

Topic	A-Frame NAVIGATION MESHES				
Class Description	Students will learn how to create navigation meshes models. Students will learn to use the movement control component to walk around the A-Frame virtual environment scene without passing through objects.				
Class	C164				
Class time	45 mins				
Goal	<ul style="list-style-type: none"> ● Learn to create navigation meshes. ● Learn about the A-Frame movement component. 				
Resources Required	<ul style="list-style-type: none"> ● Teacher Resources <ul style="list-style-type: none"> ○ Visual Studio Code Editor ○ laptop with internet connectivity ○ earphones with mic ○ notebook and pen ● Student Resources <ul style="list-style-type: none"> ○ Visual Studio Code Editor ○ laptop with internet connectivity ○ earphones with mic ○ notebook and pen 				
Class structure	Warm-Up Teacher-led Activity Student-led Activity Wrap-Up	5 mins 15 mins 20 mins 5 mins			
WARM-UP SESSION - 5 mins					
<u>CONTEXT</u>					
<ul style="list-style-type: none"> ● Create navigation meshes. ● Use an A-Frame movement component. 					

 <p>Teacher Starts Slideshow Slide 1 to 3</p> <p>Refer to speaker notes and follow the instructions on each slide.</p>		
Hey <student's name>. How are you? It's great to see you! Are you excited to learn something new today?	ESR: Hi, thanks! Yes I am excited about it!	Click on the slide show tab and present the slides
WARM-UP QUIZ Click on In-Class Quiz		
 <p>Continue WARM-UP Session Slide 4 to 16</p>		
Following are the session deliverables: <ul style="list-style-type: none"> • Greet the student. • Revision of previous class activities. • Quizzes. 		
Class Steps	Teacher Action	Student Action

	<p>Using an A-Frame environment, we created our game scene very easily.</p> <p>We added a few more elements in the scene but there is some issue that we discussed, do you remember that?</p> <p>Can you tell what were issues in the game scene?</p> <p>Yes we need to fix this.</p> <p>Remember how we used to fix these things while we were using p5.js and p5.play.js library to make games?</p> <p>Yes. In p5.play.js library, we had a collider radius for every sprite object and we had written collide or touching functions to detect collision between objects.</p> <p>In A-Frame virtual environment scenes we have bounding shapes that decide the collision area for the 3D objects in the scene.</p>	<p>ESR: Yes.</p> <p>ESR: We were able to cross through objects in the scene. Also every time we were loading the game, the objects were at random positions.</p> <p>ESR: Yes. We were using collider radius and touching functions to check whether the sprite objects are colliding with each other or not.</p>
--	--	---

	<p>And for every object in the game scene we can write the collision detection algorithm whether the object is colliding with the shooter's gun weapon such that we don't cross through those objects.</p> <p>Well, that is going to be a very long way to make objects as obstacles and not let the weapon pass through them.</p> <p>In an A-Fame virtual environment we can create user navigation meshes to create walkable and non-walkable areas for the user.</p> <p>In today's class we will be learning how to create a navigation mesh model that will help us to walk around the scene without passing through the objects.</p> <p>Are you excited? Let's quickly solve the quiz based on the previous class first and then we can begin with the class.</p>	
	Let's get started then.	
▶ <div style="text-align: center;"> ▶ Teacher Ends Slideshow  </div>		
TEACHER-LED ACTIVITY - 15 mins		
Teacher Initiates Screen Share		

CHALLENGE

- Create the navigation meshes in A-Frame.
- Learn about the A-Frame movement component.

**Step 2:
Teacher-led
Activity
(15 mins)**

<The teacher clones the code from the Teacher Activity 1 and runs the code.>

[Teacher Activity 1]

Here I have a very basic scene with few boxes lying around forming a very basic maze like structure.

Let's see the arrangement of boxes in the A-Frame Visual Inspector tool.

