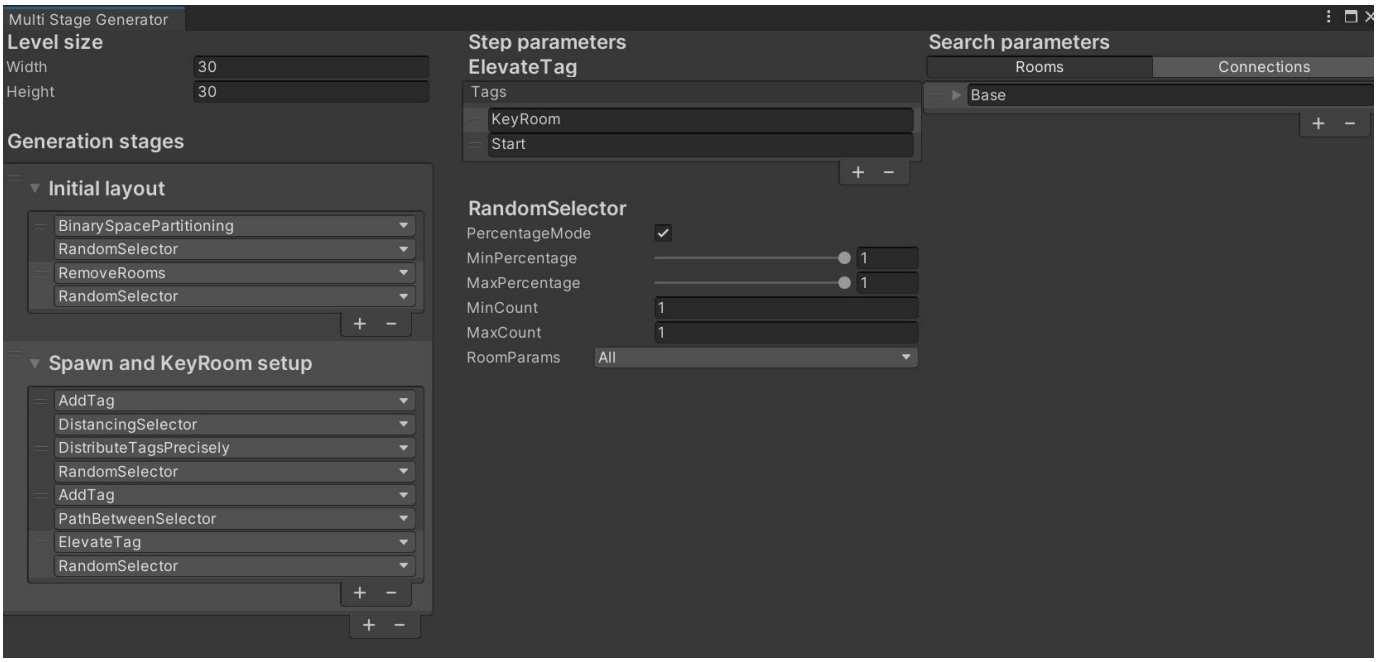
AS a result, we have 1 Start and 1 KeyRoom, and they are situated in some distance from each other, therfore player will have to find a way to get from one to another.



7. Add a new step with AddTag and PathBetweenSelector. Set Tag to Orange, and PathBetween to Base.



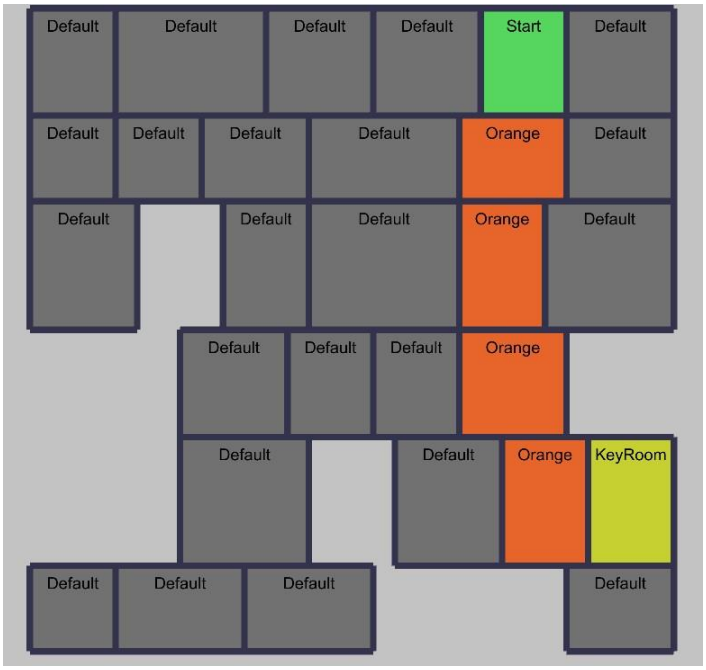
Next to it add step with ElevateTags and RandomSelecter, with "KeyRoom" and "Start" specified tags.



