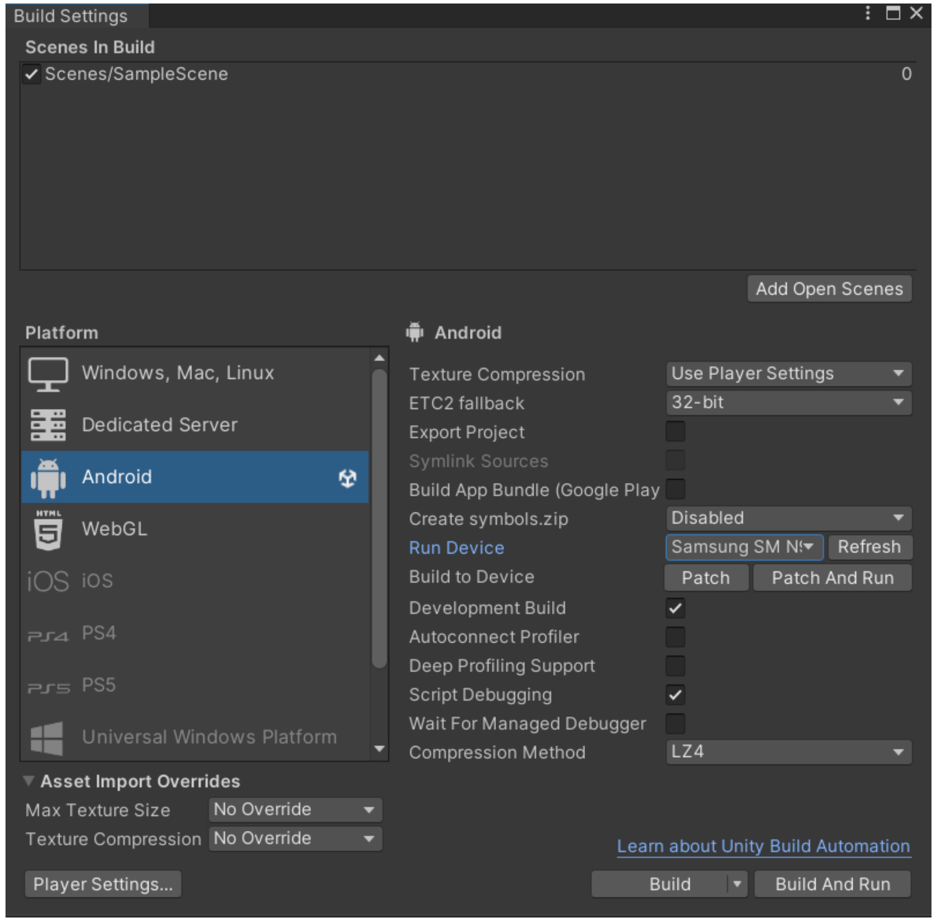
INDUSTRIAL UNIVERSITY OF HOCHIMINH CITY

NEW TECHNOLOGY IN IT APPLICATION DEVELOPMENT

Mobile in Unity

Setup for Android



Build Settings

Scenes In Build

✓ Scenes/SampleScene 0

Add Open Scenes

Platform

- Windows, Mac, Linux
- Dedicated Server
- Android
- WebGL
- iOS
- PS4
- PS5
- Universal Windows Platform

Android

Texture Compression	Use Player Settings
ETC2 fallback	32-bit
Export Project	
Symlink Sources	
Build App Bundle (Google Play	
Create symbols.zip	Disabled
Run Device	Samsung SM N(Refresh
Build to Device	Patch / Patch And Run
Development Build	✓
Autoconnect Profiler	
Deep Profiling Support	
Script Debugging	✓
Wait For Managed Debugger	
Compression Method	LZ4

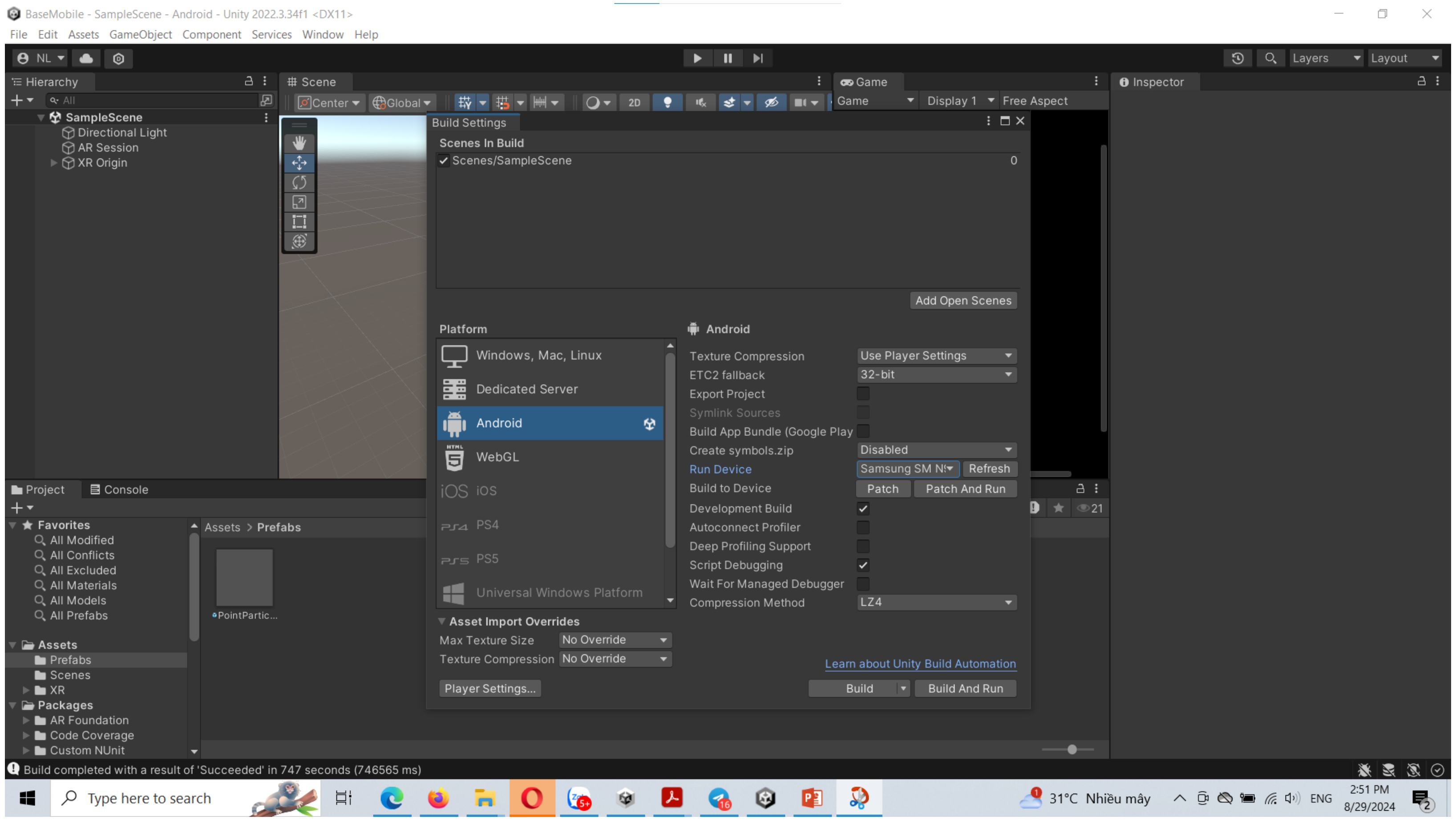
Asset Import Overrides

Max Texture Size	No Override
Texture Compression	No Override

Learn about Unity Build Automation

Player Settings... Build Build And Run

Setup for Android


BaseMobile - SampleScene - Android - Unity 2022.3.34f1 <DX11>

Build Settings

Scenes In Build

✓ Scenes/SampleScene 0

Add Open Scenes

Platform: Windows, Mac, Linux; Dedicated Server; Android (selected); WebGL; iOS; PS4; PS5; Universal Windows Platform

Android

Texture Compression	Use Player Settings
ETC2 fallback	32-bit
Export Project	☐
Symlink Sources	☐
Build App Bundle (Google Play	☐
Create symbols.zip	Disabled
Run Device	Samsung SM N
Build to Device	Patch / Patch And Run
Development Build	✓
Autoconnect Profiler	☐
Deep Profiling Support	☐
Script Debugging	✓
Wait For Managed Debugger	☐
Compression Method	LZ4

Asset Import Overrides

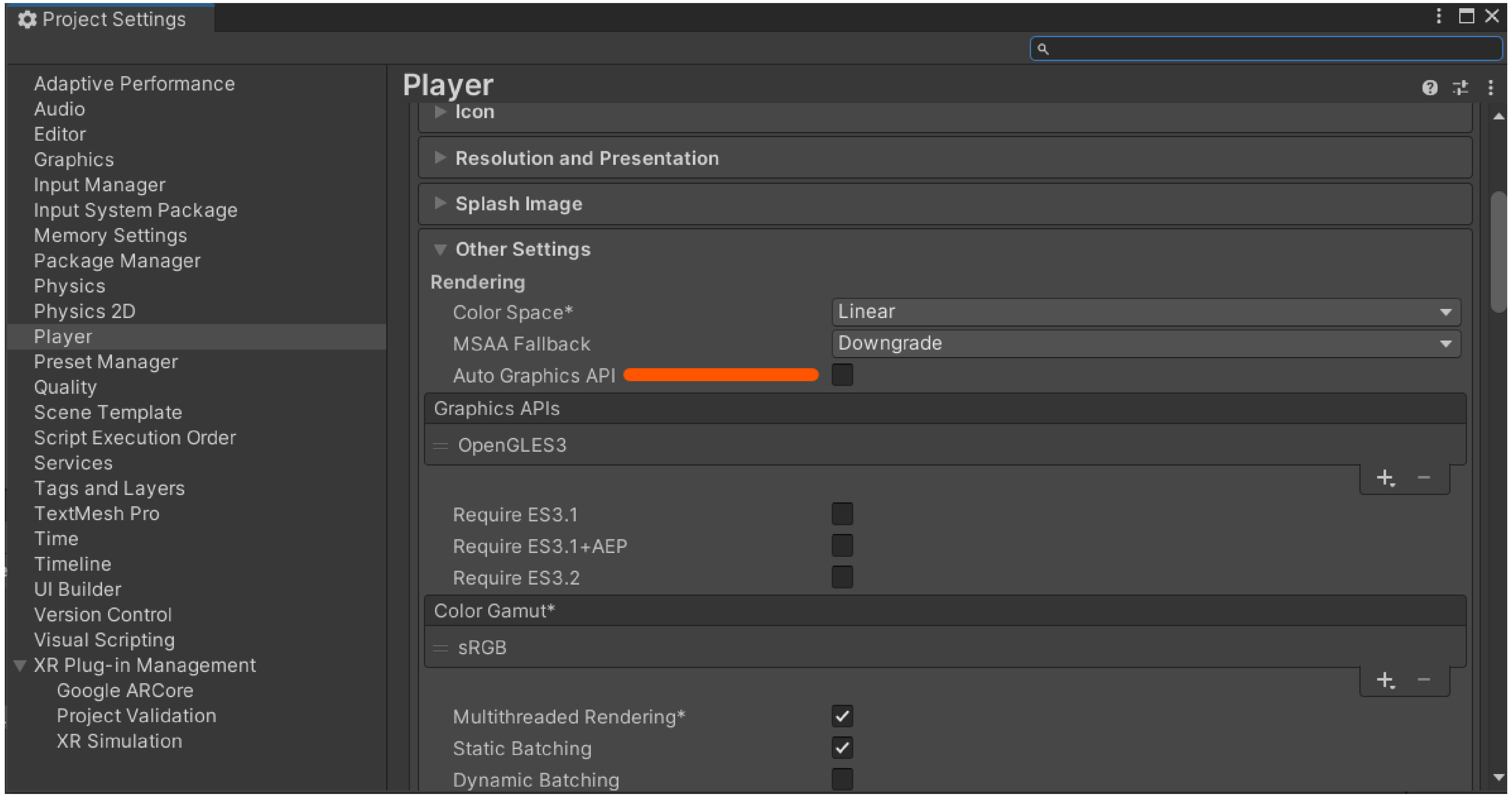
Max Texture Size	No Override
Texture Compression	No Override

Learn about Unity Build Automation

Player Settings... Build Build And Run

Build completed with a result of 'Succeeded' in 747 seconds (746565 ms)

Configuration


Project Settings

Adaptive Performance
Audio
Editor
Graphics
Input Manager
Input System Package
Memory Settings
Package Manager
Physics
Physics 2D
Player
Preset Manager
Quality
Scene Template
Script Execution Order
Services
Tags and Layers
TextMesh Pro
Time
Timeline
UI Builder
Version Control
Visual Scripting
XR Plug-in Management
Google ARCore
Project Validation
XR Simulation

Player

Icon

Resolution and Presentation

Splash Image

Other Settings

Rendering

Color Space*	Linear
MSAA Fallback	Downgrade
Auto Graphics API	☐

Graphics APIs
OpenGLES3

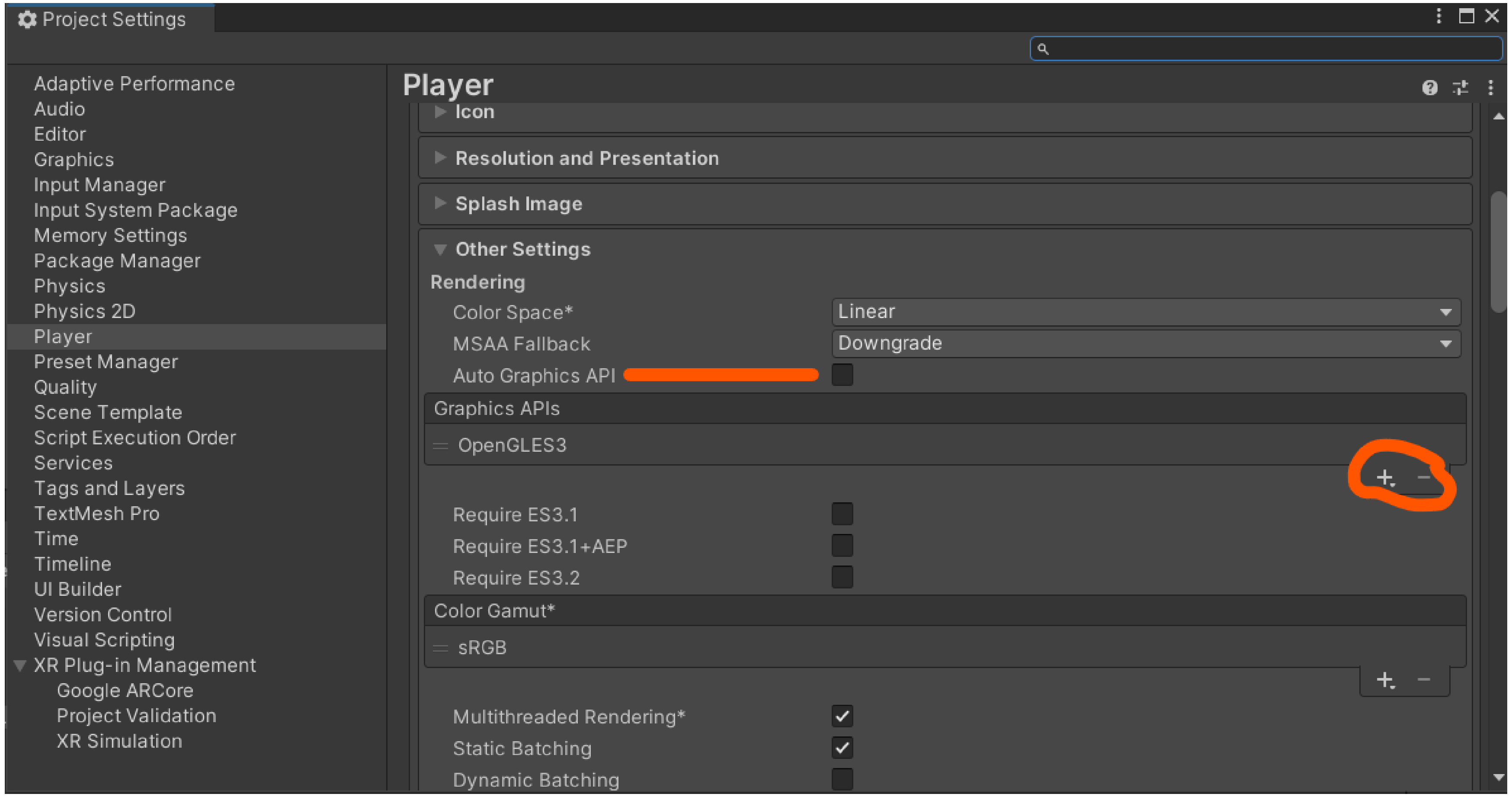
Require ES3.1	☐
Require ES3.1+AEP	☐
Require ES3.2	☐

Color Gamut*
sRGB

Multithreaded Rendering*	✓
Static Batching	✓
Dynamic Batching	☐

Configuration

Remove Vukan


Project Settings

- Adaptive Performance
- Audio
- Editor
- Graphics
- Input Manager
- Input System Package
- Memory Settings
- Package Manager
- Physics
- Physics 2D
- Player
- Preset Manager
- Quality
- Scene Template
- Script Execution Order
- Services
- Tags and Layers
- TextMesh Pro
- Time
- Timeline
- UI Builder
- Version Control
- Visual Scripting
- XR Plug-in Management
  - Google ARCore
  - Project Validation
  - XR Simulation

Player

- Icon
- Resolution and Presentation
- Splash Image
- Other Settings

Rendering

Color Space*	Linear
MSAA Fallback	Downgrade
Auto Graphics API	☐

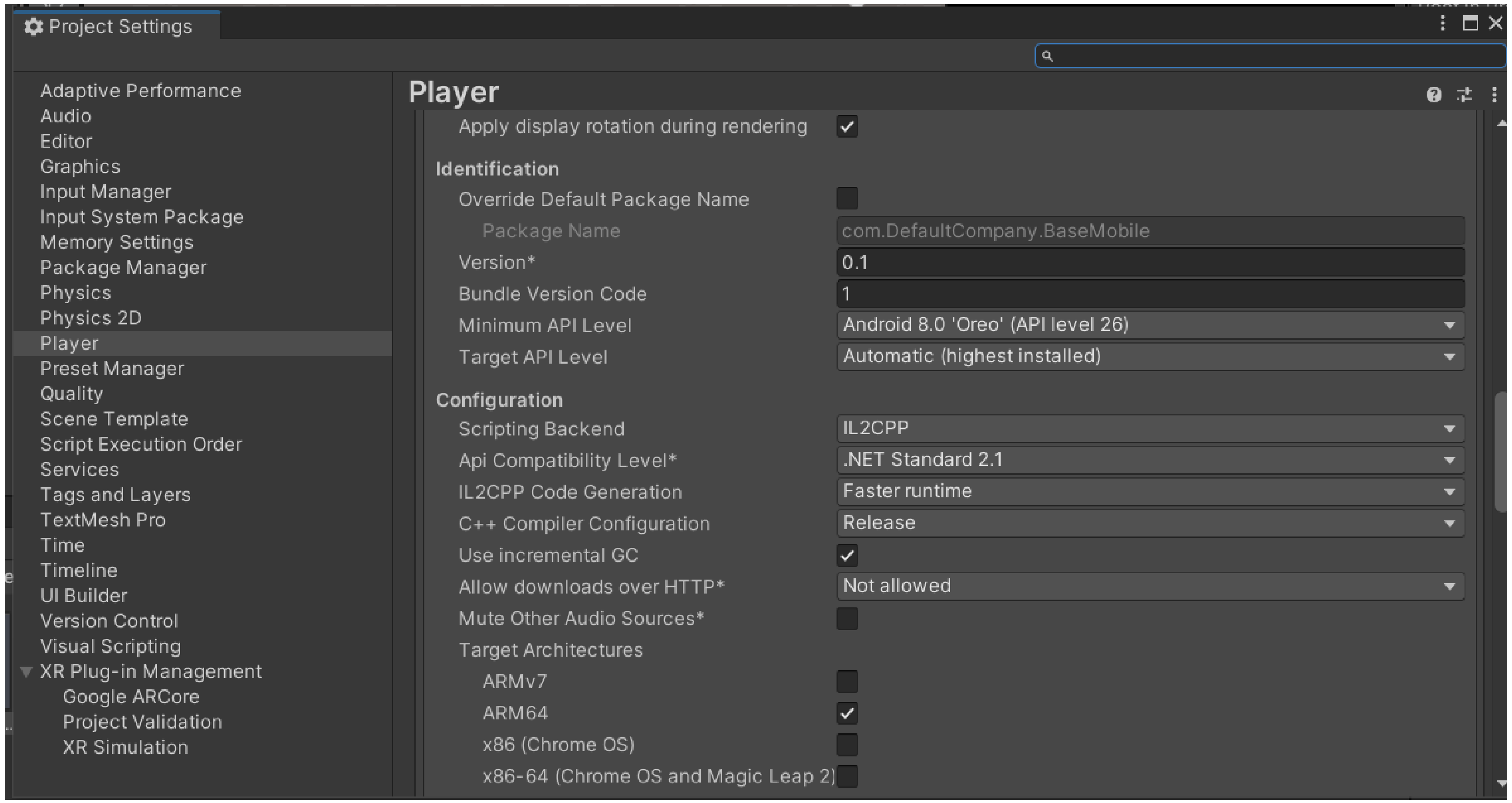
Graphics APIs
- OpenGLES3

Require ES3.1	☐
Require ES3.1+AEP	☐
Require ES3.2	☐

Color Gamut*
- sRGB

Multithreaded Rendering*	✓
Static Batching	✓
Dynamic Batching	☐

Configuration

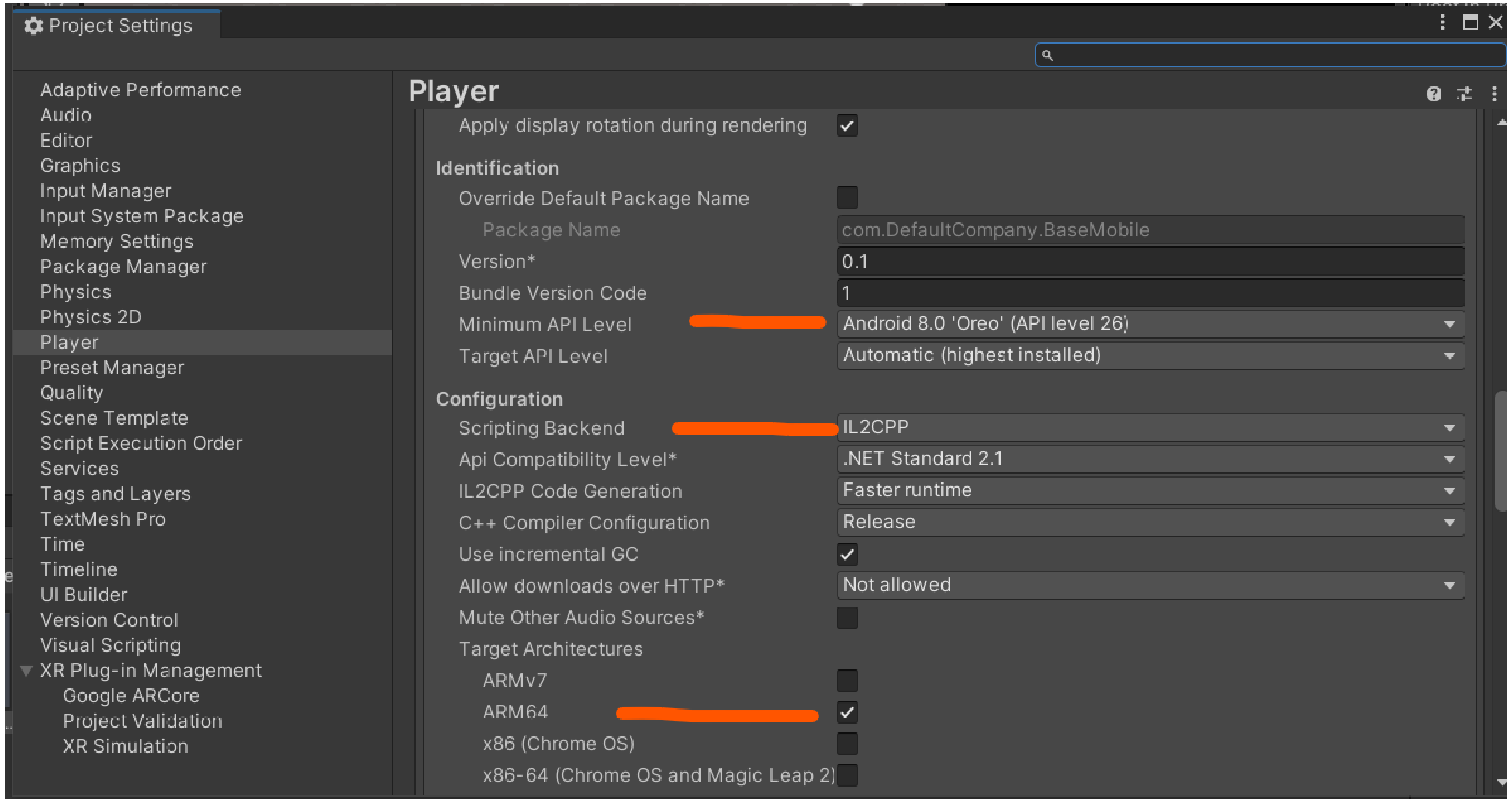

Project Settings

- Adaptive Performance
- Audio
- Editor
- Graphics
- Input Manager
- Input System Package
- Memory Settings
- Package Manager
- Physics
- Physics 2D
- Player
- Preset Manager
- Quality
- Scene Template
- Script Execution Order
- Services
- Tags and Layers
- TextMesh Pro
- Time
- Timeline
- UI Builder
- Version Control
- Visual Scripting
- XR Plug-in Management
  - Google ARCore
  - Project Validation
  - XR Simulation

Player

Apply display rotation during rendering	✓
Identification	
Override Default Package Name	☐
Package Name	com.DefaultCompany.BaseMobile
Version*	0.1
Bundle Version Code	1
Minimum API Level	Android 8.0 'Oreo' (API level 26)
Target API Level	Automatic (highest installed)
Configuration	
Scripting Backend	IL2CPP
Api Compatibility Level*	.NET Standard 2.1
IL2CPP Code Generation	Faster runtime
C++ Compiler Configuration	Release
Use incremental GC	✓
Allow downloads over HTTP*	Not allowed
Mute Other Audio Sources*	☐
Target Architectures	
ARMv7	☐
ARM64	✓
x86 (Chrome OS)	☐
x86-64 (Chrome OS and Magic Leap 2)	☐

Configuration



Player

Apply display rotation during rendering ✓

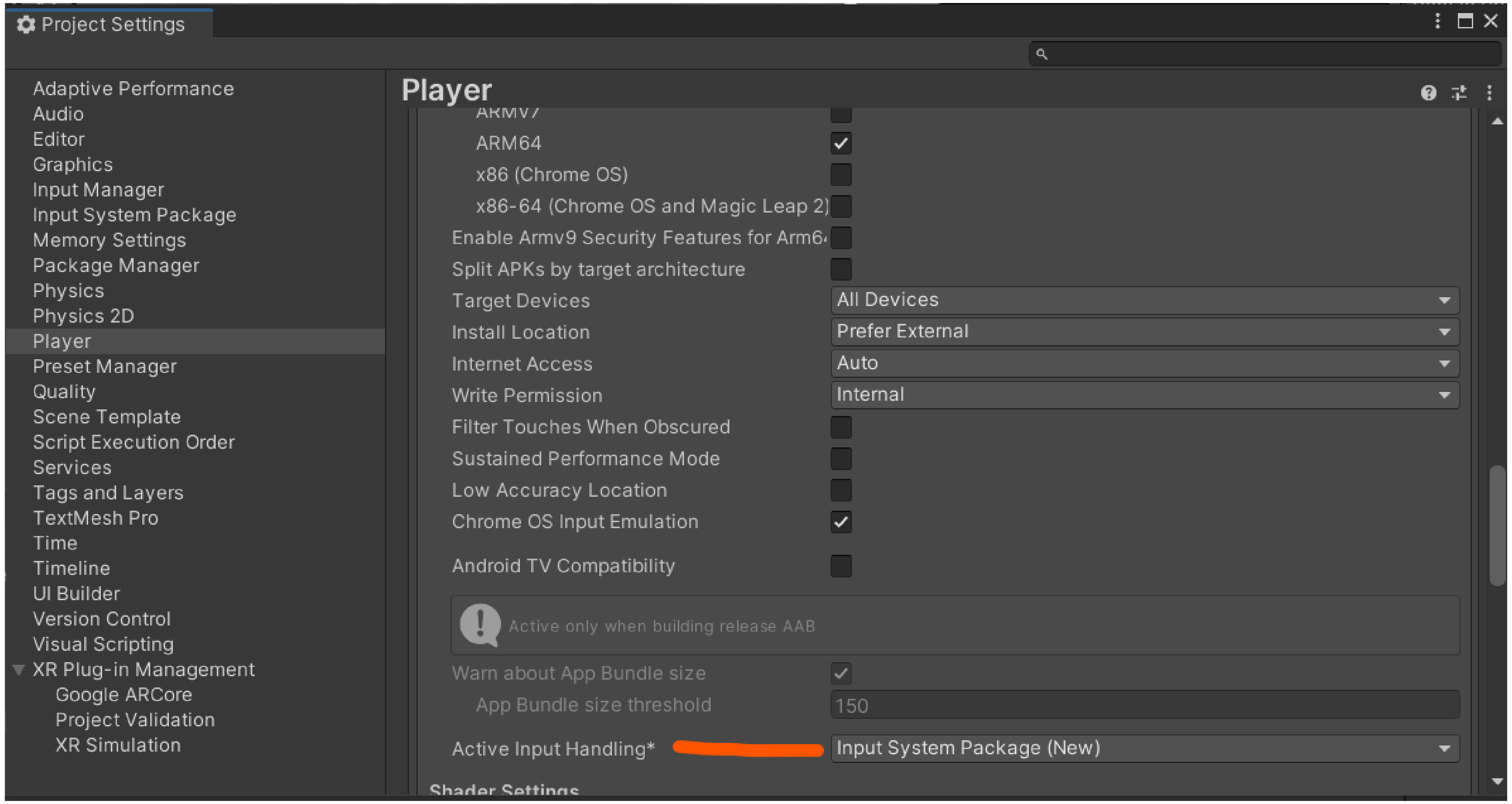
Identification

Override Default Package Name	
Package Name	com.DefaultCompany.BaseMobile
Version*	0.1
Bundle Version Code	1
Minimum API Level	Android 8.0 'Oreo' (API level 26)
Target API Level	Automatic (highest installed)

Configuration

Scripting Backend	IL2CPP
Api Compatibility Level*	.NET Standard 2.1
IL2CPP Code Generation	Faster runtime
C++ Compiler Configuration	Release
Use incremental GC	✓
Allow downloads over HTTP*	Not allowed
Mute Other Audio Sources*	
Target Architectures	
ARMv7	
ARM64	✓
x86 (Chrome OS)	
x86-64 (Chrome OS and Magic Leap 2)	

Configuration


Project Settings

Adaptive Performance
Audio
Editor
Graphics
Input Manager
Input System Package
Memory Settings
Package Manager
Physics
Physics 2D
Player
Preset Manager
Quality
Scene Template
Script Execution Order
Services
Tags and Layers
TextMesh Pro
Time
Timeline
UI Builder
Version Control
Visual Scripting
XR Plug-in Management
Google ARCore
Project Validation
XR Simulation

Player

ARMV7	
ARM64	✓
x86 (Chrome OS)	
x86-64 (Chrome OS and Magic Leap 2)	
Enable Armv9 Security Features for Arm6	
Split APKs by target architecture	
Target Devices	All Devices
Install Location	Prefer External
Internet Access	Auto
Write Permission	Internal
Filter Touches When Obscured	
Sustained Performance Mode	
Low Accuracy Location	
Chrome OS Input Emulation	✓
Android TV Compatibility	

Active only when building release AAB

Warn about App Bundle size	✓
App Bundle size threshold	150
Active Input Handling*	Input System Package (New)

Shader Settings

Building and running Point cloud

1. Create a new scene named PointCloud3D
2. In the **Hierarchy** window, delete the default **Main Camera** (*right-click* and select **Delete**, or use the *Del* keyboard key).
3. Add an AR Session object by selecting **GameObject | XR | AR Session**.
4. Add an AR Session Origin object by selecting **GameObject | XR | AR Session Origin**
5. Add a point cloud manager to the Session Origin object by clicking **Add Component** in the **Inspector** window. Then, enter ar point in the search field and select **AR Point Cloud Manager**.

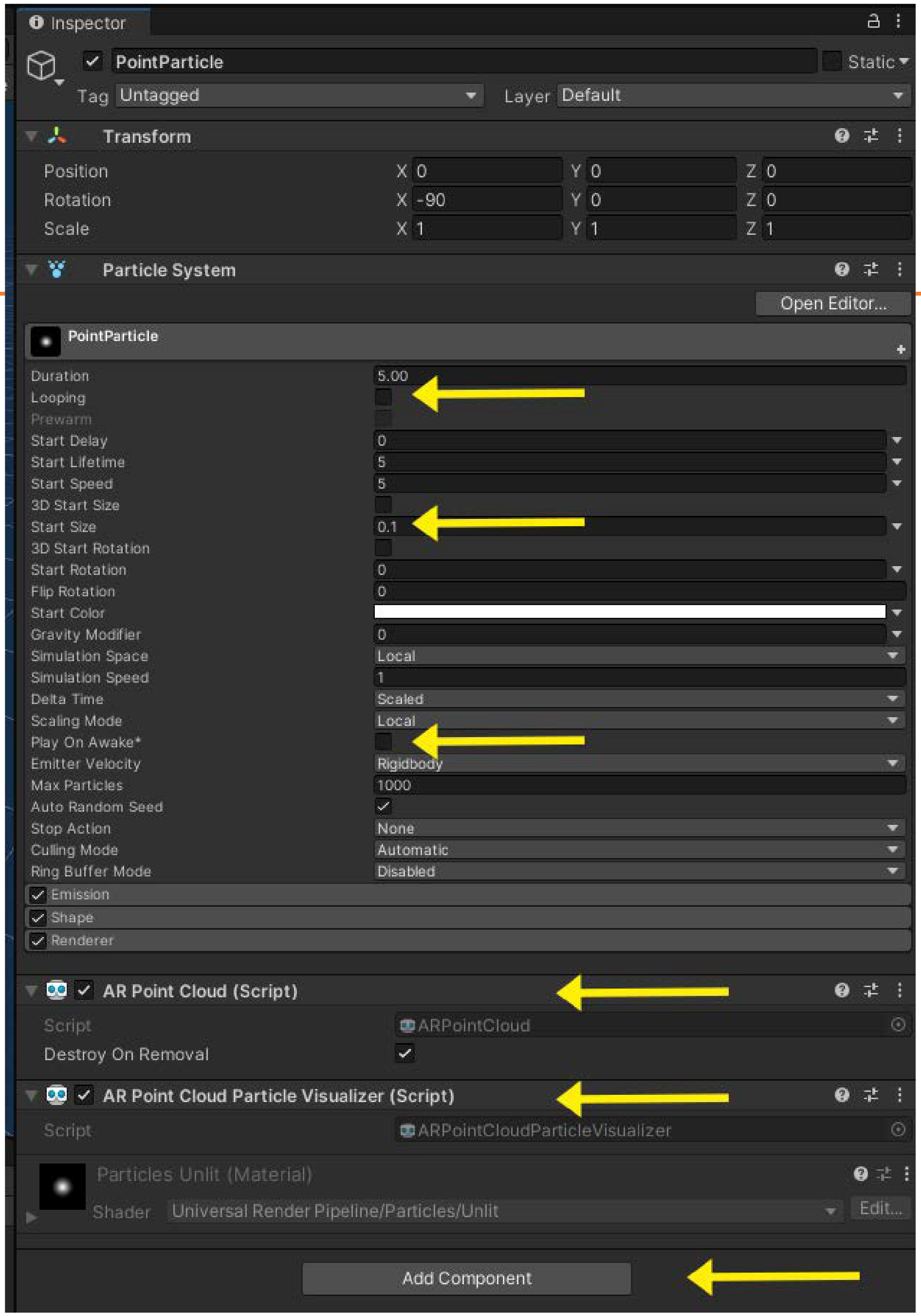
Create Particle System

1. Create a Particle System by selecting **GameObject | Effects | Particle System**.
2. In the **Inspector** window, rename it PointParticle.
3. On the **Particle System** component, uncheck the **Looping** checkbox.
4. Set its **Start Size** to 0.1.
5. Uncheck the **Play on Awake** checkbox.
6. Click **Add Component**, enter ar point in the search field, and select **AR**

Point Cloud.

1. Likewise, click **Add Component** and select **AR Point Cloud Visualizer**.
2. Drag the **PointParticle** object from the **Hierarchy** window to the **Prefabs** folder
3. in the **Project** window (create the folder first if necessary). This makes the
4. GameObject into a prefab.
5. Delete the **PointParticle** object from the **Hierarchy** window using *right-click* **|**
6. **Delete** or press the *Del* key.

INDUSTRIAL UNIVERSITY OF HOCHIMINH CITY



Inspector

PointParticle — Static

Tag: Untagged — Layer: Default

Transform

	X	Y	Z
Position	0	0	0
Rotation	-90	0	0
Scale	1	1	1

Particle System — Open Editor...

PointParticle

Duration	5.00
Looping	☐
Prewarm	☐
Start Delay	0
Start Lifetime	5
Start Speed	5
3D Start Size	☐
Start Size	0.1
3D Start Rotation	☐
Start Rotation	0
Flip Rotation	0
Start Color	⬜
Gravity Modifier	0
Simulation Space	Local
Simulation Speed	1
Delta Time	Scaled
Scaling Mode	Local
Play On Awake*	☐
Emitter Velocity	Rigidbody
Max Particles	1000
Auto Random Seed	✓
Stop Action	None
Culling Mode	Automatic
Ring Buffer Mode	Disabled

✓ Emission

✓ Shape

✓ Renderer

✓ AR Point Cloud (Script)

Script	ARPointCloud
Destroy On Removal	✓

✓ AR Point Cloud Particle Visualizer (Script)

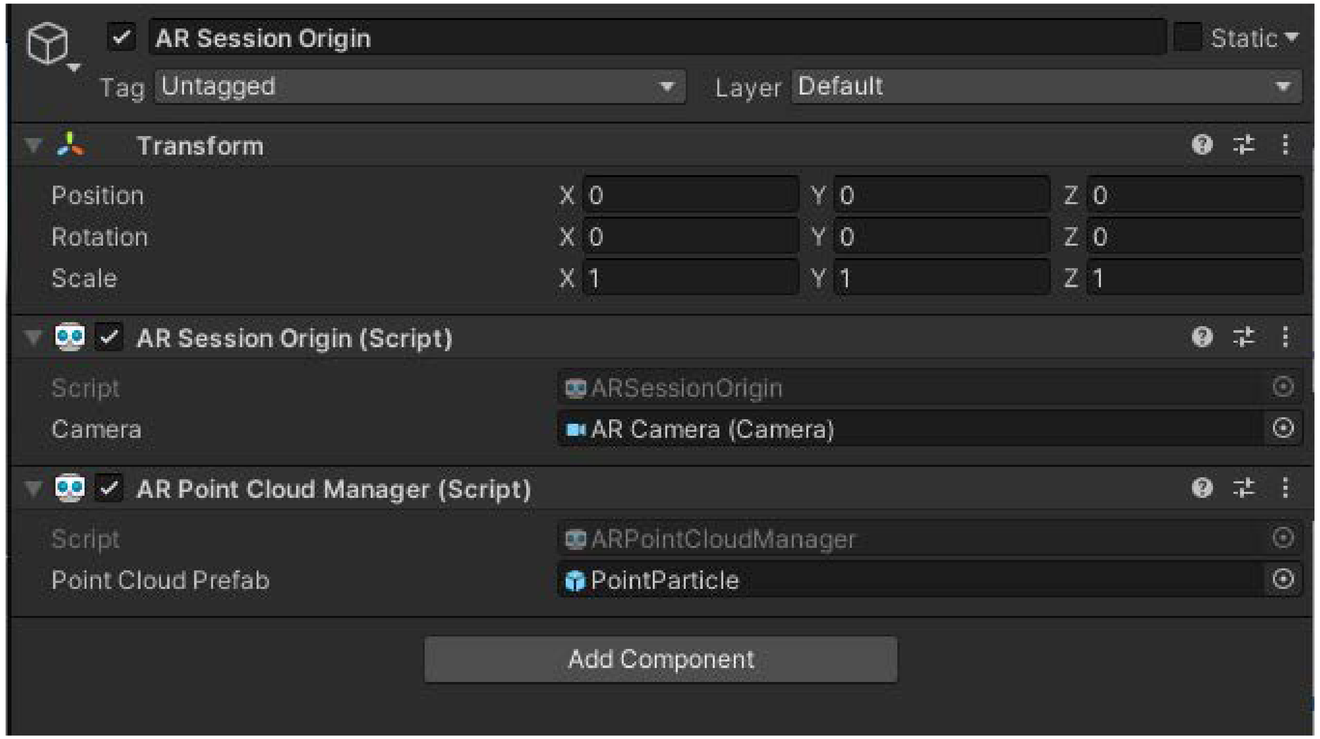
Script	ARPointCloudParticleVisualizer

Particles Unlit (Material)

Shader: Universal Render Pipeline/Particles/Unlit — Edit...

Add Component

The resulting AR Session Origin



✓ AR Session Origin Static

Tag Untagged Layer Default

Transform

Position	X 0	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

✓ AR Session Origin (Script)

Script	ARSessionOrigin
Camera	AR Camera (Camera)

✓ AR Point Cloud Manager (Script)

Script	ARPointCloudManager
Point Cloud Prefab	PointParticle

Plane Detection

- Add components
  - XR Origin > AR Plane Manager
- AR Plane Manager > Plane Prefab > ARPlane

Tap to place Objects

- Create your own plane prefab

1. Create empty object named ARPlane
2. Add components:
   - AR plane
   - AR Plane Mesh Visualizer
   - Mesh Collider
   - Mesh Filter
   - Mesh Renderer
   - Line Renderer

- Mesh Renderer > Materials > Visualizer

Line Renderer

Corner Vertices	4
End Cap Vertices	4
Use World Space	Uncheck
Cast Shadows	Off
Receive Shadows	Uncheck
Element 0	Default-Line

Tap to place Objects

```
using System.Collections;
using System.Collections.Generic;
using Unity.VisualScripting;
using UnityEngine;

using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;
using EnhancedTouch = UnityEngine.InputSystem.EnhancedTouch;
```

Tap to place Objects

```
[RequireComponent(requiredComponent:typeof(ARRaycastManager),requiredComponent2:
typeof(ARPlaneManager))]
public class PlaceObject : MonoBehaviour
{
    [SerializeField]
    private GameObject prefab;
    private ARRaycastManager aRRcM;
    private ARPlaneManager aRPM;
    private List<ARRaycastHit> hits = new List<ARRaycastHit>();
    void Awake()
    {
        aRRcM = GetComponent<ARRaycastManager>();
        aRPM = GetComponent<ARPlaneManager>();
    }
. . .
```

```
private void OnEnable()
{
  EnhancedTouch.TouchSimulation.Enable();
  EnhancedTouch.EnhancedTouchSupport.Enable();
  EnhancedTouch.Touch.onFingerDown +=
FingerDown;
}
```

```
private void OnDisable()
{
  EnhancedTouch.TouchSimulation.Disable();
  EnhancedTouch.EnhancedTouchSupport.Disable();
  EnhancedTouch.Touch.onFingerDown -=
FingerDown;
}
```

```
private void FingerDown(EnhancedTouch.Finger finger)
{
   if(finger.index != 0) return; // multi-tourh fingers down == 1
   if(aRRcM.Raycast(finger.currentTouch.screenPosition,hits,trackableTypes:TrackableType.PlaneW
ithinPolygon)){
   foreach(ARRaycastHit hit in hits){
      Pose poseH = hit.pose; // position and orientation
      GameObject obj =
Instantiate(original:prefab,position:poseH.position,rotation:poseH.rotation);
            }
         }
     }
```

Place and Move Object

- XR Origin > AR Raycast Manager
- XR Origin > your Script

Place and Move Object

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;

[RequireComponent(typeof(ARRaycastManager))]
public class ARTabToPlaceObject : MonoBehaviour
{
    public GameObject goInstaintiate;
    private GameObject spawnedObj;
    private ARRaycastManager aRRCM;
    private Vector2 touchPos;
    static List<ARRaycastHit> hits = new List<ARRaycastHit>();
```

```
void Awake()
 {
    aRRCM = GetComponent<ARRaycastManager>();
 }
```

```
bool TryGetTouchPosition(out Vector2 touchPosition)
{
    if(Input.touchCount>0){
       touchPosition = Input.GetTouch(index:0).position;
       return true;
    }
    touchPosition = default;
    return false;
}
```

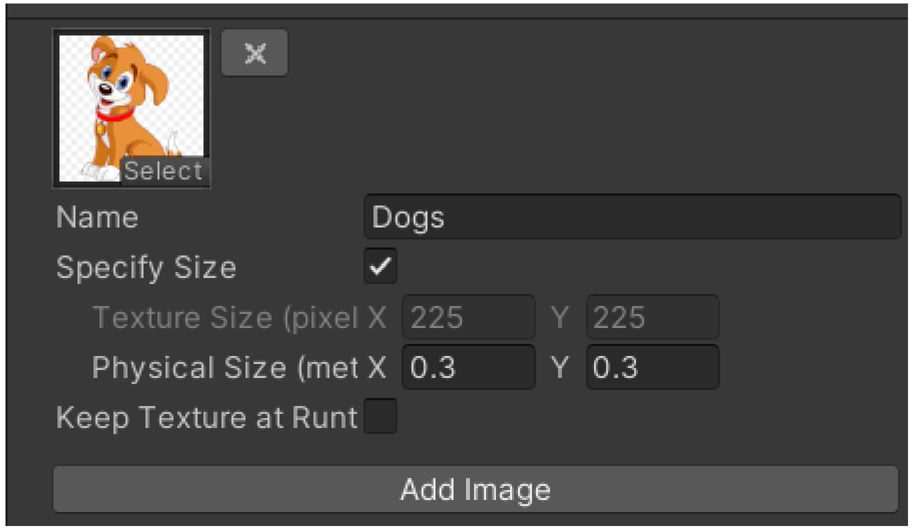
Place and Move Object

```
void Update()
{
    if(!TryGetComponent(out touchPos)){
            return;
    }
    if(aRRCM.Raycast(touchPos,hits,trackableTypes:TrackableType.PlaneWithinPolygon
)){
       Pose hitPose = hits[0].pose;
       if(spawnedObj == null){
          spawnedObj =
Instantiate(goInstaintiate,hitPose.position,hitPose.rotation);
       }
       else
            spawnedObj.transform.position = hitPose.position;

    }
}
```

Tracking Image

- ReferenceImageLibrary



Name: Dogs
Specify Size: ✓
Texture Size (pixel X 225 Y 225
Physical Size (met X 0.3 Y 0.3
Keep Texture at Runt

Tracking Image

```
using UnityEngine;
using UnityEngine.XR;
using UnityEngine.XR.ARFoundation;
// using UnityEngine.XR.ARSubsystems;

public class ImageRecognize : MonoBehaviour
{
    private ARTrackedImageManager aRTIManager;
    void Awake()
    {
        aRTIManager = FindObjectOfType<ARTrackedImageManager>();
    }
```

Tracking Image

```
void OnEnable()
    {
        aRTIManager.trackedImagesChanged += OnImageChanged;
    }

    void OnDisable()
    {
        aRTIManager.trackedImagesChanged -= OnImageChanged;
    }

    public void OnImageChanged(ARTrackedImagesChangedEventArgs args)
    {
        foreach(var trackedImage in args.added){
                Debug.Log(trackedImage.name);
        }
    }

}
```