We can use **<Ctrl+Alt+i>** keys together to open the Visual Inspector tool.

```

<a-scene id="scene">
  <!--Camera and Cursor-->
  <a-entity id="camera" camera position="0 1.6 0" wasd-controls="acceleration:200"
    look-controls="pointerLockEnabled: false">
    <a-cursor></a-cursor>
  </a-entity>

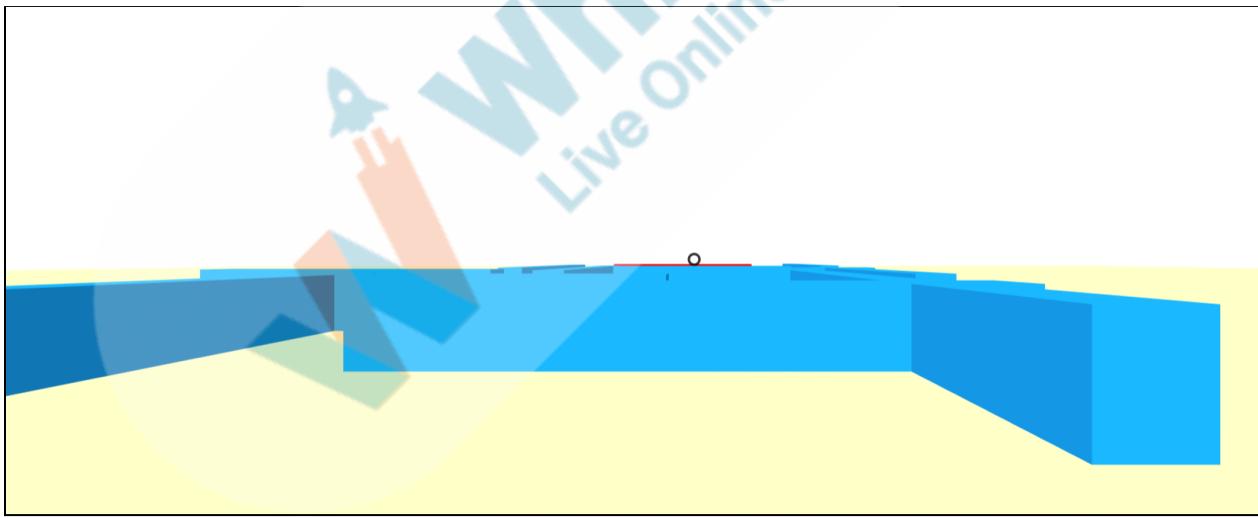
  <!--Bullets-->
  <a-entity bullets></a-entity>

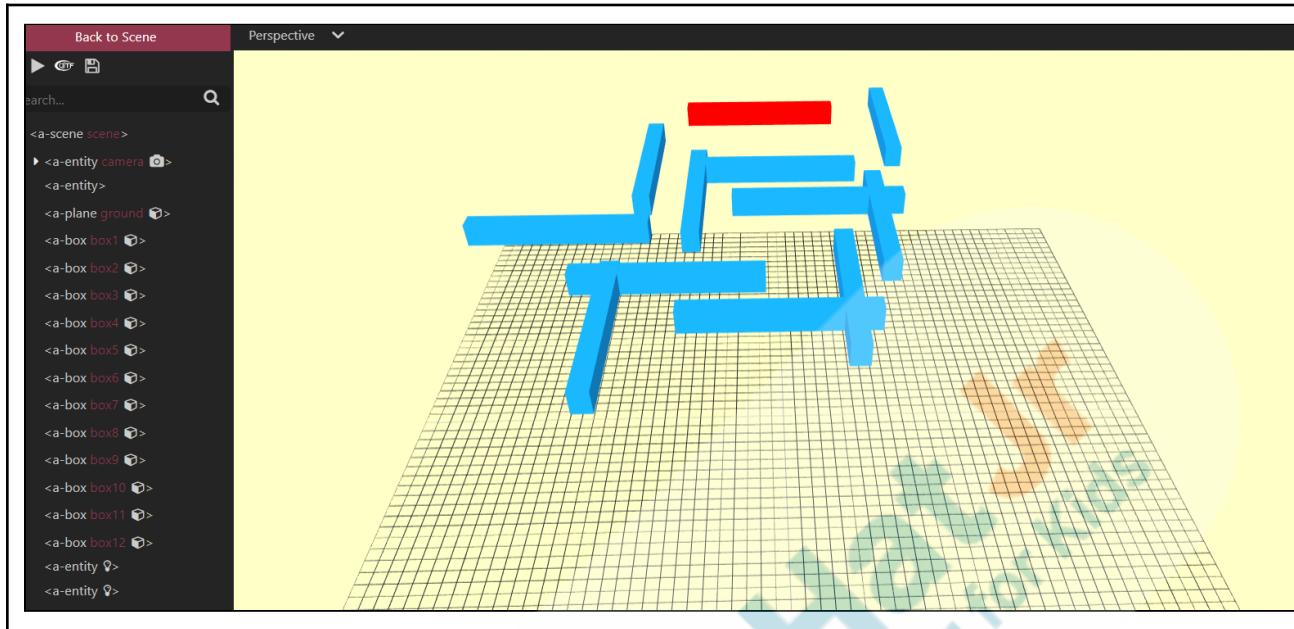
  <!--Ground-->
  <a-plane id="ground" position="0 0 0" rotation="-90 0 0" height="200" width="200"
    color="#FBF2B0" static-body visible="true">
  </a-plane>

  <!--Boxes-->
  <a-box id="box1" position="0 0.5 -6.33437" color="#16A2F6" depth="1" height="1.5" width="10"></a-box>
  <a-box id="box2" position="-5.63047 0.5 -9.65585" color="#16A2F6" depth="1" height="1.5" width="10"></a-box>
  <a-box id="box3" position="2.5 0.5 -17.4559" color="#16A2F6" depth="1" height="1.5" width="10"></a-box>
  <a-box id="box4" position="0 0.5 -21.58323" color="#16A2F6" depth="1" height="1.5" width="10"></a-box>
  <a-box id="box5" position="-12.01326 0.5 -14.59535" color="#16A2F6" depth="1" height="1.5" width="10"></a-box>
  <a-box id="box6" position="-7.73232 0.5 -22.17969" color="#16A2F6" depth="10" height="1.5" width="1"></a-box>
  <a-box id="box7" position="-8.54977 0.5 -5.3525" color="#16A2F6" depth="10" height="1.5" width="1"></a-box>
  <a-box id="box8" position="-4.67013 0.5 -18.0932" color="#16A2F6" depth="10" height="1.5" width="1"></a-box>
  <a-box id="box9" position="5.90667 0.5 -15" color="#16A2F6" depth="10" height="1.5" width="1"></a-box>
  <a-box id="box10" position="3.59768 0.5 -8.18822" color="#16A2F6" depth="10" height="1.5" width="1"></a-box>
  <a-box id="box11" position="7.63783 0.5 -28.06952" color="#16A2F6" depth="10" height="1.5" width="1"></a-box>
  <a-box id="box12" position="-0.82929 0.5 -30.18869" color="red" depth="1" height="1.5" width="10"></a-box>

</a-scene>

```





Now suppose we have to find a way to the red box, for this we need to move only in the plane area where there is no box.

But now if we move around the scene we can directly move to the red box. We do not have to worry about the boxes lying around our way to the red box.

In real life when you walk, you do not keep on crossing everything, right?

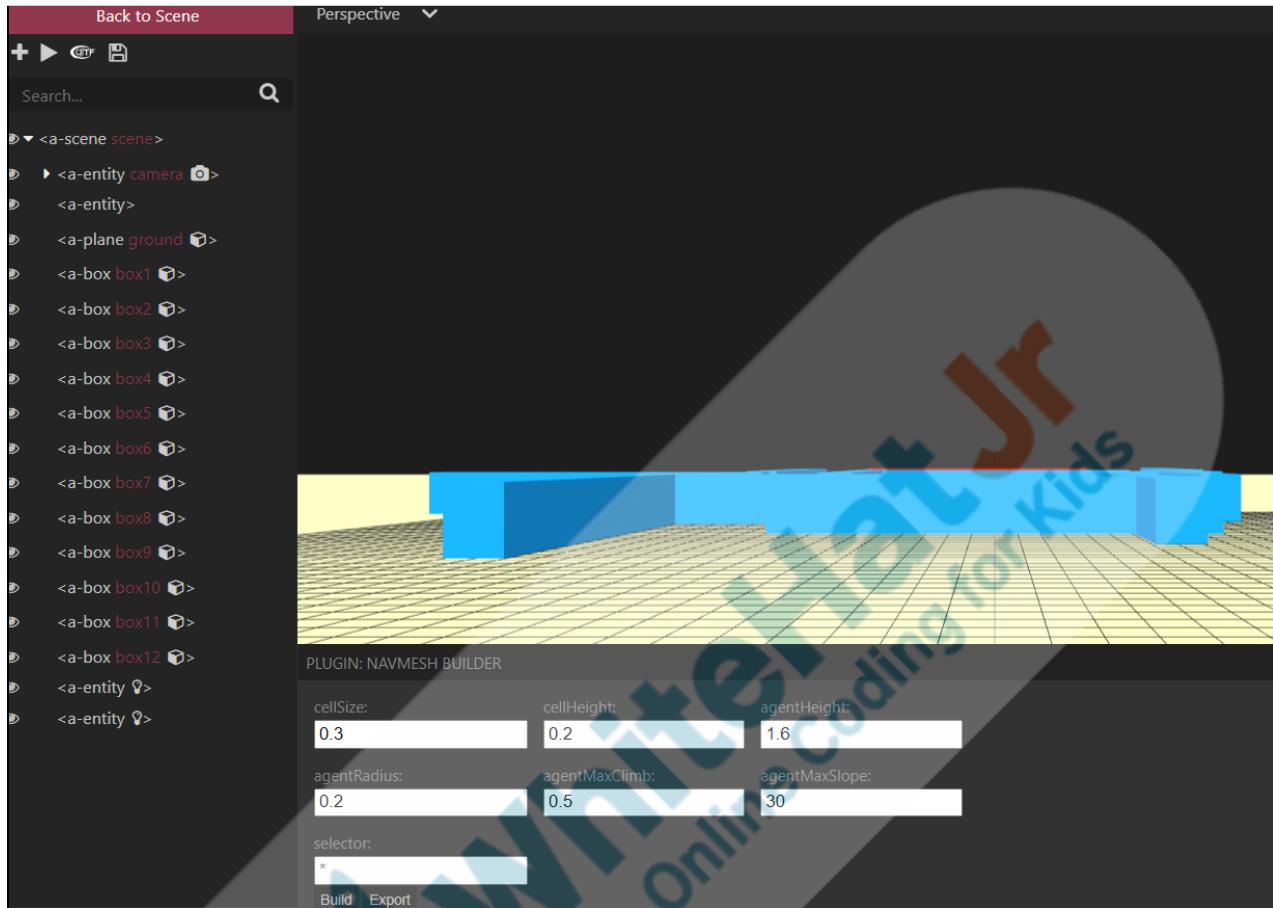
Everything that comes your way acts as obstacles.