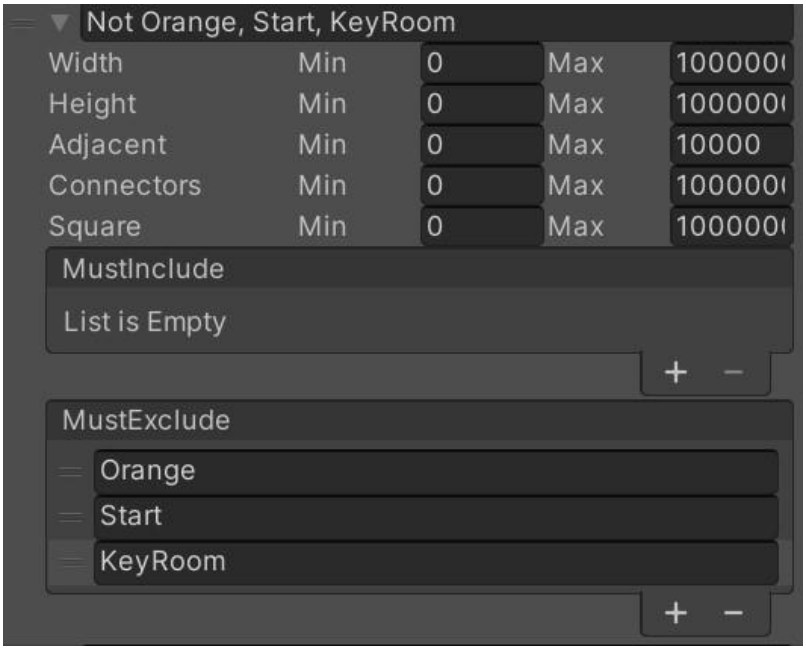
This will find a path between all rooms with "Base" tag, and then AddTag will add "Orange" to all of them, including start and end.

After that ElevateTags will examine each room, and if it has one of specified tags("KeyRoom" or "Start") - all other tags will be removed.

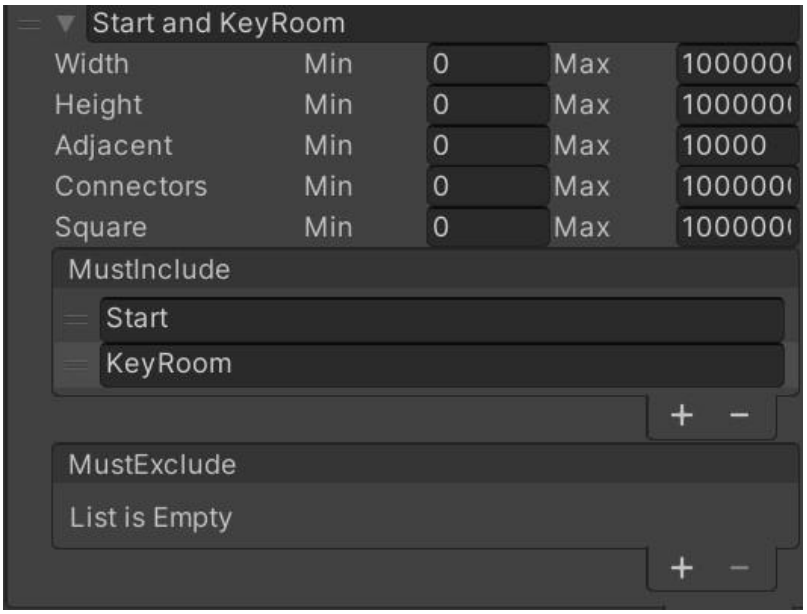
As a result, we have Start and KeyRoom, and path of Orange rooms between them.



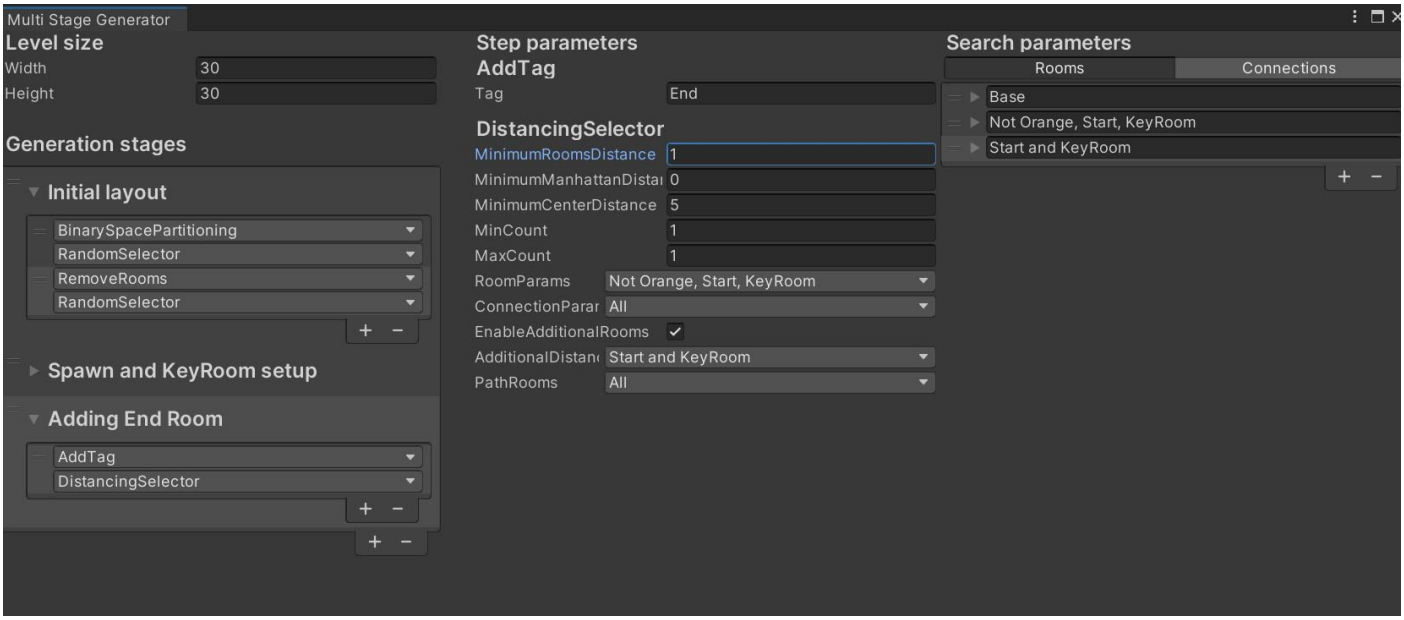
8. Add two more Rooms in SearchParameters:
One that excludes Orange, Start and KeyRoom



