

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
Факультет інформаційних технологій

УДК

«ПОГОДЖЕНО»

Декан факультету
інформаційних технологій

Глазунова О.Г., д.п.н., професор

«ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ»

Завідувач кафедри комп'ютерних наук

Голуб Б.Л., к.т.н., доцент

_____ 202_ р.

6 листопада 2023 р.

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему Підсистема аналізу даних в ігровому додатку на
двигуні Unity для платформи Android

Спеціальність 122 Комп'ютерні науки

(код і назва)

Освітня програма Комп'ютерний еколого-економічний моніторинг

(назва)

Орієнтація освітньої програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Гарант освітньої програми

д.т.н., професор _____

(науковий ступінь та вчене звання)

Семко В.В.

(ПІБ)

Керівник магістерської кваліфікаційної роботи

к. фіз.-мат. н., доцент _____

(науковий ступінь та вчене звання)

Кириченко Віктор Вікторович

(підпис)

(ПІБ)

Виконав

(підпис)

Гасюк Олексій Олексійович

(ПІБ студента)

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Факультет інформаційних технологій

ЗАТВЕРДЖУЮ

Завідувач кафедри

к.т.н., доцент

Голуб Б. Л.

(науковий ступінь, вчене звання)

(підпис)

(ПІБ)

“29” грудня 2022 року

ЗАВДАННЯ

ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ СТУДЕНТУ

(прізвище, ім'я, по батькові)

Спеціальність 122 Комп'ютерні науки

(код і назва)

Освітня програма Комп'ютерний еколого-економічний моніторинг

(назва)

Орієнтація освітньої програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Тема магістерської кваліфікаційної роботи Підсистема аналізу даних в ігровому додатку на двигуні Unity для платформи Android затверджена наказом ректора НУБіП України від “29” грудня 2022 р. № 1917 “С”

Термін подання завершеної роботи на кафедру 5 листопада 2023 року

(рік, місяць, число)

Вихідні дані до магістерської кваліфікаційної роботи Ігровий 3D додаток First Person Shooter, в якому GamePlay підлаштовується під стиль та навички гравця. Фізика, аудіо підстанова, анімації та 3D моделі які гарантують візуальну складову.

Перелік питань, що підлягають дослідженню:

1. Дослідити методи створення ігрової сцени

2. Спроекувати і створити повноцінний ігровий додаток

3. Спроекувати Data Analyst, який буде аналізувати та підлаштовувати гру під

користувача

Перелік графічного матеріалу (за потреби) Постер

Дата видачі завдання “29” грудня 2022 р.

Керівник магістерської кваліфікаційної роботи _____

(підпис)

(прізвище та ініціали)

Завдання прийняв до виконання _____

(підпис)

(прізвище та ініціали студента)

Анотація

Метою даної магістерської роботи було створення 3D гри з унікальною функціональністю динамічної адаптації геймплея під індивідуальний стиль кожного гравця. Система аналізує поведінку учасника під час гри і на основі отриманих даних коригує інтенсивність та складність наступних хвиль мобів. Використання методів машинного навчання замість традиційних баз даних дозволило системі швидше та точніше адаптуватися до змін у поведінці гравців, що підтверджено результатами дослідження. Такий підхід не тільки покращує інтерактивність та захопленість від гри, але й сприяє постійному вдосконаленню навичок гравців завдяки непередбачуваності ігрового процесу.

The goal of this master's thesis was the creation of a 3D game featuring a unique functionality of dynamically adapting the gameplay to the individual style of each player. The system analyzes the participant's behavior during the game and adjusts the intensity and difficulty of subsequent waves of mobs based on the acquired data. The use of machine learning methods instead of traditional databases has allowed the system to adapt more quickly and accurately to changes in player behavior, as confirmed by the research findings. This approach not only enhances the interactivity and engagement of the game but also fosters the continuous improvement of players' skills due to the unpredictability of the game proc

ЗМІСТ

| | |
|--|-----------|
| ВСТУП..... | 4 |
| 1. СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ..... | 6 |
| 1.1 Опис предметної області..... | 6 |
| 1.2 Визначення вимог..... | 9 |
| 1.4 Аналіз наявних рішень..... | 12 |
| 1.5 Формулювання технічного завдання..... | 14 |
| 2. МОДЕЛЮВАННЯ СИСТЕМИ..... | 18 |
| 2.1 Діаграма прецедентів..... | 18 |
| 2.2 Постановка завдання..... | 20 |
| 2.3 Архітектура системи..... | 21 |
| 2.4 Опис джерела даних..... | 24 |
| 3. РОЗРОБКА СИСТЕМИ..... | 28 |
| 3.1 Опис архітектури системи..... | 28 |
| 3.1 Опис архітектури системи..... | 29 |
| 3.2 Вибір інструментарію для створення програмного забезпечення... | 30 |
| 3.3 Вибір та розробка бази даних..... | 32 |
| 3.4 Реалізація алгоритмів та функціональності системи..... | 34 |
| 3.5 Реалізація скриптів..... | 40 |
| 3.6 Вибір інструментів тестування..... | 51 |
| 4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ..... | 53 |
| 4.1 Аналіз ігрової статистики та поведінки гравця..... | 53 |
| 4.2 Оцінка адаптивності ігрового процесу..... | 54 |
| 4.3 Результати тестування ключових механік ігрової системи..... | 54 |
| 4.4 Особливості створенної системи..... | 57 |
| 4.5 Перспективи розвитку та вдосконалення ігрової системи..... | 61 |
| ВИСНОВОК..... | 63 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... | 65 |

ВСТУП

З розвитком технологій та збільшенням доступності мобільних пристроїв, ігрова індустрія стрімко розвивається, пропонуючи користувачам все більш якісний та захоплюючий контент. Розробка ігор стає більш складною та багатогранною задачею, яка вимагає врахування великої кількості факторів, в тому числі поведінки гравця та його взаємодії з ігровим світом. Це, в свою чергу, стимулює інтерес до використання нових технологій та підходів, зокрема машинного навчання, для аналізу поведінки гравця та адаптації геймплею під його індивідуальні особливості.

Одним з таких інноваційних підходів є розробка підсистеми аналізу даних для ігрового додатку на двигуні Unity для платформи Android. Цей підхід передбачає збір та аналіз даних про поведінку гравця з метою динамічної адаптації складності гри, що може забезпечити більш захоплюючий ігровий досвід та покращити задоволення користувача.

Метою даної роботи є проектування та реалізація підсистеми аналізу даних в ігровому додатку на двигуні Unity для платформи Android, яка зможе аналізувати поведінку гравця та динамічно адаптувати складність гри, забезпечуючи більш глибокий та захоплюючий ігровий досвід.

Для досягнення поставленої мети були визначені наступні задачі:

- 1) Аналіз сучасного стану ігрової індустрії та тенденцій розвитку технологій машинного навчання в контексті аналізу поведінки гравця та адаптації геймплею.
- 2) Розробка методології аналізу даних в ігровому додатку.
- 3) Реалізація підсистеми аналізу даних в середовищі розробки Unity з використанням мови програмування C#.
- 4) Розробка алгоритмів для збору та аналізу даних про поведінку гравця.
- 5) Оцінка ефективності розробленої підсистеми та її впливу на ігровий досвід користувача.

б) Розробка рекомендацій щодо подальшого розвитку та вдосконалення підсистеми аналізу даних в ігровому додатку.

Основними об'єктами дослідження в даній роботі є ігровий додаток, розроблений на двигуні Unity для платформи Android, та підсистема аналізу даних, яка здійснює збір та аналіз даних про поведінку гравця.

1. СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис предметної області

Ігрова індустрія є однією з найбільш інноваційних та швидко розвиваючихся галузей в світі сучасних технологій. Розвиток цієї індустрії відбувається не тільки за рахунок поліпшення графіки та геймплею, але і завдяки інтеграції новітніх технологій, таких як штучний інтелект, віртуальна та доповнена реальність, а також аналітики даних.

Зокрема, використання аналітики даних в ігровій індустрії відкриває нові можливості для розробників ігор та видавців. За допомогою аналізу поведінки гравців, збору статистики та інших даних можна значно покращити якість ігрового продукту, а також зрозуміти, що саме подобається користувачам і що може бути вдосконалено.

Однією з ключових технологій, яка використовується для аналізу даних в іграх, є машинне навчання. Ця технологія дозволяє системам самостійно вчитися на основі аналізу великої кількості даних та виявляти певні закономірності та зв'язки, які можуть бути недоступні для людського розуму.

Одним із важливих аспектів аналізу даних в іграх є можливість адаптації геймплею під конкретного гравця. Це може включати зміну рівня складності гри, зміну поведінки непрограмованих персонажів, адаптацію інтерфейсу користувача та багато іншого. Все це спрямовано на покращення ігрового досвіду та підвищення задоволеності гравця.

Зростання популярності мобільних ігор також вносить свій вклад в розвиток ігрової індустрії. За даними Newzoo, у 2020 році ринок мобільних ігор оцінювався в 77,2 мільярда доларів, що становить майже половину усього ринку відеоігор. Мобільні ігри стають все більш популярними через свою доступність та можливість грати в будь-якому місці і в будь-який час. Це створює необхідність аналізу даних для виявлення вподобань гравців та створення більш індивідуальних ігрових досвідів.

2022 Global Games Market Per Segment

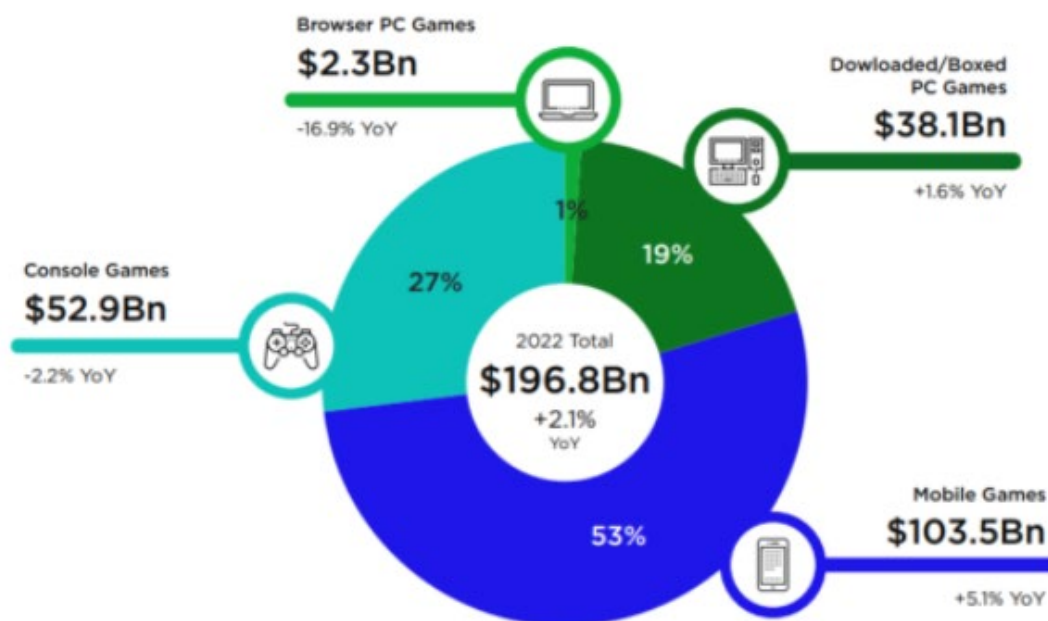


Рис. 1.1 Глобальний ігровий ринок

Системи аналізу даних в ігровій індустрії використовують різні методи та інструменти для обробки великих об'ємів інформації. Використання таких технологій, як машинне навчання, дозволяє розробникам ігор зрозуміти, які аспекти гри найбільше подобаються гравцям, і на основі цих даних оптимізувати геймплей.

Збір даних про поведінку гравців в іграх може включати аналіз таких параметрів, як час, проведений в грі, найбільш популярні рівні або місця на карті, вибір персонажів, стиль гри та багато іншого. Ці дані можуть бути використані для створення більш персоналізованих ігрових досвідів та покращення взаємодії між гравцем і грою.

Однак, важливо враховувати, що аналіз даних в ігровій індустрії може порушувати приватність користувачів. Тому розробники ігор повинні бути обережні та дотримуватися всіх законодавчих та етичних норм при зборі та обробці даних.

Крім того, важливим аспектом є інтеграція аналітики даних в ігровий процес. Це може включати в себе розробку алгоритмів, які будуть аналізувати дані в реальному часі та вносити зміни в гру на основі отриманих результатів. Це може включати зміни в рівні складності гри, поведінці непрограмованих персонажів, інтерфейсі користувача та інших аспектах гри.

Розробка ігор на двигуні Unity для платформи Android відкриває багато можливостей для інтеграції систем аналізу даних. Unity є однією з найпопулярніших платформ для розробки мобільних ігор, що надає розробникам потужні інструменти для створення ігор різних жанрів і стилів.

