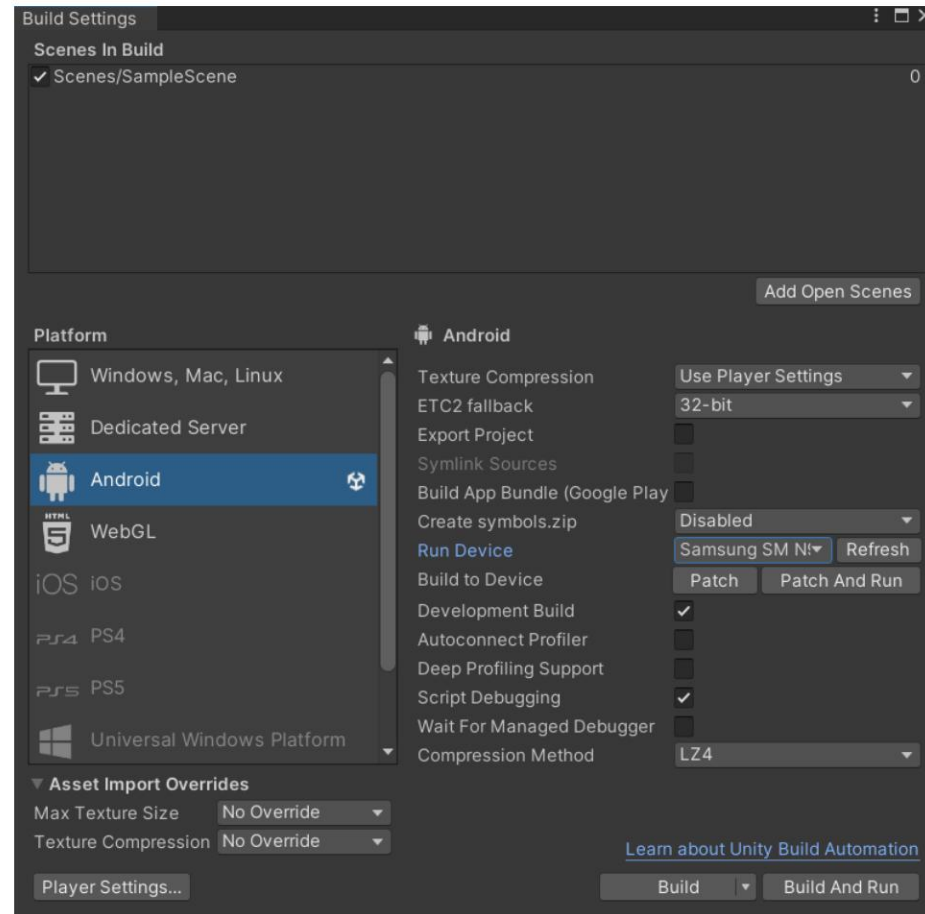


CÔNG NGHỆ MỚI TRONG CNTT

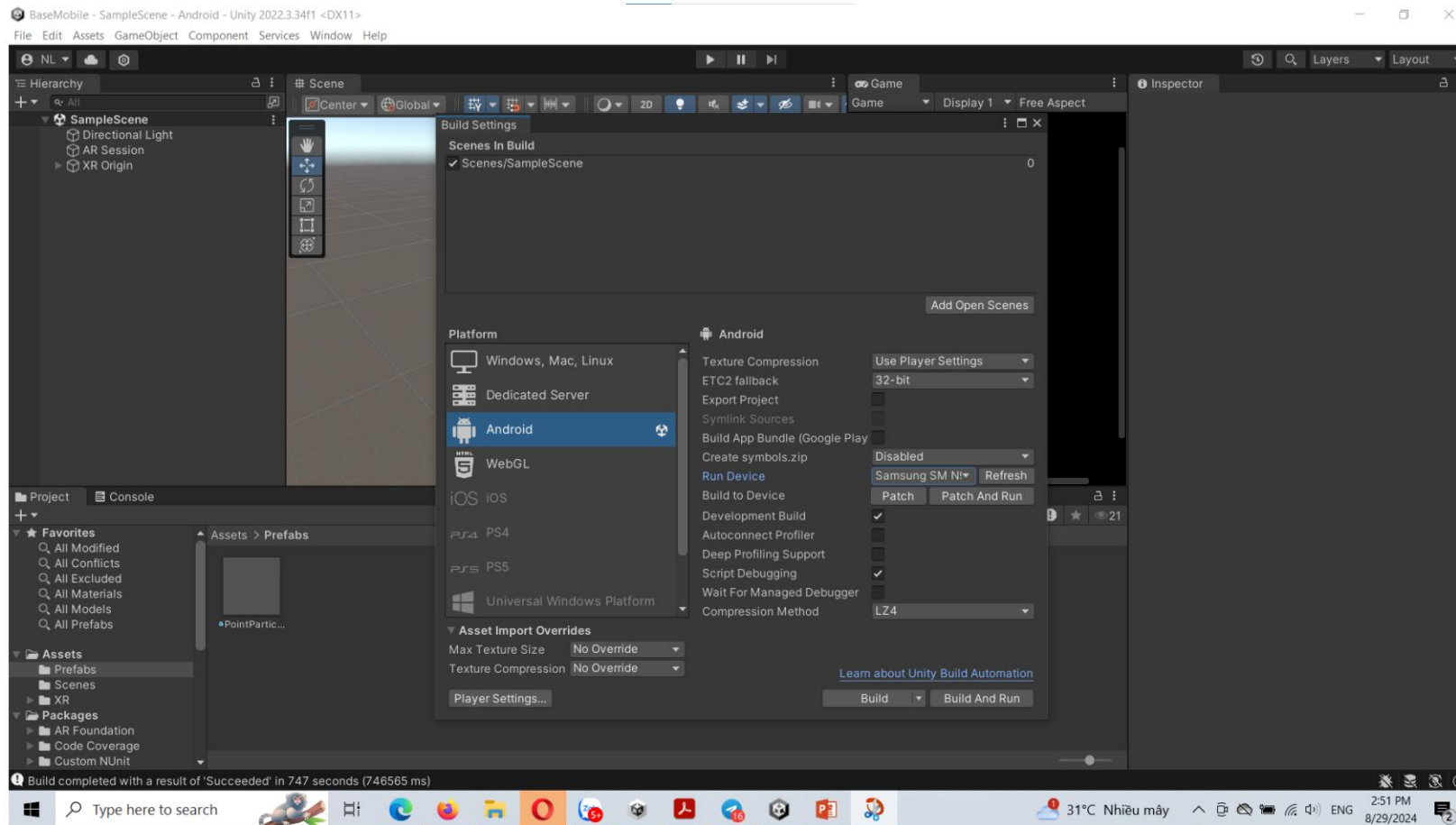
PHÁT TRIỂN ỨNG DỤNG

Di động trong Unity

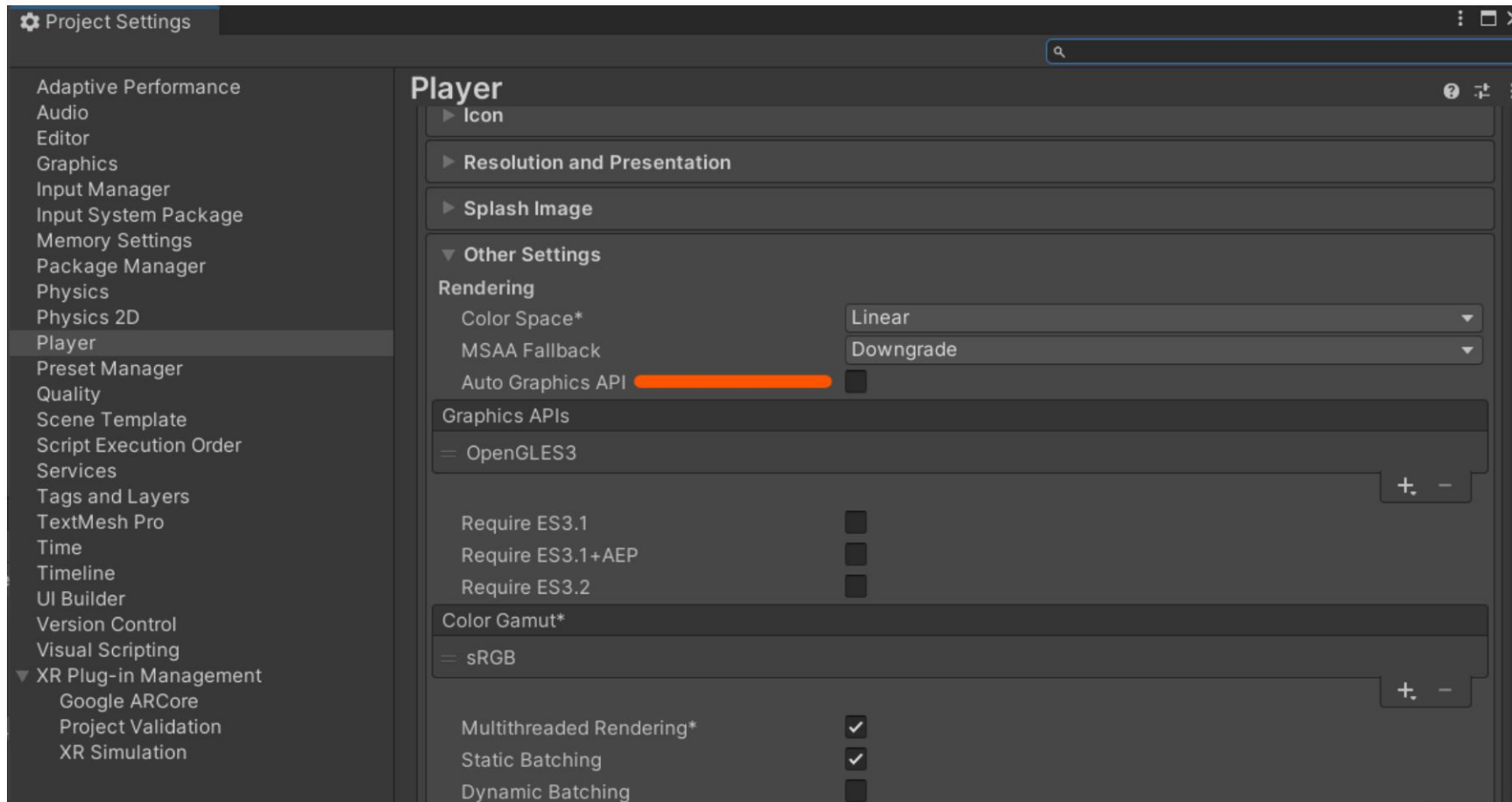
Thiết lập cho Android



Thiết lập cho Android

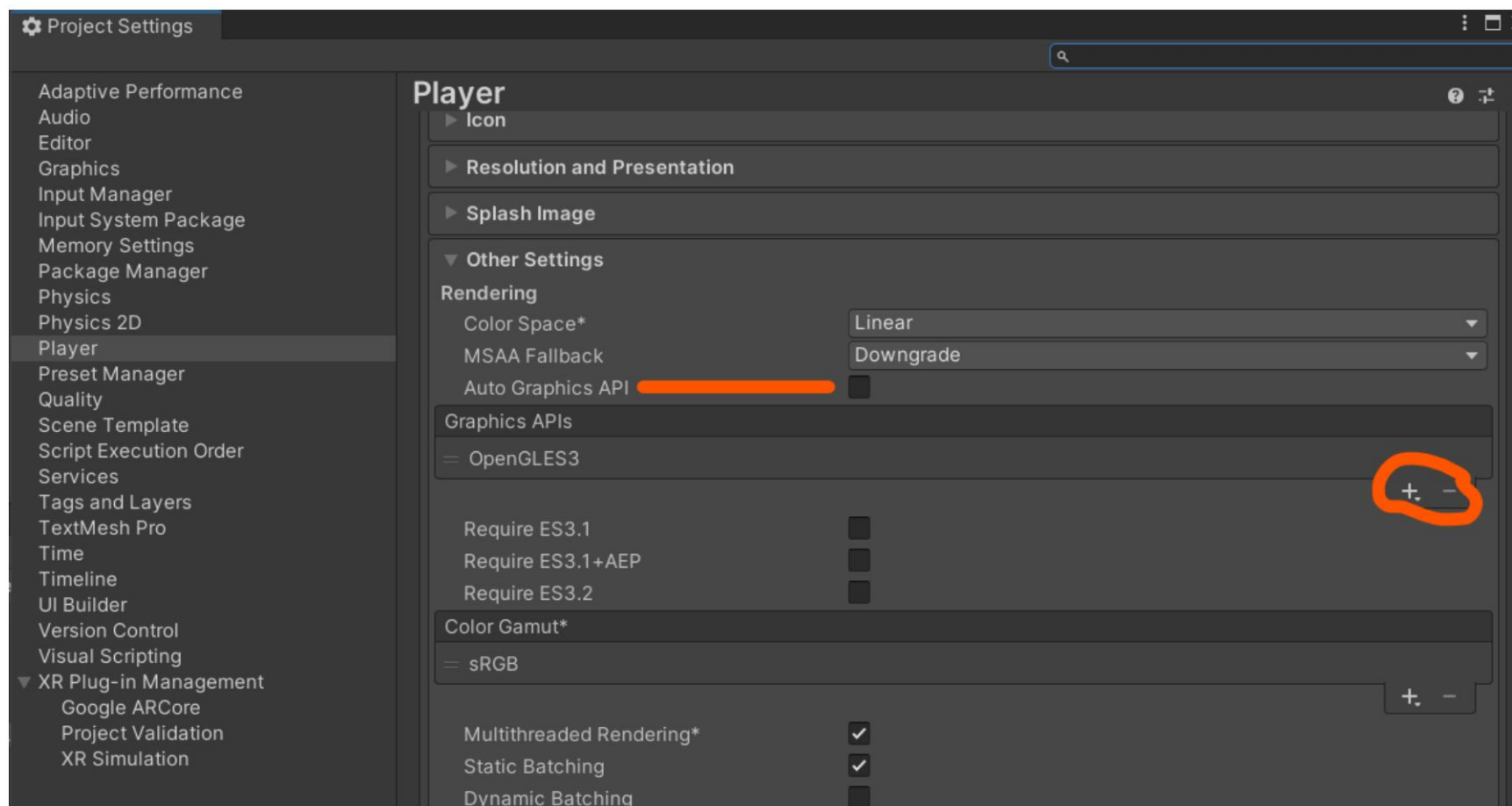


Cấu hình

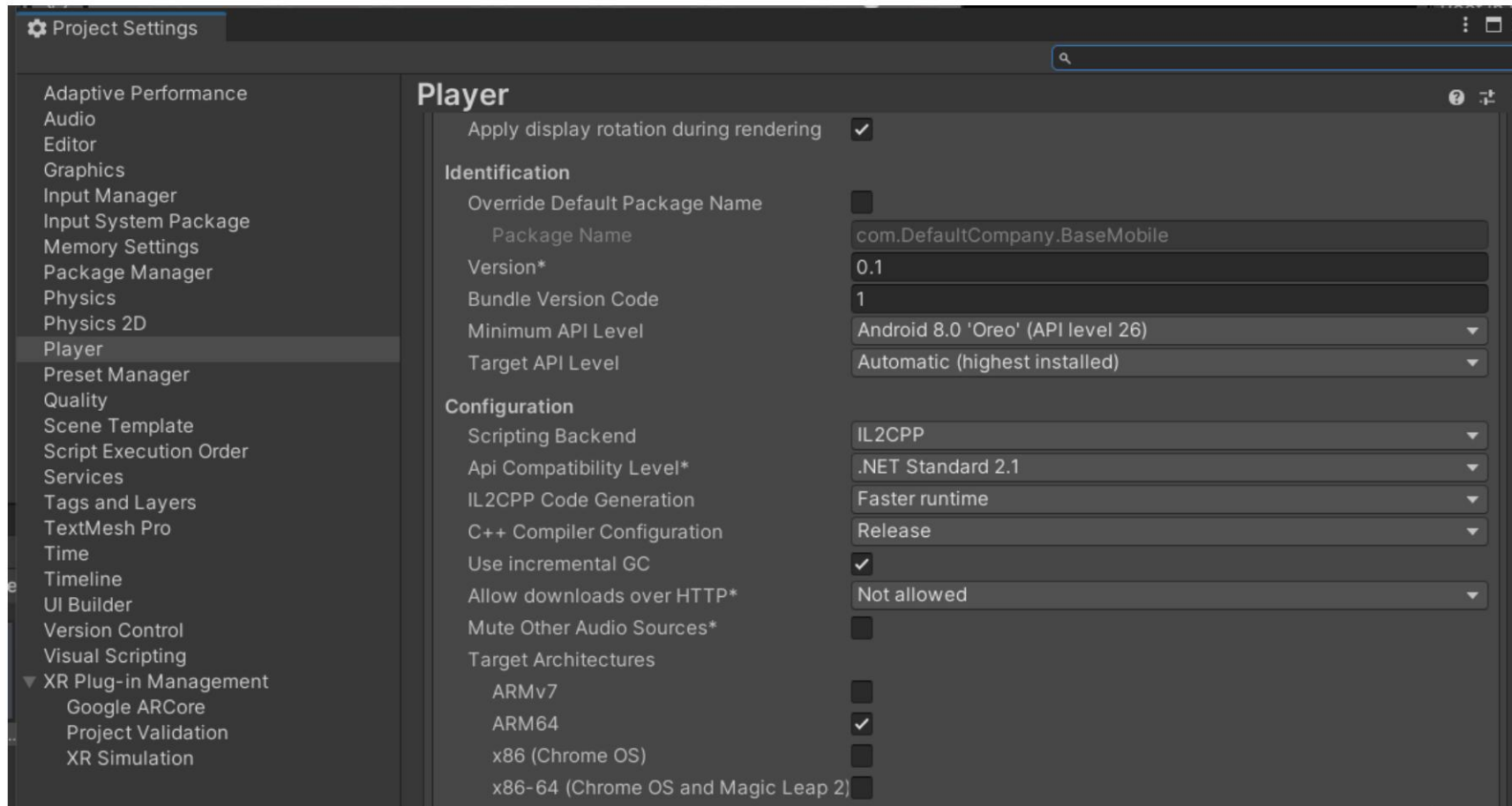


Cấu hình

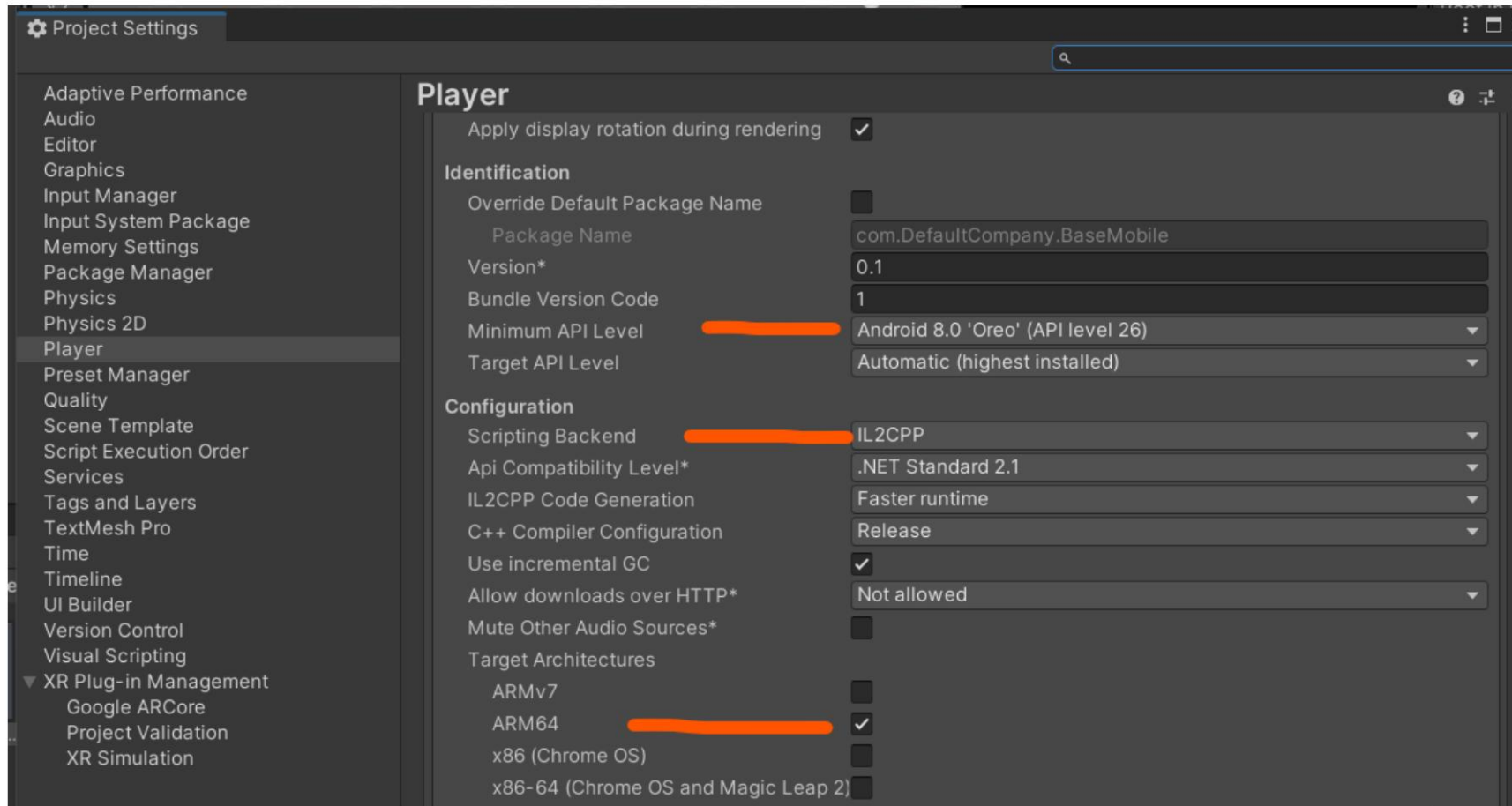
Xóa Vulkan