Another that include Start and KeyRoom.



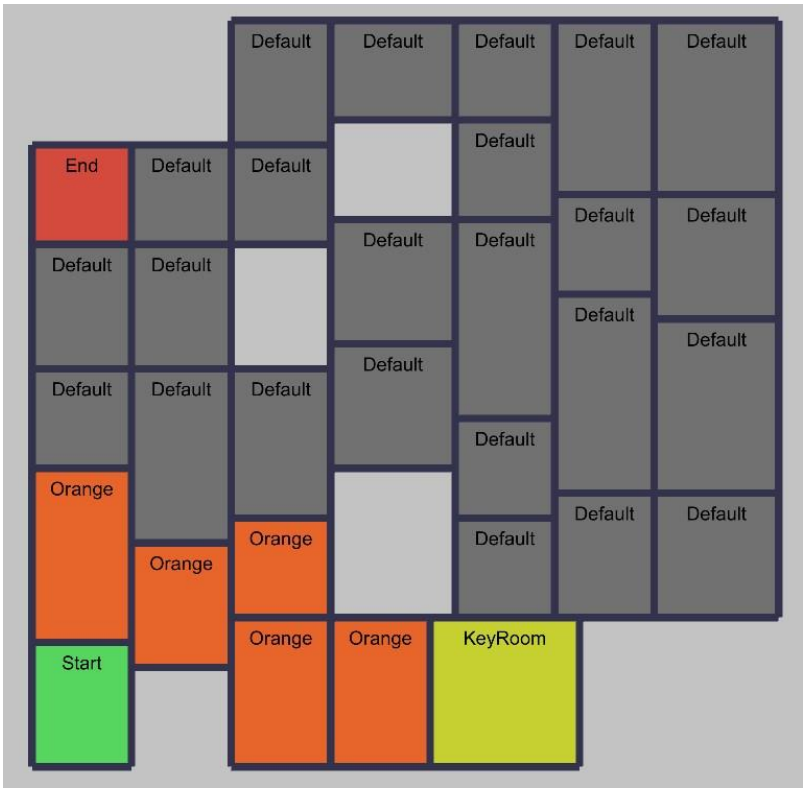
Add a new stage, and it's first step with AddTag and DistancingSelector, with parameters as follows:



This will select all rooms, that don't have Orange, Start or KeyRoom tags, and have specified distance between each other, and return only 1 of them. Additionlly, distance from selected rooms to Start and KeyRoom to all selected rooms will follow the reuiqrements too.

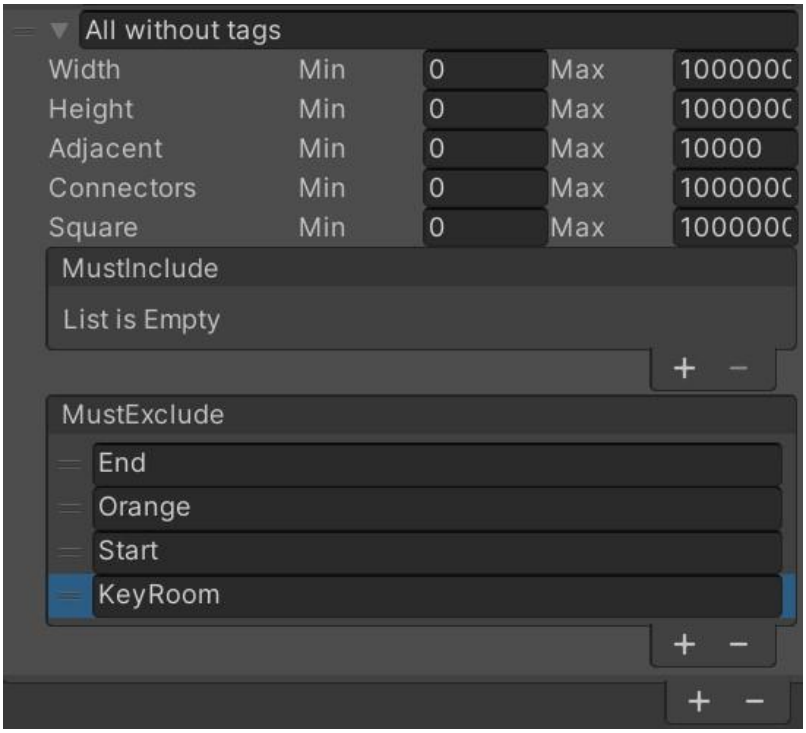
As a result, End room will not disturb any of our rooms, and will be placed in some

distance from Start and KeyRoom.



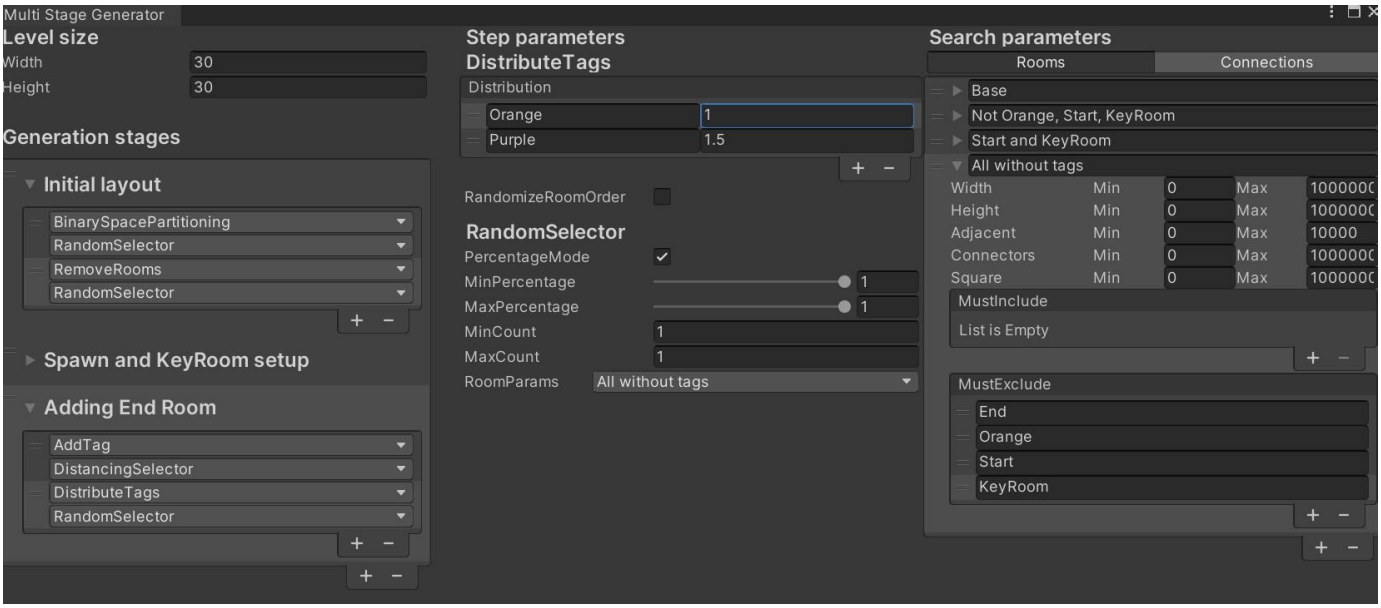
8. Now we can add tags to other rooms.

In SearchParameters create new Room, and exlude Start, End, Orange, KeyRoom.



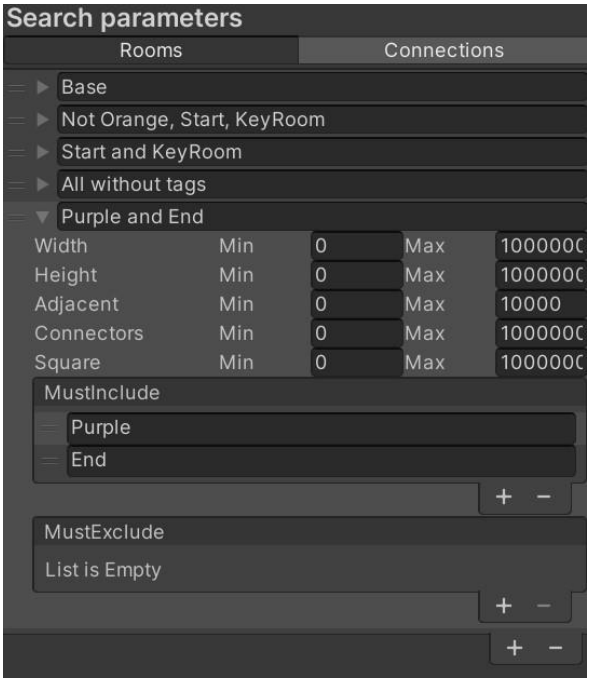
Add a new step with DistributeTags and RandomSelector. Set DistributeTags to

distribute Orange and Purple rooms, and RandomSelector to choose all rooms without tags.

