Assignment-1

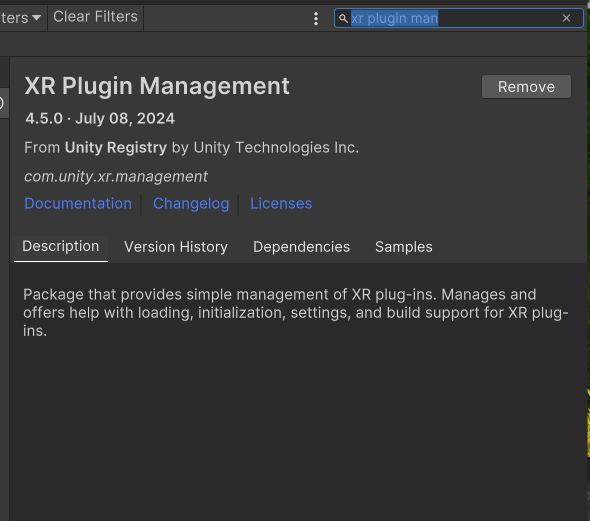
# Objective:

To create a basic virtual environment in Unity that includes a ground plane, a skybox, environmental objects, lighting, and simple VR interaction. The player should be able to grab and move the grabale objects in the environment

**Task 1:** Set Up Your Unity Project & Configure the VR Environment [5 marks]

Steps:

* Open Unity Hub and create a new project using 3D template.
* Name your project and choose the save location.
* Install XR Plugin Management
* Go to Edit > Project Settings > XR Plugin Management.
* Select the platform (e.g., Windows, Android) and install OpenXR for cross-platform VR support.
* Import XR Interaction Toolkit:
* Go to Window > Package Manager.
* Search for XR Interaction Toolkit and install it.
* Right-click in the Hierarchy and choose XR > Device-based > XR Rig.



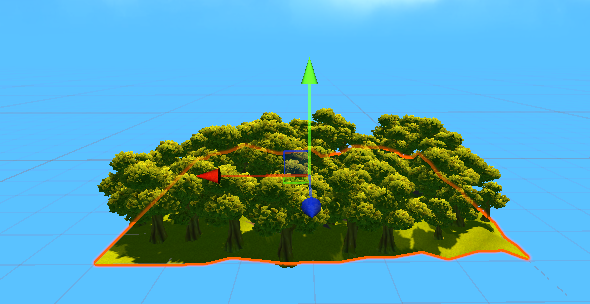
**Task 2:** Create the Ground Plane [5 marks]

* + Create a larger ground area so player can move around
  + You can use Terrain object for that

Steps:

* Right-click in the Hierarchy and select 3D Object > Terrain.
* Click on the Terrain object and go to the Inspector panel.
* Under Terrain Settings, adjust the width and length to make it larger (e.g., 50 x 50).
* Select the Paint Terrain tool in the Inspector.
* Choose Set Height and flatten the terrain to 0 or desired height.
* Select Terrain > Paint Texture and assign a ground texture (e.g., grass, dirt).

Terrain Image :

