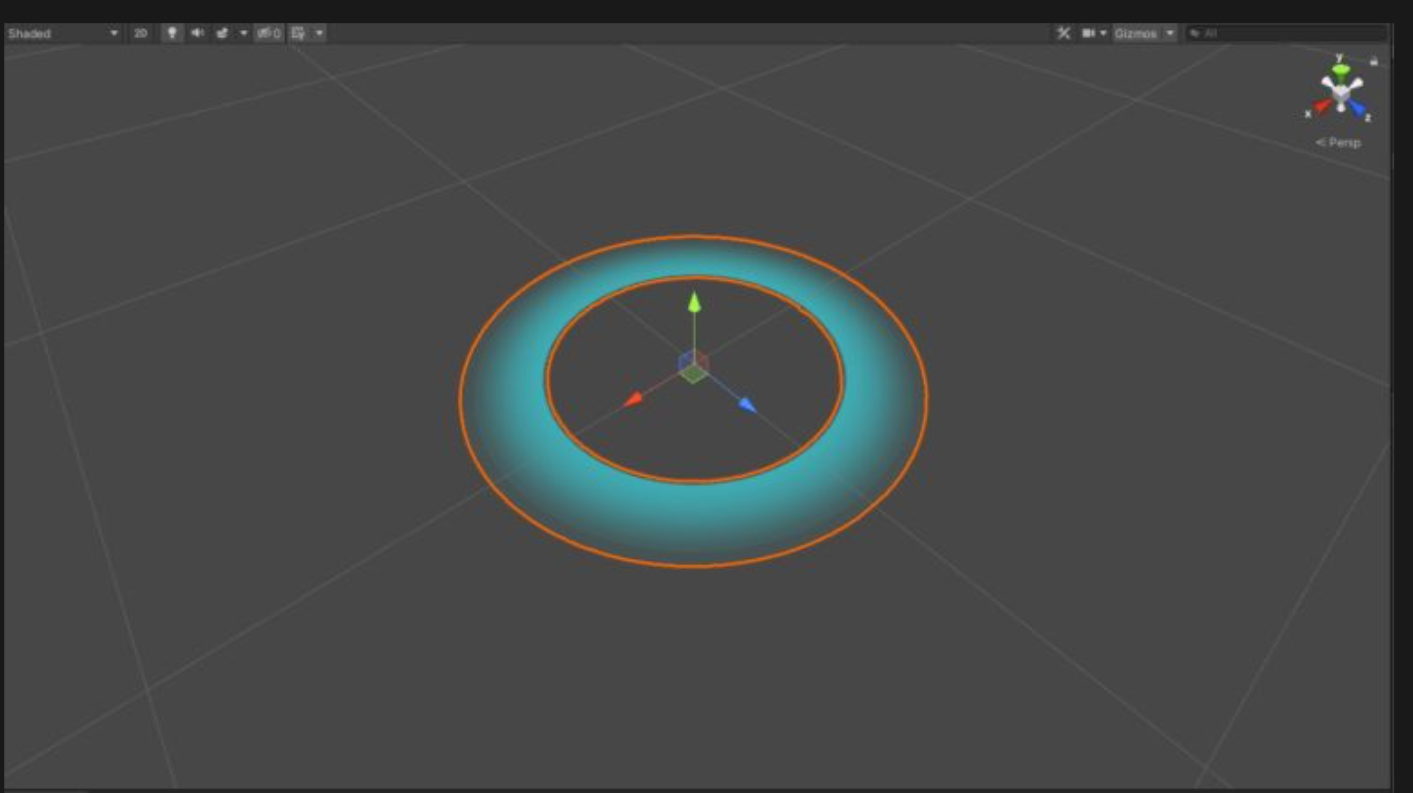
FINAL ASSIGNMENT AR/VR

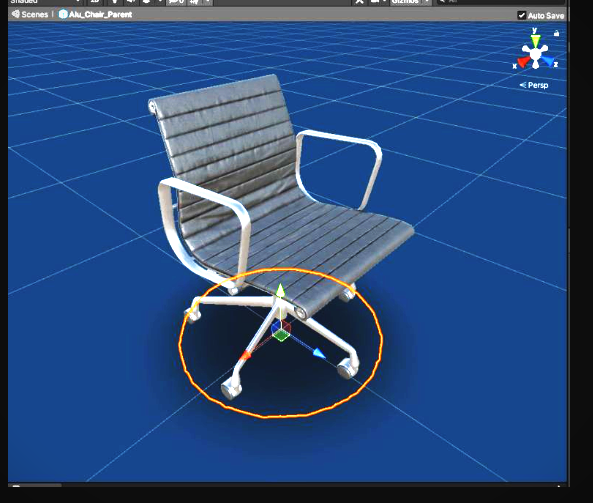
[COURSE INCHARGE : Darshan Parmar ]

