Warmerise | Red vs Blue

CUSTOM MAP TUTORIAL

Introduction

Custom map system allow players to create their own maps in Unity 3D and upload them to the game.

This tutorial features a step-by-step guide on how to setup a new map, so it can be used in the game.

Requirements:

- Unity 3D Game Engine (version equal or above 2019) (https://unity3d.com/)
- Understanding Of Unity 3D And Level Design

The following topics will be covered:

Installing The Export Package

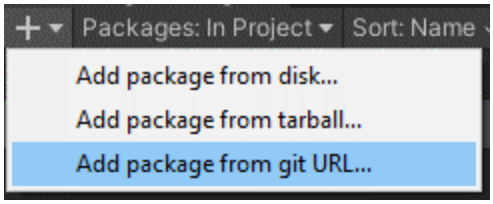
Skip this step if you already have the export package installed

Export package is a combination of necessary scripts and files that are needed to prepare/export the map.

The package is hosted on GitHub.

Below are steps to install it using Unity's Package Manager:

- In your Unity project go to Window -> Package Manager
- In the top left corner click (+) symbol -> Add package from git URL…



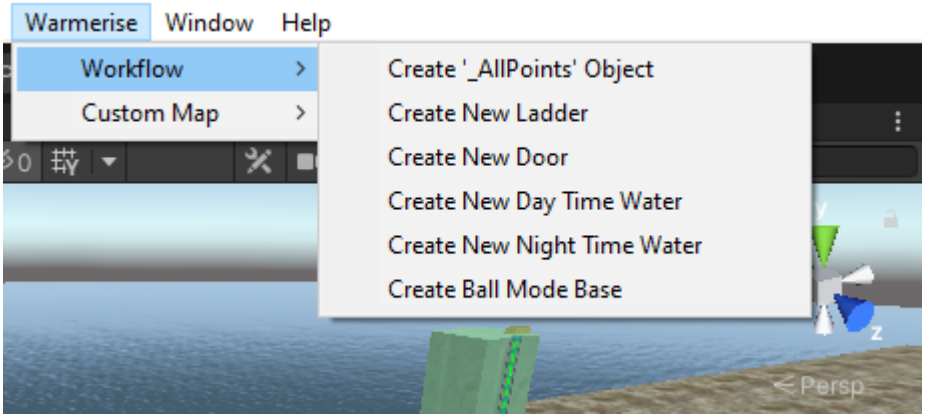
- Paste the link below and click Add

  https://github.com/nsdesigngames/WarmeriseMapExport.git

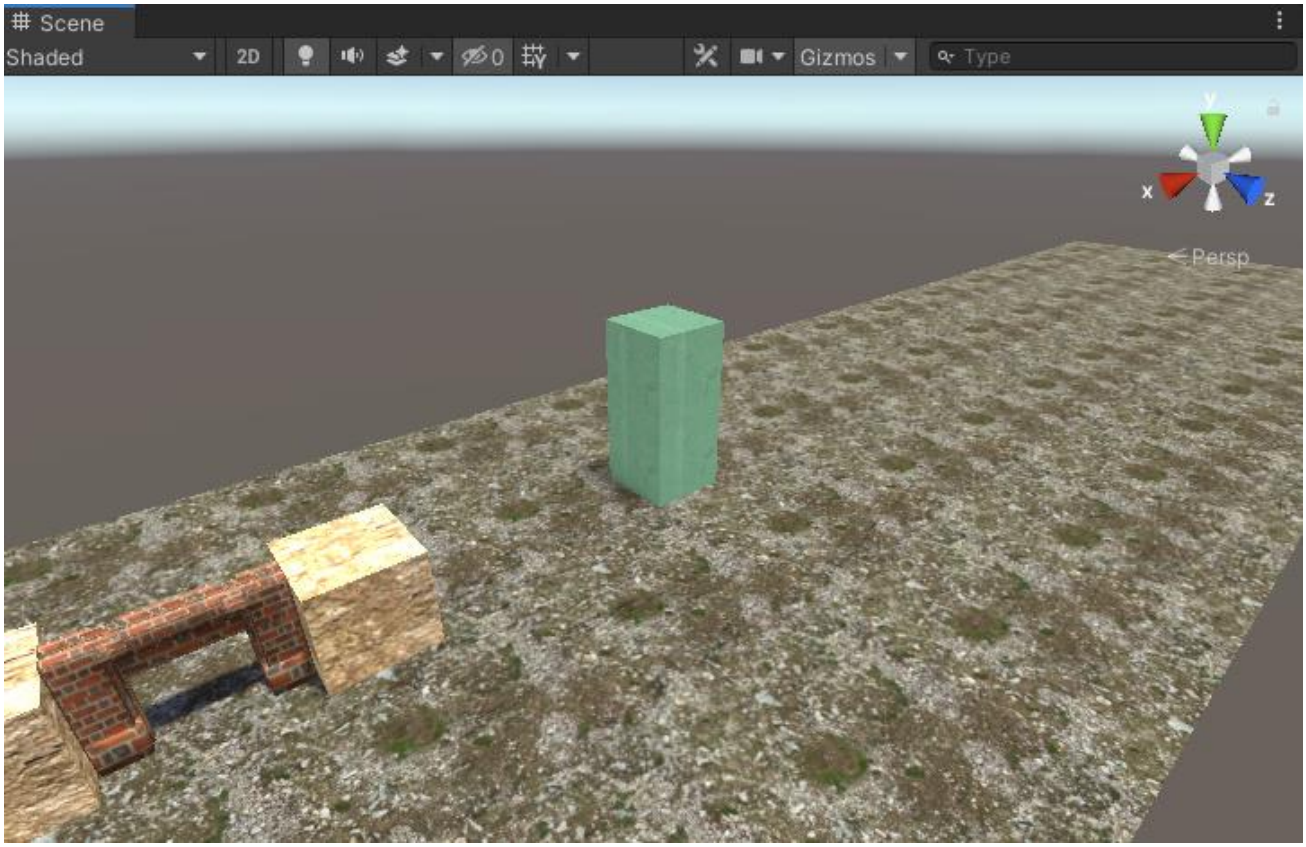
  If there are no errors in console, the package should now be installed in your project.

Placing Special Objects In The Map

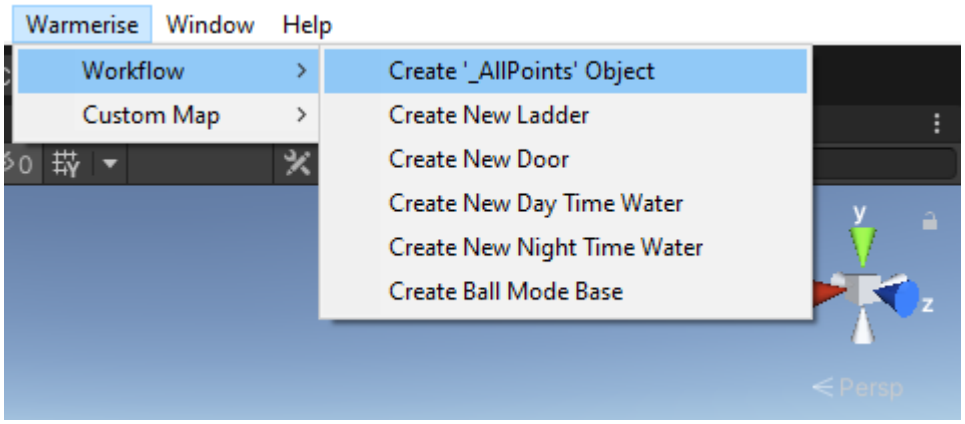
If you installed the export package correctly, the new menu item with the name ‘Warmerise’ should appear, which contains all the options needed to prepare and export the map:

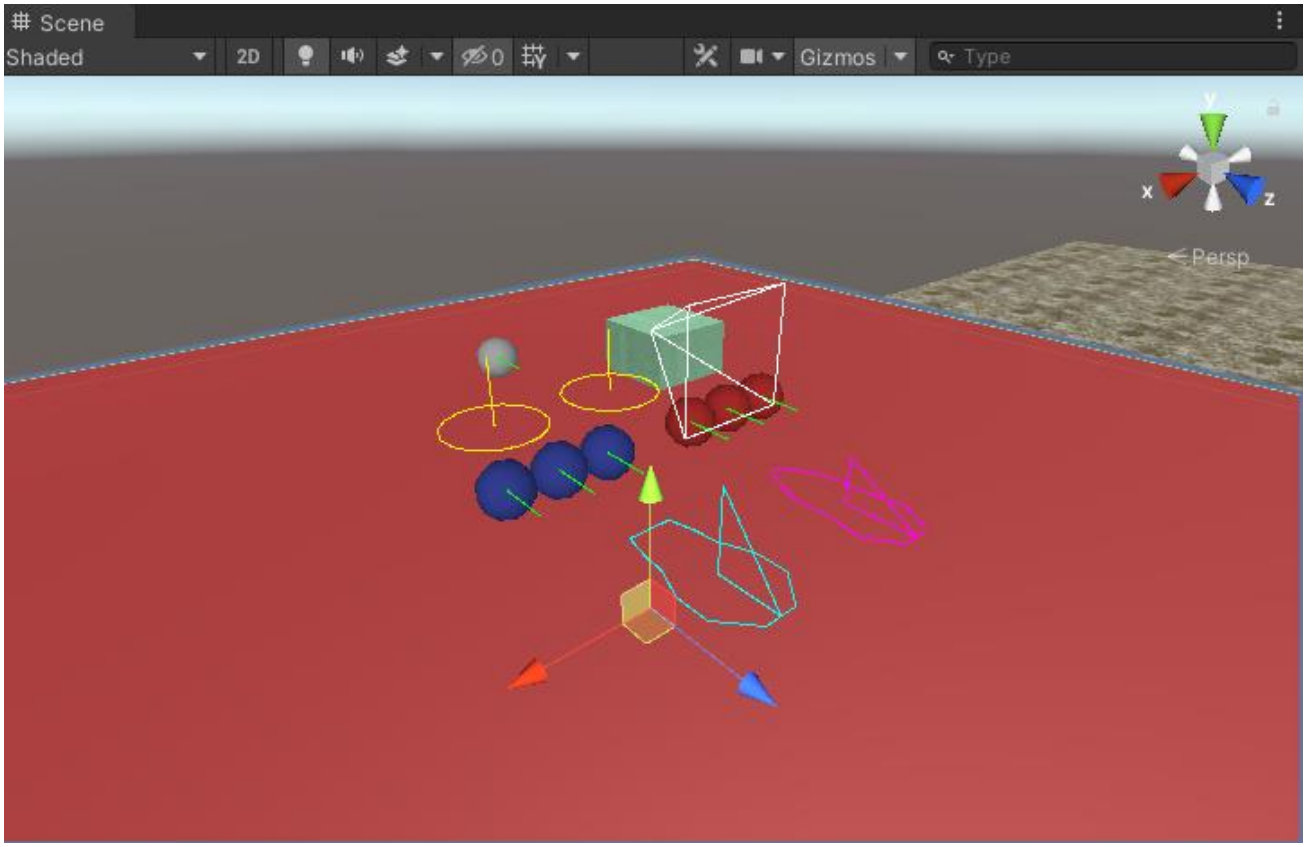


- Open Scene with your map

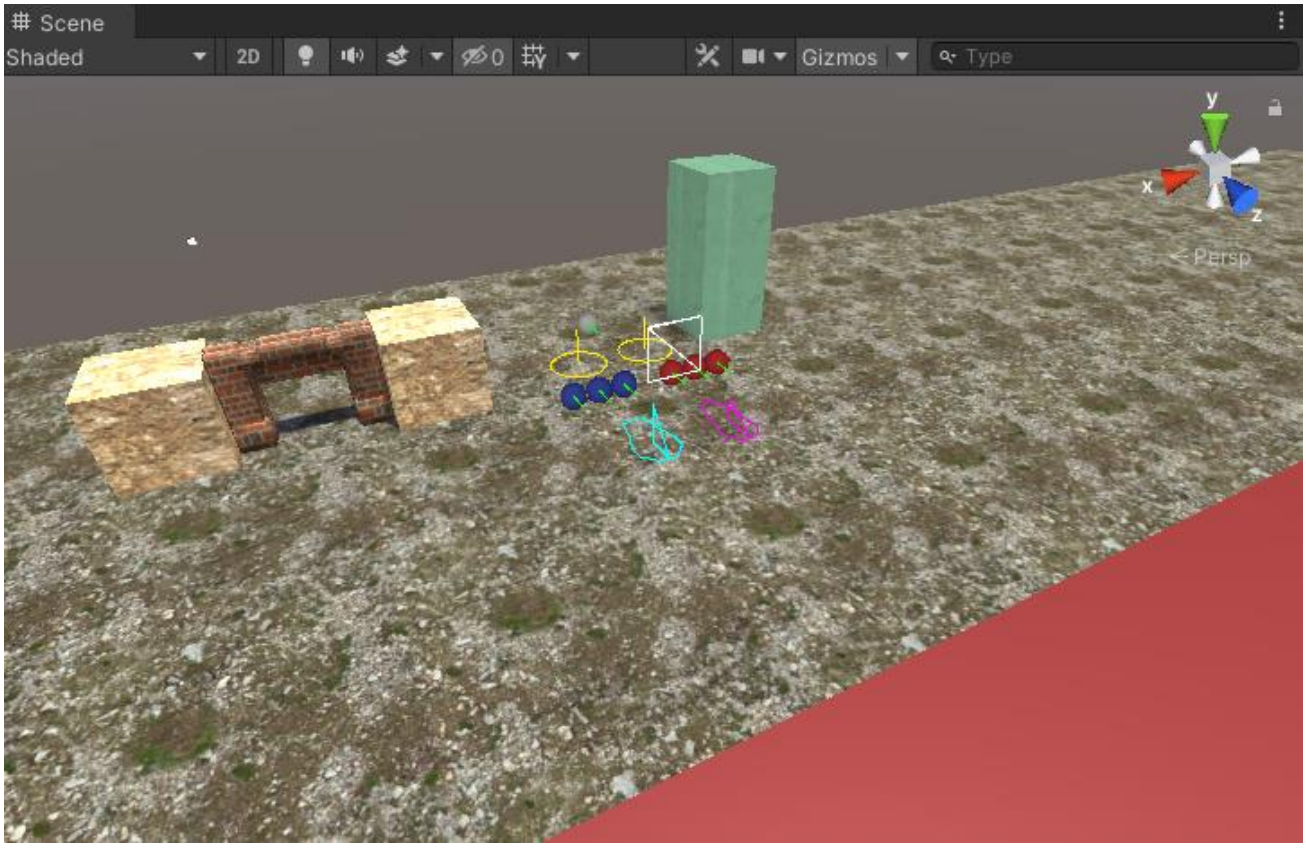


- Create special object by clicking Warmerise -> Workflow -> Create '_AllPoints' Object

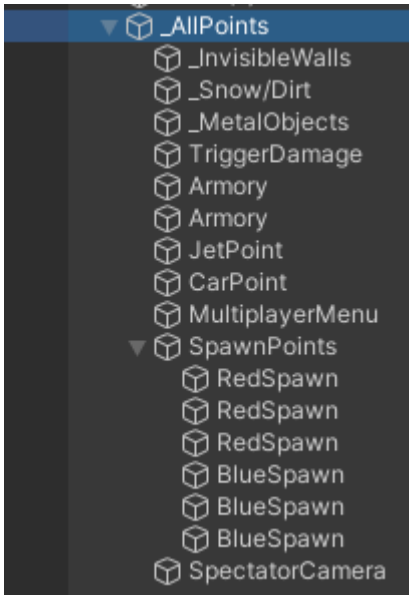




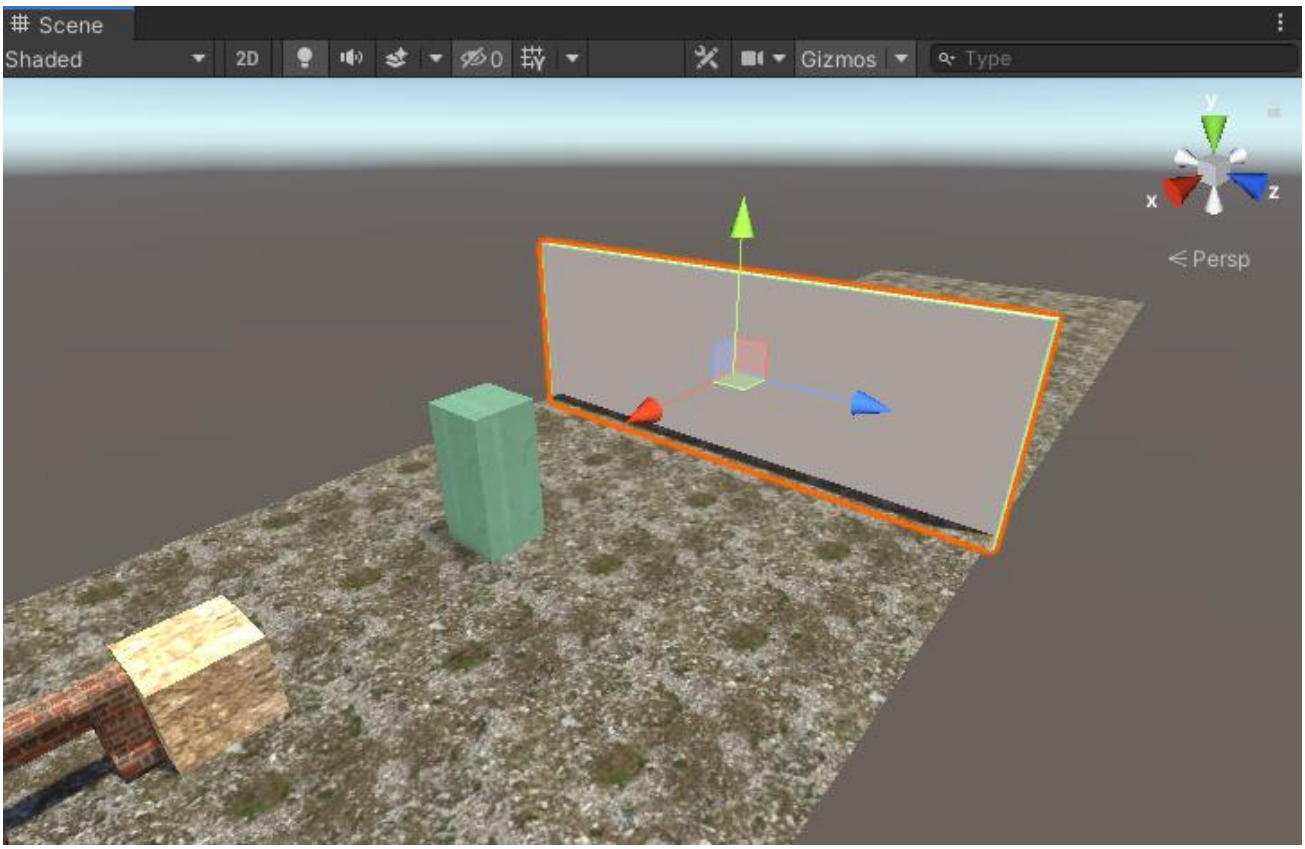
- Adjust the object position until it matches the surface of the map:



- Looking at the hierarchy view, you can see that the object '_AllPoints' has many objects inside it, each with its own purpose:



_InvisibleWalls – this object should contain the objects with collider component, which will be used as an invisible wall to limit the boundaries of the map. [This Object Can Be Removed But Should Not Be Duplicated]



_Snow/Dirt – this object should contain map objects that need to exhibit snow/dirt surface behavior (predominately used for detecting bullet hit particles and/or footstep sounds). [This Object Can Be Removed But Should Not Be Duplicated]

_MetalObjects – this object should contain map objects that need to exhibit metal surface behavior (predominately used for detecting bullet hit particles and/or footstep sounds). [This Object Can Be Removed But Should Not Be Duplicated]

TriggerDamage – this object inflicts damage to players on contact. One of the uses, is placing it under the map, so players who fall from the boundaries are instantly killed. [This Object Can Be Duplicated or Removed]

Armory – point where the Armory Spot will be placed in the game (good practice is placing at least one armory spot for each team, or having just one armory spot in the middle of the map) [This Object Can Be Duplicated or Removed Until 1 is Remaining]

JetPoint – position where Jet Ship will be spawned [This Object Can Be Duplicated or Removed]