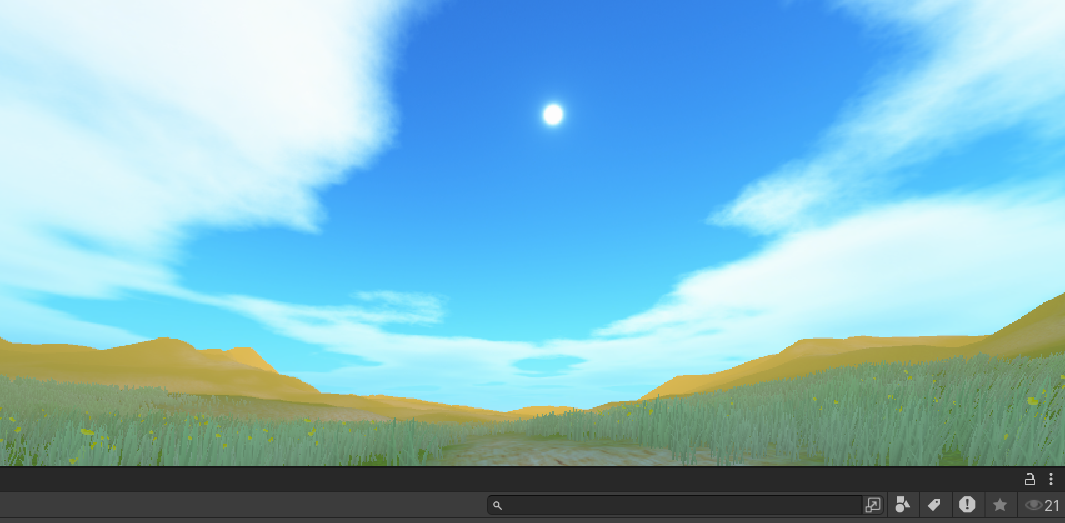


**Task 3:** Add a Skybox [5 marks]

* + You can use sky presets
  + Try to make it more detailed and more interesting

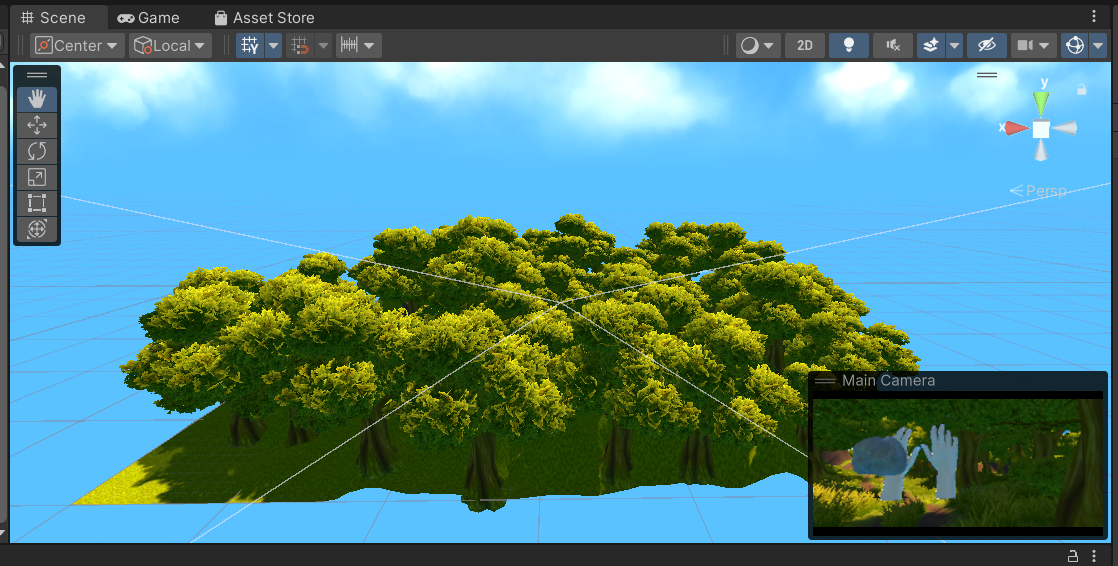
Steps:

* Go to Window > Rendering > Lighting Settings.
* In the Skybox Material field, click the circle and choose a skybox preset or import one from the Unity Asset Store.
* Customize the Skybox:
* Test the Skybox by pressing Play to see how it looks in VR.



**Task 4:** Add Environment Objects [15 marks]

* + You can use Assets to create an engaging Environment
  + Create a grabbable objects spawning at random locations
* Go to Assets > Import Package > Custom Package.
* Import any environmental objects (e.g., trees, rocks) from the Unity Asset Store or external sources.
* Right-click in the Hierarchy and choose 3D Object to place objects such as trees, houses, or other assets.
* Position them around the terrain for a natural environment.
* Import a 3D object (e.g., cube, ball).
* Right-click in the Hierarchy, select Create Empty, and name it "GrabbableObject."
* Drag your object under this Empty GameObject.
* Position the grabbable objects at various points on the terrain.
* In the Inspector, manually adjust the Transform (position) values for each object to randomize the location.