To be able to reach the red box without passing through the blue boxes we should divide the scene

ESR: Yes.

	<p>into walkable and non-walkable (obstacles) areas.</p> <p>To do this, we will use A-Frame navigation meshes.</p> <p>Before that, can you tell me what you understand by the term navigation?</p> <p>Navigation is basically determining the position to follow a particular route.</p> <p>And do you remember how 3D structures are made?</p> <p>Amazing!</p> <p>Yes, and these 3D structures are also called meshes.</p> <p>Now, we will be learning to create navigation mesh or in short the navmesh of our scene.</p> <p>A navigation mesh is a polygon mesh that defines which areas of an environment are traversable.</p> <p>In other words, a character in a game could freely walk around within these areas of the environment.</p>	<p>ESR: Varied.</p> <p>ESR: Yes. 3D structures are 2D polygon shapes combined together.</p>
--	--	---

	<p>To create navmesh in A-Frame we will use the “aframe-inspector-plugin-recast” library.</p> <p>Link:</p> <p>https://recast-api.donmccurdy.com/aframe-inspector-plugin-recast.js</p> <p><i><The teacher adds the library link in the <head> tag.></i></p> <p>And we will use the “inspector-plugin-recast” component of the library.</p> <p>This will add the navmesh builder plugin window at the bottom of the A-Frame Visual Inspector tool.</p> <p><i><The teacher attaches the component to the scene element.></i></p> <p><i><The teacher opens the Visual Inspector and shows the navmesh builder plugin added.></i></p>	
	<pre><script src="https://recast-api.donmccurdy.com/aframe-inspector-plugin-recast.js"></script> <a-scene id="scene" inspector-plugin-recast> <!--Camera and Cursor--> <a-entity id="camera" camera position="0 1.6 0" wasd-controls="acceleration:200" look-controls="pointerLockEnabled: false"> <a-cursor></a-cursor> </a-entity></pre>	



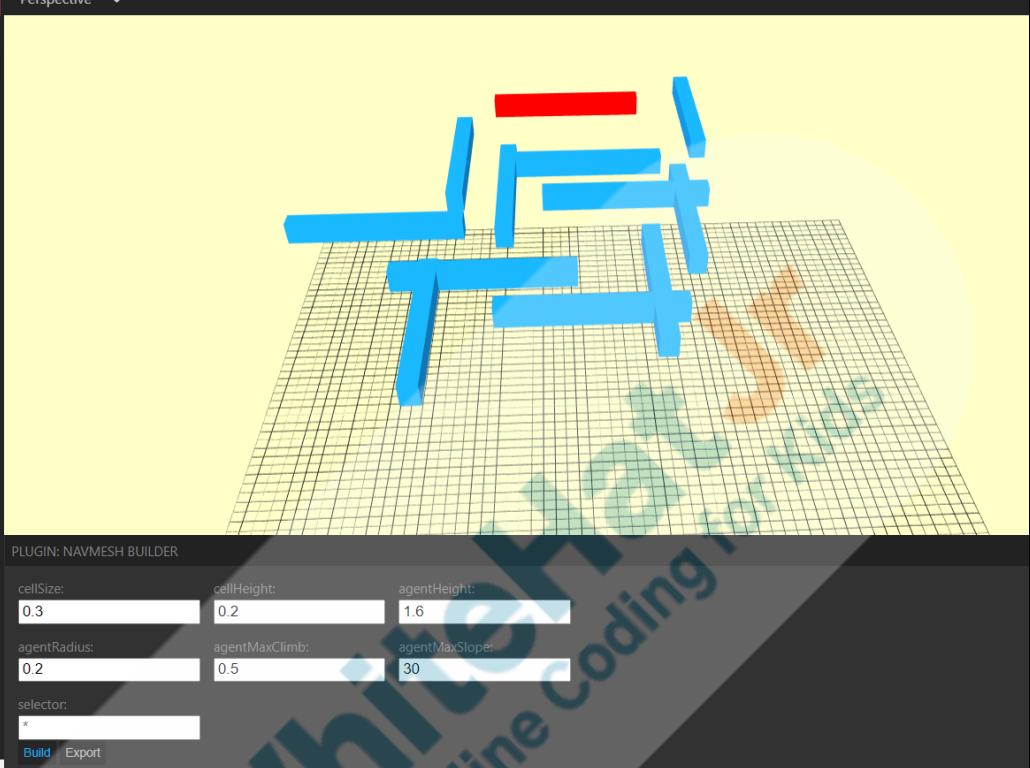
The screenshot shows the A-Frame developer console interface. On the left, there's a sidebar with a search bar and a list of entities in the scene, including a camera, ground plane, and twelve boxes labeled box1 through box12. On the right, the main view shows a 3D scene with a grid floor and several blue and yellow rectangular boxes. A large gray sphere is positioned above the boxes. At the bottom, a plugin interface for 'NAVMECH BUILDER' is displayed with the following settings:

cellSize:	0.3	cellHeight:	0.2	agentHeight:	1.6
agentRadius:	0.2	agentMaxClimb:	0.5	agentMaxSlope:	30
selector:	*				
		Build	Export		

Now once the navmesh builder is added, all we have to do is click on the “Build” button to create the navmesh.

<The teacher clicks on the Build button.>

Perspective



PLUGINS: NAVMESH BUILDER

cellSize:	0.3	cellHeight:	0.2	agentHeight:	1.6
agentRadius:	0.2	agentMaxClimb:	0.5	agentMaxSlope:	30
selector:	x				

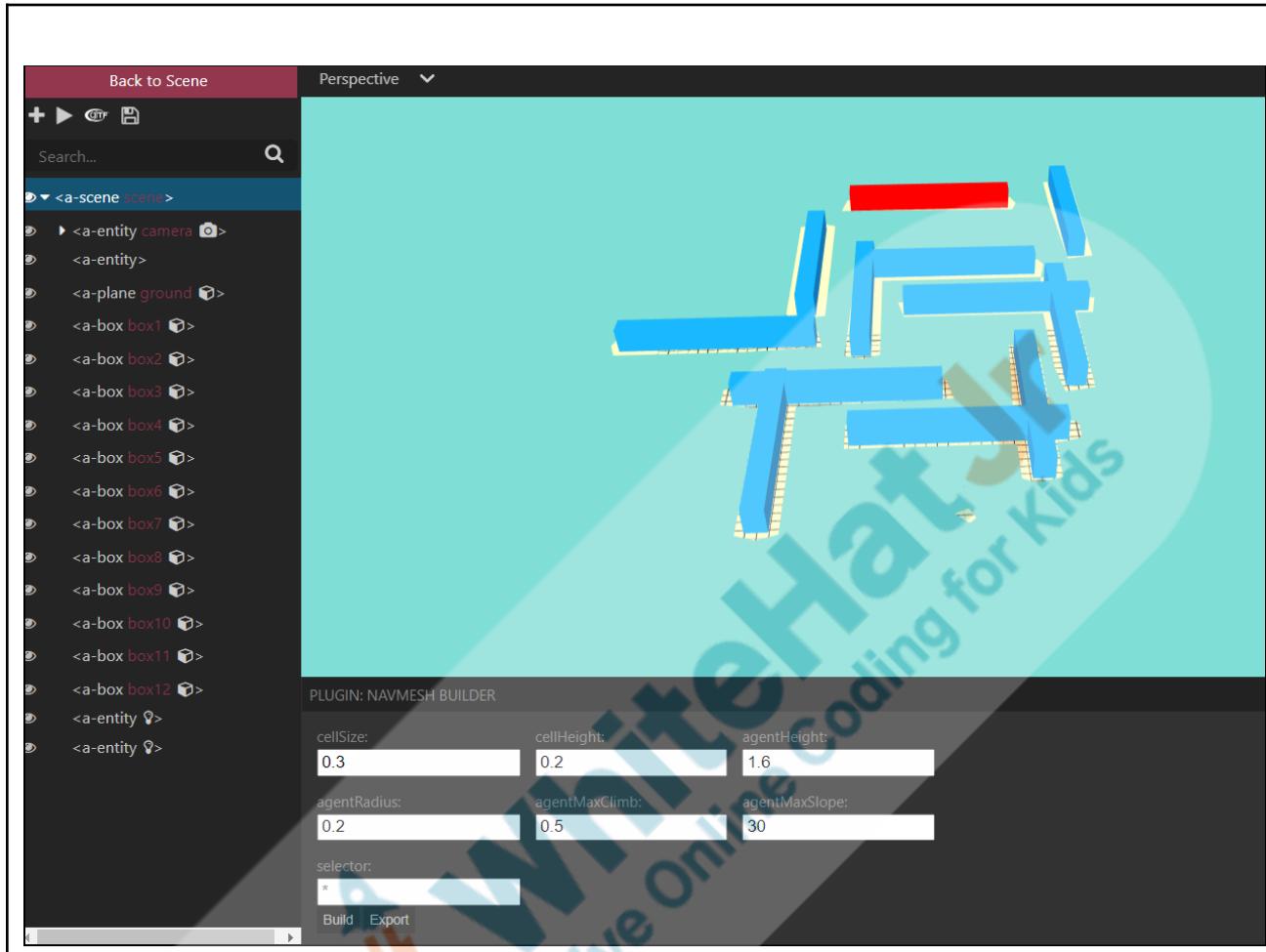
Build Export

Once you click on Build, the navmesh is created.

The blue area is the one which is **walkable**.

The **non-walkable area** is converted into holes where the objects of the scenes are placed.

You can notice that the area around the boxes is not blue.



We do not need to modify the default properties in the navmesh builder but we can just change one property, cellSize, to increase the precision of the holes cut around the boxes.

Also did you notice there is an extra hole in front of all the shapes even though there is nothing on the plane there?

Can you tell me why there is hole cut at the place?

ESR: Yes.

ESR: Varied.



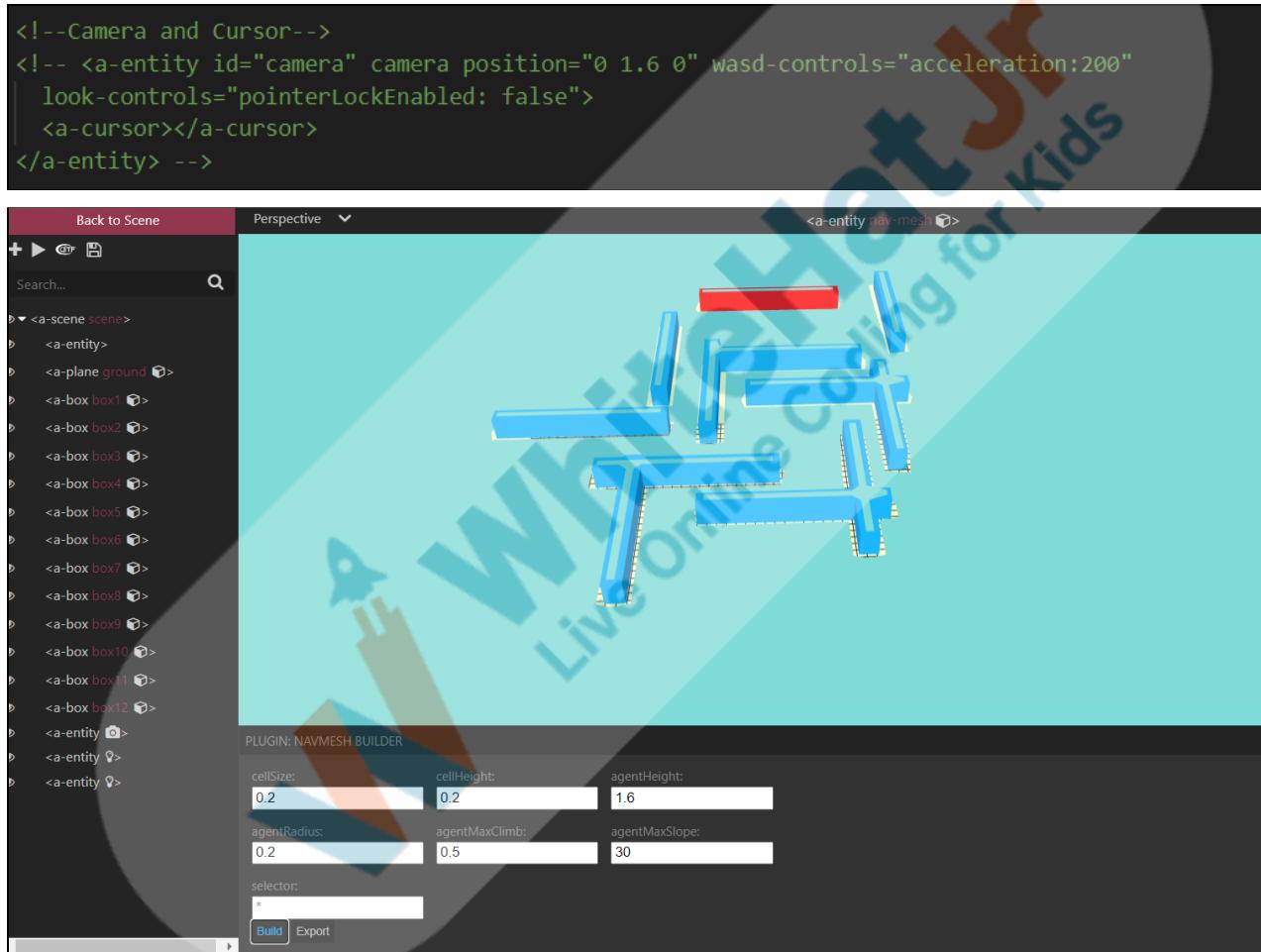
Well, this is because every entity which is a part of the scene is considered as an object, and the camera is placed which has also become a part of the navmesh.

