

Лабораторна робота №1

Налаштування та запуск AR проєкту в рушії Unity

Мета: Ознайомити з компонентами та пакунками ігрового рушія Unity для створення AR проєктів, навчити застосовувати їх для налаштування проєктів, отримати практичні навички із налаштування мобільних пристроїв, що працюють на ОС Android, для запуску AR проєктів.

Література

<https://docs.unity3d.com/6000.5/Documentation/Manual/AROverview.html>

<https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.5/manual/features/plane-detection.html>

<https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.5/manual/features/raycast.html>

Методичні рекомендації

Ігровий рушій Unity має компоненти та пакунки, що призначені для роботи з мобільними пристроями. Вони дозволяють аналізувати зображення з камер пристроїв та реалізовувати різні сценарії роботи доповненої реальності.

Методичні вказівки з даної дисципліни засновані на роботі з Unity 6.5 версії 6000.5.9f1 (рис. 1). Перед встановленням необхідно обрати підтримку пристроїв, що працюють на ОС Android та iOS, як додаткові модулі рушія (рис. 2).

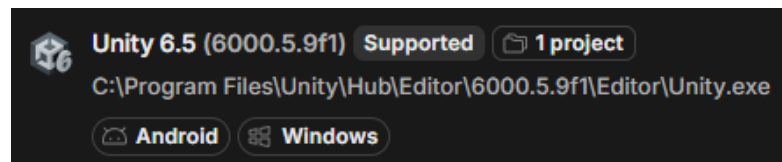


Рисунок 1 – Рекомендована версія ігрового рушія Unity

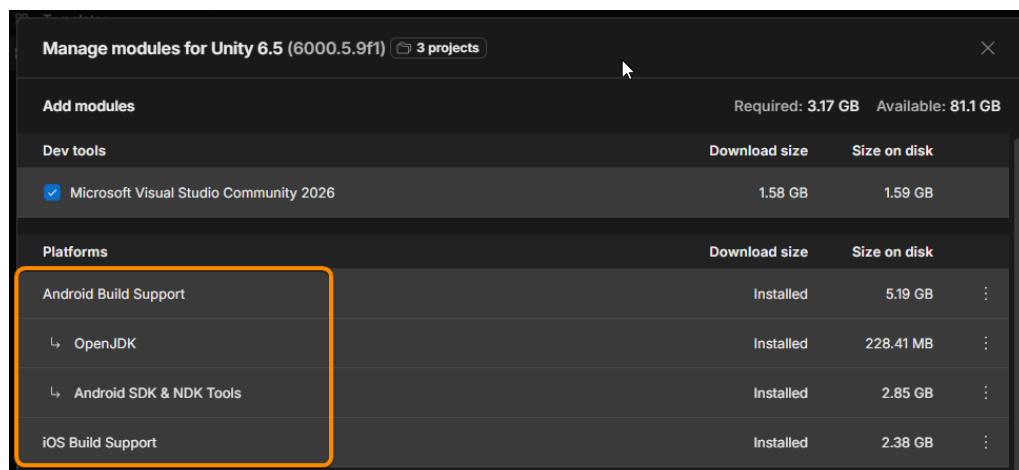


Рисунок 2 – Рекомендовані модулі для встановлення на ігровий рушій

Ігровий рушій має багато шаблонів, з яких можна почати розробку власної гри, зокрема зразок готової сцени під роботу AR на мобільних пристроях. Мета поточної роботи передбачає налаштування будь-якого 3D проєкту для реалізації технологій AR, тому подальші дії будуть описані на базі порожнього проєкту Universal 3D (рис. 3).

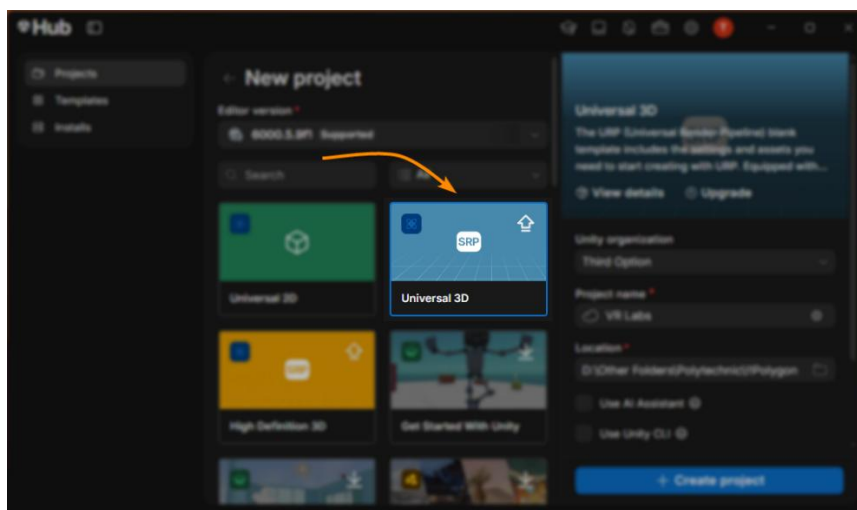


Рисунок 3 – Шаблон проєкту

Налаштування проєкту відбувається за наступними кроками:

1. Встановлення пакунку XR Plugin Management.

XR Plugin Management є основним пакунком для роботи AR в ігрових проєктах.

Встановлення пакунку відбувається через Project Settings за наступним алгоритмом: Edit → Project Settings → XR Plugin Management → Install XR Plugin Management. Візуально алгоритм представлено на рисунку 4.

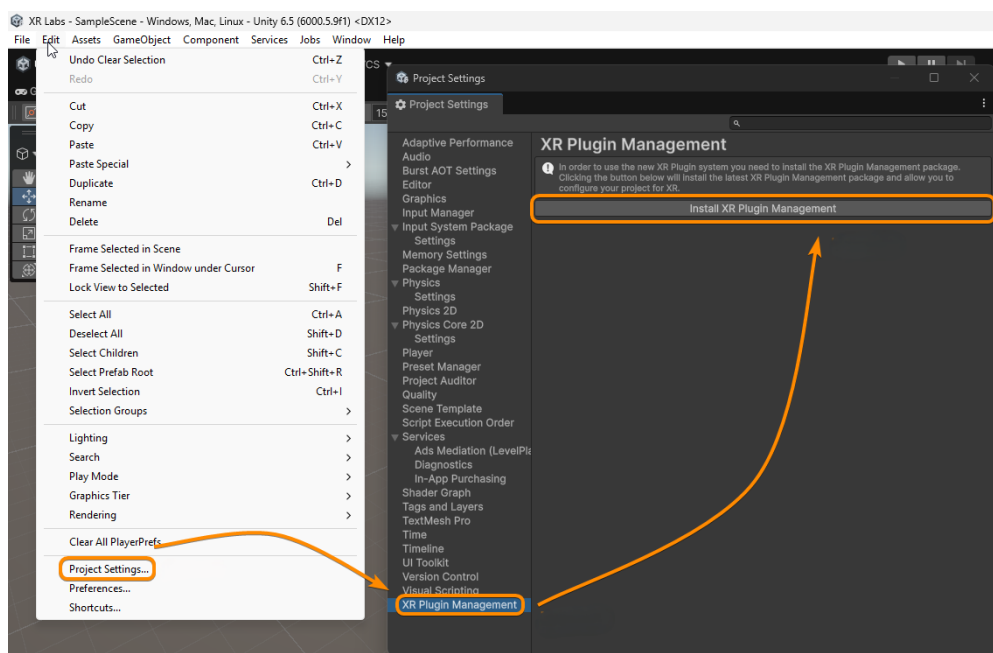


Рисунок 4 – Алгоритм встановлення пакунку XR Plugin Management

2. Налаштування пакунку XR Plugin Management.

Подальші дії будуть відбуватись на сторінці XR Plugin Management вікна Project Settings. Алгоритм відкриття вікна є в пункті 1 налаштування проєкту.

Для забезпечення коректної роботи проєкту на мобільних пристроях, що працюють на ОС Android, необхідно обрати Google ARCore серед списку провайдерів для Android пристроїв, а для ОС iOS – Apple ARKit у відповідній вкладці. Для можливості тестування додатку без мобільного пристрою, необхідно обрати XR Simulation у вкладці для персональних комп'ютерів. Покрокове налаштування представлено на рисунку 5.

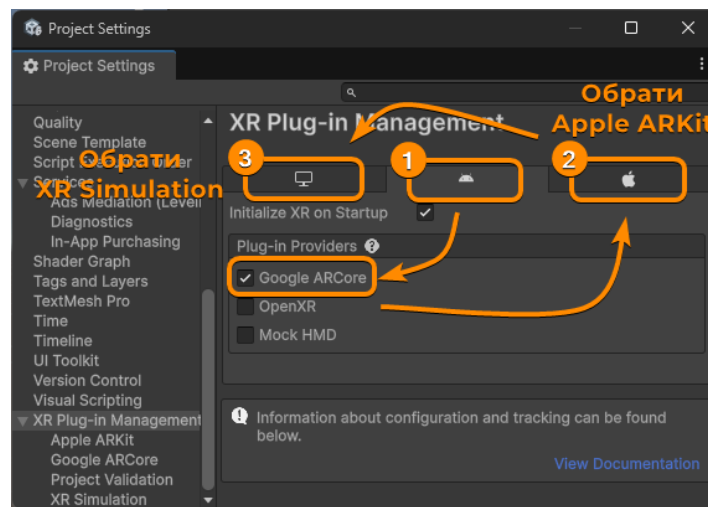


Рисунок 5 – Налаштування провайдерів для XR Plugin Management

Наступний крок передбачає перевірку та автоматичне виправлення помилок, що можуть нашкодити подальшій роботі з мобільними пристроями.

Для його виконання необхідно перейти в пункт Project Validation та натиснути Fix All у всіх вкладках. Якщо залишаються помилки, протисніть на Fix та збережіть проєкт комбінацією клавіш Ctrl+S (рис. 6).

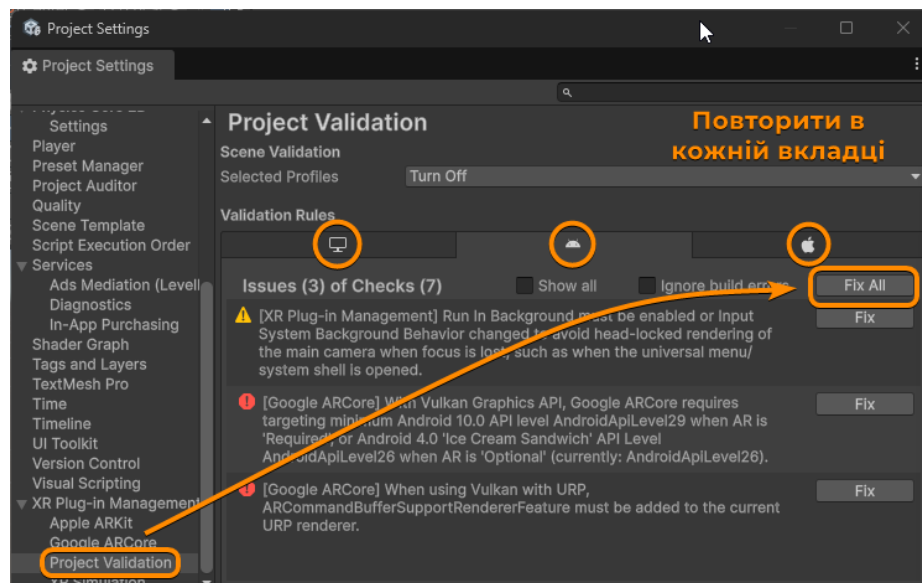


Рисунок 6 – Алгоритм виправлення помилок, пов'язаних з роботою AR

3. Виправлення основних налаштувань проєкту.

Поточний крок передбачає виправлення помилки, що може перешкоджати коректному запуску проєкту на мобільних пристроях, що працюють на ОС Android.

Помилкою є виставлений API Vulkan як той, що відповідає за графіку. Зараз він не підтримується на мобільних пристроях, тому під час запуску проєкт може показати лише чорний екран.

Вирішенням проблеми є прибирання Vulkan зі списку графічних API для пристроїв Android, що знаходиться у вкладці Other Settings налаштувань Player (рис. 7).

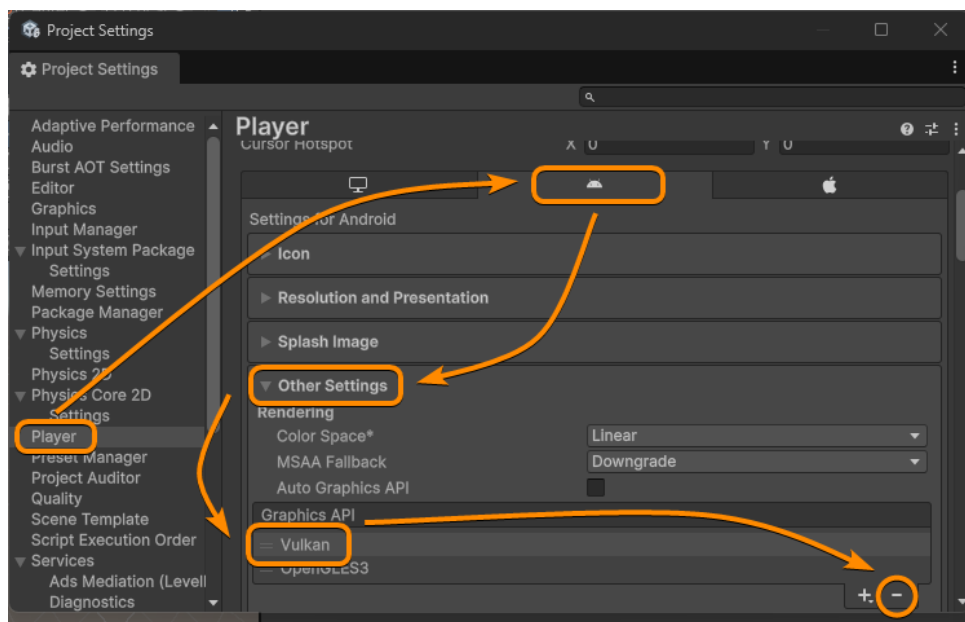


Рисунок 7 – Алгоритм прибирання API Vulkan для мобільних пристроїв на базі Android

4. Налаштування ігрової сцени.

Сцена потребує додавання певних об'єктів, що забезпечить роботу AR через камеру мобільного пристрою. Результатом налаштування повинна стати ієрархія об'єктів, схожим на приклад, що представлений на рисунку 8.

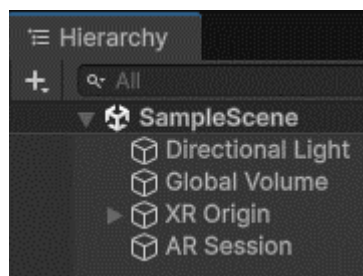


Рисунок 8 – Приклад ієрархії ігрових об'єктів на сцені для початкової роботи з AR

Для досягнення мети необхідно виконати наступні кроки:

- 1) Видалити об'єкт Main Camera.

2) Додати об'єкти XR Origin (Mobile AR) та AR Session, що знаходяться в контекстному меню за шляхом XR → XR Origin (Mobile AR) та XR → AR Session відповідно, в зазначеній послідовності (рис. 9). Вони забезпечують рух та поворот камери відповідно до реального положення мобільного пристрою, а також зчитують зображення з камери пристрою.



Рисунок 9 – Алгоритм додавання об'єкта XR Origin (XR)

3) Додати до об'єкту XR Origin компоненти AR Plane Manager та AR Raycast Manager. Перший відповідає за розпізнавання поверхонь на камері пристрою та їх відображення на сцені, а другий – за пошук поверхні, на яку буде розміщуватись бажаний об'єкт, через запуск променя (рис. 10).

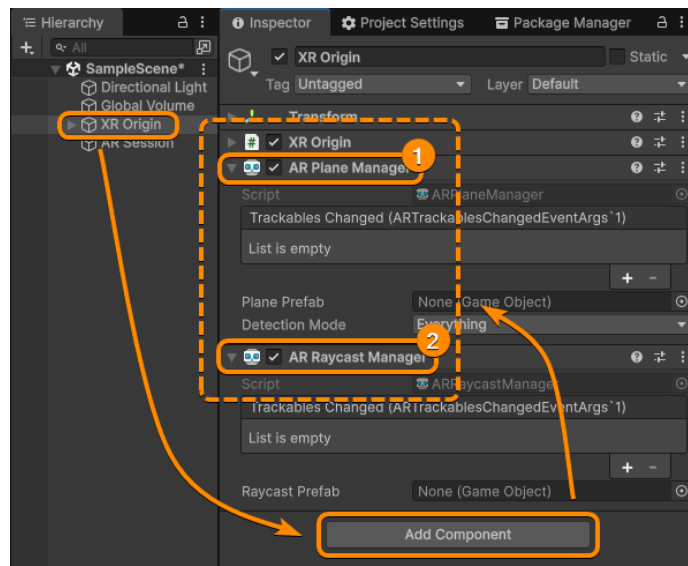


Рисунок 10 – Алгоритм додавання компонентів AR Plane Manager AR та Raycast Manager до об'єкта XR Origin

4) Налаштувати компонент AR Plane Manager:

- а. додати до сцени об'єкт AR Default Plane, що знаходиться в контекстному меню за шляхом XR → AR Default Plane, що відповідатиме за зовнішній вигляд розпізнаних поверхонь на сцені (рис. 11);



Рисунок 11 – Алгоритм додавання об'єкта AR Default Plane

- b. створити з об'єкта AR Default Plane префаб в довільному місці (бажано в окремій папці для префабів з назвою «Prefabs») (рис. 12);

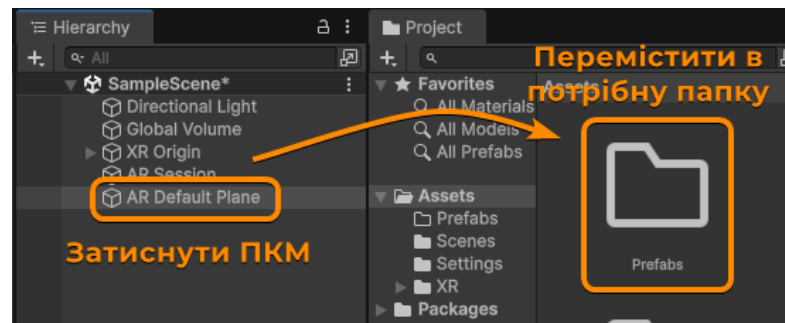


Рисунок 12 – Створення префабу з об'єкта AR Default Plane

- c. видалить об'єкт AR Default Plane зі сцени;
- d. вкажіть в поле Plane Prefab компонента AR Plane Manager, що був доданий до об'єкта XR Origin, префаб AR Default Plane (рис. 13);

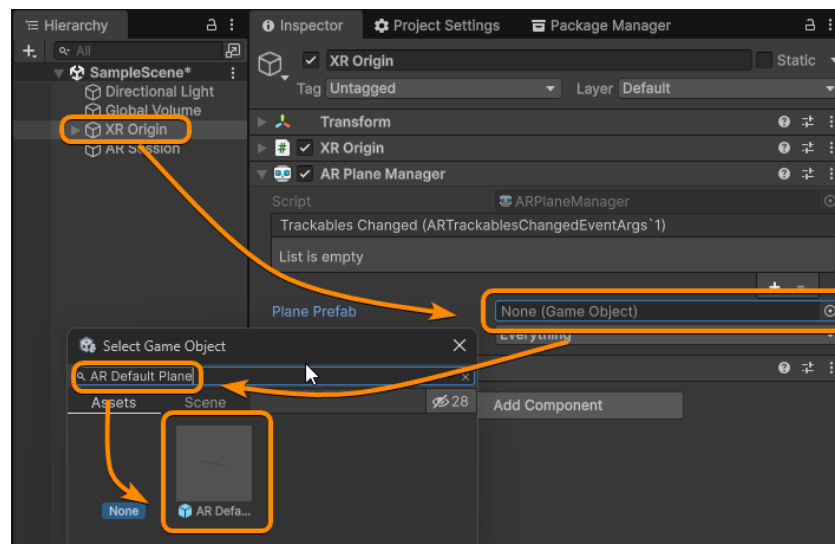


Рисунок 13 – Створення префабу з об'єкта AR Default Plane

5) Налаштувати приклад середовища для тестування AR можливостей на комп'ютері:

- a. відкрити вікно XR Environment, що знаходиться за шляхом Window → XR → AR Foundation → XR Environment (рис. 14);
- b. завантажити тестове середовище, вибравши відповідний пункт в списку панелі XR Environment (рис. 15) та імпортувати необхідні файли через натискання кнопки Import (рис. 16);
- c. закрити вікно після відображення у вікні моделі будинку та заднього дворику;

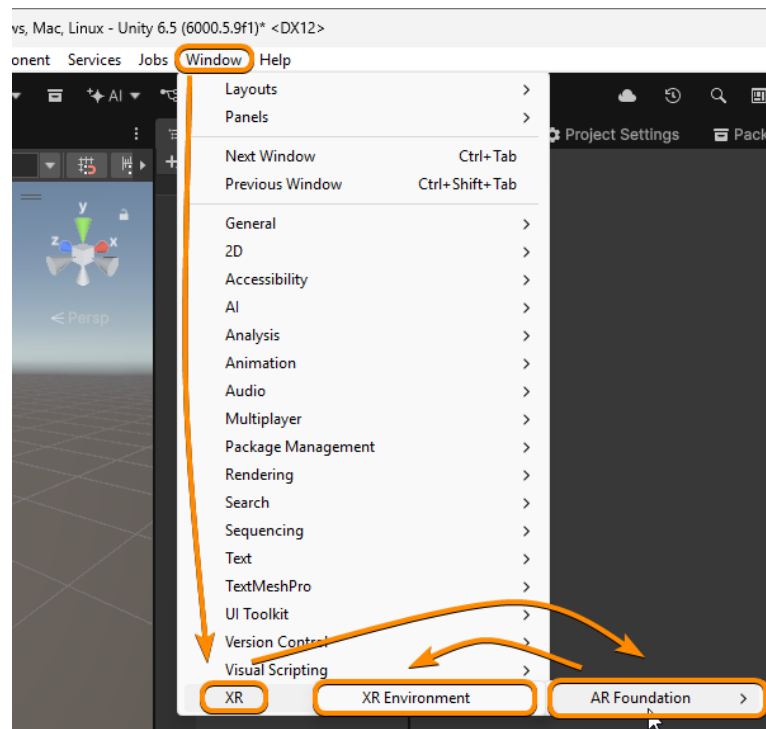


Рисунок 14 – Алгоритм відкриття вікна XR Environment

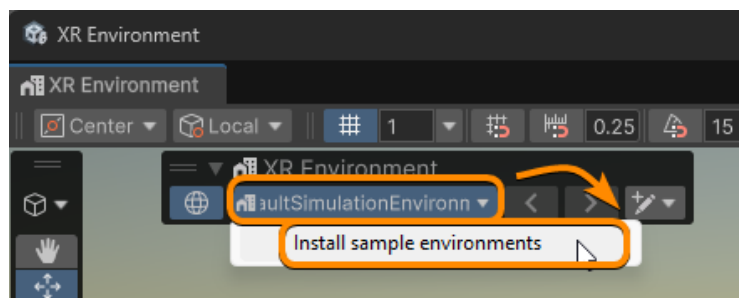


Рисунок 15 – Алгоритм початку встановлення тестового середовища

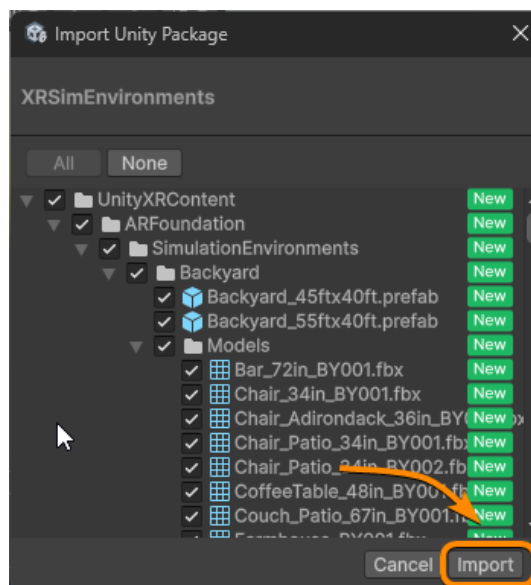


Рисунок 16 – Імпорт файлів тестового середовища

6) Зменшити параметр Intensity до значення 1 в компоненті Light об'єкта Directional Light (рис. 17);

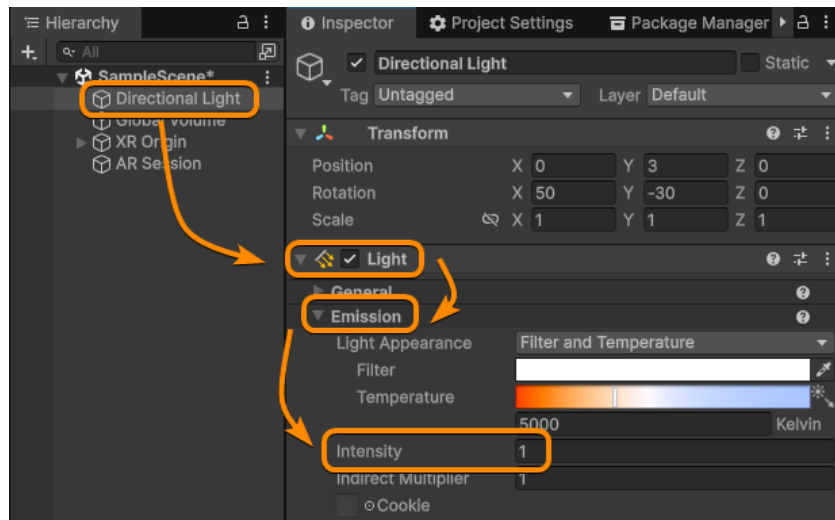


Рисунок 17 – Алгоритм зміни параметра Intensity для світла

7) Додати до файлів Mobile_Renderer та PC_Renderer компонент AR Background Renderer Feature (рис. 18).

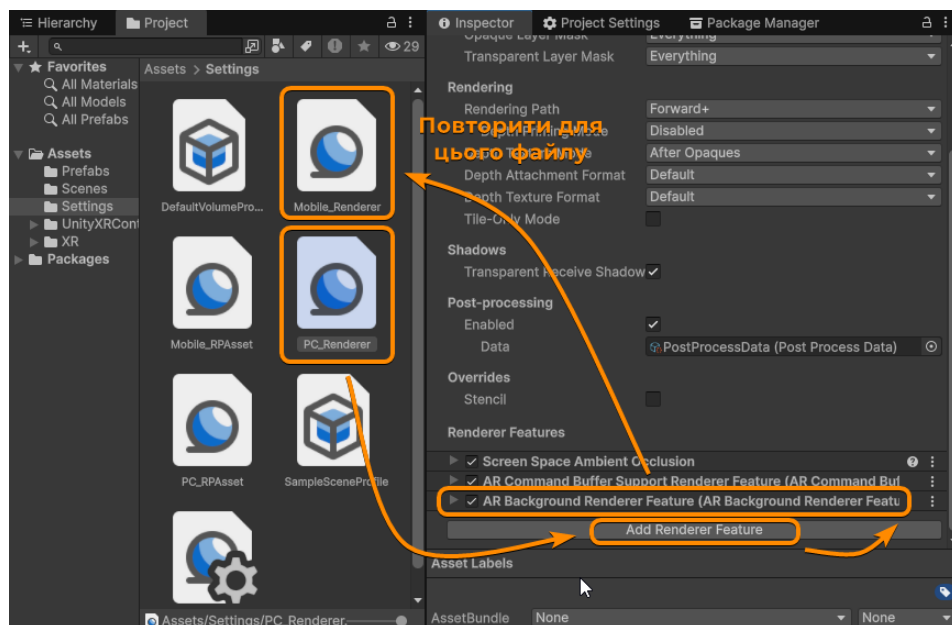


Рисунок 18 – Алгоритм додавання компоненту AR Background Renderer Feature

Після того, як було виконано налаштування сцени, можна запустити проєкт та перевірити розпізнавання поверхонь на тестовій сцені. Для повороту камери необхідно затиснути праву кнопку миші, а для руху камерою – ПКМ та кнопку руху (WASD). Під час зміни положення камери будуть поступово появлятися невеликі площини, які було розпізнано, та з часом зливатись в суцільну площину, які можна програмно застосувати для різних цілей.

Якщо сцену було успішно налаштовано, запуск проєкту та рух камерою дасть результат, представлений на прикладі з рисунку 19.

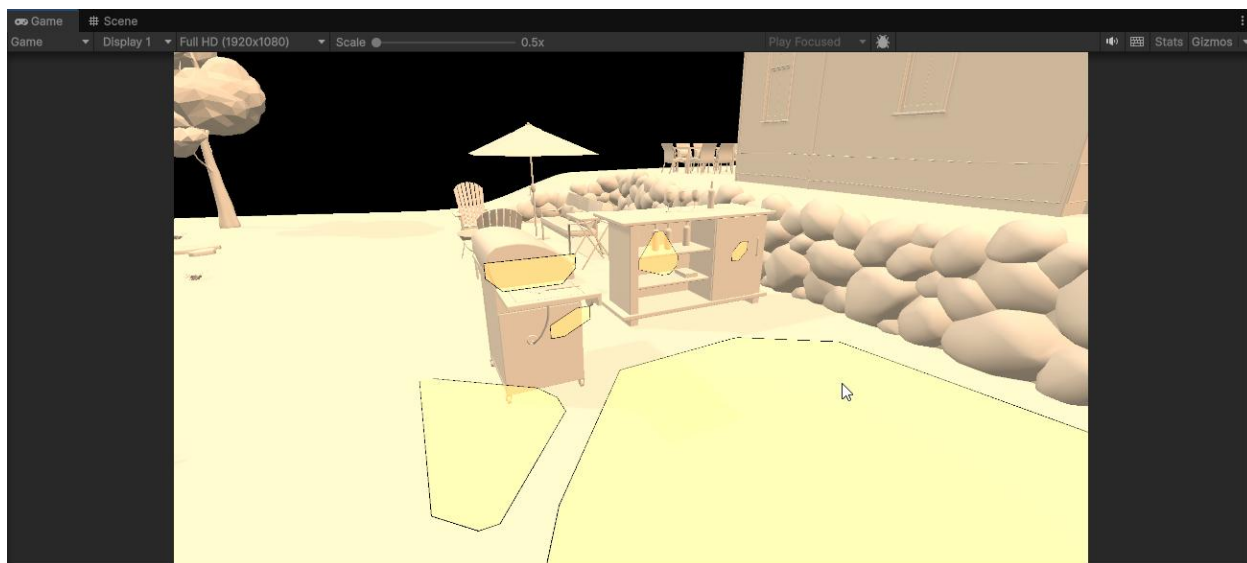


Рисунок 19 – Приклад розпізнавання поверхонь після успішного налаштування проєкту та сцени

Поверхні, що були розпізнані, в початкових налаштуваннях відображаються у вигляді напівпрозорих жовтих областей з чорним контуром. Якщо необхідно змінити візуалізацію поверхонь, або прибрати її, необхідно зайти в префаб AR Default Plane, створений раніше, та скоригувати наступні параметри:

- 1) Material компоненту Mesh Renderer, що відповідає за «заливку» поверхні (можна змінити або видалити);
- 2) Color компоненту Line Renderer, що відповідає за контур поверхні (можна змінити колір та прозорість).

В рамках роботи передбачається розміщення об'єктів в просторі, тому потрібно написати програмний код, що дозволить це зробити, а також префаб об'єкта для його передачі в програмний код з ціллю його створення.

Щоб створити потрібний префаб, необхідно виконати наступні дії (приклад ієрархії об'єкта представлено на рисунку 20):

- 1) створити на сцені об'єкт PrefabParent з нульовими координатами;
- 2) до об'єкта PrefabParent необхідно додати будь-який 3D об'єкт (для прикладу буде додано куб);
- 3) налаштувати розміри 3D об'єкта, а також його положення таким чином, щоб його нижня частина «стояла» на розмітченій сітці сцени (приклад налаштування компонента Transform представлено на рисунку 16);
- 4) створити з об'єкта PrefabParent префаб так само, як це було представлено в минулих пунктах;
- 5) видалити об'єкт зі сцени;

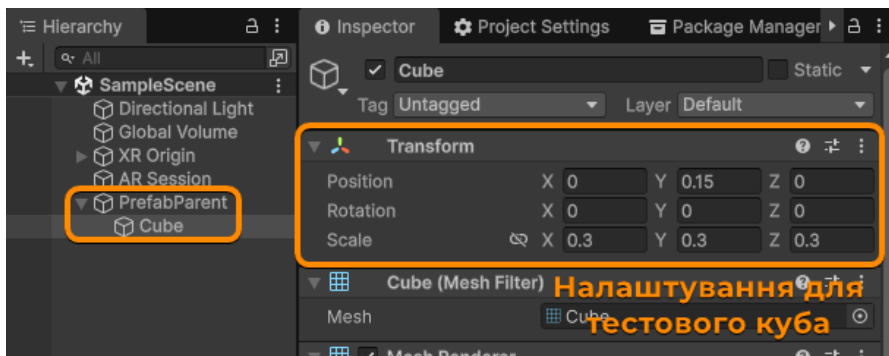


Рисунок 20 – Приклад ієрархії об'єкта PrefabParent та налаштування положення та розмірів тестового куба

Далі потрібно створити список дій, які буде зчитувати скрипт кожен кадр. Буде використано пакунок Input System, що дозволяє додавати підтримку зчитування сигналів з поширених пристроїв введення.

Для налаштування списку дій необхідно виконати наступні дії:

- 1) у вікні проєкту, де знаходяться файли, натиснути контекстне меню та створити Input Actions з назвою AR Inputs (рис. 21);

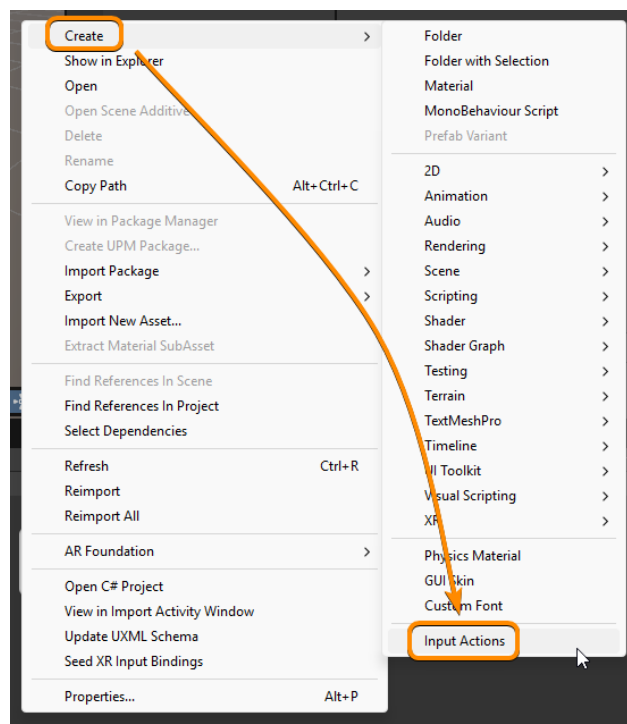


Рисунок 21 – Алгоритм додавання файлу зі списком дій Input Actions

- 2) відкрити файл та додати нову мапу дій (рис. 22);

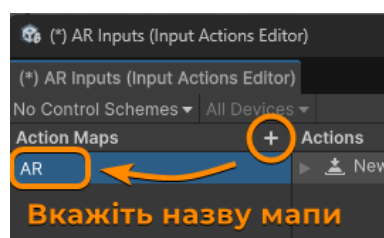


Рисунок 22 – Алгоритм додавання нової мапи дій

3) створити 2 дії: Click з типом дії Button для зчитування натискання та Click Position з типом дії Value та типом даних Vector 2, що передає координати натискання (рис. 23);

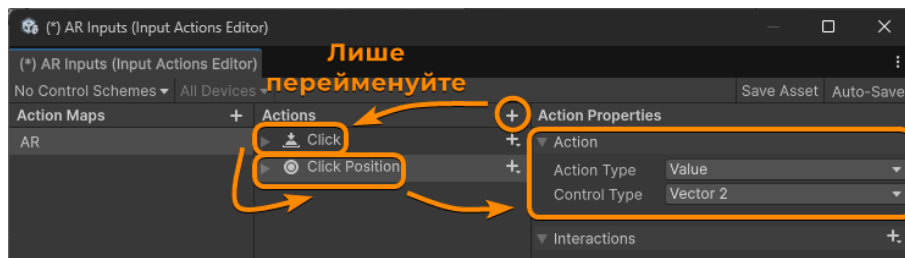


Рисунок 23 – Алгоритм додавання дій

4) повторити прив'язку сигналів з миші та сенсорного екрану до дій з рисунку 24 (алгоритм додавання нових сигналів представлений на рисунку 25);

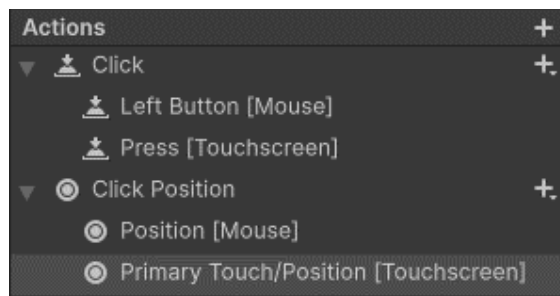


Рисунок 24 – Приклад налаштованих дій з прив'язаними сигналами

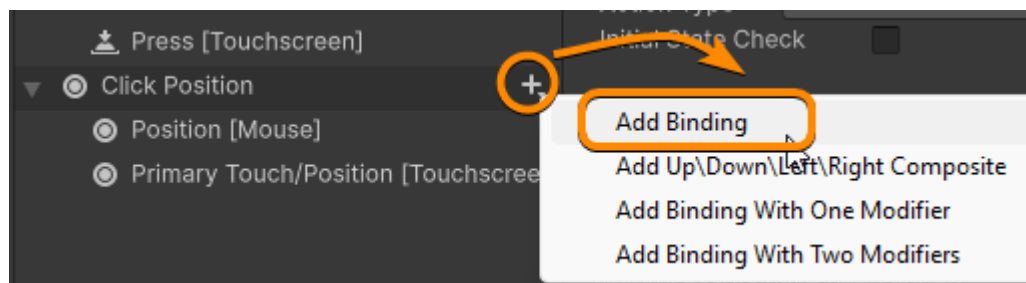


Рисунок 25 – Алгоритм додавання нової прив'язки сигналу до дії

5) додайте до об'єкта XR Origin компонент Player Input, вкажіть в поле Actions створений файл списку дій, а в поле Default Map – назву створеної мапи дій (рис. 26).

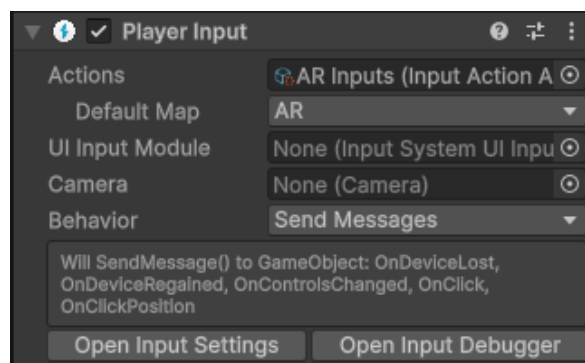


Рисунок 26 – Приклад налаштування компоненту Player Input

Наступним кроком є створення скрипту. Для цього необхідно створити окрему папку Scripts та в ній створити скрипт з назвою ARObjectPlacer, що наслідуює MonoBehaviour.

Будуть використані наступні бібліотеки:

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

Необхідно додати наступні поля, значення яких вказуються через Інспектор:

```
// Компонент, що відповідає за пошук поверхонь для розміщення об'єктів
[SerializeField] private ARRaycastManager raycastManager;

// Об'єкт, що буде створюватись
[SerializeField] private GameObject prefab;

// Файли сигналів з Input System
[SerializeField] private InputActionProperty clickAction;
[SerializeField] private InputActionProperty clickPositionAction;
```

В кінці скрипту потрібно додати ввімкнення та вимкнення покращеного зчитування натискань по екрану:

```
private void OnEnable() => EnhancedTouchSupport.Enable();
private void OnDisable() => EnhancedTouchSupport.Disable();
```

Алгоритм скрипту повинен бути наступним:

1) В Update зчитується спрацьовування змінної дії clickAction: clickAction.action.triggered. При спрацьовуванні відбуватиметься зчитування координат в полі clickPositionAction: clickPositionAction.action.ReadValue<Vector2>(), після чого відбувається процес створення об'єкта.

Зчитування натискань та координат можна реалізувати в інший спосіб за бажанням студента. В прикладі після натискання спрацьовує окремий метод PlaceObject, алгоритм якого буде описано далі:

```
if (clickAction.action.triggered)
{
    PlaceObject(clickPositionAction.action.ReadValue<Vector2>());
}
```

2) Викликається метод raycastManager.Raycast(), в який передається координати натискання, список зіткнень з променем по цій координаті (rayHits), а також бажаний тип об'єктів, з якими повинен зіштовхнутись промінь. TrackableType.PlaneWithinPolygon вказує, що об'єкт точно повинен бути поверхнею в полігоні (без вказування цього типу об'єкт

створюватиметься в некоректному місці). Лістинг необхідних рядків коду представлено нижче:

```
var rayHits = new List<ARRaycastHit>();  
raycastManager.Raycast(touchPosition, rayHits,  
    TrackableType.PlaneWithinPolygon);
```

3) Якщо список зіткнень (в прикладі ним є список rayHits) не порожній, зчитується position та rotation зіткнення в просторі з першого зіткнення rayHits[0].pose, і відбувається створення об'єкта на сцені за допомогою методу Instantiate():

```
if (rayHits.Count > 0)  
{  
    Vector3 hitPosePosition = rayHits[0].pose.position;  
    Quaternion hitPoseRotation = rayHits[0].pose.rotation;  
    Instantiate(prefab, hitPosePosition, hitPoseRotation);  
}
```

Також потрібно реалізувати таймер між створеннями об'єктів для запобігання хибних спрацьовувань. Реалізація таймера є довільною та базується на дисциплінах, пройдених раніше: створення корутини або фіксування часу наступного можливого створення в окремій змінній класу, яка буде перевірятись в Update.

Готовий скрипт потрібно додати до об'єкта XR Origin та вказати значення необхідних полів:

- Raycast Manager знаходиться в об'єкті та існує в одному екземплярі;
- Prefab передбачає додавання створеного раніше префаба з об'єктом;
- Click Action та Click Position Action повинні містити дії, що були створені раніше. Алгоритм перенесення дій до полів вказаний на рисунку 27.

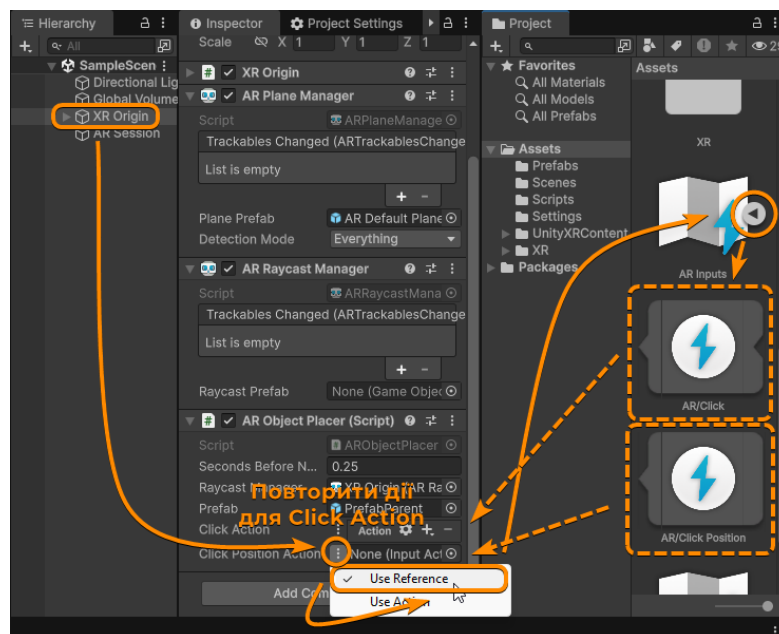


Рисунок 27 – Алгоритм перенесення дій до полів компоненту AR Object Placer

Після всіх кроків можна запуснути проєкт. У випадку успішної реалізації скрипту та списку дій, при натисканні по розпізнаним поверхням будуть створюватись об'єкти, що «стоять» на них (рис. 28).

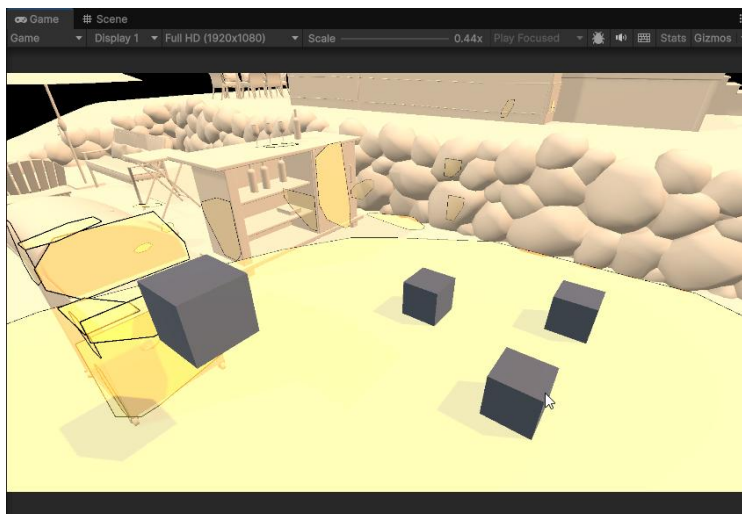


Рисунок 28 – Приклад успішної роботи розміщення об'єктів на поверхнях

Останнім кроком є запуск проєкту на мобільному пристрої.

Перед експортом необхідно увімкнути на пристрої можливість налагодження через USB. Для цього необхідно зайти в розділ Параметри розробника, увімкнути «Використовувати Параметри розробника», знайти параметр «Налагодження USB» та увімкнути його. Приклад налаштування пристрою Google Pixel 9 Pro представлено на рисунку 29.

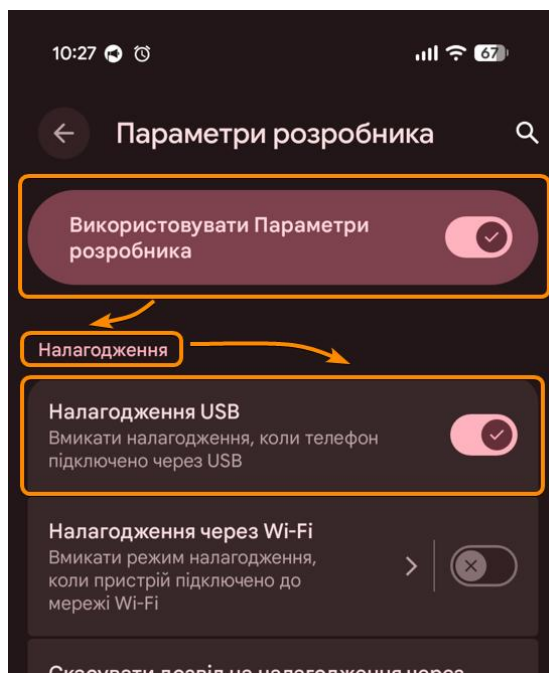


Рисунок 29 – Приклад увімкнення налагодження через USB для пристрою Google Pixel 9 Pro

Запуск проєкту на мобільному пристрої відбувається за наступним алгоритмом:

1) Відкрити вікно Build Profiles, що знаходиться за шляхом File → Build Profiles (рис. 30).

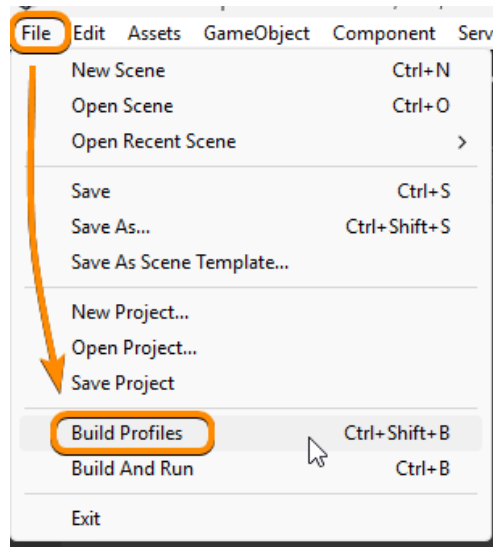


Рисунок 30 – Алгоритм відкриття вікна Build Profiles

2) Обрати в списку платформ Android та натиснути кнопку Switch Platform (рис. 31). Необхідно зачекати певний час поки відбувається перехід на платформу.

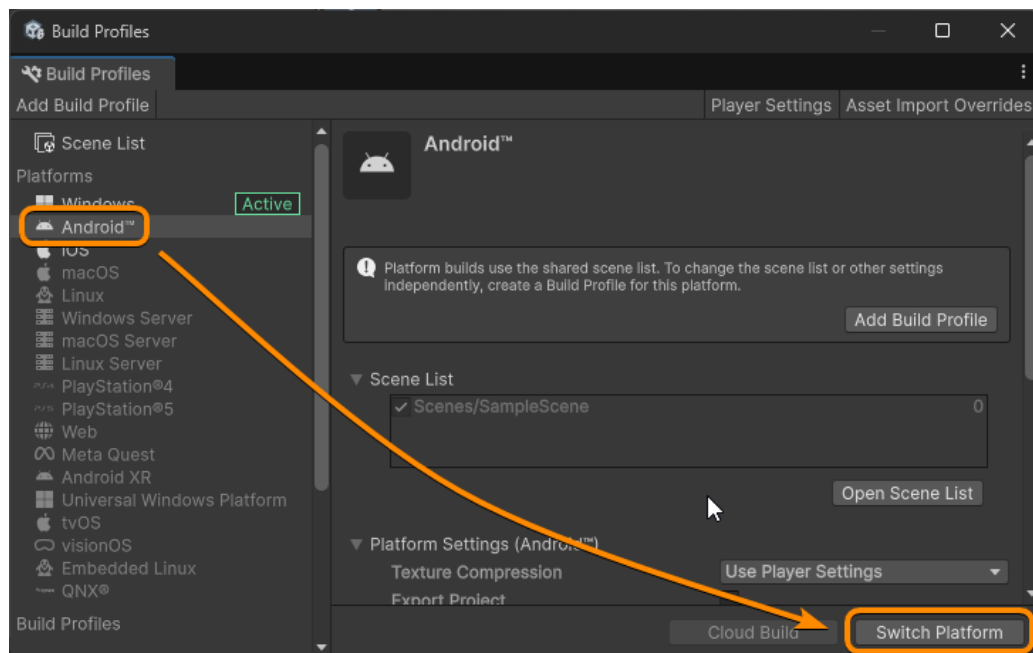


Рисунок 31 – Алгоритм вибору платформи Android

3) Під'єднати мобільний пристрій до комп'ютера через USB провід та дозволити налагодження USB (рис. 32). Прапорець «Завжди дозволяти з цього комп'ютера» дає можливість в наступні рази під'єднувати пристрій до комп'ютера без запиту на дозвіл (увімкнення за бажанням студента).