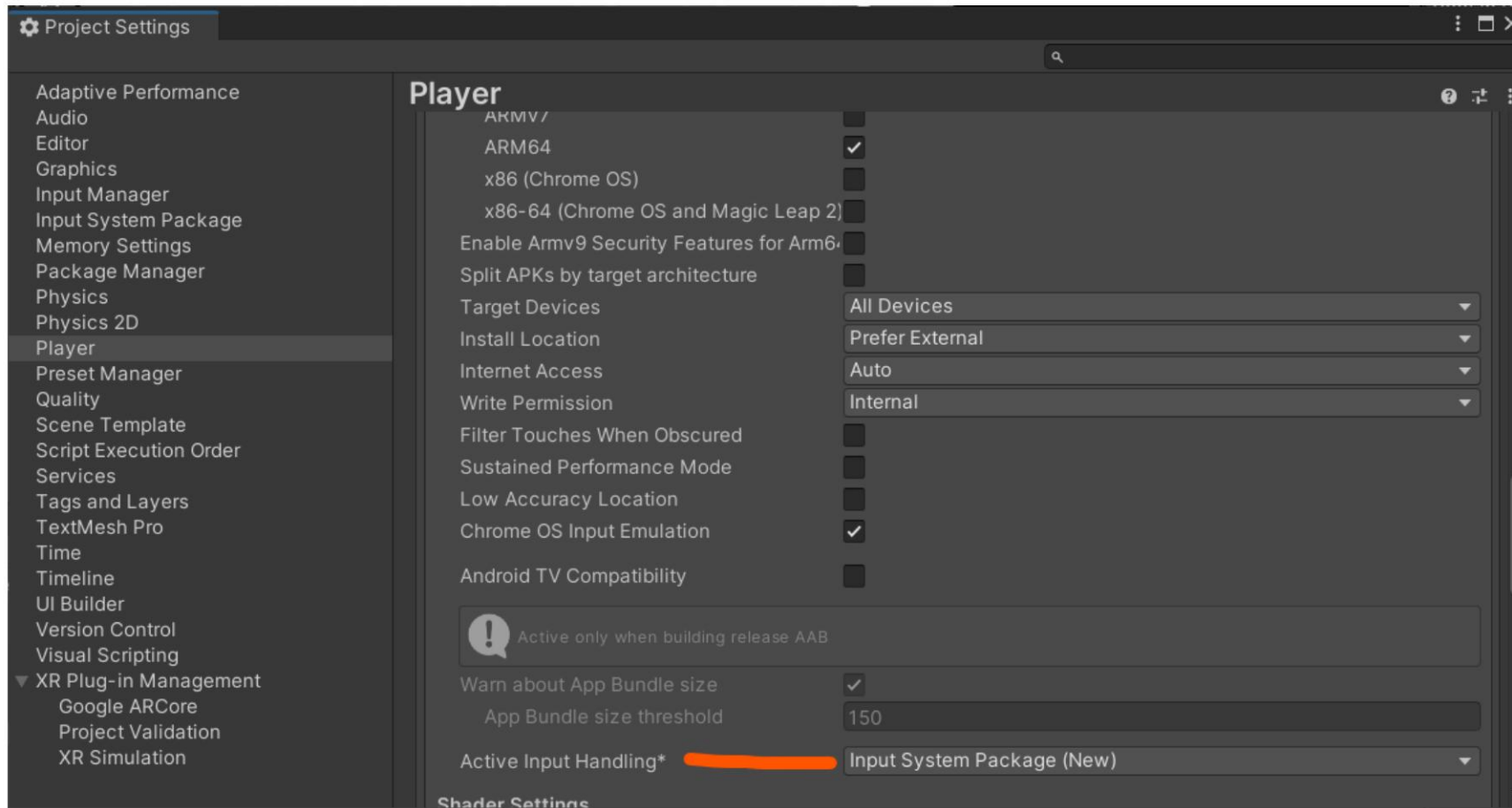
Cấu hình



Cấu hình



Cấu hình

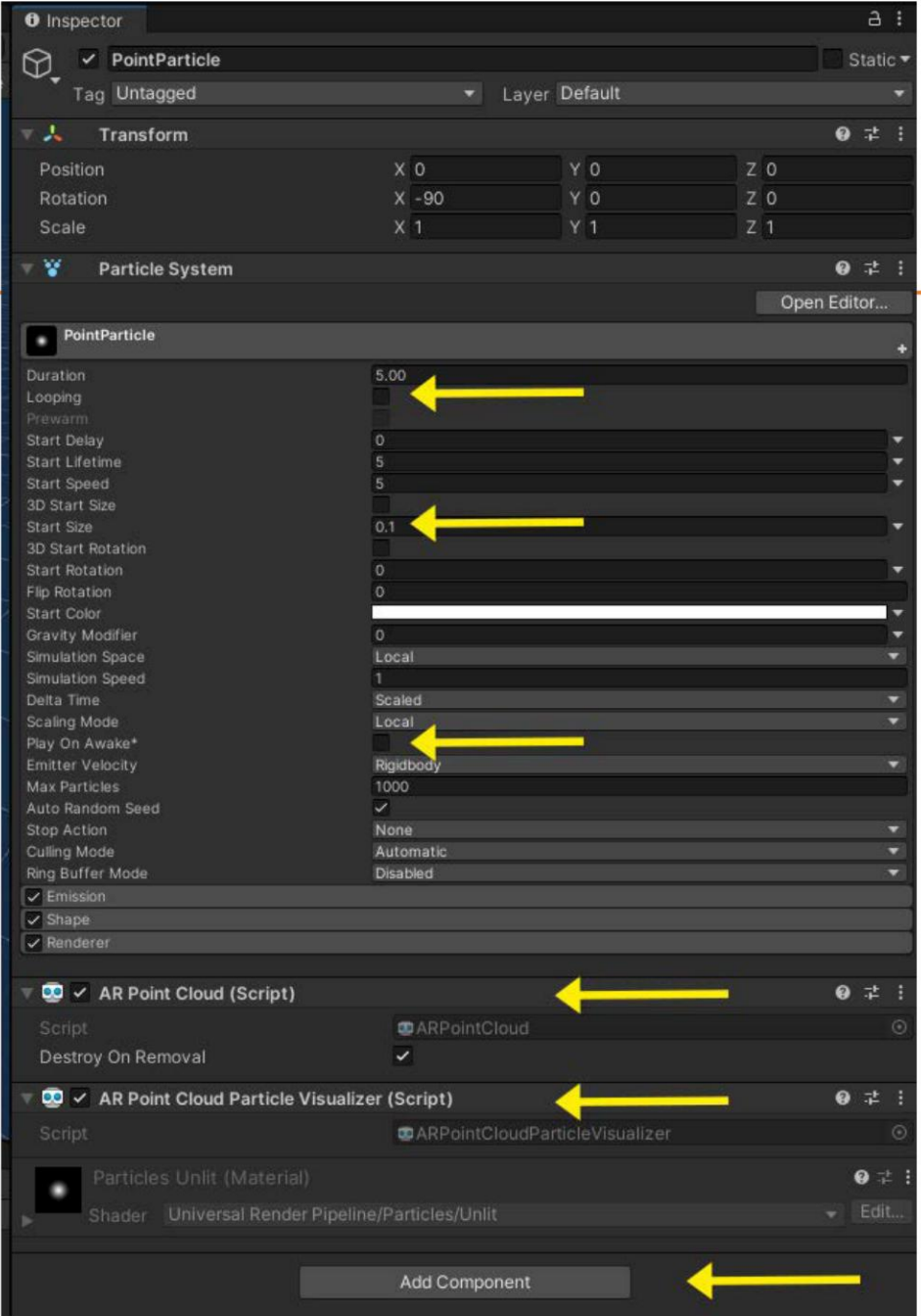


Xây dựng và chạy Point cloud

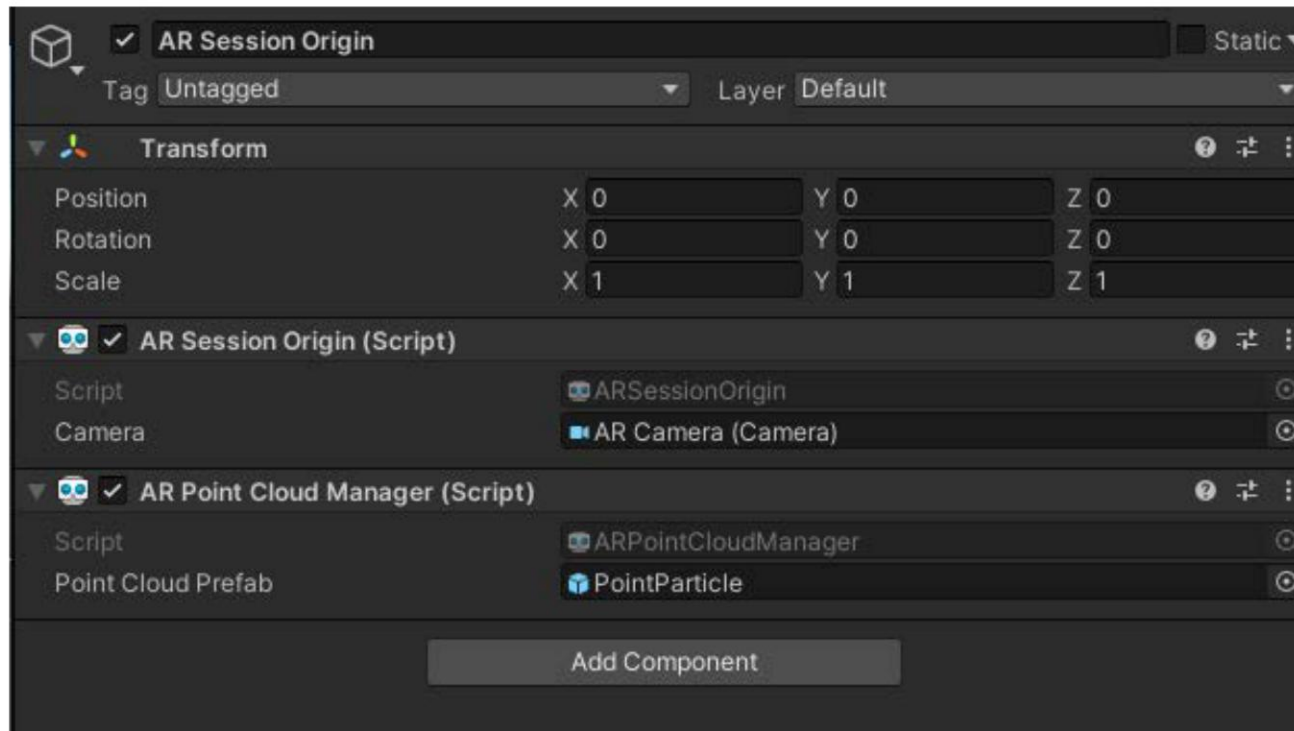
1. Tạo một cảnh mới có tên là PointCloud3D
2. Trong cửa sổ Phân cấp , hãy xóa Camera chính mặc định (nhấp chuột phải và chọn Xóa hoặc sử dụng phím Del trên bàn phím).
3. Thêm đối tượng Phiên AR bằng cách chọn GameObject | XR | Phiên AR.