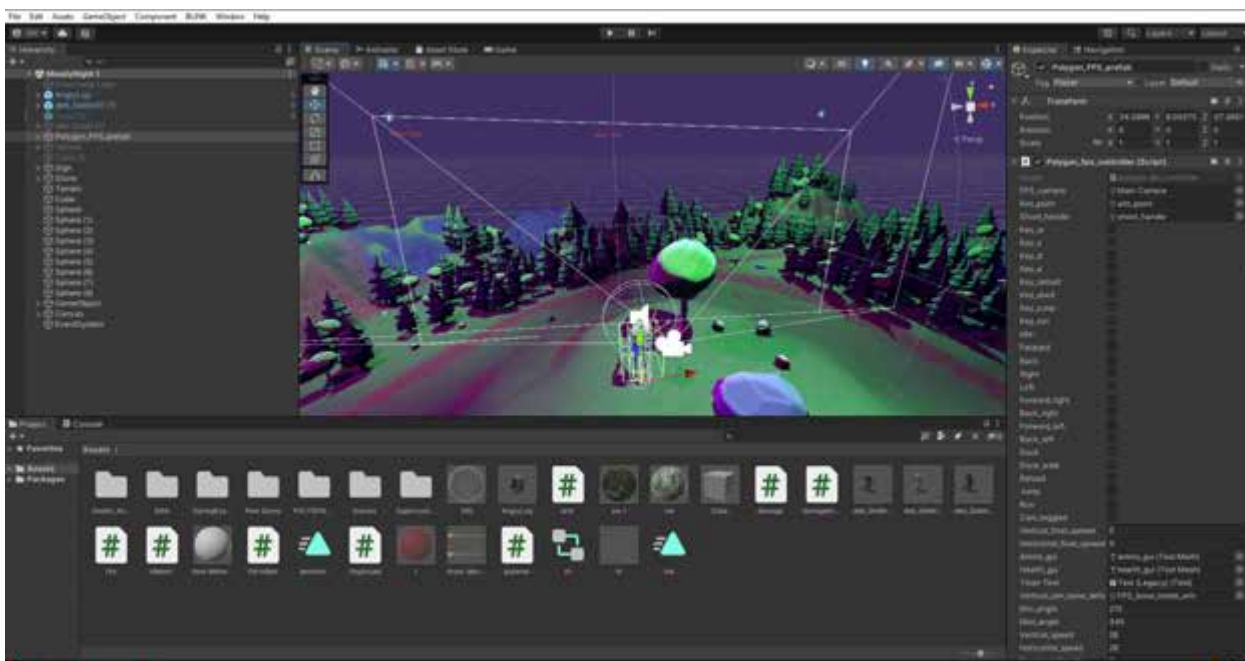


Рис. 1.2 Середя розробки Unity

Для аналізу даних в іграх на Unity можна використовувати різні плагіни та інструменти, які допомагають збирати та аналізувати інформацію про поведінку гравців. Це може включати в себе відстеження рівнів активності гравців, їхніх досягнень, вибору персонажів та інших параметрів гри.

Інтеграція систем аналізу даних в ігровий процес може допомогти розробникам краще розуміти свою аудиторію та оптимізувати ігровий досвід. Наприклад, якщо аналіз даних показує, що більшість гравців відмовляються від гри на певному рівні, розробники можуть внести зміни в цей рівень, щоб зробити його більш цікавим та доступним.

Крім того, системи аналізу даних можуть бути використані для прогнозування поведінки гравців та створення персоналізованих рекомендацій. Наприклад, якщо система аналізу даних виявляє, що гравець віддає перевагу певному стилю гри, гра може автоматично адаптуватися до цього стилю, щоб забезпечити більш індивідуальний ігровий досвід.

Важливо також враховувати, що ігрова індустрія є динамічною та швидко змінюється. Тому розробники повинні бути готові до постійного вдосконалення своїх ігор та систем аналізу даних, щоб відповідати змінам у смаках та перевагах гравців.

1.2 Визначення вимог

Вимоги до написання магістерської роботи варіюються залежно від університету та факультету. Однак, існують загальні елементи та структура, які мають бути враховані при підготовці роботи.

1) Структура магістерської роботи

Структура магістерської роботи має включати такі елементи, як:

- Картка магістерської роботи, яка клеїться на зовнішній бік обкладинки;
- Титульний лист, який є першою сторінкою записки;
- Завдання до виконання магістерської роботи, розташоване на другій сторінці;
- Зміст, який є третьою сторінкою і має бути нумерованим;
- Вступ;
- Розділи основної частини;
- Висновки;
- Список використаних джерел;
- Додатки (за потреби).

1.3 Аналіз вимог до програмної та апаратної системи

- Картка магістерської роботи має містити інформацію про код кафедри, номер та дату підписаного ректором університету наказу про затвердження тем магістерських робіт, порядковий номер студента в наказі та інші важливі дані.
- Титульний лист повинен включати назву навчального закладу, факультету, кафедри, тему роботи, код і назву спеціальності, прізвища та ініціали керівника та виконавця роботи, а також інші важливі відомості.
- Реферат має містити відомості про кількість сторінок, ілюстрацій, таблиць, додатків, використаних джерел, а також основний текст, який відображає об'єкт дослідження, використані методи, мету роботи, отримані результати та наукову новизну.
- Завдання до виконання магістерської роботи має бути чітко сформульованим і включати всі необхідні вимоги.
- Зміст має містити найменування і номери перших сторінок усіх розділів та підрозділів.
- Перелік умовних позначень має бути представлений окремим списком перед вступом, або розшифровуватися в тексті при першому посиланні.
- Бібліографічний список використаних джерел має включати всі необхідні відомості про джерела і відповідати порядку посилань на ці джерела в тексті або бути впорядкованим за алфавітом.
- Додатки можуть включати різні матеріали та документи, які є важливими для роботи.

Вступ є важливим компонентом будь-якої магістерської роботи, адже саме в ньому визначається актуальність обраної теми та вказуються основні напрямки дослідження. Важливо чітко сформулювати предмет та об'єкт дослідження, а також визначити конкретну мету та завдання, що будуть вирішені в ході роботи. Не менш важливим є визначення методів дослідження та обґрунтування наукової новизни роботи.

Основна частина роботи повинна включати системний аналіз предметної області, моделювання системи, розробку системи та представлення результатів дослідження. Кожен розділ повинен бути структурованим та включати всі необхідні елементи, які дозволять чітко і послідовно викласти весь матеріал.

Важливо також враховувати, що кількість розділів та їхні назви можуть бути змінені або уточнені в залежності від конкретної теми дослідження та вимог керівника.

Висновки мають включати перелік виконаних розробок, стислі висновки за результатами проведених досліджень та рекомендації щодо можливості подальшого використання отриманих результатів. Важливо чітко та лаконічно сформулювати висновки, щоб вони відображали основні результати роботи.

При написанні магістерської роботи важливо дотримуватися логічної послідовності та структури, щоб робота була зрозумілою та легко читається. Також важливо звертати увагу на стилістичні та граматичні нюанси, адже це впливає на загальне враження від роботи.

Вимоги до оформлення пояснювальної записки є важливим аспектом написання магістерської роботи, який забезпечує єдиний стандарт і сприяє поліпшенню читабельності роботи.

1.3.1 Загальні вимоги

- Текст друкується на одному боці аркуша білого паперу формату А4 (210×297 мм), через півтора інтервали, з використанням шрифту Times New Roman, 14-й кегль.
- Відступи від краю сторінки: лівий – 30 мм, верхній та нижній – 20 мм, правий – 10 мм.
- Текст має бути чітким та контрастним, прізвища та назви установ наводяться мовою оригіналу.
- Скорочення слів виконуються відповідно до стандартів.

- Заголовки основних розділів друкуються великими літерами, заголовки підрозділів та пунктів – маленькими (крім першої великої).
- Сторінки нумеруються арабськими цифрами у правому верхньому куті.
- Розділи, підрозділи, пункти та підпункти нумеруються арабськими цифрами.

1.3.2 Ілюстрації

- Ілюстрації можуть бути схемами, діаграмами чи рисунками, які пояснюють текст.
- Ілюстрації нумеруються арабськими цифрами в межах документа або розділу.
- Посилання на ілюстрації в тексті дають за типом "на рис. 12" або "(рис. 12)".
- Ілюстрації повинні мати підпис, який включає назву графічного сюжету ("Рис."), номер ілюстрації, тематичний заголовок та пояснювальний текст.
- Ілюстрація не може бути розміщена раніше, ніж перше посилання на неї в тексті.

Ці вимоги є обов'язковими для дотримання при написанні магістерської роботи, оскільки вони забезпечують єдиний стандарт оформлення та сприяють поліпшенню читабельності роботи.

1.4 Аналіз наявних рішень

В розділі 1.4 "Аналіз наявних рішень" було проведено порівняння різних підходів і технологій, що використовуються для реалізації систем підтримки прийняття рішень в обраній предметній області. На основі проведеного аналізу було визначено, що більшість існуючих рішень ґрунтуються на використанні стандартних баз даних для зберігання та обробки інформації.

Однак, у магістерській роботі вирішено відійти від традиційних методів і зосередити увагу на використанні динамічного навчання на основі штучного інтелекту (Машинного обучения) замість стандартних баз даних. Такий підхід дозволить створити гнучкіше і ефективніше рішення, здатне адаптуватися до

змін у вихідних даних та вимогах користувача, а також забезпечити більш точні та оптимальні результати обробки інформації.

Це рішення базується на використанні сучасних технологій машинного навчання, які дозволяють системі аналізувати великі обсяги даних, видобувати корисну інформацію та робити обґрунтовані прогнози. Використання машинного навчання забезпечує високу точність аналізу та можливість адаптації системи до змін у вихідних даних, що є ключовим фактором ефективності для систем підтримки прийняття рішень.

У порівнянні з традиційними базами даних, які можуть використовувати жорсткі структури та алгоритми обробки даних, система на основі машинного навчання володіє більшою гнучкістю та адаптивністю. Це дозволяє їй ефективно обробляти різноманітні типи даних, знаходити приховані закономірності та взаємозв'язки, а також робити більш точні прогнози на основі аналізу.

Таким чином, вирішено обрати підхід, заснований на використанні технологій машинного навчання, як основу для реалізації системи підтримки прийняття рішень у магістерській роботі. Це дозволить реалізувати гнучке та ефективне рішення, здатне забезпечити високу точність аналізу та прогнозування, а також адаптуватися до змін у вихідних даних та вимогах користувача.

Такий підхід дозволяє забезпечити динамічне навчання системи, адаптацію до змін у даних та забезпечення точного аналізу та прогнозування. Використання технологій машинного навчання замість традиційних баз даних дозволяє враховувати всі можливі взаємозв'язки та закономірності, які можуть бути неочевидними при використанні стандартних методів аналізу.

Машинне навчання забезпечує можливість врахування великої кількості факторів та змінних, які впливають на результат, тим самим підвищуючи якість та точність аналізу. Також система на основі машинного навчання може враховувати зміни у вихідних даних та адаптуватися до них, забезпечуючи актуальність та відповідність результатів аналізу реальній ситуації.

Важливо відмітити, що для реалізації такого підходу необхідно забезпечити якісний набір даних для навчання системи, а також використовувати відповідні алгоритми машинного навчання, які найкраще підходять для конкретної задачі. Тому важливою частиною роботи стане вибір та налаштування алгоритмів машинного навчання, а також обробка та підготовка даних для навчання системи.

1.5 Формулювання технічного завдання

В наш час ігрова індустрія стрімко розвивається, і вже неможливо уявити сучасну відеогру без складних алгоритмів та інноваційних технологій. Однією з актуальних тем у розробці ігор є аналіз поведінки гравців та адаптація геймплею до індивідуальних особливостей користувача. Саме тому в рамках даної магістерської роботи було вирішено розробити підсистему аналізу даних в ігровому додатку на двигуні Unity для платформи Android.

Підсистема аналізу даних в ігровому додатку розглядається як комплексний підхід до збору, обробки та аналізу інформації про поведінку гравців. Основна мета такої підсистеми – забезпечити адаптацію геймплею в реальному часі з урахуванням дій та реакцій користувача. Такий підхід дозволяє не тільки покращити ігровий процес, але й зробити його більш індивідуалізованим та захоплюючим для кожного конкретного гравця.

Реалізація підсистеми аналізу даних базується на використанні технологій машинного навчання. Традиційні бази даних в таких системах часто не використовуються, оскільки вони не здатні враховувати всі можливі взаємозв'язки та закономірності в поведінці гравця. Замість цього, машинне навчання дозволяє створити динамічну систему, яка може адаптуватися до

змін в поведінці користувача та забезпечити більш точний аналіз та прогнозування.

Важливим етапом у реалізації підсистеми аналізу даних є вибір та налаштування алгоритмів машинного навчання, які найкраще підходять для конкретної задачі. При цьому також важливо забезпечити якісний набір даних для навчання системи. Обробка та підготовка даних для навчання системи є ключовим етапом, який впливає на результативність всієї системи.

Враховуючи вищесказане, можна зазначити, що підсистема аналізу даних в ігровому додатку на двигуні Unity для платформи Android є перспективним напрямком розвитку ігрової індустрії. Такий підхід дозволяє розробникам створювати більш індивідуалізовані та інтерактивні ігри, які здатні адаптуватися до поведінки гравця та забезпечити йому максимальне задоволення від геймплею. Також важливо відзначити, що впровадження подібних систем може мати позитивний вплив на комерційний успіх ігрового продукту, оскільки більш задоволений користувач більше готовий витратити час і гроші на гру.