**Task 6**: Configure Lighting and Shadows [5 marks]

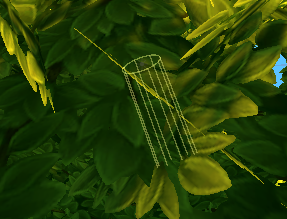
Steps:

Go to GameObject > Light > Directional Light.

Adjust its rotation to simulate sunlight.

In the Inspector, find the Shadow Type setting for the Directional Light and set it to Soft Shadows.

In the Lighting Settings, adjust Ambient Light and Reflection Intensity to enhance realism.



**Task 7:** Add Audio [5 marks]

Steps:

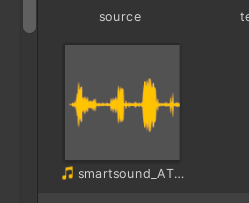
Go to Assets > Import New Asset and select an audio file.

Right-click in the Hierarchy and select Audio > Audio Source.

In the Inspector, assign your audio file to the AudioClip field.

Drag the Audio Source to a relevant location in the environment (e.g., a waterfall, forest area).

Test the Audio by playing the scene and moving closer to the audio source.

**Task 8:** Implement Basic VR Interaction [25 marks]

* + Create a Grabbable Object
  + Add Grabbable and Grabber Components

Steps:

* Create a Grabbable Object:
  + Select one of the grabbable objects you created earlier (Task 4).
* Add Components:
  + In the Inspector, add the component XR Grab Interactable from the XR Interaction Toolkit.
* Add Grabber:
  + On the XR Rig, ensure you have LeftHand Controller and RightHand Controller.
  + Add the XR Direct Interactor component to both hand controllers.
* Test Grabbing:
  + Press Play and test grabbing the objects in VR by moving your hands toward them.



**Task 9:** Write the VR Interaction Script [25 marks]

Steps:

* Right-click in the Project panel and select Create > C# Script.
* Name the script GrabObjectScript.
* Write Interaction Logic:
* In the script, write logic to handle object grabbing and releasing.
* Use OnSelectEnter and OnSelectExit events from the XR Interaction Toolkit to manage the grabbing behavior.
* Attach Script:
* Attach the script to the GrabbableObject from Task 4.
* Test the Script:
* Press Play and test the grab/release interaction with the VR controllers.

C# Script :

using UnityEngine;

using UnityEngine.XR.Interaction.Toolkit;

public class GrabObjectScript : MonoBehaviour

{

private XRGrabInteractable grabInteractable;

void Awake()

{

// Initialize XRGrabInteractable component

grabInteractable = GetComponent<XRGrabInteractable>();

}

void OnEnable()

{

// Subscribe to events

grabInteractable.selectEntered.AddListener(OnSelectEnter);

grabInteractable.selectExited.AddListener(OnSelectExit);

}

void OnDisable()

{

// Unsubscribe from events to prevent memory leaks

grabInteractable.selectEntered.RemoveListener(OnSelectEnter);

grabInteractable.selectExited.RemoveListener(OnSelectExit);

}

private void OnSelectEnter(SelectEnterEventArgs args)

{

// Logic for grabbing the object

Debug.Log("Object Grabbed");

// Additional logic can be added here for when the object is grabbed

}

private void OnSelectExit(SelectExitEventArgs args)

{

// Logic for releasing the object

Debug.Log("Object Released");

// Additional logic can be added here for when the object is released

}

}

**Task 10:** Demo application [5 marks]

Steps:

* Ensure that project runs smoothly by optimizing lighting, reducing unnecessary objects, and lowering the texture size.
* Build Settings:
* Go to File > Build Settings.
* Select the platform (Windows, Android, etc.) and set the project to VR Supported.
* Run the Application:

