Міністерство освіти і науки України

Ніжинський державний університет імені Миколи Гоголя

Навчально-науковий інститут природничо-математичних, медико-біологічних наук та інформаційних технологій

Кафедра інформаційних технологій, фізико-математичних та економічних наук

Освітня програма: Комп’ютерні науки

Спеціальність: 122 Комп’ютерні науки

**КВАЛІФІКАЦІЙНА РОБОТА**

на здобуття освітнього ступеня *бакалавр*

**Створення гіпер-казуальної гри на движку Unity**

студента **Федана Андрія Володимировича**

**Науковий керівник:**

Фетісов Валерій Сергійович,

кандидат економічних наук, доцент

**Рецензент:**

Казачков Іван Васильович,

доктор технічних наук, професор

**Допущено до захисту:** \_\_\_\_ \_\_\_\_\_\_\_ 2023 р.

Завідувач кафедри

проф. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Казачков І.В.

Ніжин – 2023

ЗМІСТ

[ВСТУП 4](#_Toc137486301)

[РОЗДІЛ 1. ОСОБЛИВОСТІ СТВОРЕННЯ ІГОР 5](#_Toc137486302)

[1.1 Аналіз актуальності жанру на ринку 5](#_Toc137486303)

[1.2 Аналіз движків для створення гри 5](#_Toc137486304)

[РОЗДІЛ 2. ПІДГОТОВКА ДО РОБОТИ З ПРОЄКТОМ 7](#_Toc137486305)

[2.1 Встановлення програмного забезпечення Unity 2022 та Visual Studio 2022 7](#_Toc137486306)

[2.2 Перегляд документації Unity 10](#_Toc137486307)

[2.3 Вивчення роботи основних компонентів движку 11](#_Toc137486308)

[2.4 Створення та робота скриптів 13](#_Toc137486309)

[РОЗДІЛ 3. СТВОРЕННЯ ГРИ 18](#_Toc137486310)

[3.1 Продумування прототипу гри 18](#_Toc137486311)

[3.2 Створення пустого проєкту 18](#_Toc137486312)

[3.3 Контроллер гравця 19](#_Toc137486313)

[3.3.1 Джойстик 19](#_Toc137486314)

[3.3.2 Персонаж 20](#_Toc137486315)

[3.3.3 Камера 21](#_Toc137486316)

[3.4 Навколишній світ та взаємодія з ним 22](#_Toc137486317)

[3.4.1 Місце дії 22](#_Toc137486318)

[3.4.2 Границі 23](#_Toc137486319)

[3.4.3 Домівка 24](#_Toc137486320)

[3.4.4 Візуальні елементи 25](#_Toc137486321)

[3.4.5 Допоміжні та небезпечні об’єкти 25](#_Toc137486322)

[3.5 Збереження та завантаження даних 26](#_Toc137486323)

[3.5.1 Шифрування 26](#_Toc137486324)

[3.5.2 Збереження 27](#_Toc137486325)

[3.5.3 Завантаження 29](#_Toc137486326)

[3.6 Меню 29](#_Toc137486327)

[3.7 Кастомізація 32](#_Toc137486328)

[3.8 Валюта та покупки 33](#_Toc137486329)

[3.9 Музика 34](#_Toc137486330)

[3.10 Завершення всіх основних елементів гри 38](#_Toc137486331)

[3.11 Монетизація 38](#_Toc137486332)

[3.11.1 Вибір сервісу для показу реклами 38](#_Toc137486333)

[3.11.2 Показ реклами 40](#_Toc137486334)

[РОЗДІЛ 4. ПУБЛІКАЦІЯ ГРИ 42](#_Toc137486335)

[4.1 Тестування та виправлення помилок 42](#_Toc137486336)

[4.2 Публікація гри 42](#_Toc137486337)

[ВИСНОВКИ 47](#_Toc137486338)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 48](#_Toc137486339)

[ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ 49](#_Toc137486340)

# ВСТУП

**Актуальність теми.** Сучасний світ інформаційних технологій супроводжується значним розвитком комп'ютерних ігор, які стали невід'ємною частиною культури та розваг сучасного суспільства. Широке розповсюдження цих ігор та постійне зростання популярності серед гравців стимулюють постійний пошук нових форм ігрового досвіду, які б відповідали їхнім потребам та очікуванням.

Одним з популярних напрямків в ігровій індустрії є гіпер-казуальні ігри. Вони пропонують швидкі, захоплюючі та легкі за правилами ігрові випуски, які легко освоюються гравцями та відповідають їхнім коротким інтервалам вільного часу. Гіпер-казуальні ігри набули значної популярності серед гравців різного віку та ігрового досвіду, завойовуючи їхнє зацікавлення та відданість.

У контексті зростаючої популярності гіпер-казуальних ігор, актуальність полягає у створенні конкретної гіпер-казуальної гри, яка задовольнятиме високі вимоги гравців та приверне їхню увагу.

**Метою** роботи є ефективне створення ігор з використанням інноваційних підходів та методик, що враховують сучасні тенденції геймдизайну та технологічні можливості.

Основні завдання дослідження включають аналіз та узагальнення існуючих підходів до створення гіпер-казуальних ігор на практиці, виявлення факторів успішності та популярності таких ігор, а також розробку нових методик, що сприятимуть покращенню геймплею та відчуттю задоволення від гри. Результати дослідження матимуть практичну цінність для розробників комп'ютерних ігор, ігрових студій та всіх зацікавлених осіб, що прагнуть створювати якісні та привабливі гіпер-казуальні ігри.

**Ключові слова:** гіпер-казуальна гра, ігрова індустрія, геймдизайн, інноваційні підходи, методологія, популярність.

# РОЗДІЛ 1. ОСОБЛИВОСТІ СТВОРЕННЯ ІГОР

## Аналіз актуальності жанру на ринку

У даному підрозділі проведено аналіз жанрів ігор на ринку. Враховано останні тренди та нововведення, що сприяють актуальності на сучасному ринку геймдеву. Основні напрямки розвитку мобільних ігор, виявлені під час аналізу, включають:

* зростання популярності гіпер-казуальних ігор, які привертають широку аудиторію за рахунок простого інтерфейсу, цікавих концепцій та оригінальних рішень;
* використання віртуальної та доповненої реальності, а також хмарних технологій у мобільних іграх, що забезпечує нові можливості для захоплюючого геймплею та розширених взаємодій;
* інтеграція з соціальними мережами та створення метавсесвітів, де гравці можуть спілкуватися, змагатися та розвивати сюжетні лінії гри;
* З'явлення конкурентів для провідних магазинів, що викликає пошук нових шляхів розповсюдження ігрового контенту. Розширення способів монетизації мобільних ігор, включаючи різні моделі оплати та розміщення реклами, що дозволяють розробникам отримувати прибуток. Зростання ринків у розвивающихся країнах, які відображається у збільшенні прибутків та кількості користувачів.

Ці ключові тенденції в мобільній геймінговій індустрії підтверджують актуальність гіпер-казуальних ігор та вказують на потенціал для інновацій та розвитку у цьому сегменті.

## Аналіз движків для створення гри

У цьому підрозділі проведено аналіз різних движків, які можуть бути використані для створення гри.

При аналізі основних движків для створення гри, таких як Unity, Unreal Engine, Godot та Cocos2d, виокремлюються різні особливості та переваги кожного з них. Однак, в даному випадку, ми зосередимося на порівнянні цих движків та висловимо переваги Unity.

Unity є одним з найпопулярніших движків для створення ігор, особливо у сфері мобільних ігор. Ось деякі переваги Unity:

1. Платформна підтримка: Unity забезпечує розробку ігор для різних платформ, включаючи Android, iOS, Windows, macOS, Xbox, PlayStation та інші. Це дозволяє легко портати гру на різні пристрої та максимально охопити аудиторію.
2. Легкість використання: Unity має дружній інтерфейс та просту навігацію, що робить його доступним для початківців. Він також має широкий спектр документації, навчальних матеріалів та активної спільноти розробників, що полегшує процес вивчення та розвитку.
3. Розширені можливості: Unity пропонує багато готових компонентів, ресурсів та інструментів для розробки ігор. Він має велику бібліотеку додатків, дозволяє імпортувати сторонні ресурси та підтримує велику кількість плагінів, що сприяє розширенню можливостей розробки.
4. Кросплатформеність: Unity дозволяє розробляти гру один раз і експортувати її на різні платформи. Це зберігає час та зусилля, спрощує процес розробки та підтримки гри на різних пристроях.
5. Велика спільнота: Unity має велику та активну спільноту розробників, яка надає підтримку, ділиться знаннями та надає корисні поради. Це створює благоприятну середу для обміну досвідом та вирішення проблем.

# РОЗДІЛ 2. ПІДГОТОВКА ДО РОБОТИ З ПРОЄКТОМ

## Встановлення програмного забезпечення Unity 2022 та Visual Studio 2022

Для розробки ігор з використанням Unity необхідно мати встановлене потрібне програмне забезпечення. Unity 2022 є останньою версією движка, яка надає найновіші функції та можливості для розробки ігор.

Процес встановлення Unity 2022 зазвичай включає наступні кроки:

1. Завантаження інсталятора Unity Hub з офіційного веб-сайту Unity та його встановлення.

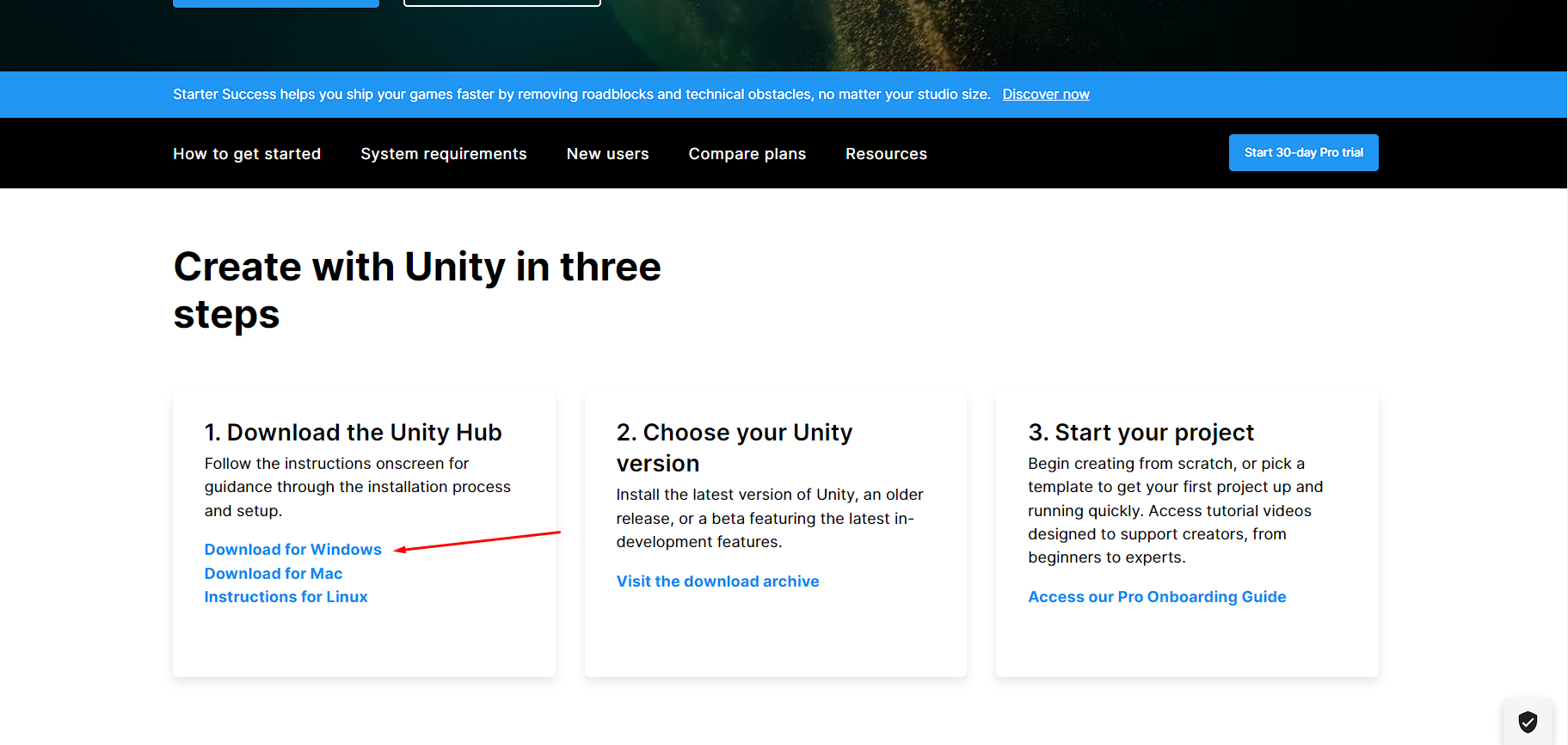


Рисунок 2.1 – Сторінка завантаження Unity Hub

1. Запуск Unity Hub та вибір версії Unity для встановлення.

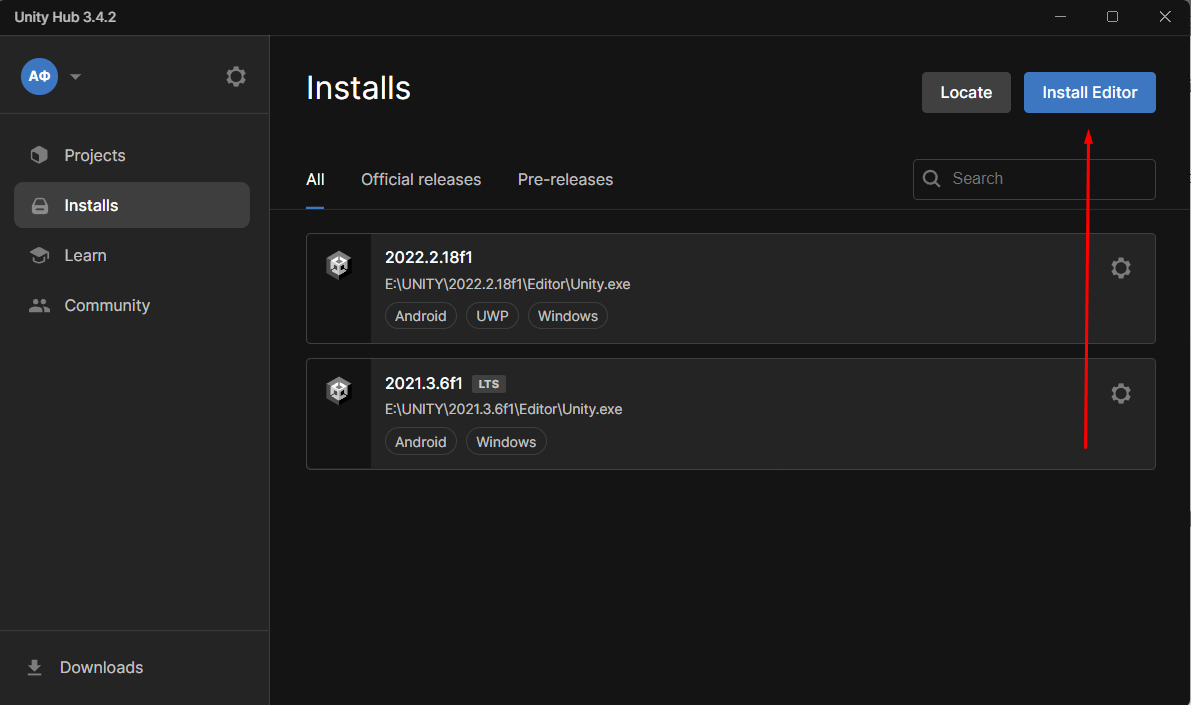


Рисунок 2.2 – Unity Hub, інтерфейс додання нового редактора

1. Встановлення Unity 2022 згідно інструкцій на екрані.

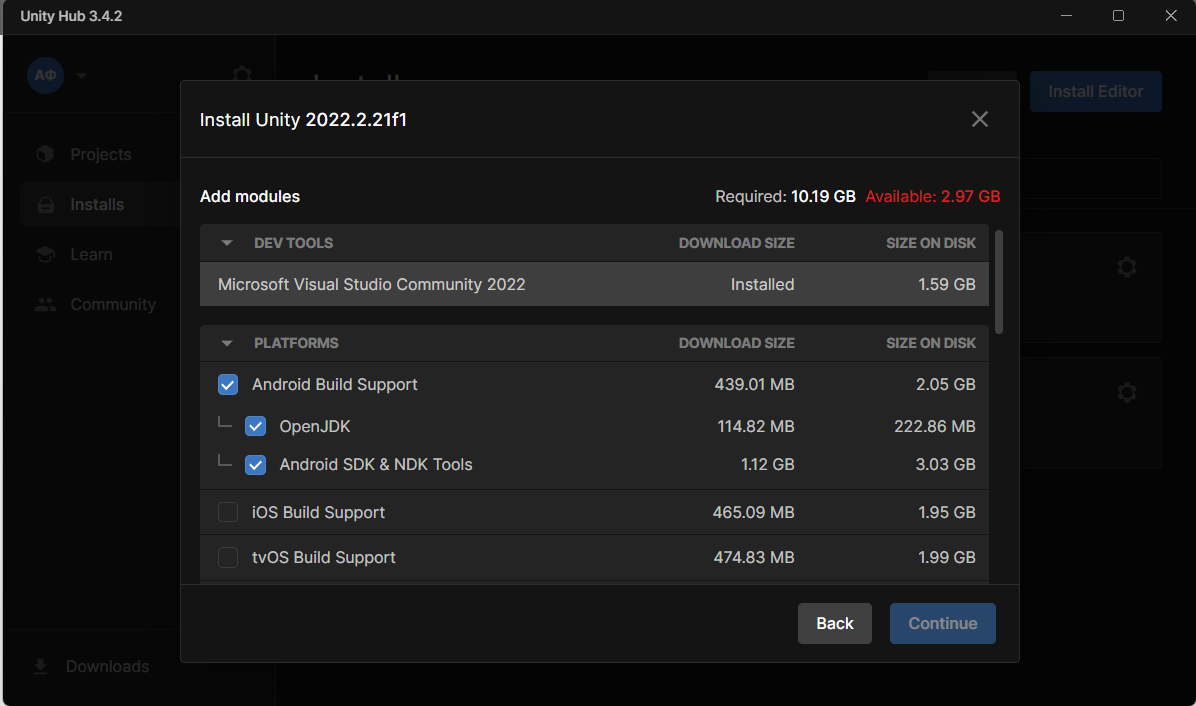


Рисунок 2.3 – Сторінка налаштувань при встановленні нового редактора

Окрім того, Visual Studio 2022 може бути використаний як інтегроване середовище розробки (IDE) для програмування ігор з використанням Unity. Воно надає багато корисних інструментів та можливостей для розробки ігрового коду.

Для встановлення Visual Studio 2022 можна виконати наступні кроки:

1. Завантаження інсталятора Visual Studio 2022 з офіційного веб-сайту Microsoft.

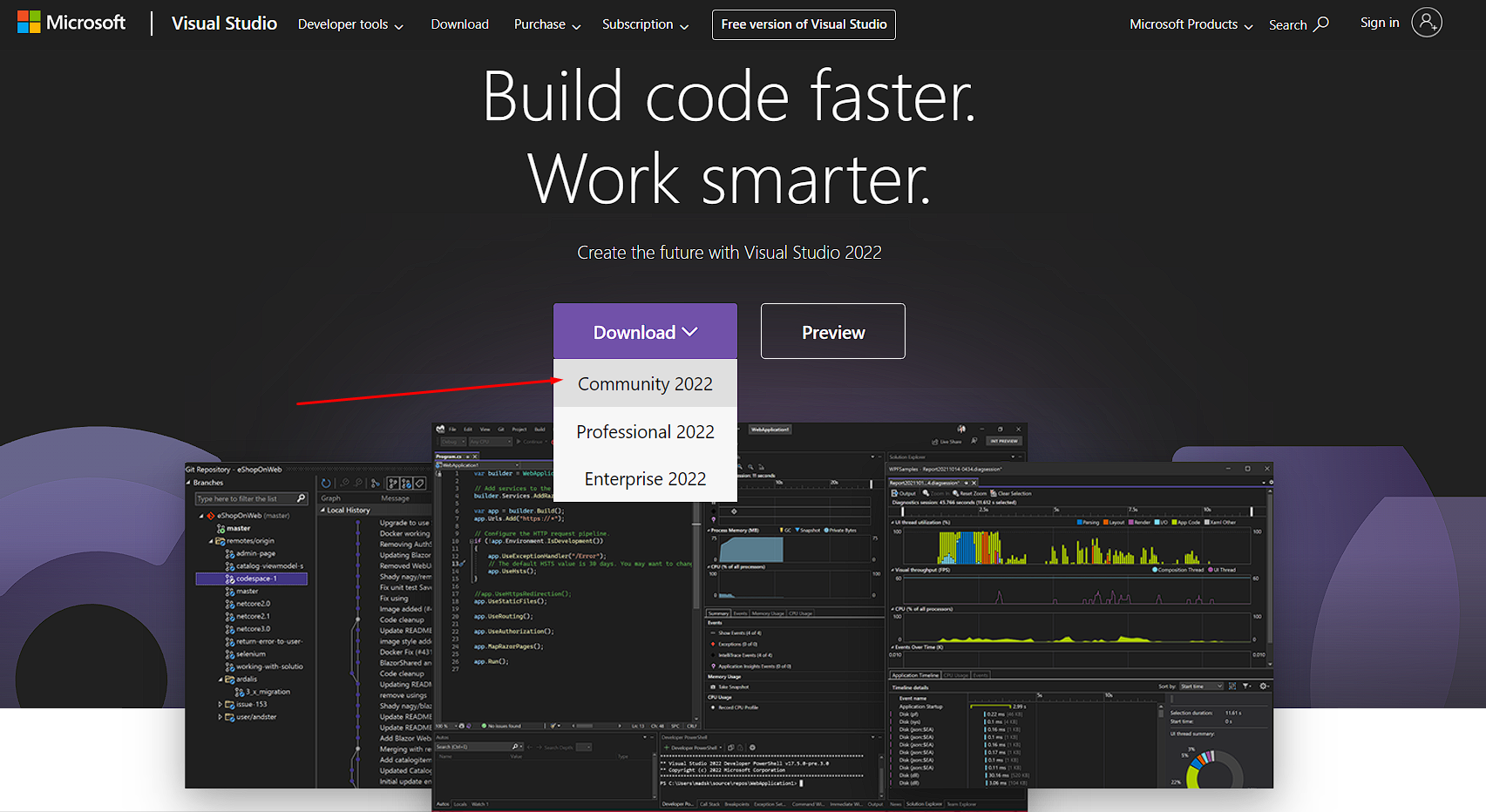


Рисунок 2.4 – Сторінка завантаження Visual Studio Community 2022

1. Запуск інсталятора та вибір необхідних компонентів для встановлення.

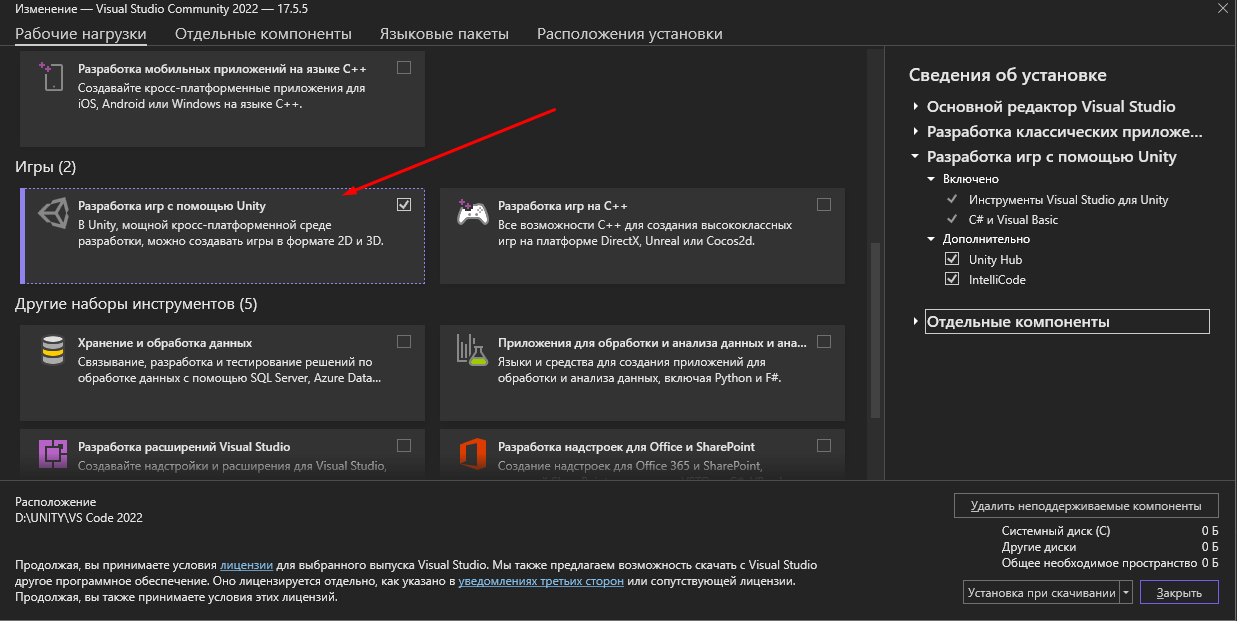


Рисунок 2.5 – Інтерфейс встановлення компонента Unity в

Visual Studio Installer

1. Встановлення Visual Studio 2022 згідно інструкцій на екрані.

Після успішного встановлення Unity 2022 та Visual Studio 2022 все готово щоб розпочати роботу з проєктом і розробляти власні ігри з використанням цих потужних інструментів.

## Перегляд документації Unity

Документація Unity надає вичерпну інформацію про функції, можливості та особливості движка. Вона включає офіційні посібники, посібники користувача, приклади коду, опис класів та інші корисні матеріали.

Перегляд документації Unity має наступні переваги:

1. Навчання: Документація Unity є відмінним джерелом для вивчення різних аспектів розробки ігор з використанням Unity. Вона надає пояснення, приклади та кращі практики, що допомагають вам зрозуміти роботу з движком та його можливості.
2. Довідник: Документація Unity є зручним довідником, який містить інформацію про різні класи, методи, властивості та інші елементи движка Unity. Вона дозволяє швидко знайти необхідну інформацію та використовувати її під час розробки.
3. Оновлення: Документація Unity постійно оновлюється, враховуючи нові функції та зміни у версіях движка Unity. Це дозволяє вам бути в курсі останніх можливостей та змін у розробці ігор.
4. Спільнота: Документація Unity також має форуми та коментарі, де розробники можуть обговорювати конкретні аспекти та отримувати додаткову підтримку від спільноти. Це створює можливість обміну досвідом та вирішення проблем у співпраці з іншими розробниками.

Для перегляду документації Unity ви можете скористатися офіційним веб-сайтом Unity, де доступні різні розділи та посібники. Ви можете шукати конкретну інформацію за допомогою пошуку або переглядати статті за тематикою.

Перегляд документації Unity є важливим етапом підготовки до роботи з проєктом, оскільки він допомагає вам краще розуміти можливості та принципи роботи движка Unity, що сприяє більш успішній та ефективній розробці ігор.

## Вивчення роботи основних компонентів движку

У даному підрозділі розглядається процес вивчення роботи основних компонентів движку Unity. Це важлива частина підготовки до роботи з проєктом, оскільки дозволяє розібратися з ключовими елементами, що використовуються при розробці ігор.

Основні компоненти движку Unity включають:

1. Сцени: Сцени в Unity визначають окремі рівні або екрани гри. Вони включають об'єкти, освітлення, камери та інші елементи, що визначають візуальну та просторову складові гри.

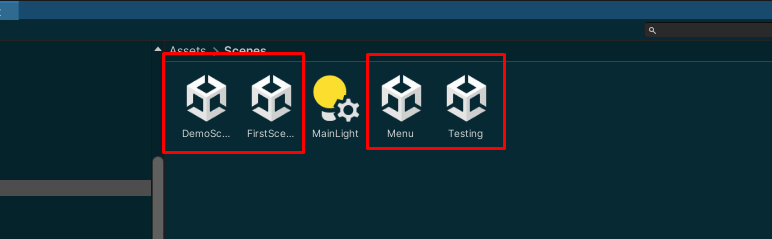


Рисунок 2.6 – Сцени



Рисунок 2.7 – Ієрархія об’єктів та вікно сцени

1. Об'єкти: Об'єкти є основними елементами гри. Вони можуть бути гравцями, ворогами, об'єктами оточення та іншими інтерактивними або статичними об'єктами. Кожен об'єкт має свої властивості, поведінку та можливості.

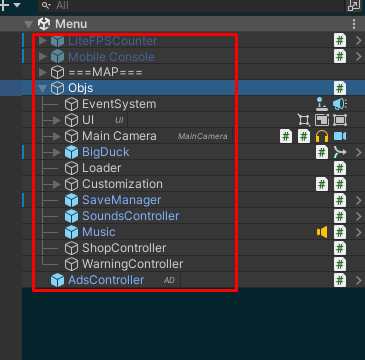


Рисунок 2.8 – Ієрархія об’єктів

1. Компоненти: Компоненти є частинами об'єктів і задають їх поведінку, властивості та можливості. Наприклад, компонент "Рух" визначає спосіб руху об'єкта, а компонент "Колізія" дозволяє виявляти зіткнення з іншими об'єктами.

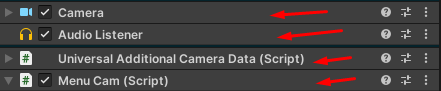


Рисунок 2.9 – Список компонентів на об’єкті

1. Скрипти: Скрипти в Unity написані на мові програмування, такі як C# або JavaScript, і додають функціональність до об'єктів та компонентів. Вони виконують різні дії, обробляють події та контролюють взаємодію об'єктів у грі.



Рисунок 2.10 – Скрипти

1. Фізика: Фізичний рух та взаємодія об'єктів моделюються за допомогою системи фізики в Unity. Це дозволяє створювати реалістичну поведінку об'єктів, їх реагування на сили та зіткнення.

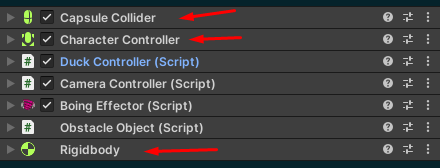


Рисунок 2.11 – Список основних фізичних компонентів

Вивчення роботи цих компонентів дозволяє розуміти, як вони взаємодіють між собою та як впливають на створення геймплею та візуального втілення ігор. Це дає можливість ефективно використовувати можливості движку Unity та розробляти високоякісні ігри з потужною функціональністю та привабливим виглядом.

## Створення та робота скриптів

У даному розділі розглядається процес створення та роботи зі скриптами в середовищі Unity. Скрипти є важливою частиною розробки ігор, оскільки вони визначають поведінку об'єктів та контролюють їх взаємодію.

Створення скриптів в Unity відбувається за допомогою мови програмування C#. Основні кроки створення скрипта включають:

1. Створення нового скрипта: У Unity ви можете створити новий скрипт, клацнувши правою кнопкою миші у вікні "Project" та вибравши "Create" -> "C# Script". Надайте скрипту відповідне ім'я.

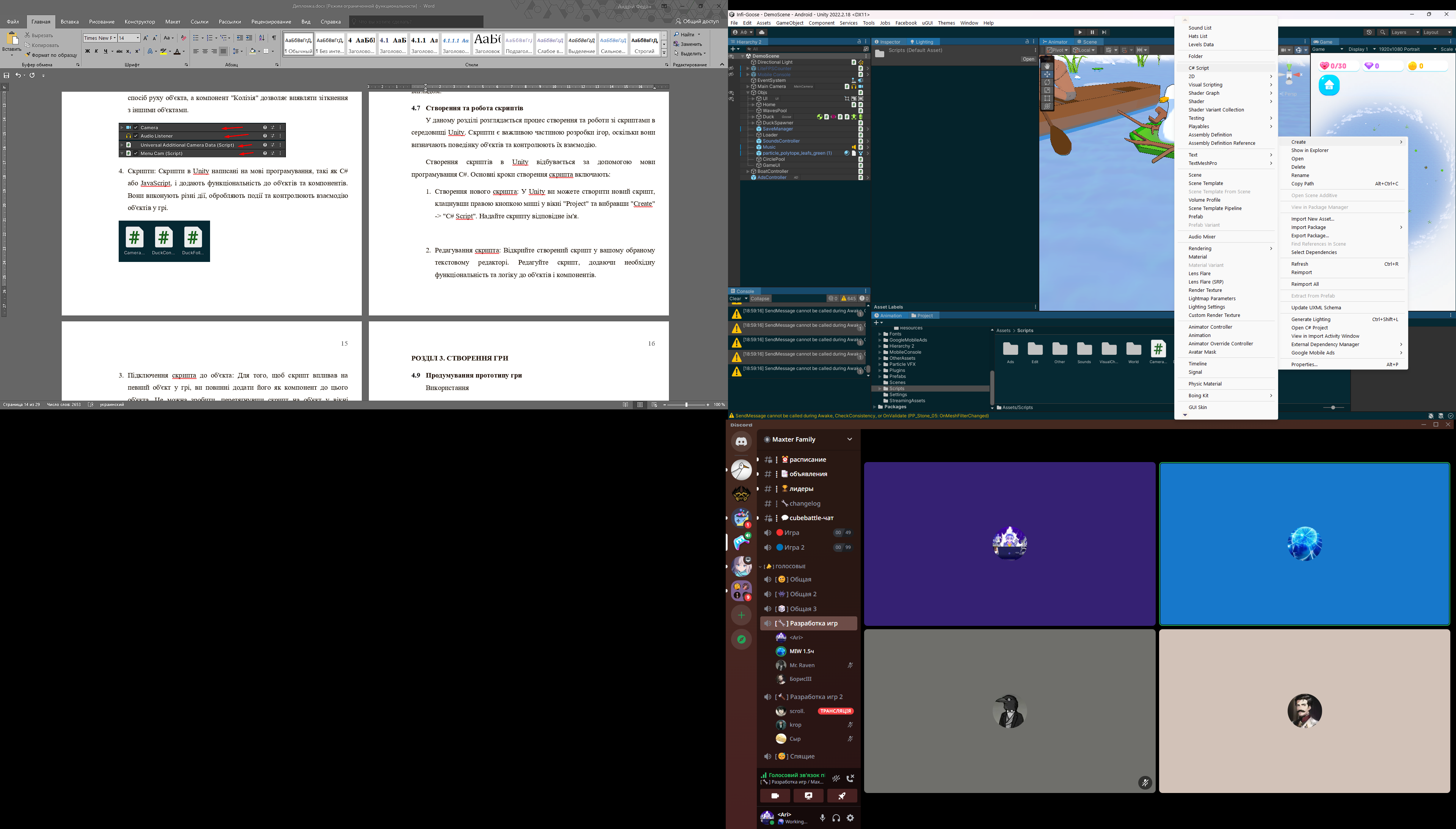


Рисунок 2.12 – Інтерфейс створення нового скрипта

1. Редагування скрипта: Відкрийте створений скрипт у вашому обраному текстовому редакторі. Редагуйте скрипт, додаючи необхідну функціональність та логіку до об'єктів і компонентів.

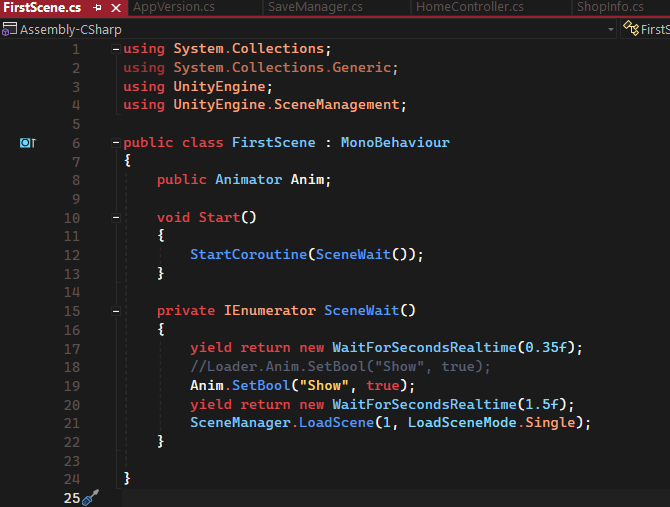


Рисунок 2.13 – Вікно редагування скрипта

1. Підключення скрипта до об'єкта: Для того, щоб скрипт впливав на певний об'єкт у грі, ви повинні додати його як компонент до цього об'єкта. Це можна зробити, перетягнувши скрипт на об'єкт у вікні "Hierarchy" або додавши його через інші методи підключення компонентів.

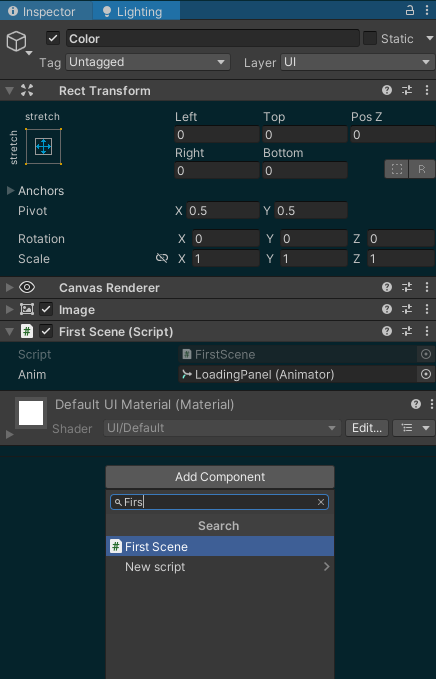


Рисунок 2.14 – Інтерфейс додавання компонента до об’єкта

Робота зі скриптом: Після підключення скрипта до об'єкта ви можете використовувати його функціональність та методи для керування поведінкою об'єкта, розробки геймплею та взаємодії з гравцем.

Під час роботи зі скриптами важливо мати розуміння основних понять програмування, таких як змінні, функції, умовні оператори та цикли. Також корисно знати специфічні API та функції, які надає Unity для роботи з об'єктами та компонентами.

Вивчення роботи зі скриптами дозволяє створювати власні логіку та інтерактивність у грі, реалізовувати унікальні функціональність та досягати бажаних результатів у розробці.

# РОЗДІЛ 3. СТВОРЕННЯ ГРИ

# 3.1 Продумування прототипу гри

Ідея гри полягає у керуванні качкою, яка збирає каченят на озері і переносить їх до гнізда. За кожне зібране каченя гравець отримує нагороду у вигляді монет. Зібрані монети можна використовувати для налаштування та кастомізації своєї качки. Цей елемент гри додає мотивацію та можливість індивідуалізації персонажа. Продумування прототипу гри є важливим кроком для визначення основних механік гри та створення цікавого геймплею.

## 3.2 Створення пустого проєкту

Для початку роботи над грою необхідно створити новий проєкт, в якому будуть зберігатися всі необхідні компоненти, ресурси та скрипти. Під час створення проєкту важливо обрати налаштування, такі як платформа, версія редактора, назва та місце зберігання. Створення проєкту є першим кроком у розробці гри і встановлює основні параметри та налаштування, які будуть використовуватися протягом всього процесу розробки.

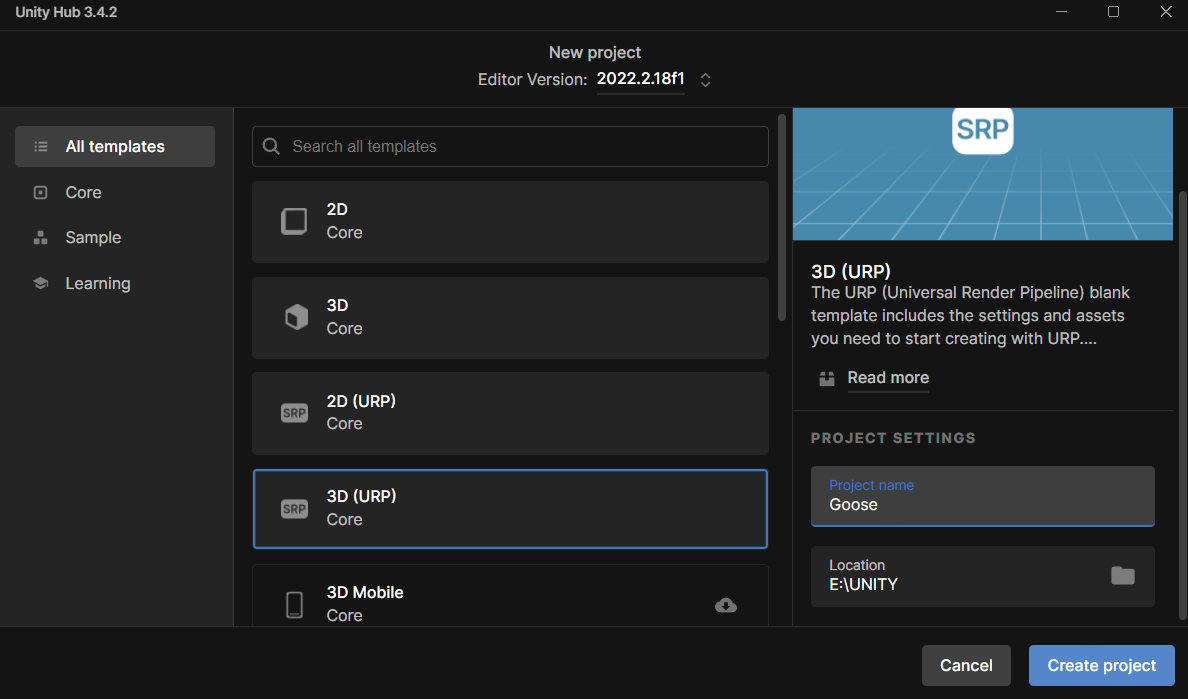


Рисунок 3.1 – Інтерфейс створення нового проєкту

## Контроллер гравця

### Джойстик

Для зручного контролю гравця, так як гра призначена для Android, потрібно використовувати джойстик. Його стиль виконаний у простому мінімалістичному вигляді.

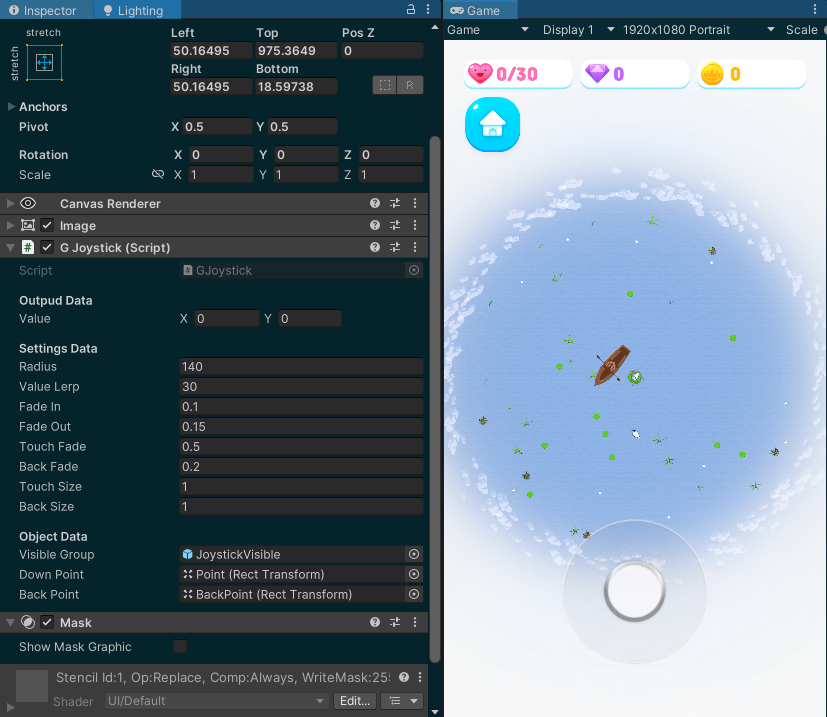


Рисунок 3.2 – Об’єкт джойстика та її компоненти

Алгоритм його роботи:

1. При натисканні у відповідній зоні відбувається дія визначення нового центру джойстика і джойстик стає видимим.
2. При русі точки натискання процес візуально відображається і зберігається у змінну значення у вигляді двомірного вектору.
3. При відпусканні відбувається перехід джойстика у прозорий вигляд.

### Персонаж

На об’єкт качки був доданий Capsule Collider для фізичних границь тіла, Character Controller для зручного контролю переміщення та створений C# скрипт Duck Controller для контролю качки.

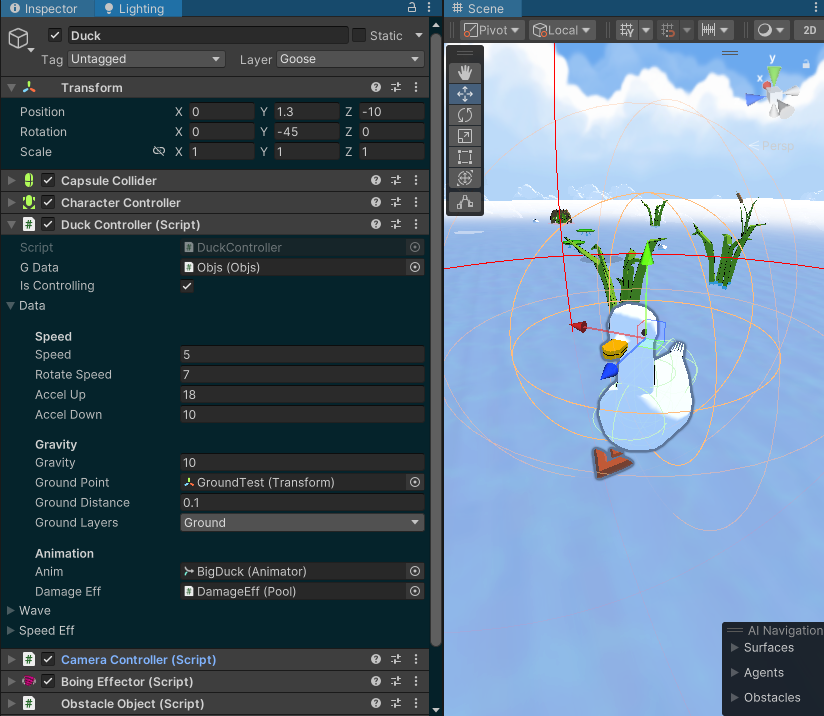


Рисунок 3.3 – Об’єкт качки та її компоненти

Взагалі персонаж має такий функціонал:

* рух (напрям бере з джойстика);
* перемикач контролю (може контролюватися джойстиком або ні);
* отримання шкоди;
* переродження (у випадку отримання шкоди);
* контролювати ефекти прискорення та уповільнення;
* мати анімацію та залишати позаду себе хвилі.

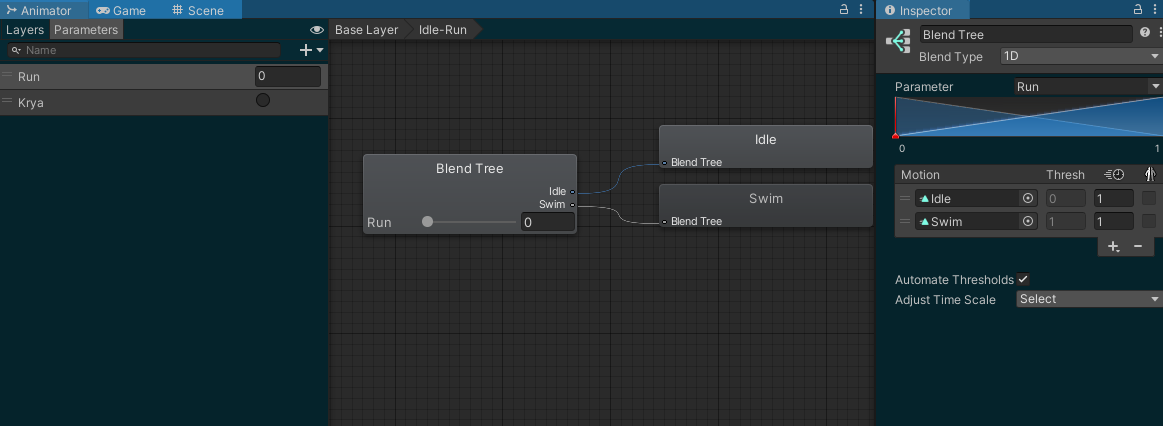


Рисунок 3.4 – Аніматор качки

### Камера

Даний об’єкт показує на екрані те, що входить в його поле зору, і досить важливо, щоб він працював плавно та зі зручної точки зору.

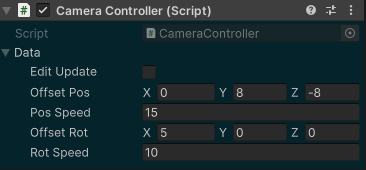


Рисунок 3.5 – Компонент скрипта Camera Controller

Контроллер камери вміє:

* налаштовувати кут зору та дистанцію до цільового об’єкта (гравця);
* мати невеликий поворот при русі конкретну сторону;
* плавний рух.

Сам компонент Camera налаштований, щоб відображати лише потрібні шари, мінімальну, максимальну відрисовки та кут видимості.



Рисунок 3.6 – Компонент Camera

## Навколишній світ та взаємодія з ним

### Місце дії

За місце дії взято невелике озеро у вигляді кола. Дане озеро має чіткі границі та створене у вигляді імітації води з анімацією та текстури піску на дні.