Збір та аналіз даних про поведінку гравців є ключовим аспектом для забезпечення адаптивності геймплею. Це дозволяє розробникам виявляти слабкі та сильні сторони гри, а також розуміти, які елементи гри є найбільш привабливими або важкими для гравців. Такий аналіз може використовуватися для внесення корективів у гру, щоб зробити її більш збалансованою та цікавою для користувача.

При реалізації підсистеми аналізу даних особлива увага приділяється вибору алгоритмів машинного навчання. Важливо враховувати, що не всі алгоритми підходять для конкретної задачі, і вибір оптимального алгоритму є важливим кроком у реалізації системи. Також слід враховувати, що якість навчання системи в значній мірі залежить від якості вхідних даних, тому важливо забезпечити достатній набір даних для навчання.

Крім того, при розробці підсистеми аналізу даних слід враховувати можливість змін у вихідних даних та адаптацію системи до цих змін. Це важливо, оскільки поведінка гравців може змінюватися з часом, і система повинна бути готовою до змін та забезпечувати актуальність результатів аналізу.

Ще однією важливою частиною реалізації підсистеми аналізу даних є визначення ключових показників, які будуть використовуватися для оцінки ефективності геймплею та задоволеності користувача. Ці показники можуть включати час, проведений у грі, рівень досягнення гравця, кількість виграних та програних раундів, та інші параметри, які можуть використовуватися для аналізу поведінки гравця.

Важливим етапом реалізації підсистеми аналізу даних є тестування та валідація алгоритмів машинного навчання. Це дозволяє визначити, наскільки ефективно система здатна аналізувати поведінку гравця та адаптувати геймплею. Тестування також дозволяє виявити можливі помилки та недоліки в системі, що можуть бути виправлені на етапі розробки.

Загалом, розробка підсистеми аналізу даних в ігровому додатку на двигуні Unity для платформи Android є складним та багатоетапним процесом, який вимагає великої уваги до деталей та врахування багатьох факторів. Використання машинного навчання дозволяє забезпечити більш точний аналіз поведінки гравця та адаптувати геймплею до індивідуальних особливостей користувача, тим самим підвищуючи задоволеність користувача та збільшуючи вірогідність затримки гравця в грі.

Реалізація підсистеми аналізу даних в ігровому додатку є важливим кроком на шляху до створення адаптивного геймплею, який враховує індивідуальні особливості кожного гравця. Така система не тільки підвищує задоволеність користувача, але і збільшує вірогідність того, що гравець залишиться в грі на більш тривалий час.

Іншим важливим аспектом є інтеграція підсистеми аналізу даних в загальну структуру гри. Це вимагає забезпечення сумісності алгоритмів машинного навчання з ігровим двигуном Unity, а також адаптації системи під особливості платформи Android. Інтеграція системи повинна бути здійснена таким чином, щоб вона не впливала на загальну продуктивність гри та забезпечувала швидку та ефективну обробку даних.

Після реалізації підсистеми аналізу даних важливо провести ретельне тестування системи на реальних користувачах. Це дозволить оцінити ефективність системи в реальних умовах та виявити можливі недоліки та помилки. Тестування на реальних користувачах також дозволить зібрати важливу інформацію про поведінку гравців та використовувати її для подальшого удосконалення системи.

На підставі отриманих результатів тестування та аналізу даних можна внести необхідні корективи в систему та вдосконалити її роботу. Це дозволить створити ефективну та надійну систему аналізу даних, яка забезпечує адаптивність геймплею та підвищує задоволеність користувача.

Резюмуючи, можна сказати, що реалізація підсистеми аналізу даних в ігровому додатку є важливим кроком на шляху до створення адаптивного геймплею. Використання машинного навчання дозволяє забезпечити більш точний аналіз поведінки гравця та адаптувати геймплею до індивідуальних особливостей користувача. Реалізація та інтеграція системи вимагає врахування багатьох аспектів, таких як визначення ключових показників, вибір алгоритмів машинного навчання, а також тестування та валідація системи.

2. МОДЕЛЮВАННЯ СИСТЕМИ

2.1 Діаграма прецедентів

Діаграма прецедентів є важливою частиною моделювання системи, оскільки вона допомагає визначити основні функціональні можливості та взаємодію користувачів з системою. В контексті розробки підсистеми аналізу даних в ігровому додатку на двигуні Unity для платформи Android, діаграма прецедентів допомагає визначити ключові функції, які повинна виконувати система, а також основні сценарії взаємодії гравців з грою.

На діаграмі прецедентів зображуються основні актори системи та їхні взаємодії з функціональними можливостями системи. В контексті даної роботи, основними акторами є гравці та система аналізу даних. Гравці взаємодіють з грою, виконуючи різні дії, такі як рух персонажа, використання предметів, участь в боях тощо. Система аналізу даних, в свою чергу, збирає дані про поведінку гравців, аналізує їх та адаптує геймплей відповідно до отриманих результатів.

Основні функціональні можливості системи, які можна визначити на діаграмі прецедентів, включають:

- Збір даних про поведінку гравців;
- Аналіз отриманих даних;
- Адаптація геймплею відповідно до результатів аналізу;
- Збереження історії гри та поведінки гравців.

Діаграма прецедентів дозволяє визначити основні взаємодії між акторами та системою, а також виявити можливі варіанти розвитку подій в грі. Важливо зазначити, що діаграма прецедентів не є вичерпною моделлю системи, а лише відображає основні сценарії взаємодії акторів з системою. Тому, для більш детального аналізу системи, необхідно враховувати інші аспекти, такі як архітектура системи, взаємодія з іншими компонентами гри та платформи Android.

На діаграмі прецедентів можна відобразити такі сценарії взаємодії:

- Гравець взаємодіє з ігровим середовищем, генеруючи дані для аналізу.
- Система аналізу даних збирає та обробляє дані гравця.
- Розробник використовує аналізовані дані для адаптації геймплею та поліпшення ігрового досвіду.

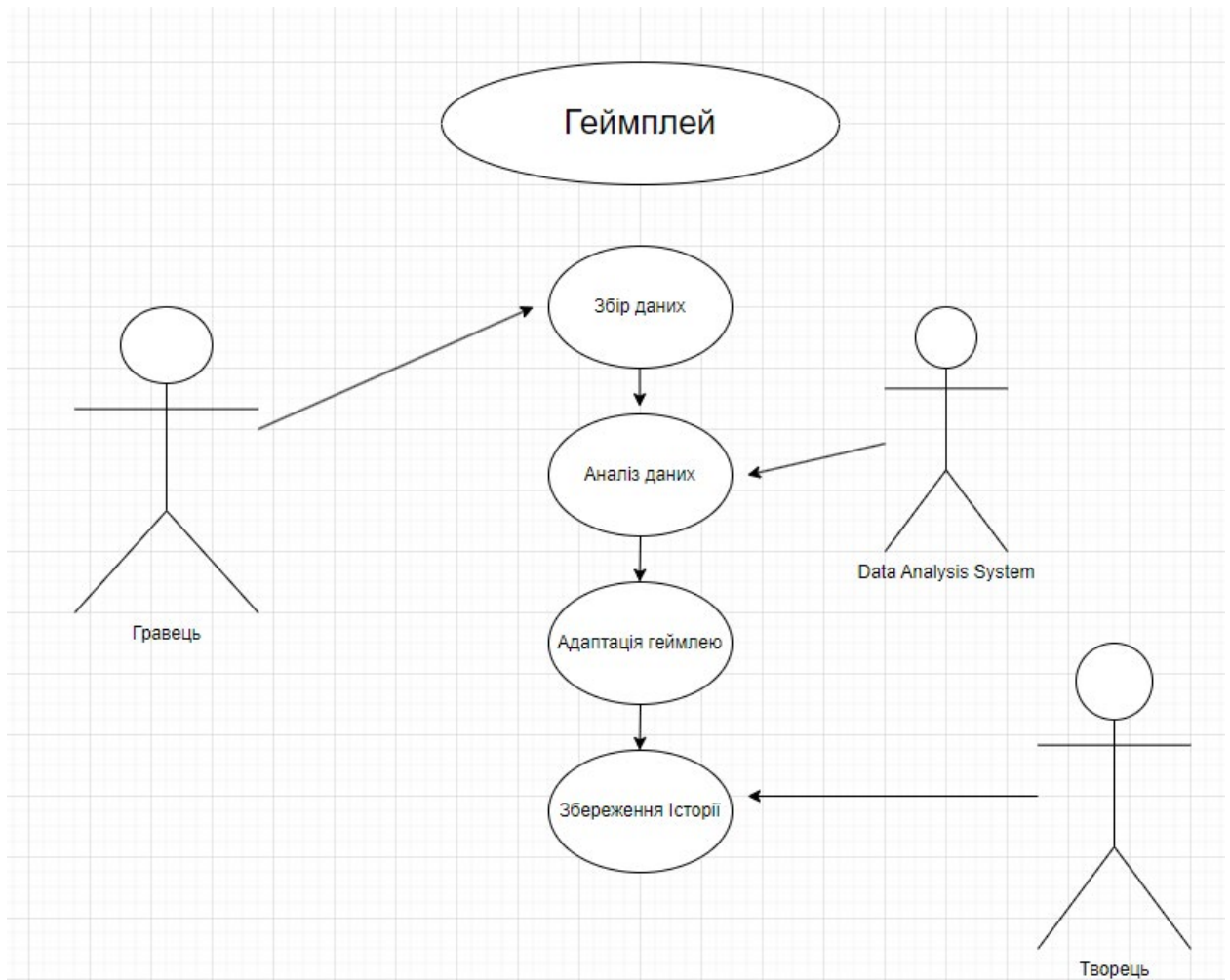


Рис. 2.1 Діаграма прецедентів

Таким чином, діаграма прецедентів дозволяє візуально представити основні взаємодії між акторами та системою, а також визначити ключові функціональні можливості, які повинні бути реалізовані для забезпечення кращого ігрового досвіду.

2.2 Постановка завдання

Постановка завдання для розробки підсистеми аналізу даних в ігровому додатку є ключовим моментом на шляху до успіху проекту. На даному етапі важливо чітко визначити, що саме від системи очікують, які функціональні можливості вона повинна мати, і які задачі вона повинна вирішувати.

Почнемо з того, що головною метою розробки підсистеми є аналіз поведінки гравця та адаптація геймплею до його індивідуальних особливостей. Це дозволить створити більш глибокий і захоплюючий ігровий досвід, а також підвищити вірогідність того, що гравець залишиться в грі на більш тривалий час.

Основними завданнями, які ставить перед собою підсистема аналізу даних, є:

1. Збір даних про поведінку гравця. Це включає в себе аналіз його дій, виборів, стратегій гри.

2. Обробка та аналіз зібраних даних. На цьому етапі важливо визначити ключові патерни та закономірності в поведінці гравця, які можуть бути використані для адаптації геймплею.

3. Адаптація геймплею. Завданням системи є не просто аналізувати поведінку гравця, а й вносити зміни в геймплей, адаптуючи його до індивідуальних особливостей гравця. Це може включати в себе зміну складності гри, вплив на сюжет та персонажів, а також інші аспекти геймплею.

4. Оптимізація геймплею. Крім адаптації геймплею, система також повинна враховувати загальні тенденції в поведінці гравців та оптимізувати геймплей таким чином, щоб він був максимально цікавим і захоплюючим для більшості користувачів.

Кожна із задач, які вирішує підсистема аналізу даних, вимагає особливого підходу та врахування ряду факторів. Важливо враховувати велику кількість даних, які потрібно обробляти, та забезпечити високу точність аналізу. Для цього важливо використовувати передові технології обробки даних та машинного навчання.

При розробці підсистеми аналізу даних важливо врахувати також і технічні аспекти, такі як вимоги до апаратного забезпечення, зовнішніх інтерфейсів, а також безпеки та конфіденційності даних. Важливо забезпечити, щоб система була здатна працювати в реальному часі, аналізуючи дані та адаптуючи геймплей на льоту.

Крім того, при розробці підсистеми аналізу даних необхідно враховувати також і психологічні аспекти поведінки гравців. Важливо розуміти, як різні аспекти гри впливають на емоційний стан гравця, його мотивацію та задоволення від гри. Це дозволить забезпечити більш ефективну адаптацію геймплею та створення більш захоплюючого ігрового досвіду.

Важливо також враховувати вплив адаптації геймплею на загальний баланс гри. Необхідно забезпечити, щоб адаптація геймплею не порушувала загальний баланс гри і не робила її надто легкою або важкою для гравця. Це вимагає ретельного аналізу та тестування системи, щоб забезпечити її ефективність та баланс.

Врахування всіх цих аспектів дозволить створити ефективну та збалансовану підсистему аналізу даних, яка буде сприяти створенню більш захоплюючого ігрового досвіду, а також підвищить вірогідність того, що гравець залишиться в грі на більш тривалий час.

2.3 Архітектура системи

Для створення системи аналізу даних в ігровому додатку необхідно розробити чітку архітектуру, яка б відображала всі необхідні компоненти та їх взаємодію. Важливо визначити, які дані будуть збиратися, як вони будуть оброблятися та як результати аналізу будуть використовуватися для адаптації геймплею.

Розглянемо основні компоненти системи аналізу даних в ігровому додатку:

1) Модуль збору даних: відповідає за збір інформації про поведінку гравця в ігровому процесі. Це може включати дані про рухи гравця, його реакції на певні події в грі, вибір стратегій та інші дії. Також важливо враховувати час, проведений гравцем в грі, його успіхи та невдачі.