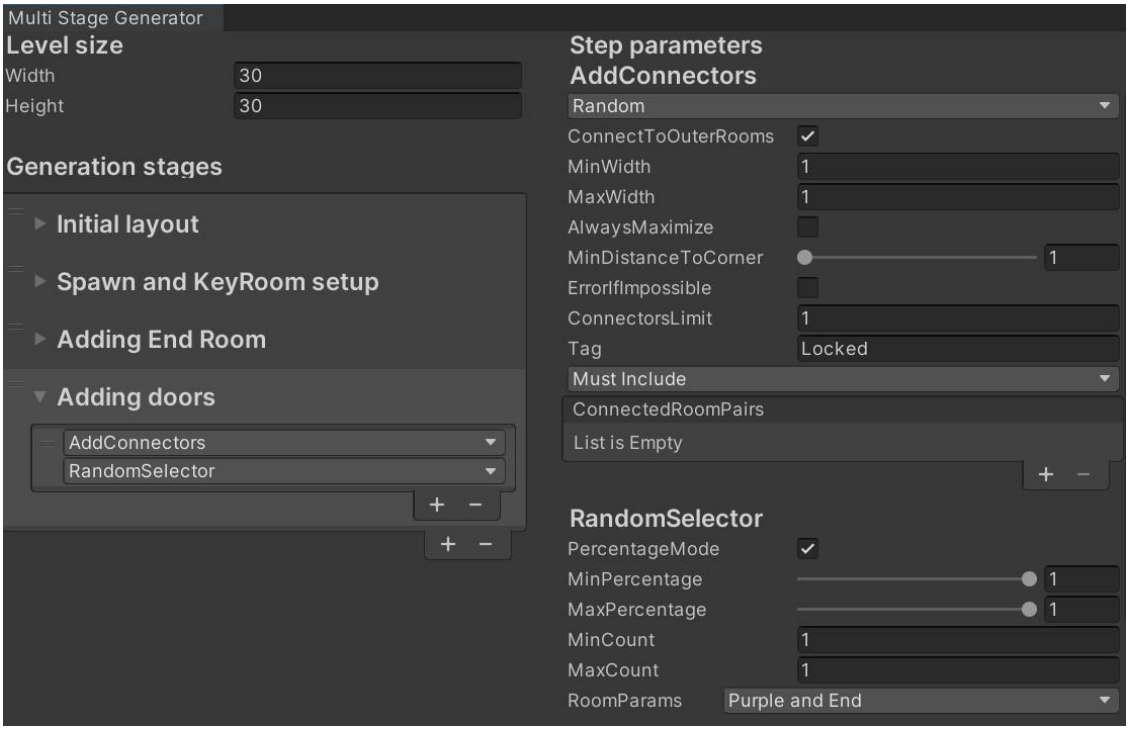


Now all unused rooms were given either Orange or Purple tag, while player still has guaranteed way from Start to KeyRoom.

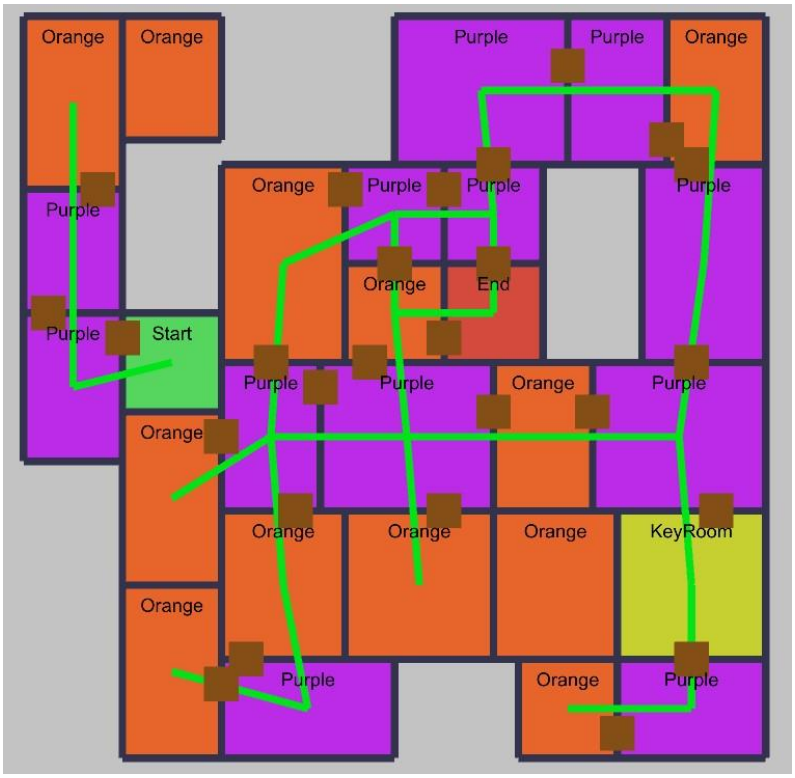
8. Add another room to Search Parameters, that includes only Purple and End rooms.



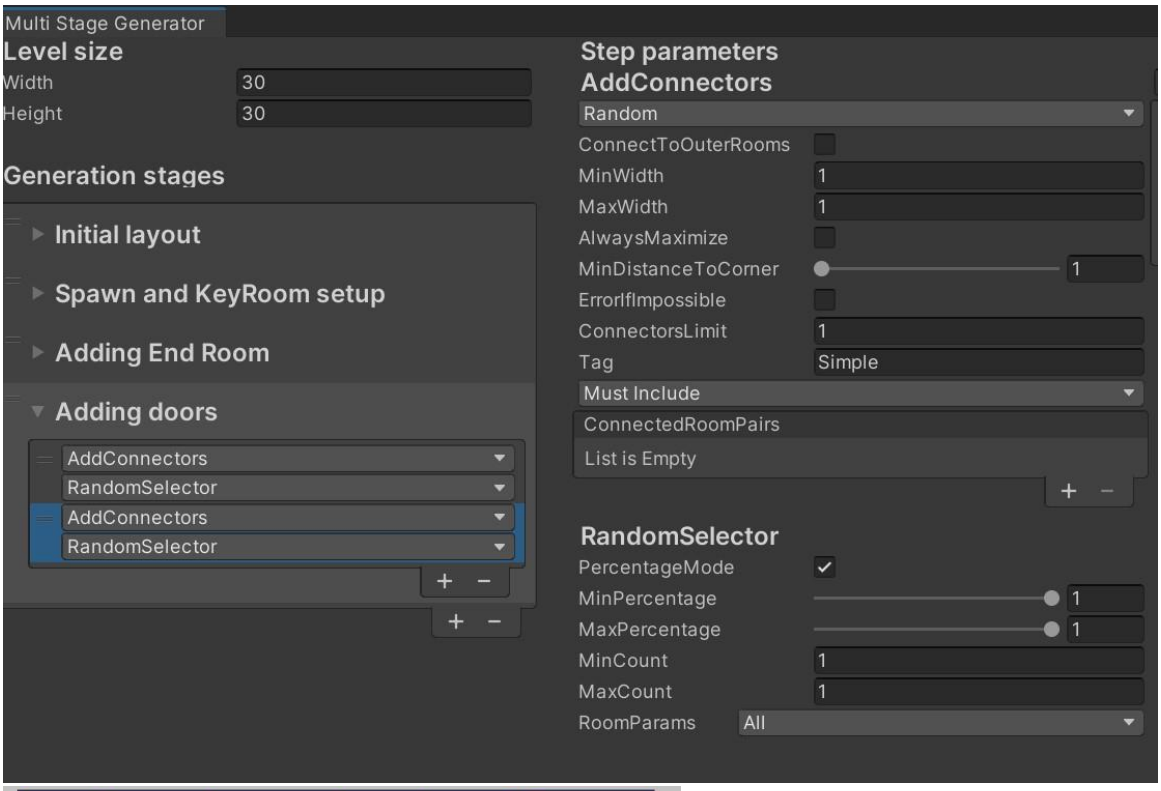
Add a new stage with AddConnectors and RandomSelector. RandomSelector must select only Purple and End rooms. Set AddConnectors Tag to Locked and check ConnetToOuterRooms

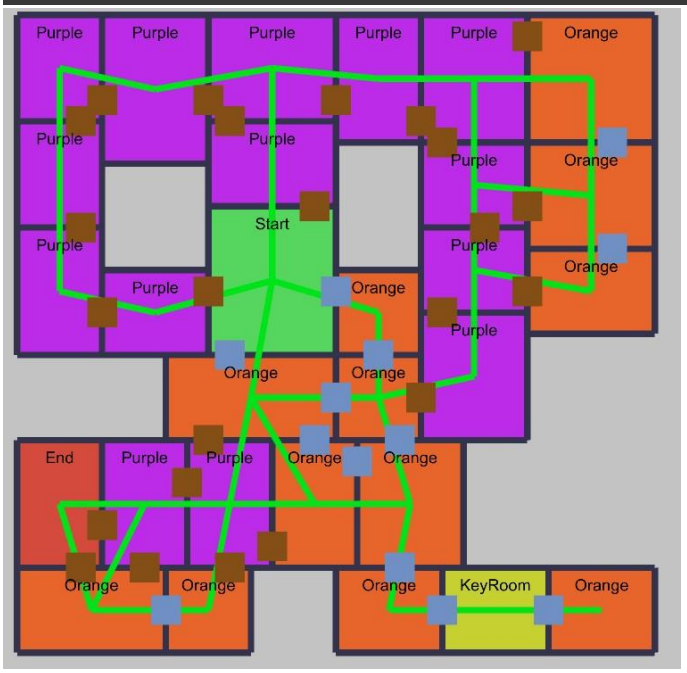


This will add connectors between all Purple and End rooms, as well as to all rooms, to which they are connected.

