

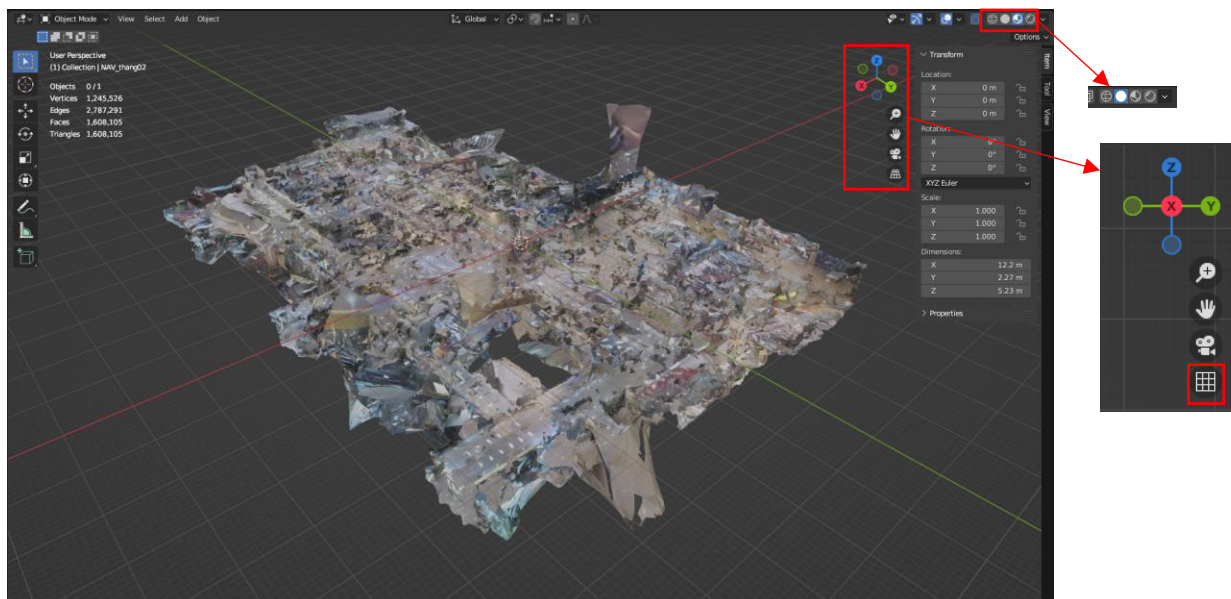
Hướng dẫn tạo NavMesh cho nội dung điều hướng

1. Mở đầu

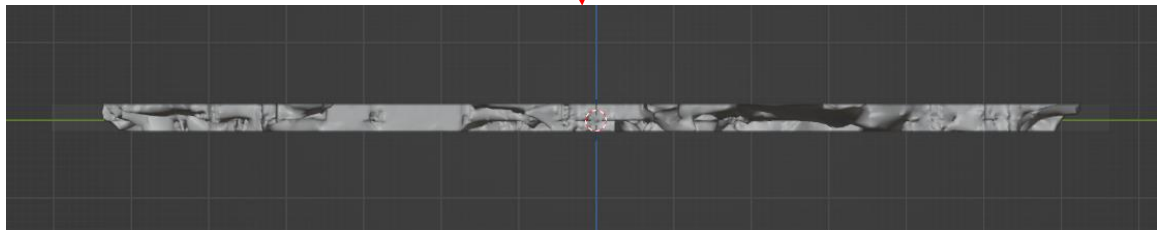
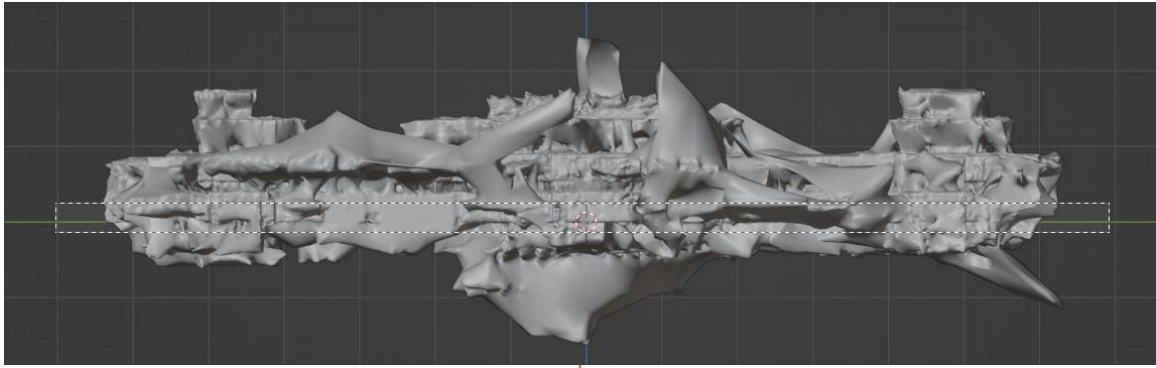
Tài liệu hướng dẫn các bước tạo và thêm NavMesh cho nội dung điều hướng sử dụng sau khi đã có Mesh của khu vực 3D tạo ra dựa trên video. Nội dung được tạo trên phần mềm Blender và Unity Editor.

2. Vẽ mesh

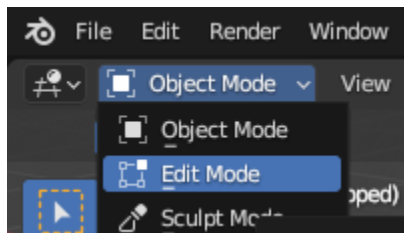
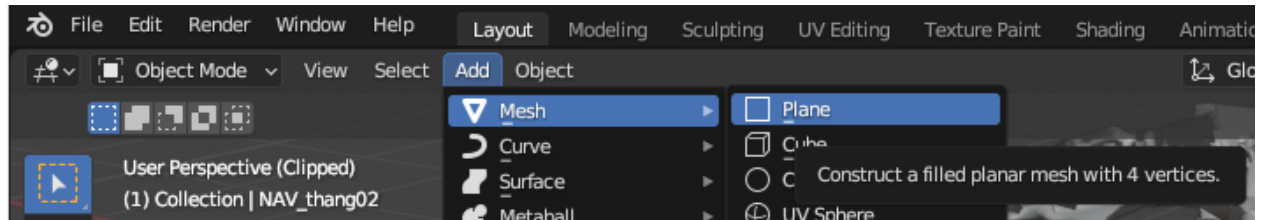
Sau khi đã có mesh, mở ứng dụng đồ họa 3D Blender và nhập file Mesh và mở chế độ xem Solid, góc nhìn ngang:



Xem mặt cắt 3D của từng tầng để vẽ bằng cách nhấn **Alt+B** và kéo thả chuột theo khu vực cắt:

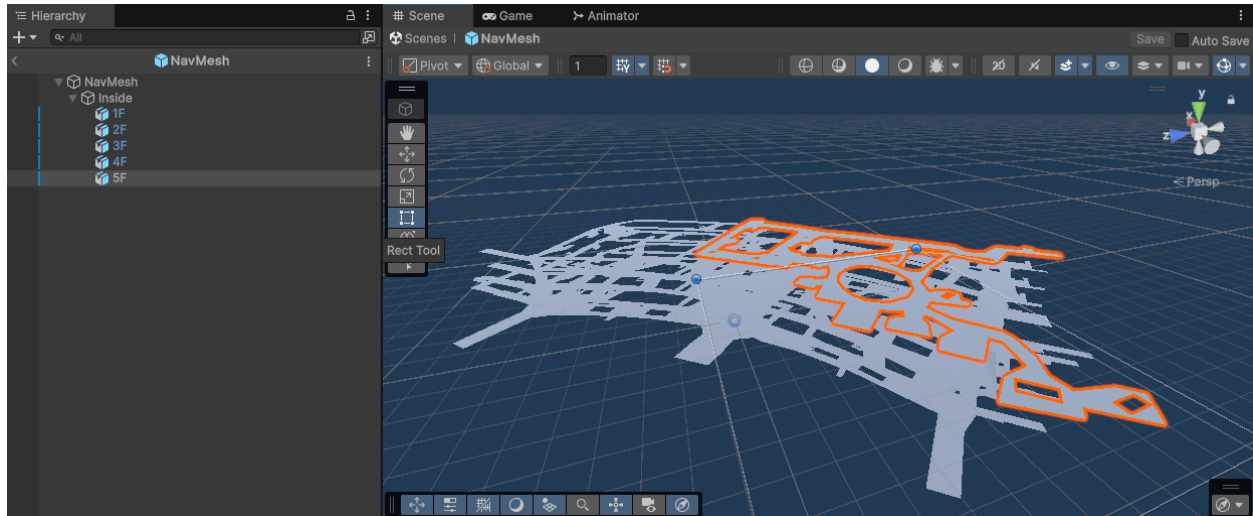


Thực hiện thêm một mesh mới và vẽ theo các khu vực di chuyển:



2.1. Mỗi một tầng được vẽ thành một mesh riêng tách biệt với nhau

Mỗi một tầng nên được vẽ thành một mesh riêng tách biệt với nhau. Trong mesh của mỗi tầng, các thành phần khác nhau cũng có thể được chia làm các mesh khác nhau với tên khu vực. Việc này giúp hệ thống điều hướng hiển thị được tên khu vực khi người dùng thực hiện điều hướng. Nếu không cần hiển thị khu vực thì chỉ tạo một mesh cũng được.



2.2. Liên tục thay đổi qua lại giữa view solid và material preview

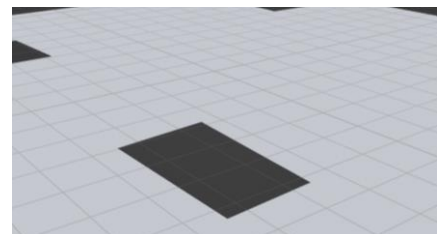
So sánh giữa mesh solid đang vẽ và mesh có ảnh để xác định chính xác mình có đang vẽ đúng không



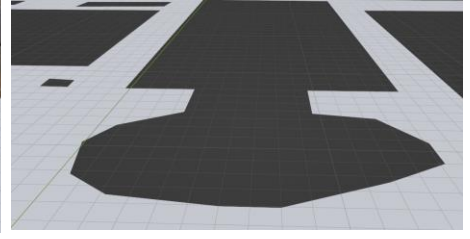
2.3. Không vẽ Mesh tại những khu vực không thể di chuyển đến

Các khu vực này bao gồm:

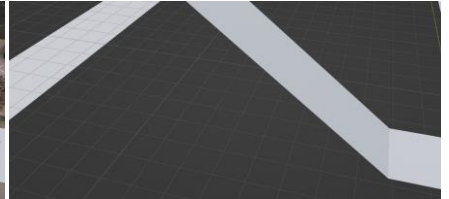
- Cột chống:



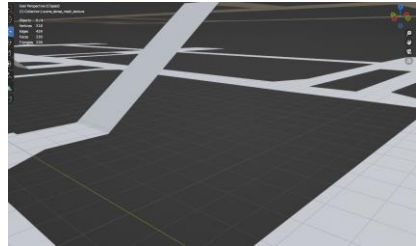
- Quầy hàng, sẵn khấu:



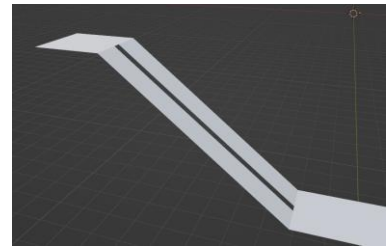
- Bên dưới chân cầu thang:



- Giếng trời:

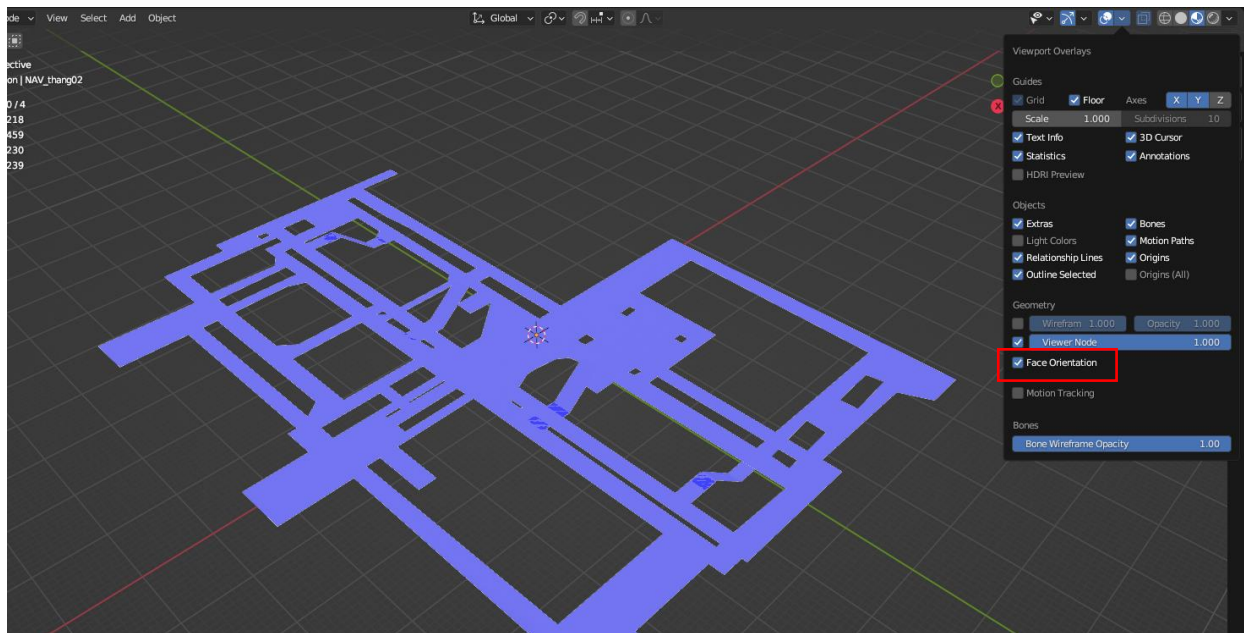


- Khoảng trống giữa hai thang liên kề



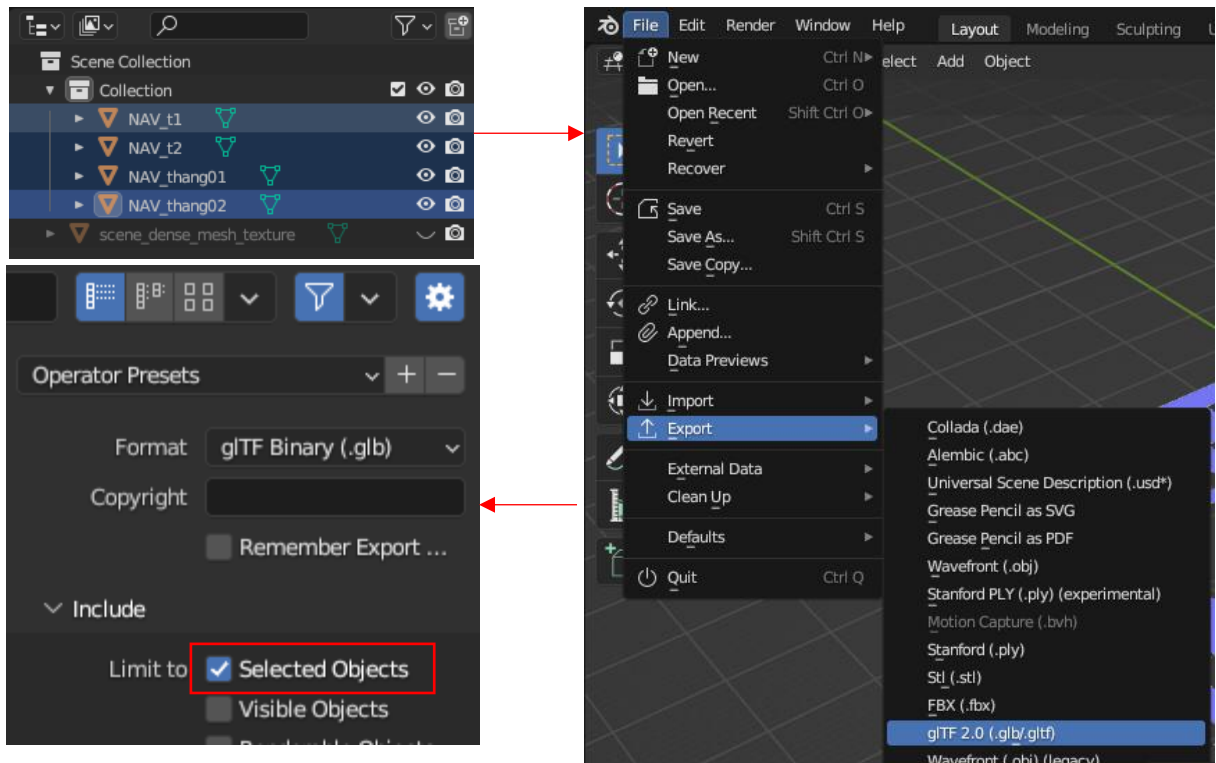
2.4. Đảm bảo hướng quay của NavMesh

Sau khi vẽ NavMesh hãy đảm bảo tất cả các mặt đã vẽ đều hướng lên trên:



2.5. Xuất file đã vẽ

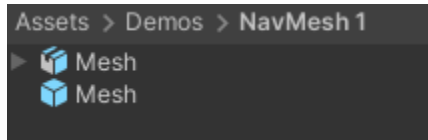
Sau khi vẽ xong, ta thực hiện xuất file:



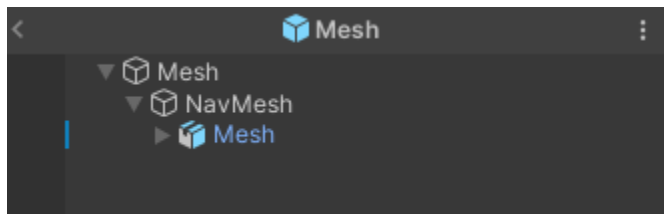
3. Tạo AssetBundle cho NavMesh

3.1. Chuẩn bị bundle

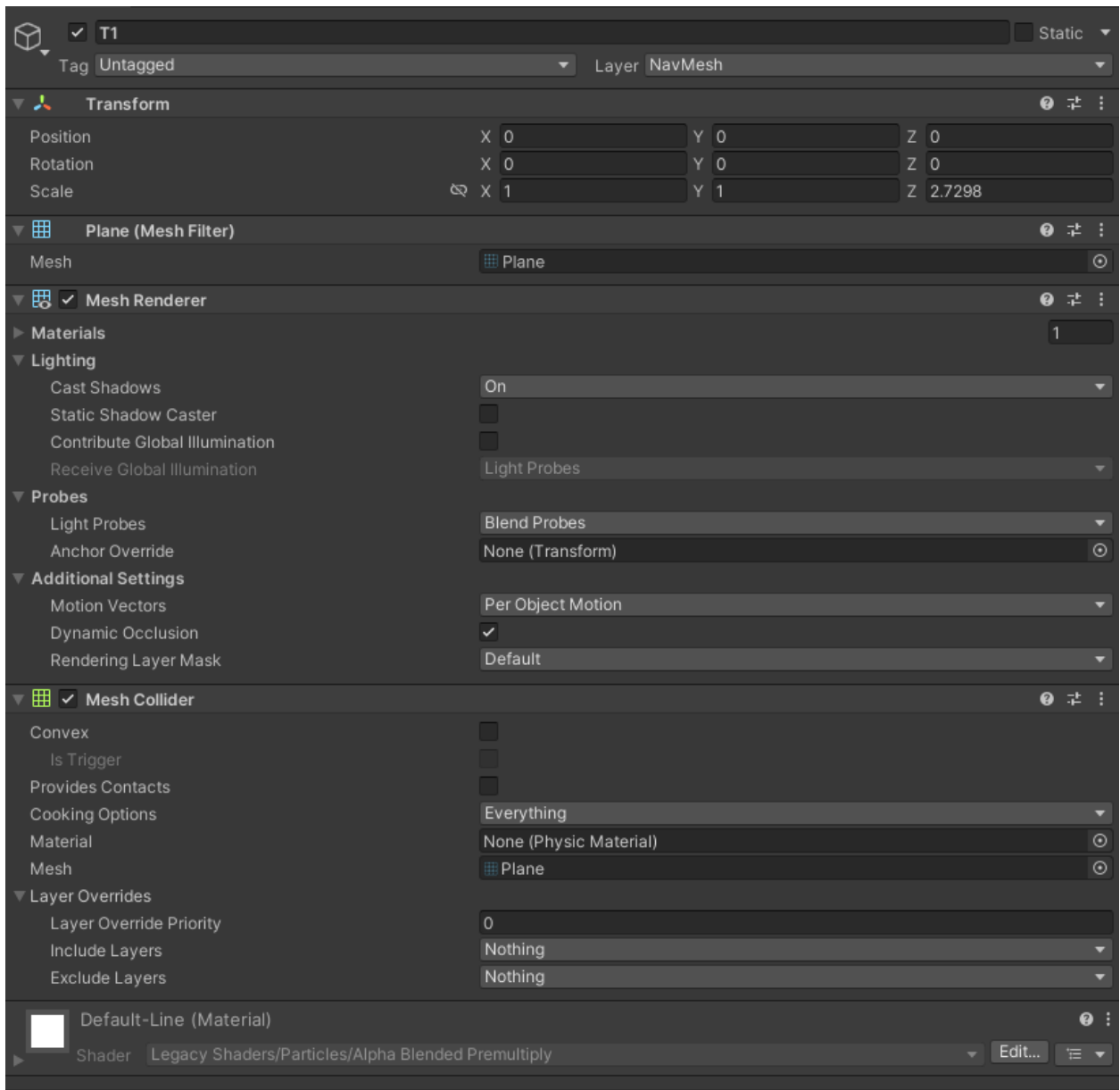
Trong ứng dụng Unity Xense Editor/Unity Xense Asetbundle Builder, tạo ra một prefab mới bên cạnh file Mesh đã tạo:



Mở Prefab và thêm một gameObject con tên “NavMesh”, thực hiện thêm Mesh đã tạo vào như gameObject con của NavMesh:

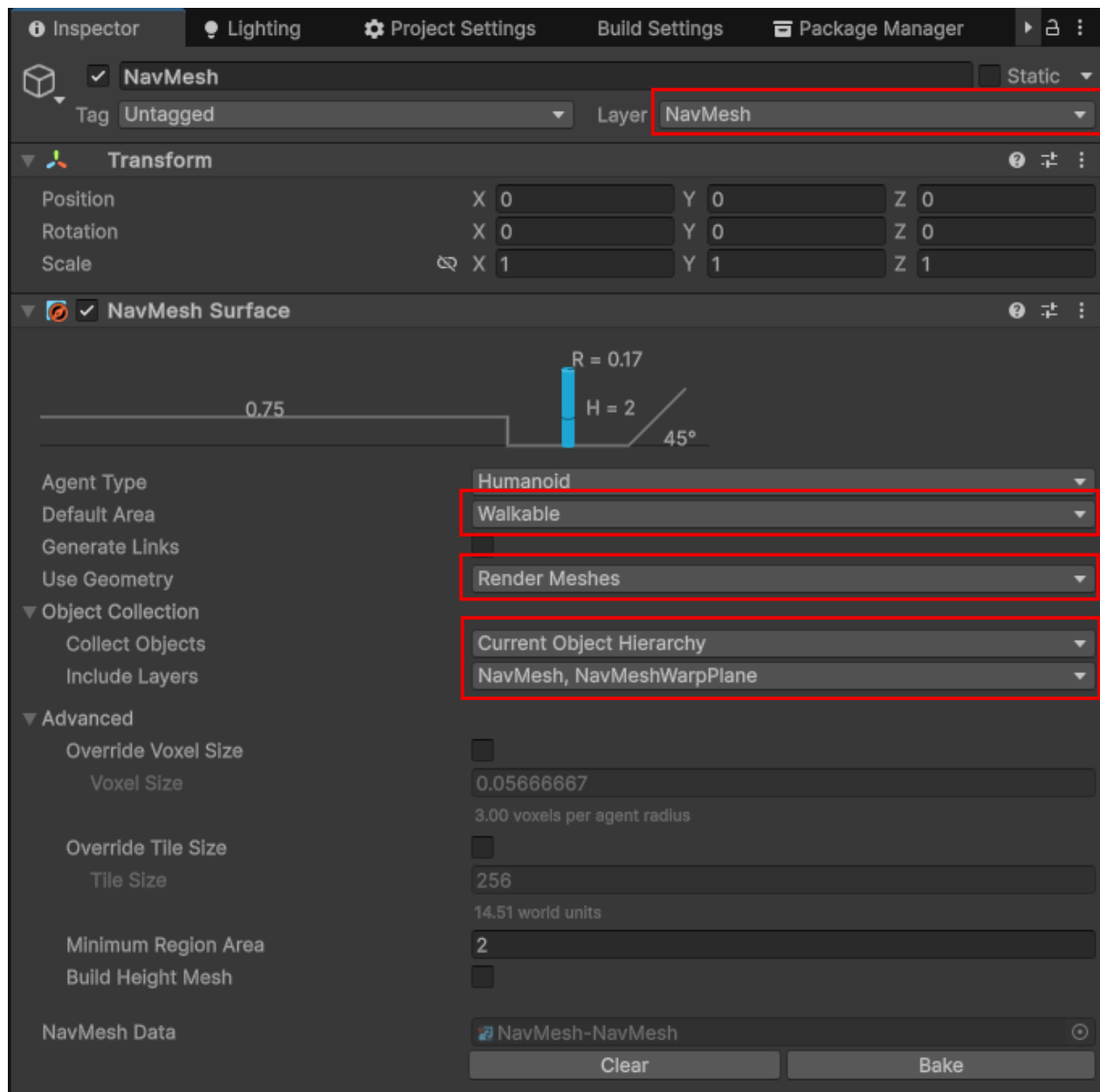


Thay đổi tên mesh con thành rỗng hoặc tên khu vực, việc này giúp hiển thị vị trí người dùng khi thực hiện điều hướng.



Thực hiện thay đổi trên NavMesh:

- Thay đổi layer thành layer 6: “NavMesh”, thực hiện cùng lúc cho các object con
- Thêm thành phần NavMeshSurface
- Chọn: **Use GeoMetry -> Render Meshs**
- Chọn: **Object Collection -> Collect Objects -> Curent Object Hierarchy**



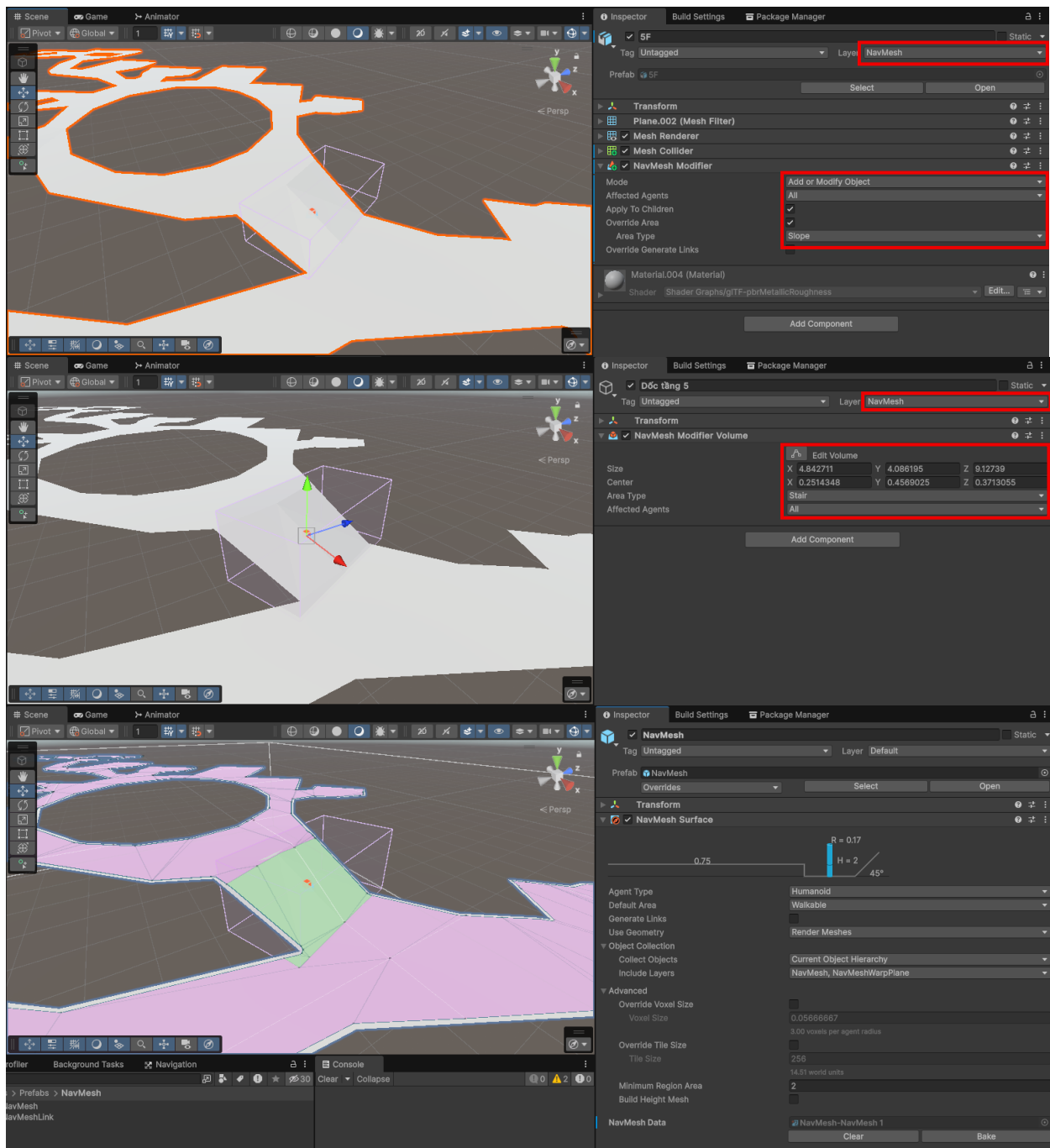
3.2. Chọn Area Type cho Mesh

Trên một Tầng/Mesh có thể có nhiều loại khu vực khác nhau như con dốc, cầu thang, cổng một chiều,... Để đánh dấu các khu vực này, thực hiện:

- Thêm một GameObject với component **NavMesh Modifier Volume** với đúng **Layer**
- Di chuyển object mới đến khu vực và đặt bound bao quát khu vực cần thay đổi
- Chọn **Area Type** tương ứng