4. Thêm đối tượng AR Session Origin bằng cách chọn GameObject | XR | AR Session Origin
5. Thêm trình quản lý đám mây điểm vào đối tượng Session Origin bằng cách nhấp vào Add Thành phần trong cửa sổ Inspector . Sau đó, nhập ar point vào trường tìm kiếm và chọn AR Point Cloud Manager.

Tạo Hệ thống Hạt

1. Tạo Hệ thống hạt bằng cách chọn GameObject | Effects | Hệ thống hạt.
2. Trong cửa sổ Inspector , đổi tên nó thành PointParticle.
3. Trên thành phần Hệ thống hạt , bỏ chọn hộp kiểm Lặp .
4. Đặt Kích thước bắt đầu thành 0,1.
5. Bỏ chọn hộp kiểm Phát khi thức .
6. Nhấp vào Thêm thành phần, nhập điểm ar vào trường tìm kiếm và chọn AR Đám mây điểm.
 1. Tương tự như vậy, hãy nhấp vào Thêm thành phần và chọn AR Point Cloud Visualizer.
 2. Kéo đối tượng PointParticle từ cửa sổ Hierarchy vào thư mục Prefabs
 3. trong cửa sổ Project (tạo thư mục trước nếu cần). Điều này làm cho
 4. Chuyển GameObject thành một mô hình dự ng sẵn.
 5. Xóa đối tượng PointParticle khỏi cửa sổ Hierarchy bằng cách nhấp chuột phải |
 6. Xóa hoặc nhấn phím Del .



Nguồn gốc phiên AR kết quả



Phát hiện máy bay

- Thêm thành phần •
XR Origin > AR Plane Manager
- AR Plane Manager > Plane Prefab > ARPlane

Nhấn để đặt Đối tượng

- Tạo bản dự ng máy bay của riêng bạn 1. Tạo đối tượng rỗng có tên là ARPlane 2. Thêm các thành phần:
 - Máy bay AR • Công cụ trực quan hóa lưới
 - máy bay AR • Bộ va chạm lưới • Bộ lọc lưới • Bộ kết xuất lưới • Bộ kết xuất đường
- Trình kết xuất lưới > Vật liệu > Trình trực quan hóa

Trình kết xuất dòng

Đỉnh góc	4
Đỉnh nắp cuối	4
Sử dụng Không gian Thế giới	Bỏ chọn
Bóng đổ	Tắt
Nhận bóng tối	Bỏ chọn
Phần tử 0	Dòng mặc định

Nhấn để đặt Đối tượng

sử dụng `System.Collections;`

sử dụng `System.Collections.Generic;`

sử dụng `Unity.VisualScripting;`

sử dụng `UnityEngine;`

sử dụng `UnityEngine.XR.ARFoundation;`

sử dụng `UnityEngine.XR.ARSubsystems;`

sử dụng `EnhancedTouch = UnityEngine.InputSystem.EnhancedTouch;`

Nhấn để đặt Đối tượng

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

```

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