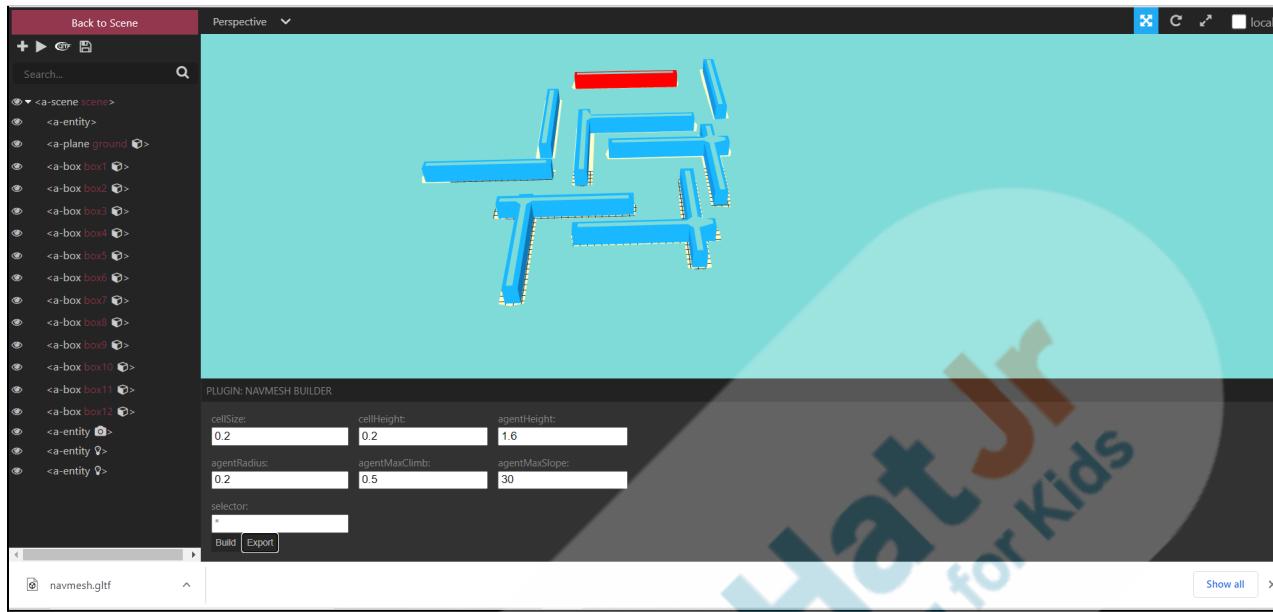
We should **not include cameras in the navmesh**, otherwise we won't be able to walk in the scene. Can you tell me why?

Yes. Great!

While building meshes we have to keep in mind that we do not include the camera in the scene.

ESR: We move with the camera in the scene.

	<p>Let's comment the camera entity in the code for now and open Visual Inspector again.</p> <p>We can also set the cellSize property to 0.2 and build the navmesh.</p>	
		
	<p>Once the navmesh is created now we can “Export” it.</p> <p>This will create a gLTF model file of navmesh.</p>	

			
	<p>Once the file is downloaded, go to the download folder and copy the file into your working folder.</p>		
<table border="1"> <tr> <td style="text-align: center;">  shooter <hr/>  navmesh Type: 3D Object </td> </tr> </table>			 shooter <hr/>  navmesh Type: 3D Object
 shooter <hr/>  navmesh Type: 3D Object			
	<p>To see how navmesh looks after it has been converted to the gLTF file, we can use the gLTF viewer.</p> <p>Link: https://gltf-viewer.donmccurdy.com/</p>		

gltf Viewer | threejs r123 | THREEGLTFLoader@r123 | help & feedback | source

Drag glTF 2.0 file or folder here

Upload

We can directly upload the file to view.

Organize ▾ New folder

This PC

3D Objects

Desktop

Documents

Downloads

Music

Pictures

Videos

Windows-SSD (C)

Data (D):