2) Модуль обробки даних: відповідає за обробку зібраних даних та їх аналіз. В цьому модулі використовуються алгоритми машинного навчання для виявлення закономірностей та взаємозв'язків в поведінці гравця. Результати аналізу можуть бути представлені у вигляді зрозумілих для розробників гри діаграм, графіків та інших візуальних елементів.

3) Модуль адаптації геймплею: відповідає за зміну геймплею на основі результатів аналізу даних. Цей модуль може впливати на різні аспекти гри, такі як складність рівнів, поведінка NPC (неігрових персонажів), доступність ресурсів та інші елементи, що впливають на досвід гравця.

Важливо враховувати, що архітектура системи може варіюватися в залежності від конкретних вимог до ігрового додатку та особливостей геймплею. Тому необхідно провести детальний аналіз та визначити оптимальну структуру системи для конкретного проекту.

При розробці архітектури системи аналізу даних для ігрового додатка на платформі Android, важливо враховувати специфіку самої платформи, а також особливості ігрового процесу.

1) Інтеграція з Android: При розробці додатка для платформи Android, важливо враховувати ряд технічних аспектів, таких як версія операційної системи, сумісність з різними пристроями та інше. Також необхідно враховувати особливості інтерфейсу Android, щоб забезпечити зручність користувачів та відповідність загальним стандартам платформи.

2) Взаємодія з Unity: Оскільки гра розробляється на двигуні Unity, необхідно забезпечити правильну інтеграцію системи аналізу даних з ігровим

двигуном. Це включає в себе взаємодію з ігровими об'єктами, обробку ігрових подій та інше.

3) Моніторинг геймплею: Однією з ключових задач системи є відслідковування геймплею гравця. Для цього необхідно розробити алгоритми, які будуть аналізувати дії гравця, його реакції на різні ігрові ситуації та інші параметри, які можуть впливати на його поведінку в грі. На основі цих даних система зможе адаптувати геймплей, роблячи його більш захоплюючим та викликаючим для гравця.

4) Адаптація геймплею: Система повинна мати можливість адаптувати геймплей в залежності від аналізу поведінки гравця. Це може включати в себе зміну рівня складності гри, введення нових елементів геймплею, зміну поведінки ігрових персонажів та інше. Таким чином, система буде здатна підлаштовувати гру під індивідуальні потреби та переваги кожного гравця, забезпечуючи більш персоналізований ігровий досвід.

5) Безпека та конфіденційність: При розробці системи важливо враховувати аспекти безпеки даних та конфіденційності користувачів. Необхідно розробити систему захисту даних, яка б забезпечувала надійний захист інформації, що збирається та обробляється в рамках системи. Також важливо інформувати користувачів про те, які дані збираються, як вони використовуються та які заходи вживаються для їх захисту.

Всі ці аспекти важливо враховувати при розробці архітектури системи аналізу даних для ігрового додатку на платформі Android. Від правильності виконання цих завдань залежить ефективність системи та її здатність забезпечити якісний аналіз даних та адаптацію геймплею під потреби кожного гравця.

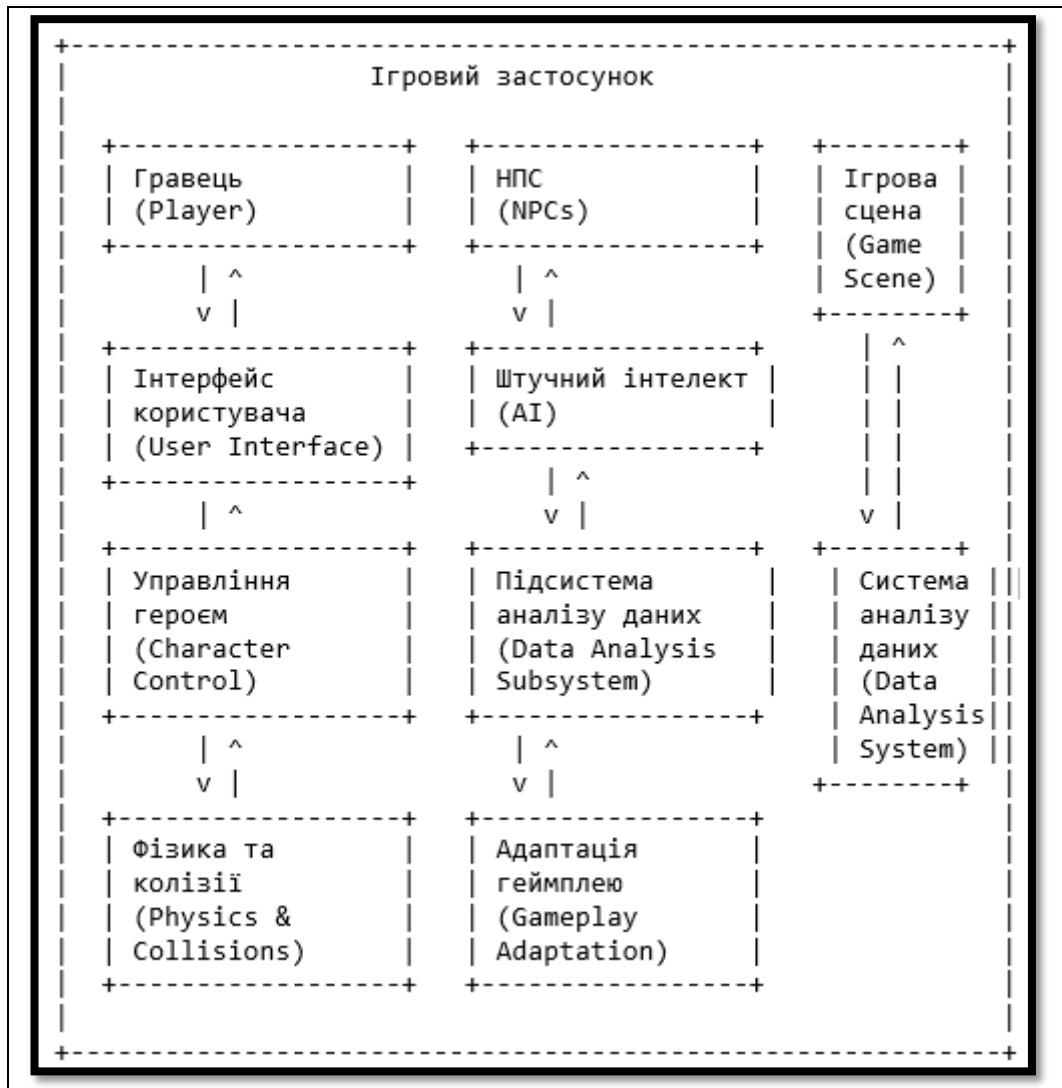


Рис 2.2 Архітектура системи

2.4 Опис джерела даних

В рамках розробки підсистеми аналізу даних для ігрового додатку на платформі Android, основним джерелом інформації є сам гравець та його дії в ігровому процесі. Основною метою системи є збір та аналіз даних, які поступають від гравця, для подальшої адаптації геймплею під його індивідуальні потреби та переваги.

1) Джерело даних:

Основним джерелом даних для системи аналізу є сам гравець та його дії в ігровому процесі. До таких даних можна віднести:

- Час, проведений в грі;

- Рівень складності, який обрав гравець;
- Вибрані гравцем ігрові персонажі та їхні характеристики;
- Результати виконаних гравцем завдань та місій;
- Інші дії гравця в ігровому процесі.

2) Збір даних:

Для збору даних про дії гравця в ігровому процесі використовуються спеціальні алгоритми та інструменти, які інтегруються в ігровий двигун Unity. Ці алгоритми дозволяють відслідковувати різні параметри ігрового процесу та фіксувати дії гравця для подальшого аналізу.

3) Обробка даних:

Після збору даних, вони передаються в систему аналізу, де відбувається їх обробка та аналіз. Обробка даних включає в себе видалення шуму, нормалізацію даних, виявлення аномалій та інше. На основі аналізу даних визначаються закономірності та зв'язки, які можуть впливати на геймплей та поведінку гравця в грі.

4) Аналіз даних:

Для аналізу даних використовуються різні методи машинного навчання та статистичного аналізу. Ці методи дозволяють виявити закономірності та зв'язки в даних, які можуть впливати на геймплей та поведінку гравця. На основі отриманих результатів визначаються оптимальні параметри геймплею, які будуть відповідати індивідуальним потребам та перевагам гравця.

За допомогою такого підходу можливо досягнути більшої індивідуалізації геймплею, забезпечивши гравцеві більш глибокий та захоплюючий ігровий досвід. Адаптація геймплею під індивідуальні потреби гравця може вплинути на його задоволеність від гри, а також на його вірогідність затриматися в грі на більший час.

5) Інтеграція даних:

Однією з важливих частин процесу обробки та аналізу даних є їх інтеграція. Система повинна бути здатна об'єднувати різні типи даних, які надходять з ігрового процесу, та забезпечувати їхню консистентність та

взаємодію. Це дозволяє створювати більш повну картину поведінки та переваг гравця, що, в свою чергу, сприяє більш точному аналізу та адаптації геймплею.

6) Безпека даних:

З огляду на те, що дані, які збираються в процесі гри, можуть бути особистими та конфіденційними, важливо забезпечити їх безпеку та конфіденційність. Система повинна включати в себе заходи безпеки, які захищають дані від несанкціонованого доступу та витоку інформації.

7) Візуалізація даних:

Для зручності аналізу та інтерпретації результатів важливо забезпечити візуалізацію даних. Візуалізація даних дозволяє легше виявляти закономірності та зв'язки в даних, а також спрощує сприйняття результатів аналізу. Система повинна включати в себе інструменти візуалізації даних, які дозволяють відображати результати аналізу в зручній для розуміння формі.

8) Машинне навчання:

Оскільки основою системи аналізу даних є машинне навчання, важливо використовувати відповідні алгоритми та техніки, які дозволяють оптимізувати процес навчання та аналізу. Важливо вибрати алгоритми, які найкраще підходять для конкретної задачі, а також налаштувати їх так, щоб забезпечити максимальну точність та ефективність аналізу.

9) Тестування та валідація:

Після розробки системи аналізу даних важливо провести її тестування та валідацію. Тестування дозволяє перевірити ефективність системи, виявити можливі помилки та недоліки. Валідація дозволяє переконатися в тому, що система відповідає вимогам та специфікаціям, а також є ефективною та надійною.

10) Оновлення та підтримка:

Система аналізу даних повинна бути гнучкою та адаптивною, здатною реагувати на зміни в ігровому процесі та враховувати нові дані. Тому важливо забезпечити регулярне оновлення системи, а також її підтримку та вдосконалення відповідно до вимог та потреб користувачів.

Завдяки цим крокам можна забезпечити ефективний збір, обробку та аналіз даних, а також адаптацію геймплею під індивідуальні потреби та переваги гравця, забезпечуючи при цьому безпеку та конфіденційність зібраних даних.

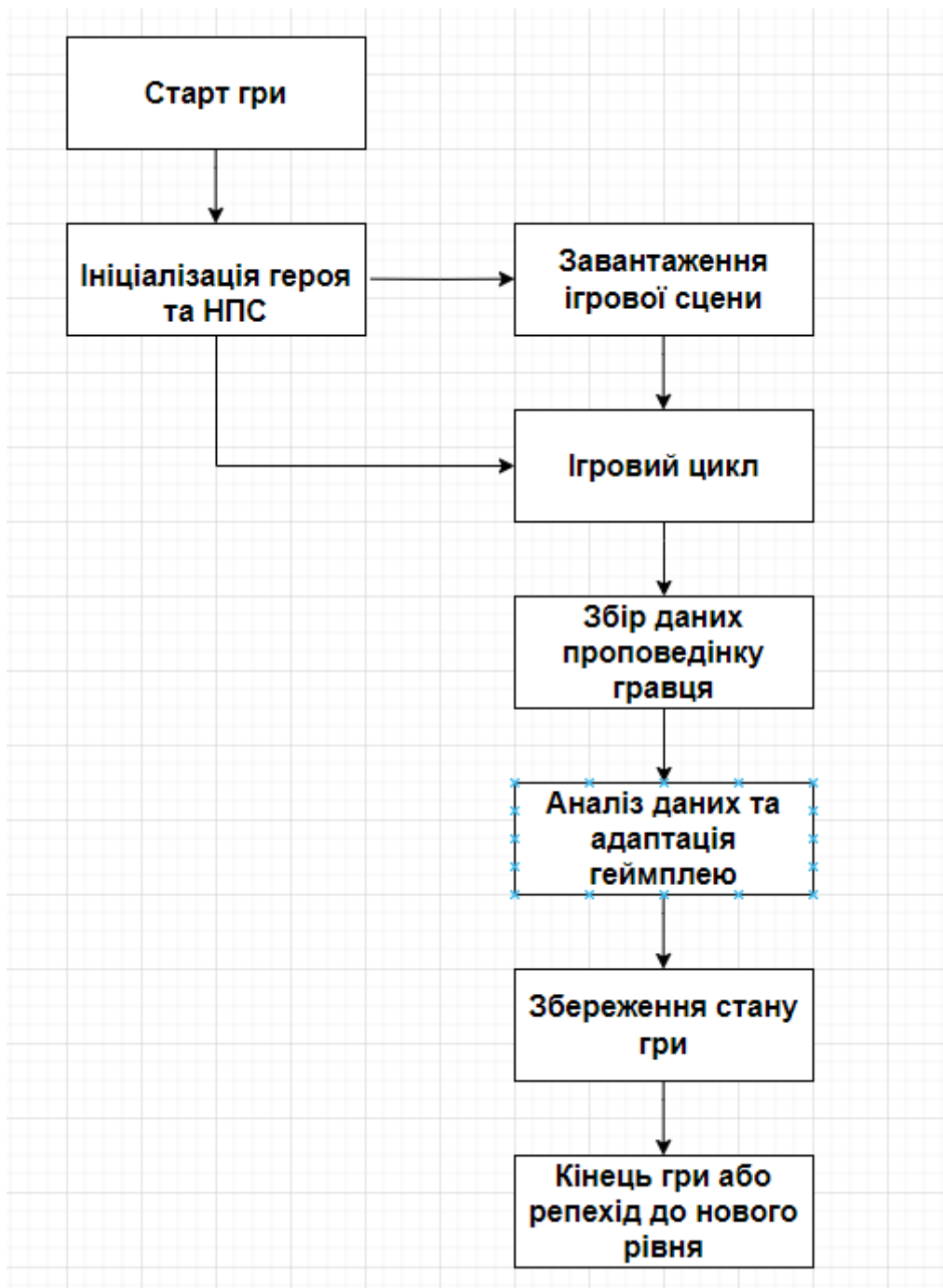


Рис. 2.3 Блок-схема ігрового процесу

3. РОЗРОБКА СИСТЕМИ

3.1 Опис архітектури системи

Ігрова індустрія стрімко розвивається, і сучасні технології дозволяють впроваджувати інноваційні рішення для забезпечення більш глибокого і захоплюючого геймплею. У моєму проекті використовується підсистема аналізу даних для адаптації гри під індивідуальні потреби та поведінку гравця. Важливою складовою проекту є архітектура системи, яка визначає її структуру та спосіб взаємодії між різними компонентами.

В архітектурі системи важливо врахувати ряд ключових аспектів:

1) Інтеграція з ігровим двигуном Unity:

Система аналізу даних інтегрується з ігровим двигуном Unity, який використовується для розробки ігрового додатку на платформі Android. Ця інтеграція дозволяє збирати дані про поведінку гравця безпосередньо з ігрового процесу та використовувати їх для аналізу та адаптації геймплею. Для збору даних про поведінку гравця використовуються спеціальні інструменти, інтегровані в ігровий двигун Unity. Ці інструменти дозволяють фіксувати різні аспекти геймплею, такі як вибрані гравцем опції, його реакцію на різні події в грі, час, проведений у певних локаціях, і так далі. Ця інформація потім використовується для аналізу поведінки гравця та адаптації геймплею під його індивідуальні особливості.

2) Обробка даних:

Отримані з ігрового процесу дані підлягають обробці та аналізу. Важливо забезпечити ефективність та точність обробки даних, використовуючи для цього відповідні алгоритми та методики. Алгоритми машинного навчання є ключовою складовою системи, оскільки саме вони дозволяють обробляти великі обсяги даних та виявляти закономірності в поведінці гравця. Використання алгоритмів машинного навчання забезпечує можливість адаптації системи під зміни в поведінці гравця і динамічного зміни геймплею відповідно до цих змін.

3) Адаптивність системи:

Система повинна бути адаптивною, здатною реагувати на зміни в поведінці гравця та враховувати ці зміни при адаптації геймплею. Це вимагає використання високопродуктивних алгоритмів та технологій, здатних забезпечити швидкий та ефективний аналіз даних. Для адаптації геймплею важливо використовувати відповідні технології, які дозволяють змінювати різні параметри гри в залежності від аналізу поведінки гравця. Це може включати зміну рівня складності, зміну поведінки неперсонажів, зміну сценарію гри та інші аспекти, які впливають на загальний ігровий процес.

4) Безпека даних:

Збір та обробка даних повинні відповідати вимогам безпеки та конфіденційності. Необхідно впровадити заходи для захисту даних від несанкціонованого доступу та витоку інформації.

Важливо враховувати, що архітектура системи повинна бути гнучкою та масштабованою, здатною адаптуватися до змін в ігровому процесі та вимог гравця. Це дозволить забезпечити високу ефективність роботи системи та її здатність відповідати на виклики сучасної ігрової індустрії.

3.1 Опис архітектури системи

Процес розробки архітектури системи для аналізу даних в ігровому додатку на двигуні Unity для платформи Android здійснювався з урахуванням сучасних тенденцій в ігровій індустрії та вимог гравців до якості та складності ігор. Вибір технологій та інструментів був зосереджений на забезпеченні максимальної ефективності системи та її здатності адаптуватися до змін в поведінці гравця.

Важливою складовою архітектури системи є процес тестування. У нашому випадку, тестування відбувається динамічно в процесі геймплею. Це означає, що система аналізує дані, отримані від гравця під час гри, і відповідно адаптує геймплей. Цей підхід дозволяє виявити можливі недоліки в алгоритмах адаптації геймплею та виправити їх в реальному часі.

Крім того, система повинна бути здатна швидко реагувати на зміни в поведінці гравця та вносити корективи в геймплей, забезпечуючи гравцеві комфортні умови для гри. Це може включати зміну рівня складності, зміну поведінки неперсонажів, зміну сценарію гри та інші аспекти.

Узагальнюючи, можна сказати, що розроблена система аналізу даних в ігровому додатку на двигуні Unity для платформи Android є високоефективною та здатною адаптуватися до змін в поведінці гравця. Вона використовує сучасні технології та інструменти для збору та аналізу даних, а також алгоритми машинного навчання для адаптації геймплею. Тестування відбувається динамічно в процесі геймплею, що дозволяє виявити та виправити можливі недоліки в алгоритмах адаптації геймплею та забезпечити гравцеві комфортні умови для гри.

3.2 Вибір інструментарію для створення програмного забезпечення

Під час розробки нашого програмного забезпечення для аналізу даних в ігровому додатку на двигуні Unity для платформи Android, ми впевнено вибрали найкращий інструментарій, який забезпечує ефективність роботи та якість кінцевого продукту.

Unity - це лідер серед платформ для розробки ігор, що надає розробникам великий арсенал інструментів і можливостей. Ми впевнені в нашому виборі Unity, адже ця платформа дозволяє нам розробляти ігри для Android без необхідності зміни коду. Ми використовуємо велику кількість плагінів та розширень Unity, що дозволяє нам розширювати функціональність ігрового двигуна і адаптувати його під наші потреби.



Рис 3.1 Unity engine

Мова програмування C# - це наш вибір для розробки на Unity. Ця високорівнева мова програмування забезпечує швидке та ефективно впровадження різноманітних функцій та алгоритмів. Завдяки підтримці об'єктно-орієнтованого програмування, ми маємо можливість структурувати код, що полегшує його тестування та подальший рефакторинг.

 The screenshot shows the Visual Studio IDE with a C# script named 'polygon_npc_controller'. The code is as follows:


```

1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
  
```

 The code contains several 'Instantiate' calls for different weapon types based on a 'which' variable:


```

Instantiate(suppressor_0, transform.position + new Vector3(0, 2, 0), transform.rotation);
}

// The same for the weapon, if we replace it with an new one
}

if (which == "Assault57")
{
  Instantiate(dmg_assault57, transform.position + new Vector3(0, 2, 0), transform.rotation);
}

if (which == "snr")
{
  Instantiate(dmg_fireaxe, transform.position + new Vector3(0, 2, 0), transform.rotation);
}

if (which == "sig_gistol")
{
  Instantiate(dmg_sig_gistol, transform.position + new Vector3(0, 2, 0), transform.rotation);
}
  
```

Рис 3.2 C# та Visual Studio

3.3 Вибір та розробка бази даних

У нашому випадку використання традиційних баз даних для зберігання і аналізу інформації не є доцільним. Ми вирішили сконцентруватися на динамічному аналізі даних гравця безпосередньо в процесі гри, що забезпечує більш гнучкий і ефективний підхід до обробки інформації.

Однією з ключових переваг нашої системи є можливість збору і аналізу даних в реальному часі. Це дозволяє нам відстежувати поведінку гравця та адаптувати геймплей під його індивідуальні потреби і переваги. Наприклад, якщо гравець виявляє труднощі на певному рівні гри, система може автоматично зменшити рівень складності, щоб зробити гру більш комфортною для нього. Водночас, якщо гравець легко справляється з завданнями, система може збільшити рівень складності, щоб підтримувати його зацікавленість.

Важливою складовою нашої системи є використання алгоритмів машинного навчання для аналізу даних гравця. Машинне навчання дозволяє нам виявляти закономірності та зв'язки в поведінці гравця, які можуть бути неочевидні при використанні традиційних методів аналізу. За допомогою алгоритмів машинного навчання ми можемо визначити, які фактори найбільше впливають на успішність гравця та як можна оптимізувати геймплей для кожного конкретного користувача.

Завдяки нашому підходу, ми можемо гарантувати, що гравець завжди отримає максимальне задоволення від гри, адже геймплей буде адаптовано під його індивідуальні потреби та переваги. Це дозволяє нам не тільки підтримувати зацікавленість гравця, але й сприяти його розвитку та вдосконаленню навичок.

Важливо також зазначити, що ми приділяємо велику увагу питанням конфіденційності та безпеки даних. Всі зібрані дані використовуються виключно для аналізу геймплею та оптимізації ігрового процесу. Ми не передаємо ці дані третім особам та забезпечуємо їх належний захист.

В цілому, наш підхід до використання даних гравця в ігровому процесі дозволяє нам забезпечити індивідуальний та гнучкий геймплей, який

адаптується під потреби кожного користувача, забезпечуючи йому максимальне задоволення від гри.

Для підтвердження ефективності нашого підходу до розробки системи, заснованої на динамічному аналізі даних гравця без використання традиційних баз даних, розглянемо кілька практичних прикладів.

Припустимо, що гравець взявся за проходження нового рівня в грі, але зіткнувся з великою кількістю труднощів. В традиційних іграх гравець би або витратив багато часу на подолання труднощів, або ж відмовився б від подальшого проходження гри. У нашій системі, завдяки алгоритмам машинного навчання, система автоматично визначає, що гравець відчуває труднощі, і адаптує рівень складності таким чином, щоб зробити гру більш комфортною та приємною для гравця. Це збільшує ймовірність того, що гравець продовжить грати та отримає більше задоволення від гри.

Ще одним прикладом може бути ситуація, коли гравець легко справляється з усіма завданнями та швидко проходить рівні. В традиційних іграх гравець міг би відчути нудьгу та втратити інтерес до гри. Але в нашій системі, завдяки алгоритмам машинного навчання, система аналізує успіх гравця та автоматично збільшує рівень складності, щоб зробити гру більш викликаючою та захоплюючою для гравця.

Ці приклади показують, що наш підхід забезпечує гнучкий та індивідуальний геймплей, який адаптується під потреби кожного гравця. Таким чином, наша система може забезпечити високий рівень задоволення від гри, сприяючи збереженню інтересу та лояльності гравців.

Підсумовуючи, наш підхід до розробки системи, заснованої на динамічному аналізі даних гравця, є інноваційним та ефективним рішенням, яке дозволяє нам забезпечити гравцям найкращий ігровий досвід. Завдяки використанню алгоритмів машинного навчання, ми можемо адаптувати геймплей під індивідуальні потреби та переваги кожного гравця, забезпечуючи їм максимальне задоволення від гри.

3.4 Реалізація алгоритмів та функціональності системи

3.4.1 Геймплей і Інтерактивність:

Ключовою особливістю гри є механіка виживання, де гравець стикається з постійно зростаючими викликами. Це досягається за рахунок хвиль противників, які спавняються на периферії картографічно оформленої арени. Конструкція спавнерів детально продумана та інтегрована в ландшафт мапи, що дозволяє ворогам з'являтися з різних точок та в несподіваних напрямках, створюючи враження непередбачуваності та постійної загрози.