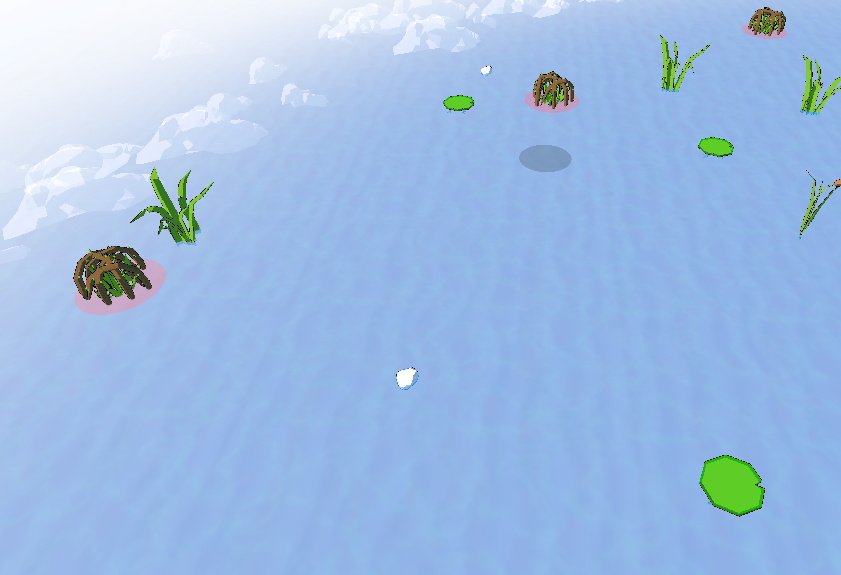


Рисунок 3.7 – Вигляд місця дії

### Границі

Використання границь є обов’язковим у грі з обмеженим у розмірі карті. Її ціль полягає у обмеженні переміщення гравця тільки в дозволеній зоні.

Границя була оформлена у вигляді хмаринок, які процедурно генеруються при старті рівня і в залежності від заданого розміру озера, вся місцевість заново генерується. Для кращої атмосфери, чим далі від границі назовні, тим більший білий туман.



Рисунок 3.8 – Вигляд границі

### Домівка

Домівкою у грі виступатиме гніздо з іншою качкою. Її функція, це дія збору каченят у гравця і повернення монет гравцю за кожне каченя. За кожне каченя є ймовірність 95% на отримання однієї монети, 4% на отримання 2-х монет та 1% на отримання 1-го критала.

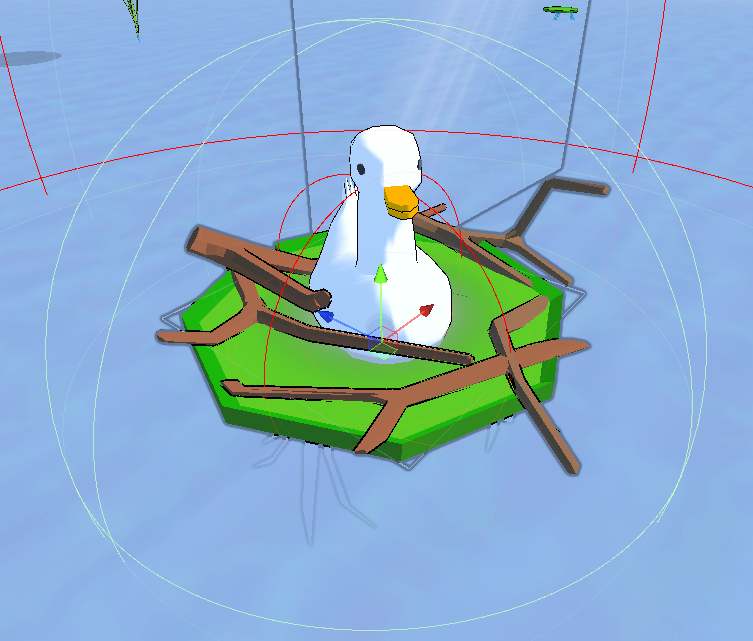


Рисунок 3.9 – Вигляд домівки

### Візуальні елементи

Для більш-естетичного вигляду території, на якій гравець грає, потрібно додавати візуальні елементи.

В грі наявні очерет, трава (різних розмірів), латаття та камінці у ролі візуальних елементів (тобто не мають ролі якоїсь із механік). Всі візуальні елементи при контакті з ними можуть відхилятися у протилежну сторону для естетичності.



Рисунок 3.10 – Візуальні елементи

### Допоміжні та небезпечні об’єкти

Для більшого інтересу гри, потрібно мати небезпеку персонажу та певні бонуси. Вони також додають більшої наповненості карти.

Наявні дві групи інтерактивних об’єктів:

* допоміжні (зона прискорення);
* небезпечні (зона уповільнення, колючі кущі та човен).

Човен має свій алгоритм роботи:

1. Чекати не точну кількість секунд.
2. Беруться дві точки на карті з мінімальною дистанцією в 90% діаметру озера.
3. Будується між точками крива в не точне значення сили викривлення, це і є траєкторія руху човна.
4. Рухається по траєкторії з конкретною швидкістю від першої точки до другої.
5. Після закінчення руху повертається до пункту 1.



Рисунок 3.11 – Вигляд човна, куща, зони уповільнення та прискорення

## Збереження та завантаження даних

### Шифрування

Для забезпечення безпеки даних гравця (кількість монет, кристалів, дату останньої сесії гри і тд.) потрібно збережені дані зашифрувати, а в грі при потребі розшифрувати.

Щоб не використовувати якийсь важкий алгоритм з ключами і скопійовані дані гравця з іншого пристрою не працювали на новому пристрої, вирішено шифрувати з використанням унікального ідентифікатора пристрою.

Ключ отримується за таким алгоритмом:

1. Записуємо ідентифікатор пристрою як рядок.
2. Якщо довжина рядка менше 32-х символів, то до цього рядка додаємо його ж ще раз.
3. Повертаємо значення в байтах системи ASCII рядка від нульового символу до 32-го.

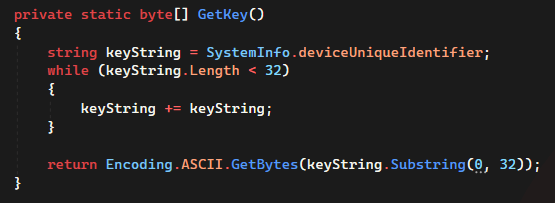


Рисунок 3.12 – Алгоритм отримання ключа для шифрування/дешифрування

### Збереження

Зберігання даних гравця в проєкті можна реалізувати різними способами, проте у всіх них є свої плюси та мінуси:

1. PlayerPrefs.

Плюси: простота використання, зручно для невеликої кількості данних.

Мінуси: незручно при великій кількості або довжині даних.

1. Json.

Плюси: можна зберігати де завгодно, зручно при великій кількості або довжині даних, група даних за одним посиланням.

Мінуси: файл збереження фактично існує та з ним можна робити що завгодно.

1. Хмарне середовище.

Плюси: безпека, зручно працювати з будь-якою кількістю даних.

Мінуси: для роботи обов’язкове підключення до інтернету, затримка при завантаженні даних.

В проєкті використано збереження даних з допомогою Json. Тобто всі дані зберігалися за алгоритмом:

1. Конвертація групи даних в один рядок.
2. Шифрування (два рази).
3. Перезаписування файлу збереження, вихід.

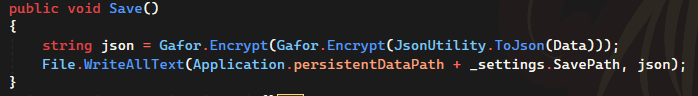


Рисунок 3.13 – Збереження даних



Рисунок 3.14 – Клас даних

### Завантаження

Завантаження даних працює в основі обернено до збереження, за алгоритмом:

1. Перевіряємо наявність файлу збереження.
2. За відсутності файлу збереження (при першому запуску гри) створюється файл збереження з початковими даними та виходимо з алгоритму.
3. За наявності файлу збереження зчитуємо текст.
4. Розшифрування (два рази).
5. Конвертуємо розшифрований рядок в групу даних.
6. Записуємо завантажені дані в нову групу, з якою далі буде йти праця, вихід.

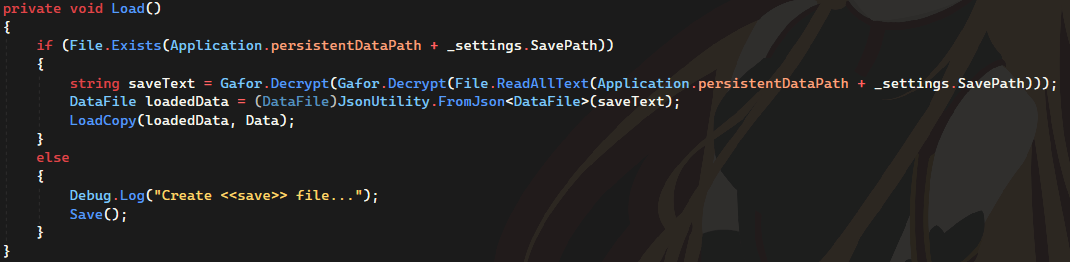


Рисунок 3.15 – Завантаження даних

## Меню

Меню гри відіграє важливу роль при відкритті гри, так як це перше після заставки, що побачить гравець.

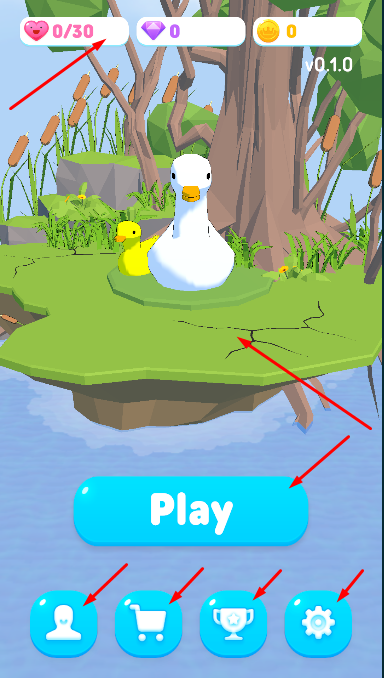


Рисунок 3.16 – Меню

Меню складається з кількох пунктів:

1. Візуал сцени територія острова з декораціями та качкою, яка повертається до гравця різними боками (можливо повертати самому), який знаходиться посеред озера.
2. Панель валют відіграє роль показу інформації наявних ресурсів для покупок.
3. Старт гри виконує перехід на ігрову сцену, де вже буде проходити основний геймплей.
4. Вікно кастомізації представляє собою список доступних до використання та покупки капелюхів.



Рисунок 3.17 – Вікно кастомізації

1. Вікно покупок є список обміну та покупки валют.