```

```

void riêng OnDsable {
    EnhancedTouch.TouchSimulation.Disable();
    EnhancedTouch.EnhancedTouchSupport.Disable();
    EnhancedTouch.Touch.onFingerDown -=
FingerDown; }

```

```

riêng từ void FingerDown(EnhancedTouch.Finger finger) {

    if(finger.index != 0) return; // nhiều ngón tay di chuyển xuống == 1
    if(aRRcM.Raycast(finger.currentTouch.screenPosition,hits,trackableTypes:TrackableType.PlaneW ithinPolygon))
{ foreach(ARRaycastHit
    hit in hits){
        Pose poseH = hit.pose; // vị trí và hướng GameObject
        obj =
        Khởi tạo(bản gốc:prefab,vị trí:poseH.position,xoay:poseH.rotation);
    }
}
}

```

Đặt và di chuyển đối tượng

- XR Origin > AR Raycast Manager •

XR Origin > Script của bạn

Đặt và di chuyển đối tượng

```
sử dụng System.Collections;  
sử dụng System.Collections.Generic;  
sử dụng UnityEngine;  
sử dụng UnityEngine.XR.ARFoundation;  
sử dụng UnityEngine.XR.ARSubsystems;
```

```
[RequireComponent(typeof(ARRaycastManager))] lớp  
công khai ARTabToPlaceObject : MonoBehaviour {
```

```
    công khai GameObject goInstaintiate;  
    riêng tư GameObject spawnedObj;  
    riêng tư ARRaycastManager aRRCM;  
    riêng tư Vector2 touchPos;  
    tỉ nh Danh sách <ARRaycastHit> hits = danh sách mới <ARRaycastHit>();
```

```
khoảng trống Awake()
```

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

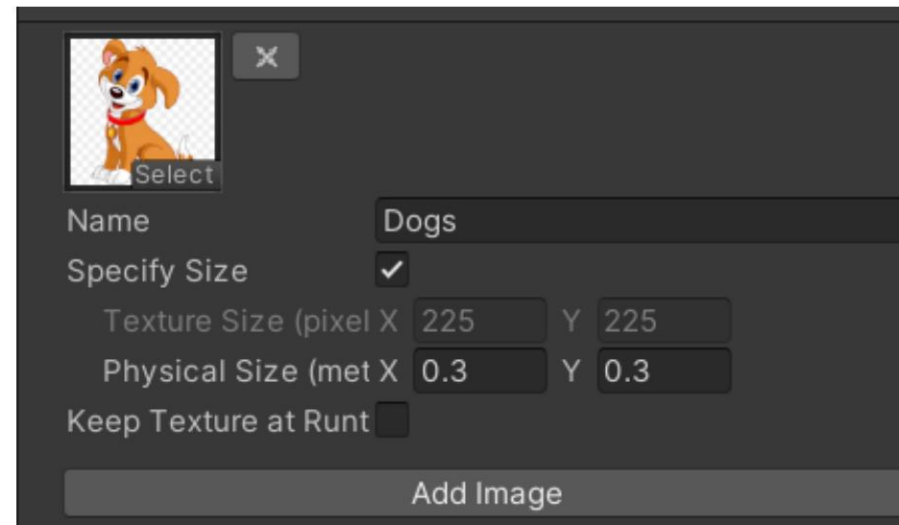
```
bool TryGetTouchPosition(ra Vector2 touchPosition) {  
  
    nếu(Đầu vào.touchCount>0){  
        touchPosition = Input.GetTouch(index:0).position; trả về true;  
  
    } touchPosition = mặc định;  
    trả về false;  
}
```

Đặt và di chuyển đối tượng

```
void Cập nhật() {  
  
    nếu( !TryGetComponent(out touchPos)){ trả về;  
  
    } nếu(aRRCM.Raycast(touchPos,hits,trackableTypes:TrackableType.PlaneWithinPolygon  
)){  
        Tư thế hitPose = hits[0].pose;  
        nếu(spawnedObj == null)  
            { spawnedObj =  
                Khởi tạo(goInstaintiate,hitPose.position,hitPose.rotation);  
  
            } khác  
            spawnedObj.transform.position = hitPose.position;  
  
    }  
  
}
```

Theo dõi hình ảnh

- Thư viện hình ảnh tham khảo



Theo dõi hình ảnh

```
sử dụng UnityEngine;
sử dụng UnityEngine.XR;
sử dụng UnityEngine.XR.ARFoundation; //
sử dụng UnityEngine.XR.ARSubsystems;

lớp công khai ImageRecognize : MonoBehaviour {

    riêng tư ARTrackedImageManager aRTIManager;
    void Awake()
    {
        aRTIManager = FindObjectOfType<ARTrackedImageManager>();
    }
}
```


Theo dõi hình ảnh

```
vô hiệu OnEnable()  
{  
    aRTIManager.trackedImagesChanged += OnImageChanged;  
}  
  
vô hiệu OnDisable()  
{  
    aRTIManager.trackedImagesChanged -= OnImageChanged;  
}  
  
công khai void OnImageChanged(ARTrackedImagesChangedEventArgs args)  
{  
    foreach(var trackedImage trong args.added){  
        Debug.Log(trackedImage.name);  
    }  
}
```