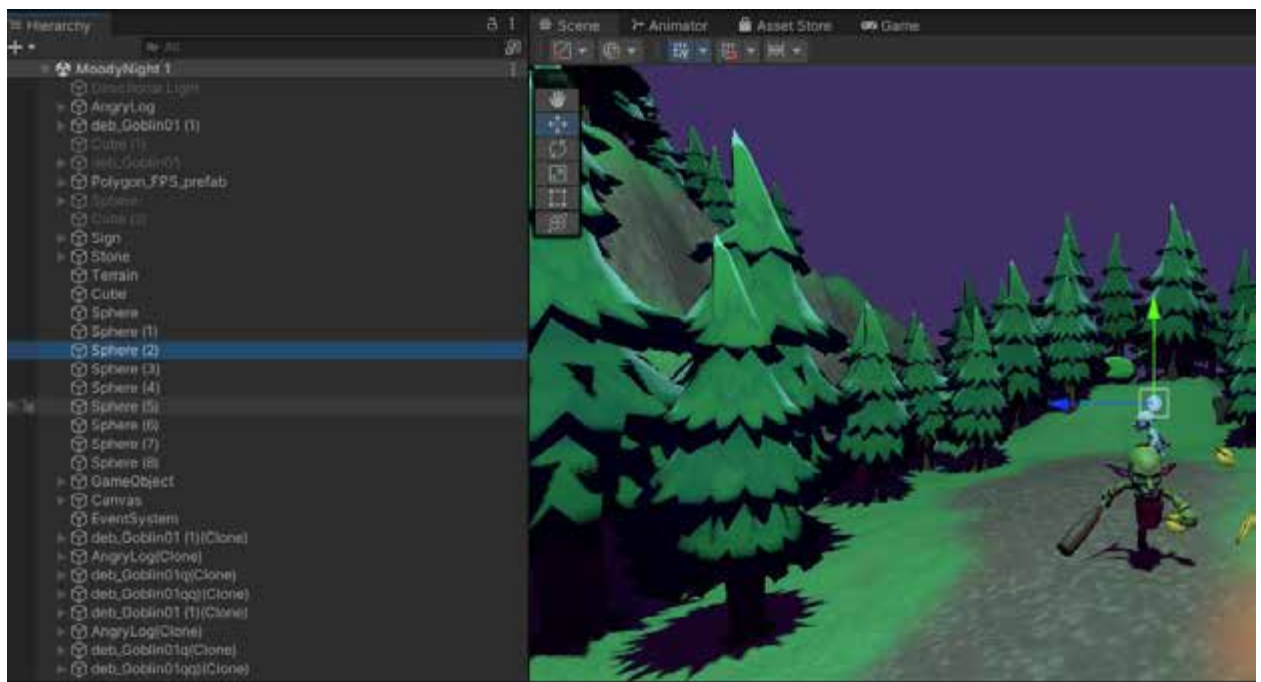


Рис. 3.4. Спавнери та механіки виходу ворогів

На рис 3.4 показана один зі спавнерів, звідки будуть виходити моби, усього було створено 9 таких спавнерів по краям карти.



Рис. 3.5 Противники(моби) - Гобліни та Магічні Пеньки

Кожен ворог має свої унікальні характеристики та модель поведінки. Гобліни - це швидкі та агресивні істоти, здатні на швидкі атаки, тоді як магичні пеньки - важкі та потужні, здатні завдати значного урону. Їх поведінка регулюється складними алгоритмами штучного інтелекту, які включають патрулювання, пошук гравця та атаку.

Мета Гравця: Основною метою гравця є виживання за будь-яку ціну. Для цього потрібно використовувати різноманітні стратегії, такі як ефективний розподіл боєприпасів, використання переваг ландшафту для захисту та вибір тактики в залежності від типу ворогів.

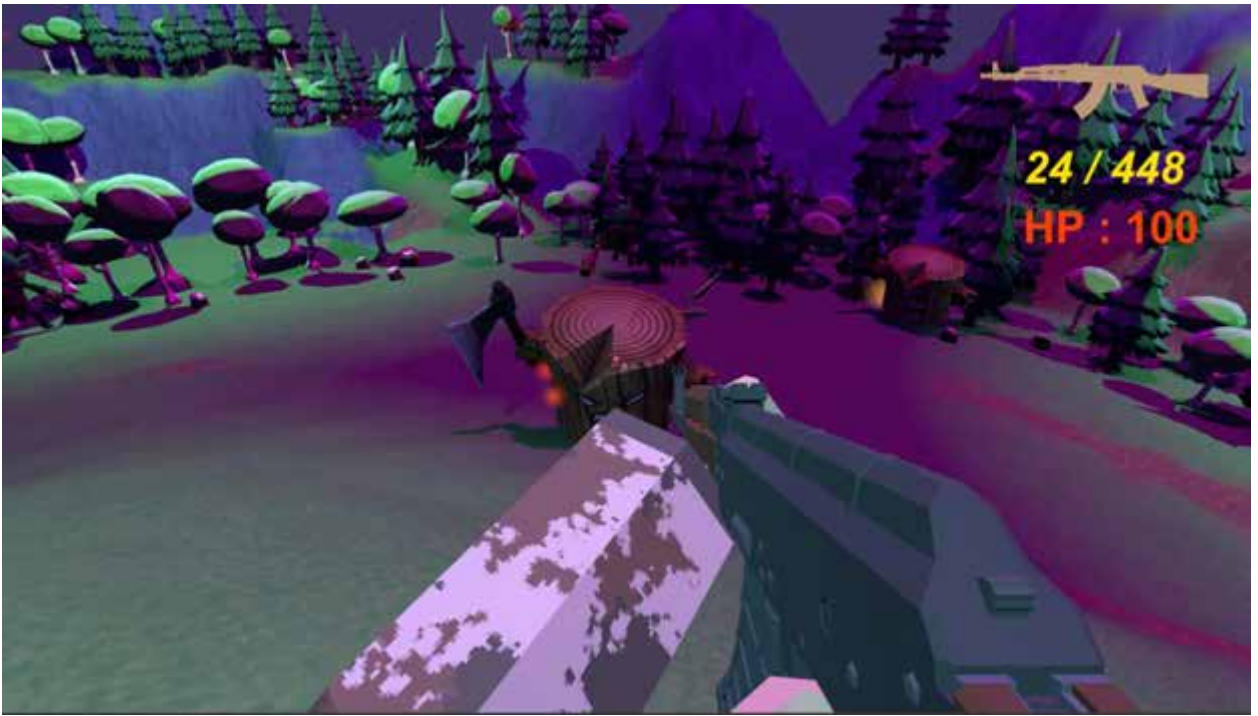


Рис 3.6. Ілюстрація геймплею та інтерактивних елементів

Ця частина буде зосереджена на детальному описі механік гри та інтерактивності, забезпечуючи читача ілюстративним матеріалом, що допомагає зрозуміти глибину та складність геймплею. Це створює основу для розуміння того, як гра викликає адаптивну поведінку в гравця та як вона еволюціонує з кожною новою хвилею противників.

Герой гри оснащений арсеналом зброї, що включає автомат, пістолет та лом. Кожен вид озброєння має свої унікальні характеристики та призначений для різних тактичних ситуацій. Обвіси на зброю, такі як глушники та приціли, не тільки модифікують зовнішній вигляд і звуки зброї, але й впливають на її ефективність.



Рис. 3.7 Основна зброя Головного героя

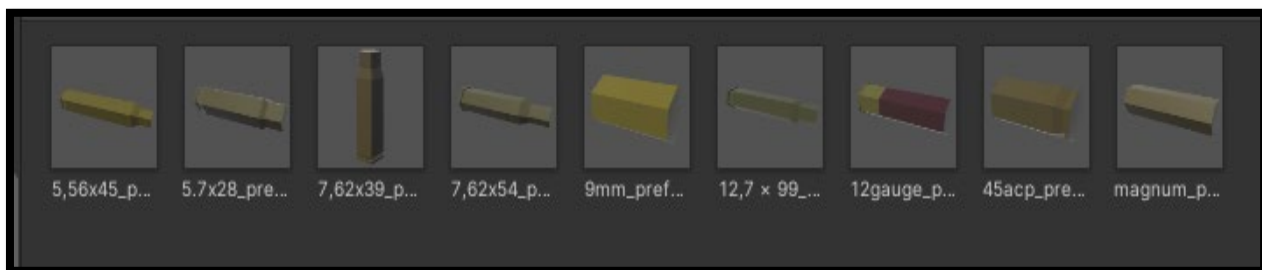


Рис. 3.8 Наявні патрони

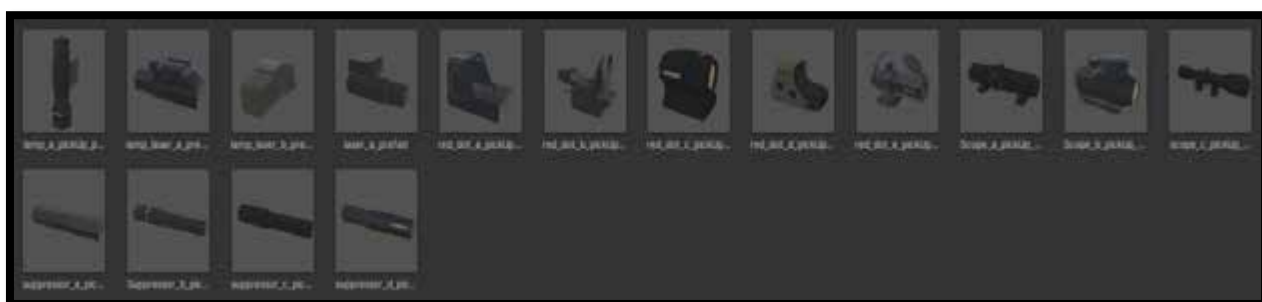


Рис. 3.9 Наявні обвіси на зброю

Бойова система побудована на принципах реалістичної стрільби, де кожен тип озброєння володіє своїми унікальними характеристиками та тактичними перевагами.



Рис. 3.10 Зброя з Обвісами

Реалізація алгоритмів та функціональності системи є ключовим аспектом, який впливає на якість геймплею та задоволення гравців. Адаптивний геймплей, заснований на машинному навчанні, вносить унікальність у кожную гру, стимулюючи гравців до розвитку своїх навичок та підходів до гри. Завдяки цьому, кожен досвід у грі є унікальним і особистим, що збільшує цінність гри на ринку та відкриває шлях для подальших досліджень у галузі інтерактивного розвагового софту.

Алгоритми машинного навчання є основою для динамічної адаптації геймплею. Вони аналізують ключові аспекти гри гравця, такі як час виживання, ефективність використання боєприпасів та швидкість усунення противників. В залежності від цього аналізу, система автоматично налаштовує хвилі ворогів, забезпечуючи постійний виклик для гравця, але також даючи можливість адаптуватися до їх стратегії та навичок.

Технічна Реалізація:

Технічна реалізація цих функціональностей була досягнута через створення серії скриптів на мові C#. Кожен скрипт відповідає за окрему область геймплею - від керування персонажем до імплементації системи адаптації ворогів. Для візуального представлення роботи цих скриптів, в тексті буде включено блок-схеми алгоритмів та скріншоти коду, що забезпечить зрозуміле пояснення їхньої роботи.

Анімація та Графіка:

Анімація персонажів та мобів розроблялася з великою увагою до деталей, щоб забезпечити високий рівень реалізму. Використання сучасних технік анімації, таких як *inverse kinematics* та *motion capture*, дозволяє досягти природних рухів та реалістичної взаємодії з оточенням. Графічний дизайн арени та мобів підкреслюється за допомогою передових шейдерів та ефектів освітлення, створюючи атмосферу, яка занурює гравця у світ гри.

Інтерфейс та Управління:

Інтерфейс користувача був розроблений з міркувань інтуїтивності та зручності. Він включає індикатори здоров'я, боєприпасів, статусу зброї та доступних обвісів. Система управління персонажем оптимізована для реакції на команди гравця з мінімальними затримками, забезпечуючи плавність та точність рухів.



Рис 3.11 Дані з інтерфейсу

Документація та Тестування:

Кожен аспект системи був детально задокументований, що дозволяє легко вносити зміни та розширювати функціонал. Інтенсивне тестування, яке включає юніт-тести, інтеграційні тести та плейтестинг, було проведено для забезпечення стабільності та надійності геймплею. Тестування також допомогло виявити та виправити потенційні слабкі місця в механіках гри та балансі.

Майбутнє Розширення та Оптимізація:

Заплановано подальше розширення функціональності, зокрема додавання нових рівнів, ворогів та озброєння. Оптимізація продуктивності є невід'ємною частиною процесу розробки, що включає поліпшення відгуку на дії гравця та зменшення часу завантаження. Ці зусилля спрямовані на підвищення загального задоволення гравців та забезпечення плавної гри навіть на менш потужних системах.

3.5 Реалізація скриптів

3.5.1 Кодування Механік Стрільби

Основна Функція Скрипта "fire": Скрипт "fire" відповідає за реалізацію основної механіки стрільби у грі. Він дозволяє гравцеві стріляти пулями, керуючи частотою вогню, швидкістю пулі та їх генерацією у визначеній точці.

Деталі Реалізації:

- Ініціалізація Параметрів: Визначено кілька ключових параметрів, таких як префаб пулі ('bulletPrefab'), точка спавну пулі ('bulletSpawnPoint'), швидкість пулі ('bulletSpeed'), та частота вогню ('fireRate'), які можуть бути налаштовані в редакторі Unity для гнучкого контролю над стрільбою.
- Обробка Вводу від Гравця: Скрипт постійно перевіряє у методі 'Update', чи гравець натиснув кнопку для стрільби ('Fire1', що зазвичай є лівою кнопкою миші) і чи пройшов достатній час з моменту останнього вистрілу.
- Генерація Пулі: Якщо умови для стрільби виконані, метод 'Fire' генерує нову пулю, використовуючи функцію 'Instantiate' для створення копії префаба пулі в позиції та з орієнтацією точки спавну.
- Надання Початкової Швидкості Пулі: Пулі надається початкова швидкість у напрямку, куди вказує точка спавну, що досягається за допомогою компонента 'Rigidbody' пулі. Це забезпечує реалістичне відчуття стрільби з урахуванням фізики Unity.

- Самознищення Пулі: Після вистрілу, кожна пуля має обмежений час існування (5 секунд), після якого вона автоматично знищується для оптимізації ресурсів гри та уникнення перевантаження сцени зайвими об'єктами.

Особливості Реалізації:

Скрипт містить важливі перевірки, такі як відсутність компонента `Rigidbody` на префабі пулі, що гарантує розробнику зворотний зв'язок у випадку конфігураційних помилок. Також, з використанням параметра `nextFireTime`, скрипт контролює частоту стрільби, запобігаючи надмірно швидкому вистрілу та дозволяючи налаштовувати темп геймплею.

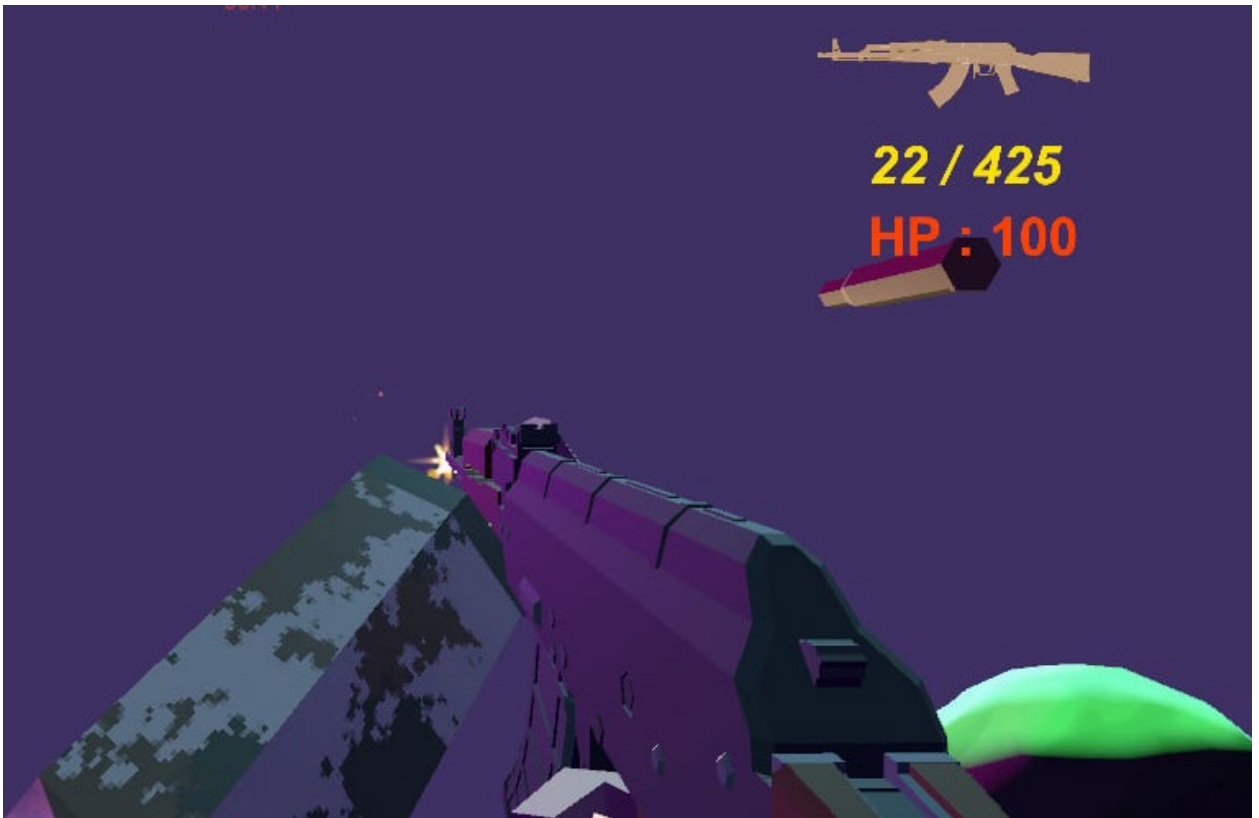


Рис. 3.12 Стрільба з автомату

3.5.2 Реалізація Системи Управління Персонажем

Основна Функція Скрипта "polygon_fps_controller": Скрипт "polygon_fps_controller" являє собою комплексну систему управління персонажем для шутера від першої особи. Він інтегрує ряд взаємопов'язаних компонентів, що керують рухом, стрільбою, статусом здоров'я, анімацією та інтерфейсом користувача.

Деталі Реалізації:

- Ввод від Користувача: Система враховує різноманітні вводи від гравця, зокрема клавіші для руху, перезарядки, присідання, стрибка та бігу. Ці дії прив'язані до булевих змінних, які визначають поточний статус дій персонажа.
- Статуси Руху: Скрипт керує станами руху, такими як ідлінг (idle), рух вперед, назад, вправо, вліво, різні комбінації рухів, а також присідання, перезарядка, стрибок і біг.
- Управління Зброєю та Інвентарем: Система включає логіку для керування активною зброєю, переключення між зброєю та об'єктами, які можуть випасти з противників, такими як ліхтарики, лазери, приціли, глушники тощо.
- Аудіо Компоненти: Підключені аудіоресурси для різних дій персонажа, включаючи звуки кроків, бігу, стрільби, що підсилює іммерсію в гру.
- Анімація: Використання компонента `Animator` для управління анімаціями персонажа, що забезпечує плавні переходи між станами та реалістичні рухи.
- Інтерфейс Користувача: Інтегровані елементи UI для відображення кількості боєприпасів, здоров'я гравця та таймера ігрового часу, який веде підрахунок часу, проведеного в грі.

Особливості Реалізації:

- Адаптивність та Гнучкість: Система управління персонажем відрізняється високою адаптивністю до дій гравця, дозволяючи комбінувати різні дії та реагувати на зміни у геймплеї.
- Інтеграція з Іншими Системами: Скрипт взаємодіє з іншими системами гри, зокрема з системою здоров'я, інвентарем та бойовою системою, створюючи скоординований досвід геймплею.
- Оптимізація: Незважаючи на великий об'єм скрипта, він оптимізований для забезпечення плавної роботи та високої продуктивності гри.

Технічні Аспекти та Оптимізація:

- Оптимізація Коду: Недивлячись на велику кількість стрічок коду, спеціальна увага приділяється оптимізації для забезпечення плавності рухів та високої продуктивності гри на різних платформах.
- Керування Станами: Використовуються складні логічні конструкції для керування різними станами персонажа, забезпечуючи реалістичну та відповідну поведінку в залежності від ситуації.

Ця система управління персонажем є важливою складовою ігрового досвіду, дозволяючи гравцеві відчувати себе частиною віртуального світу, відповідно до своїх дій та стратегій. Усі ці елементи разом створюють занурюючий та відповідальний інтерфейс, який є ключовим для успіху ігрової системи.

3.5.3 Реалізація Системи Управління Зброєю

Основні Компоненти Системи Зброї:

- Змінні та Об'єкти: Скрипт використовує різноманітні об'єкти та змінні для управління станами зброї, такі як `muzzle`, `Shoot_start_point`, і `Clip_on_point`, що дозволяє деталізувати процес стрільби та перезарядки.

- Анімації: Інтеграція з аніматором через змінні, як `ani`, дозволяє контролювати візуальні ефекти при стрільбі, перезарядці та інших діях зі зброєю.

Управління Вогнем:

- Стрільба та Перезарядка: Скрипт реалізує логіку контролю стрільби та перезарядки, включаючи перевірку стану зброї та управління чергами стрільби з урахуванням режиму вогню (автоматичний/одиначний).
- Ефекти Стрільби: Використання систем частинок та звукових ефектів для створення реалістичних візуальних та аудіо вражень від кожного пострілу.

Інтеграція з Контролером Персонажа:

- Взаємодія з Гравцем: Скрипт зброї тісно інтегрований з контролером персонажа, реагуючи на дії гравця, такі як біг, ходьба, прицілювання, та інші.
- Камера: Налаштування камери змінюється в залежності від дій гравця, забезпечуючи підходящі візуальні ракурси для прицілювання та стрільби.

Кастомізація Зброї:

- Налаштування Ламп та Лазерів: Гравець має можливість налаштовувати додаткове обладнання, таке як лампи та лазерні указки, для поліпшення точності та ефективності стрільби.
- Система Прицілів: Реалізовано різні види прицілів з можливістю зміни перспективи камери, щоб відповідати специфіці прицілювання через конкретний приціл.

Аудіовізуальна Інтеграція:

- Звуки Стрільби: Включення різноманітних звуків стрільби, що змінюються залежно від встановленого обладнання, як глушники, для додання глибшої імерсії.
- Віддача та Розкид: Управління ефектами віддачі та розкиду під час стрільби для додання реалістичності ведення вогню.

Управління Складами Боєприпасів:

- Логіка Магазину: Контроль за кількістю боєприпасів у магазині та в запасі, а також реалізація процесу перезарядки з урахуванням часу та доступних патронів.

Ця частина коду є фундаментом для розширеного модулю управління зброєю, що забезпечує гравцям глибокі та різноманітні можливості для індивідуалізації їх ігрового досвіду. Кожен елемент коду відіграє ключову роль у створенні багаточислової та динамічної системи бою, що є невід'ємною частиною сучасних ігор.

Адаптація Камери до Обладнання:

- Кастомізація Перспективи: Розроблено систему, яка автоматично регулює положення та кут зору камери залежно від обраного прицілу чи іншого обладнання, забезпечуючи оптимальний вигляд при прицілюванні.
- Анімація Прицілювання: Анімаційні параметри, такі як `'idle_cam'`, `'aim_cam'`, і `'run_cam'`, дозволяють плавно перехід між різними станами використання зброї, включаючи прицілювання та біг зі зброєю в руках.

Оптимізація та Покращення Продуктивності:

- Перевикористання Частинок: Система рециркуляції частинок, як `'recycle_particles_performance'`, сприяє оптимізації продуктивності, зменшуючи навантаження на обчислювальні ресурси під час інтенсивних бойових дій.
- Управління Станами: Логіка переходів між різними станами зброї, які контролюються через змінні стану (`'in_reload'`, `'in_shoot'`), дозволяє плавні та точні переходи між різними діями.

Інтеграція з Системою Керування Персонажем:

- Синхронізація з Рухами Гравця: Код враховує стан гравця (біг, стрільба, перезарядка) для забезпечення неперервної взаємодії зброї та персонажа, що є важливим для створення одночасно реалістичного та інтуїтивно зрозумілого управління.

Заключення:

Цей модуль зброї, що включає в себе складну логіку та глибоку інтеграцію з різними аспектами гри, створює солідну основу для розширення та адаптації під різноманітні ігрові сценарії. Завдяки гнучкості та масштабованості, код може бути легко адаптований під нові види зброї та геймплейні механіки, забезпечуючи розробникам потужний інструмент для креативності та інновацій у дизайні бою.

3.5.4 Реалізація Штучного Інтелекту для Ворожих Сутностей

Патрулювання:

- Ініціалізація: При входженні в стан патрулювання, AI через `NavMeshAgent` отримує вказівку переміщуватися до визначеної точки на карті. Всі точки патрулювання марковані тегом "Points", що дозволяє легко зібрати їх у список та використовувати для навігації.

- Переміщення: Використовуючи `NavMeshAgent`, AI обирає наступну точку патрулювання випадковим чином зі списку, що забезпечує варіативність поведінки та ускладнює передбачення дій ворога гравцем.

Погоня: Активація Погоні: Якщо гравець потрапляє у поле зору ворога, що вимірюється в радіусі змінної `obzor`, стан AI перемикається на погоню, активуючи тим самим агресивнішу поведінку.

- Поведінка під час Погоні: У стані погоні, `NavMeshAgent` прискорюється, що робить ворога швидшим та небезпечнішим. Ворог безперервно оновлює свою ціль на позицію гравця, прагнучи зменшити дистанцію для атаки.

Атака:

- Оцінка Дистанції: В стані атаки, ворог постійно оцінює дистанцію до гравця. Якщо гравець віддаляється на відстань більше заданої, ворог виходить із стану атаки.
- Виконання Атаки: Під час атаки, АІ використовує `LookAt` для того, щоб ворог завжди був повернутий до гравця, що забезпечує точність та реалізм виконання атакуючих дій.

Ці скрипти є ключовими компонентами для реалізації поведінки ворожих NPC у грі, дозволяючи створити більш живий та непередбачуваний ігровий світ. Перейдемо тепер до другої частини опису, де більш детально розглянемо кожен з цих станів.

Детальний Розгляд Поведінки Патрулювання:

- Алгоритм Вибору Точки: Після досягнення цільової точки, АІ використовує генератор випадкових чисел для вибору наступної точки патрулювання, що забезпечує непередбачуваність маршруту та додає реалізму до поведінки NPC.
- Реагування на Гравця: У випадку, коли гравець наближається на відстань менше ніж `obzor`, АІ відразу переходить у стан погоні, залишаючи патрулювання. Це дозволяє швидко реагувати на зміни в ігровому середовищі та збільшує інтерактивність гри.

Особливості Погоні та Атаки:

- Стратегія Погоні: Під час погоні, АІ збільшує свою швидкість та безперервно оновлює місцезнаходження гравця як свою цільову точку, створюючи тиск та виклик для гравця.
- Транзиція до Атаки: Якщо ворог наближається достатньо близько до гравця, він активує стан атаки. Це забезпечує плавний перехід між різними видами поведінки, підвищуючи іммерсивність геймплею.
- Контроль Дистанції: У стані атаки, ворог використовує контроль дистанції для підтримання оптимальної близькості до гравця, забезпечуючи, що атаки є ефективними та небезпечними.

