МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Львівський національний університет імені Івана Франка

Факультет електроніки та комп'ютерних технологій

Кафедра системного проектування

Допустити до захисту
Завідувач кафедри
_____ доц. Шувар Р.Я.
«___» _____2023 р.

Кваліфікаційна робота

Бакалавр
(освітній ступінь)

Використання AR технологій для розробки віртуального 3D гіду по місту

Виконав:

студент групи ФЕП – 41

спеціальності 121 – Інженерія програмного забезпечення

_____ **Крикун Р.В.**

Науковий керівник:

_____ **ас. Каськун О.Д.**

«____» _____2023 р.

Рецензент:

_____ **проф. Свелеба С.А.**

Львів 2023

АНОТАЦІЯ

Метод дипломної роботи є розробка мобільного застосунку, який дозволить користувачам користуватись ним як гідом по місту. Особливістю програми стане наявність технологій доповненої реальності, що покращить досвід взаємодії та загальне враження під час подорожі.

Під час виконання дипломної роботи було порівняно різноманітні туристичні програми, які використовують AR технології для туризму, проаналізовано їхні плюси та мінуси. Досліджено інструменти для створення мобільних програм, які підтримують доповнену реальність. Описано можливості рушія Unity в поєднанні з мовою програмування C# для створення мобільних програм.

Результатом дослідження та розробки стало створення мобільного застосунку під систему Android, спеціально призначений для туристичних потреб.

ABSTRACT

The purpose of the diploma work is to develop mobile application that allow users to use it as a city guide. The application’s major feature is the usage of augmented reality technologies, which enhance the interaction experience and overall impression during the trip.

During the work on the diploma, different tourism applications that use augmented reality technologies for tourism were compared, and their pros and cons were analyzed. Tools with the support of augmented reality for mobile development were researched. The capabilities of the Unity engine in combination with the C# programming language for creating mobile applications were described.

The result of the research and development was the creation of a mobile application for the Android system, specially designed for tourism needs.

ЗМІСТ

ВСТУП 5
РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА 7
1.1. Сучасний стан та перспективи розвитку туризму в Україні 7
1.2. Аналіз існуючих рішень 8
1.3. Визначення цілей програми та платформи для розробки 10
РОЗДІЛ 2. ОПИС ВИКОРИСТАНИХ ПРОГРАМНИХ ЗАСОБІВ 15
2.1. Використані засоби та технології 15
2.2. Інструменти та технології Unity, Unity Hub, Unity Editor та Unity Asset Store 15
2.3. Інтегроване середовище розробки Visual Studio Community 2022 .. 18
2.4. Мова програмування C# 20
2.5. Canvas в Unity 22
2.6. Build and Run та диспетчер пристроїв Android Device Monitor 25
2.7. Набір API та SDK Google Maps Platform 27
2.8. Набір AR інструментів AR Foundation 28
2.9. 3D редактор Blender 31
РОЗДІЛ 3. РОЗРОБКА ЗАСТОСУНКУ 32
3.1. Проектування 32
3.2. Створення графічного інтерфейсу програми, компонента RectTransform 33
3.3. Розгортка програми на Android 34
3.4. Скрипти та API, клас MonoBehaviour 37
3.5. Створення програми 40
3.6. Розробка AR складової 51
РОЗДІЛ 4. ДЕМОНСТРАЦІЯ РОБОТИ ЗАСТОСУНКУ 55
4.1. Аналіз розробленої програми 55
4.2. Застосування програми 56
ВИСНОВКИ 61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 62
ДОДАТОК А 65

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

LBS	Application Programming Interface (укр. Служба, заснована на місцезнаходженні) – тип інформаційних послуг заснований на визначенні поточного місця розташування мобільного телефону користувача.
AR	Augmented Reality (укр. Доповнена реальність) – коли реальні об’єкти інтегруються у віртуальне середовище.
API	Application Programming Interface (укр. Прикладний програмний інтерфейс) - це набір чітко визначених методів для взаємодії різних компонентів програми.
UI	User Interface (укр. Інтерфейс користувача).
SK	Software Development Kit (укр. Комплекс для розробки ПЗ) – набір засобів, утиліт та документації для створення програм за визначеною технологією.
GUI	Graphical User Interface (укр. Графічний інтерфейс користувача) – тип інтерфейсу, побудований на взаємодії користувача з пристроєм через графічні зорбраження.
Фреймворк	Framework (укр. каркас, стуктура) – інфраструктура програмних рішень, що полегшує розробку складних систем.
Асет, асети	Asset (укр. об'єкт) – відображення любого засобу, що може використовуватись в програмі.

ВСТУП

Все більше людей подорожує по світу. Подорожування стало невід'ємною частиною життя багатьох людей. Близько одного мільярда людей подорожували в 2022 році за даними statista [1]. Хтось подорожує заради відпочинку, хтось заради навчання або бізнесу. Також багато людей відкривають для себе нові культури і традиції через подорожі в інші міста, регіони своєї країни, або до іноземних держав.

Опинившись в незнайомому місці люди стараються зібрати якомога більше інформації про те, що їх оточує. Це напряму допоможе їм в орієнтуванні по місту чи на місцевості, а також збагатить їхні знання про культуру та історію цього місця.

Існує багато способів знаходження інформації про якесь місце або пам'ятку. Це можуть бути таблички або буклети, які розкажуть про те, де туристи знаходяться та про те, що вони бачать перед своїми очами. Також є гіди, або екскурсоводи – люди, які проводять екскурсії та можуть ввести в історію конкретного місця чи об'єкту.

З розвитком Інтернету та мобільних телефонів шукати інформацію стало зручніше та простіше. Використовуючи мобільний телефон з доступом до мережі Інтернет туристи мають доступ до величезної кількості інформації про те, що їх оточує або цікавить. Технології LBS, така як GPS допомагає визначати, де ви є. Застосування технологій доповненої реальності поглиблює досвід взаємодії з історією конкретних місць та дозволяє отримати нові враження від подорожі.

Для зручності туристів були розроблені спеціальні програми, які допомагають в орієнтуванні в незнайомому місці, в пошуку готелю,

ресторану тощо. Проте не всі програми є безкоштовними. Деякі мають невелику базу даних про місця, або не мають про них згадок взагалі. Також наявність реклами відштовхує деяких користувачів.

В дипломній роботі було досліджено мобільні програми для туризму. Було розглянуто різноманітні програми та особливу увагу було приділено тим, які використовують AR технології.

Було досліджено їх функціонал, проаналізовано можливості таких програм. Визначено основні вимоги та потреби туристів від таких програм.

Мета дипломної роботи є розробка мобільного застосунку, який дозволить користувачам використовувати його як путівник(гід) по місту.

В ході роботи над дипломною роботою було досліджено аналогічні програми для мобільних пристроїв. Було розглянуто основні концепції та технології, що використовуються для розробки AR програм та описано технології для створення таких програм.

В дипломній роботі було розроблено мобільний застосунок під операційну систему Android за допомогою Unity, проведено порівняння Unity з альтеративними варіантами розробки мобільних застосунків під систему Android, описано способи розробки мобільних програм з застосування AR технологій, висвітлено можливі шляхи для розвитки даного застосунку.

РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА

1.1. Сучасний стан та перспективи розвитку туризму в Україні

Сучасний стан туризму в Україні відображає позитивні тенденції та перспективи подальшого розвитку. Країна має багатий культурний, історичний та природний потенціал, що приваблює як внутрішніх, так і закордонних туристів.

Туристична сфера в Україні стикається з рядом проблем та викликів. Туристична інфраструктура може бути недостатньо розвинена в деяких регіонах України. Це може проявлятись в обмеженій кількості готелів, ресторанів та інших туристичних послуг, так і в обмеженій кількості транспортних зв’язків або неефективній громадській транспортній системі.

Проте, найважливішою перепоною для розвитку туризму в Україні є безпека. Туристи, які вирішили відвідати Україну, повинні бути обережними та дотримуватися всіх офіційних правил та порад щодо безпеки.

Не менш важливою проблемою з туризмом в Україні є погана інформаційна доступність. Інформація про місця, маршрути та інші релевантні деталі часто можуть бути не завжди доступними або недостатньо розповсюдженими, особливо щодо менш відомих місць.

Однак, збільшення інформаційної доступності має потенціал значно підвищити привабливість України для туристів та сприяти сталому розвитку туризму у країні. Цього можна досягнути шляхом покращення інформаційних систем готелів чи ресторанів та збільшенням точної інформації про туристичні об’єкти, відомі місця, маршрути тощо.

Використовуючи потенціал AR технологій можна покращити туристичний досвід. Застосування AR технологій може допомогти створити

віртуальні туристичні гіди, які будуть надавати додаткову інформацію про пам'ятки, історію та культуру країни. Це може зробити подорожі більш інтерактивними для туристів.

1.2. Аналіз існуючих рішень

Програма для туризму – загальна назва для ряду програм, які орієнтовані на туристів. Вони присутні на всіх популярних платформах та в Інтернеті. Такі програми пропонують планувати, організовувати та насолоджуватись подорожами.

Існує широкий спектр типів програм для туризму, серед яких можна виділити:

1. Мапи та навігаційні програми. Програми, що надають детальні карти, маршрутизацію та навігацію. Серед найвідоміших можна виділити Google Maps та Apple Maps
2. Пошукові додатки, які надають інформацію про готелі, ресторани, визначні місця та багато іншого. Прикладами є TripAdvisor та Booking.com.
3. Мовні додатки та перекладачі по типу Google Translate чи Duolingo, які дозволяють зменшити мовні бар’єри між туристами та місцевими жителями.
4. Гіди та аудіопутівники, що фокусують на надаванні інформації про культурні визначні місця, історію конкретних місць. Прикладом такої програми може слугувати Lonely Planet Guides.

Програм для туризму існує безліч, залежно від потреб туристів. Проте нас цікавить останній тип програм, а саме гіди. Існує багато програм, які можуть працювати як гіди, але ними не є. До прикладу, Google Maps може

працювати як гід по місту, оскільки він містить в собі коротку інформацію про визначні місця, відгуки та місцезнаходження на карті.

Google Maps можна вважати програмою-гідом. Якщо ж дивитися на програми, які спеціально розроблені як гіди, то їх також існує певна кількість. Серйозними мінусами таких програм є або їх мала база даних, що не надає інформації про маловідомим місцям, або взагалі відсутність підтримки в деяких країнах.

Було проведено дослідження програм в Google Play Store. Ввівши в пошуку AR tourism буде показана велика кількість додатків, але в своїй більшості вони надають доступ до бази даних об’єктів, які можна розмістити через камеру смартфону. Це можуть бути об’єкти фурнітури, книги, роботи тощо. Такі програми не відносяться до туризму напряму.

З програм, які пов’язані з туризмом, можна виділити FlippAR Go (рис. 1.1) для Android та iOS, та AR Trails: Traveling with AR (рис. 1.2) для Android та iOS.

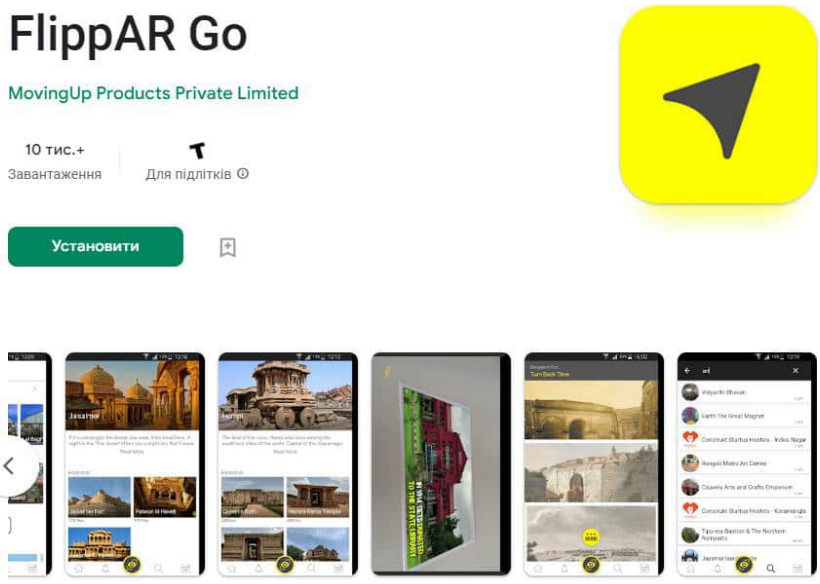

Рис. 1.1 – Зображення програми FlippAR Go в Google Play

Перша програма містить в собі об’єкти з всього світу, такі як Помпеї або Ейфелева вежа. Програма дозволяє переглянути коротку інформацію, подивитись відео про об’єкт, прослухати історію цього об’єкту та переглянути відео про те, яким цей об’єкт був в минулому. Програма дозволяє перегляд цих місць, та показує відстань до них від вашого місцезнаходження. Програма має в собі стрічку новин.

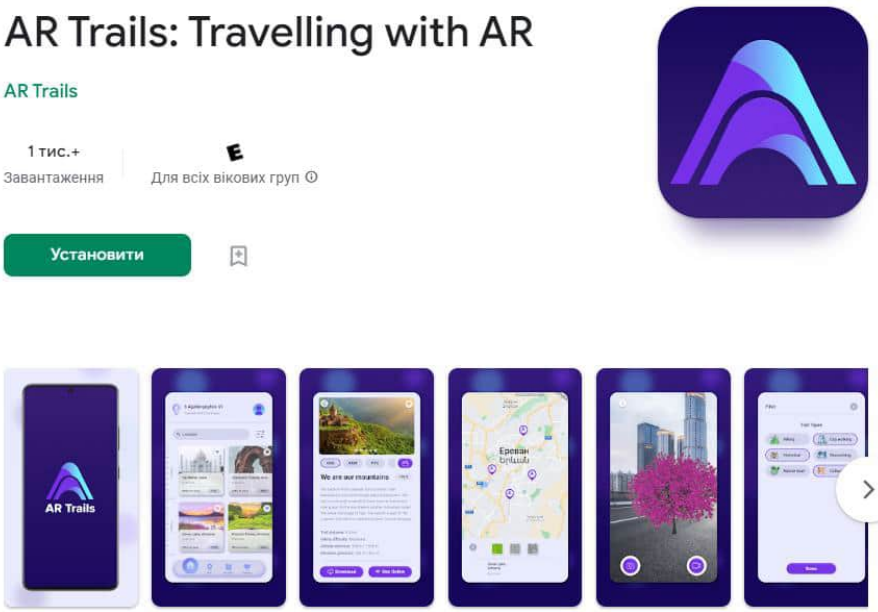

Рис. 1.2 – Зображення програми AR Trails: Traveling with AR в Google Play

Ця програма містить в собі об’єкти з наступних країн: Греція, Угорщина, Вірменія та США. Це можуть бути як ботанічні сади, так і різноманітні історичні об’єкти. Програма дозволяє перегляд цих місць, та показує відстань від вашого місцезнаходження до об’єкту.

1.3. Визначення цілей програми та платформи для розробки

Для початку потрібно визначити, щоб нам потрібно зробити в програмі. Метою розробки є створення програми, де користувач зможе

переглядати інформацію про визначні місця Львова, бачити коротку інформацію про них, їхнє розташування на мапі та можливість побачити деякі з цих визначним місць з іншого боку через використання AR технологій за допомогою камери його пристрою.

Оскільки планується, що користувач зможе взаємодіяти з камерою, не доцільно розробляти програму під не мобільні платформи. Також потрібно визначити операційну систему, на якій буде розроблятися програма. На сьогоднішній день в сегменті мобільних пристроїв переважають дві системи: Android та iOS.

Далі потрібно визначити, на яку мобільну платформу буде розроблятися програма. На сьогоднішній день в сегменті мобільних пристроїв домінує дві системи: Android та iOS. Станом на 4 квартал 2022 Android займає 71.8% мобільного ринку, коли частка iOS складає 27.6% [2].

Доцільно вибрати Android, оскільки ця система є більш поширеною. Для розповсюдження застосунку Android є вигіднішим для розробників, оскільки поширення програми через Google Play Store чи App Store у випадках Android та iOS відповідно є платним: 25 доларів США в випадку Android є разовим платежем, коли на iOS щорічна плата становить 99 доларів США.

Умова використання AR накладає ряд обмежень на розробку. Існує декілька підходів для розробки AR додатків для Android, серед яких є Android Studio, Unity та Unreal Engine. Android Studio та Unreal Engine використовують SDK AR Core від Google, коли Unity використовує власний фреймворк AR Foundation, який включає в себе різні AR інструменти, серед яких є AR Core [3].

Було обрано Unity з AR Foundation через те, що Unity надає більшу гнучкість при розробці AR застосунків в порівнянні з Android Studio і ARCore. Також не потрібно забувати, що AR це тільки одна частина програми. В частині написання коду Unity виграє в Unreal Engine через легкість мови програмування для більшості користувачів: Unity використовує C#, коли Unreal Engine використовує C++.

Підсумовуючи, Unity пропонує більш просунуті інструменти в AR сегменті, ніж Android Studio з ARCore, та рівні з Unreal Engine з їх надбудовою над ARCore. Але в частині зручності для новачків Unity переважає над Unreal Engine.

Не менш важливою складовою успішної мобільної програми є дизайн. Було проаналізовано сучасні тренди в дизайні мобільних програм.

Говорячи про тренди, важливо згадати, що за останні роки жести по екрану смартфона, такі як свапи, тапи та інші стали тим, що люди роблять щоденно і рефлекторно [12]. Також серед вагомих трендів для дизайну є наявність Dark mode в програмі. Dark mode, або темний режим – це підхід до дизайну програм, де чорний текст на білому фоні інвертований.

Не менш важливими сучасними трендами в дизайні є мінімалізм інтерфейсу та анімації. Лаконічний інтерфейс дозволяє зробити легку, з сучасним виглядом та просту для використання мобільну програму. Анімації та супровідні зображення допомагають користувачам орієнтуватись в інтерфейсі та роблять програму більш виразною.

На мобільних пристроях існує два види орієнтації пристрою: Landscape та Portrait, або альбомна та портретна орієнтації. Орієнтація пристрою залежить від того, як його тримають. Портретний режим використовується для читання тексту чи гортання соціальних мереж.

Альбомний режим, в свою чергу, підходить для перегляду відео чи грання в відеоігри.

Розширення екрану є важливою складовою при розробці мобільного застосунку. Оскільки “стандартними” розширеннями для пристрої можна назвати співвідношення сторін екрану 16:9 та 4:3, мобільні пристрої використовують співвідношення 16:9, проте останнім часом це співвідношення не є стандартним, коли все більше виробників смартфонів використовують співвідношення, яке може дорівнювати 18:9, 17.5:9 та інші.

Проаналізувавши сучасні тренди в дизайні та особливості екрану на мобільних пристроях, можна вивести те, як має виглядати інтерфейс програми:

1. Програма повинна мати органічно інтегровані жести, які дозволять легко взаємодіяти з програмою.
2. Програма повинна мати простий і зрозумілий інтерфейс, що збільшить привабливість програми та зробить навігацію в програмі зрозумілою для кінцевого користувача.
3. Програма повинна мати підтримку темного режиму, що загалом покращить вигляд програми для користувача.
4. Програма повинна враховувати особливості екрану та автоматично підлаштовуватись під розширення та співвідношення сторін екрану пристрою користувача.
5. Програма повинна використовувати портретну орієнтацію, оскільки користувач буде орієнтуватись по програмі як по сучасним соціальним мережам.

Визначивши платформу та інструменти для розробки, а також продумавши інтерфейс програми, можна вивести те, як буде виглядати кінцевий продукт.

Метою розробки є створення програми на Android за допомогою Unity. Програма повинна мати зручний, сучасний інтерфейс та показувати визначні пам’ятки Львова, надавати коротку інформацію про них, показувати їх розташування та розташування користувача на карті. Програма має мати підтримку AR технологій за допомогою AR Foundation, що дозволить побачити ці місця з іншого ракурсу.

РОЗДІЛ 2. ОПИС ВИКОРИСТАНИХ ПРОГРАМНИХ ЗАСОБІВ

2.1. Використані засоби та технології

При розробці даного застосунку були використані такі технології та програмі засоби:

1. Персональний комп’ютер з операційною системою Windows 11
2. Інструменти та технології Unity, Unity Hub, Unity Asset Store
3. Середовище розробки Visual Studio 2022
4. Мова програмування C#
5. Диспетчер пристроїв Android Device Monitor
6. Набір API та SDK Google Maps Platform
7. Набір AR інструментів AR Foundation
8. 3D редактор blender.
9. Android сумісні пристрої

2.2. Інструменти та технології Unity, Unity Hub, Unity Editor та Unity Asset Store

Unity – це сучасна платформа для розробки ігор та створення інтерактивного досвіду взаємодії [5]. Перш за все, Unity – це ігровий рушій і платформа розробки. Рушій надає доступ до широкого спектру технології для створення 2D та 3D ігор, застосунків з віртуальною та доповненою реальність (VR/AR), симуляторів та іншого. Створені за допомогою Unity програми працюють на настільних комп'ютерних системах, мобільних пристроях та гральних консолях у дво- та тривимірній графіці, та на пристроях віртуальної чи доповненої реальності. Загалом програми, що написані на Unity можуть запускатись на 25 платформах, найпопулярнішими з яких є Windows, Android, iOS, Mac та Linux.

Unity використовується не тільки в галузі відеоігор. Рушій активно використовується в галузях кіно, автомобілебудування, архітектура, інженерія та будівництво.

Для роботи з Unity потрібно середовище розробки. Для цього було розробленого Unity Editor, який можна запускати напряму, або через Unity Hub. Unity Hub – це програма для керування та запуску різних версій Unity, а також для керування проектами та доступу до навчальних ресурсів (рис. 2.1). Дозволяє легко переключатися між різними версіями рушія Unity, керувати розташуванням проектів та програм.

Для створення нового проекту через Unity Hub потрібно перейти на вкладку Projects та натиснути кнопку New Project. Після цього відкриється нове вікно, де можна обрати шаблон проекту, версію Unity, назву та розташування проекту в файловій системі. Деякі шаблони не доступні без завантаження їх з Інтернету.

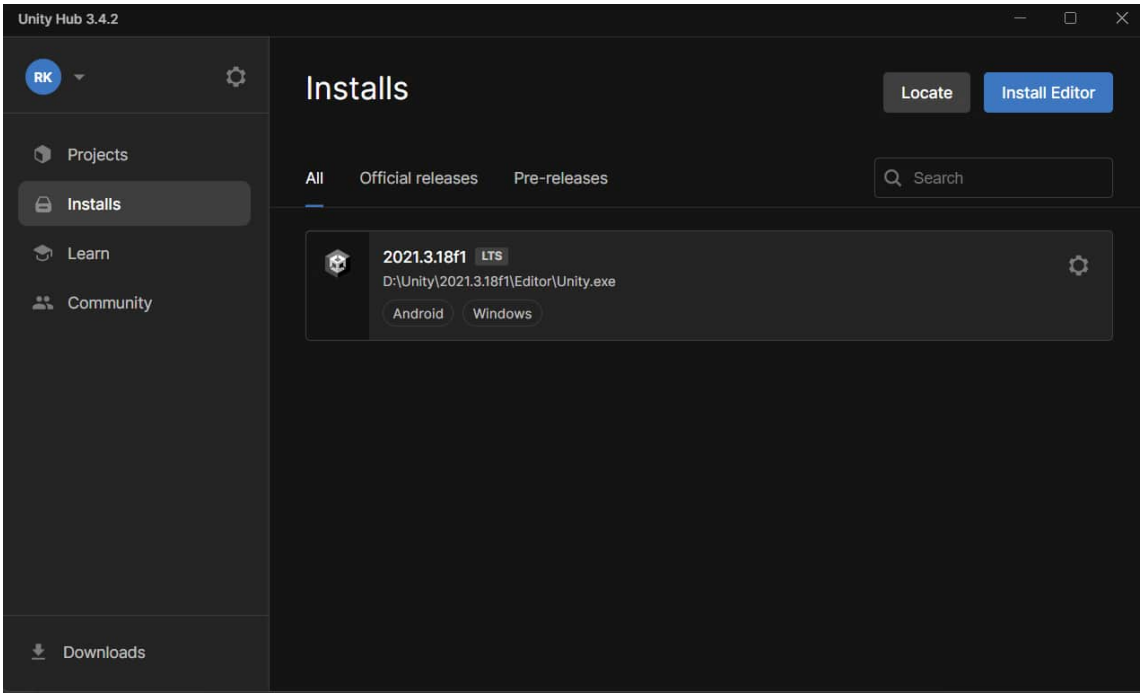

Рис. 2.1 – Інтерфейс програми Unity Hub

Unity Editor – це основний інструмент для розробки і керування проектів на Unity (рис. 2.2). Це GUI, який надає доступ до великого набору інструментів і технологій для розробки ігор та програм для взаємодії.

Включає в себе різноманіття вікон та панелей, серед яких варто виділити перегляд сцени та перегляд виду гри. Ці два режими надзвичайно важливі при розробці, оскільки дозволяють бачити інтерфейс гри або програми, яка розробляється. Зображення з сцени є тим, яке бачить розробник, коли зображення з гри є зображенням фінального результату і вид того, що буде бачити перед собою користувач.

Панель ієрархії показує список об’єктів в сцені та як різні об’єкти в сцені взаємо відносяться один з одним. Панель інспектора показує інформацію про вибраний об’єкт з панелі ієрархії. Ця інформація – це різні компоненти, які представляють собою відносне розташування, колір, зображення, скрипти та різні модулі, які можуть знадобитися цьому об’єкту.

Панель проекту показує список всіх асетів в проекті. Це можуть бути скрипти, зображення, текстури, моделі, сторонні модулі тощо. Панель консолі дозволяє відображати інформацію з скриптів або повідомлення про помилки, якщо такі стаються.

Асети можна створювати в редакторі Unity, або імпортувати з зовнішніх джерел. До прикладу, 3D модель можна зробити в спеціальному 3D редакторі, такому як Blender чи Maya, і потім імпортувати як асет в Unity.

Асетами можна керувати через панель проекту, де перелічені всі асети в вибраному проекті. Асети можна організовувати по папкам, шукати по назві і переглядати їх інформацію через панель інспектора. Також асети можна призначати об’єктам в сцені і редагувати через панель інспектора.

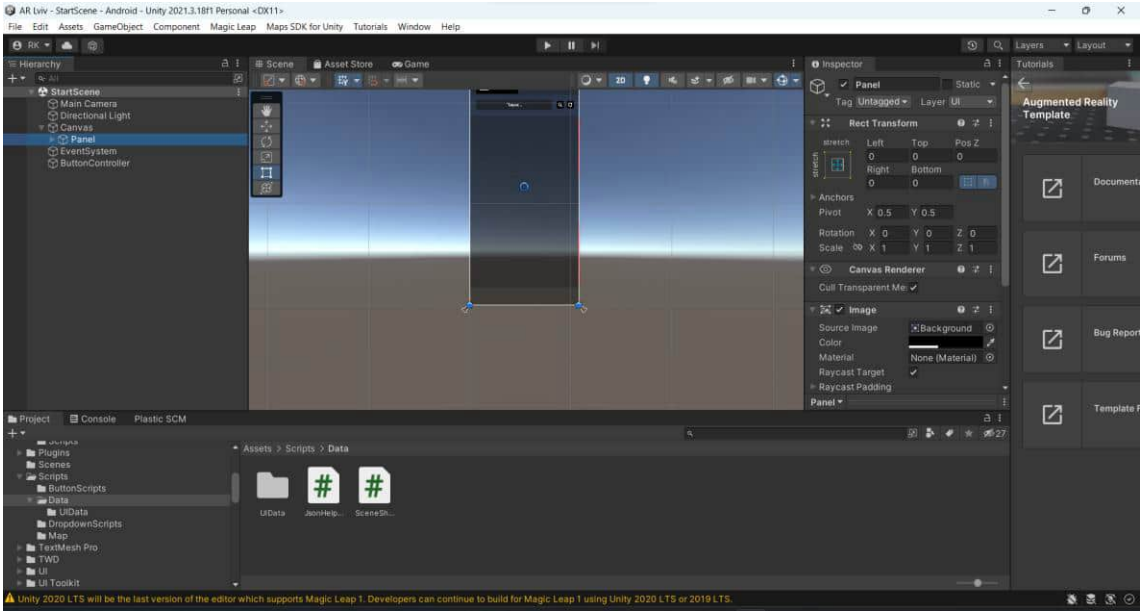

Рис. 2.2 – Інтерфейс програми Unity Editor

Unity має власний маркетплейс асетів, де розробники можуть ділитися асетами один з одним. Вони можуть безкоштовно поширювати, продавати та купувати асети, які потім можна використовувати в Unity. Unity Asset Store інтегрований з Unity Editor, що робить пошук та імпортування асетів в проект простим ділом. Також слугує як хаб для навчальних ресурсів та форуму користувачів.

2.3. Інтегроване середовище розробки Visual Studio Community 2022

Visual Studio – це інтегроване середовище розробки, розроблене компанією Microsoft (рис. 2.3) [6]. Використовується для створення широкого ряду додатків, включаючи комп’ютерні, веб, мобільні та ігрові програми. Представляє собою повне середовище розробки, яке містить в собі редактору коду, дебаггер, юніт-тести, засоби для роботи з базами даних та інші інструменти, які спрощують цикл розробки програмного забезпечення та життя програмістів.

Редактор коду знаходиться по центру екрану. Дозволяє писати, редагувати, копіювати та видаляти код.

Верхня панель містить в собі дебагу, запуску програми, повернення змін, збереження коду, збірки рішення та багато інших. Також верхня панель надає доступ до налаштувань самої Visual Studio, налаштувань вигляду редактора та різноманітних інструментів, серед яких є взаємодія з Git, аналіз коду та інші.

З правого боку знаходяться панель взаємодії з віддаленим репозиторієм та панель провідника, яка показує відкритий проект та файли в ньому. Також під час роботи програми панель справа замінюється панеллю продуктивності, де можна побачити використання ОЗУ та процесора.

Нижня панель містить в собі вивід програми, консоль PowerShell та список помилок, якщо такі є. Також нижня панель має в собі оглядач тестів, де можна переглядати стан проходження тестів.

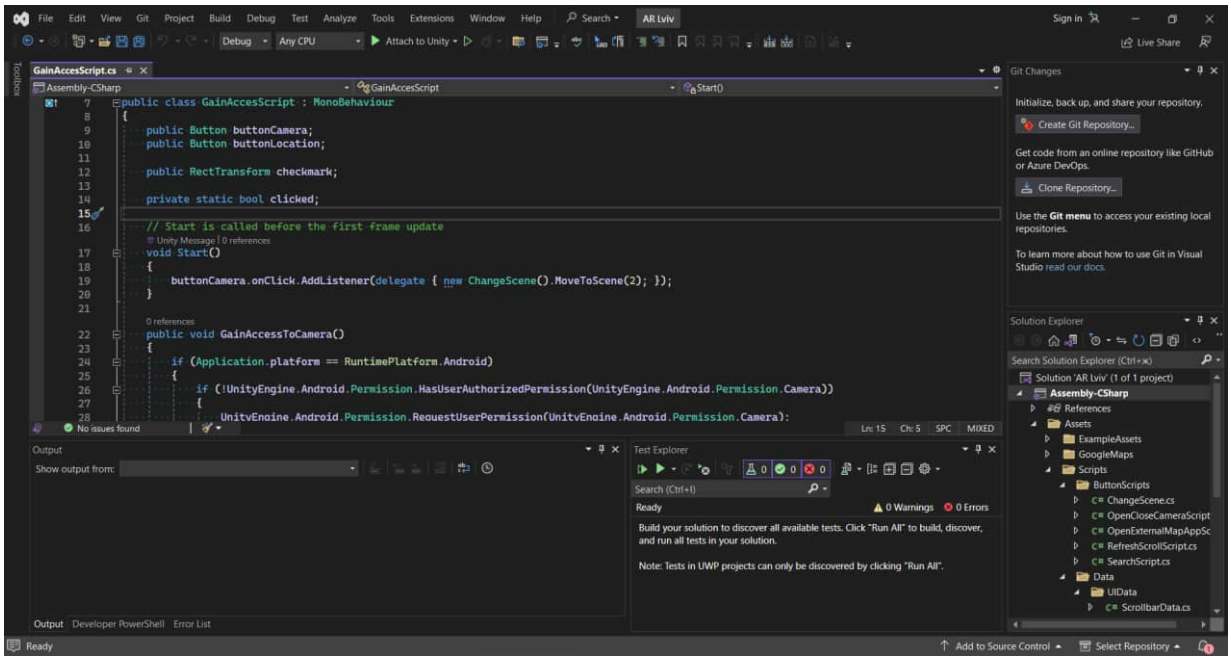

Рис. 2.3 – Інтерфейс середовища розробки Visual Studio Community 2022

При написанні коду програмістам допомагає IntelliSence – набір засобів для спрощення написання коду: автоматично дописує назву функції або методу при введенні початкової букви; використовується для доступу до метаданих конкретного метода або класу; допомагає при перевантаженні методів, надаючи доступ до інформації про сигнатури методів.

2.4. Мова програмування C#

C# - це сучасна об’єктно орієнтована мова програмування, яка широко застосовується для розробки комп’ютерних, веб сайтів та інших типів застосунків [7, 8]. Представлена в ранніх 2000-их як частина .NET платформи.

C# є однією з найпопулярніших мов програмування. Станом на травень 2023 року C# є 5 по популярності мовою програмування за індексом TIOBE [9]. Мова дозволяє розробляти додатки під більшість популярних операційних систем, таких як Windows, Linux, Android та iOS. Спільнота розробників має важливу роль в вдосконаленні C#, оскільки ця мова є з відкритим кодом і програмісти можуть пропонувати свої варіанти покращень для мови, але останнє слово завжди стоїть за оригінальним розробником – Microsoft.

C# - це строго типізована мова програмування. Мова має декілька особливостей, які роблять програмування на C# простішим. Наприклад:

1. Об’єктно-орієнтоване програмування. Перш за все, C# це об’єктно-орієнтована мова програмування, що дозволяє розробникам створювати і використовувати об’єкти, які посилаються на real-life концепти та сутності. Це полегшує керування коду, зменшується повторне використання коду та дозволяє створювати більш гнучкі програми.

2. Автоматичне керування пам’яттю. C# включає в собі збирач сміття, який відповідає за автоматичне виділення та звільнення пам’яті, що звільняє розробника від ручного керування пам’яттю. Хоч збирач сміття працює автоматично, розробник всеодно може виділяти пам’ять вручну, якщо це потрібно. Для ручного виділення пам’яті розробник може використати “небезпечні” вказівники та ключові слова, хоча не рекомендується цього робити і краще покладатися на автоматичне виділення пам’яті.
3. Безпека типів. Оскільки C# - це строго типізована мова програмування, це означає, що кожен об’єкт повинен мати спеціально визначений тип. Це дозволяє уникати помилок в “рантаймі” програми.
4. Підтримка декількох платформ. Програми написані на C# можна запускати на різних операційних системах та на різних системних архітектурах.
5. Підтримка асинхронного програмування. C# підтримує асинхронне програмування через ключові слова async/await, що дозволяє розробникам створювати програми, які можуть оперувати великим набором даних та працювати доволі швидко, при цьому не займаючи основний потік програми.
6. LINQ. Одна з основних особливостей C#. LINQ, або інтегровані в мову запити – це набір функцій, які дозволяють розробникам виконувати комплексі запити даних. За допомогою цього програмісти можуть отримувати запити з баз даних, XML документів чи колекцій. LINQ є частиною мови C# і це означає, що запити можна писати напряму в коді C# через різні оператори та вирази. Синтаксис LINQ такий самий, як запити SQL, але він спеціально розроблений для роботи з структурами даних та об’єктами C#.

7. Делегати та події. Ключові функції в C#. Делегати – це вказівники на функції, які можуть викликатися в рантаймі. Делегати корисні, коли в програмі потрібно реагувати на дії з боку користувача або на інші зовнішні події. В свою чергу події – це механізм для обробки повідомлень в програмі. Події зазвичай йдуть з парою методів: один викликає подію, а інший реагує на неї.

C# та Unity тісно пов’язані. C# є основою мовою програмування в Unity, яка використовується для розробки ігор та програм. Unity надає інтегроване середовище розробки, яке включає в собі редактор коду для C#, а також ряд інших функцій, які роблять розробку продуктивнішою.

В Unity C# використовується для визначення поведінки ігрових об’єктів, створень користувацьких компонентів та опрацювання вхідних даних та подій. Всі скрипти в Unity написані на C#.

Також Visual Studio має налаштований IntelliSence для написання коду на C# для Unity, який йде окремим набором і може бути завантажений з Visual Studio Installer.

2.5. Canvas в Unity

Сучасна програма не може обійтися без user-friendly інтерфейсу [10]. Зручність інтерфейсу інколи набагато важливіша за функції, бо якщо користувачі не будуть знати, як користуватися програмою, то вони і не зможуть зрозуміти, як користуватися всіма наявними в програмі функціями. Останнім часом популярність GUI росте і все більше програм пропонують сучасний інтерфейс замість застарілого UI чи консолі.

User-friendly інтерфейс є надзвичайно важливим для успіху мобільного застосунку. Інтерфейс є основним способом взаємодії з програмою. Якщо з інтерфейсом виникають якісь складнощі або

орієнтуватися в програмі стає важко, то більшість користувачів знайдуть такий інтерфейс незручним і, скоріш за все, користувачі перестануть використовувати цю програму і почнуть шукати альтернативи.

Інтерфейс повинен бути простим і інтуїтивно зрозумілим, з чітко визначеними значеннями, іконками та інструкціями. Дизайн повинен бути продуманим, візуально приємним та бути логічним, щоб користувач міг переходити по програмі та використовувати всі її функції.

Unity пропонує широкий набір інструментів для створення інтерфейсу. Серед цих інструментів варто виділити Canvas та UI Components. Існують також інші можливості для створення 2D інтерфейсу, серед яких анімації, системи подій та розгортки і спрайти.

1. Canvas – це основний контейнер для 2D UI елементів в Unity. Надає систему координат і систему розгортки. Працює як шар між UI елементами та камерою Unity, та дозволяє розміщувати їх незалежно в ігровому світі. Canvas має власну систему координат, побудовану на пікселях з початком в (0; 0) з лівого верхнього кута екрану. Система розгортки дозволяє налаштовувати позиціювання та відносний розмір елементів відносно один одного та до Canvas. Також дозволяє контролювати рендер UI елементів, їх порядок та інші налаштування.
2. UI elements – це основні блоки, з яких складається будь який 2D дизайн любої гри чи програми. Ці елементи представляють собою графічні об’єкти, які можуть бути додані до Canvas та для створення взаємодії з інтерфейсом. Приклади:
    2.1. Text – елемент, який відповідає за показ тексту на екрані. Має різні параметри, серед яких шрифт, розмір шрифту, колір та інші.

2.2. Image – елемент, який відповідає за вивід зображення на екран. Може виводити як асет, так і зображення з інтернету.

2.3. Button – елемент, який дозволяє користувачу робити дії в програмі натискаючи на цей елемент.

2.4. Input Field – елемент, який дозволяє користувачу вводити текст або цифри.

2.5. Scroll View – елемент, який дозволяє користувачу “слайдити” через набір або список елементів.

Також існують і інші елементи, такі як Dropdown чи RawImage, які мають своє призначення в UI.

Unity надає основні можливості для налаштування вигляду UI елементів. Це і налаштування кольору, тексту, розміру, але деякі UI елементи мають в собі компоненту Image, а також існують елементи Image та RawImage. Unity по замовчуванню містить в собі деякі текстури, які можна використати для зображень, але список цих текстур невеликий і може не відповідати всім потребам розробника.

Unity дозволяє призначати елементам Image та RawImage імпортовані як асети зображення. Існує декілька способів, звідки можна взяти зображення для програми. Існує Unity Asset Store, де можна знайти асети на майже любу тему, або можна скачати зображення з Інтернету і імпортувати їх як асети.

Також не всі асети в Unity Asset Store можуть відповідати вимогам розробника, так само як і зображення в Інтернеті. Тоді залишається або замовити зображення, або створити власні. Для останнього варіанту існують графічні редактори.

Існує багато графічних редакторів, серед найвідоміших з яких є Photoshop та Paint. Оскільки Photoshop є платним, не всі можуть дозволити собі щомісячну підписку, а Paint є доволі слабкою програмою для серйозної роботи над зображеннями. Існує багато безкоштовних графічних редакторів, серед яких є можна виділити GIMP [11].

2.6. Build and Run та диспетчер пристроїв Android Device Monitor

Процес розробки та тестування на мобільні пристрої відрізняється від таких на ПК. Перш за все, всі мобільні програми розробляються на ПК. Через різницю в архітектурах між системами та пристроями розробники повинні враховувати ці особливості.

Unity дозволяє розробникам надсилати їх ігри і програми на тестування на мобільні пристрої з операційною системою Android чи iOS. Це можна зробити через опцію Build and Run.

Build and Run можливий через підключення мобільного пристрою до комп'ютера, де запущений Unity Editor. Підключення відбувається напряму через USB кабель. Якщо пристрій не виявлено, то Unity Editor попросить усунути несправності з підключенням. Коли підключення встановлене, Unity Editor надсилає гру чи програму на мобільний пристрій для компілювання.

В випадку Android, якщо встановлюваної програми немає на мобільному пристрої, Android запитає доступ на встановлення додатку з невідомого джерела через USB. Назва встановлюваної програми йде від загальних налаштувань в Unity Editor.

Якщо програма вже встановлена, Unity замінить її останньою версією, надісланою через Build and Run та запустить її (рис. 2.4).

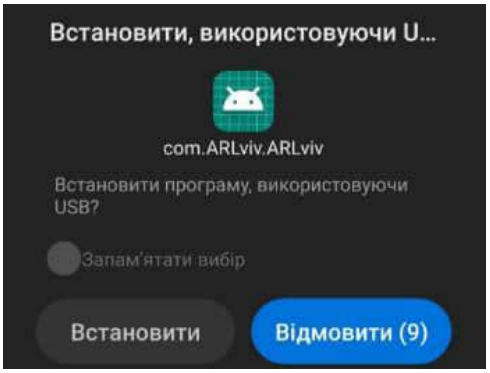

Рис. 2.4 – Зображення з мобільного пристрою при встановленні програми, використовуючи Build and Run

Якщо виникає якась помилка в програмі на ПК, розробники можуть побачити повідомлення про помилку, код самої помилки, текст та де в коді сталась помилка, а також можливі рішення цієї помилки. Якщо якась помилка стається на телефоні, зрозуміти, чому сталась помилка стає важко. Для цього було розроблено утиліту Android Device Monitor, яка використовується для моніторингу і взаємодії з Android пристроєм [13]. Надає GUI для дебагу та тестування застосунків (рис. 2.5).

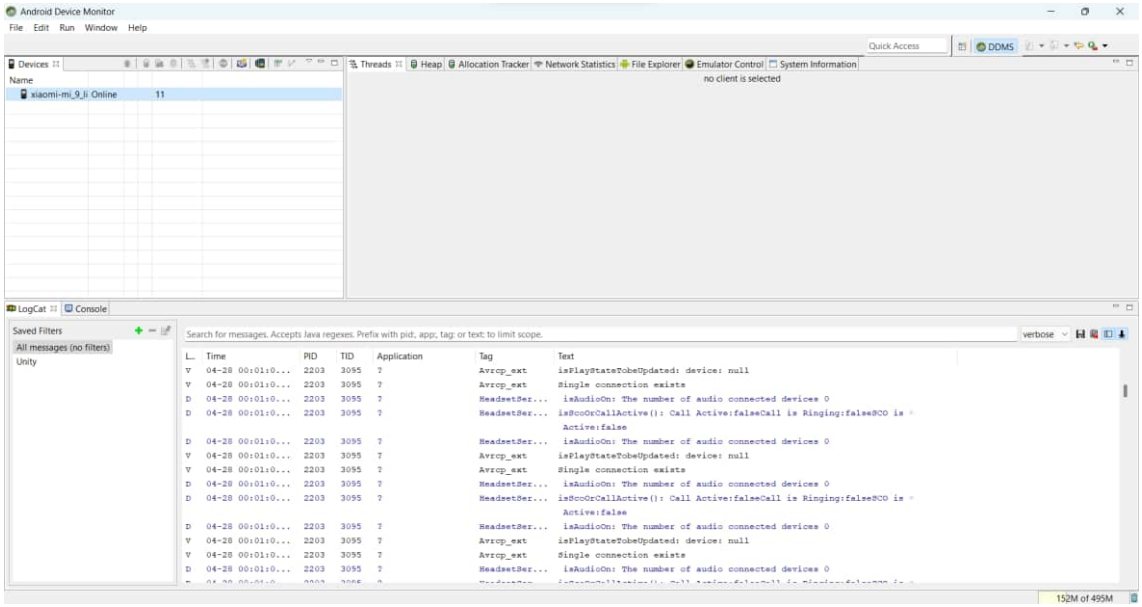

Рис. 2.5 – Інтерфейс програми Android Device Monitor з підключеним Android пристроєм Xiaomi Mi 9 Lite

Програма постійно виводить інформацію з телефону. Це може бути інформація про мережу, використання пам’яті чи продуктивність CPU. Підтримує сортування по ключовим словам, надає файловий провідник для доступу до файлів на пристрої, а також Logcat для виведення системних повідомлень та інформації по дебагу.

Всі повідомлення з пристрою йдуть з певним тегом, що дозволяє проводити сортування повідомлень та виводити тільки потрібну інформацію.

2.7. Набір API та SDK Google Maps Platform

Туристичну програму неможливо уявити без можливостей LBS: визначення місцезнаходження та прокладення шляху. Найвідомішим прикладом LBS є GPS. GPS працює на всіх мобільних пристроях та дозволяє визначати розташування користувача, використовуючи інформацію зі супутників.

Google Maps Platform – це набір API та інструментів, які дозволяють розробникам інтегрувати LBS в їх програми [14]. API включає в собі багато функцій, серед яких є геокодування, пошук місця, прокладення шляхів та інформація про затори в реальному часі. Використовуючи ці API розробники надають користувачам релевантну інформацію в реальному часі про їх місцезнаходження та те, що їх оточує, таке як відомі місця, події, готелі тощо.

Google Maps Platform також надає широкий набір інших інструментів, які дозволяють розробникам налаштовувати карти під себе, використовувати власний стиль, ставити власні маркери та шари.

Google Maps Platform доступний як частина Google Cloud Platform. Для роботи з Google Cloud Platform потрібен акаунт Google Developer.

Google Developer надає безкоштовний період в 90 днів після реєстрації та 200 доларів США як кредит на користування різноманітними API, а також дає ще 300 доларів США на 90 місяців, які будуть зняті в випадку перевищення ліміту звернень до API на 200 доларів США. За місяць безкоштовно можна зробити понад 28000 звернень до API.

Для реєстрації потрібно вказувати персональні дані та дані банківської картки. Після реєстрації Google проводить невелике опитування про те, як ви будете використовувати їх API та наддасть персональний API ключ, який потрібен для звернень до API.

Для використання Google Maps Platform в Unity потрібно імпортувати Google Maps SDK. SDK включає в собі всі необхідні скрипти та різноманітні приклади використання API.

Також можна використовувати Google Maps Platform API напряму через методи Unity, не завантажуючи при цьому сторонні SDK.

2.8. Набір AR інструментів AR Foundation

AR Foundation – фреймворк, спеціально розроблений для розробки доповненої реальності, який дозволяє створити програму один раз і розгортати її на різних мобільних та AR сумісних пристроях.

AR Foundation включає основні функції з ARKit, ARCore, Magic Leap і HoloLens, а також унікальні можливості Unity для розробки надійних додатків, які завжди готові до розгортки. Не всі функції доступні при переході з однієї платформи на іншу. Якщо якась функція доступна в одній, але не доступна в іншій, розробники додають спеціальні механізми, щоб функція була готова до використання пізніше. Коли вона стане доступна, достатньо просто оновити пакети, а не перебудовувати додаток з нуля.

ARKit – набір інструментів від Apple, які дозволяють створювати AR додатки для їх платформ: iOS та iPadOS.

ARCore – це платформа Google для створення доповненої реальності. Деякі з API доступні як для Android, так і для iOS.

Magic Leap – американська компанія, яка випускає власний віртуальний дисплей Magic Leap One, який накладає 3D-згенеровані комп'ютером зображення на об'єкти реального світу шляхом «проекції цифрового світлового поля в око користувача».

HoloLens – окуляри змішаної реальності від Microsoft. Працює схожим чином до Magic Leap.

Для розробки AR застосунків на платформу Android потрібно використовувати ARCore.

ARCore дозволяє телефону взаємодіяти зі світом через використання різних API. ARCore використовує три ключові можливості для інтеграції віртуального контенту з реальним світом через камеру смартфона:

- Відстеження руху дозволяє телефону розуміти і відстежувати своє положення відносно світу;
- Розуміння навколишнього світу дозволяє телефону визначити розмір і розташування всіх типів поверхонь: горизонтальних, кутових, вертикальних, таких як земля, стіни тощо;
- Оцінка рівня освітленості дозволяє телефону оцінювати умови освітлення в навколишньому середовищі.

ARCore підтримується на смартфонах, версія Android на яких вища за 7.0 та на iOS версії 11.0 і вище.

ARCore виконує дві речі: відстежує позицію телефону, коли він рухається та будує своє розуміння реального світу.

Технологія відстеження руху ARCore використовує камеру телефону для виявлення точок інтересу, або ключових точок, які називаються функціями, і відстежує, як ці точки рухаються в часі. Поєднуючи рух цих точок з показаннями інерційних датчиків телефону, ARCore визначає положення та орієнтацію телефону під час його переміщення в просторі.

Розуміння ARCore реального світу дозволяє розміщувати об'єкти, анотації або іншу інформацію таким чином, щоб вона бездоганно інтегрувалася з реальним світом.

ARCore шукає скупчення ключових точок, які лежать на спільних горизонтальних або вертикальних поверхнях, таких як стіл або стіна, і робить ці поверхні доступними для додатку як геометричні площини. ARCore також може визначити межі кожної геометричної площини і зробити цю інформацію доступною для програми.

Оскільки ARCore використовує ключові точки для визначення площин, плоскі поверхні без текстури, такі як біла стіна, можуть бути визначені неправильно.

Площини можуть змінюватися, оскільки ARCore покращує своє розуміння власного положення та оточення. Коли користувач хоче розмістити віртуальний об'єкт, потрібно визначити прив'язку, щоб гарантувати, що ARCore відстежує положення об'єкта в часі.

Той факт, що площини можуть змінюватися, означає, що ARCore може оновлювати положення об'єктів середовища, таких як геометричні площини і точки, з плином часу. Площини і точки - це особливий тип об'єктів, які називаються відстежуваними. Як випливає з назви, це об'єкти,

які ARCore буде відстежувати у часі. Можна прив'язати віртуальні об'єкти до певних відстежуваних об'єктів, щоб гарантувати, що зв'язок між вашим віртуальним об'єктом і відстежуваним об'єктом залишається стабільним, навіть коли пристрій рухається.

2.9. 3D редактор Blender

Оскільки AR розміщує об’єкти в просторі через камеру мобільного пристрою, потрібно мати базу об’єктів для розміщення їх у просторі. Найчастіше цими об’єктами є 3D моделі різноманітних речей, але також це можуть бути анімації чи текст.

Для туристичної програми достатньо 3D моделей. Це можуть бути якісь пам’ятки чи будівлі, які або не збереглись до нашого часу, або мають вигляд, який відрізняється від того, який в них був в минулому.

3D моделі можна знайти безкоштовно в інтернеті, або купити їх на спеціальних сайтах, які продають різноманітні 3D моделі в популярних форматах. Але не завжди в інтернеті є те, що потрібно розробникам. Інколи деяких моделей просто не існує, або вони можуть бути платними. Тоді розробникам залишається створити власні 3D моделі.

Для цього існує багато програм, серед яких є як безкоштовні, так і платні рішення. Серед безкоштовних програм виділяється Blender.

Blender – це безкоштовна, з відкритим кодом програма для створення 3D. Може бути використана для створення анімацій, моделей, візуальних ефектів тощо. Доступна на Windows, Mac та Linux.

Blender надає широкий набір інструментів для створення 3D контенту, серед яких є моделювання, текстурування, такелаж (або створення скелету для руху), анімування, симуляція, рендеринг та композиція.

РОЗДІЛ 3. РОЗРОБКА ЗАСТОСУНКУ

3.1. Проектування

При розробці любого програмного продукту важливо мати вигляд того, як програма буде виглядати в кінці циклу розробки. Це дозволяє сфокусуватись на тому, що важливо і не робити непотрібний функціонал.

Якщо порівнювати Unity з іншими варіантами розробки на мобільні пристрої, наприклад Xamarin, то одна сцена в Unity відповідає одній сторінці в Xamarin.

Для початку потрібно зрозуміти, які сторінки потрібно розробити. Список всіх сторінок можна представити так:

1. Початкова сторінка – сторінка, яка відкривається першою при запуску програми. На ній буде представлена інформація про цікаві місця Львова. Також з цієї сторінки будуть відбуватись переходи на інші сторінки.
2. Сторінка для місць з AR – сторінка, де користувачу буде наданий доступ до вікна камери. Користувач, надавши доступ до камери пристрою, зможе відкрити камеру в програмі та побачити AR об’єкт на екрані свого смартфона.
3. Сторінка для місць – сторінка, де користувачу буде надана ширша інформація про місце, яке його цікавить. Оскільки не всі об’єкти можуть мати підтримку AR, ця сторінка буде альтернативою, де користувачу замість камери буде надано більше інформації про об’єкт.
4. Сторінка для налаштувань – сторінка, де користувач зможе надавати та відкликати доступи до камери та локації. Оскільки користувач повинен надати доступи до камери та розташування, варто розробити

окрему сторінки, де користувач зможе керувати доступати через саму програму, а не через налаштування програми в телефоні.

Сервіси LBS, які дозволять бачити місцезнаходження користувача та об’єкта, який його цікавить, на карті можна інтегрувати в сторінки для місць.

Користувач має мати змогу шукати цікаві для нього місця. Для цього потрібно реалізувати функцію пошуку по назвам або ключовим словам для визначних місць.

3.2. Створення графічного інтерфейсу програми, компонента RectTransform

Сформулювавши вимоги, потрібно проаналізувати інструменти, які надає Unity, що дозволять створити інтерфейс програми.

В кожного UI елемента в Unity є компонента RectTransform (Rectangle Transform), який відповідає за розмір, позиціювання, повертання та співвідношення елемента на елементі Canvas, який тримає в собі всі UI елементи [13]. Позиціювання та розмір налаштовуються в пікселях, які є в співвідношенні 1:1 до пікселів екрану, який репрезентує Canvas.

Також в всіх компонентах RectTransform є “Anchor Presets”, або “Попередньо налаштовані прив’язки”, що є набором завчасно визначених конфігурацій точок прив’язки. Ці точки прив’язки визначають розташування елемента відносно його батьківського елементу (елемент, який йде вище в ієрархії об’єктів) і те, як цей елемент буде змінюватись при зміні розміру його батьківського елементу.

Проте, позиціювання елементів є тільки однією з складових для успіху інтерфейсу. Наступною важливою складовою є палітра кольорів та зображення, які зроблять інтерфейс візуально приємним для користувачів.

Для налаштувань зображень та кольорів в Unity існує компонента Image (та інші, в залежності від UI елементу та його компонент), яка застосовується для більшості елементів як стандартна. В елементах, які не мають завчасно визначеної компоненти Image існує власна система налаштування кольору. Компонента Image відповідає за колір елементу та його зображення. Колір представляється в форматах RGB та HSV. Зображення можна вибрати з любого асету в проекті, в якому тип текстури визначений як спрайт. Спочатку застосовується зображення як фон, коли колір може змінювати початкове зображення.

Розробники можуть бачити результати зміни параметрів компонент елементів відразу після їх зміни. Результати дій напряму відображаються на екрані сцени та гри. На екрані сцени розробник може маніпулювати UI елементами: змінювати їх розташування, розмір, повертання через спеціальні кнопки або через інспектор об’єктів, напряму змінюючи параметри в компонентах елементу. На екрані гри розробник може бачити те, як програма буде відображатись кінцевому користувачу.

3.3. Розгортка програми на Android

Розробники можуть встановлювати програми на Android пристрої через опцію Build and Run в Unity Editor. Це можливе тільки в тому випадку, якщо мобільний пристрій підключений до комп’ютера через USB та має включений режим розробника. Після натискання кнопки Build and Run Unity запросить вказати назву та розташування на комп’ютері APK файлу, який потім буде встановлений на мобільний пристрій. Після цього відбудеться

збірка всіх сцен програми та виклик різних процесів, що призведе до імпортування APK файлу на мобільний пристрій, який підключений через USB, та встановлення програми через зібраний APK файл.

По замовчування режим збірки в Unity Editor зазначений як Windows, Mac, Linux. Розробники повинні змінити платформу розгортки через вікно Build Settings (рис. 3.1), якщо їхня платформа розробники відрізняється від вказаних вище. Для розгортки програми на іншій платформі потрібно вибрати саму платформу в вікні Build Settings, та якщо вона не зазначена як вибрана, натиснути кнопку Switch Platform, після чого Unity проведе пере-збірку проекту для вибраної цільової платформи. Деякі з платформ не доступні для розгортки. Якщо цільова платформа недоступна для розгортки, її можна завантажити через Unity Hub.

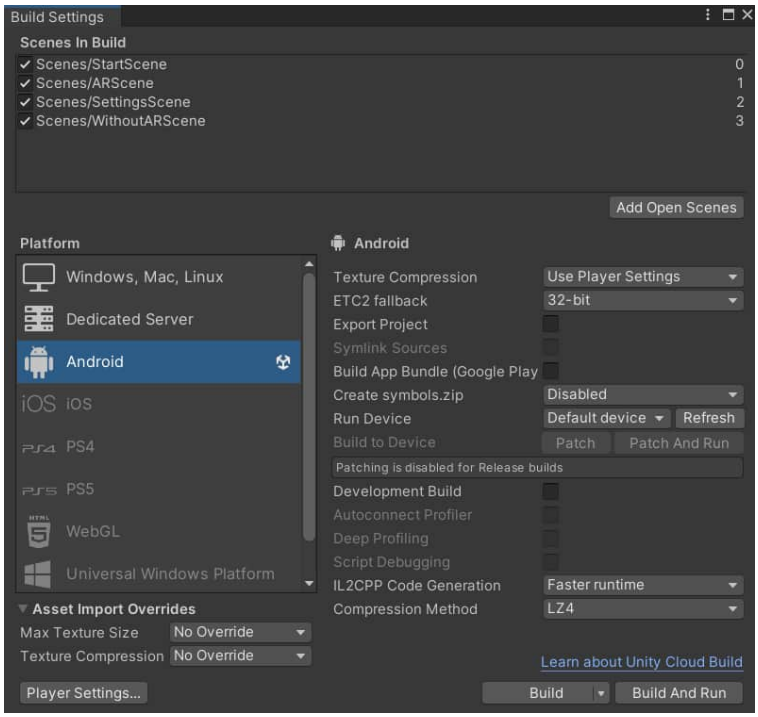

Рис. 3.1 – Інтерфейс вікна Build Settings

Після вибору цільової платформи розробки, розробники можуть зібрати свою програму через вікно Build Settings, натиснувши на кнопки

Build або Build And Run. Кнопка Build пропонує опцію Clean Build, що дозволяє зібрати програму “з нуля”, тобто без врахування попередніх збірок та кешованих даних. Це важливе в випадку непередбачуваної поведінки програми або помилок після збірки.

Unity надає дві можливості збірки: release та development. Збірка типу release, або реліз, включає в собі тільки те, що потрібно для коректного запуску програми. Це тип збірки є стандартним по замовчуванню. Збірка типу development, або розробка, включає в себе скрипти для дебагу.

Для збірки програми розробники повинні додати сцени, які повинні бути в програмі, через вікно Build Settings. Кожній сцені призначається порядковий номер, який розпочинається від нуля.

Також через вікно Build Settings розробники можуть перейти на вікно Player Settings (рис 3.2), яке є набором опцій, які визначають поведінку збірки гри або програми на визначеній платформі. Серед опцій є назва програми, версія, іконка, мінімальна версія системи та багато іншого.

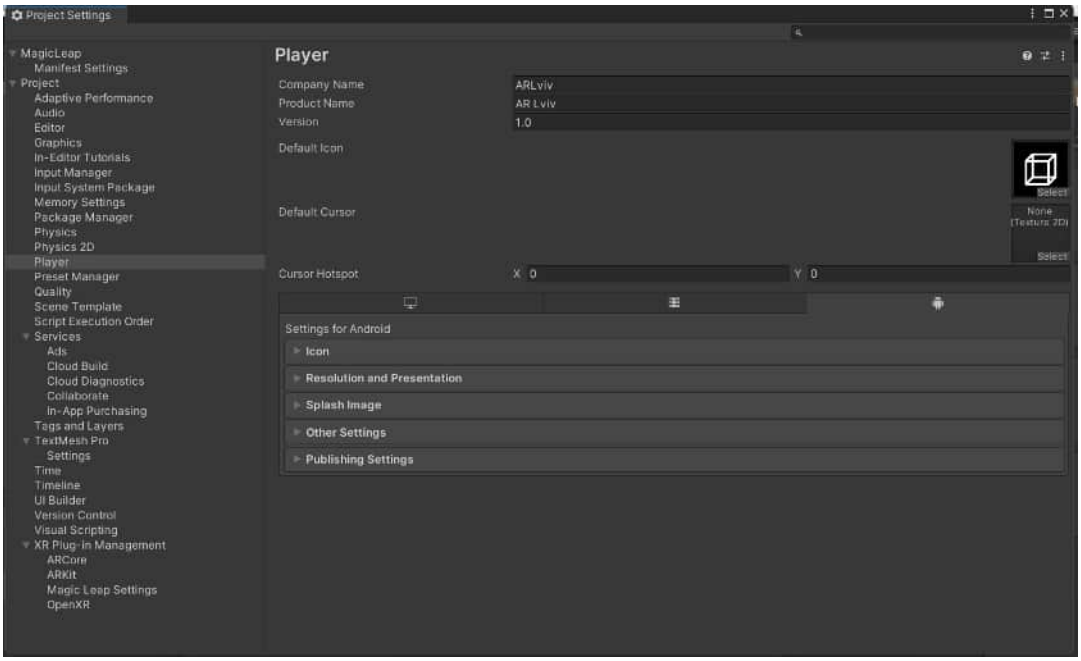

Рис. 3.2 – Інтерфейс вінка Player Settings

При збірці проекту на платформу Android Unity включає важливий файл AndroidManifest.xml, який є файлом з розширення XML, що надає важливу інформацію про програму операційній системі Android [15]. Ця інформація включає в себе назву програму пакету, необхідні дозволи для коректної роботи програми, мінімальну і цільову версію Android SDK та інше.

В Unity файл AndroidManifest.xml генерується автоматично під час процесу збірки. Розробники можуть самостійно змінювати вміст файлу та включати необхідні їм індивідуальні налаштування.

Для коректної роботи засобів LBS розробники повинні надати доступ до розташування через AndroidManifest.xml (рис. 3.3). Цей доступ є важливий для програм, які мають функціональність, яка пов’язана з визначенням розташування. Цей дозвіл надає можливість визначення місцезнаходження пристрою через використання точних GPS координат.

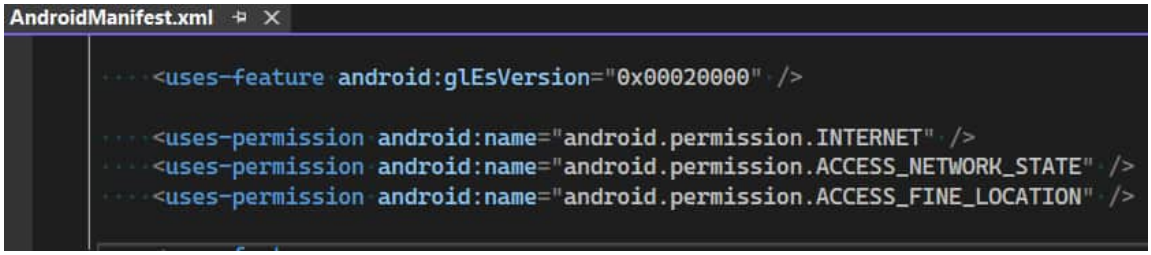

Рис. 3.3 – Надання доступу до місцезнаходження та інтернету в файлі AndroidManifest.xml

3.4. Скрипти та API, клас MonoBehaviour

Скрипти та скриптинг є невід’ємною частиною Unity. Скрипти використовуються для написання ігрової логіки та контролю ігрових об’єктів. Вони дозволяють задавати поведінку ігрових об’єктів, опрацьовувати вхідні дані від користувача, створювати анімації, обробляти фізику об’єктів та багато іншого.

Скрипти пишуться на мові C#. Це означає, що розробники можуть використовувати Visual Studio IDE для розробки їх програм та користуватись всіма переваги, які надає ця IDE. Unity дозволяє проводити дебаг коду через Unity Editor та Visual Studio.

Для створення скрипта розробники повинні натиснути правою кнопкою мишки по панелі проекту та вибравши опцію Create > C# Script. Існують інші способи створення скриптів, серед яких є можливість створення через вікно Assets або через вікно інспектора. Другий спосіб автоматично призначить цей скрипт до об’єкту.

Кожен скрипт потрібно призначати до ігрового об’єкту, якщо цей скрипт є наслідником класу MonoBehaviour [16]. В інакшому випадку скрипт не буде впливати напряму на поведінку об’єктів в сцені. Скрипти призначаються через кнопку Add Component на вікні інспектора об’єкту (рис. 3.4), або перетягуванням скрипта з панелі проекту до об’єкту в вікні інспектора, або в вікні ієрархії.

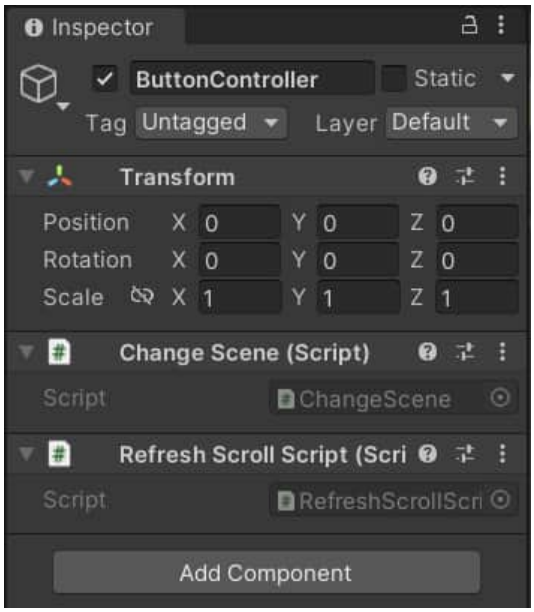

Рис. 3.4. Вид об’єкту ButtonController з двома призначеними скриптами

Скрипт є C# класом та є наслідником класу MonoBehaviour, який може використовувати всі можливості C# та функціонал простору імен UnityEngine. Клас може мати приватні та публічні змінні, мати власні функції, запускати асинхронний код тощо. Unity дозволяє створювати звичайні класи та інтерфейси на мові C#, але їх не можна буде призначити напряму до об’єктів в сцені.

Для роботи з 2D UI елементами Unity має простір імен UnityEngine.UI. Цей простір імен надає доступ до всіх класів об’єктів, які визначені як UI в Unity. Unity може створювати елементи через скрипт, або працювати вже з існуючими елементами в сцені. Для роботи вже з існуючими елементами розробники повинні зробити public змінну з назвою об’єкту, визначним Unity (рис. 3.5). Після оголошення змінних розробники можуть взаємодіяти з їх компонентами та статичними змінними.

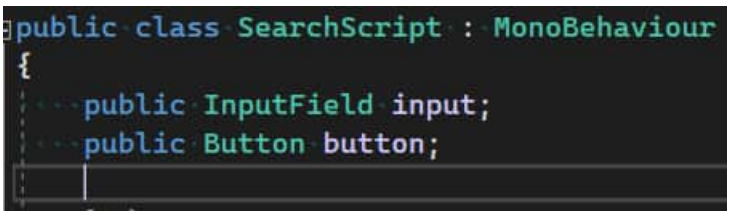

Рис 3.5 – Зображення класу SearchScript з двома public змінними типу InputField та Button

По замовчування всі створені змінні з простору імен UnityEngine.UI посилаються на NULL, тобто вони не призначені ні до якого об’єкту. Існує декілька способів створення об’єктів в сцені через скриптинг: використання GUI або DefaultControls. Для призначення вже створених об’єктів потрібно виконати наступний механізм дій:

1. Створити public змінну з відповідним типом в C# файлі.
2. Призначити цей файл до об’єкту в Unity Editor.

3. Перетягнути об’єкт з відповідним типом до відповідної йому змінної на екрані інспектора.

Якщо тип об’єкту не збігається з типом, визначеним в коді, то Unity не дозволить призначити такий об’єкт. Призначати можна як UI елемент, так і асет з проекту.

Для роботи з Google Maps Platform в Unity потрібно завантажити Maps SDK від Google, яке надає можливості та інструменти для інтегрування даних з Google Maps в програму на Unity. SDK надає можливості для створення та відображення карт, додавання маркерів та власних шарів та засоби взаємодії з картами.

Інший спосіб взаємодії є прямим запитом до API Google Maps Platform. Цей спосіб доступний через засоби Unity. Обидва способи вимагають мати створений акаунт Google Developer та API ключ для роботи.

Для другого способу потрібно створити запит, який буде включати в себе: довготу, широту, приближення, розмір зображення, масштаб, тип карти та API ключ (рис 3.6). За необхідності можна добавити маркери на карті.

```
url = $"https://maps.googleapis.com/maps/api/staticmap?center={currentSite[0].Latitude},{currentSite[0].Longitude}" +
    $"&zoom={zoom}&size={mapImage.rectTransform.sizeDelta.x}x{mapImage.rectTransform.sizeDelta.y}" +
    $"&scale={mapResolution}&maptype={mapType}" +
    $"&markers=color:red%7C{currentSite[0].Latitude},{currentSite[0].Longitude}" +
    $"&markers=color:blue%7C{Input.location.lastData.latitude},{Input.location.lastData.longitude}" +
    $"&key={apiKey}";
```

Рис. 3.6 – Зображення запиту на отримання карти з двома маркерами

3.5. Створення програми

Для початку потрібно створити новий проект в Unity Hub. Типом проекту для створення буде шаблон AR, оскільки він містить в собі

попередньо завантажені AR складові, які потрібні для розробки AR (рис. 3.7).

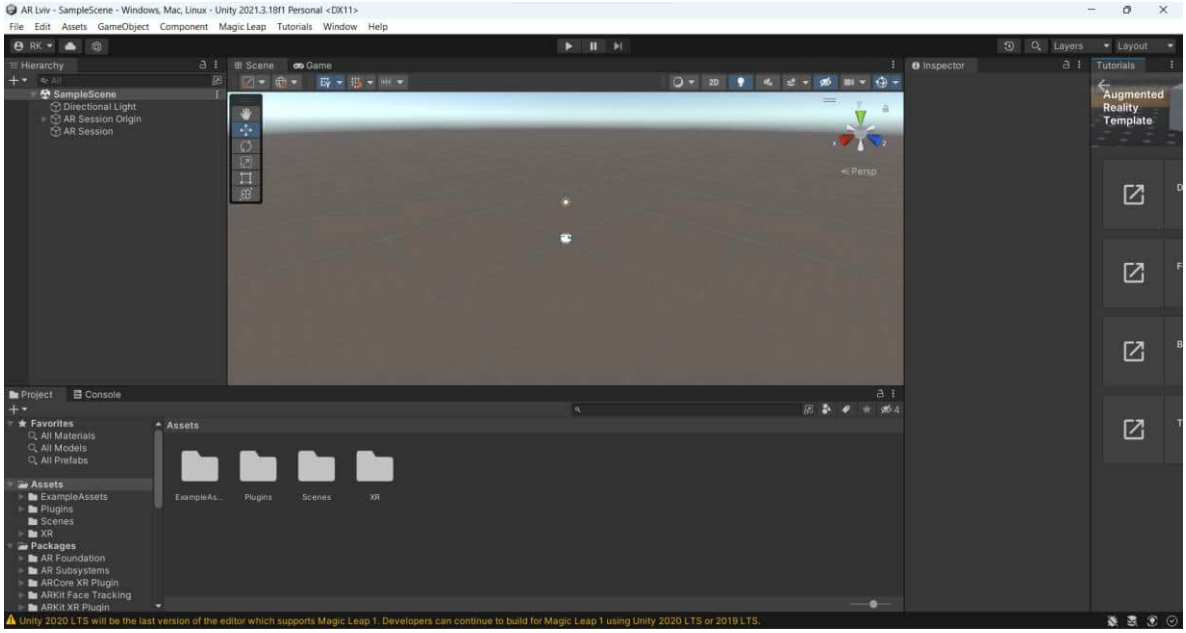

Рис. 3.7 – Вид нового проекту по AR шаблону в Unity Editor

Хорошою практикою буде організувати проект. Для початку потрібно завантажити необхідні пакети з асетами з Unity Asset Store. Ці пакети мають набір UI, які спростять розробку програми. Було вибрано два пакети: Dark Theme UI та Simple Button Set 02 [17][18]. Ці два пакети є безкоштовними для завантаження та надають доступ до текстур, які можна використовувати як зображення.

Далі необхідно організувати структуру проекту. Для цього потрібно створити дві папки: UI та Scripts. Перша папка буде містити в собі текстури, які будуть використовуватись в розробці, коли друга папка буде мати в собі всі скрипти програми. В папці UI створимо наступні папки: 128x128, 256x256, 512x512, які будуть містити в собі зображення відповідних розмірів. Також в папці Scripts потрібно створити ще три папки: AR, Data та UI, які будуть містити в собі скрипти, які відносяться до конкретної

категорії розробки. При необхідності структури цих папок можна розшири або змінити.

Після цього створимо 4 сцени, про які було описано в пункті 3.1. Для цього перейдемо в папку Scenes та створимо 3 сцени: StartScene, WithoutARScene та SettingsScene. Сцену для AR не потрібно створювати, оскільки вона створюється по замовчуванню, достатньо її перейменувати з BlankAR в WithARScene.

Почергово перейдемось по всім сценам та добавимо їх в Build, як показано в розділі 3.3. Також змінимо цільову платформу на Android.

Наступний механізм дій полягає в створенні Canvas для кожної сцени. Потрібно завчасно передбачити те, як кожен елемент в інтерфейсі буде взаємодіяти один з одним.

Canvas для основної сторінки виглядатиме наступним чином:

- В верхній частині знаходиться логотип та кнопка переходу на сторінку налаштувань. Також там знаходиться поле для вводу, де користувач зможе вводити пошукові запити.
- В центральній частині екрану знаходиться слайдер, на якому буде відображатись вся наявна в програмі інформація про цікаві місця Львова. З цього слайдера буде відбуватись навігація по об’єктам.

Canvas для сторінок з можливістю AR перегляду наступний:

- В верхні частині знаходиться кнопка з поверненням на головну сторінку та випадний список з розширеною текстовою інформацією. Також там знаходиться випадний список з картою, де користувач зможе бачити своє місцезнаходження та розташування конкретного об’єкту.

- По центру знаходиться екран для камери, через який користувач зможе бачити AR об’єкт на екрані свого смартфона.
- Під екраном для камери знаходиться кнопка для оновлення екрану камери. Це потрібне для повторного розміщення 3D об’єкту.

Canvas для сторінки без можливостей AR схожий на попередньо описаний Canvas, але з деякими відмінностями:

- В верхній частині присутня тільки кнопка повернення.
- Центральна частина з камерою замінена на панель з текстом, де буде відображатись розгорнута інформація про об’єкт.
- Кнопки з включенням/виключенням камери немає, оскільки немає самої камери.

Canvas для сторінки налаштувань виглядатиме наступним чином:

- В верхній частині присутня кнопка повернення на головну сторінку та службовий випадний список, який буде пояснювати деякі параметри в налаштуваннях.
- В центральній частині знаходяться параметри, які користувач зможе змінювати по натисканню кнопок.

Проаналізувавши вимоги до сторінок, можна починати розташовувати UI елементи та створювати UI елементи для дизайну. Необхідний список UI елементів наступний:

- Button TMP
- Input Field TMP
- Dropdown TMP
- Image
- Toggle

- Scroll View
- Text

Щодо елементів для дизайну список наступний:

- Логотип
- Кнопка повернення
- Кнопка пошуку
- Зображення відсутності інтернету
- Зображення відсутності доступу до розташування
- Кнопка з знаком питання
- Зображення для об’єктів

Елементи дизайну можна як створити через графічний редактор GIMP, так і завантажити з інтернету. У випадку зображень для визначних місць Львова було використано безкоштовні зображення з Інтернету, які можна вільно завантажити. У випадку інших зображень, таких як логотип чи різних технічних кнопок, їх було розроблено з використання графічного редактора GIMP.

Після розробки дизайну важливою частиною залишається вирішення способу того, як будуть зберігатись дані про місця в програмі. Існує декілька варіантів: грубе вписування кожного елемента в дизайн, зчитування інформації з коду, зчитування з JSON файлу чи зчитування з інтернет ресурсу. Для зберігання інформації було вибрано спосіб з зчитуванням інформації з коду програми.

Спосіб з вписування окремо кожного елементу в дизайн є непрактичним та незручним. Розробнику доведеться вручну кожного разу добавляти новий об’єкт на Canvas вручну, що погано впливає на продуктивність розробки програми, а також ускладнить написання функції

пошуку для розробника. Перевагою такого способу є швидкодія, оскільки всі об’єкти зразу є на сторінці, а не завантажуються з часом після запуску.

Спосіб з зберігання в JSON є, мабуть, найбільш логічним, але через різницю в операційних системах та тому, як Unity працює з JSON, використовування JSON для складних об’єктів, які мають декілька полів та таких об’єктів є декілька, не є практичним в Unity.

Завантаження інформації з інтернету не є варіантом для розробки, оскільки програма повинна працювати оф-лайн, де тільки функції LBS вимагатимуть підключення до інтернету.

Існують і інші варіанти, але для розробки було вибрано варіант, де вся інформація зберігається в окремому класі, інформація з якого парситься в програму. Такий спосіб дозволяє уникнути помилок при читанні, які можуть виникнути при зчитуванні даних з JSON або інтернету. Спосіб зберігання інформації в коді не є найкращим, але його можна використовувати протягом деякого часу.

Визначившись з форматом зберігання даних потрібно визначити, які дані потрібно зберігати. Даними для зберігання буде: ID – порядковий номер, Name – назва локації, Description – опис локацій, ShortName – коротке ім’я для роботи з скриптами, IsArSupported – чи є підтримка AR, Latitude – широта, Longitude – довгота, URL – посилання на сайт в інтернеті з детальнішою інформацією (рис 3.8).

```
public class SitesInfo
{
    public int Id;
    public string Name;
    public string Description;
    public string ShortName;
    public bool IsArSupported;
    public float Latitude;
    public float Longitude;
    public string URL;
```

Рис. 3.8 – Приклад класу, який містить в собі інформацію про об’єкт

Так як дані будуть зберігатись в класі доцільно зробити інакший клас, в якому буде статична функція, що буде повертати масив елементів з тип типом класу, який було визначено для зберігання даних.

Створення основної сторінки

Після створення класу, який зберігає інформацію, цю інформацію потрібно виводити на основній сторінці в області Scroll View. Це буде робитися через скрипт: створення об’єктів Canvas та присвоєння інформації створеним об’єктам. Проте перед цим потрібно реалізувати перехід між сторінками, як було показано в розділі 3.3.

Після реалізації функції переходу між сторінками, можна починати працювати над виводом інформації на головну сторінку. Це можна зробити через DefaultControls – клас для створення стандартних UI елементів. Він дозволяє створювати UI елементи, що містять в собі стандартний набір компонент для цих елементів. Оскільки цей клас створює UI елементи, цими елементами можна маніпулювати одразу після створення.

Для кращої організації виводу інформації було створено клас, який містить в собі інформацію про вибране користувачем місце, а також масив всіх місць. Масив місць допоможе при написанні функції пошуку, а інформація про вибране місце допоможе виводити правильну інформацію про об’єкт.

Функція пошуку допоможе користувачам відшукати те, що їх цікавить. Функція пошуку буде працювати по запиту користувача через поле для вводу та натиску кнопки пошуку. Після введення пошуку за допомогою LINQ C# формує масив місць, в яких назва містить в собі введену інформацію з поля для вводу, та передає цю інформацію в статичний масив місць.

Після організації функції пошуку потрібно розробити кнопку “скидання” пошуку. Ця функція очищає статичний масив та заново завантажує сторінку, таким чином завантажуючи знову всі об’єкти в сцені.

Створені скрипти не будуть працювати самі по собі. Потрібно призначити їх до об’єктів в сцені та надати їм всі необхідні асети. Після всіх цих дій отримуємо такий результат (рис. 3.9).

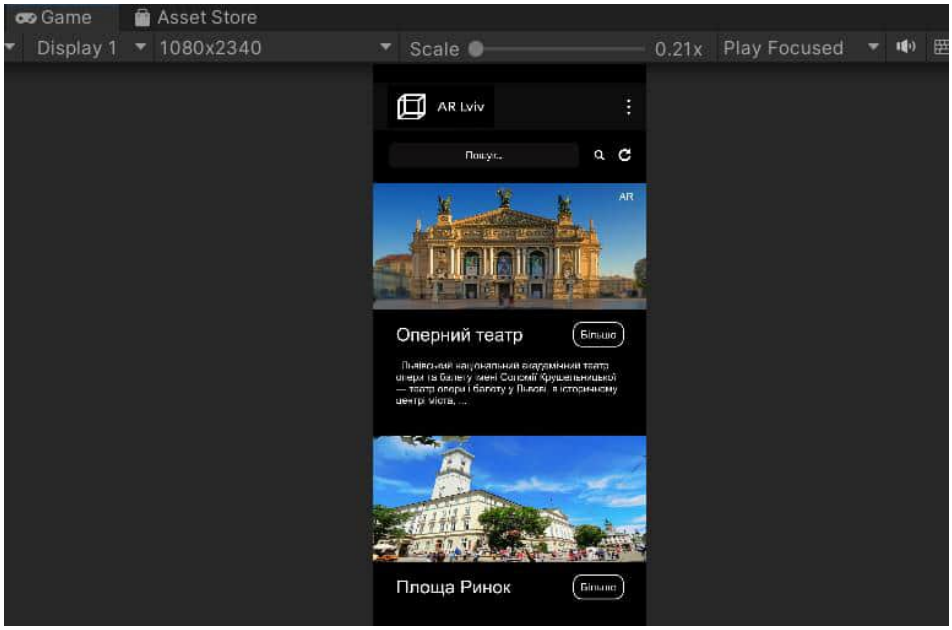

Рис. 3.9 – Основна сторінка програми в вікні гри в Unity Editor

Створення сторінки налаштувань

Наступною сторінкою для реалізації буде сторінка налаштувань, оскільки на ній користувач зможе надавати та відкликати доступ до камери та розташування свого пристрою. Без цих дозволів користувач не зможе отримати доступ до AR та служб LBS у програмі.

В верхній частині присутня кнопка, яка відповідає за повернення на головну сторінку. Також там присутній випадний список, що дозволяє переглянути інформацію по дозволам в програмі, та те, для чого вони потрібні.

Під верхньою панеллю знаходяться дві кнопки, які відповідають за надання дозволів до камери та розташування. Якщо користувач немає дозволу, то стан кнопки є виключеним та при натиску на неї у користувача запитають дозвіл.

Керування дозволами в Unity відбувається через клас Permission та його методи. RequestUserPermission та HasUserAuthorizedPermission. Перший метод при виклику відкриває діалогове вікно, де у користувача запитується дозвіл на доступ до камери, розташування, доступу до пам’яті тощо. Другий метод перевіряє, чи користувач надав доступ до спеціального доступу в Android. Обидва методи приймають як параметр один з дозволів Android, або декілька дозволів у випадку функції RequestUserPermissions, яка приймає масив стрічок з назвами дозволів.

Після створення всіх елементів вигляд сторінки буде наступним (рис. 3.10):

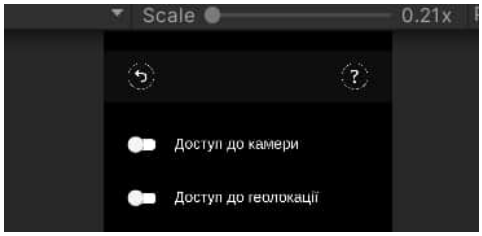

Рис. 3.10 – Вид верхньої частини сторінки налаштувань

Створення сторінки для місця без AR

Створення сторінки для місця є важливою частиною застосунку. Верхня частина схожа на ту, яка є на сторінці налаштувань, за винятком того, що на цій сторінці існує тільки кнопка повернення на головну сторінку.

Під верхньою панеллю знаходиться випадний список, де при його відкритті зображується карта з місцезнаходженням користувача та вибраного місця. Також у цьому випадному списку існує кнопка, яка відповідає за відкриття цієї ж самої карти у зовнішній програмі Google Maps.

Центральну частину сторінки займає область, де знаходиться розгорнутий текст про місце. Для виводу цієї інформації знадобиться попередньо описаний клас, який містить в собі статичну інформацію. Скрипт, який завантажує текст бере назву місця та через Linq шукає текст про це місце в загальному масиві всіх місць.

Під текстом знаходиться кнопка, яка переадресовує на офіційний сайт місця/пам’ятки в інтернеті, або на ресурс, де користувач зможе дізнатись більше про це місце.

На цій сторінці можна виділити 3 типи скриптів: скрипт для зображень, скрипт для тексту та скрипти для кнопок.

Скрипти для кнопок працюють в випадку натискання на відповідну кнопку. Два з них відкриваються зовнішні ресурси за допомогою класу Application та методу OpenUrl, де в параметри передається URL посилання на сайт. Якщо відсутнє Інтернет з’єднання, перехід не відбувається. Перевірка наявності доступу до Інтернету відбувається умовою Application.internetReachability == NetworkReachability.NotReachable. У випадку true Інтернет відсутній. Останній скрипт для кнопок відповідає за повернення на основну сторінку.

Створення сторінки для місця з AR

Сторінка для AR схожа до попередньо розробленої сторінки для місця без AR, за трьома винятками: область з текстом замінена камерою, верхня панель містить випадний список з розширеною інформацією про місце та

кнопка посилання на сторонній сайт знаходиться в цьому випадному списку, коли нижня кнопка замінена кнопкою оновлення екрану.

Створення скрипта для роботи з API

Робота по завантаження карти має відбуватись асинхронно. Для цього в C# існують механізми async/await, а також в Unity є тип Coroutine, що дозволяє розробникам призупиняти роботу методу на певний час або поки не виконана спеціальна умова. Corountine не призупиняє роботу головного потоку, що дозволяє обробляти запит на отримання карти на фоні.

При створенні скрипта потрібно враховувати *культуру* системи. Культура в C# - це набір конвенцій для форматування даних. Наприклад, в американській культурі форматування чисел з плаваючою комою відбувається через точку, коли в європейських та українській культурах воно відбувається через кому. Це створює проблему для роботи скрипта, оскільки для отримання зображення карти через посилання це посилання повинно відповідати всім вимогам:

https://maps.googleapis.com/maps/api/staticmap?center=49.84146,24.03127&...

де формат запису широти та довготи є через точку. В українській культурі ці числа підставляються з комою, що не дозволяє отримати зображення карти, оскільки Google Maps Platform API не може опрацювати такий запит.

Для подолання розбіжностей в культурах в C# існує CultureInfo – клас, який виводить інформацію про окрему культуру [19]. За допомогою методів цього класу культуру можна привести до вибраною існуючої культури або інваріантної, тобто такої, що не залежить від мови та регіональних параметрів. Інваріантна культура пов’язана з англійською, але не з якоюсь країною чи регіоном.

На персональному комп’ютері при запуску програми через Unity Editor культура виставляється en-US, що відповідає англійській мові та американській культурі. Водночас, на протестованих мобільних пристроях культура програми виставлялась як uk-UA, що дорівнює українській культурі. Через цю особливість служби LBS працюють некоректно на протестованих мобільних пристроях.

Для вирішення цієї проблеми культура, якщо вона не дорівнює en-US, то вона приводиться до цієї культури.

3.6. Розробка AR складової

При початку роботи з AR важливо пересвідчитись, що у проекті встановлені всі необхідні пакети даних для розробки AR. Зробити це можна у Unity Editor через вікно Window та його опцію Package Manager. Тип пакетів потрібно змінити з Packages: In Project на Packages: Unity Registry (рис 3.11).

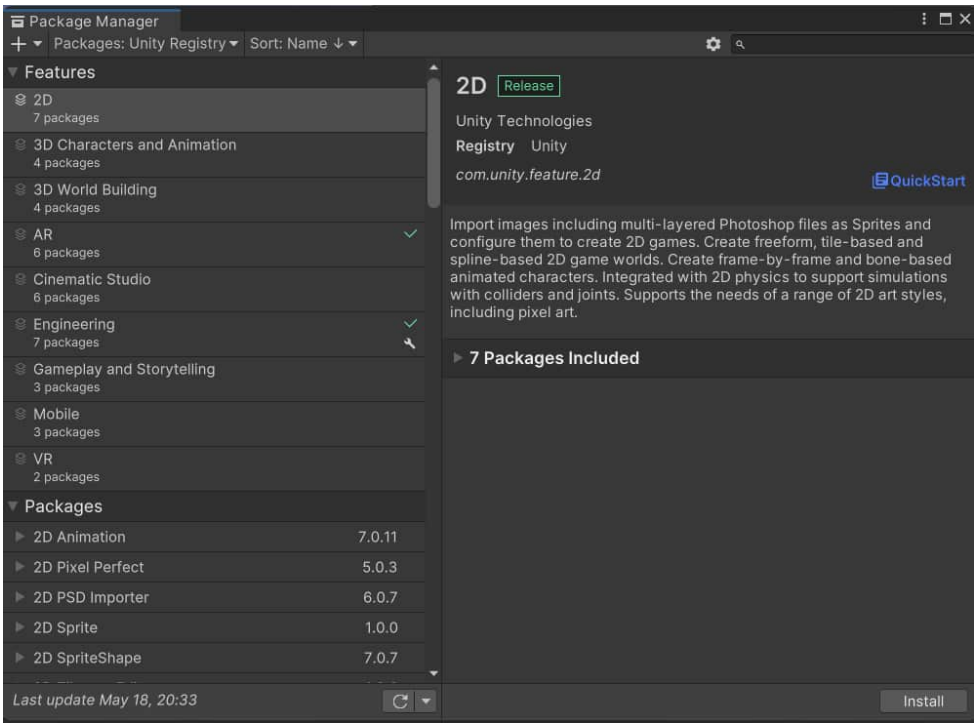

Рис. 3.11 – Вікно Package Manager

Встановленими пакетами для роботи з AR мають бути: AR Foundation та ARCore/ARKit пакет, в залежності від системи розробки, де ARCore відповідає за розробку під Android, коли ARKit є під розробку на iOS. Після встановлення пакетів важливо у вікні Project Settings на вкладці XR Plug-in Management ввімкнути ARCore для Android. Це дозволить Unity AR задіяти камеру.

Unity по замовчуванню створює сцену з майже всіма важливими елементами для розробки AR. Існує декілька способів розробки AR програм серед яких можна виділити два варіанти: варіант, коли Unity автоматично шукає пласкі поверхні та дозволяє розміщувати на них об’єкти та варіант, коли є набір попередньо завантажених текстур і Unity автоматично розміщує 3D об’єкти як тільки він розпізнає таку текстуру.

Для розробки було обрано перший варіант, оскільки другий варіант є не практичним в масштабах розпізнавання текстур, задіяних в даній програмі. В обох випадках потрібен елемент, який буде розпізнавати пласкі поверхні в реальному світі та розміщувати об’єкти на них. Таким елементом є AR Default Plane [20].

AR Default Plane автоматично ідентифікує поверхні та створює віртуальне відображення визначених поверхонь. Слугує як початкова точка для розміщення віртуальних об’єктів в AR середовищі.

Unity для розміщення AR використовує камеру, тому принцип розробки AR частини, пов’язаної з камерою, схожий до розробки звичайної камери. Має бути постійне оновлення даних в реальному часі для коректного відображення картинки.

Сам скрипт на відображення 3D об’єкту працює від того, чи існує поверхня для відображення цього об’єкту та чи користувач натиснув на

екран. Коли всі умови виконані завчасно призначений 3D об’єкт з’являється на місці натиску на екран.

Створення 3D об’єкту

Після створення скрипта нам потрібен сам 3D об’єкт, який ми будемо розміщувати. Безкоштовні 3D моделі можна знайти в відкритому доступі в інтернеті, або створити самому, використовуючи 3D редактори, такі як Blender чи Maya.

Для створення 3D моделі було обрано безкоштовний 3D редактор Blender. Самою 3D моделлю для створення було обрано колишній фонтан перед Львівською Національною Оперою. Ця 3D модель дозволить побачити площу перед оперою такою, якою вона була декілька років тому. Сама модель є доволі простою та її можна зробити відносно швидко.

Використовуючи інструменти моделювання в blender можна робити наступне:

- створювати базові геометричні об’єкти, такі як пряма, куб, сфера та інші
- моделювати полігони об’єктів та надавати їх власні контури
- розгортати 3D моделі на 2D поверхню для кращого нанесення текстур і матеріалів.

Також до використаних інструментів потрібно віднести нанесення матеріалів та текстур на моделі.

Кожен з цих перелічених методів був використаний для створення 3D моделі фонтану. Для створення форми об’єкту використовувались базові геометричні об’єкти, а саме квадрат, циліндр та коло. В подальшому до цих базових об’єктів застосовувались різноманітні операції для моделювання

полігонів, а саме видавлювання (extruding), масштабування (scaling), та маніпулювання вершинами, ребрами та гранями об’єктів. Для створення матеріалів та текстур було використано як інструменти blender, так і зовнішній графічний редактор по-типу GIMP.

Всі файли з моделями blender може зберігати в своєму власному форматі blend, так і інших форматах. Файли з розширення blend можна легко імпортувати в Unity як асет. Проте, їх потрібно додатково перетворити на Prefab.

Prefab в Unity – це асет, який містить в собі колекцію GameObjects. Оскільки 3D модель це не один елемент, як, наприклад, зображення, а набір полігонів, цю 3D модель потрібно перетворити в формат Prefab в Unity. Це дозволить взаємодіяти з елементами всередині моделі (рис. 3.12).

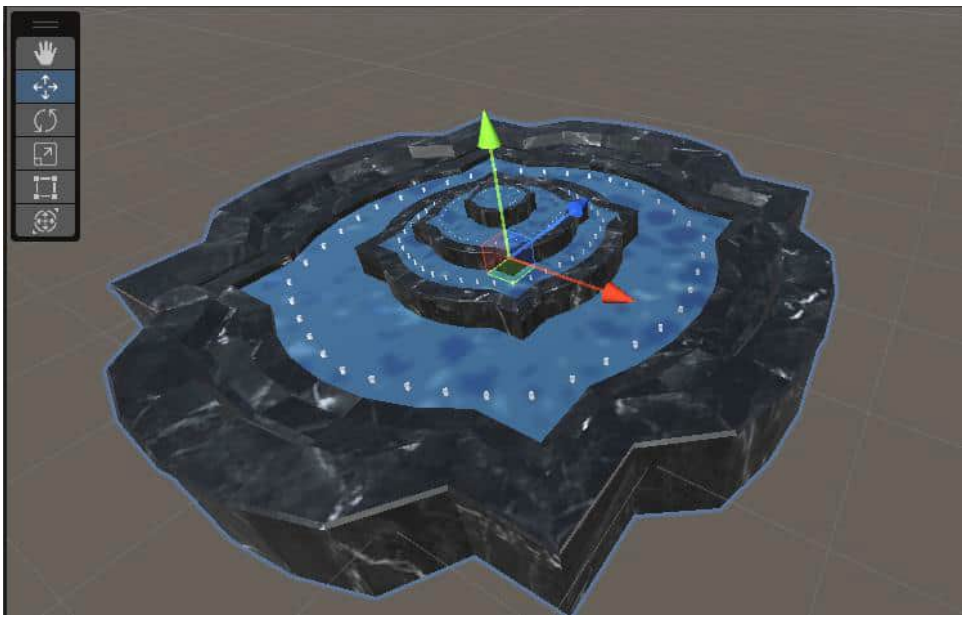

Рис. 3.12 – 3D модель в Unity Editor

РОЗДІЛ 4. ДЕМОНСТРАЦІЯ РОБОТИ ЗАСТОСУНКУ

4.1. Аналіз розробленої програми

В ході розробки програми для туризму з використанням AR технологій було спроектовано та реалізовано програмний застосунок для мобільної платформи Android на рушії Unity.

В ході розробки було зроблено:

- 4 сцени (сторінки) на Unity
- Реалізовано сервіси LBS
- Реалізовано AR складову програми

Програма має наступний функціонал:

- На головній сторінці користувач може переглянути список всіх цікавих місць у місті, а також використовувати функцію пошуку та скидання пошуку відповідними кнопками. Також доступні кнопки для переходу до сторінки налаштувань та сторінок окремих місць.
- На сторінці налаштувань користувач може налаштовувати дозволи програми, такі як доступ до камери та місцезнаходження.
- На сторінках окремих місць можна переглянути детальнішу інформацію про обране місце, включаючи розширений опис та його місцезнаходження на Google Maps. Також є можливість переходу до зовнішньої програми Google Maps для перегляду місця або до зовнішнього веб-сайту, де можна знайти більше інформації про обране місце.
- У AR-частині користувач може переглядати місце з іншого ракурсу за допомогою камери смартфона. Користувач може розмістити 3D об'єкт, натиснувши на екран пристрою, і переглянути його в

реальному світі. Функція скидання об'єкту та можливість розміщення його в іншому місці також доступні.

Мінімальною версію Android для встановлення та запуску програми є Android 7.0 ‘Nougat’, оскільки для роботи ARCore потрібне мінімальний API level 24, який є на системі Android 7.0. Також для роботи програми необхідно мінімум 200 МБ пам’яті, оскільки програма з даними користувача та кешом важить ~90 МБ.

4.2. Застосування програми

Застосунок було запущено на пристрої Xiaomi Mi 9 Lite з операційною системою Android 11. Розширення екрану цього пристрою складає 2340:1080 пікселів. Програму було встановлено через функцію Unity Build and Run. Також альтернативним варіантом встановлення є APK файл, який Unity створює під час збірки проекту.

Кінцева програма містить в собі 4 сцени, одну 3D модель та Prefab цієї моделі, три текстури до 3D моделі

Збірка першої версії програми містить в собі 4 сцени, одну 3D модель та Prefab цієї моделі, три текстури до 3D моделі, 17 скриптів, 12 спрайтів (зображень). Збірка зайняла 13 хвилин та 53 секунди, оскільки проект збирав всі ресурси в перший раз. APK файл програми 39.520 МБ пам’яті, коли вже встановлена програма зайняла 86.01 МБ пам’яті пристрою, після запуску програма додатково починає важити більше на ~5 МБ пам’яті, 130 КБ з яких йде на кеш, коли інші МБ йдуть на дані користувача. Програма може отримати доступи до геолокації пристрою та камери.

Програма містить в собі інформацію про 3 місця у Львові: Львівський національний академічний театр опери та балету імені Соломії Крушельницької, Площа Ринок та Високий Замок. Програма дозволяє

переглянути коротку інформацію на основній сторінці та більш розгорнуту інформацію на сторінці місця. Присутня можливість переглянути площу перед Оперним Театром через AR камеру.

Загальна інформація про програму

Програма працює в автономному режимі. Є дві функції, які потребують підключення до Інтернету: функція визначення місцезнаходження та функція переходу на зовнішній веб-сайт. Перша функція дозволяє отримати доступ до GPS та Google Maps Platform API. Якщо відсутній доступ до Інтернету, на екрані з картою, що відображає місцезнаходження користувача та місця, буде показане спеціальне повідомлення про відсутність підключення до Інтернету. Друга функція відповідає за перехід користувача на зовнішній веб-сайт у веб-браузері. У випадку відсутності з'єднання з Інтернетом, функція переходу на зовнішній веб-сайт буде недоступна.

Загальне користування програмою

Програма встановлюється через APK файл. При успішному встановлення програми її можна буде запустити через відповідний ярлик на екрані. Оскільки для створення програми використовувався Unity Personal, при кожному запуску програми відображається логотип Unity та текст на зображенні з написом Made With Unity.

При запуску програми відкривається перша, або основна сторінка, яка містить в собі логотип програми, перелік місць, поле для вводу тексту та інші елементи. Користувач переміщується по програмі жестами та тапами на екрані (рис. 4.1а).

Функція пошуку працює шляхом пошуку збігів введеного користувачем тексту з назвами місць, присутніх в програмі. Після введеного

запиту, якщо є збіги, то сторінка перезавантажується та відображає результати (рис. 4.1б).

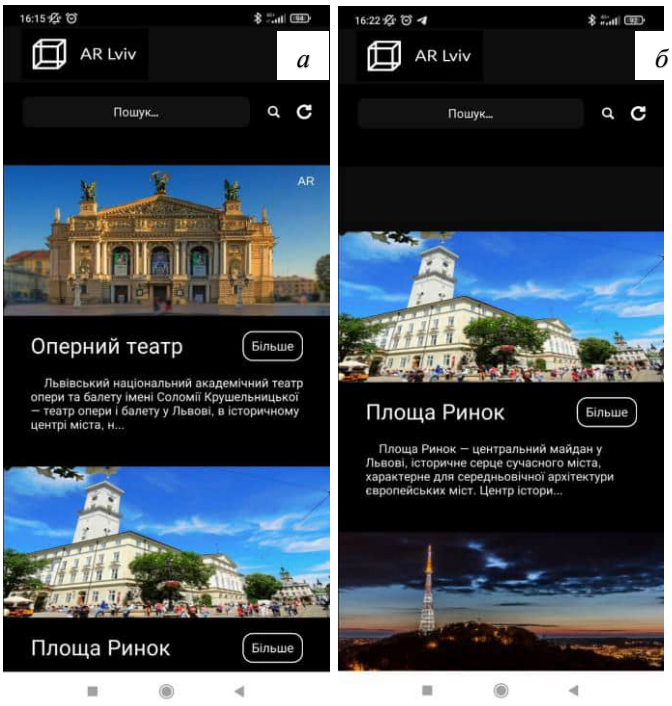

Рис. 4.1 – Основна сторінка застосунку:

а – після запуску програми, б – після виконання пошуку

На сторінці налаштувань можна надавати дозволи програмі на використання камерою та розташуванням. Якщо дозвіл не наданий, відповідний toggle, або перемикач, не буде включеним та при натисканні на нього відриється діалогове меню, де користувач зможе налаштувати дозвіл. Якщо дозвіл наданий, toggle буде ввімкненому положенні (рис. 4.2).

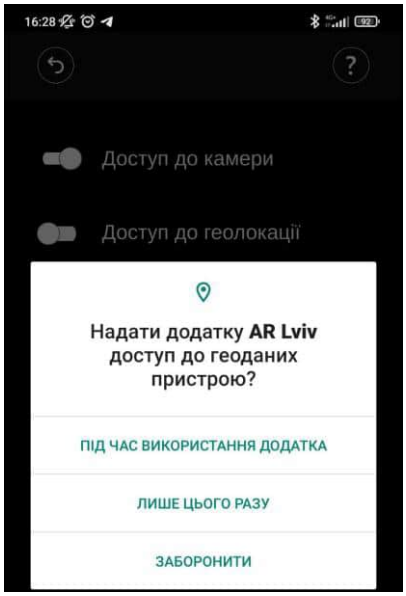

Рис. 4.2 – Сторінка налаштувань та запит дозволу до геоданих

На сторінці місця без AR користувач може побачити розширену інформацію про обране місце та перейти на зовнішній веб-сайт в Інтернеті та дізнатись більше про це місце (рис. 4.3а). На сторінці з AR текст винесений в окреме випадне вікно, яке можна відкрити натисканням на кнопку з зображенням знаку питання в правому верхньому куту екрану.

Доступ до карти однаковий на цих сторінках. Вона знаходиться в випадному окремому вікні. На карті відображається розташування місця червоним маркером, та розташування користувача синім маркером (рис. 4.3б). Присутня можливість перейти в програму Google Maps.

На сторінці з AR замість тексту на звичайній сторінкці є AR камера, де в реальному часі оновлюється інформація від AR Default Plane. Користувач може натиснути на любе місце на згеренованій площині та розмістити 3D об’єкт (рис. 4.3в).

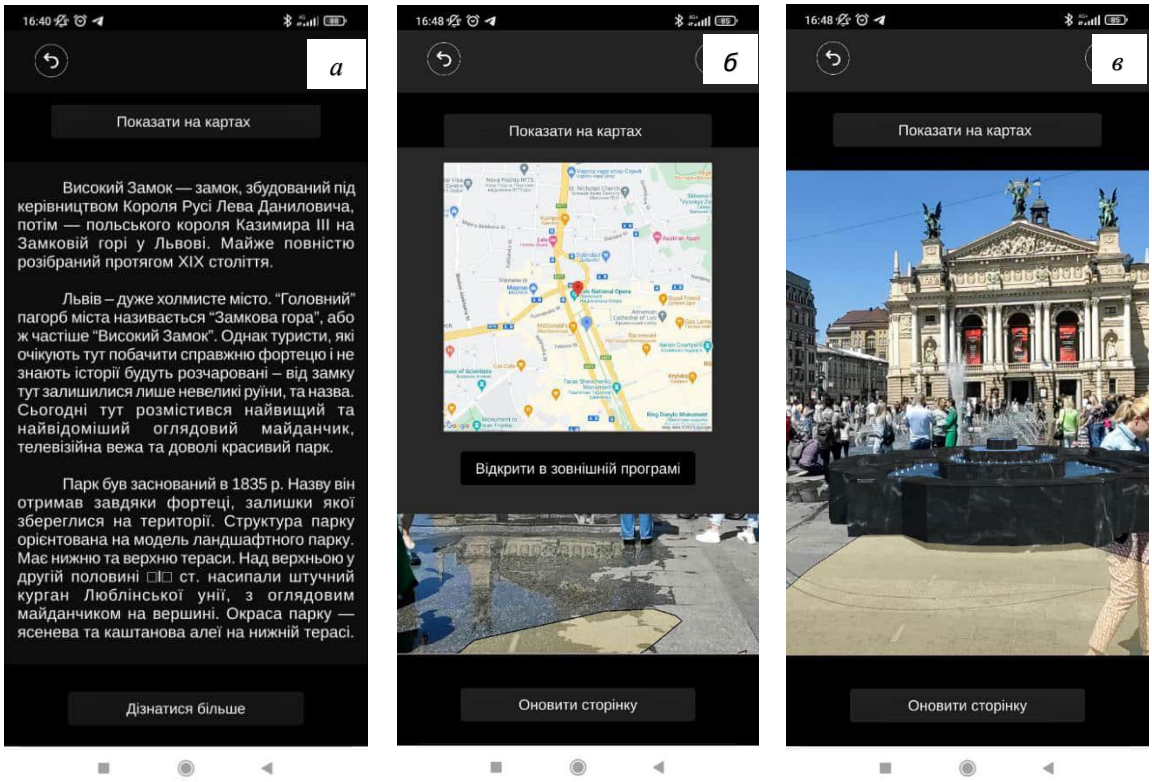

Рис. 4.3 – Робота програми на різних сторінках:

а – сторінка для місця без AR, *б* – випадне вікно з картою, *в* – AR сторінка з розміщеним 3D об'єктом

ВИСНОВКИ

Результатом виконання дипломної роботи стало створення мобільного застосунку про туризм з використанням AR технологій.

Попередньо проведений аналіз програм з схожим функціоналом дозволив виявити спільні риси між програмами, їх переваги та недоліки. Також був проведений аналіз ринку мобільних програм для виявлення трендів в розробці під мобільні системи.

Застосунок розроблений на рушії Unity з використанням мови C#, як основної мови для написання програмної логіки. Програма підтримується на мобільній операційній системі Android. AR складова програми розроблена за допомогою фреймворка AR Foundation та його складової AR Core. Сервіси LBS надаються за допомогою Google Maps Platform та Maps Static API.

Використання AR технологій для розробки є перспективним напрямком. Їх застосування дозволить створити унікальний досвід для туристів, збагатить їхнє розуміння та сприйняття навколишнього світу.

Подальшим розвитком цього програмного засобу є:

1. Розширення бази даних більшою кількістю місць та додавання нових місць з використанням AR технологій.
2. Вдосконалення якості 3D моделей, їх деталізацію та реалістичність.
3. Покращення інтерфейсу для більш зручної навігації.
4. Розширення охоплення програми, включення нових місць з інших міст або країн.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Number of international tourist arrivals worldwide from 1950 to 2022 [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://www.statista.com/statistics/209334/total-number-of-international-tourist-arrivals/
2. Mobile operating systems' market share worldwide from 1st quarter 2009 to 4th quarter 2022 [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/.
3. Future-proof your AR apps [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://unity.com/unity/features/arfoundation
4. Mobile App Design Trends 2023: Comprehensive Collection [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://fireart.studio/blog/mobile-app-design-trends-2020-comprehensive-collection/
5. UNITY [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://unity.com/
6. VISUAL STUDIO COMMUNITY [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://visualstudio.microsoft.com/vs/community/
7. C# [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://learn.microsoft.com/en-us/dotnet/csharp/
8. Jeffrey Richter. CLR via C# FOURTH EDITION / Microsoft prePress. – Redmond, Washington, 2012. – 250, с. 389, с. 502-510, 661, с.
9. TIOBE Index for May 2023 [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://www.tiobe.com/tiobe-index/

10. Why is UI UX design important for your website/app and business in general? [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://techbullion.com/why-is-ui-ux-design-important-for-your-website-app-and-business-in-general/
11. GNU IMAGE MANIPULATION PROGRAM [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://www.gimp.org/
12. Android Device Monitor [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://developer.android.com/studio/profile/monitor
13. Welcome to Google Maps Platform [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://mapsplatform.google.com/
14. RectTransform [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://docs.unity3d.com/ScriptReference/RectTransform.html
15. Android App Manifest [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://docs.unity3d.com/Manual/android-manifest.html
16. MonoBehaviour [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://docs.unity3d.com/ScriptReference/MonoBehaviour.html
17. Free Simple Dark Theme UI perfect for dark themed 2D game [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://assetstore.unity.com/packages/2d/gui/dark-theme-ui-199010
18. This Simple button set includes buttons and controls covering basic needs for designing a mobile game [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://assetstore.unity.com/packages/2d/gui/icons/simple-button-set-02-184903
19. CultureInfo Class [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://learn.microsoft.com/en-us/dotnet/api/system.globalization.cultureinfo?view=net-7.0

20. Configuring Plane Detection for AR Foundation [Електронний ресурс]: [Веб-сайт] – Режим доступу до ресурсу: https://learn.unity.com/tutorial/configuring-plane-detection-for-ar-foundation#