

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інженерії програмного забезпечення

ПОЯСНЮВАЛЬНА ЗАПИСКА

Бакалаврська роботи

на ступінь вищої освіти бакалавр

на тему:

«Розробка відеогри «Immortal heroes» мовою C#»

Виконав: студент 5 курсу, групи ППЗ–51

спеціальності

121 – Інженерія програмного забезпечення

Конкін В.О.

Керівник Жебка В.В.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Нормоконтроль _____

(прізвище та ініціали)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти – «Бакалавр»

Напрямок підготовки -121 – “Інженерія програмного забезпечення”

ЗАТВЕРДЖУЮ

Завідувач кафедри Інженерії
програмного забезпечення

Негоденко О.В.

« ____ » _____ 2022 року

ЗАВДАННЯ

НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Конкін Владислав Олександрович

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка відеогри «Immortal heroes» мовою C#»

Керівник роботи: д.т.н. Жебка Вікторія Вікторівна

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “18” лютого 2022 року №22.

- | | Строк |
|---|-------|
| 1. подання студентом роботи 03.06.2022 _____ | |
| 2. Вхідні дані до роботи: | |
| 2.1 Положення побудови гри | |
| 2.2 Методи побудови гри; | |
| 2.3 Розробка моделі гри; | |
| 2.4 Науково-технічна література _____ | |
| 4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити). | |
| 4.1 Загальні положення побудови гри; | |
| 4.2 Аналіз технології і методів побудови гри; | |
| 4.3 Дослідження даних користувачів. | |
| 5. Перелік графічного матеріалу | |
| 1_Аналоги мобільної гри | |
| 2.Ігрові двигуни | |
| 3.Інтерфейс Unity | |
| 4.Аналітичні сервіси | |

б. Дата видачі завдання 11.04.2022

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	11.04-14.04	Виконано
2	Аналіз ринку розробки мобільних ігор	15.04-20.04	Виконано
3	Розробка Immortal Heroes	21.04-2.05	Виконано
4	Тестування гри	3.05-10.05	Виконано
5	Написання висновків та підготовка презентації	11.05-15.05	Виконано
6	Попередній захист роботи	16.05-1.06	
7	Подання роботи в деканат	3.06	

Студент _____ **Конкін В.О.**

(підпис) (прізвище та ініціали)

Керівник роботи _____ **Жебка В.В.**

(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи 55с., 41 рис., 17 джерел.

Ключові слова: ГРА, ІГРОВА ІНДУСТРІЯ, ІГРОВИЙ ДВИГУН, Unity, C#, СИСТЕМА АНАЛІТИКИ.

Об'єкт дослідження - процес розробки гри жанру RogueLike мовою C# на двигуні Unity.

Предмет дослідження - методи за засоби створення гри жанру RogueLike Immortal Heroes.

Мета роботи - проходження процесу накопичення даних поведінки користувачів в умовах гри та подальший її аналіз.

Галузь використання – користувачі мобільних пристроїв з системою Android.

Для реалізації мети було сформульовано та вирішено наступні завдання:

1. Провести аналіз схожих робіт.
2. Дослідити середовище розробки Unity3d та мову C#.
3. Розробити гру для проходження тестування.
4. Провести тестування. Предметом дослідження є гра Immortal heroes.

Стислий опис результатів дослідження: реалізовано гру за допомогою ігрового двигуна Unity та мови програмування C#, проведено тестування гри на конкурентному ринку, зібрано та проаналізовано поведінку більш ніж 1000 користувачів.

ЗМІСТ

ВСТУП	9
1. АНАЛІЗ РИНКУ РОЗРОБКИ МОБІЛЬНИХ ІГОР	11
1.1 Аналіз Ринку Відеоігор	11
1.2 Вибір жанру гри	15
2 МЕТОДИ РОЗРОБКИ IMMORTAL HEROES	23
2.1 Вибір Ігрового Двигуна	23
2.1.1 Unity	24
2.1.2 Unreal Engine	26
2.1.3 Cryengine	28
2.1.4 Godot	31
2.1.5 Порівняння ігрових двигунів	33
2.2 Проектування та створення програмного продукту	34
2.3 Архітектура мобільної гри	44
2.4 Компіляція проекту під Android та її публікація	48
2.4.1 Компіляція проекту під Android	48
2.4.2 Публікація проекту	50
3 ТЕСТУВАННЯ ГРИ ВИХОДЯЧИ З ДАНИХ АНАЛІТИЧНИХ СЕРВІСІВ	53
3.1 Аналітичний сервіс Firebase	53
3.2 Тестування гри виходячи з отриманих даних	57
4 ВИСНОВКИ	61
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛ	64

ВСТУП

Для багатьох людей відеоігри давно стали повсякденною частиною життя як новий вид хобі для задоволення соціальних і емоційних потреб. Суть відеоігор у тому, щоб передати той або інший досвід і разом з ним емоційний відгук.

Відеоігри самі по собі дуже різноманітні, вони різняться жанрами, стилями, підходами розповісти ту чи іншу історію, за це дехто вважає їх окремою галуззю мистецтва. Вони затягують нас, переносять в незвичайні та незнайомі умови, змушує активно взаємодіяти з середовищем, вирішувати завдання. Завдяки відеоіграм люди вчаться і відкривають для себе нове, при цьому відпочивають.

Актуальність теми. У сучасному суспільстві важлива кожна хвилина, коли комп'ютерні або консольні відеоігри потребують багато часу на порятунок приходять мобільні ігри. На цей час ринок мобільних ігор займає 49% від індустрії відеоігор. Мобільні відеоігри стають все популярнішими, завдяки розвитку мобільних пристроїв, та приваблює приблизно 15% нової аудиторії щорічно.

Об'єкт дослідження - процес розробки гри жанру RogueLike мовою C# на двигуні Unity.

Предмет дослідження - методи за засоби створення гри жанру RogueLike Immortal Heroes.

Мета роботи - проходження процесу накопичення даних поведінки користувачів в умовах гри та подальший її аналіз.

Методи дослідження – концепція технології створення ігрового мобільного додатку, розбиття його на рівні, та опис розроблених ігрових механік, які були описані в ході розробки.

Для реалізації мети було сформульовано та вирішено наступні завдання:

1. Провести аналіз схожих робіт.

2. Дослідити середовище розробки Unity3d та мову C#.
3. Розробити гру для проходження тестування.
4. Провести тестування.

Практичне значення результатів полягає у написанні функціоналу гри Immortal heroes з використання ігрового двигуна UNITY та мови програмування C# на мобільну платформу.

В роботі розглянуто основні етапи створення ігор і досліджено можливості технічних засобів для розробки гри.

1. АНАЛІЗ РИНКУ РОЗРОБКИ МОБІЛЬНИХ ІГОР

1.1 Аналіз Ринку Відеоігор

У 2021 році ігровий ринок оцінювався в 198,40 мільярдів доларів США, а до 2027 року він, як очікується, досягне 339,95 мільярда доларів, зареєструвавши сукупний середньорічний темп зростання 8,94% протягом 2022-2027 років. Через пандемію COVID-19, деякі люди звернулися до ігрових платформ, щоб скоротити час. Таким чином, ці платформи привернули до онлайн-трафіку сотні й тисячі нових відвідувачів. Останнім часом тенденції відеоігор відчули величезний сплеск гравців і доходів. Це було досягнуто здебільш завдяки:

- Постійні технологічні досягнення в ігровій індустрії значно сприяють зростанню індустрії. Вони покращують спосіб створення ігор і покращують загальний ігровий досвід користувачів.
- Розробники ігор у країнах, що розвиваються, постійно прагнуть покращити ігровий досвід, запускаючи та переписуючи коди для різноманітних консолей/платформ, таких як PlayStation, Xbox та Windows PC, які входять до окремого продукту, який надається геймерам через хмарну платформу.
- Зростання підключення до Інтернету, зростання популярності смартфонів і поява високошвидкісного мережевого підключення, такого як 5G, ще більше підвищили попит на ігровому ринку в усьому світі. Крім того, згідно зі звітом GSMA, станом на листопад 2021 року 170 операторів мобільного зв'язку запустили комерційні послуги 5G з охопленням населення 7% наприкінці 2021 року, що відкриває нові можливості для постачальників мобільного зв'язку для впровадження на ринок смартфонів 5G.

- Крім того, згідно зі статистикою, опублікованою DataReporal, кількість користувачів Інтернету в 2021 році зросла на 7,7% порівняно з 2020 роком. Більше того, у січні 2022 року їх кількість зросла на 4% до 4950 мільйонів порівняно з січнем 2021 року, коли їх було 4758 мільйонів.
- Хмарні ігрові сервіси зосереджені на використанні гіпермасштабних хмарних можливостей, потокових медіа-сервісів і глобальних мереж доставки контенту для створення нового покоління соціальних розважальних платформ. Ці фактори мають передбачуваний позитивний вплив на зростання ринку.
- Крім того, використання хмарних технологій на ігровому ринку, ймовірно, сприятиме попити та залученню кількох гравців до різних ігор, стимулюючи зростання ринку протягом прогнозованого періоду.

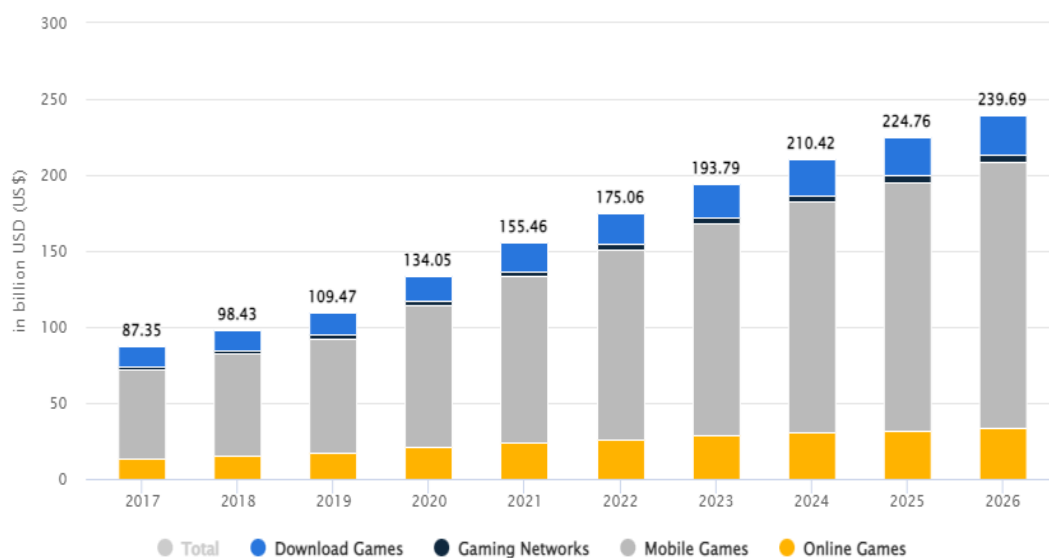


Рисунок 1.1.1 – Ріст ринку відеоігор

Мобільні ігри є найпопулярнішим видом ігор у всьому світі, випереджаючи ігри як на консолях, так і на ПК. Однією з головних причин

популярності мобільних ігор є доступність. Майже у кожного є смартфон з іграми.

У обговореннях в який момент зародився мобільний часто стверджується, що iPhone від Apple (2007) зробив революцію у всій індустрії. Примітно, що ця трансформація не була викликана жодним технологічним проривом. Усі елементи оригінального iPhone були доступні раніше, але новий досвід користувача, наданий дуже спрощеним інтерфейсом користувача на сенсорному екрані, зробив вплив iPhone більшим, ніж сума його частин. Останній ключовий елемент успіху iPhone був представлений у 2008 році: App Store. До появи Apple iOS App Store було кілька спроб створити інтернет-магазини для продажу та цифрового розповсюдження програмного забезпечення, але в поєднанні зі зростанням популярності iPhone, мобільна «екосистема додатків» раптово вибухнула. Кількість доступних додатків для багатьох індивідуальних випадків заворожувала.

Вбудований магазин iPhone виріс із початкових 500 додатків до мільйонів, які доступні сьогодні. Були паралельні події, зокрема в екосистемі Android від Google (також вперше представлена в 2008 році). Завдяки новим поколінням стільникових мереж (3G, 4G, 5G), потужним і відносно доступним споживачеві пристроям із сенсорним екраном та постійно оновлюваному програмному забезпеченню та послугам, сучасний інтерфейс мобільних ігор виглядає зовсім інакше, ніж на рубежі століть. Відповідно змінилося розуміння того, що таке мобільні ігри та що вони можуть виглядати. Ранні етапи дизайну мобільних ігор часто були зосереджені на мініатюризації та спрощенні існуючих відеоігор, в основному через обмеження доступної обчислювальної потужності, пам'яті та обмежених інтерфейсу користувача в невеликих портативних пристроях. У міру того, як дослідження та розробки просувалися вперед, ставало все більш очевидним, що мобільні ігри можуть мати деякі унікальні переваги, яких не можуть забезпечити інші ігрові платформи. Хоча значна частина комерційної енергії для розробки мобільних ігор була спрямована на подолання численних

технічних і бізнес-задач у цьому секторі промисловості, що розвивається, інтерес дослідників і новаторів був на ранньому етапі націлювання на те, що може бути справді «мобільною грою». Це було засновано на усвідомленні того, що мобільні телефони створені насамперед для спілкування та підтримки соціальних відносин. Крім того, оскільки пристрій тісно пов'язаний із повсякденним життям і рухами людей, майбутній потенціал може полягати в різних ігрових програмах, які враховують таку інформацію, як час, місце або ситуаційний контекст, пропонуючи нові ігрові можливості.

Зростаючий попит на мобільні ігри є результатом різноманітних технологічних досягнень та вдосконалень, таких як AR, VR, хмарні ігри та 5G.

Мобільні ігри мають кілька структурних особливостей, які відрізняють їх від ігор на комп'ютерах і консолях, включаючи портативність, більший акцент на коротких подіях і легкий у засвоєнні повторюваний ігровий процес, а також варіанти закупівлі в додатку. Відмінності між ігровими платформами можуть обмежувати цінність дослідження проблеми консолі та комп'ютерних ігор до проблем мобільних ігор, де, наприклад, може бути більше уваги зосереджено на аватарах, розповіді, складних механік або елементів керування, а також командній роботі та інших соціальних особливостей. Отже, цікавим є питання, чи демонструє більша участь у мобільних іграх подібну позитивну асоціацію *toproblem gaming* (PG) до того, що спостерігалось в дослідженнях консольних і комп'ютерних геймерів

AR стає ідеальним для мобільних ігор завдяки своїй захоплюючій та інтерактивній технології. Більше того, мобільні ігри є найвідомішою категорією AR у магазинах додатків. Окрім раніше випущених мобільних ігор AR, які все ще відомі, таких як *Pokemon Go* та *Ingress*, розробники на світовому ринку вносять багато нових доповнень до цього жанру.

Протягом 2020 та 2021 років у секторі розробки мобільних ігор відбулося багато змін. Отже, ігрові бренди витрачають більше на рекламні інвестиції, щоб залучити більше користувачів і створити достатню частку

ринку. Однак одним з ефективних способів зниження витрат на процес розробки є проведення раннього тесту CTR (рейтинг кліків). Це особливо вірно, якщо ви перебуваєте в секторі «гіперказуальних ігор» індустрії мобільних ігор.

Гендерна статистика мобільних ігор показує, що 4,5% жінок, які грають в мобільні ігри, платять додатково за вміст ігор, порівняно з 3,3% чоловіків. За даними Unity Technologies, 75-й перцентиль 30-денного утримання користувачів покращився на 7,1% з 2016 по 2020 рік. Крім того, коефіцієнт конверсії для покупок у мобільних відеоіграх зріс приблизно на 7,7% за той же період.

1.2 Вибір жанру гри

Ідей стільки, скільки людей. Кожному подобаються різні ідеї, тому люди мають різні цілі для досягнення при грі на мобільних пристроях. Багато розробників створюють ігри, щоб мати змогу отримувати прибуток з гри для Android. Але не всі створюють відеоігри з тієї ж причини. У деяких випадків розробники створюють додаток не для того, щоб почати бізнес, а для можливості передати досвід гри, який він вважає цікавим. Тому причини розробки програми можуть відрізнятися від розробника до розробника, але одна річ однакова. Кожен розробник створює додаток, щоб все більше і більше аудиторії використовувало його, і в результаті він отримував більше доходу. Завоювати аудиторію та заробляти гроші – це те, чого неодмінно хоче будь-який розробник, незалежно від того, з якою метою він створив свій додаток.

Якщо ви хочете отримувати все більше і більше користувачів, то вам не слід покладатися вдачу, шанс що вашу гру помітять серед мільйонів інших ігор занадто мала. За кожного користувача і аудиторію вам, як розробнику, потрібно думати. Вам потрібно подумати, щоб почати розробляти більшу кількість ігор та додатків для можливого влучання в потрібну аудиторію. Це

допоможе вам отримати розумний прибуток, а завдяки такій кількості хороших програм ви заробите гарне ім'я у світі розробки відеоігор. Тому, щоб заробляти більше, ви повинні створити багато відеоігор. Ви можете створювати велику кількість ігор в надії на влучання, або використовувати спеціальні сервіси аналітики.

Для розповсюдження мобільних ігор використовуються магазини додатків, найпопулярнішими являються Google Play та App Store. Для більш ефективного розповсюдження додатків в магазинах використовується система рекомендацій. Базуючись на даних які система збирає вона завдяки алгоритму групує подібних користувачів по аудиторіям та виходячи з цього надалі рекомендує додатки які мають більший коефіцієнт переходів та встановлення. Ми повинні переконатися, що те, що ви створюєте, відповідає вимогам, які потрібні більшій частині користувачів.

Для того щоб дізнатися потреби користувачів можна скористуватися сервісами аналітики. Сервіс аналітики – сервіс для отримання, порівняння даних з магазинів додатків та отримання рекомендацій щодо покращення вашого додатку в системі рекомендацій.

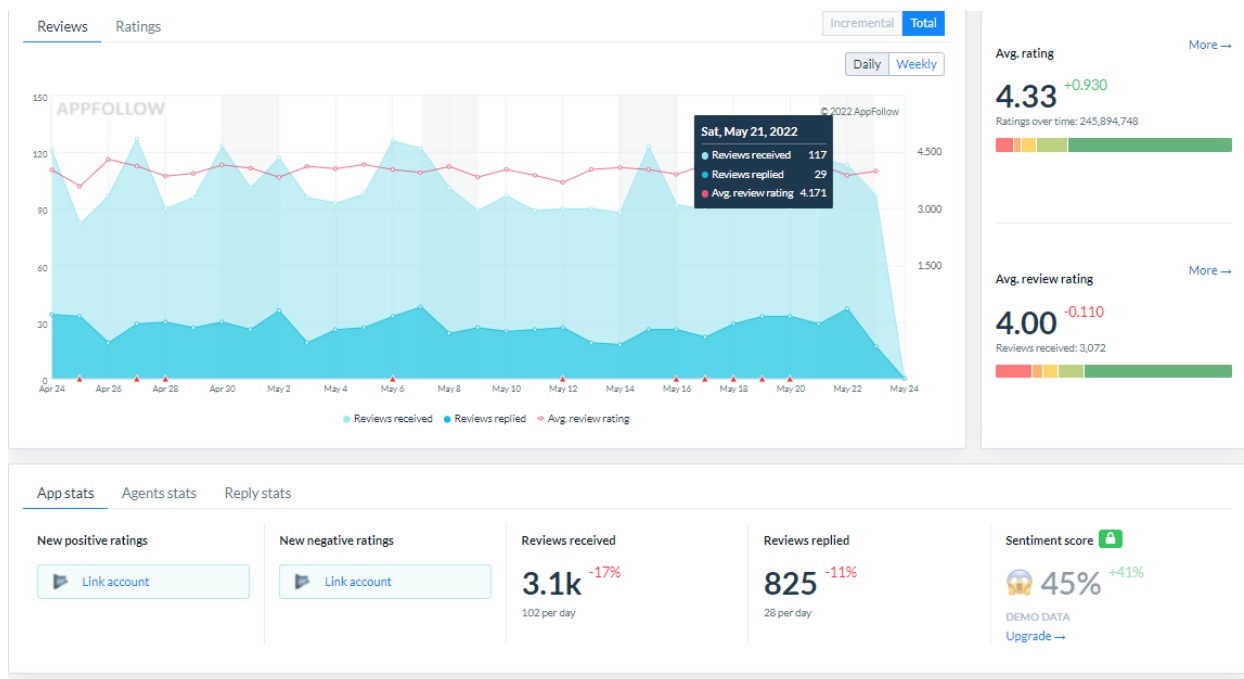


Рис 1.2.1 – Сервіс аналітики Appfollow

Для того щоб дізнатися чи може стати гра популярною не обов'язково її розроблювати, перед початком розробки потрібно провести аналіз ринку. Для цього зазвичай потрібно знайти схожі на вашу ідею ігри. Проаналізувати за допомогою яких слів їх знаходять і чи зможете скласти їм конкуренцію.

Keyword Rankings Update now

Hide Unpopular Translation Select Move to Columns: 3/7

Filter by Keyword	POPULARITY	DIFFICULTY	EFFECTIVENESS	RANK	CHANGE	BEST RANK	% OF SEARCH DOWNLOADS	APPS
mobile	75	100	3	-	-	-	-	249
offline games	73	58	17	-	-	-	-	249
little	72	80	9	-	-	-	-	249
dark	71	71	10	26	+4	11-15	-	249
heroes	69	61	10	Calculating	Calculating	-	-	-
elder scrolls	68	32	38	-	-	46	-	249
legends	67	75	10	-	-	-	-	249
roguelike	66	38	36	41	+4	20-21	-	249
offline games for free	65	53	19	Calculating	Calculating	-	-	-
dungeon	65	54	18	-	-	9	-	249
archery	65	79	9	Calculating	Calculating	-	-	-
adventure	65	82	4	-	-	-	-	249
awesome games	64	66	11	-	-	-	-	249
alchemy	64	39	36	-	-	-	-	249
single player games	63	55	19	-	-	-	-	249

Рис 1.2.2 – Таблиця запитів для знаходження додатків

Слова для пошуку додатків мають 3 основні характеристики:

- Популярність – характеристика яка відповідає за кількість людей які можуть бути зацікавлені цим запитом.
- Складність – характеристика яка відповідає за кількість додатків які вже використовують цей запит.
- Ефективність – характеристика яка відповідає за ефективність використання цього запита для подальшої конверсії.

Для аналізу існуючих мобільних ігор потрібно використовувати схожі між собою жанри, не має сенсу використовувати для аналізу мобільні ігри які не мають між собою майже нічого спільного. Користувачей можна поділити на 3 великі групи:

- гіпер-казуали

- казуали
- мідкорщики

В першу чергу кожна наступна група має більший час життя в грі, базуючись на цьому потрібно правильно використовувати механіки, баланс час користувача. Багато користувачів грають в мобільні ігри в будь-яку вільний хвилину, в залежності від кількості вільного часу та бажання зануритися в гру ми і аналізуємо аудиторію. Головними показниками індустрії час гри та відсоток повернення користувача по дням. Для Гіперказуалів добрий час гри рахується від 20 хвилин на день, а повернення першого дня більше 40% але при цьому повернення 3 дня в 15% рахується нормою. Для казуалів гарний час гри починається від 40 хвилин на день, а повернення першого дня більше 35% при цьому повернення 3 дня в 20% рахується нормою. Для мідкорщиків зазвичай показники йдуть не на дні а на неділі та місяці так як гра такого типу розрахована на великий час находження користувача в грі, наприклад, гарний час може бути менший 40 хвилин в день, а повернення першого дня меншою з-за 35% але завдяки поверненню користувачів першої неділі в більше ніж 20% відіграє значну роль в створенні спеціалізованої аудиторії.

Для аналізу було вибрано 3 гри:

- Dungeon Knight
- Dread Rune
- Nonstop Knight 2 - Action RPG



Рисунок 1.2.3 - Dungeon Knight

Dungeon Knight – має велику кількість активних користувачів. Представляє собою гру категорії мідкор.

- Має в собі можливість грати в онлайн режимі з іншими користувачами
- Представляє собою повну залежність від вибору користувача, що несе під собою додаткову важкість для новачків
- Має в собі багаторівневу систему прогресу, що змушує користувача заходити в неї для досягання нових вершин.
- Розроблений з використанням 3д графіки
- Активна допомога при виникненні проблем у користувачів
- Часті оновлення з додаванням нового контенту

Dungeon Knight – має активну команду розробки та підтримки гри. Аудиторія гри перевищує 1 мільйон користувачів. З мінусів гра має велику вагу 138 мб в не встановленому стані, багато реклами яка впливає на відчуття потоку від гри.

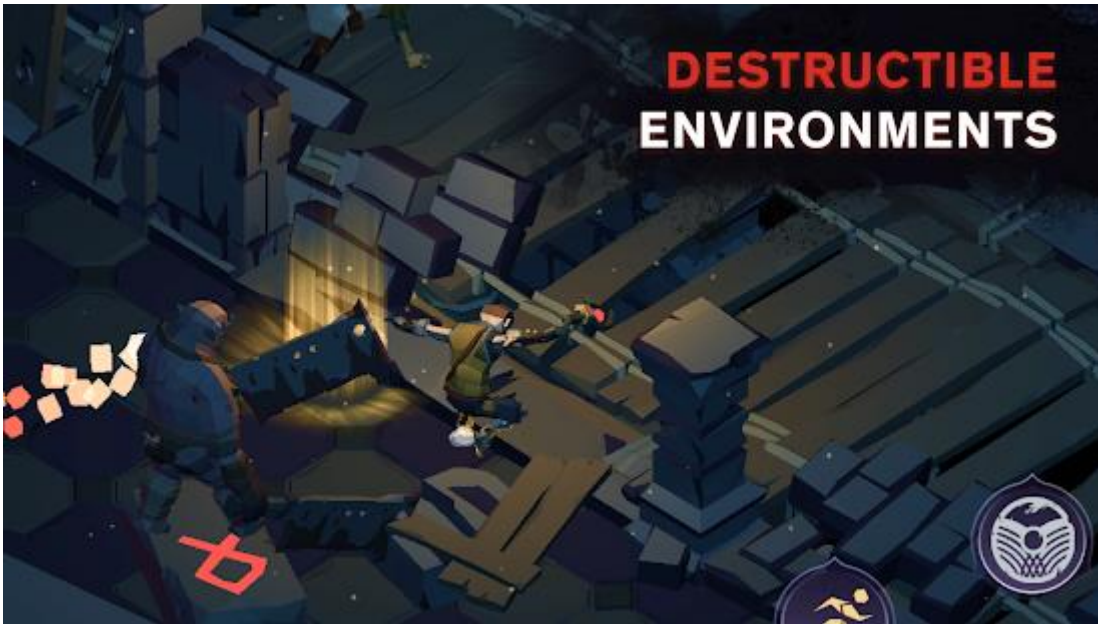


Рисунок 1.2.4 - Dread Rune

Dread Rune - має велику кількість активних користувачів. Представляє собою гру категорії мідкор.

- Представляє собою повну залежність від вибору користувача, що несе під собою додаткову важкість для новачків
- Випадкова генерація карти додає різноманіття для кожної гри
- Більшість елементів карти інтерактивна, що несе під собою додаткову залучення користувача до дослідження гри
- Фізична система переміщення персонажу завдяки кісткової анімації
- Розроблений з використанням 3д графіки
- Активна допомога при виникненні проблем у користувачів
- Часті оновлення з додаванням нового контенту

Dread Rune - має активну команду розробки та підтримки гри. Аудиторія гри перевищує 100000 користувачів. Так як гра використовує тяжку для підрахунків систему фізичної кісткової анімації гра не має рівномірну кількість кадрів, що на сам перед погіршує позитивний досвід від гри.



Рисунок 1.2.5 - Nonstop Knight 2 - Action RPG

Nonstop Knight 2 - Action RPG - має велику кількість активних користувачів. Представляє собою гру категорії казуал.

- Представляє собою майже повну відсутність залежності від вибору користувача, що несе під собою легкість для новачків
- Випадкова генерація карти додає різноманіття для кожної гри
- Більшість елементів карти інтерактивна, що несе під собою додаткову залучення користувача до дослідження гри
- Розроблений з використанням 3д графіки
- Активна допомога при виникненні проблем у користувачів
- Часті оновлення з додаванням нового контенту

Nonstop Knight 2 - Action RPG - має активну команду розробки та підтримки гри. Аудиторія гри перевищує 1 мільйона користувачів.

Гра займає казуальний тип тому залежність від дій користувача зменшена до максимуму.

Назва	Dungeon Knight	Dread Rune	Nonstop Knight 2 - Action RPG	Immortal Heroes
+	Великі рівні	Фізичні елементи мапи	Насичена графіка	Унікальні рівні
+	Велика кількість ефектів	Інерційна система переміщення	Велика кількість ворогів	Велика кількість предметів
-	Поганий інтерфейс	Погана оптимізація	Простий цикл геймплею	Мала варіативність ворогів
-	Погана оптимізація	Мала кількість предметів	Мала кількість предметів	Мала кількість персонажів

Таблиця 1.2.1 Порівняння з аналогами

2 МЕТОДИ РОЗРОБКИ IMMORTAL HEROES

2.1 Вибір Ігрового Двигуна

Що таке ігровий або ігровий движок? Ігровий движок — це середовище розробки програмного забезпечення, яке також називають «ігровою архітектурою» або «ігровим фреймворком», з налаштуваннями та конфігураціями, які оптимізують і спрощують розробку відеоігор на різних мовах програмування. Ігровий механізм може включати механізм візуалізації 2D або 3D графіки, сумісний з різними форматами імпорту, фізичний двигун, який імітує реальні дії, штучний інтелект (AI), який автоматично реагує на дії гравця, звуковий двигун, який керує звуковими ефектами, двигун анімації та безліч інших функцій.

Чому ігрові движки важливі?

Ранні відеоігри були розроблені з власними двигунами візуалізації, кожен спеціально розроблений для однієї гри. З часом ігрові движки еволюціонували від запатентованих власних двигунів до комерційно розроблених движків, які сьогодні широко доступні. Розробники ігор, які користуються надзвичайною попитом, можуть спростити та прискорити процес розробки ігор, використовуючи комерційно розроблені ігрові движки для створення нових ігор або розширення існуючих ігор на додаткові платформи.

На сьогодні існує багато двигунів під будь-які потреби. Розглянемо найпопулярніші.

2.1.1 Unity



Рис 2.1.1.1 – Логотип Unity

Unity, добре розроблений, комплексний інструмент розробки ігор, також має свої недоліки серед численних позитивних моментів. Деякі переваги та недоліки наведені нижче, узагальнено серед інших розробників Unity Game, але, безперечно, не обмежуючись:

Плюси:

- Шалено висока швидкість розробки гри Unity. Це слідує за швидким створенням ігор і сприяє швидкому створенню прототипів і безперервному випуску.
- Простий та швидкий процес імпорту підсистеми ресурсів, що веде до оптимізованого уніфікованого конвеєра активів і підтримує більшість пакетів 3D-пакетів із зображеннями, аудіо, відео та текстовими форматами.
- Чудовий інтегрований редактор рівнів із підтримкою JavaScript та C# для написання сценаріїв.
- Покращена підтримка усунення помилок і налаштувань, оскільки всі ігрові змінні відображаються під час гри, що дозволяє систематично налаштовувати та налагоджувати процеси під час виконання.

- Величезні спільноти та маркетплейси 3D-ігор Unity, які пропонують широкі вбудовані компоненти для звуку, фізики, візуалізації, елементів керування тощо. Магазин активів дуже нагадує будь-які магазини додатків для телефону.
- Багатопоточність, колекції, функції введення-виводу та виразні функції бібліотеки LINQ
- Багатоплатформні платформи для розробки та розгортання ігор Unity для консолей, настільних комп'ютерів, браузерів і мобільних пристроїв.
- Чудова та проста реалізація об'ємного 3D/панорамування та аудіосистема.

Мінуси:

- Забезпечує більше пам'яті, це може спричинити помилки нестачі пам'яті на мобільних пристроях та проблеми з налагодженням.
- Оскільки вихідні коди не надаються, проблеми з продуктивністю важко знайти, вирішити та виправити.
- Розробка великої гри AAA потребує багато оптимізації.
- Легко деорганізувати свою ієрархічну структуру і має занадто багато папок, активів тощо.
- Зміна цілей збірки вимагає повторного імпорту всього та витрачання часу під час роботи над багатоплатформною грою.

Загальна оцінка UNITY серед розробників ігор Unity і ігрових двигунів дійсно оцінені дуже високо, і більшість користувачів розробили численні способи обходу недоліків, чудові магазини ігрових активів. Варіанти розробки та конкуренція з Unreal, CRY, безумовно, підштовхнуть UNITY до подальшого вивчення нових горизонтів у сфері розробки ігрових додатків.

2.1.2 Unreal Engine



Рис 2.1.2.1 – Логотип Unreal Engine

Ігровий движок Unreal Engine був розроблений компанією Epic Games. Вперше він був показаний в шутері від першої особи Unreal 1998 року. Хоча він був розроблений в основному для шутерів від першої особи, він ефективно використовувався в багатьох інших жанрах, включаючи стелс, файтинги, MMORPG та інші RPG. Код написаний на C++, Unreal Engine має високу портативність і сьогодні також використовується багатьма розробниками ігор. Unreal Engine отримав кілька нагород і навіть отримав нагороду Книги рекордів Гіннеса як «найуспішніший движок відеоігор у 2014 році.

Що таке Unreal Engine?

Unreal Engine — це велика група інструментів для розробки ігор, створених для тих, хто працює з технологіями реального часу. Починаючи

від корпоративних додатків і кінематографа до високоякісних ігор для ПК, консолей, мобільних пристроїв, віртуальної та доповненої реальності, Unreal Engine дає вам усе, що вам потрібно, щоб почати розроблювати відеоігри.

Набір інструментів світового класу та легкий доступ до робочих процесів дозволяють розробникам швидко переробляти ідеї та бачити миттєві результати, а право введення повного базового коду дає кожному в суспільстві Unreal Engine свободу налаштувати та розширення характеристик двигуна.

Можливі платформи випуск гри на Unreal Engine: Microsoft Windows, macOS, Linux, SteamOS, HTML5, iOS, Android, Nintendo Switch, PlayStation 4, Xbox One, Magic Leap One та віртуальної реальності (SteamVR/HTC Vive , Oculus Rift, PlayStation VR, Google Daydream, OSVR і Samsung Gear VR.

Плюси Unreal Engine

- Unreal Engine має кращі характеристики для багатьох типів графіки. Таким чином цей движок відомий створенням високотехнологічних ігор AAA класу, в якій головну роль відіграє досягнення найкращої графіки.
- У Unreal є змога розроблювати логіки завдяки графічному програмуванню, так що навіть художники без незначних знань зможуть використовувати движок для створення базової логіки. Також ця функція підходить для створення прототипів. Але для основної роботи все одно доведеться використовувати C++.
- Завдяки різним функціям технологія рендеринга ефективніша та швидша, ніж інші ігри. Цей ігровий движок насправді працює досить швидко. Для оптимізації гри це найкращий інструмент.

Мінуси Unreal Engine

- Для створення простих ігор Unreal engine не підходить. Навіть для короткострокових завдань нереальний двигун не можна використовувати. Для проектів, які мають тривалий термін

експлуатації, найкраще підходить нереальний двигун, але вартість такої роботи буде значно вищою за інше джерело.

- Для людини без команди цей ігровий двигун не підходить.
- Коли справа доходить до візуалізації або виконання збірки, комп'ютер не можна використовувати для іншого процесу. Також користувацькі інтерфейси трохи заплутані.
- Служба підтримки клієнтів майже не існує. Крім того, деякі плагіни та інструменти від сторонніх розробників надзвичайно дорогі та завищені, що може змусити методи хакерів вирішувати проблеми.

З точки зору візуальної картинки, Unreal Engine чудово підходить для ігор. Однак при роботі над комерційними іграми проекти Unreal Engine можуть вийти з-під контролю. Існує багато причин, чому створення нової гри в Unreal може бути приголомшливим або навіть гарною ідеєю. Unreal Engine може бути хорошим інструментом, якщо вся ваша команда знайома з ним та залежно від типу гри, яку ви створюєте.

2.1.3 Cryengine



Рис 2.1.3.1 Логотип CryEngine

CryEngine відомий як двигун, який працював над оригінальним Far Cry – Crytek не просто постачав технологію, а й розробив гру, і її

партнерство з Ubisoft на початку 2000-х стало тим, що прискорило зростання CryEngine. Вся франшиза Far Cry заснована на технічній демонстрації, створеній Crytek, щоб продемонструвати можливості графічного процесора Nvidia.

Коли Far Cry був запущений в 2004 році, Crytek також дозволив ліцензувати свій движок і підписав партнерство з EA для розробки того, що стане однією з її найвідоміших франшиз і почерком для CryEngine: Crysis.

До цього дня CryEngine все ще досягає успіхів у фотореалістичних шутерах та іграх від першої особи загалом.

Але він жодним чином не обмежується шутерами і підтримує широкий спектр жанрів і форматів. Він також дозволяє розробляти різні платформи, включаючи PlayStation 4, Xbox One, Windows, Linux, Oculus Rift, OSVR, PSVR і HTC Vive. У березні 2020 року було оголошено про підтримку Android.

При використанні CryEngine також є можливість в отриманні доступу як до движка, так і до його базового коду редактора, що забезпечує більше налаштувань та гнучкості. Багато студій розробників використовують сильно модифіковану версію CryEngine - наприклад, у Warhorse Studios Kingdom Come: Deliverance або Ubisoft з франшизою Far Cry, яка все ще створюється за допомогою модифікованої версії CryEngine під назвою Dunia Engine.

Плюси CryEngine

- CryEngine має велику кількість функцій із коробки. Незважаючи на те, що CryEngine має деякі галузі спеціалізації, що полегшує створення будь-якого проекту.
- CryEngine дозволяє створювати зображення високої. CryEngine відомий якістю візуальних зображень, тому, якщо ви розробляєте фотореалістичну гру, вона може бути особливо підходящою для вашого проекту. Розширений інструмент візуалізації движка добре

оптимізований і створює чудові візуальні елементи — саме це спонукає розробникам використовувати його.

- CryEngine забезпечує швидкий процес ітерації. Наявність такої кількості доступних функцій прямо з коробки в поєднанні з візуальною силою робить CryEngine чудовим інструментом для швидкого повторення проектів і отримання прототипів з дверей.
- CryEngine дуже добре справляється з відкритим світом, багатим рослинністю.
- CryEngine можна вважати піонером у створенні високоякісних візуальних середовищ відкритого світу, особливо з великими рівнями, що містять велику кількість рослинності. Якийсь час це був просто єдиний доступний варіант.

Мінуси CryEngine

- CryEngine не має такого масштабу, як деякі з його конкурентів. База користувачів CryEngine набагато менша, ніж Unity та Unreal, і, як наслідок, вона не має такого масштабу підтримки чи спільноти, як інші движки.
- Важко найняти досвідчених розробників CryEngine. Прямим впливом невеликої бази користувачів CryEngine є складність для студій залучити розробників, які мають досвід роботи з движком. Існують два галузеві стандарти, які люди активно намагаються вивчити, а саме Unreal і Unity, але не CryEngine.
- CryEngine не рекомендується для швидких ігор або складних RPG. Як відзначили спеціалісти, CryEngine відмінно справляється з шутерами від першої особи.

2.1.4 Godot



Рис 2.1.4.1 Логотип Godot

Для розробників, які тільки починають працювати в індустрії, завдання вибору найкращого ігрового движка може бути складним. Що стосується Godot, щоб ви могли побачити, чи це правильний ігровий движок для вашого проекту.

Ви можете переглянути наші інші докладні посібники з усіх основних ігрових движків на цій сторінці. Хоча великі ігрові движки, такі як Unity та Unreal, стають все більш доступними, інді-сцена продовжує покладатися на менші движки, які мають спільноту відданих розробників.

Ці приховані дорогоцінні камені включають Monogame і Construct, але Godot стає все більш популярним вибором. У квітні він фактично став шостим за кількістю використовуваних ігрових движків на Itch.io, обігнавши RPG Maker.

Godot розпочав своє життя в 2007 році, улюблений проект аргентинських розробників Хуана «reduz» Linietski та Ariel «punto» Manzur. Однак він був запуснений лише в 2014 році.

Godot — це безкоштовний двигун із відкритим кодом, який покладається на пожертвування зі своєї сторінки Patreon.

Godot використовує власну мову GDScript, але вона також підтримує візуальні сценарії, C# і C++. За допомогою Godot ви можете розгорнути ігри на настільних платформах, таких як Windows, macOS, Linux, UWP і Haiku, а також на мобільних платформах iOS і Android.

За останні кілька років Godot набирає силу і має продовжувати розвиватися, оскільки наступна збірка движка, яка має бути випущена цього року, забезпечить кращу підтримку 3D та багато нових функцій.

Плюси Godot

- Godot універсальний. Godot не обмежує ваші творчі зусилля, оскільки він здатний впоратися майже з кожним проектом, який ви можете придумати. Godot — це неймовірно універсальний движок, чії межі з точки зору видалення оклюзії та продуктивності 3D, ймовірно, будуть незначними через кілька місяців.
- Godot відмінно працює в 2D іграх. Корінна графіка Godot – це піксель, який надзвичайно корисний при розробці 2D-ігор із піксельним мистецтвом. Незважаючи на те, що він може працювати з усіма типами проектів, 2D – це те, де Godot чітко розкриває свої сильні сторони, що робить його порівняним з GameMaker, MonoGame і Construct. Godot насправді поставляється з окремими 2D і 3D двигунами.
- Godot має інтерфейс на основі блоків функцій, що робить його доступним. Godot має систему візуальних сценаріїв із використанням блоків, які ви можете з'єднати, що робить його доступним інструментом навіть для людей які не вміють програмувати. Ви можете просто перетягувати будь-яку інформацію за допомогою вузлів і сцен, що пришвидшує роботу.
- Godot є безкоштовним і відкритим вихідним кодом. Однією з найважливіших переваг Godot є те, що він безкоштовний і з відкритим вихідним кодом, що усуває те, що називають як «стрес ліцензій або передплати».

Мінуси Godot

- Godot не готовий до складних 3D-проектів. Як уже зазначалося, 3D – це не те, чим сяє Godot, оскільки його інструменти проектування рівнів і механізм візуалізації не такі потужні, як інші технології.
- У Godot є невелика спільнота. Як часто, коли ви маєте справу з невеликими ігровими движками, обмежена спільнота може бути проблемою. Наприклад, з таким популярним, як GameMaker, на всі ваші запитання вже часто будуть відповіді на форумах.
- Godot не підтримує консолі. Оскільки Godot є проектом з відкритим кодом, ви не можете розробляти ігри для консолей за допомогою нього. Движок повинен мати ліцензію як компанія, щоб зробити це - те, чого немає в планах для його провідних розробників.

2.1.5 Порівняння ігрових двигунів

				
Features	Unity	Unreal Engine	CryEngine	Godot
Languages	C#	C++	Lua, C++, C#	GDScript, C++, C#
VPL	Unity ECS	Blueprints	Flow Graph	Visual Script
Community	123k+	42.6k	10.2k+	24.7k+
Source Modify	Yes - Paid	Yes - Royalties	Yes - Royalties	Yes - Free
Platforms	PC, GC, WEB, MD, XR	PC, GC, WEB, MD, XR	PC, GC, WEB, MD, XR	PC, GC, WEB, MD, XR
Assets Store	AssetStore	MarketPlace	MarketPlace	Asset Library
License	EULA	EULA	EULA	MIT
Developer's Year	Unity Technologies 08.06.2005	Epic Games 28.04.1997	Crytek 02.05.2002	SFC & PLC 14.01.2014

PC - Personal Computer's / GC - Game Console's / WEB - Browser's / MD - Mobile Device's / XR - Extended Reality

Рис 2.1.5.1 Порівняння ігрових двигунів

Виходячи з проведеного аналізу найпопулярніших ігрових двигунів для розробки мобільної гри було використано ігровий двигун Unity, тому що він найкраще підходить для мобільної розробки.

2.2 Проектування та створення програмного продукту

Проект в Unity складається зі сцен на яких розташовані ігрові об'єкти з прикріпленими до них компонентами. У кожного об'єкта є обов'язковий компонент Transform, який відповідає за розташування об'єкта на сцені. Також, можуть бути підключені додаткові компоненти, як готові, так і створені користувачем. Редагування сцен виконується завдяки редактору Unity.

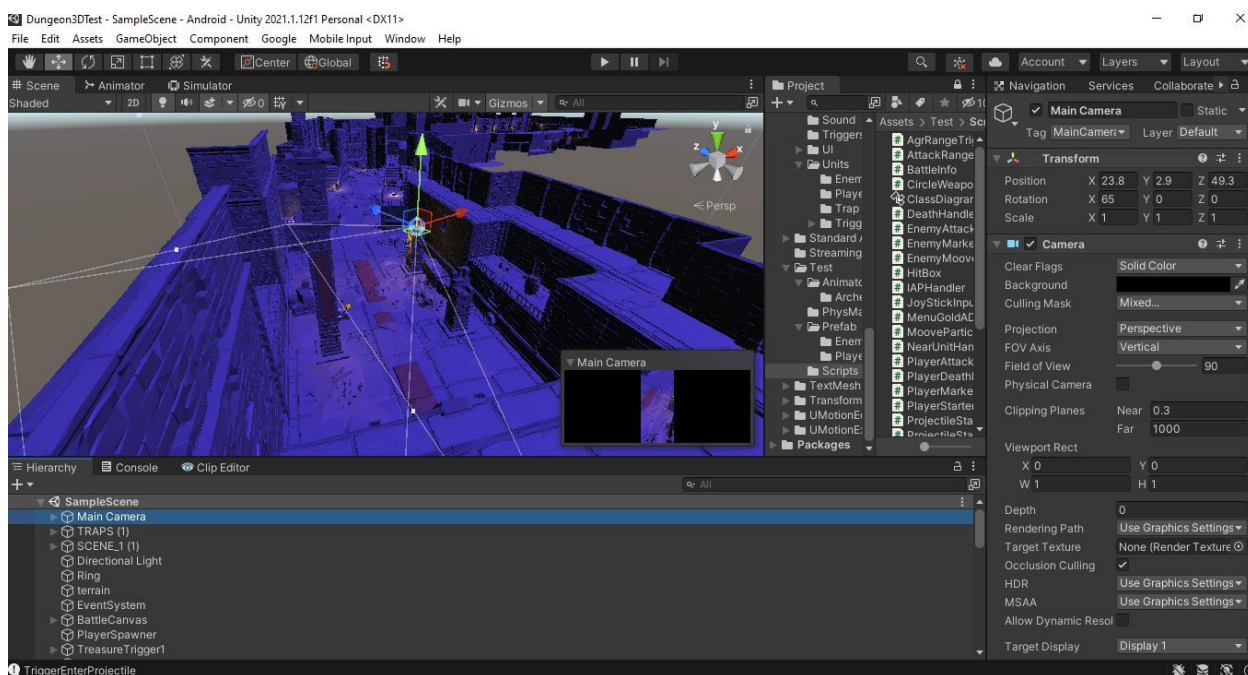


Рис 2.2.1 Редактор Unity

В Unity для реалізації компоненту використовують базовий клас MonoBehaviour. Завдяки цьому класу Unity реалізує базові функції життєвого циклу програми (Рис 2.2.2)

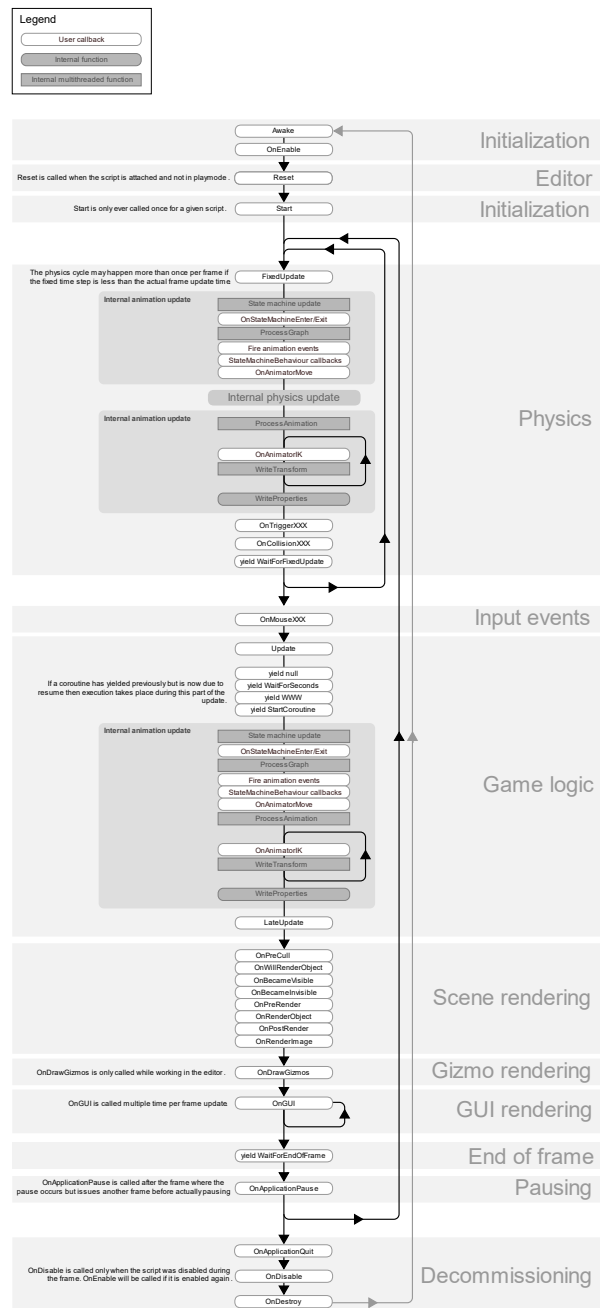


Рис 2.2.2 Життєвий цикл Unity

Для реалізації даного проекту було створено 6 окремих сцен:

- Головне меню
- 3 різних ігрові локації
- Бонусний рівень
- Арена з босом

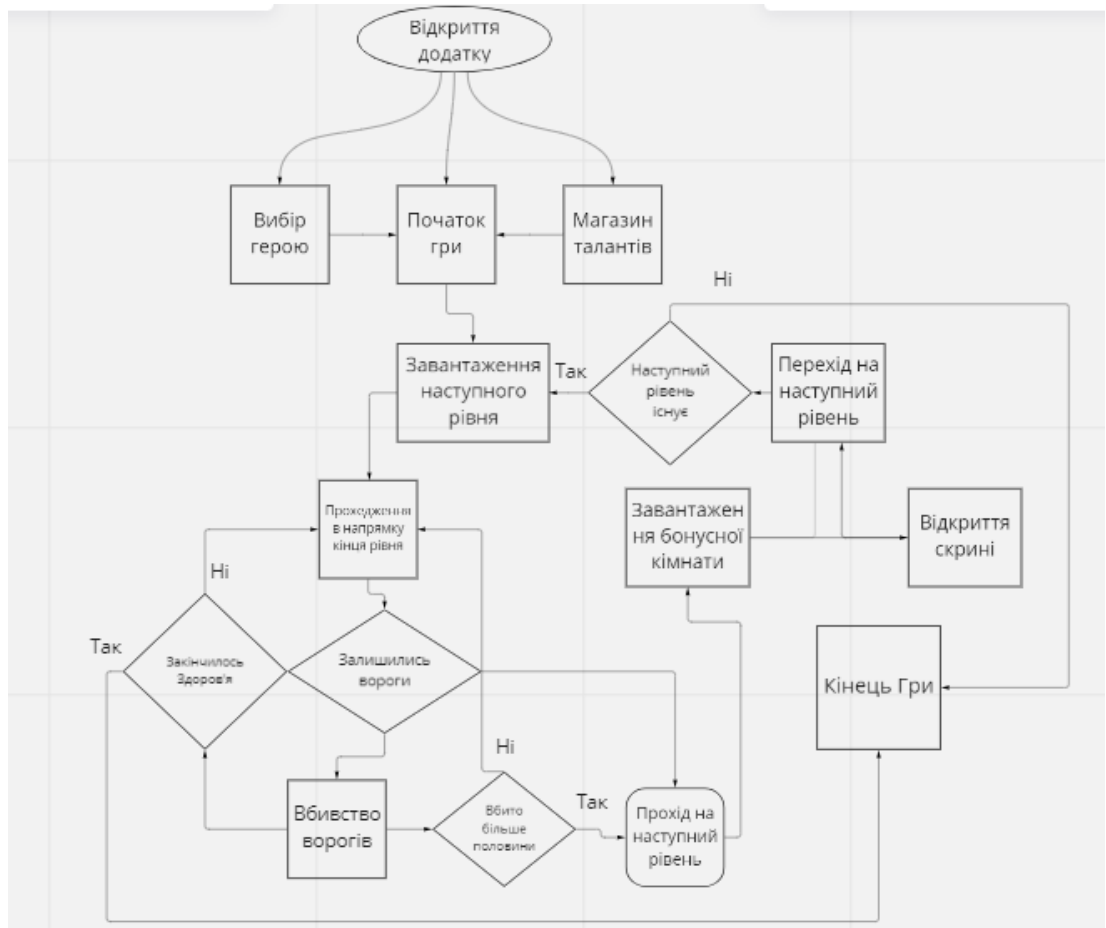


Рис 2.2.3 Діаграма можливостей користувача

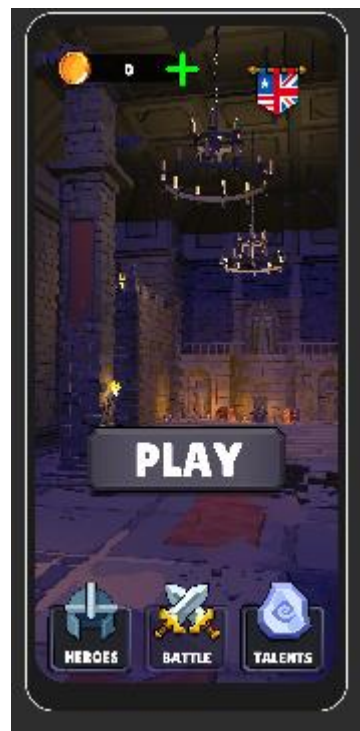


Рис 2.2.4 Головне меню

При запуску гри користувач потрапляє у головне меню(Рис 2.2.4). Під час запуску проходить перевірка на наявність даних останніх сесій, проходить ініціалізація сервісів аналітики та завантажується реклама. В головному меню маємо змогу почати гру, змінити мову, героя чи покращити їх характеристики або купити внутрішню ігрову валюту.

При виборі героя змінюється його тип атаки, модель, анімація, штучний інтелект, характеристики. При виборі рицаря користувач повинен враховувати складність підходу до ворога та можливість атакувати декілька цілей за раз. При виборі лучника користувач має змогу атакувати ціль на відстані та під час руху, що вимагає бути більш рухливим. Для покращення характеристик існує 6 унікальних властивостей, кожна з яких має свій вплив в тактиці бою.

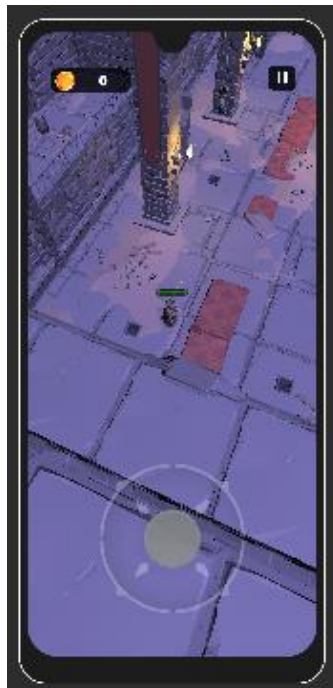


Рис 2.2.5 Ігровий рівень

Після того як користувач нажимає кнопку «Грати» він переходить на ігровий рівень, перед цим дані про додаткові характеристики ігрового персонажа додаються до базових характеристик персонажу.

Після атаки ворогу потрібно змінити своє положення для знаходження за допомогою зарання згенерованої карти переміщення NavMesh Рис 2.2.7

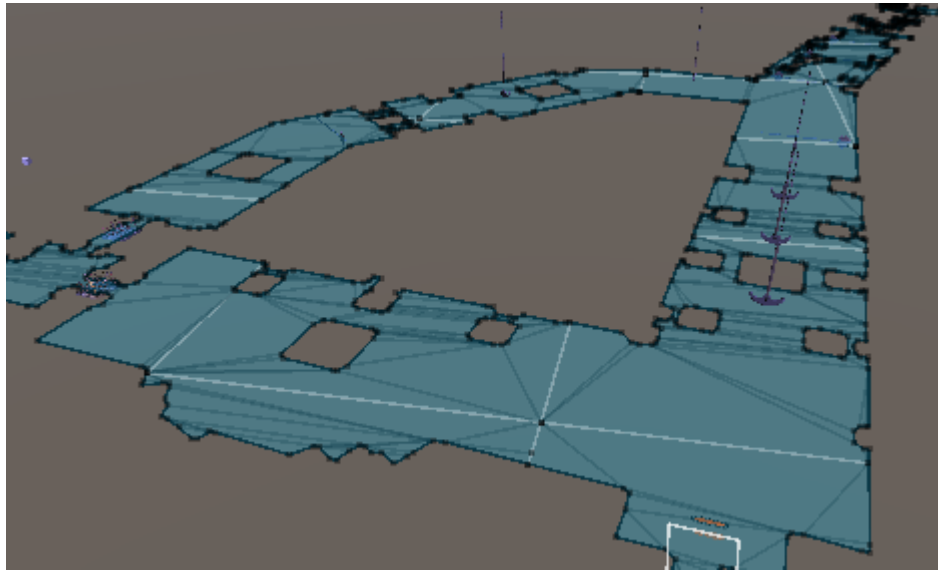


Рис 2.2.7 Запечена карта переміщення NavMesh

Навігаційна система дозволяє створювати персонажів, які можуть орієнтуватися в ігровому світі. Це дає вашим персонажам можливість зрозуміти, що їм потрібно піднятися сходами, щоб піднятися на другий поверх, або стрибнути, щоб перебратися через канаву.

Unity NavMesh система складається з таких частин:

NavMesh (скорочено від Navigation Mesh) — це структура даних, яка описує доступні для прогулянки поверхні ігрового світу та дозволяє знайти шлях від одного доступного місця до іншого в ігровому світі. Структура даних створюється або створюється автоматично на основі геометрії вашого рівня.

Компонент NavMesh Agent допоможе вам створити персонажів, які уникають один одного, рухаючись до своєї мети. Агенти міркують про ігровий світ за допомогою NavMesh, і вони знають, як уникати один одного, а також переміщувати перешкоди.

Компонент Off-Mesh Link дозволяє включати ярлики навігації, які неможливо представити за допомогою поверхні, яку можна прогуляти.

Наприклад, перестрибування через канаву чи паркан або відкриття дверей перед тим, як пройти через них, можна описати як Off-mesh посилання.

Компонент NavMesh Obstacle дозволяє вам описати рухомі перешкоди, яких агенти повинні уникати під час навігації по світу. Бочка або ящик, керований системою фізики, є хорошим прикладом перешкоди. Поки перешкода рухається, агенти роблять все можливе, щоб уникнути її, але як тільки перешкода стає нерухомою, вона вирізає отвір у навмеші, щоб агенти могли змінити свій шлях, щоб обійти її, або якщо нерухома перешкода блокує шлях. Таким чином, агенти можуть знайти інший маршрут.

В той час коли параметр здоров'я зменшується до 0, ворог втрачає можливість реагувати на влучання стріл та переходить до анімації смерті, після чого лічильник рахує нового вбитого ворога далі модель ворога знищується. Лічильник смерті ворогів рахує кількість ворогів які були знищені, після досягнення певної кількості знищених ворогів користувачу відкривається кімнати, перша бонусна кімнату можна побачити на рис 2.2.8, з можливістю збільшенню характеристик та відновленню кількістю параметру здоров'я.



Рис 2.2.8 Кімната з бонусом

Після знищення половини ворогів користувачу відкривається можливість перейти на наступний рівень. Прохід до наступного рівня знаходиться в протилежній стороні від початку. Під час переміщення до наступного рівня користувач повинен долати багато численні пастки. Існують 3 види пасток:

- Шипи в підлозі
- Літаючі сокири
- Бійниця зі стрілами

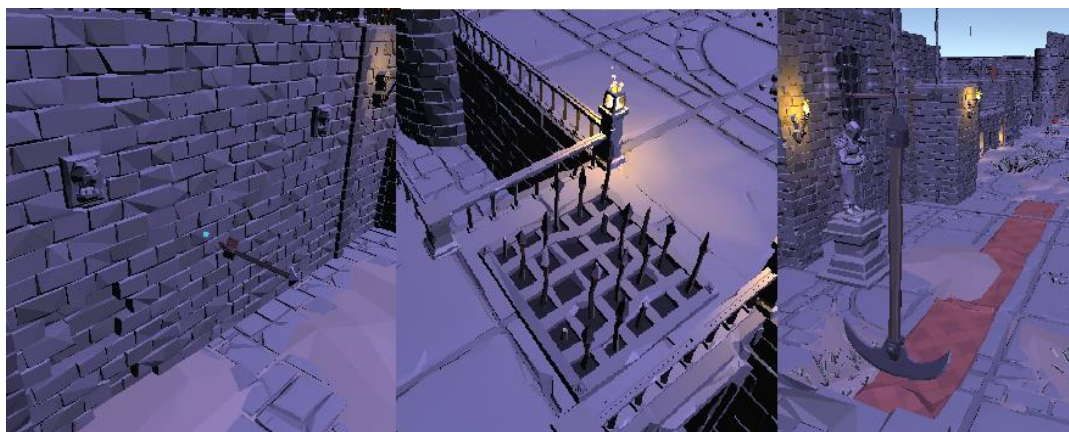


Рис 2.2.9 Види пасток

Після проходження всіх пасток та знищення як мінімум половини ворогів відкривається прохід до бонусного рівня який можна побачити на рис 2.2.10. На цьому рівні існує можливість відкрити прохід до бонусного скрині. Для відкриття проходу до скрині потрібно подивитися рекламу яка була завантажена під час запуску додатку. Під час переходу до цієї сцени характеристики користувача зберігаються та завантажуються, відправляються дані про перехід до аналітики. Після отримання бонусів можна перейти до наступного рівня.



Рис 2.2.10 Бонусний рівень

Після проходження до проходу на наступний рівень оновлені характеристики користувача записуються сцена вивантажується з пам'яті та

завантажується наступний рівень. Кожен новий рівень має свою унікальну архітектуру, тематику, характеристики ворогів та пасток. Як тільки користувач пройде 3 рівні він отримує змогу пройти до останнього рівня з босом який можна побачити на рис 2.2.11

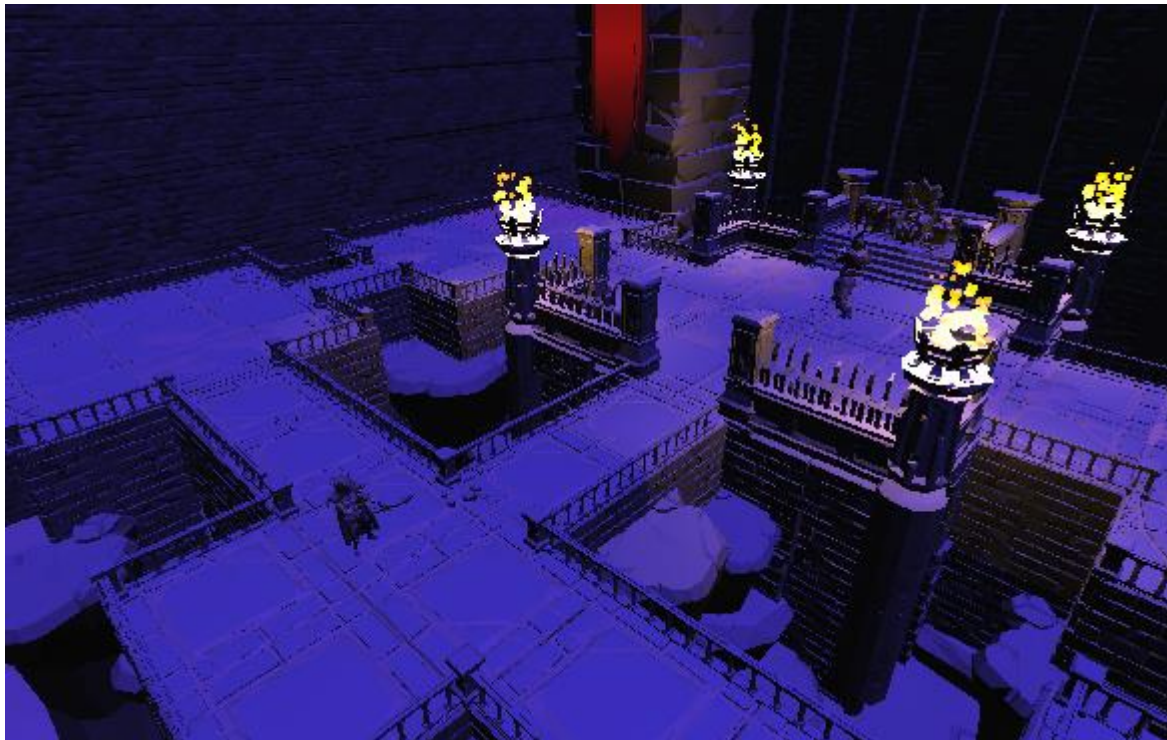


рис 2.2.11 Рівень з босом

2.3 Архітектура мобільної гри

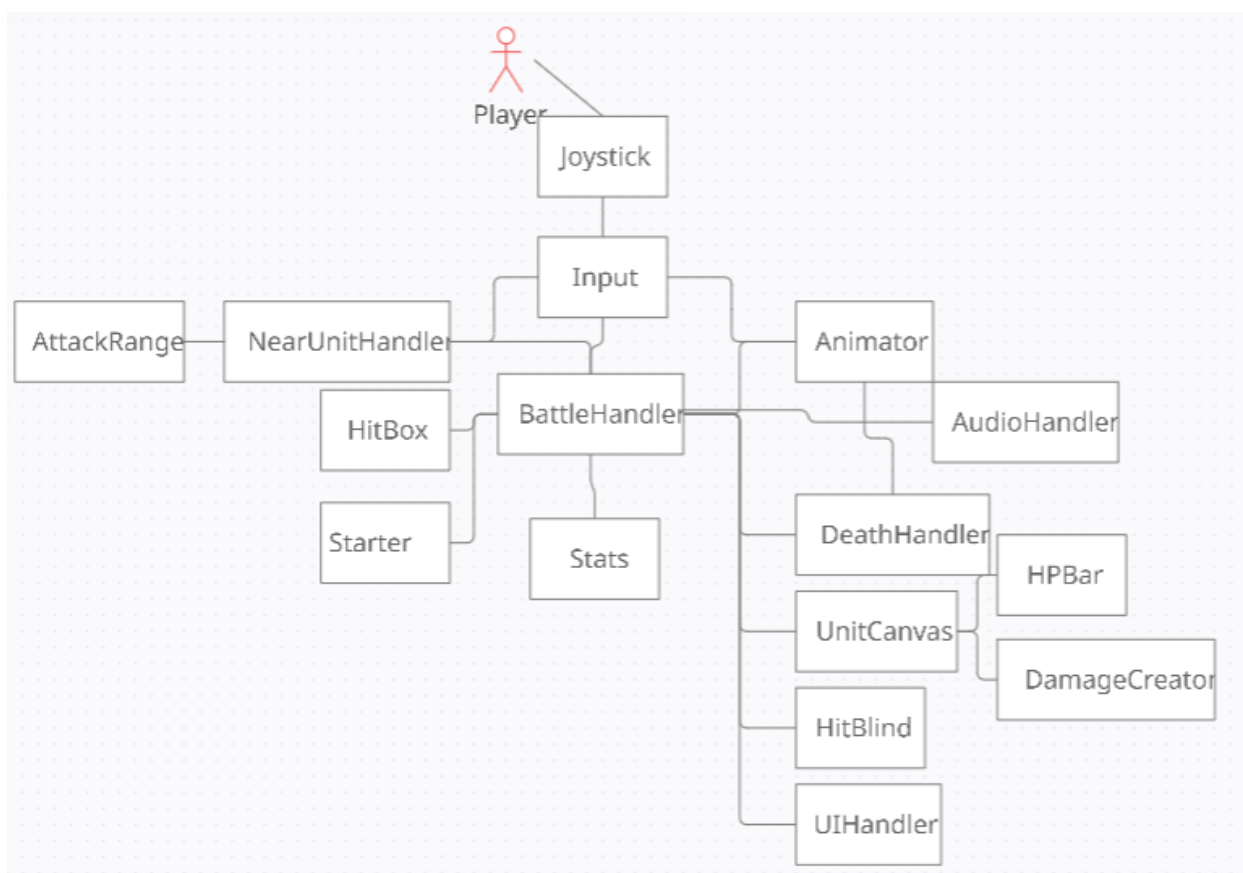


Рис 2.3.1 Структурна схема логіки головного персонажа

На рисунку 2.3.1 зображено структурна схема логіки головного персонажу, після запуску або переходу на сцену клас Starter завантажує дані про користувача та передають посилання до класу BattleHandler через нього клас Stats отримує оновлені дані.

Користувач взаємодіє з головним персонажем завдяки класу JoyStick коли користувач вперше натискає на екран Joystick отримує точку у двох-вимірній системі координат після того як точка дотику змінюється вираховується вектор напрямку за допомогою кода зображеному на рис 2.3.2

```

Vector2 value = new Vector2(startTouchPos.x + newPos.x, startTouchPos.y + newPos.y);
Vector2 moveVector = startTouchPos - value;
moveVector.y = -moveVector.y;
moveVector /= maxJoyStickRange;

```

Рис 2.3.2 Розрахунок вектору напрямку

Після того як отримано вектор напрямку потрібно дізнатися кут погляду персонажу, для цього використовується позиція найближчого ворога в радіусі атаки якщо такий є в наявності в вигляду коду це можна побачити на рис 2.3.3.

```

private void CalculateLookVector()
{
    if(_enemyTransform != null)
    {
        Vector3 targetVector = transform.position - _enemyTransform.position;
        float angle = Quaternion.LookRotation(targetVector, transform.forward).eulerAngles.y;
        angle -= 180 + transform.rotation.eulerAngles.y;
        if(angle < 0)
        {
            angle += 360;
        }

        _lookAngle = angle;

        SetAttackStatus(true);
    }
    else
    {
        _lookAngle = 0;
        SetAttackStatus(false);
    }
}

```

Рис 2.3.3 Розрахунок кута погляду персонажа

Як тільки ми отримали вектор напрямку та кут погляду ці дані потрібно відправити в аніматор. За передачу даних в аніматор відповідає клас Animator, також маючи всі дані руху в разі потреби додатково розраховується додаткова швидкість руху чи поворот тіла.

Якщо в радіусі атаки знаходиться ворог паралельно з анімацією переміщенням ніг починає працювати анімація атаки після того як вона завершується перед персонажем з'являється стріла яка має вектор польоту такий же як персонаж під час її створення. Стріла отримує дані про характеристики персонажу який її створив.


```

public override void CreateProjectile(bool isMoove,float lookAngle)
{
    GameObject projectileTemp = Instantiate(projectile,
        transform.position + new Vector3(0, 1, 0) + transform.right * -0.5f,
        transform.rotation * Quaternion.Euler(new Vector3(0,lookAngle,0)) );
    ProjectileStat projectileStat = projectileTemp.GetComponent<ProjectileStat>();
    projectileStat.SetAttacker(battleInfo);
    projectileStat.SetMarker(battleInfo.GetMarker());
    projectileStat.SetIsMoove(isMoove);
}

```

Рис 2.3.4 Створення стріли

Для того щоб зафіксувати влучання стріли використовується вбудована фізика Unity. Кожен персонаж має HitBox – фізична зона навколо персонажу яка реагує на влучання стріл. HitBox зазвичай використовують примітивних форм для економії ресурсу для розрахунку.



Рис 2.3.5 HitBox навколо персонажу

Клас DeathHandler використовується коли характеристика здоров'я персонажу падає до 0. Після смерті головного персонажу гра запропонує можливість продовжити грати, але перед цим потрібно подивитися рекламу. Возводиться можна тільки один раз за забіг. Після смерті ворога

DeathHandler змінює активну анімацію на анімацію смерті та зачисляє золото як винагороду.

Кожен персонаж має свій Canvas — це область, всередині якої мають бути всі елементи інтерфейсу. Canvas — це ігровий об'єкт із компонентом Canvas, всі елементи інтерфейсу мають бути дочірніми для такого Canvas. За зміну інформації на Canvas відповідає клас UnitCanvas. При зміні кількості здоров'я персонажа змінюється довжина смуги здоров'я. При отриманні шкоди з'являється число з кількістю нанесеної шкоди.

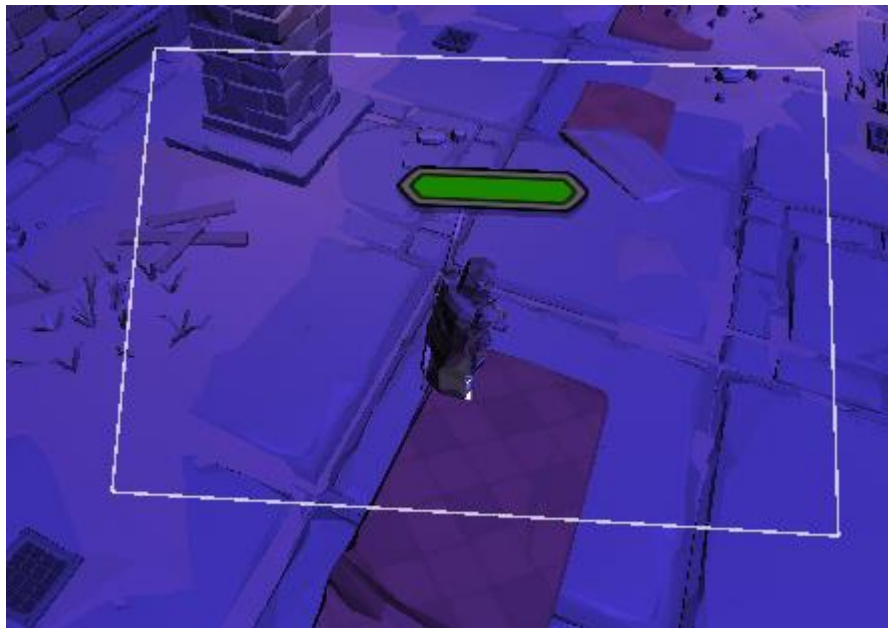


Рис 2.3.6 Canvas головного персонажу

2.4 Компіляція проекту під Android та її публікація

2.4.1 Компіляція проекту під Android

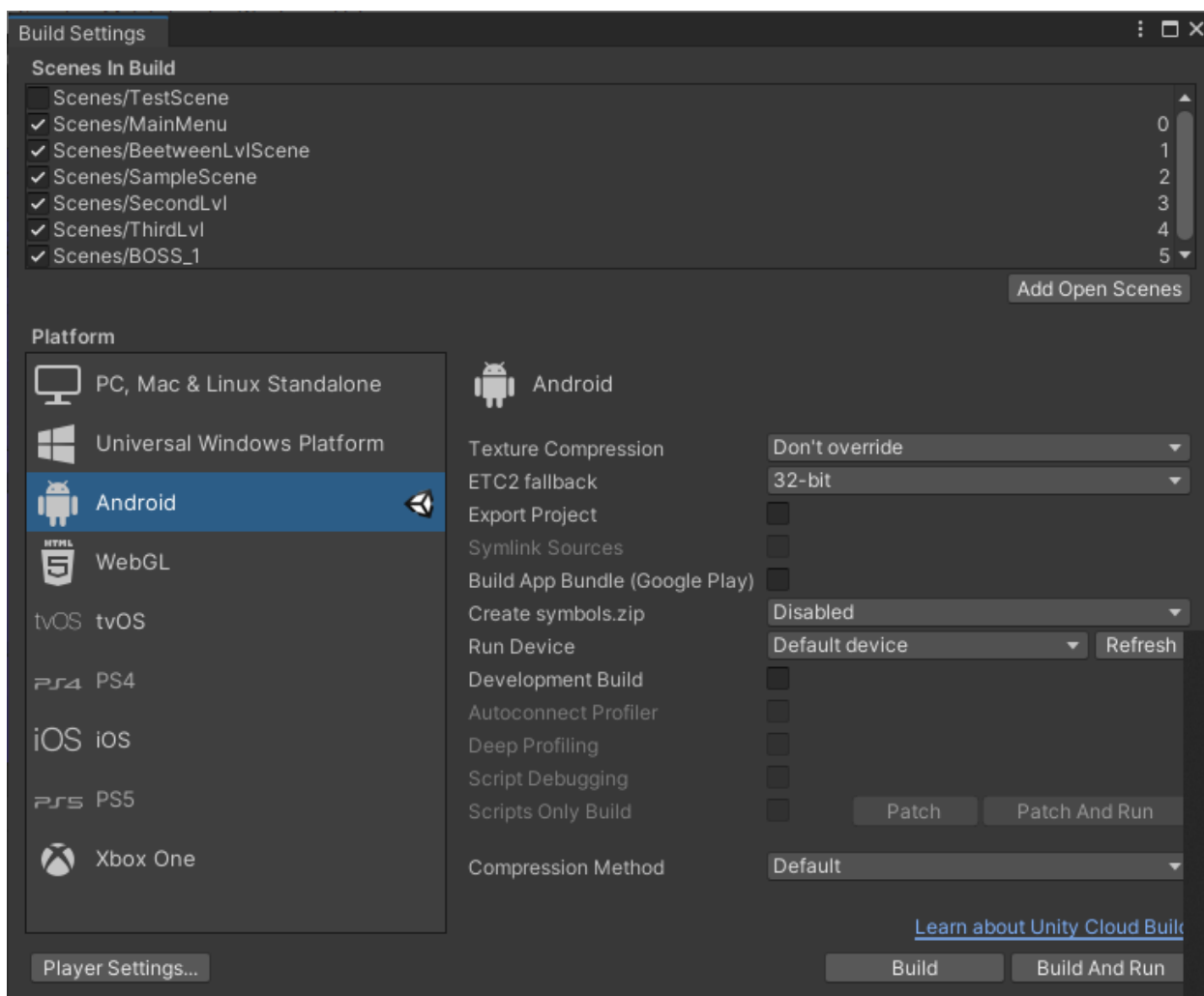


Рис 2.4.1.1 Вікно компіляції Unity

Unity має можливість компіляції під всі найпопулярніші платформи та широку можливість до налаштування самої компіляції. Для самої компіляції потрібно попередньо завантажити Android NDK, після завантаження відкривається можливість перевести проект до платформи Android.

Після переведення платформи на Android з'являється можливість використовувати внутрішній емулятор телефоні як на Рис 2.4.1.2

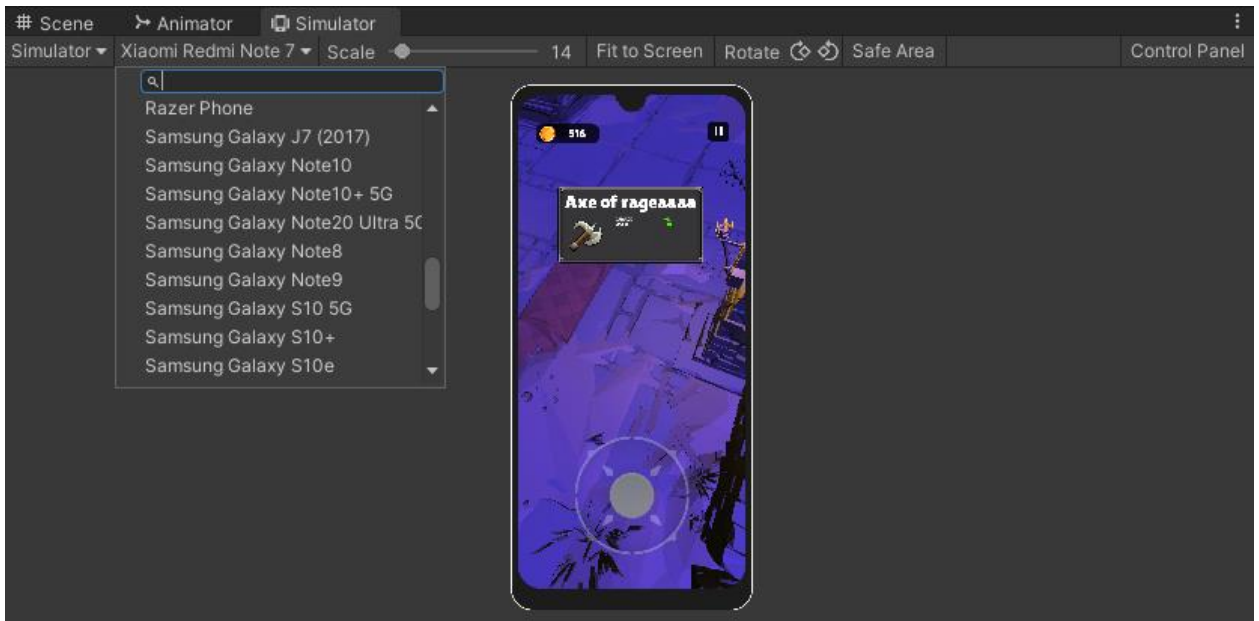


Рис 2.4.1.2 Внутрішній емулятор мобільного пристрою

В Unity компілюється кожна сцена окремо. Сцени — це те, де ви працюєте з вмістом у Unity. Це активи, які містять всю гру чи програму або її частину. Наприклад, ви можете створити просту гру в одній сцені, тоді як для більш складної гри ви можете використовувати одну сцену на рівень, кожна зі своїм власним середовищем, персонажами, перешкодами, прикрасами та інтерфейсом користувача. У проекті можна створити будь-яку кількість сцен. Коли ви створюєте новий проект і відкриваєте його вперше, Unity відкриває зразок сцени, яка містить лише камеру і світло.

Для компіляції на android можна вибрати декілька видів розширень арк та abb.

Android App Bundle – це формат публікації, який включає весь скомпільований код і ресурси вашої програми, а також відкладає створення файлів .apk і підписання в Google Play.

Google Play використовує ваш App Bundle для створення й обслуговування оптимізованих файлів .apk для кожної конфігурації пристрою, тому для запуску вашої програми завантажуються лише код і ресурси, необхідні для певного пристрою. Вам більше не потрібно створювати, підписувати й керувати кількома файлами .apk, щоб

оптимізувати підтримку різних пристроїв, а користувачі отримують менші й оптимізовані завантаження.

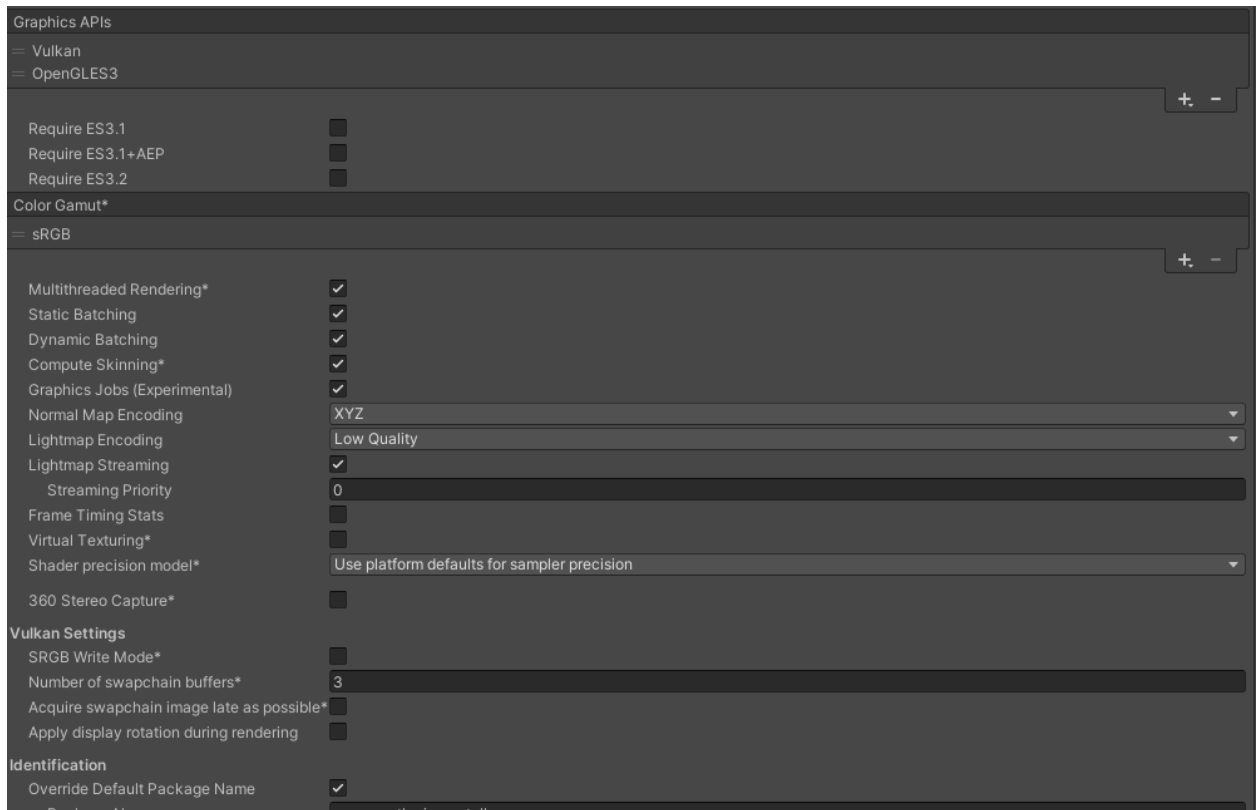


Рис 2.4.1.3 Налаштування компіляції

Перед самою компіляцією є можливість гнучкого налаштування базових параметрів, такі як графічне API, цільова версія операційної системи, якість графіки, технології звуку, якість освітлення, компілятор, систему збереження.

Для компіляції потрібно створити сховище ключів яке зможе забезпечити додаткову надійність додатку, та в наступному змінювати гру тільки при наявності цього сховищу.

2.4.2 Публікація проекту

Ви можете випустити свої програми Android кількома способами. Зазвичай ви випускаєте програми через ринок програм, наприклад Google

Play, але ви також можете випускати програми на власному веб-сайті або надсилаючи програму безпосередньо користувачеві.

Якщо ви хочете розповсюджувати свої програми серед якомога ширшої аудиторії, ідеальним варіантом є випуск через ринок додатків, як-от Google Play.

Google Play — це провідний ринок програм для Android, який особливо корисний, якщо ви хочете поширювати свої програми серед широкої світової аудиторії. Однак ви можете розповсюджувати свої програми через будь-який ринок додатків, який хочете.



Рис 2.4.2.1 Сторінка в Google Play

Для публікації гри в Google play спершу потрібно створити акаунт у Google play console. Для початку публікації потрібно заповнити дані про гру як показано на рис 2.4.2.2.

Создать приложение

О приложении

Название приложения

Это название будет показано в Google Play.

0/30

Язык по умолчанию

Английский (Соединенные Штаты) – en-US

Приложение или игра

Этот параметр можно изменить позже в настройках Google Play.

Приложение

Игра

Рис 2.4.2.2 Форма Google play console при створені додатку

Після заповнення даних про додаток з'явиться можливість завантажити до магазину файл з розширенням AAB. Файл повинен важити менш ніж 150 мб. та відповідати всім правилам безпеки даних платформи.

Також важливу роль в популярності гри займає створення її сторінки. В магазині додатків Google Play для поліпшення кількості показів гри користувачам потрібно правильно створити опис, заголовок та назву. Саме завдяки цим текстам ігровий додаток індексується у магазині додатків.

3 ТЕСТУВАННЯ ГРИ ВИХОДЯЧИ З ДАНИХ АНАЛІТИЧНИХ СЕРВІСІВ

3.1 Аналітичний сервіс Firebase

Створення програми або веб-сайту на перший погляд може здатися дуже вичерпним процесом, який включає значні інвестиції часу та коштів. Проте існує багато доступних сервісів, які можуть полегшити роботу з розробки різними способами. Firebase є одним з них. Цей гігант готовий зняти значну частину тягаря з плечей розробника.

Firebase була заснована Ендрю Лі та Джеймсом Тампліном ще в 2011 році, але була офіційно запущена в квітні 2012 року. Спочатку передбачалося, що фреймворк буде базою даних у режимі реального часу, яка надає свої API, що дозволить користувачам зберігати та синхронізувати інформацію для різних клієнтів. Однак плани дещо змінилися, коли через два роки після випуску Firebase перейшла до Google. Сьогодні сервіс має різноманітні функції, які мудрий підприємець може вважати винятково цінними. Firebase досяг 110 000 розробників, коли він був спочатку отриманий Google у жовтні 2014 року, і зараз це 450 000 розробників.

Firebase — це фреймворк, який корисний для створення портативного та веб-додатку для вашого бізнесу з базою даних у реальному часі, що означає, що коли один користувач оновлює запис у базі даних, це оновлення буде передано кожному окремому користувачеві, будь то користувачі веб-сайту пристрою iOS або Android. Це дає базову та уніфіковану платформу з великою кількістю функцій Google. Вам не потрібно налаштовувати сервер, коли ви використовуєте Firebase. Firebase подбає про все автоматично.

Є багато елементів, які роблять роботу з Firebase чудовою з точки зору розробника, які стосуються основної технології розробки. Це допомагає підтримувати стан гармонії між розробником і клієнтом, викликаючи

мінімальну затримку роботи. У цьому випадку використання Firebase для розробки додатків має багато переваг.

Firebase — це платформа для розробки додатків, яка допомагає вам створювати та розвивати програми та ігри, які подобаються користувачам. За підтримки Google і довіри мільйонів компаній по всьому світу.

Google Analytics збирає дані про використання та поведінку вашого додатка. SDK реєструє два основних типи інформації:

- Події - це те, що відбувається у вашій програмі, як від дії користувача, системні події чи помилки.
- Властивості користувача: атрибути, які ви визначаєте для опису сегментів вашої бази користувачів, як-от налаштування мови або географічне розташування.

Аналітика автоматично реєструє деякі події та властивості користувачів. Вам не потрібно додавати код, щоб увімкнути їх. Якщо вашій програмі потрібно збирати додаткові дані, ви можете налаштувати до 25 різних властивостей користувача Analytics і зареєструвати до 500 різних типів подій Analytics у своєму додатку. Загальна кількість подій, які реєструє ваш додаток, не обмежена.

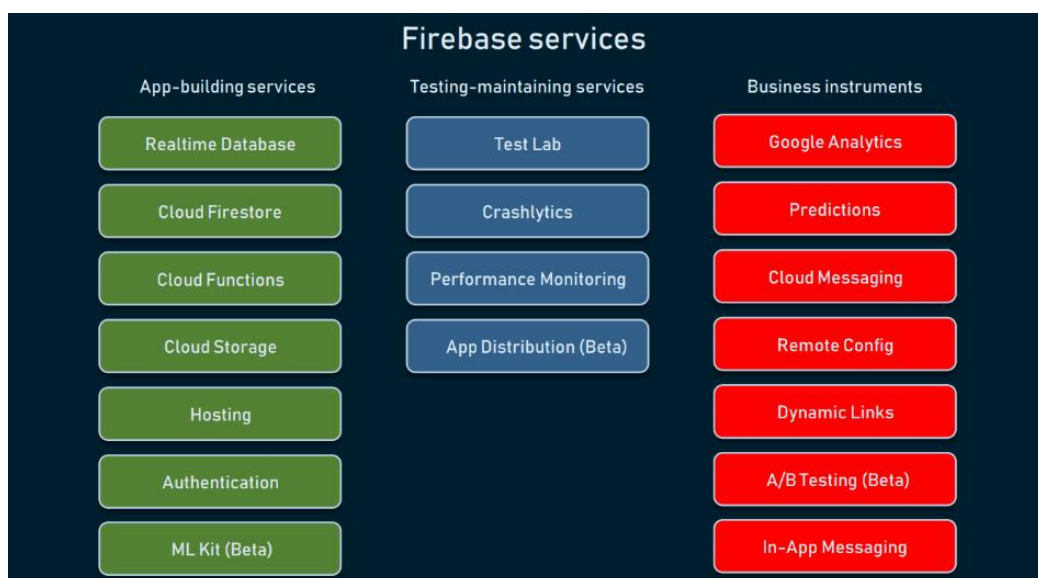


Рис 3.1.1 Сервіси Firebase

Для проекту було виділені необхідні сервіси, а саме:

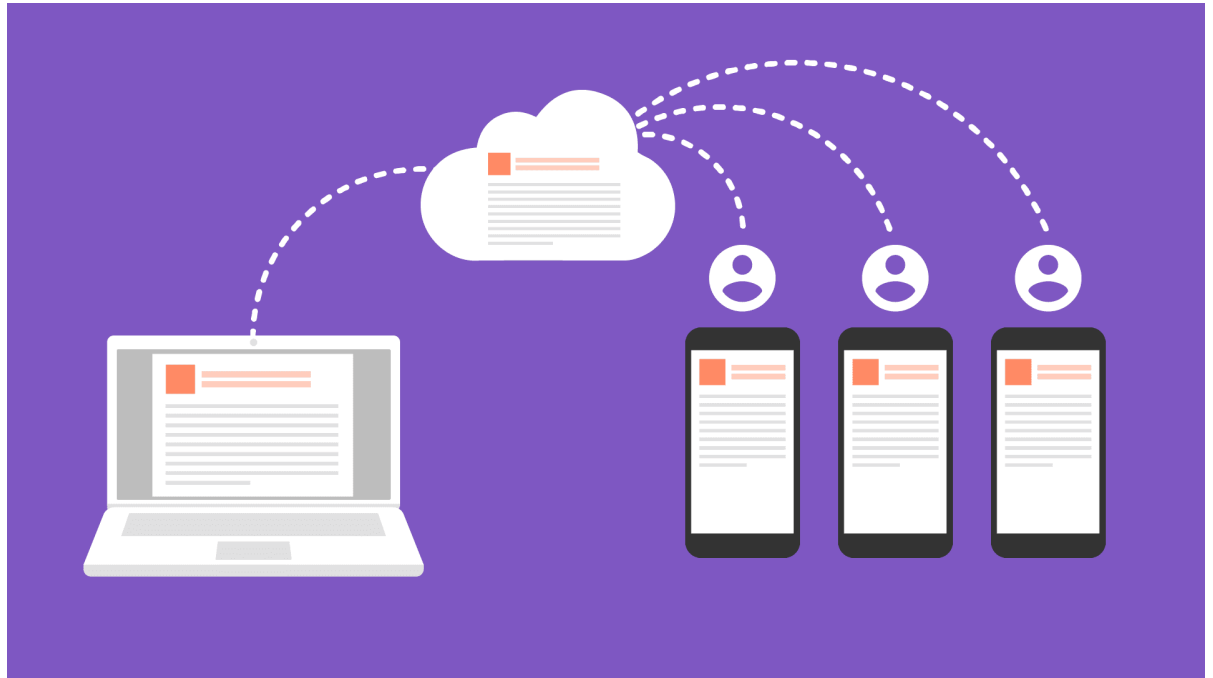


Рис 3.1.2 Бази даних Firebase

Firestore Realtime Database була першим продуктом під прапором Firebase, тому це найвідоміший і стабільний сервіс на всій платформі. База даних реального часу — це, по суті, хмарне сховище NoSQL, яке можна підключити до програми, щоб забезпечити доступ до даних у режимі реального часу на різних платформах. Однією з переваг є те, що база даних може працювати в автономному режимі, кешуючи дані в пам'яті пристрою, а після повторного підключення до Інтернету синхронізувати його.

Дані зберігаються у форматі JSON, і користувачі можуть запитувати їх. З точки зору безпеки, база даних реального часу надає доступ до даних на основі дозволів. Це можна зробити за допомогою автентифікації Firebase та надання дозволів за ідентифікацією користувача або правилами безпеки.

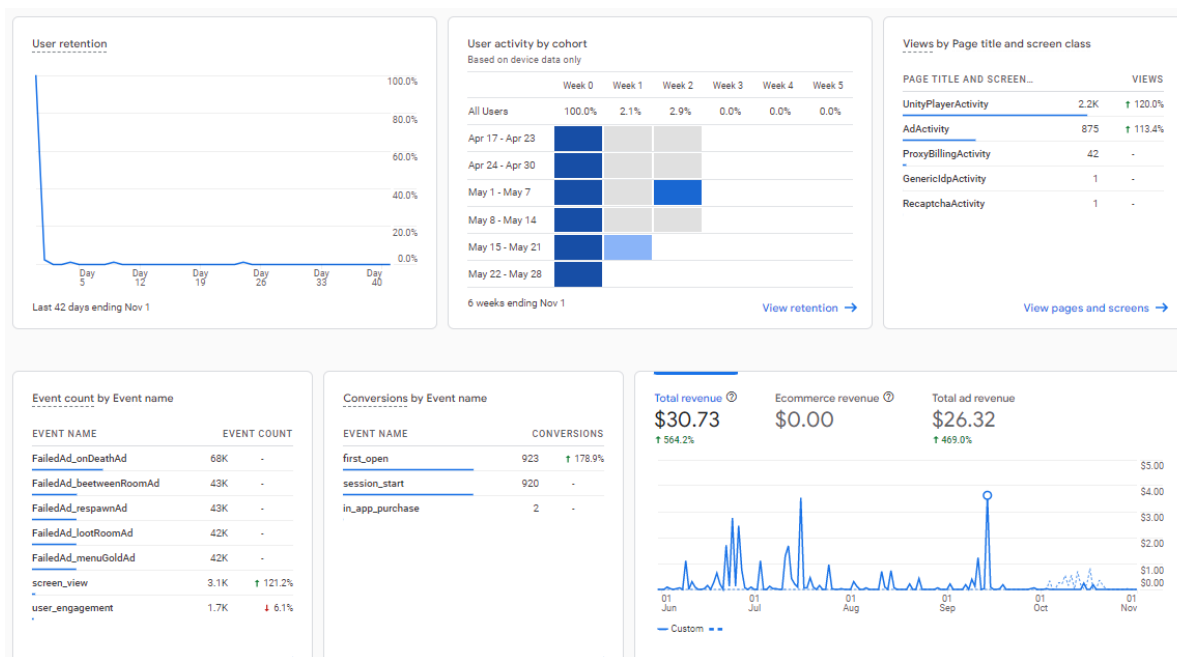


Рис 3.1.3 Аналітика Firebase

Коли ваш додаток опубліковано, існує також ряд стандартних і спеціальних служб Google, які допоможуть вам керувати тим, як користувачі взаємодіють із вашим додатком у бізнес-аспекті.

Google Analytics стоїть на першому місці, оскільки це добре відомий інструмент, інтегрований у платформи Firebase. Google Analytics надає цінні показники щодо утримання користувачів, рівня залучення користувачів або будь-яких інших даних про поведінку користувачів. З точки зору Firebase, можливості звітування необмежені та безкоштовні. Ви можете вибрати, які події відстежувати, і інтегрувати їх з аналітичними партнерами.

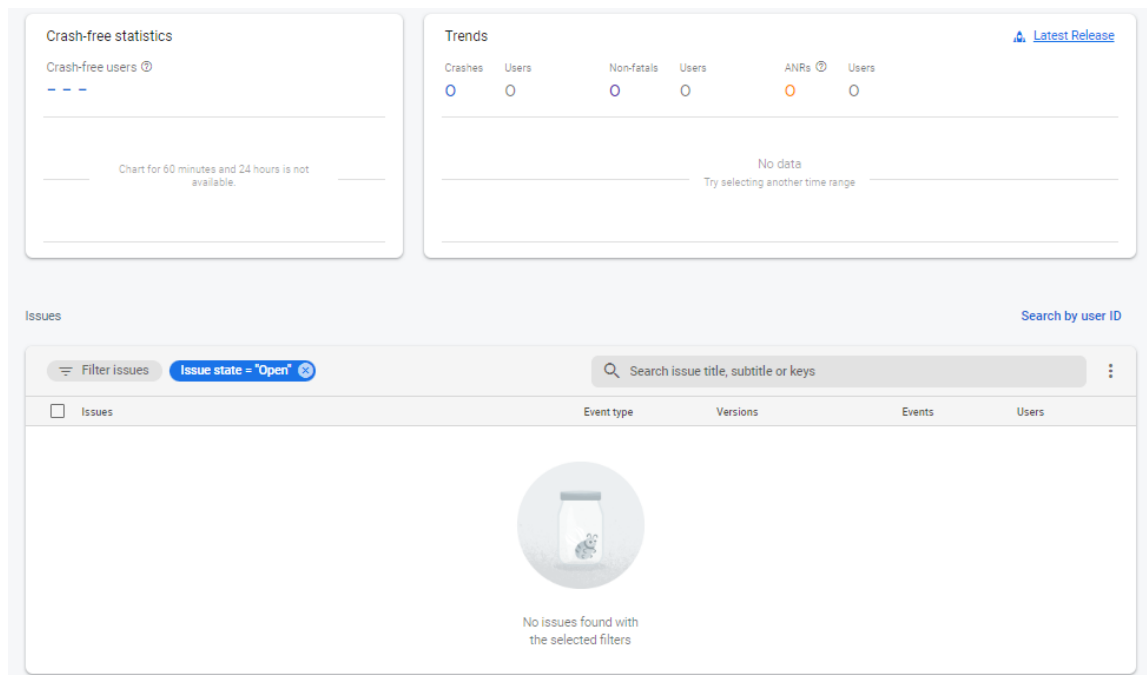


Рис 3.1.4 Crashlytics Firebase

Crashlytics — це інструмент звітів про аварійне завершення роботи в реальному часі для швидкого та безболісного усунення несправностей стабільності вашої програми. Звіти про аварійне завершення роботи будуть формуватися у структуровані та пріоритетні списки проблем, які відповідають кореню проблеми, тому легко організувати процес усунення

3.2 Тестування гри виходячи з отриманих даних

Тестування гри проходило 4 місяці в магазині додатків Google play. Основною метою тестування є порівняння параметрів з ринковими нормами. Основними параметрами є:

- Середній час проведений в грі.
- Середній процент повернення користувача на 1 день.
- Середній процент повернення користувача на 7 день.
- Процент критичних відказів.
- Кількість завантажень.

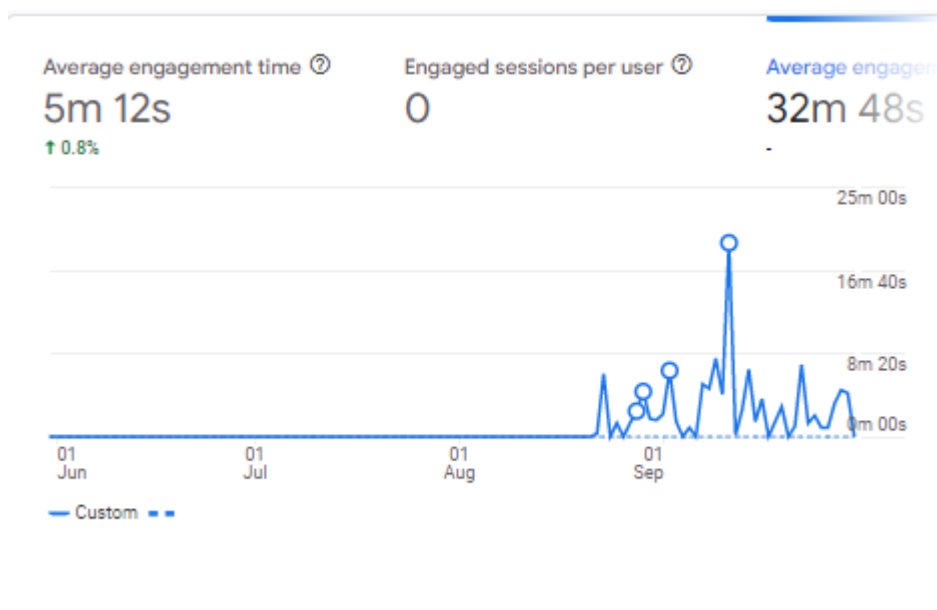


Рис 3.2.1 Середній час проведений в грі

При порівнянні параметра середнього часу гри потрібно по-перше зрозуміти тип гри та точність попадання в аудиторію. Так як Immortal heroes рахується грою з типом Midcore, для типу таких ігор потрібен середній час в грі від 20 хв, ми можемо бачити на рис 3.2.1, що кількість часу проведеного в грі в середньому не відповідає ринковим вимогам, але є користувачі які проводили в грі час більше ніж ринковий. Тому можна сказати, що шанс на успіх існує при умові кращого попадання в аудиторію.