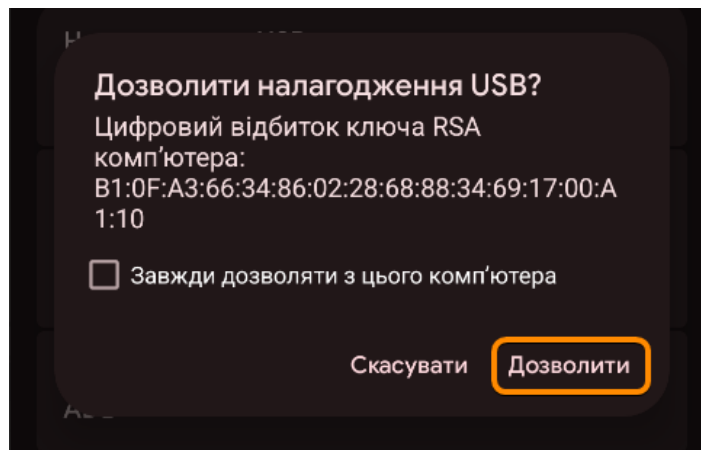


Рисунок 32 – Приклад запиту на дозвіл налагодження USB

4) Знайти параметр Run Device, обрати власний мобільний пристрій, що pojawíться в списку (якщо ще не появи́вся, натисніть на кнопку Refresh), та натисніть на кнопку Build And Run (рис. 33). Також необхідно обрати місце, де буде збережено збірку.

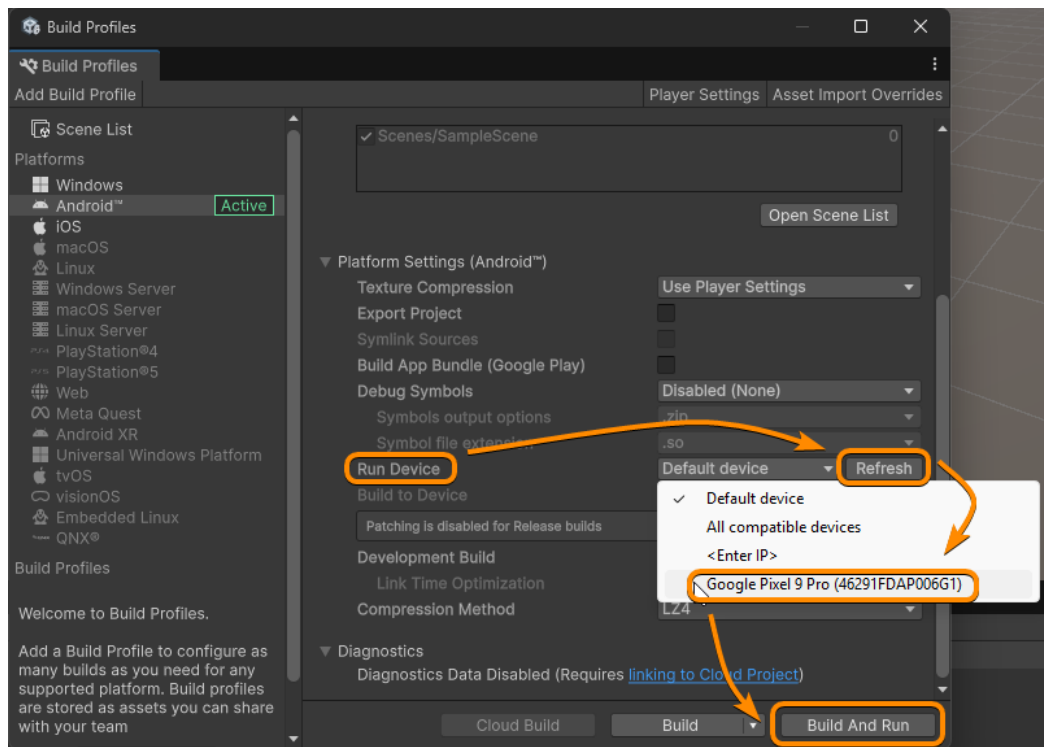


Рисунок 33 – Приклад запиту на дозвіл налагодження USB

5) Зачекати певний час, поки збірка створюється (в перший раз це може зайняти до 10 хвилин). В кінці створення проєкт запуститься на пристрої автоматично.

В разі коректного виконання вищевказаних вказівок, проєкт відкриє камеру мобільного пристрою, розпізнаватиме поверхні в просторі та створюватиме площини в режимі реального часу, а також дасть можливість створювати об'єкти в просторі. Приклад успішної роботи проєкту представлено на рисунку 34.



Рисунок 34 – Скріншот із запущеного проєкту безпосередньо з мобільного пристрою (зі звичайною візуалізацією поверхонь)

Поверхні зі звичайними налаштуваннями можуть створюватись таким чином, що відбувається накладання однієї площини на іншу, від чого може бути не зрозуміло яку саме поверхню в просторі вони відображають. На сторінці 9 методичних рекомендацій вказано алгоритм зміни візуального відображення поверхонь. Налаштування дадуть можливість отримати результат, представлений на рисунку 35.



Рисунок 35 – Скріншот із запущеного проєкту з налаштованими поверхнями (прозора «заливка», напівпрозорий контур)

Зміст роботи

Завдання 1. Створити проєкт з підтримкою технологій AR:

- додати в проєкт пакунки для роботи з AR;
- зробити налаштування проєкту для коректної роботи на мобільних пристроях, що працюють на ОС Android;
- побудувати сцену з базовими об'єктами для роботи AR;

Завдання 2. Реалізувати додаток з розміщення об'єктів в просторі:

- додати можливість розміщувати 5+ видів 3D моделей;
- мінімум одна модель повинна мати анімацію руху (на місці або по невеликій області);
- реалізувати інтерфейс, через який користувач може вибрати об'єкт для розміщення;
- дозволяється використовувати готові та власні 3D моделі;

Завдання 3. Налаштувати смартфон на ОС Android та запустити збірку проєкту на пристрої:

- завдання вважається виконаним у разі успішного запуску додатку на смартфоні, коректного розпізнавання та відображення поверхонь, а також успішного розміщення об'єктів в просторі.

Додатково. Реалізувати додаток з керованими персонажами:

- додати можливість розмістити 3 персонажа;
- кожен персонаж може переміщуватись по площині, на якій він був розміщений;
- реалізувати інтерфейс, через який користувач може вибрати персонажа для керування в поточний момент часу.

Контрольні питання:

1. Що таке AR?
2. Що таке XR Plugin Management?
3. Який компонент відповідає за розпізнавання та відображення поверхонь в просторі?
4. Які параметри натискання потрібні для створення об'єкту на поверхні?
5. Який компонент відповідає за пошук поверхонь для розміщення об'єктів в просторі?
6. Як знайти поверхню, на яку потрібно поставити об'єкт після натискання?
7. Що використовується в мобільному пристрої для реалізації AR?

8. Який графічний API необхідно прибрати з налаштувань для коректної роботи AR проєкту на мобільному пристрої?
9. Як вмикається налагодження мобільного пристрою через USB?
10. Які компоненти візуалізації поверхонь можна налаштовувати?