*PRSENTING : TAD (AR FURNITURE CHAIR )*

**Placement Indicator** : The first thing we will do is to add is a placement indicator. We won’t be displaying the AR Planes, so we need indicators to let the user know if AR Foundation detected a surface



**Lighting and Shadows :** Creating prefabs for the furniture objects. The pivot is at the height where the object makes contact with the ground. Moreover, I have add blob shadows. They make your objects feel grounded instead of just floating on thin air. A simple quad with a transparent material can look good enough**.**

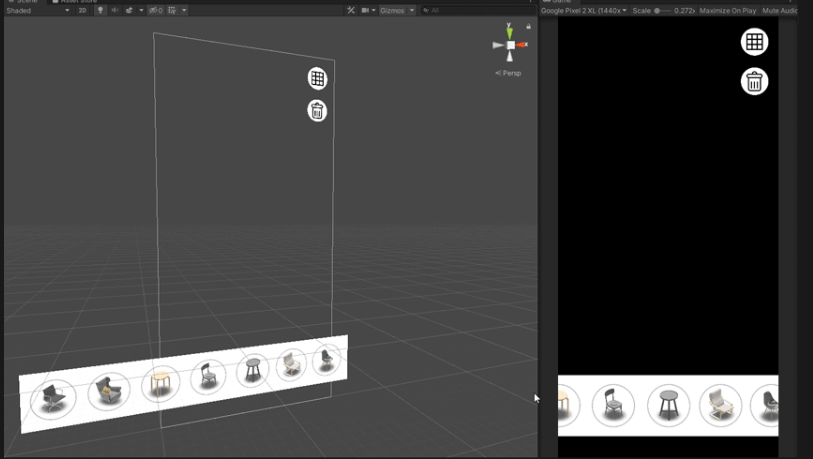
****

**UI Icons :**

One icon for each 3d model, one delete icon, and another icon to hide the grid or the placement indicator. For the furniture assets, I took a screenshot of each model in a white background**.**



I have Placed these icons into the canvas and use [Scroll Rect](https://www.youtube.com/watch?v=tcU8yzv_xEw) to create a scrolling menu containing the spawnable furniture.



**Object Scale Animation:**

I added a simple animation [Vector3.Lerp](https://docs.unity3d.com/ScriptReference/Vector3.Lerp.html) using to make spawning objects look better instead of appearing out of thin air.

**CODE :**

IEnumerator LerpObjectScale(Vector3 a, Vector3 b, float time, GameObject lerpObject)

{

float i = 0.0f;

float rate = (1.0f / time);

while (i < 1.0f)

{

i += Time.deltaTime \* rate;

lerpObject.transform.localScale = Vector3.Lerp(a, b, i);

yield return null;

}

}



**Object Selection**

I used [TouchPhase](https://docs.unity3d.com/ScriptReference/TouchPhase.html) and [Raycast](https://docs.unity3d.com/ScriptReference/Physics.Raycast.html) for object selection. When the user taps an object, it becomes the active object and is highlighted. When the user taps on an empty space, all objects are deselected.

**CODE :**

if (Input.GetTouch(0).phase == TouchPhase.Ended)

{

Ray raycast = Camera.main.ScreenPointToRay(Input.GetTouch(0).position);

RaycastHit raycastHit;

if (Physics.Raycast(raycast, out raycastHit))

{

if (raycastHit.collider.CompareTag("Object"))

{

HighlightObject();

}

}

else { DeselectObject(); }

}

else { DeselectObject(); }

**Moving Objects**

When the user moves his finger, the active object moves along with it. There are several ways to do this. The easiest way is to set the object’s position based on where the user touches on the screen

CODE :

private GameObject activeObject;

private float speedModifier = 0.0005f;

private Vector3 translationVector;

// Object movement logic within the Update function

if (Input.GetTouch(0).phase == TouchPhase.Moved && activeObject != null)

{

// Convert X-Y touch movement to object translation in world space

translationVector = new Vector3(Camera.main.transform.forward.x, 0f, Camera.main.transform.forward.z);

activeObject.transform.Translate(translationVector \* Input.GetTouch(0).deltaPosition.y \* speedModifier, Space.World);

translationVector = new Vector3(Camera.main.transform.right.x, 0f, Camera.main.transform.right.z);

activeObject.transform.Translate(translationVector \* Input.GetTouch(0).deltaPosition.x \* speedModifier, Space.World);

}

[ SUCCESSFULLY IMPLEMENTED ! ]