Network

shooter

navmesh

3D Object

23.1 KB

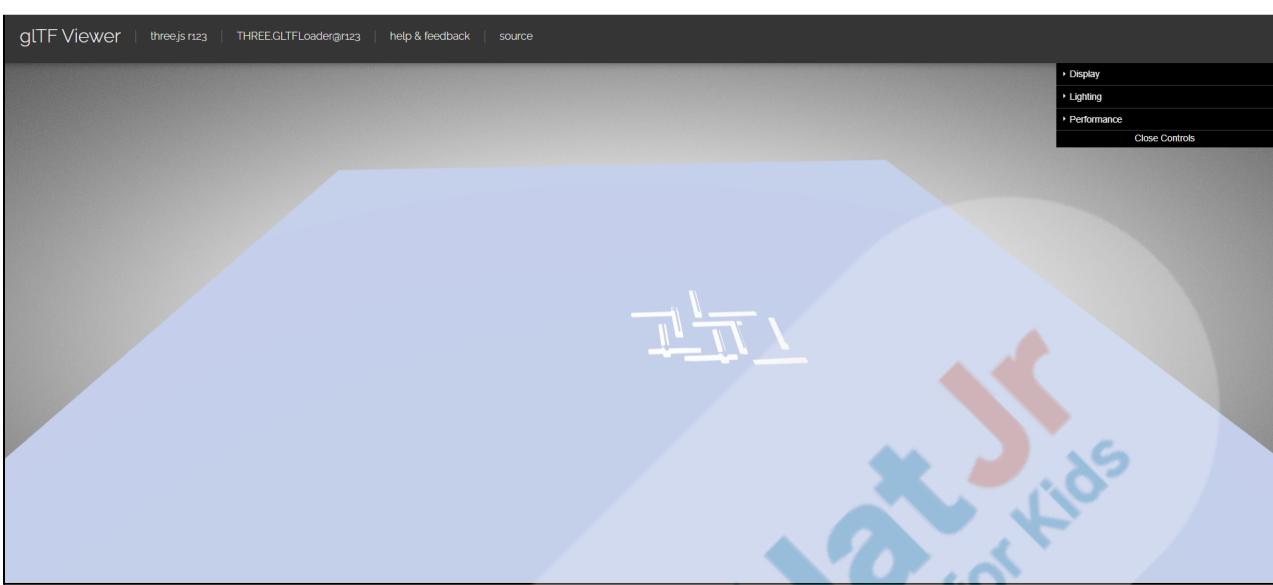
File name: navmesh

All Files

Open Cancel

file or folder here

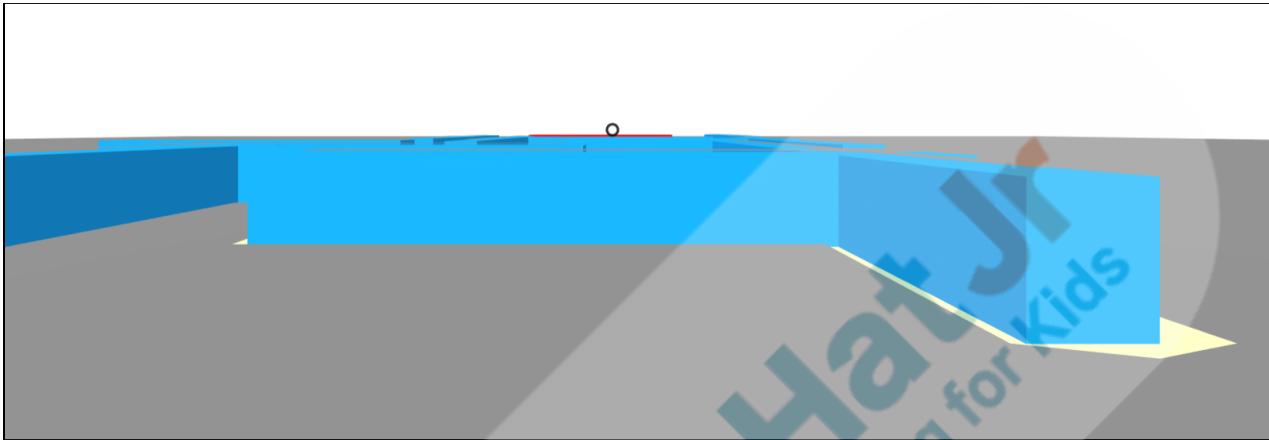
Upload



	<p>You can see the navmesh model in the gLTF viewer and holes being cut into that.</p> <p>Now once we have the file in the working folder, we can use any other glTF model that we use in the scene.</p> <p>We need to attach the “nav-mesh component” to the “gltf-model” entity to make the navmesh model work.</p>	
--	--	--

```
<!--Assets-->
<a-assets>
| <a-asset-item id="collider" src="../models/navmesh.gltf"></a-asset-item>
</a-assets>
```

```
<!-- Navigation Mesh -->
<a-entity id="navigation-mesh" gltf-model="#collider" visible="true" nav-mesh>
</a-entity>
```



Still the navigation mesh does not work!

To move around the scene containing the navmesh, we need to use the “**movement-controls**” component which is a part of “**aframe-extras**” library.

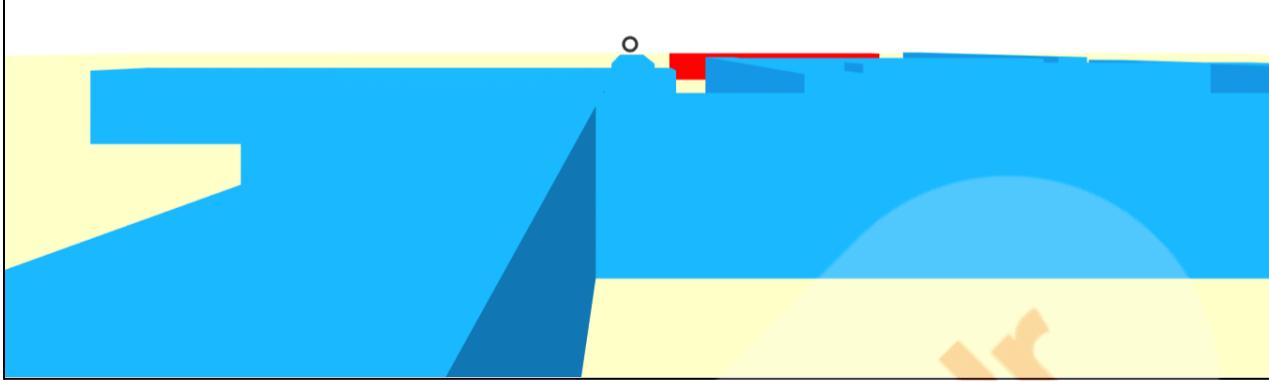
Link:

<https://cdn.jsdelivr.net/gh/donmccurdy/aframe-extras@v6.1.1/dist/aframe-extras.min.js>

```
<script
src="https://cdn.jsdelivr.net/gh/donmccurdy/aframe-extras@v6.1.1/dist/aframe-extras.min.js">
</script>
```

Let's add the movement-controls component we using <a-entity>:

	<ul style="list-style-type: none"> We can set the speed and constrainToNavMesh properties of the movement-controls component. 	
<pre><a-entity movement-controls="speed: 0.2; constrainToNavMesh: true"> </a-entity></pre>		
	<ul style="list-style-type: none"> The camera and cursor entity must be the child entity of the movement-controls entity. Also the position of the movement-controls entity must be ahead of the camera in the positive z-axis. <p>Note: Set the visible attribute of the navmesh glTF model to false for the final scene.</p>	
<pre><a-entity movement-controls="speed: 0.2; constrainToNavMesh: true" position="0 0 2"> <!--Camera and Cursor--> <a-entity id="camera" camera position="0 1.6 0" look-controls="pointerLockEnabled: false"> <a-cursor></a-cursor> </a-entity> </a-entity></pre>		
<pre><!-- Navigation Mesh --> <a-entity id="navigation-mesh" gltf-model="#collider" visible="false" nav-mesh> </a-entity></pre>		
	You can see that now the camera entity will not cross through the objects in the scene.	

		
	Since you have learned how to create navigation meshes, you will create the navigation of our game environment.	
Teacher Stops Screen Share		
	Now it's your turn. Please share your screen with me.	
 <p>Teacher Starts Slideshow Slide 17 to 19 Refer to speaker notes and follow the instructions on each slide.</p>		
We have one more class challenge for you. Can you solve it?		
Let's try. I will guide you through it.		
 <p>Teacher Ends Slideshow</p>		
STUDENT-LED ACTIVITY - 20 mins		
<ul style="list-style-type: none">Ask the student to press the ESC key to come back to the panel.Guide the student to start screen share.Teacher gets into fullscreen.		