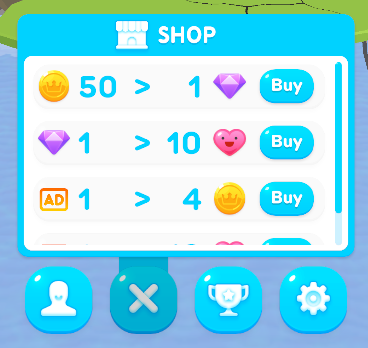


Рисунок 3.18 – Вікно покупок

1. Вікно списку лідерів показує рейтинг гравців по балам.



Рисунок 3.19 – Вікно списку лідерів

1. Вікно налаштувань має регулятори гучності звуків та музики.

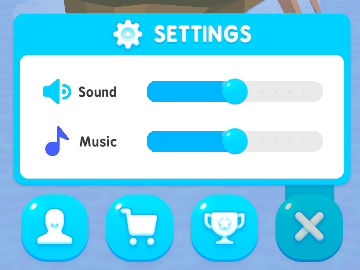


Рисунок 3.20 – Вікно налаштувань

При створенні меню було використано різні ефекти та анімації для естетичного вигляду.

## Кастомізація

Кастомізація важливий елемент гіпер-казуальних ігор, який додає інтерес довше залишитись у грі, щоб відкрити більше різних естетичних прикрас.



Рисунок 3.21 – Кастомізація

Кастомізація влаштована з такою логікою:

* при виборі капелюха, він одягається на качку та каченя, щоб відразу можна було бачити як вони виглядають в ньому;
* після покупки капелюха він одразу надягається;
* зберігання списку куплених капелюхів йде у вигляді назви капелюха та значення, чи надягнутий він.

## Валюта та покупки

Валютою у грі виступають три елементи:

* сердця (кількість каченят, які можна зібрати на озері);
* кристали (рідкісна валюта, для покупок);
* монети (найпоширеніша валюта для покупок);
* переглянута реклама (валюта, яка отримується під час покупки).

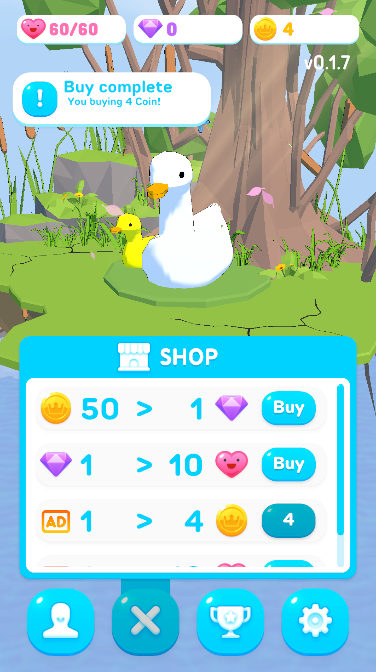


Рисунок 3.22 – Покупки та валюта

При вдалій покупці виводиться інформація, що товар був успішно куплений. При невдачі, виводиться інформація, що у гравця немає такої кількості ресурсу.



Рисунок 3.23 – Інформація про невдачу покупки

## Музика

Для створення підходящої атмосфери у грі наявні звукові ефекти та музика.

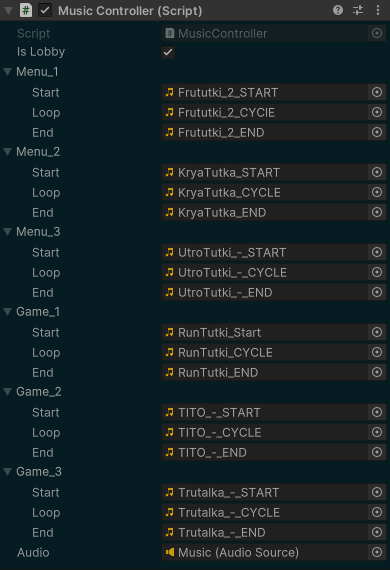


Рисунок 3.24 – Компонент з музикою

Наявні 6 треків (3 активні для геймплею на озері та 3 спокійніші для меню).

Кожен трек складається з трьох частин:

* початок (трек до 30с, який веде до циклічної частини);
* цикл (трек до 60с, який після свого завершення може знову починатися і це буде звучати гармонічно);
* кінець (трек до 2.5с, який грає в момент початку переходу між сценами).

Вибір треку залежить від завантаженої сцени, в меню свій набір, під час геймплею свій. Із набору трек вибирається методом рандому. Спочатку грає старт, далі циклічно грає цикл до тих пір, поки не буде виконаний перехід на іншу сцену.



Рисунок 3.25 – Скрипт відтворення музики

## Завершення всіх основних елементів гри

На цьому етапі додано багато маленьких елементів, виправлень знайдених помилок (як графічних, так і логічних).

Повністю завершена робота геймплейної, естетичної та технічної сторін гри.

Підготовлена перша версія для тестування на Android.

## Монетизація

### Вибір сервісу для показу реклами

Сервісів, які можна використати, щоб показувати рекламу в грі існує багато, основні з них:

* Google AdMob;



Рисунок 3.26 – Лого Google AdMob

* Unity Ads;



Рисунок 3.27 – Лого Unity Ads

* Facebook Audience Network.



Рисунок 3.28 – Лого Facebook Audience Network

Через надійність та високу ціну перегляду в грі використано сервіс Google AdMob.

Підключення даного сервісу виконано:

1. Завантаження пакету sdk.
2. Додання до проєкту sdk.
3. Створення Google AdMob проєкту.
4. Додання кількох оголошень.
5. Налаштування sdk в движку.
6. Створення скрипту, який буде відповідати за показ реклами.
7. Налаштування всіх значень оголошень та способів їх показу.
8. Завантаження проєкту на платформу мобільних ігор Google Play та підтвердження роботи реклами.

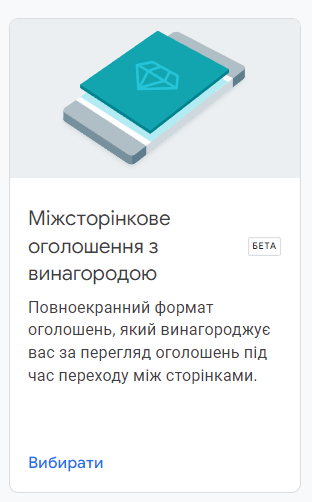


Рисунок 3.29 – Міжсторінкове оголошення з винагородою

### Показ реклами

Для зручності роботи з рекламою створений скрипт, який управляє завантаженням, ініціалізацією та показом реклами.

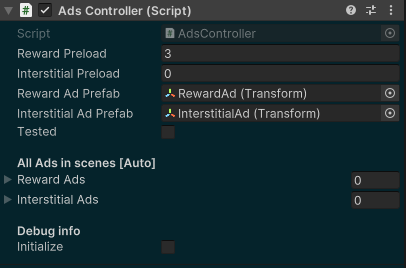


Рисунок 3.30 – Скрипт менеджера реклами

Алгоритм роботи менеджера реклами:

1. Перевірка на наявність інших менеджерів реклами (при наявності, цей менеджер видається).
2. Перевірка доступу в інтернет, при наявності переходимо до пункту 3, інакше повторюємо пункт 2.
3. Ініціалізація sdk.
4. Перевірка на потребу завантаження реклами в Pool, при потребі завантажуємо, чекаємо результат завантаження і знову повторюємо пункт.
5. При розриві з’єднання з інтернетом переходимо до пункту 2.

Після показу реклами, об’єкт з пулу видаляється та йде завантаження наступного. Завантаження реклами в Pool потрібне для ануляції часу очікування завантаження реклами коли потрібно її показувати. Завантаження реклами та її показ є різними діями.

Для виклику реклами створений окремий скрипт, який має функцію для виклику реклами та вільний для налаштування реакції у випадку перегляду реклами та помилки перегляду.

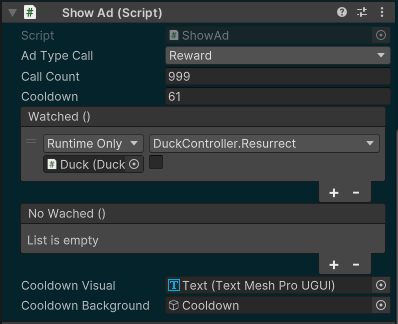


Рисунок 3.31 – Скрипт виклику реклами

# РОЗДІЛ 4. ПУБЛІКАЦІЯ ГРИ

## Тестування та виправлення помилок

Перед публікацією гри в Google Play потрібно протестувати гру, і, за можливості, виправити знайдені помилки.

Під час тестування знайдені та виправлені деякі помилки:

* масштабованість інтерфейсу;
* графічні артефакти;
* помилки при збереженні та завантаженні даних на Android;
* низьке розширення деяких текстур;
* реакції візуальних елементів;
* анімації інтерфейсу;

## Публікація гри

Для публікації гри в Google Play потрібно підтверджений акаунт розробника. Для його отримання потрібно:

1. Створити Google Account з яким він буде працювати.
2. Зайти на сторінку Google Play Console.
3. Придбати за 25$ дозвіл видачі ігор.
4. Пройти верифікацію документів (паспорт, код платника податків, місце приписки).

Після отримання повноцінного акаунта розробника потрібно створити новий проєкт.