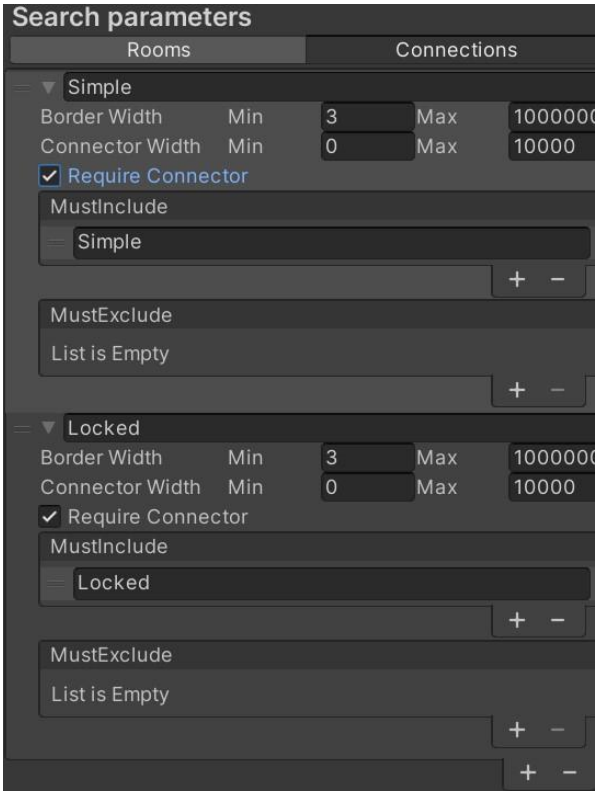


Now we can add another similar step, that adds Simple connectors to all rooms. ConnectorLimit is set to 1, so this won't add any new connectors between already conencted rooms.

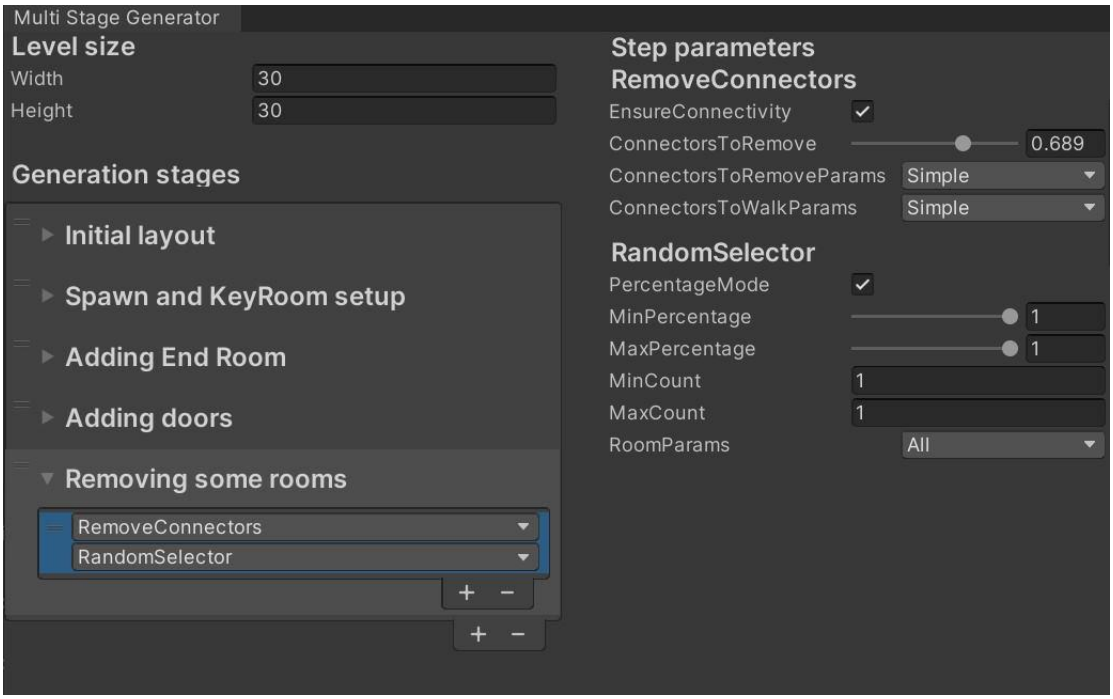

Multi Stage Generator
Level size
Width
30
Height
30
Generation stages
Initial layout
Spawn and KeyRoom setup
Adding End Room
Adding doors
AddConnectors
RandomSelector
AddConnectors
RandomSelector
Step parameters
AddConnectors
Random
ConnectToOuterRooms
MinWidth
1
MaxWidth
1
AlwaysMaximize
MinDistanceToCorner
1
ErrorIfImpossible
ConnectorsLimit
1
Tag
Simple
Must Include
ConnectedRoomPairs
List is Empty
RandomSelector
PercentageMode
MinPercentage
1
MaxPercentage
1
MinCount
1
MaxCount
1
RoomParams
All


Purple
Orange
Start
End
KeyRoom

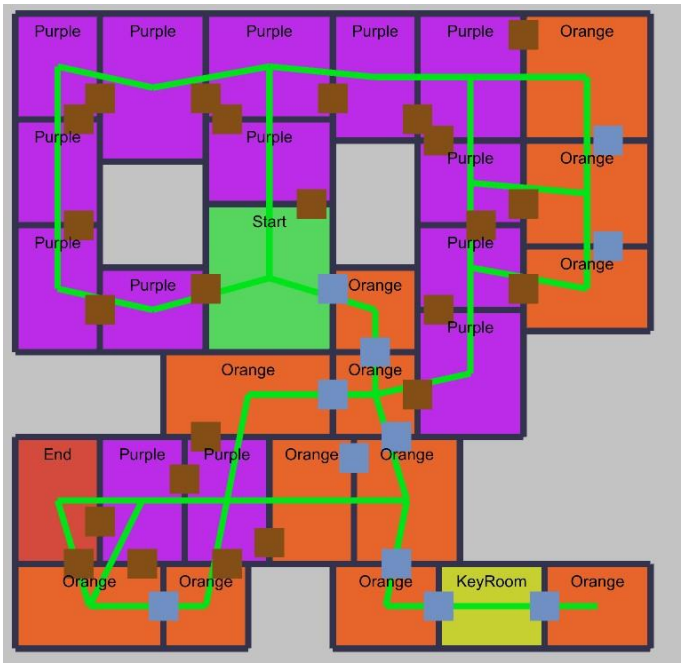
9. In Search Parameters, open Connectors tab, and add 2 new connectors. Make sure to check RequireConnector.