ACTIVITY <ul style="list-style-type: none"> • Create a Navigation Mesh of the game environment. 		
Step 3: Student-Led Activity (20 mins)	<p><i>The teacher guides the student to clone the code from the student activity 1.</i></p> <p><u>[Student Activity 1]</u></p>	
	<p>Now to create the navmesh of our game scene, the position of each object has to be fixed.</p> <p>But in the previous class we had random positions of the boxes.</p> <p>First we need to fix the positions of these.</p> <p>What do you think can be done for that?</p> <p>Well, we can keep all the values stored in an array.</p> <p><i>Guide the student to open gameObjects.js file.</i></p> <p><i>We have two arrays px and pz to store x and z positions.</i></p> <p><i>Guide the student to update the variables values using array index in the “boxes” component.</i></p>	<p>ESR: Varied.</p>

	<pre>posX = px[i]; posY = 1.5; posZ = pz[i];</pre>	
	<p>Now once the positions are fixed, let's create the navmesh of the environment.</p> <p><i>Guide the student to comment on the camera entity.</i></p>	
<pre><!--Camera and Cursor--> <!-- <a-entity id="camera" camera position="0 1.6 0" wasd-controls look-controls="pointerLockEnabled: false"> <a-entity id="weapon" gltf-model="#shooter" position="0 -4.4 3" rotation="0 180 0" scale="0.35 1 1" player-movement> </a-entity> <a-cursor></a-cursor> </a-entity> --></pre>		
		

```
<a-scene id="scene" inspector-plugin-recast>
  <!--Assets-->
  <a-assets>
    <a-asset-item id="shooter" src="./models/shooter/gltf"></a-asset-item>
```



Perspective

PLUGINS: NAVMESH BUILDER

cellSize:	cellHeight:	agentHeight:
0.3	0.2	1.6

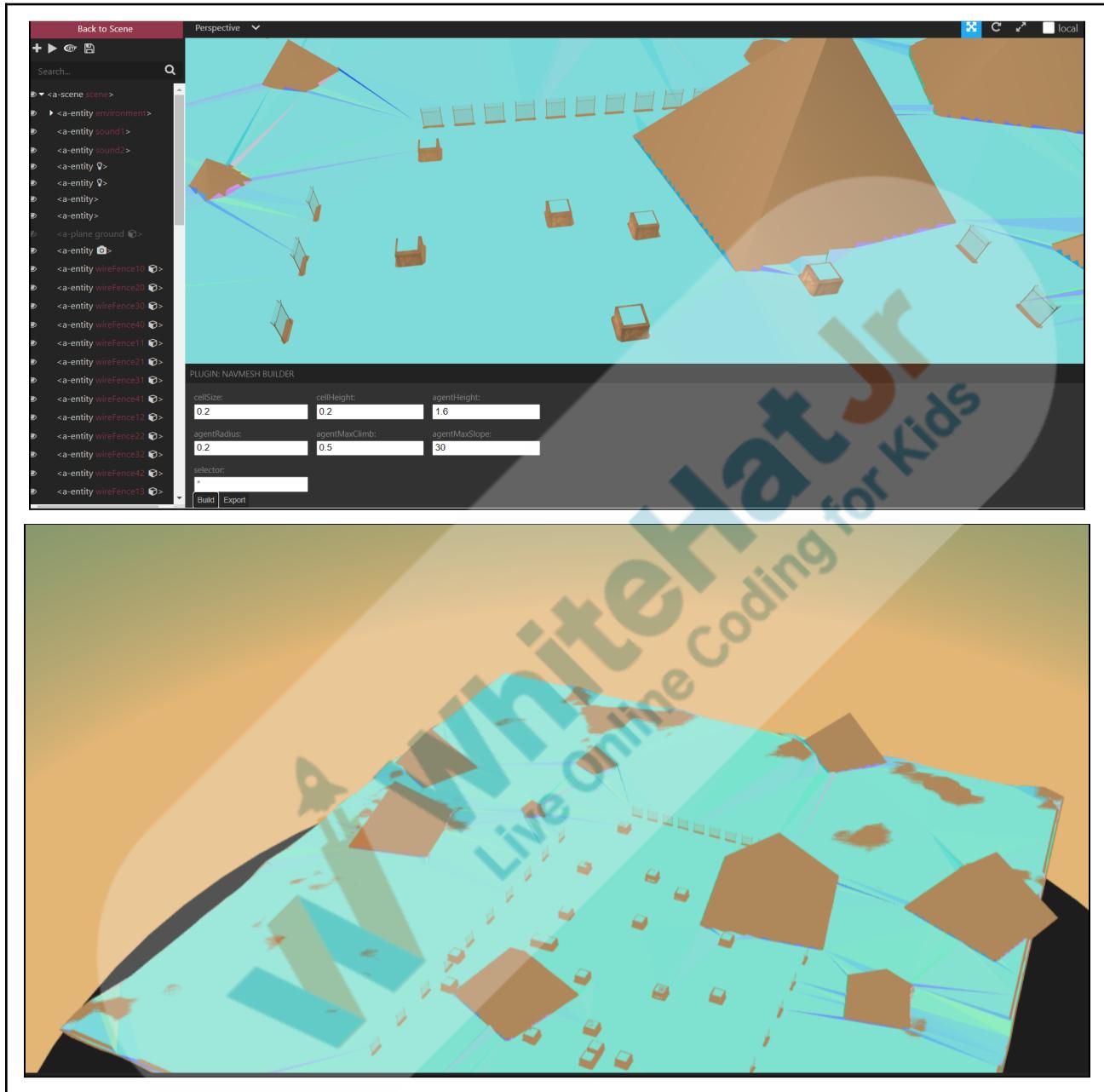
agentRadius:	agentMaxClimb:	agentMaxSlope:
0.2	0.5	30

selector: *

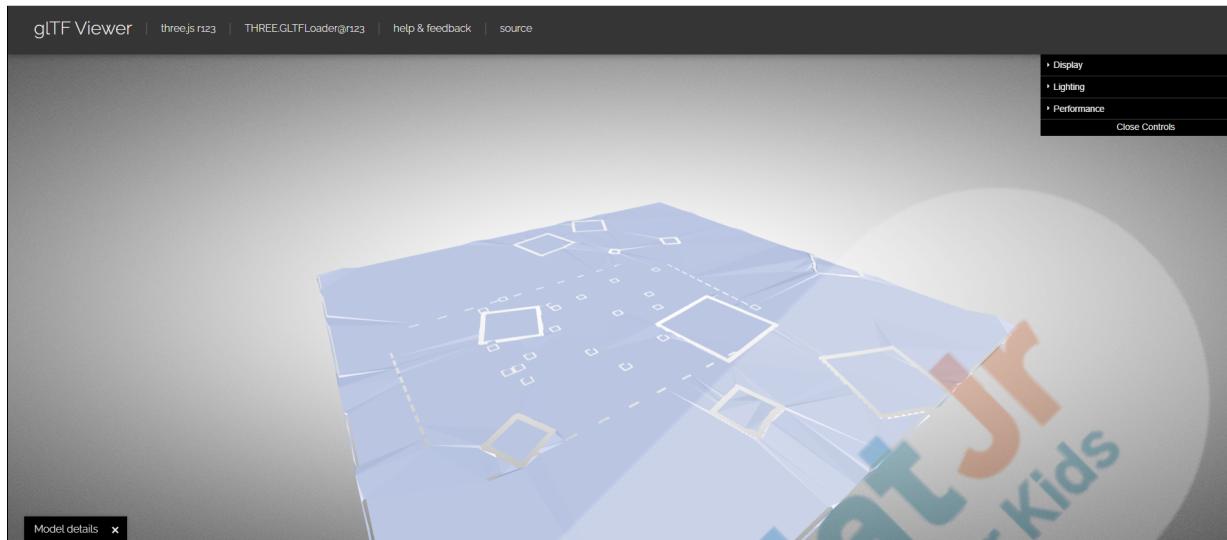
Build | Export

Guide the student to build the navmesh and to export the navmesh.