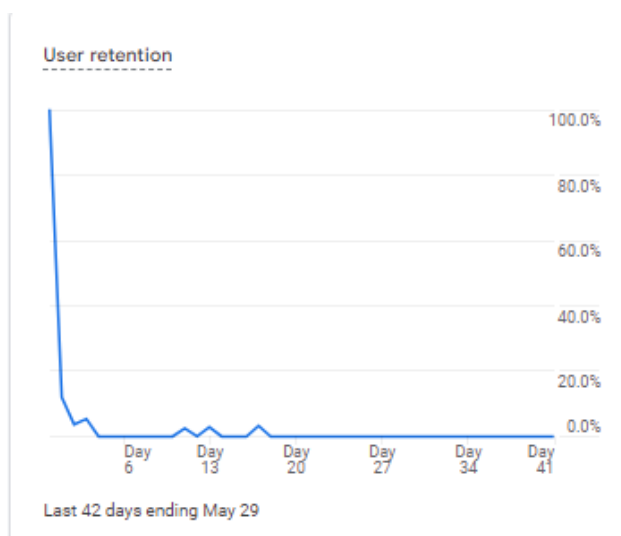


Рис 3.2.2 Середній процент повернення користувача

На сьогоднішній день гарний процент повернення першого дня рахується від 35%, якщо подивитися на рис 3.2.2 можна зрозуміти навіть не близько. Гра має 12.1% повернення користувача на наступний день, що не являється ринковим параметром. Якщо дивитися на повернення 7 дня, то цей параметр має величину в 2.4% при нормі від 15%.

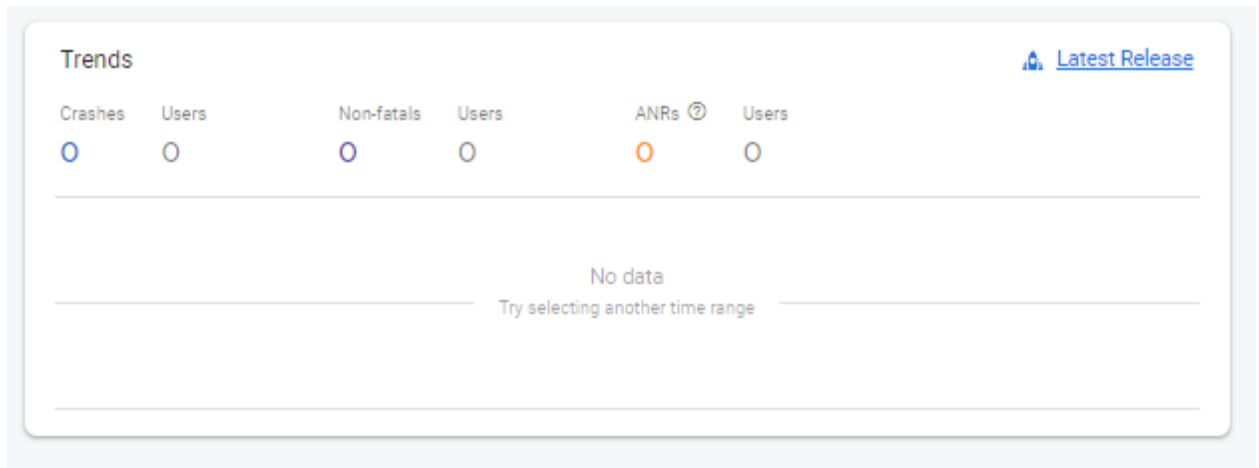


Рис 3.2.3 Кількість відказів

Кількість критичних відказів реєструється тільки за останній місяць. За останній місяць відказів виявлено не було, але якщо брати за весь час кількість відказів має показник в 0.7%, що є в декілька разів краще за ринкову яка складає в середньому 3%. Всі відкази які були зафіксовані зв'язані з не можливістю завантажити графічне ядро на телефони старого випуску.



Рис 3.2.3 Кількість завантажень

Як можна побачити на рис 3.2.3 найбільша кількість завантажень приходить на Росію, що є погано. Росія має маленький дохід на одного користувача та велику кількість негативних відгуків на основі завищених очікувань від гри. На другому місці знаходиться США, який рахується як найкращий ринок. США має великий дохід на одного користувача, оцінки які залишають США зазвичай високі чи середні з великою кількістю зворотного зв'язку.

4 ВИСНОВКИ

Підбиваючи підсумки проведеного аналізу можна сказати що ринок ігрової індустрії буде ще довго зростати. Починаючи з самих джерел в 1970-х роках закінчуючи нашими днями можна побачити як майже за півстоліття аудиторія відеоігор зросла майже з нуля до чверті населення землі.

Зараз, завдяки доступності інтернету і мобільних пристроїв ця індустрія переживає один з кращих періодів: аудиторія зростає і розширюється завдяки спрощенню доступу до ігор. За останні десять років поріг входження значно знизився. З'явилася категорія користувачів які просто хочуть невимушено скоротати небагато часу, наприклад по дорозі додому з роботи.

Під час виконання дипломної роботи була створена та проаналізована актуальність розробки гри Immortal heroes. Проведено аналіз програмним засобів для розробки нових мобільних ігрових додатків, завдяки якому було вирішено почати розробку на Unity. Unity на даний момент показав себе як найкращий варіант розробки завдяки цільовому розвитку в бік мобільного геймінгу.

За допомогою статистичних даних було виявлено переваги та недоліки гри на конкурентному ринку ігрових додатків, а саме велику перевагу у відмово стійкості, та у великій кількості користувачів з США . Для цього було використано більше 1000 унікальних користувачів та порівняння даних отриманих з них з ринковими нормами.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Gambier Y. Rapid and Radical Changes in Translation and Translation Studies International [Электронный ресурс]. – Режим Доступа : <http://ijoc.org/index.php/ijoc/article/viewFile/3824/1570>
2. The GI.biz Academy's in-depth guide to Unity Technologies' widely used Unity engine, the leader for indie and mobile game development [Электронный ресурс]. – Режим Доступа : <https://www.gamesindustry.biz/articles/2020-01-16-what-is-the-best-game-engine-is-unity-the-right-game-engine-for-you#section-1>
3. An in-depth guide to Crytek's CryEngine, which balances huge visual power against a smaller community [Электронный ресурс]. – Режим Доступа : <https://www.gamesindustry.biz/articles/2020-01-16-what-is-the-best-game-engine-is-cryengine-the-right-game-engine-for-you>
4. Introduction to Unreal Engine [Электронный ресурс]. – Режим Доступа : <https://www.educba.com/what-is-unreal-engine/>
5. Unity 3d game development: advantages and disadvantages [Электронный ресурс]. – Режим Доступа : <https://logicsimplified.com/newgames/unity-3d-game-development-advantages-and-disadvantages/>
6. tool for tracking and responding to reviews. AppFollow makes it far easier to track what customers think about your mobile apps and leave as reviews. [Электронный ресурс]. – Режим Доступа : <https://watch.appfollow.io/apps/my-first-workspace/>
7. Мобильные приложения мировой рынок [Электронный ресурс]. – Режим Доступа : [https://www.tadviser.ru/index.php/Статья:Мобильные_приложения_\(мировой_рынок\)](https://www.tadviser.ru/index.php/Статья:Мобильные_приложения_(мировой_рынок))

8. Video Game Industry Statistics, Trends and Data In 2022 [Электронный ресурс].– Режим Доступа : <https://www.wepc.com/news/video-game-statistics/>
9. Reasons Why Gaming Is Actually an Excellent Hobby [Электронный ресурс].– Режим Доступа : <https://www.nerdynaut.com/5-reasons-why-gaming-is-actually-an-excellent-hobby>
10. Sánchez, J. L. G. Playability: analysing user experience in video games. Behaviour & Information Technology [Электронный ресурс].– Режим Доступа : <http://dx.doi.org/10.1080/0144929X.2012.710648>
11. Top 100 countries by game revenues [Электронный ресурс].– Режим Доступа : <https://newzoo.com/insights/rankings/top-100-countries-by-game-revenues/>
12. Schäler, R. Linguistic resources and localisation / Schäler, R. – Limerick. : John Benjamins Press, 2008
13. Mayoral R. Concept of constrained translation. non-linguistic perspectives of translation / Mayoral R. – Boston. : Meta, 1988
14. Thayer A. Localization of digital games: The process of blending for the global games market. Technical Communication / Thayer A. – Oxford. : Lancer Books, 2004.
15. Як Україна виглядає на світовому ринку розробки ігор — 8 графіків [Електронний ресурс]. – Режим доступу: <https://businessviews.com.ua/ru/tech/id/virobnictvo-igor-v-ukrajini-1948/>
16. Top Operating System Software [Електронний ресурс]. – Режим доступу: <http://www.jegsworks.com/lessons/ComputerBasics/lesson8/lesson8-3.html>
17. Worldwide Digital Games Market [Електронний ресурс]. – Режим доступу: https://twitter.com/_SuperData/status/675337478514024448

ДЕМОНСТРАЦІЙНИ МАТЕРІАЛ



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Розробка гри «Immortal heroes» мовою с#

Виконав студент 5 курсу
Групи ППЗ-51
Конкін Владислав Олександрович
Керівник роботи
Доктор технічних наук Жебка Вікторія Вікторівна

Київ - 2022

Мета, об'єкт та предмет дослідження

- **Мета роботи** проходження процесу накопичення даних поведінки користувачів в умовах гри та подальший її аналіз.
- **Об'єкт дослідження** процес розробки гри жанру RogueLike мовою С# на двигуні Unity.
- **Предмет дослідження** методи за засоби створення гри жанру RogueLike ImmortalHeroes

Аналоги



Dungeon Knight



Dread Rune



Nonstop Knight 2 -
Action RPG

3

Назва	Dungeon Knight	Dread Rune	Nonstop Knight 2 - Action RPG	Immortal Heroes
+	Великі рівні	Фізичні елементи мапи	Насичена графіка	Унікальні рівні
+	Велика кількість ефектів	Інерційна система переміщення	Велика кількість ворогів	Велика кількість предметів
-	Поганий інтерфейс	Погана оптимізація	Простий цикл геймплею	Мала варіативність ворогів
-	Погана оптимізація	Мала кількість предметів	Мала кількість предметів	Мала кількість персонажів

4

Технічні завдання

- Створити базову логіку для реалізації механіки жанру RogueLike.
- Створити штучний інтелект для не головних персонажів.
- Спроекувати 5 унікальних рівнів для різноманіття гри.
- Розробити систему завантаження та збереження даних між рівнями.
- Підключення додатку до сервісів аналітики.
- Подальша публікація додатку до магазину додатків Google Play.

5

ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



Unity - комплексний інструмент розробки ігор.

C# - Універсальна мова програмування, яка використовується в Unity.

Visual Studio – середовище для написання коду, використовується для зручної роботи з C# та Unity.

Firebase – система аналітики від Google, використовується для швидкої інтеграції аналітики.

6

Скріншоти з гри



Рис 9.1 Головне меню



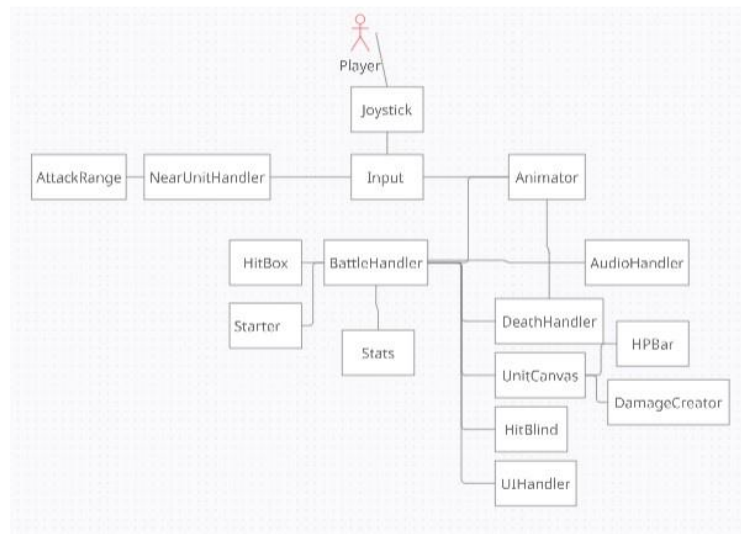
Рис 9.2 Ігровий рівень



Рис 9.3 Бонусний рівень

9

Діаграма класів головного персонажу



10

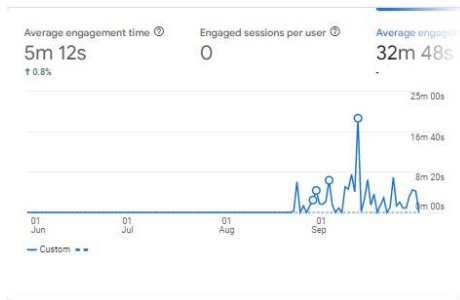


Рис 11.1 Середній час проведений в грі

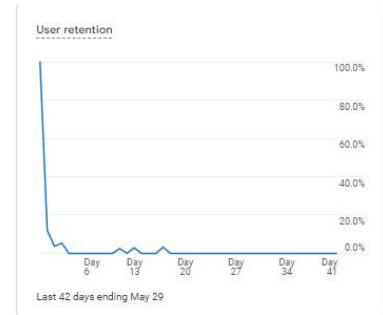


Рис 11.2 Середній процент повернення користувача

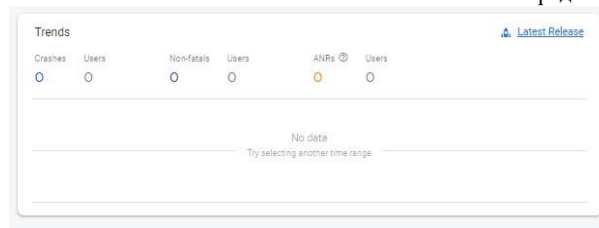


Рис 11.3 Кількість відказів

11

Апробація Результатів Дослідження

- Конкін В.О. Данно-орієнтоване програмування з використанням Unity3d: Науково-технічна конференція «Застосування програмного забезпечення в інфокомунікаційних технологіях : матеріали Міжнар. Наук.-практ. конф.,м.Київ 20.04.2022/ Держ. ун-т Телекомунікацій, Ф-т інф. технологій. Київ,2022 С.1-2»,.

12

Висновки

- Проведено дослідження в котрому аналізується актуальність розробки мобільної гри Immortal heroes
- було виявлені переваги та недоліки гри За допомогою статистичних даних .
- Проведено аналіз існуючих програмних засобів для розробки нових ігрових додатків.
- використано більше 1000 унікальних користувачів за допомогою статистичних даних коректності .
- Розроблена гра Immortal heroes з урахуванням усіх виявлених вимог.

13

ДЯКУЮ ЗА УВАГУ!

14