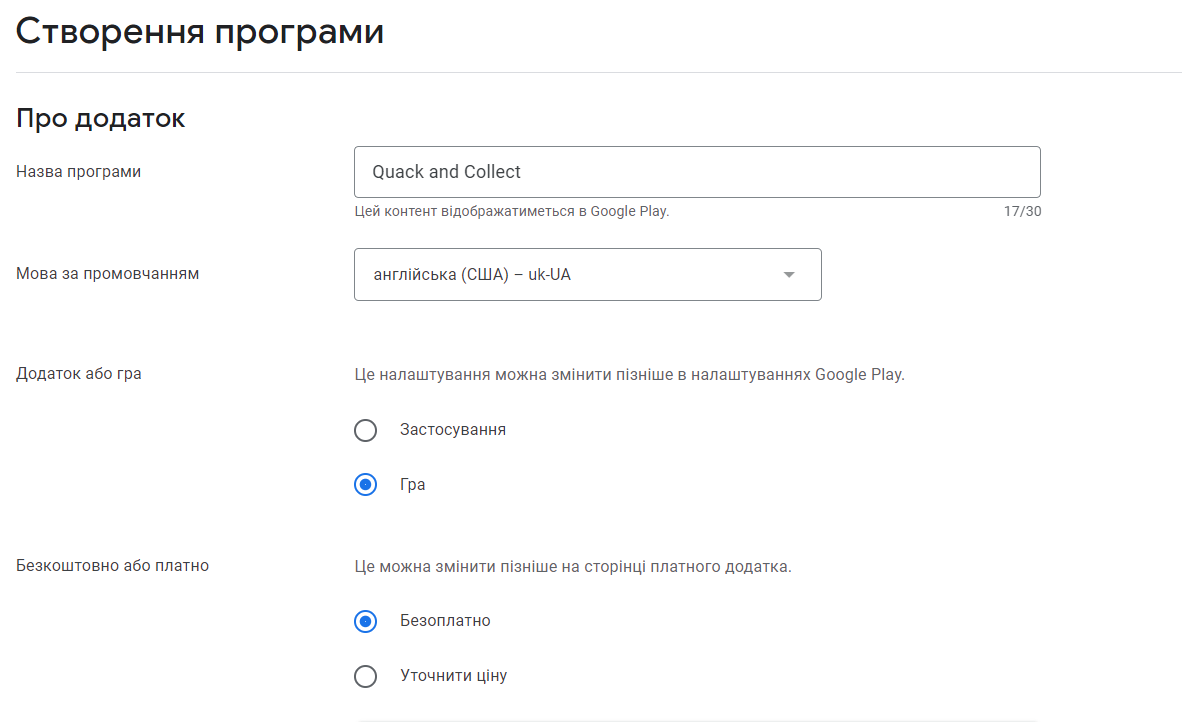


Рисунок 4.1 – Створення нового проєкту

На даному етапі вже можна проводити внутрішнє (для обмеженого кола осіб) тестування.

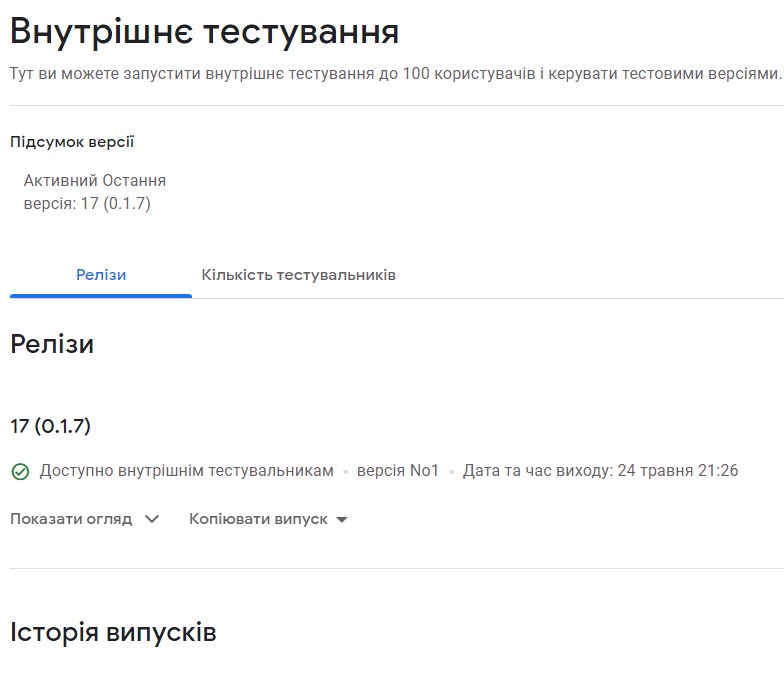


Рисунок 4.2 – Внутрішнє тестування

Потрібно оформити головну сторінку додатку в Google Play, створити лого гри та кілька картинок, які будуть на сторінці гри.



Рисунок 4.3 – Лого гри



Рисунок 4.4 – Демонстрація гри

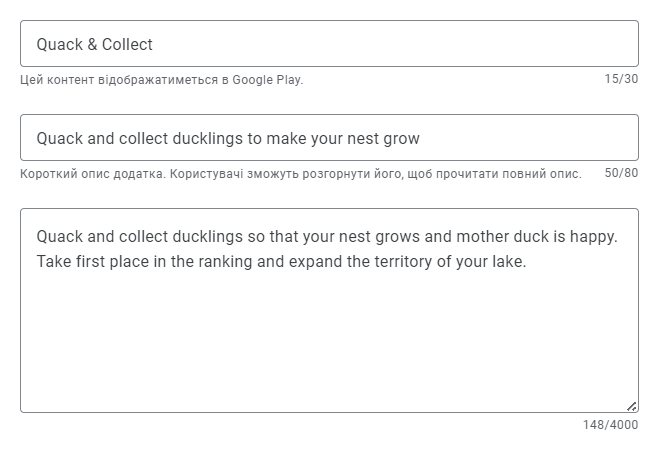


Рисунок 4.5 – Назва, короткий та повний опис гри

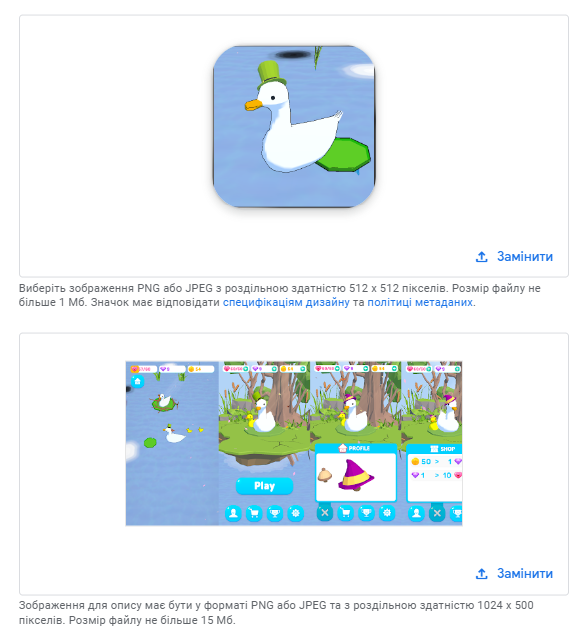


Рисунок 4.6 – Графіка публікації

Після завершення перевірки змін головної сторінки та готової версії гри, вона стане доступною у дозволених країнах (їх список можна налаштовувати у версії).

Випуск гри може бути у кількох варіантах:

* попередня реєстрація (гру встановити не можна, проте можна зареєструватися на її очікування і гравець отримає сповіщення, коли гра вийде у доступ);
* внутрішнє тестування (без очікування перевірки версії її можуть перевірити вибрані тестувальники до 100 осіб);
* закрите тестування (без очікування перевірки версії можна робити кільках груп тестувальників і надавати їм доступ до різної версії гри);
* відкрите тестування (очікується перевірка версії, всі бажаючі можуть вступити до групи тестувальників (кількість тестувальників регулюється) та пограти, оцінки ставити неможна);
* робоча версія (публічна версія для всіх у дозволених регіонах, оцінки відкриті).

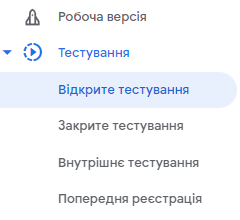


Рисунок 4.7 – Типи випусків

# ВИСНОВКИ

1. Аналізовано актуальність гіпер-казуальних ігор на ринку геймдеву,

виявлено, що жанр є популярними завдяки простому інтерфейсу та оригінальним рішенням. У цьому контексті Unity є найкращим вибором для розробки таких ігор, завдяки своїй широкій платформній підтримці, легкості використання та розширеним можливостям. Завдяки цим перевагам, ви зможете успішно реалізувати вашу ідею та залучити аудиторію.

1. Проведена підготовка до роботи з проєктом. Було встановлено

програмне забезпечення Unity 2022 та Visual Studio 2022, що є основними інструментами для розробки ігор. Також була проведена робота з документацією Unity, що дозволило ознайомитись з основними функціями та можливостями движку. Було вивчено роботу основних компонентів движку, що є важливим етапом у розумінні його функціоналу. На завершення, було створено скрипти та вивчено їх роботу, що є ключовим елементом реалізації логіки гри.

1. При виконанні всіх пунктів роботи вже маємо вигляд сучасної гри

Google Play. Виконані основні елементи сучасної гри:

* контроллер гравця;
* графіка;
* меню;
* збереження даних;
* кастомізація;
* реклама.

1. У результаті даної роботи існує гра на платформі Google Play, в яку вже

зіграли на момент написання тексту більше 100-ти людей. Гра доступна для всіх бажаючих спробувати.

Посилання гри “Quack & Collect” на Google Play

URL: <https://play.google.com/store/apps/details?id=com.GaFeam.QuackCollect>.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Документація Unity User Manual 2022.2 [Електронний ресурс].

URL: <https://docs.unity3d.com/2022.2/Documentation/Manual/UnityManual.html> (дата звернення: 03.03.2023).

1. Документація Unity | Google for Developers (Mobile Ads SDK Unity)

[Електронний ресурс].

URL: <https://developers.google.com/admob/unity/quick-start> (дата звернення: 17.03.2023).

1. Ігровий движок Unity | Download the Unity Hub[Електронний ресурс].

URL: <https://unity.com/download> (дата звернення: 01.03.2023).

1. Платформа мобільних додатків Google Play [Електронний ресурс].

URL: <https://play.google.com/store/games?hl=ru&gl=US> (дата звернення: 05.03.2023).

1. Платформа пакетів Unity Asset Store [Електронний ресурс].

URL: <https://assetstore.unity.com/> (дата звернення: 02.03.2023).

**Додаток А**

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

.cs – це розширення файлу, який використовується для файлів, що містять код на мові програмування C#.

Камера – об'єкт на сцені, який показує те, що входить в його поле зору.

Прототип – примітивна версія гри з усіма основними механіка, потрібен для оцінки проєкту вцілому.

Персонаж – елемент гри, якого гравець контролює безпосередньо.

Джойстик – елемент на екрані, з допомогою якого гравець може контролювати свого персонажа у грі.

.json – це розширення файлу, який використовується для для файлів, що містять текст (аналог текстового розширення .txt).

Pool – це набір готових до використання об’єктів (коли потрібен об’єкт, він не створюється, а береться з Pool-а).