Guide the student to add it to the working folder and view the model in the glTF viewer.






 Model details x

Guide the student to add it into <a-assets> and use the id to add it using “gltf-model”.

Guide the student to attach the nav-mesh component.

Guide the student to add the “movement-controls” and set the position and test the output.

```
<a-assets>
    <a-asset-item id="shooter" src="./models/shooter/gun.gltf"></a-asset-item>
    <a-asset-item id="collider" src="./models/navmesh.gltf"></a-asset-item>
    <audio id="shoot" src="./sounds/shoot.mp3"></audio>
    <audio id="footstep" src="./sounds/footStep.mp3"></audio>

</a-assets>
```

```
<!-- Navigation Mesh -->
<a-entity id="navigation-mesh" gltf-model="#collider" visible="false" nav-mesh>
</a-entity>
```

```
<a-entity movement-controls="speed: 0.2; constrainToNavMesh: true" position="0 1 2">
  <!--Camera and Cursor-->
  <a-entity id="camera" camera position="0 1.6 0" look-controls="pointerLockEnabled: false">
    <a-entity id="weapon" gltf-model="#shooter" position="0 -4.4 3" rotation="0 180 0" scale="0.35 1 1">
      | player-movement
      </a-entity>
      <a-cursor></a-cursor>
    </a-entity>
  </a-entity>
</a-entity>
```

Guide the student to adjust the dimensions of the boxes to avoid the gun model to enter into the box.

```
schema: {
  height: { type: "number", default: 2 },
  width: { type: "number", default: 2 },
  depth: { type: "number", default: 2 },
},
```



Guide the student to give the id movement-controls entity and use that to adjust the bullet shooting position in the shoot.js file.

```
<a-entity id="camera-rig" movement-controls="speed: 0.2; constrainToNavMesh: true" position="0 1 2">
  <!--Camera and Cursor-->
  <a-entity id="camera" camera position="0 1.6 0" look-controls="pointerLockEnabled: false">
    <a-entity id="weapon" gltf-model="#shooter" position="0 -4.4 3" rotation="0 180 0" scale="0.35 1 1">
      player-movement
    </a-entity>
    <a-cursor></a-cursor>
  </a-entity>
</a-entity>
```

```
var cam = document.querySelector("#camera-rig");
pos = cam.getAttribute("position");

bullet.setAttribute("position", {
  x: pos.x,
  y: pos.y+1,
  z: pos.z-0.5,
});
```

Teacher Guides Student to Stop Screen Share

WRAP UP SESSION - 5 mins



Teacher Starts Slideshow
Slide 20 to 23

Activity details

Following are the WRAP-UP session deliverables:

- Appreciate the student.
- Revise the current class activities.
- Discuss the quizzes.

WRAP-UP QUIZ

Click on In-Class Quiz



Continue WRAP-UP Session
Slide 24 to 29

Activity Details

Following are the session deliverables:

- Explain the facts and trivia
- Next class challenge
- Project for the day
- Additional Activity (Optional)

FEEDBACK

- Compliment the student for her/his effort in the class.
- Encourage the student to think and come up with their own solutions.

You get a “hats-off”.
Alright. See you in the next class.

Make sure you have given at least 2 Hats Off during the class for:



PROJECT OVERVIEW DISCUSSION

Refer the document below in Activity Links Sections

Teacher Clicks

✗ End Class

Additional Activities	<p><i>Encourage the student to write reflection notes in their reflection journal using markdown.</i></p> <p>Use these as guiding questions:</p> <ul style="list-style-type: none"> • What happened today? <ul style="list-style-type: none"> ◦ Describe what happened. ◦ The code I wrote. • How did I feel after the class? • What have I learned about programming and developing games? • What aspects of the class helped me? What did I find difficult? 	<p><i>The student uses the markdown editor to write their reflections in a reflection journal.</i></p>
------------------------------	--	--

Activity	Activity Name	Links
Teacher Activity 1	Teacher Activity 1	https://github.com/whitehatjr/PRO-C16-4-Teacher-Actitvity
Teacher Activity 2	A-Frame aframe-inspector-plugin-recast Link.	https://recast-api.donmccurdy.com/aframe-inspector-plugin-recast.js
Teacher Activity 3	A-Frame aframe-extras Link.	https://cdn.jsdelivr.net/gh/donmccurdy/aframe-extras@v6.1.1/dist/aframe-extras.min.js
Teacher Activity 4	Teacher Activity 1 Reference Code	https://github.com/whitehatjr/PRO-C16-4-Teacher-Activity-Ref
Teacher Activity 5	Teacher Activity 1 Output Reference	https://curriculum.whitehatjr.com/PRO+Asset/A-Frame+NavMesh.mp4
Teacher Activity 6	Output Reference	https://curriculum.whitehatjr.com/PRO+Asset/Navigation+Mesh+in+VR.mp4

Teacher Activity 7	Teacher Reference Code	https://github.com/whitehatjr/PRO-C16_4-Teacher-Ref
Student Activity 1	Boilerplate Code.	https://github.com/whitehatjr/PRO-C16_4-Student-Activity
Teacher Reference 1	The glTF Viewer	https://gltf-viewer.donmccurdy.com/
Teacher Reference 2	A-Frame Navmesh Demo1 Creation	https://drive.google.com/file/d/1l5z3sM1VSDbY-nVgsPU-SX0tXt1fgq3h/view?usp=sharing
Teacher Reference 3	A-Frame Navmesh Demo2 Creation	https://drive.google.com/file/d/1YkC_QiT2mTHfzIYTx9-3_-32VbmRLwFE/view?usp=sharing
Teacher Reference 4	Project Document	https://s3-whjr-curriculum-uploads.whjr.online/495a9078-6320-4416-8152-4413af2688c0.pdf
Teacher Reference 5	Project Solution	https://github.com/whitehatjr/PRO-C16_4-Project-Solution
Teacher Reference 6	Visual-Aid	https://s3-whjr-curriculum-uploads.whjr.online/7cbf4390-a528-4763-b471-2ca8003f99e6.html
Teacher Reference 7	In-Class Quiz	https://s3-whjr-curriculum-uploads.whjr.online/cd94eb22-3d6c-4aee-9e32-9f530ff1ff7.pdf