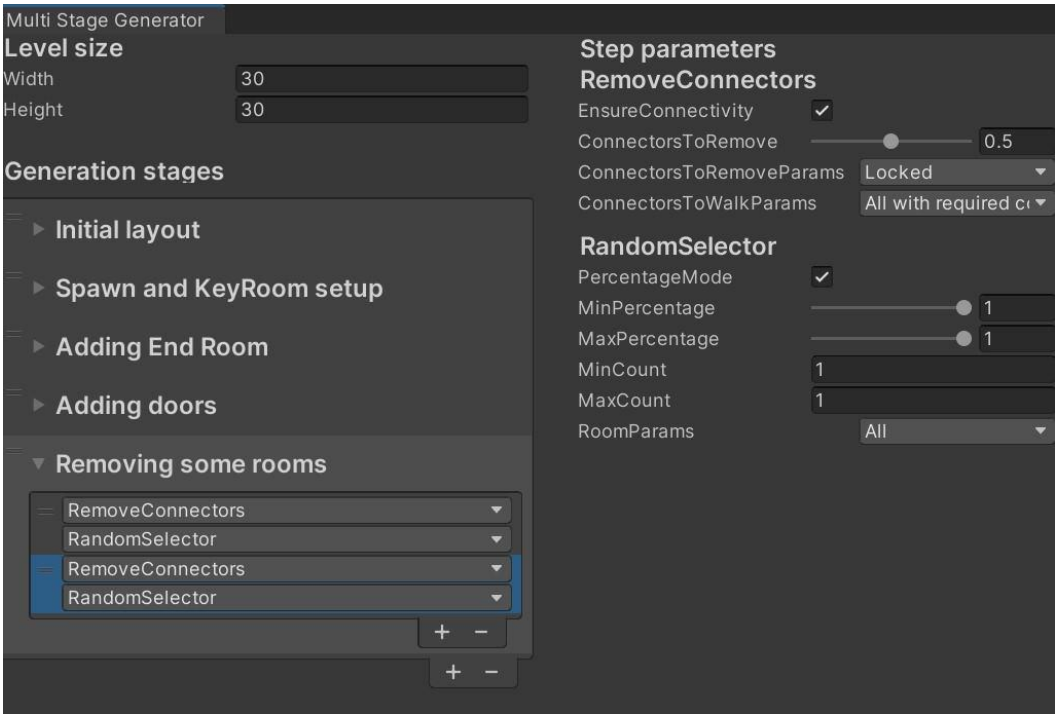
Add a new stage, with a step that has RemoveConnectors and RandomSelector. RemoveConnectors must specify Simple connectors to walk and to remove.



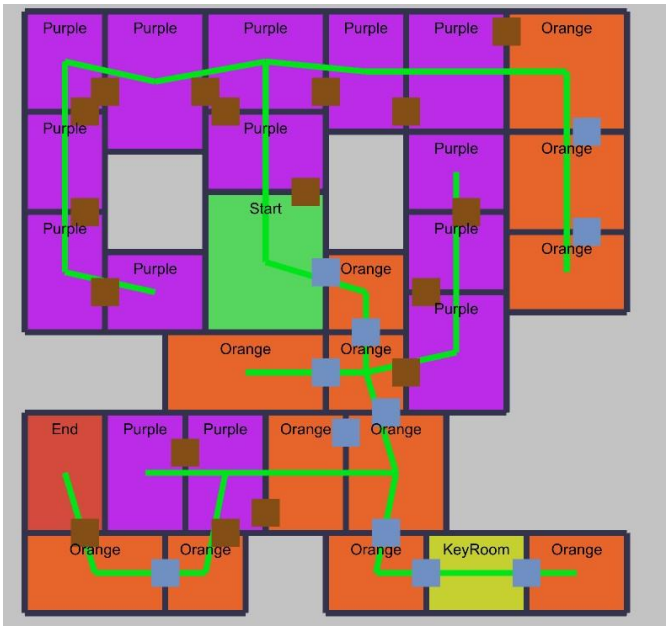
It will delete some Simple connectors, while maintaining conenctivity. In other words, if you can go from one room to another using only Simple connectors - it will not change after this modifier, when EnsureConenctivity is set, all critical connectors will remain.



Now we can also delete some of the Locked connectors. Add a similar step, with Locked ConnectorsToRemoveParams, and ConnectorsToMoveParams set to "all with required connector"



In this case much more connectors will be removed, since we can also use Simple connectors to move between rooms.



As a result, we have dungeon, where player can spawn in Start room, only with access to Simple(unclodked) door. While walking through orange doors, he can come across a keyRoom, and obtain key to unlock all Locked doors, and get to the End room.

You can learn more about Modifiers and Selectors in their documentation files.