Trường hợp muốn thay đổi cả một mesh hoặc từng khu vực riêng trên mesh đã được tách ra thành các mesh khác nhau, chỉ cần thêm **NavMesh Modifier** vào object mesh, Chọn **Override Area** và chọn **Area Type** tương ứng, trong ví dụ thì khu vực tầng 5 được coi là một con dốc (thực tế mesh tầng 5 có **Area Type** là **Walkable** với một số khu vực nhỏ là **Slope**), nhưng ta có thể thực hiện một trong các cách:

- Tách Mesh tầng 5 thành khu vực đi bộ, con dốc, cầu thang,... Thêm **NavMesh Modifier** cho từng mesh con
- Giữ Mesh tầng 5 là một mesh riêng, thực hiện thêm nhiều **GameObject** với **NavMesh Modifier Volume** để thay đổi từng phần của mesh

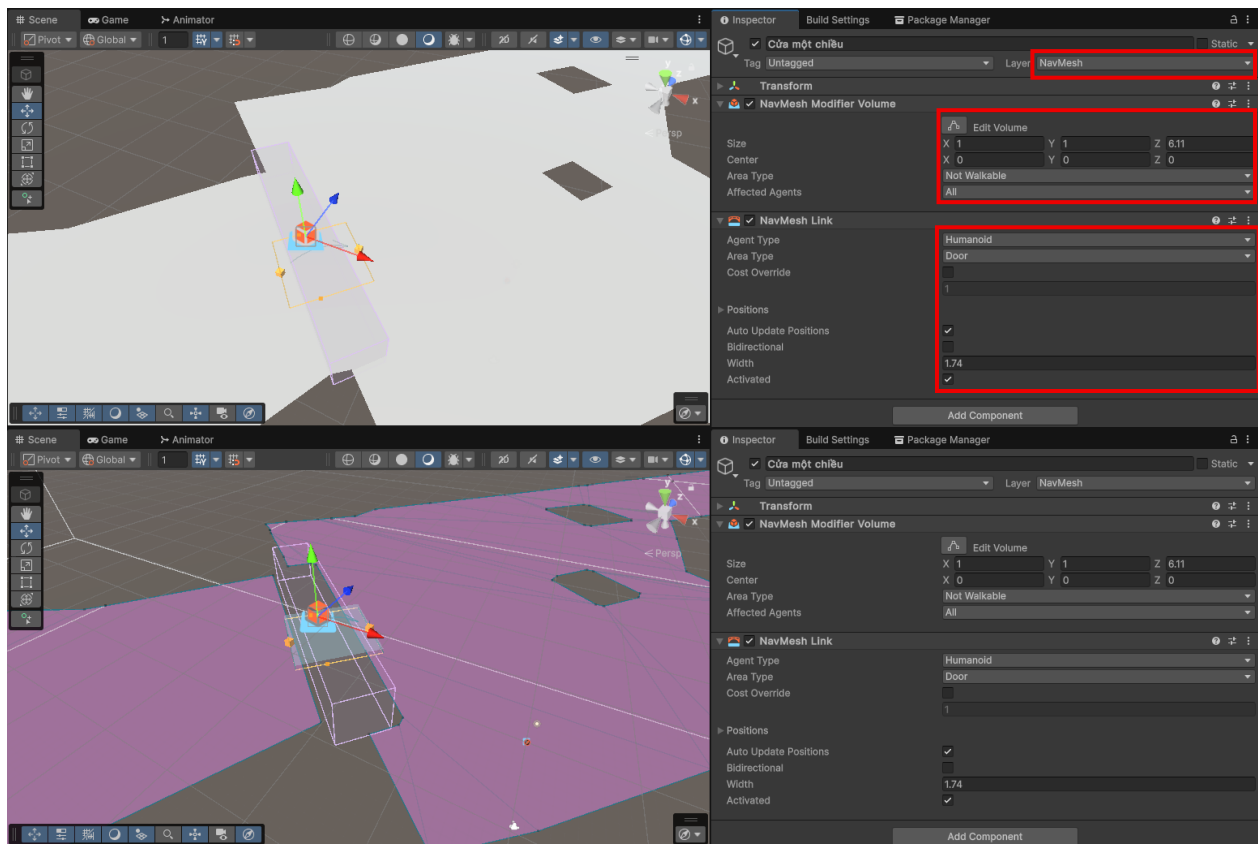


3.3. Đặt các khu vực di chuyển một chiều

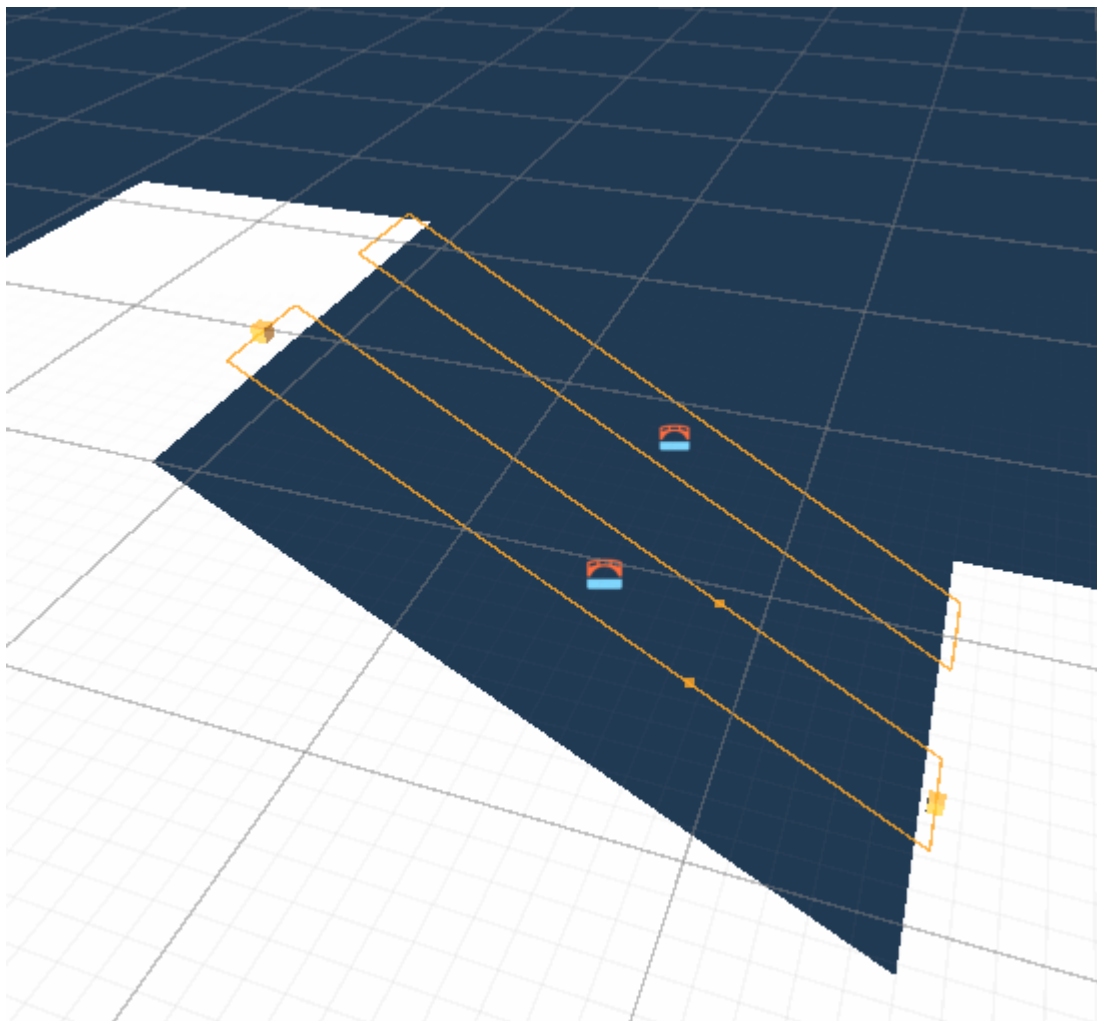
Trong không gian điều hướng có thể xuất hiện các khu vực đặc biệt chỉ cho phép đi theo một chiều như cổng ra, vào tách biệt thang cuốn. Để thực hiện điều hướng qua khu vực một cách chính xác, ta thực hiện:

- Thêm một GameObject với component **NavMesh Modifier Volume** với đúng **Layer**
- Di chuyển object đến vị trí và đặt **Area Type** là **Not Walkable**
- Thêm component **NavMesh Link** và di chuyển hai đầu link đến hai bên khu vực tương ứng
- **Start Point** di chuyển đến vị trí vào
- **End Point** di chuyển đến vị trí ra
- Thay đổi giá trị **Width** ứng với chiều rộng thực tế của khu vực.
- Chọn **Auto Update Position**
- Chọn **Area Type** tương ứng

Lưu ý **NavMesh Link** có hỗ trợ di chuyển hai chiều nếu chọn **Bidirectional**.



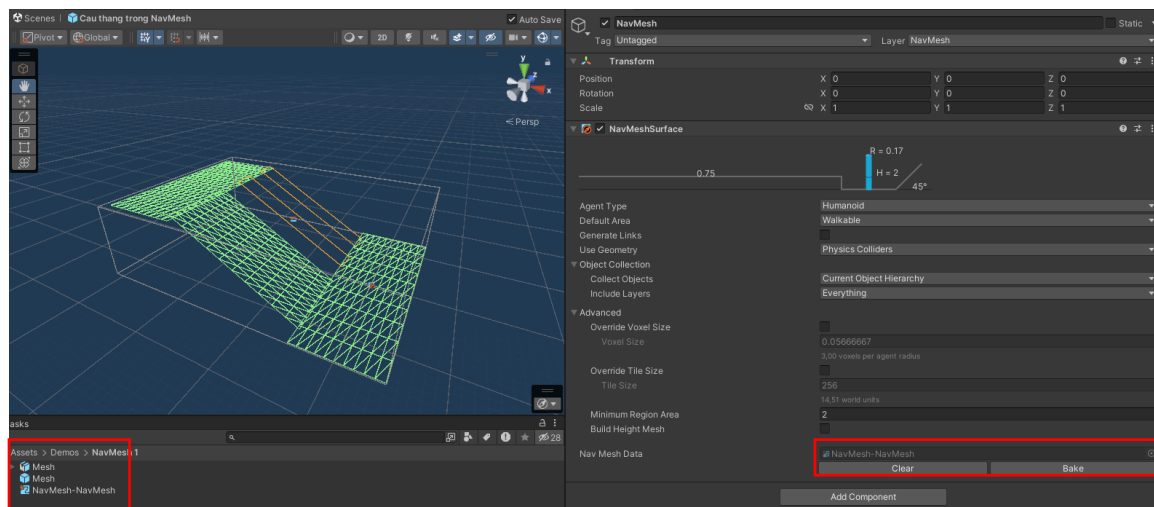
Trong trường hợp vị trí đã không có mesh thì có thể không cần thêm thành phần **NavMesh Modifier Volume**, tương tự với khu vực thang cuốn một chiều, ví dụ ở hình dưới khi người vẽ mesh chưa vẽ mesh cầu thang nhưng vẫn có thể thêm **NavMesh Link**, thực tế thì người viết bài khuyến khích vẽ mesh cầu thang và dùng thêm **NavMesh Modifier Volume** như ví dụ về khu vực của một chiều.