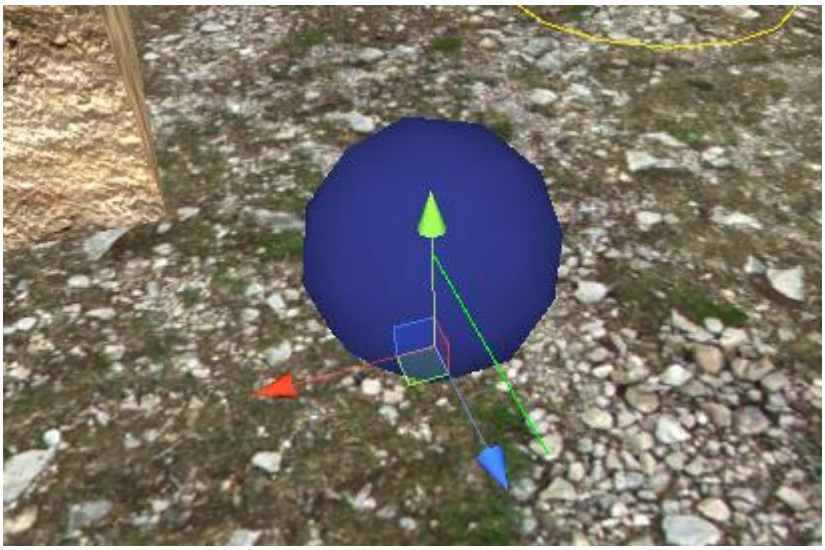
CarPoint – position where Humvee will be spawned [This Object Can Be Duplicated or Removed]

MultiplayerMenu – position where the welcome camera will be placed (for testing, move the Main Camera inside it and change its position and rotation to (0, 0, 0)). [This Object Should Not Be Duplicated or Removed]

NOTE: You might be tempted to place an image in front of the Camera, however it's better to point the Welcome Camera directly at the map, to give players a better preview.

SpawnPoints – object that contains spawn points for both teams [This Object Should Not Be Duplicated or Removed]

RedSpawn / **BlueSpawn** – spawn points for a Red Team and Blue Team respectively (the green line from the center indicates which direction the player will face during the spawn) [This Object Can Be Duplicated or Removed Until 3 are remaining]



NOTE: Ideally, try to place at least 8 spawn points per team and make sure they are 'spaced out' enough to prevent the crowded spawns, which could lead to spawn killing.

SpectatorCamera – point where flying spectator will be spawned (green line indicates the direction that the spectator will face during the spawn) [This Object Should Not Be Duplicated or Removed]

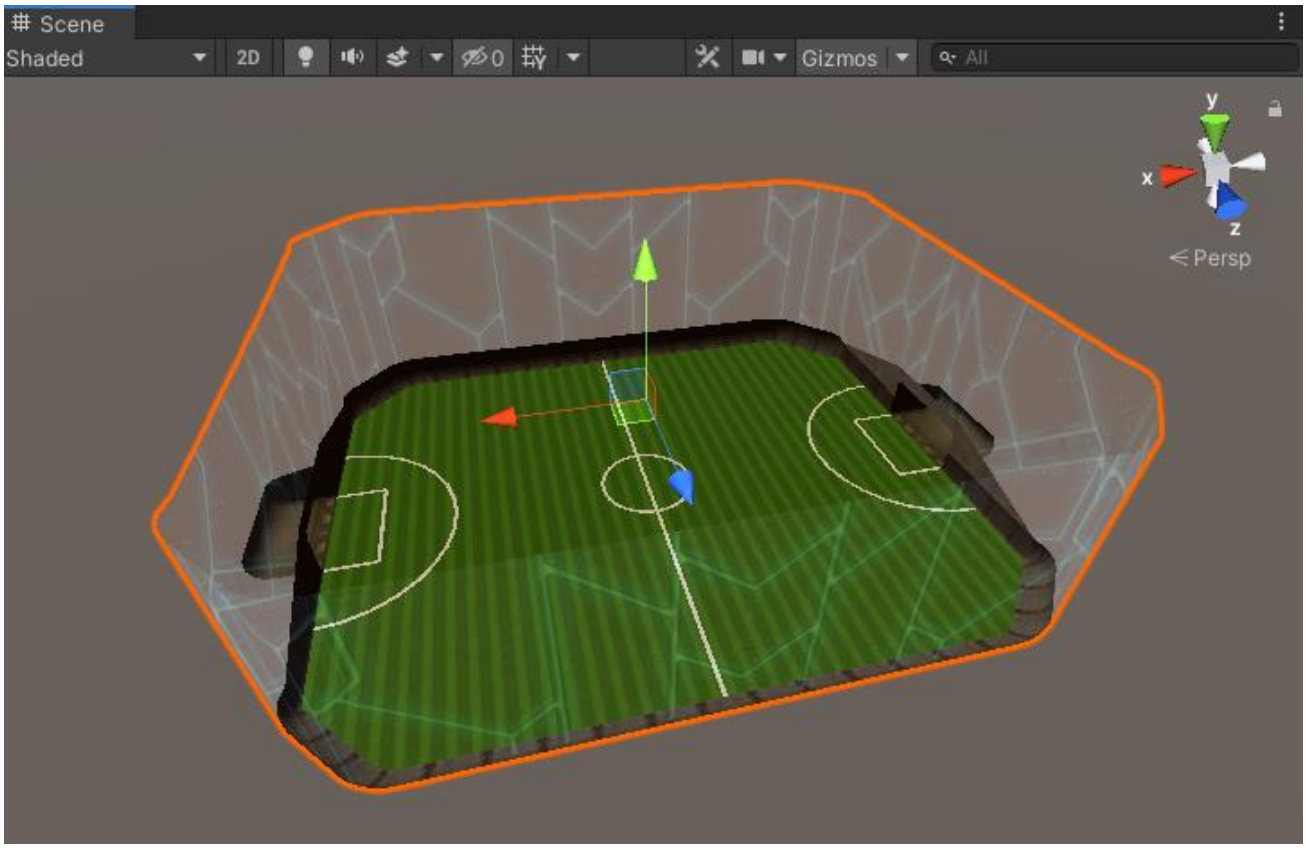
- Once the key objects are placed, the map is ready to be exported and tested in the game.

BALL MODE SPECIAL OBJECTS

Creating a map for Ball mode requires an additional key object, which is '_BallModeBaseMap'.

This object provides a layout reference, so you can see where the goals will be placed.

- Create new Scene
- Create base layout by clicking Warmerise -> Workflow -> Create Ball Mode Base

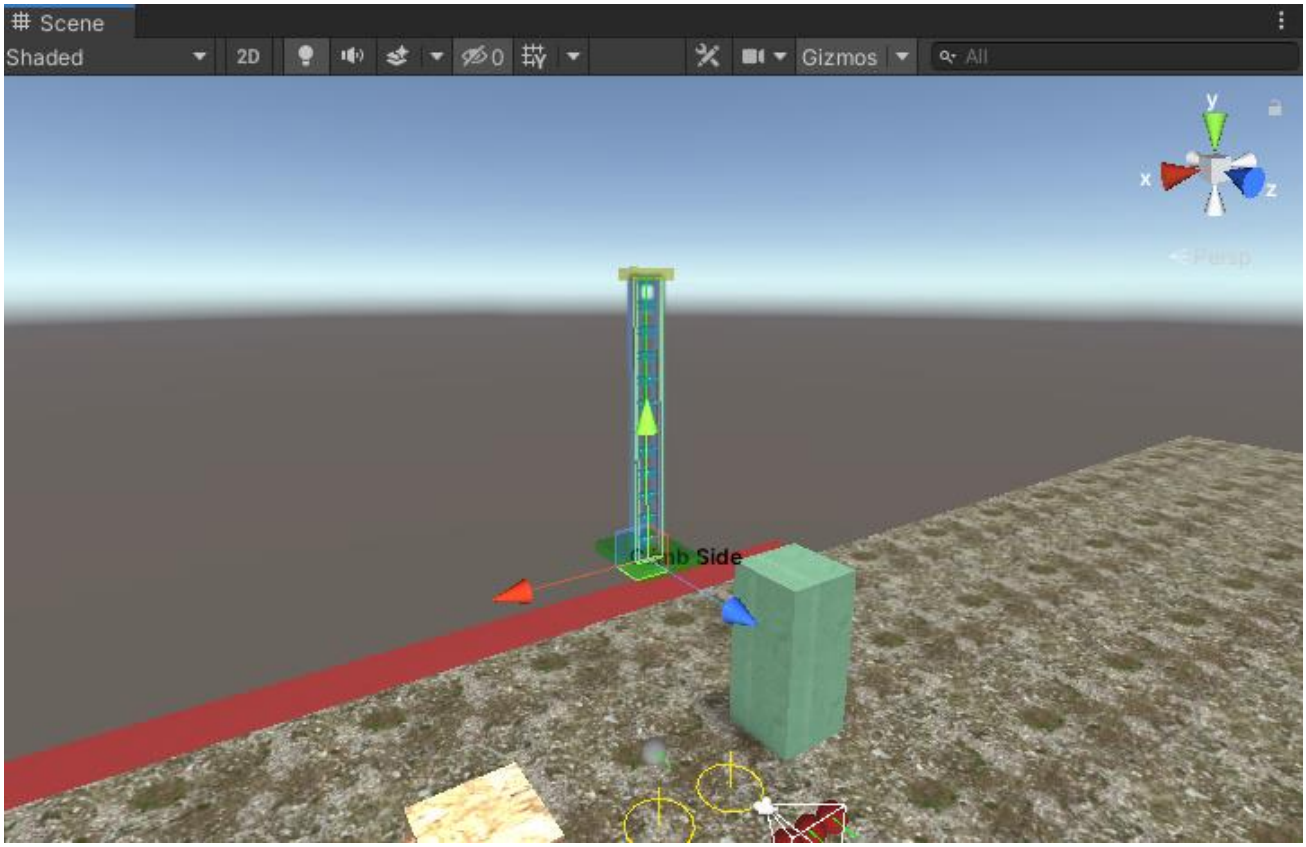


- Create '_AllPoints' object by clicking Warmerise -> Workflow -> Create '_AllPoints' Object
- Remove unnecessary object from '_AllPoints' (Armory, JetPoint, CarPoint and SpawnPoints)
- Start placing the objects around the map so it mimics the base layout, then select '_BallModeBaseMap' and disable its MeshRenderer component, so it becomes invisible.
- The Ball mode map is ready.

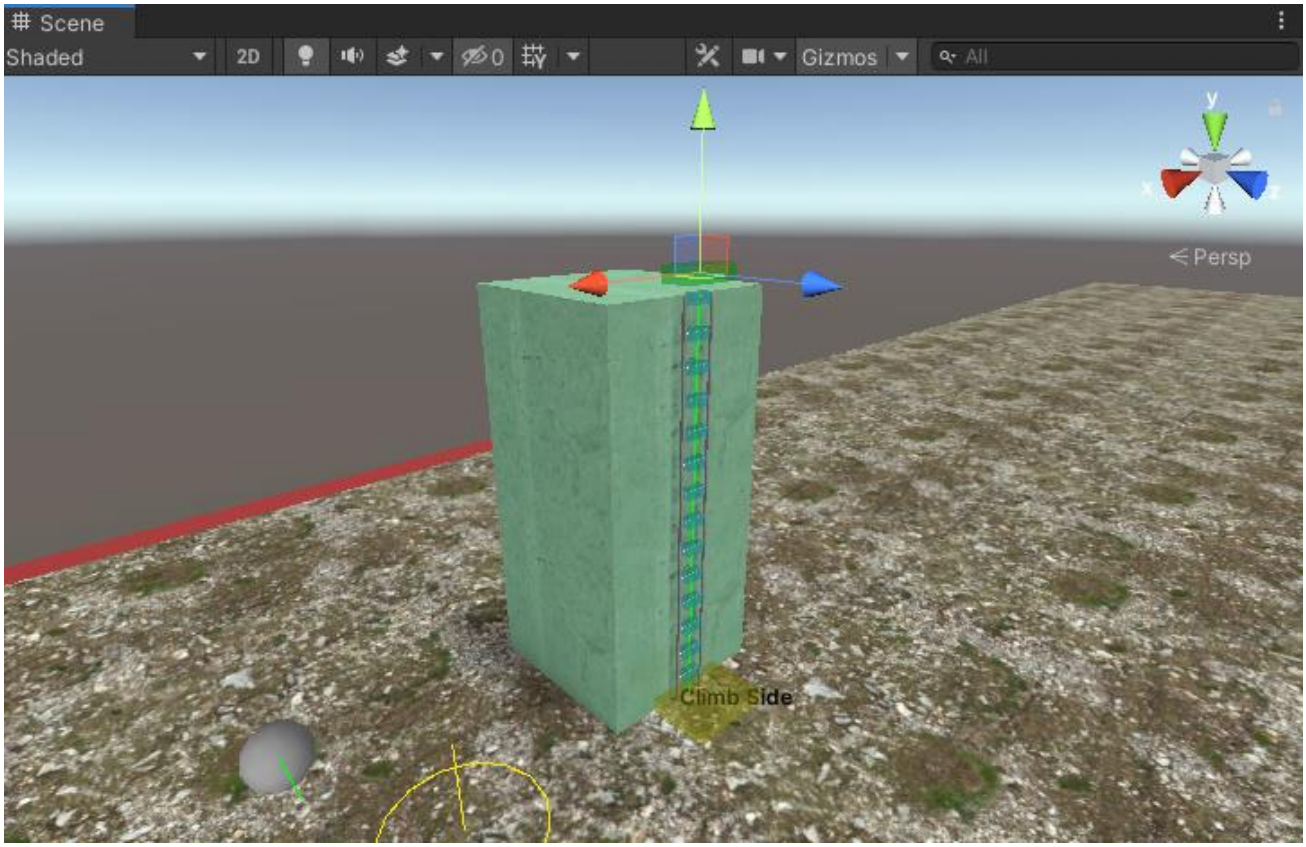
Adding Ladders

Ladders allow players to climb up and down, and are used when the map has multiple levels:

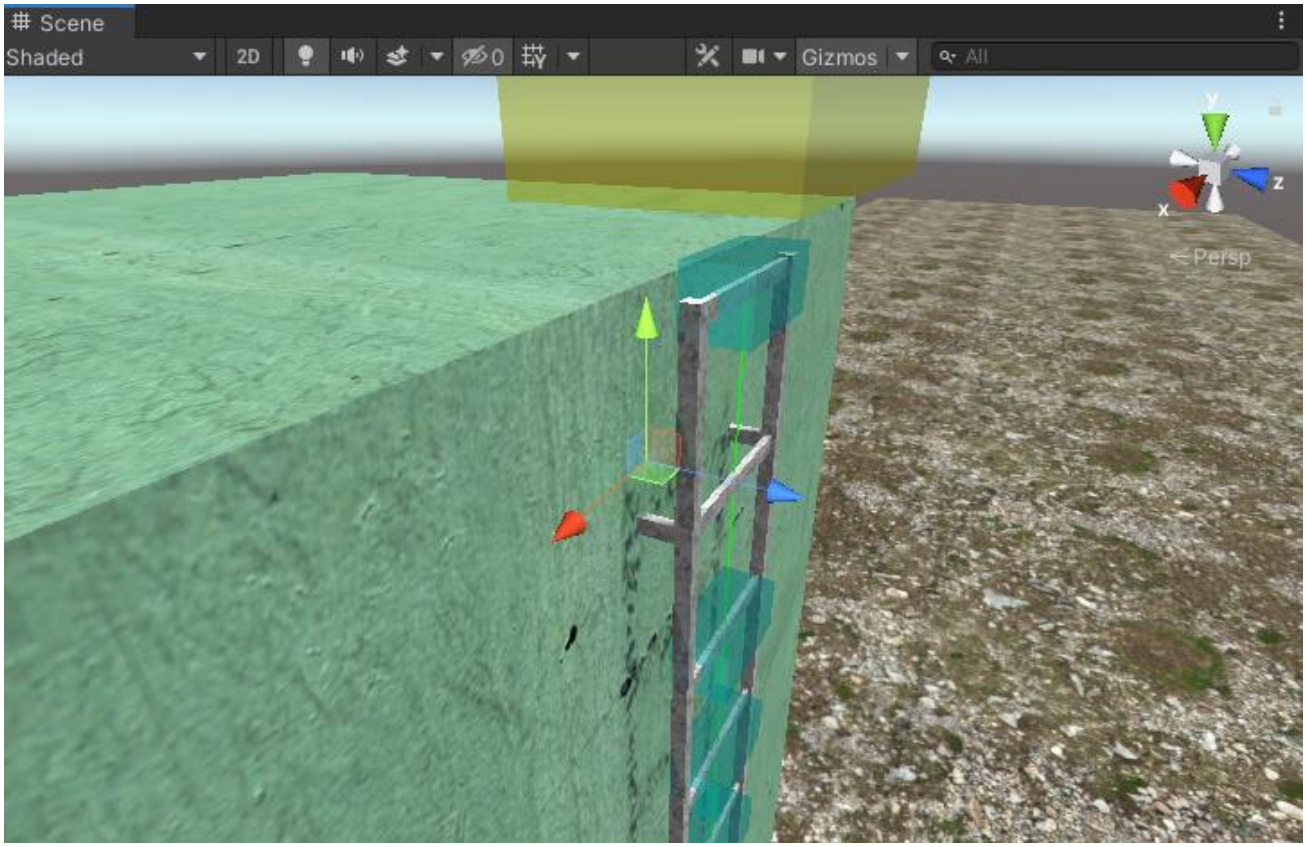
- Create new ladder by clicking Warmerise -> Workflow -> Create New Ladder



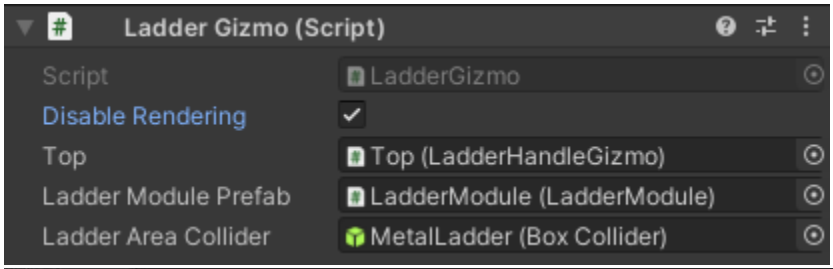
- Move the ladder to where you need it, then select the Top square and move it up or down, to adjust the height (make sure the ‘Climb Side’ is facing away from the wall):

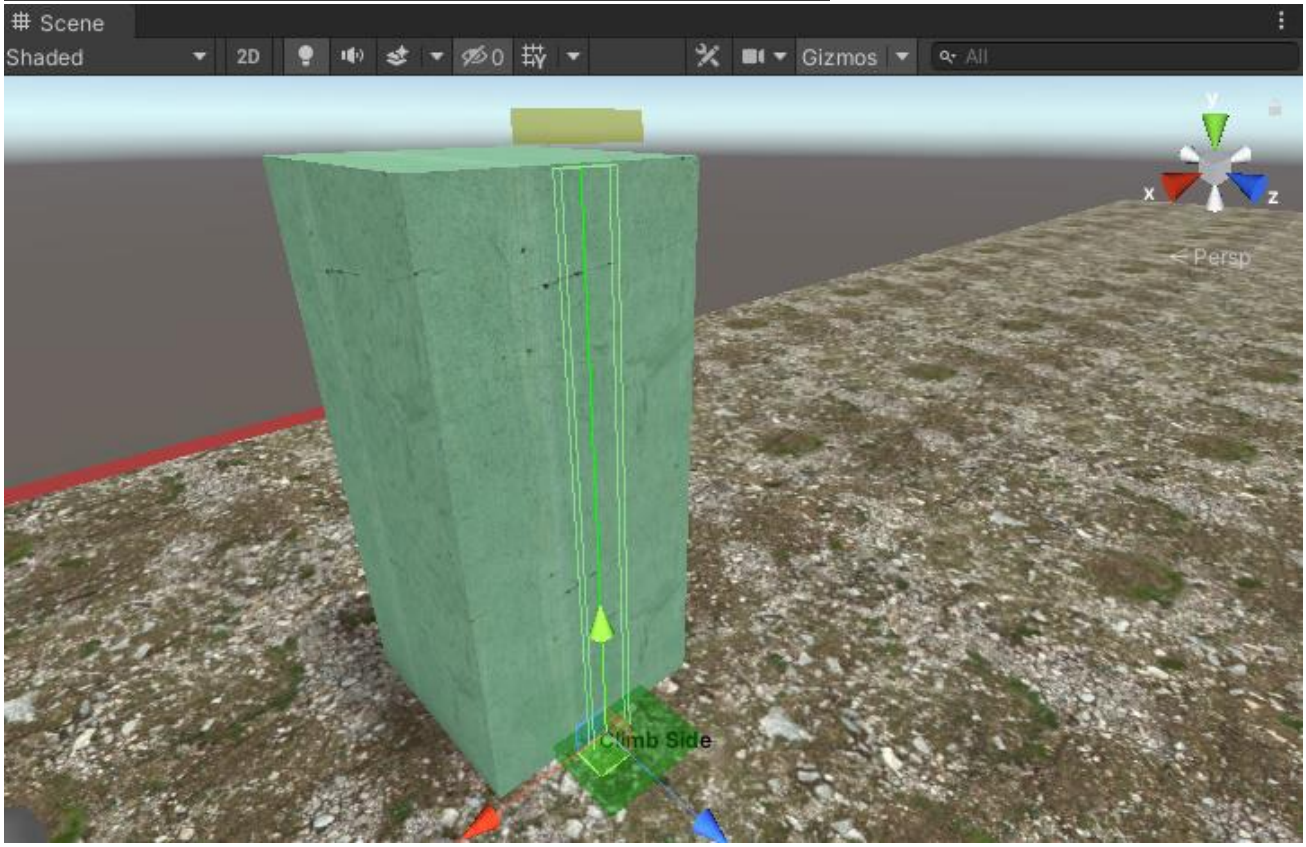


- You can tweak the support rails, by dragging the smaller squares out, towards the wall:



- If you have your own ladder model, then select the bottom square and in Inspector view select ‘Disable Rendering’ option:

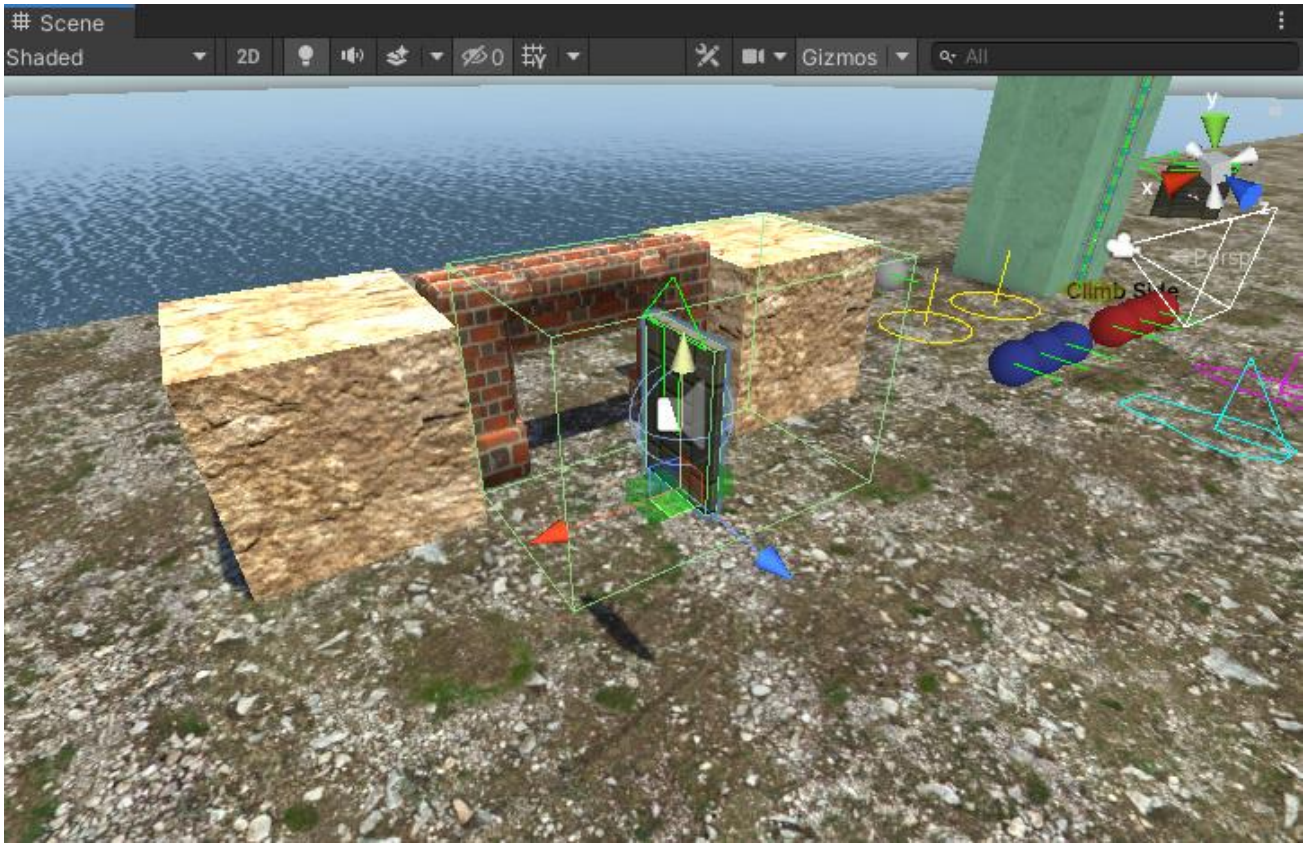




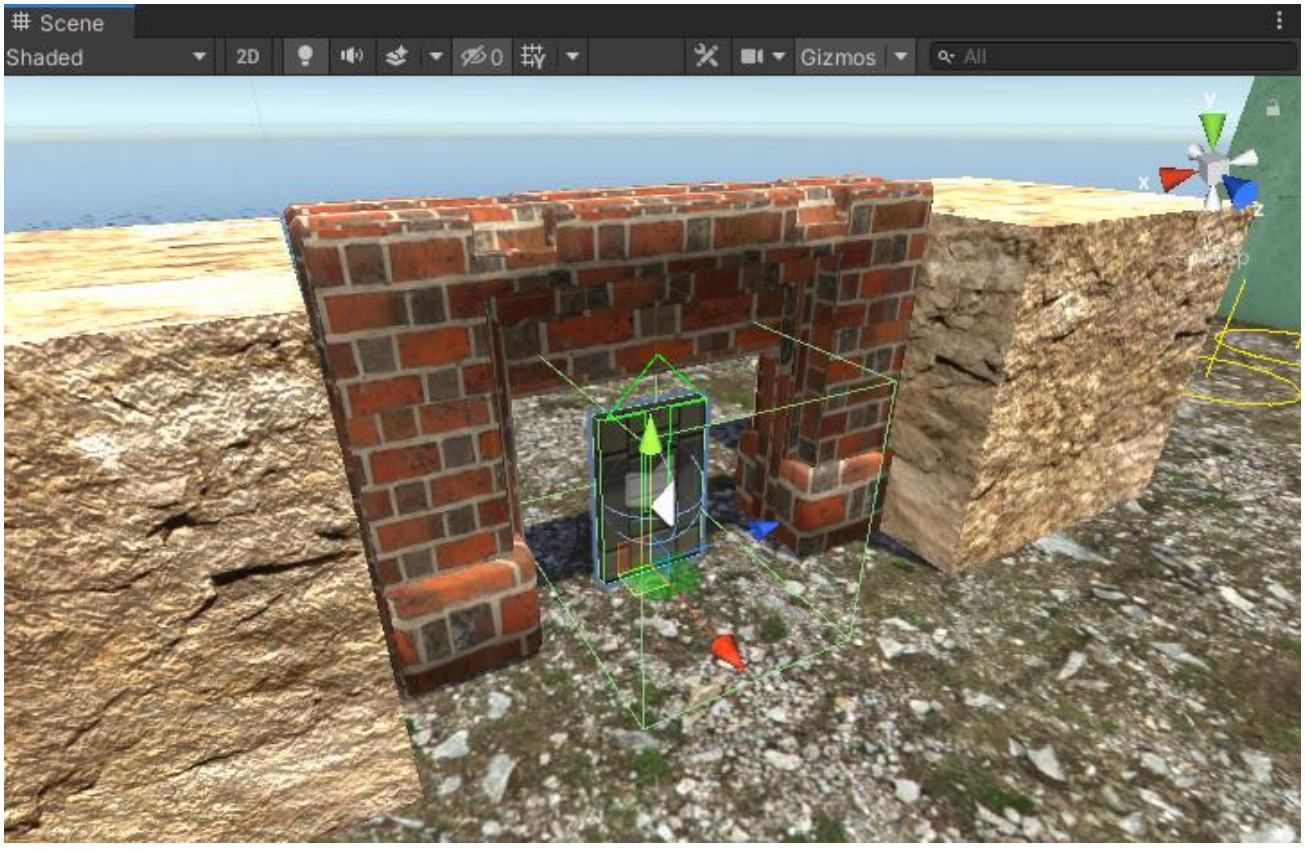
Adding Doors

Sliding doors are suitable for maps with rooms or indoor/outdoor spaces:

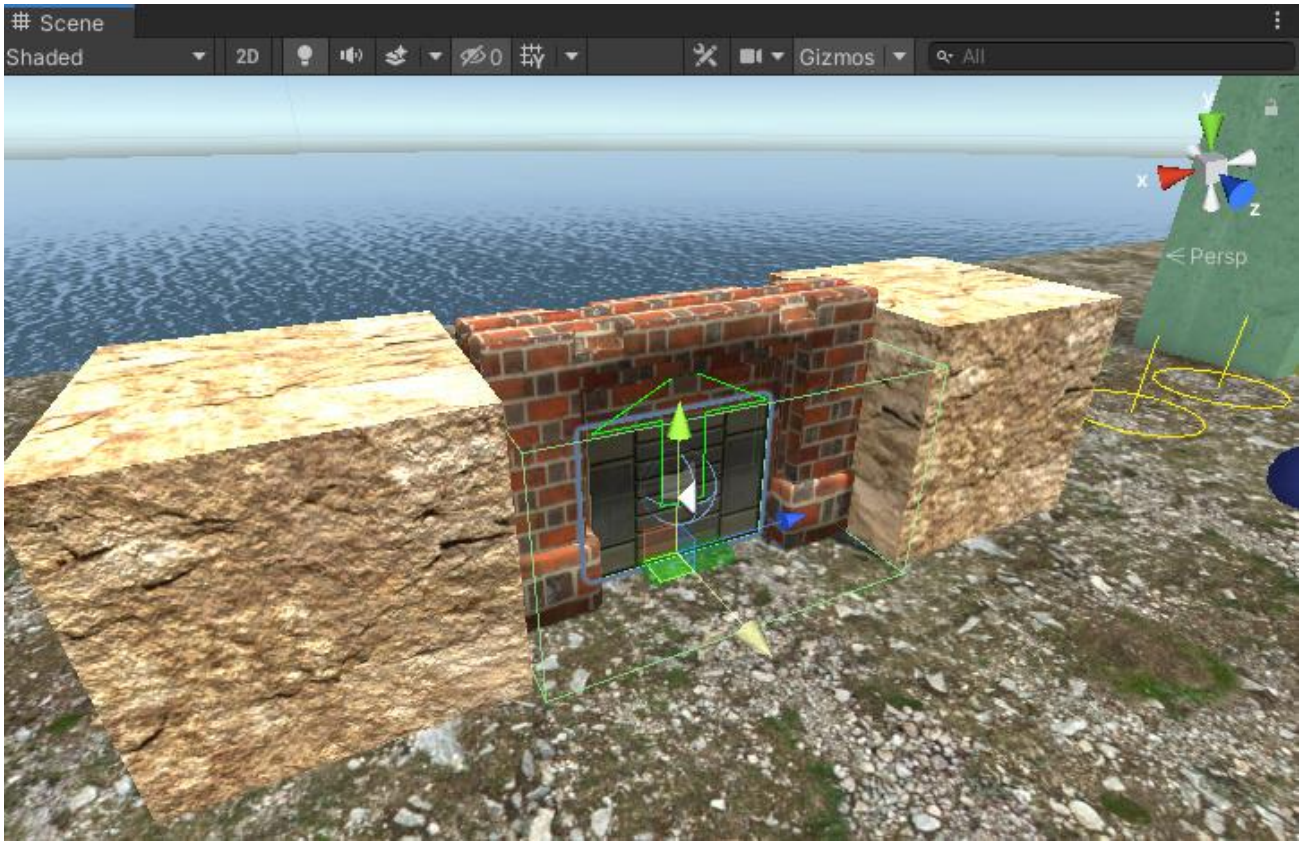
- Create new door by clicking Warmerise -> Workflow -> Create New Door



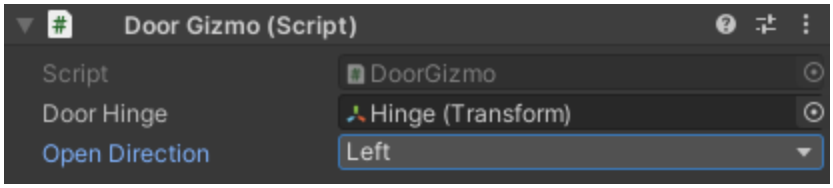
- Move the door in place



- Scale the door until it matches the passage:



- The small green box represents a handle gizmo (used for selecting the door in Editor)
- The big green outline represents the trigger Box Collider area, entering / leaving which, will open / close the door
- The green arrow represents the open direction of the door, to change it, select the door root object and change ‘Open Direction’ value in Door Gizmo:



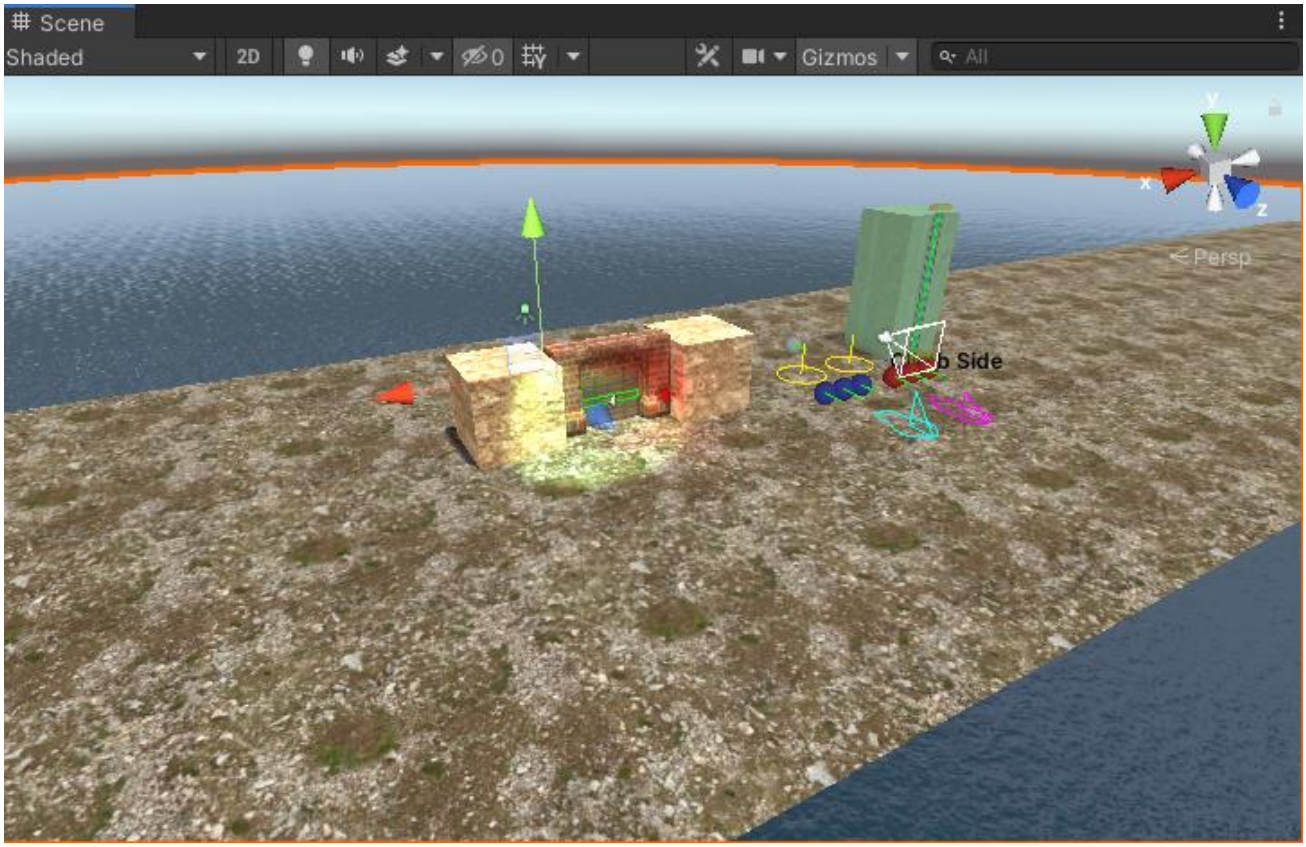
- To change door model, replace the object inside ‘Body’ object (Make sure the new door model has a collider component)
- To change door open/close sounds assign AudioClip to AudioSources attached to ‘Open’ and ‘Close’ objects.

Adding Water Plane

Optionally, you can add a reflective water, suitable for island-type maps, pounds, pools etc.

- To create water click Warmerise -> Workflow -> (Create Day Time Water or Create Night Time Water)

Day Time Water is suitable for maps with "daytime" lighting and Night Time Water is suitable for dark "night time" lighting.

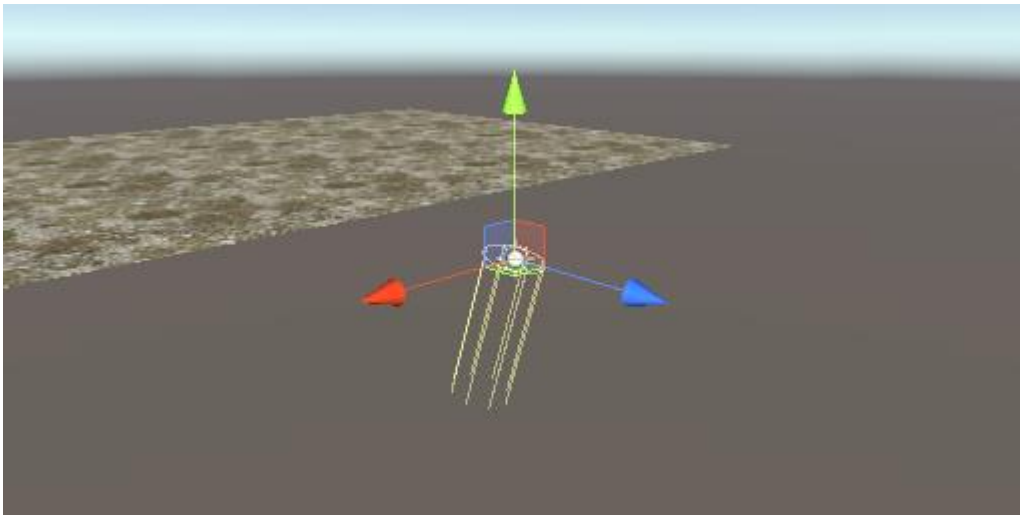


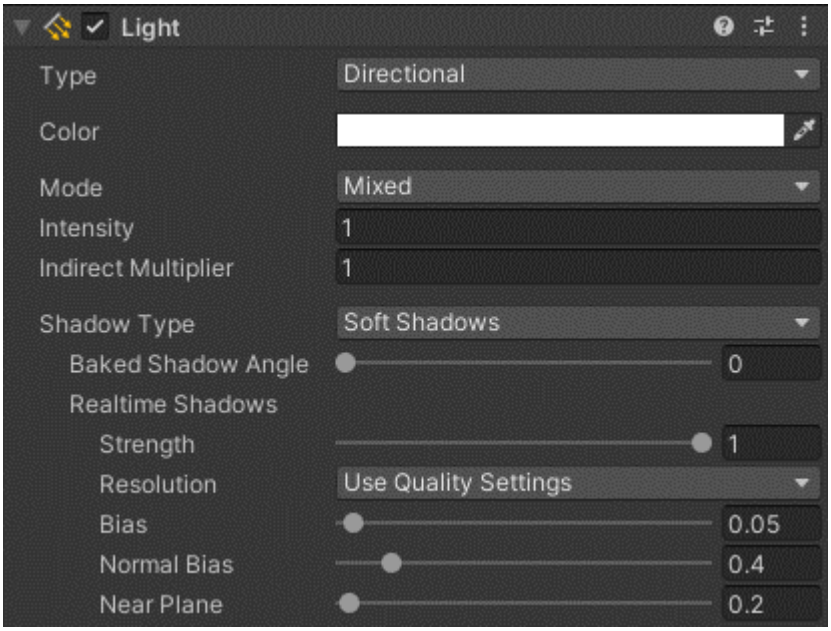
Map Lighting

Lighting plays an important role, when it comes to making an enjoyable map.

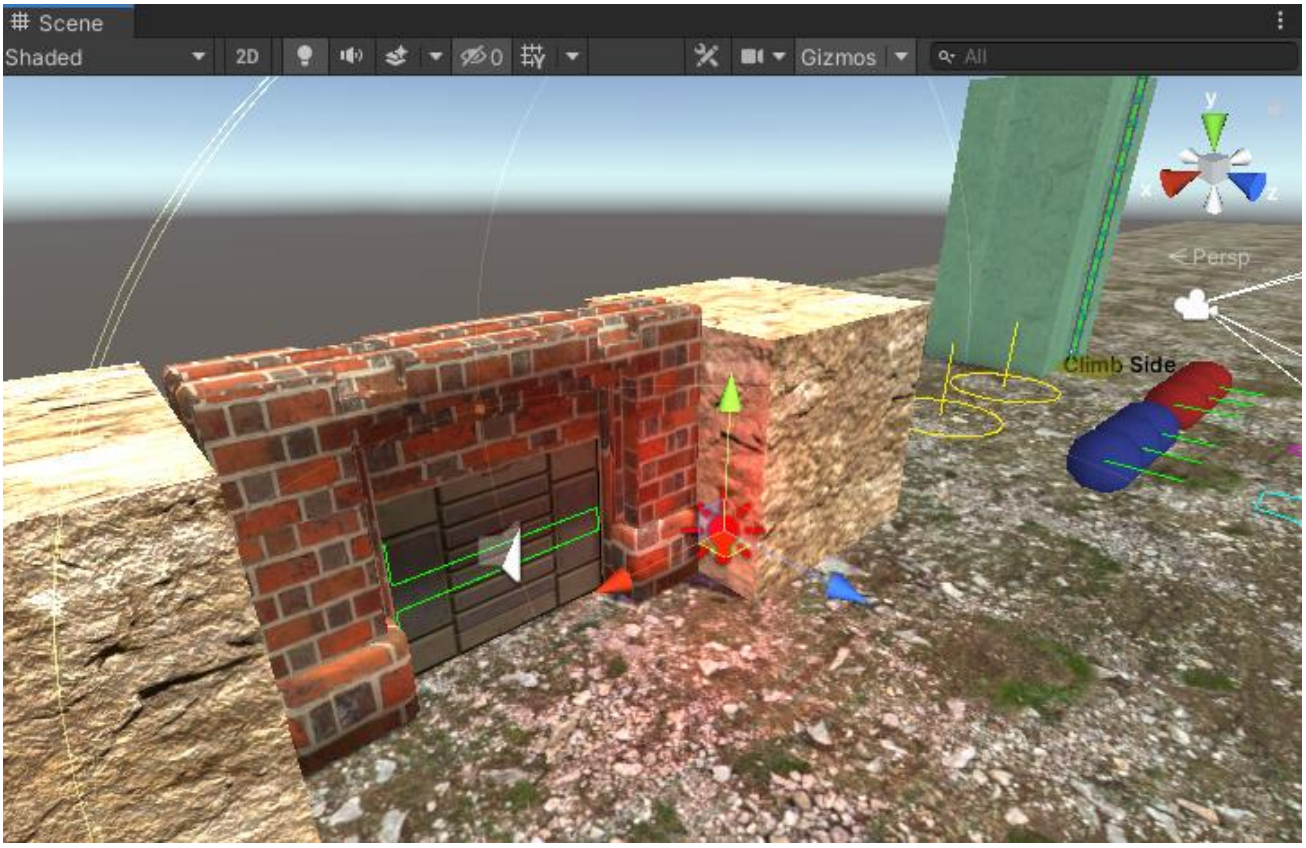
In Unity the lighting comes from multiple sources.

Directional Light – when you first create a Scene, there Directional Light gets added right way. You can select it and tweak the values such as Color, Intensity, Shadow Type etc. This type of light projects on every object in the Scene and should be the main light source.

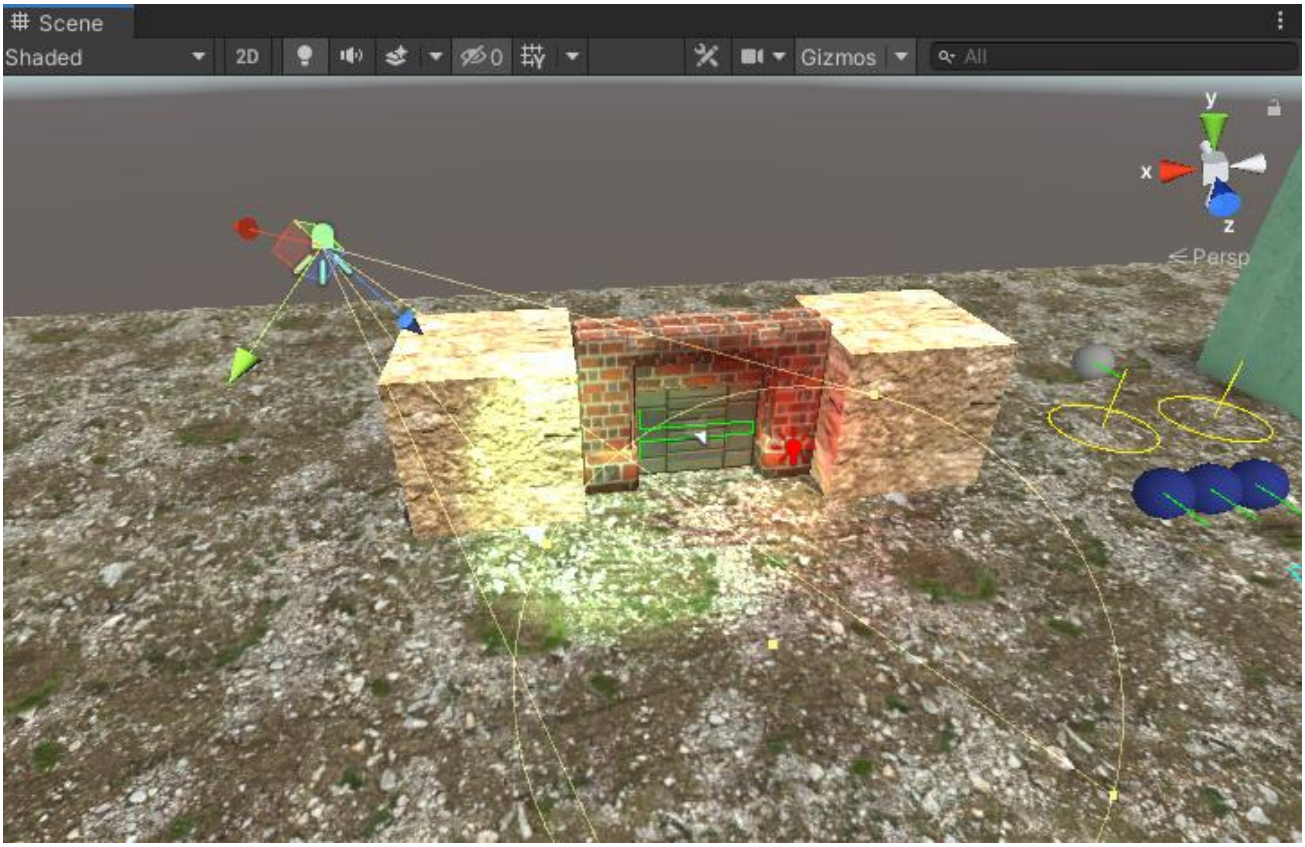




Point Light – (Game Object -> Light -> Point Light) this light illumines the area close-by (Tip: avoid having too many Point Lights, as it may negatively impact performance).

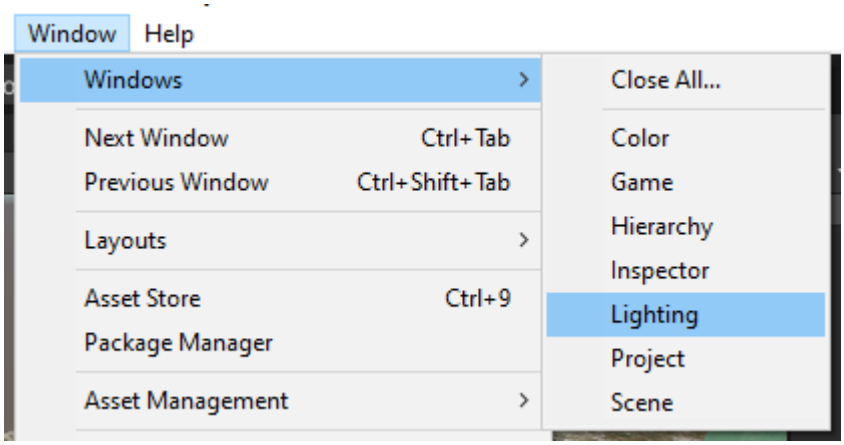


Spotlight – (Game Object -> Light -> Spotlight) just like a Point Light, it illuminates the near-by area, but only in one direction (same as Point Light, avoid having too many Spotlights)

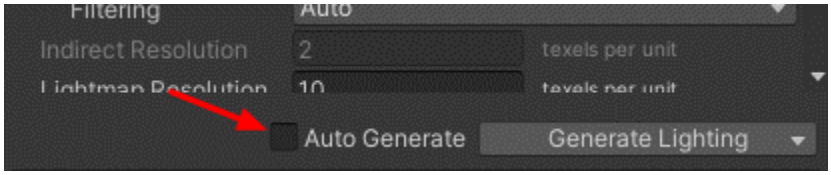


Environment Lighting – this type of lighting controls the global illumination of the Scene.

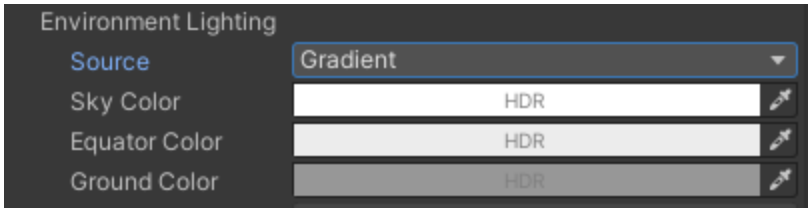
- To tweak the environment lighting, you need to open the lighting window by clicking Window -> Windows -> Lighting



TIP: To prevent inconsistent lighting between Unity and the game, disable Auto Generate lighting then reload the Scene.



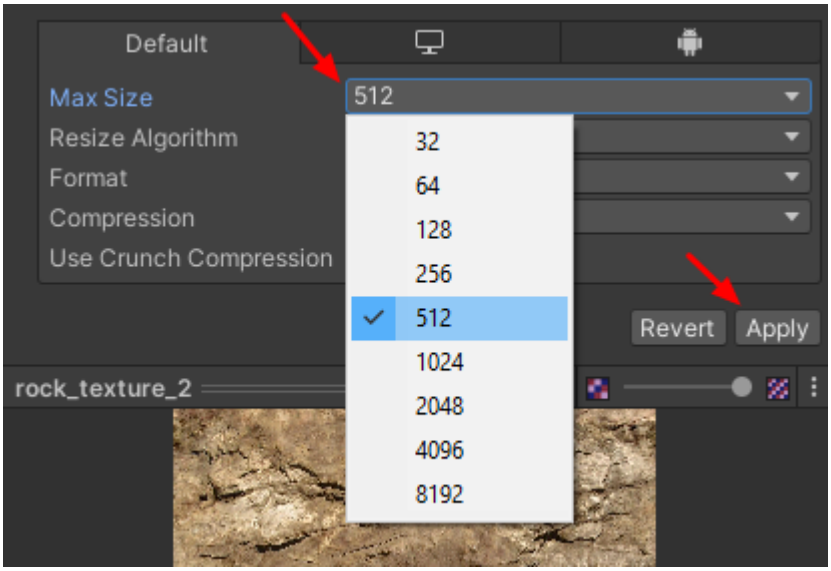
- Under ‘Scenes’ tab, the default Environment Lighting Source is set to Skybox, which is Ok, but sometimes can be a bit dark. To improve the lighting, change the Source to ‘Gradient’ and tweak the colors, until you happy with the result:



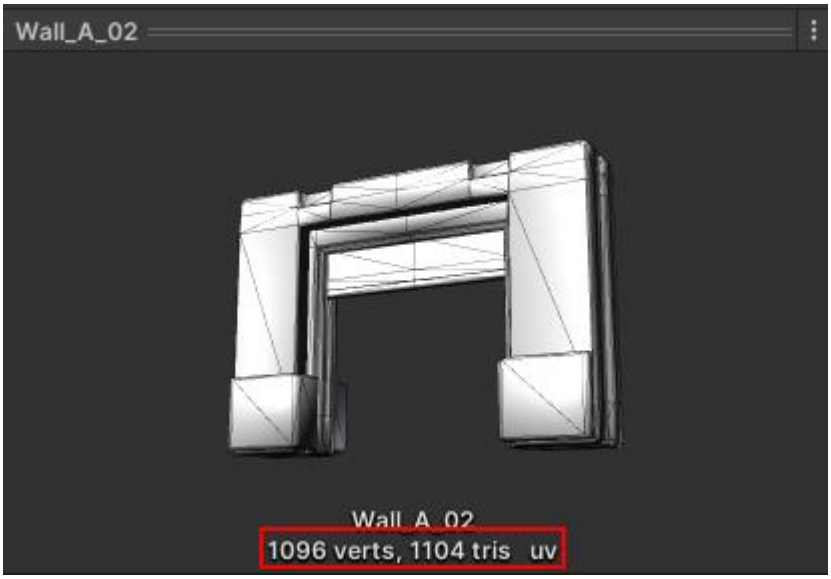
Optimization Tips

Before exporting the map, you need to ensure it's well optimized:

Tip 1: Reduce Texture size to 512 x 512 (or 1024 x 1024, but only for textures which you need to be more detailed)

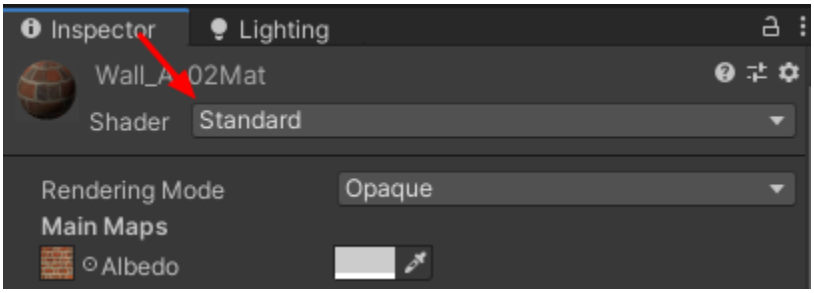


Tip 2: Only use "low-poly" models (Aim to use models under 10K verts/tris)



Supported Shaders

Shader is a component that is applied to material and change the way it looks.



Below is the list of supported shaders:

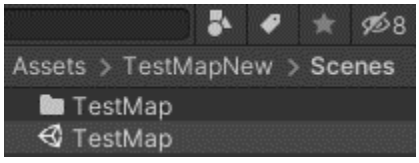
- Legacy Shaders/Diffuse
- Legacy Shaders/Specular
- Legacy Shaders/Bumped Specular
- Legacy Shaders/Bumped Diffuse
- Legacy Shaders/Parallax Specular
- Legacy Shaders/Transparent/Diffuse
- Legacy Shaders/Transparent/Bumped Specular
- Legacy Shaders/Transparent/Cutout/Diffuse
- Legacy Shaders/Particles/Additive
- Legacy Shaders/Particles/Additive (Soft)
- Legacy Shaders/Particles/Alpha Blended
- Legacy Shaders/Particles/Multiply
- Skybox/6 Sided
- Skybox/Procedural

Any unsupported shaders will be converted to ‘Legacy Shaders/Diffuse’ in the game.

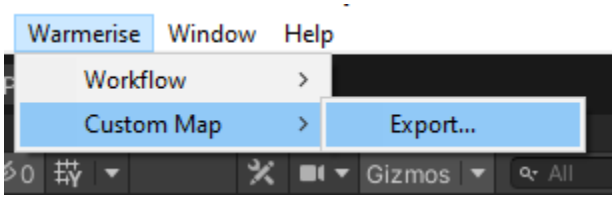
Exporting The Map

After the map is setup, it’s ready to be exported:

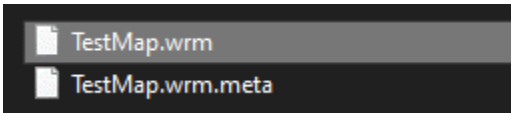
- Select Scene in the Project view
- Rename the Scene to whatever name you want your map to have (NOTE: Once the map is uploaded and published, its name cannot be changed)



- Click Warmerise -> Custom Map -> Export… (Exporting should take a few seconds)



After the exporting is done, the new folder will be created in Assets folder called '__Maps' where the exported file with extension .wrm will be saved, this is the file that can be used in the game.



Testing The Map

Now it's time to test the map, to verify that it works fine:

- Go to the game page (Make sure you are logged in)
- Click PROFILE -> CUSTOM MAPS -> Test New Map…

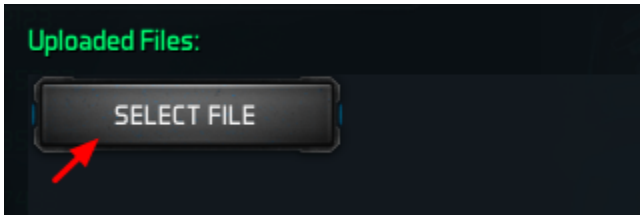
- Select the exported .wrm file.
- Walk around the map to make sure everything works correctly:

Uploading The Map

When the map was tested you can now upload it:

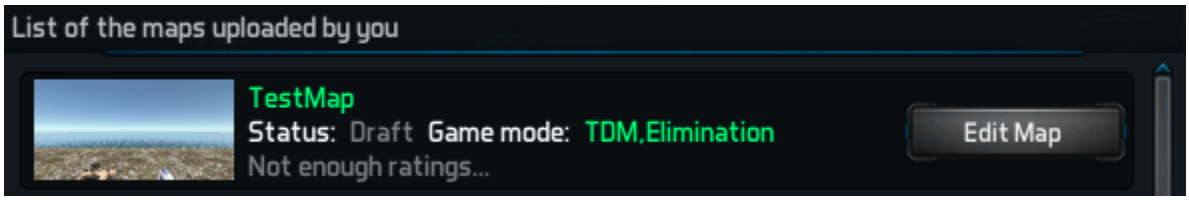
- Click 'Add New Map'

- Click ‘SELECT FILE’ and select the exported .wrm file



- After selecting the file, the game will quickly load the map and generate the preview image:
- Enter map description, click ‘SUBMIT’ then wait for it to upload.

- Once uploading is complete, the map will appear in the map list



Publishing The Map

After the map was uploaded it’s ready to be published:

- Click ‘Edit Map’

- Near the map file click ‘Options’

Delete is self-explanatory, clicking it will delete the map file

Publish will publish this map file to a pending section, where it can be reviewed and voted by the community.

Changes will open the Changelog window where you can type what’s new in this version.

If the map was just created, the appropriate changelog would be “- Initial release”, otherwise, you should include the list of the changes, for example:

- Removed object A
- Added new corridor

  Etc.

- To publish the map, click Publish
- To unpublish the map, click Unpublish

Published maps will appear in ‘Published Maps’ (once approved) and ‘Pending Maps’ (if under review) lists, when creating a server.

Updating The Map

- To update map description, go to your maps, click ‘Edit Map’, make changes to description then click ‘SUBMIT’
- To upload updated version of your map, click ‘Edit Map’ then click ‘Select File’ then click ‘SUBMIT’

- After the file is uploaded, click ‘Options’ then click ‘Changes’, and type what’s new in this version
- Lastly, click ‘Options’ then ‘Publish’ (NOTE: If there is already a published file waiting for approval, you need to unpublish it first).