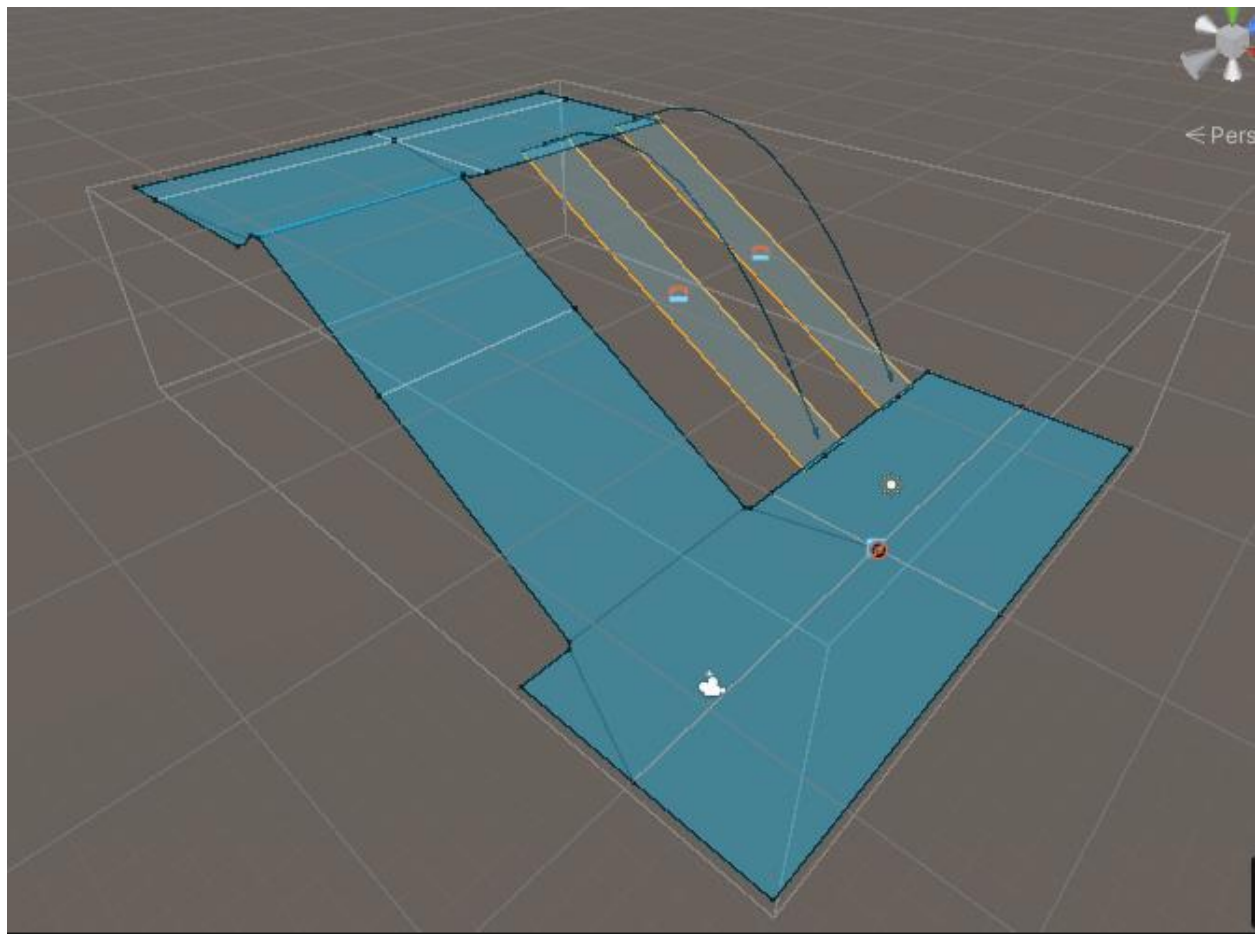
3.4. Build NavMesh

Trong các Mesh con thực hiện tắt thành phần **Mesh Renderer**

Trong “NavMesh” thực hiện nhấn nút **Build**, sau khi build sẽ xuất hiện một khung hình hộp màu trắng bao quanh toàn bộ khu vực Mesh:



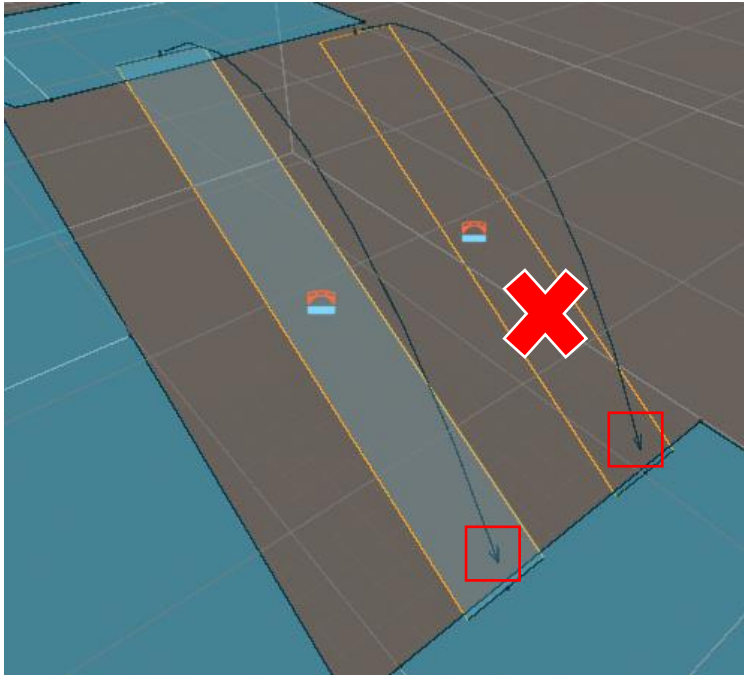
Thực hiện kiểm tra lần cuối bằng cách kéo prefab vào trong scene bất kỳ, nhấn nút hiện NavMesh:



Lúc này ta cần thấy:

- Khu vực di chuyển được xuất hiện Mesh xanh nhạt (màu mặc định của **Area Type** loại **Walkable**), bao quanh bởi khung có hình hộp màu trắng, với các **Area Type** khác sẽ xuất hiện các màu khác.

- Các **NavMeshLink** có chiều mũi tên (nằm ở đầu đường cong) chỉ đúng hướng di chuyển, thân Link có màu xanh nhạt. Nếu chọn **Bidirectional** thì sẽ có mũi tên ở cả hai đầu đường cong.



3.5. Build AssetBundle

Thực hiện build AssetBundle như bình thường:

- Đặt tên bundle cho prefab
- Chọn BuildAssetBundle

4. Thêm NavMesh trên CMS

Trong phần nội dung , tìm đến 3D space đang được dựng NavMesh và thêm đường dẫn đến file bundle vào trường nav mesh url (bỏ đi đoạn post-fix):

Source

Format - **B** *I* U ~~S~~ | x, x² | I_x | := :: | ≡ ≢ | „ | ≡ ≢ ≣ ≤ | ∫ ∫ ∫ ∫ | ∫ ∫ ∫ ∫ | ∫ ∫ ∫ ∫ | ∫ ∫ ∫ ∫

Text format **Filtered HTML** ▼

- © 2006 The Authors
Journal compilation © 2006 Blackwell Publishing Ltd

https://s3.educationxr.vn/ar-world/NavigationMesh/aeonib_nav_v2

Delete