Заклучення:

Реалізація цих скриптів для патрулювання, погоні та атаки вносить значний вклад у створення інтелектуального та відповідного на присутність гравця штучного інтелекту ворогів. Завдяки цим механікам, вороги здатні адаптуватися до дій гравця, створюючи динамічне та захоплююче ігрове середовище, яке постійно викликає гравця до зміни тактик та стратегій виживання.

Заклучні Моменти:

Ці механізми штучного інтелекту надають NPC глибокого рівня автономії та адаптивності, роблячи кожную зустріч з ворогом унікальною та запам'ятовуваною. Використання станових машин для управління поведінкою ворогів дозволяє розробникам гнучко налаштовувати та вдосконалювати ігровий досвід, відповідаючи на змінні стратегії та стилі гри різних гравців.

3.5.5 Автоматизоване Спавнення Ворогів та Адаптивне Регулювання

Основи Автоматизованого Спавнення:

- Динамічна Система Спавнення: Центральна частина коду відповідає за ініціацію ворогів у грі. Спавнення ворогів базується на різних точках, які задаються в ігровому світі, забезпечуючи варіативність та непередбачуваність геймплею.
- Контрольований Спавн: Система використовує затримку між створенням ворогів, що дозволяє уникнути перевантаження гравця та підтримує оптимальний рівень складності.

Адаптація ворогів:

- Регулювання за Часом: Код реагує на час, проведений гравцем у грі, автоматично коригуючи кількість ворогів, щоб збільшити інтенсивність зі зростанням геймплейного досвіду.
- Аналітика Балансу: Система аналізує взаємозв'язок між часом гри та досягненнями гравця (наприклад, кількістю набраних очок), адаптуючи складність відповідно до успішності гравця.

Інтеграція Машинного Навчання:

- Самовдосконалення через Геймплей Ідея впровадження машинного навчання полягає в тому, щоб зробити процес спавнення не тільки реактивним, але й прогностичним. Система могла б аналізувати попередні ігрові сесії, визначати тенденції та патерни в поведінці гравців і відповідно до цього адаптувати складність.
- Автоматичне Балансування: Використання алгоритмів машинного навчання дозволить системі самостійно калібрувати кількість та типи ворогів, створюючи оптимальні умови для кожного рівня гравця.

Ця частина коду лягає в основу адаптивного геймплею, де кожен сеанс гри унікальний і зберігає виклик, адаптуючись до здібностей та стратегії гравця. Інтеграція машинного навчання здатна значно підвищити цю адаптацію, роблячи ворогів не тільки реактивними, але й передбачувальними, створюючи більш глибокий та захоплюючий ігровий процес.

Процес Спавнення з Перспективи Розробки:

- Масштабування Загрози: Реалізація функції `ScaleEnemyCounts` є ключовим елементом для збільшення та зменшення інтенсивності атак ворогів. Це масштабування виконується через множення поточного числа ворогів на певний коефіцієнт, залежно від тривалості гри та виконання задач гравцем.

- Балансування в Часі: Функція `CheckAndAdjustEnemyCounts` запускається періодично, щоб перевіряти ігрову статистику та відповідно коригувати складність. Це гарантує, що гравець стикається з постійно змінним ігровим досвідом, що змушує його адаптуватися та розвивати свої навички.

Автоматизація і Машинне Навчання:

- Автоматичне Вдосконалення: Завдяки машинному навчанню, система здатна самовдосконалюватися на основі поведінки гравця, покращуючи алгоритми спавнення.
- Прогнозування та Реакція: Алгоритми можуть передбачати, коли гравцеві може знадобитися більше викликів або коли йому потрібно знизити темп. Такий підхід забезпечує динамічний баланс, який постійно підтримує інтерес та захоплення гравця.

Інтерфейс та Зворотній Зв'язок:

- Візуалізація Прогресу: Використання текстових елементів UI, як `txtt`, дозволяє гравцям бачити результати своїх дій в реальному часі, забезпечуючи відчуття досягнень та прогресу.
- Зворотний Зв'язок для Оптимізації: Збір даних про ефективність гравця дозволяє розробникам аналізувати та оптимізувати геймплей, вдосконалюючи ігровий досвід.

Майбутнє Розвитку:

- Інтеграція Глибокого Навчання: Розвиток системи може включати глибше інтеграцію з інструментами машинного навчання, щоб дозволити більш точне та гнучке управління ігровими параметрами відповідно до стилів гри різних гравців.
- Персоналізований Ігровий Досвід: В кінцевому підсумку, метою є створення унікального ігрового досвіду для кожного гравця, де ігрова реальність адаптується до особистих навичок та вподобань.

Розробка такої комплексної системи вимагає не лише глибокого розуміння програмування, а й інноваційного підходу до геймдизайну, забезпечуючи, що кожен елемент ігрового світу вносить свій вклад у створення глибокого та неповторного досвіду для гравця.

3.6 Вибір інструментів тестування

Стратегія тестування геймплею:

Тестування геймплею є критичним етапом розробки, що вимагає уважного аналізу та оцінки ігрового досвіду. Розроблена стратегія включає декілька рівнів перевірки:

- Ручне тестування під час розробки: Розробники регулярно проводять ручні тести нових функцій та механік у процесі кодування, що дозволяє швидко виявити та усунути очевидні помилки та недоліки.
- Сесії плейтестів: Для оцінки загального балансу та інтуїтивності геймплею проводяться планові сесії плейтестів з участю випадкових гравців. Ці сесії дозволяють отримати зовнішній погляд на ігровий процес та виявити неочікувані проблеми, на які розробники могли не звернути увагу.
- Аналіз поведінки гравців: Збір та аналіз даних про поведінку гравців допомагає розуміти, як користувачі взаємодіють з ігровим світом, що в свою чергу надає інформацію для подальшого балансування.

Останній етап тестування фокусується на загальній якості та готовності продукту до релізу:

- Beta-тестування: Проведення закритого або відкритого beta-тестування, залучаючи реальних користувачів з ігрової спільноти. Це дозволяє оцінити гру в умовах, найближчих до реального використання, та зібрати важливі дані про баги, баланс геймплею та оптимізацію процесу.
- Інтеграція з аналітичними системами: Встановлення інтеграції з аналітичними платформами, такими як Google Analytics або Unity Analytics, для збору даних про поведінку гравців, їхні успіхи та

проблеми в грі. Це дозволяє розробникам бачити велику картину та приймати обгрунтовані рішення щодо подальшої оптимізації геймплею.

- Аудит коду та безпеки: Здійснення регулярного аудиту коду на предмет вразливостей та помилок, які можуть вплинути на безпеку ігрових даних або стабільність гри. Використання інструментів статичного аналізу коду та безпеки для автоматичного виявлення потенційних проблем.

Заключне тестування та оптимізація:

Завершальний етап перед випуском гри включає детальну перевірку та фіналізацію:

- Полірування геймплею: Після завершення основного циклу розробки та тестування, команда зосереджується на поліпшенні вражень від гри, виправленні незначних помилок та оптимізації інтерфейсу користувача.
- Оптимізація продуктивності: Виконується детальний аналіз продуктивності, з метою забезпечення плавної роботи гри на різноманітному апаратному забезпеченні. Це включає оптимізацію ресурсів, зменшення часу завантаження та підвищення загальної стабільності.
- Контроль якості на фінальному етапі: Непосередно перед релізом проводиться остаточний контроль якості, щоб переконатися, що всі елементи гри працюють належним чином та відповідають високим стандартам.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ:

4.1 Аналіз ігрової статистики та поведінки гравця

Огляд функцій системи для збору та аналізу ігрової статистики:

Ігрова система була спроектована таким чином, щоб збирати та аналізувати різноманітну статистику в ході ігрової сесії. Це включає дані про точність стрільби гравця, частоту використання певних видів зброї, вибір тактик в бою, а також способи взаємодії з ігровим середовищем. Система реєструє кожен важливу подію в ігровому сеансі, що дозволяє в подальшому аналізувати поведінку гравця та адаптувати ігровий процес для підвищення його зацікавленості та задоволення від гри.

Методи визначення ключових показників успіху гравця:

Ключові показники успіху (KPI) для гравця були визначені на основі таких параметрів як: загальна кількість вбитих ворогів, час виживання, та інші специфічні досягнення. Ці показники дозволяють оцінити ефективність гравця та його прогрес у віртуальному світі. За допомогою машинного навчання система може виявляти закономірності у поведінці гравців, що дозволяє прогнозувати їх подальші дії та оптимізувати ігрові сценарії.

Візуалізація результатів ігрової сесії:

По завершенню кожної ігрової сесії система надає гравцю деталізовану візуалізацію результатів. Гравці можуть побачити свій прогрес, порівняти його з попередніми сеансами, та оцінити ефективність своїх дій. Така візуалізація допомагає гравцям зрозуміти свої сильні сторони та аспекти гри, які потрібно покращити.

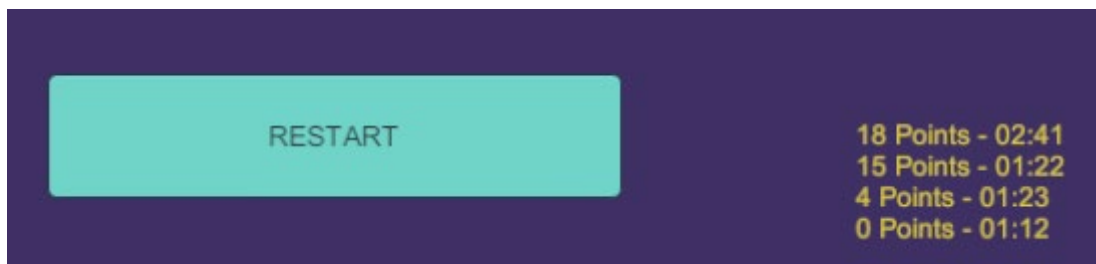


Рис 4.1 Меню закінчення гри та результати попередніх ігор

4.2 Оцінка адаптивності ігрового процесу

4.2.1 Методики тестування рівнів складності та адаптації геймплею

Для забезпечення адаптивності ігрового процесу були використані різні методики тестування, які включали аналіз ігрових сесій з різними рівнями складності. Використовуючи A/B тестування та аналітику зібраних даних, система оцінює реакції гравців на зміни у геймплеї, зокрема на різні ворожі патерни, інтенсивність бою, та доступні ресурси. Ці дані допомагають зрозуміти, як гравці адаптуються до змін та які рівні складності є оптимальними для різних типів гравців.

4.2.2 Аналіз зворотного зв'язку від системи до гравця

Аналізуючи гравецькі дані, система може визначити, коли гравець зіштовхується з труднощами та надати рекомендації або змінити рівень складності в реальному часі. Це допомагає створити більш гнучкий та залучаючий ігровий досвід.

4.2.3 Реакція гри на різні стилі гри та поведінку гравця

Ігрова система була налаштована таким чином, щоб розпізнавати та адаптуватися до різних стилів гри. Завдяки використанню машинного навчання та штучного інтелекту, система аналізує поведінку гравця та вносить зміни в ігровий світ, забезпечуючи оптимальний рівень викликів та заохочення для кожного індивідуального гравця.

Основні реакції:

- При зменшенні здоров'я у Головного персонажа, гра трохи полегшує геймплей (Менше ворогів, менше сили у ворогів).
- При високій швидкості знешкодження ворогів, гра ускладнюється, але є більше можливості знайти нову зброю або обвіси на неї.
- Якщо тримати баланс між атакою та захистом збільшується шанс на більш довгу та результативну гру.

4.3 Результати тестування ключових механік ігрової системи

4.3.1 Тестування системи управління персонажем та бойової системи

У ході розробки ігрова система пройшла ретельне тестування з метою забезпечення інтуїтивно зрозумілого та відгукового управління персонажем. Бойова система була випробувана на збалансованість, відчуття удару та реалістичність фізики. Тестування проводилось з використанням як автоматизованих тестів, так і ручного тестування за участю живих гравців, щоб оцінити різні аспекти, включаючи реакцію на вводи користувача, точність ціління та задоволеність від геймплею.

4.3.2 Аналіз ефективності системи спавну ворогів та їх інтелекту

Система спавну ворогів була розроблена таким чином, щоб створювати виклики, які адаптуються до рівня навичок та стилю гри кожного гравця. Інтелект НПС був оптимізований для реалістичного та прогнозованого поведінки, здатного реагувати на дії гравця. В ході тестування аналізувалися час реакції НПС, стратегії обходу перешкод та ефективність тактик бою.

4.3.3 Тестування системи патрулювання та відслідковування гравця НПС

Система патрулювання та відслідковування була тестована на здатність НПС виявляти гравця в різних ігрових сценаріях. Тестування включало оцінку правильності маршрутів патрулювання, здатності НПС переходити від одного стану поведінки до іншого та їхню реакцію на втрату зорового контакту з гравцем. Результати тестування вказують на високу адаптивність НПС до змін у поведінці гравця та ефективність їхнього відслідковування.



Рис 4.3 Реакція мобів на героя, після зорового контакту



Рис 4.3.1 Патрулювання без зорового контакту

4.4 Особливості створенної системи

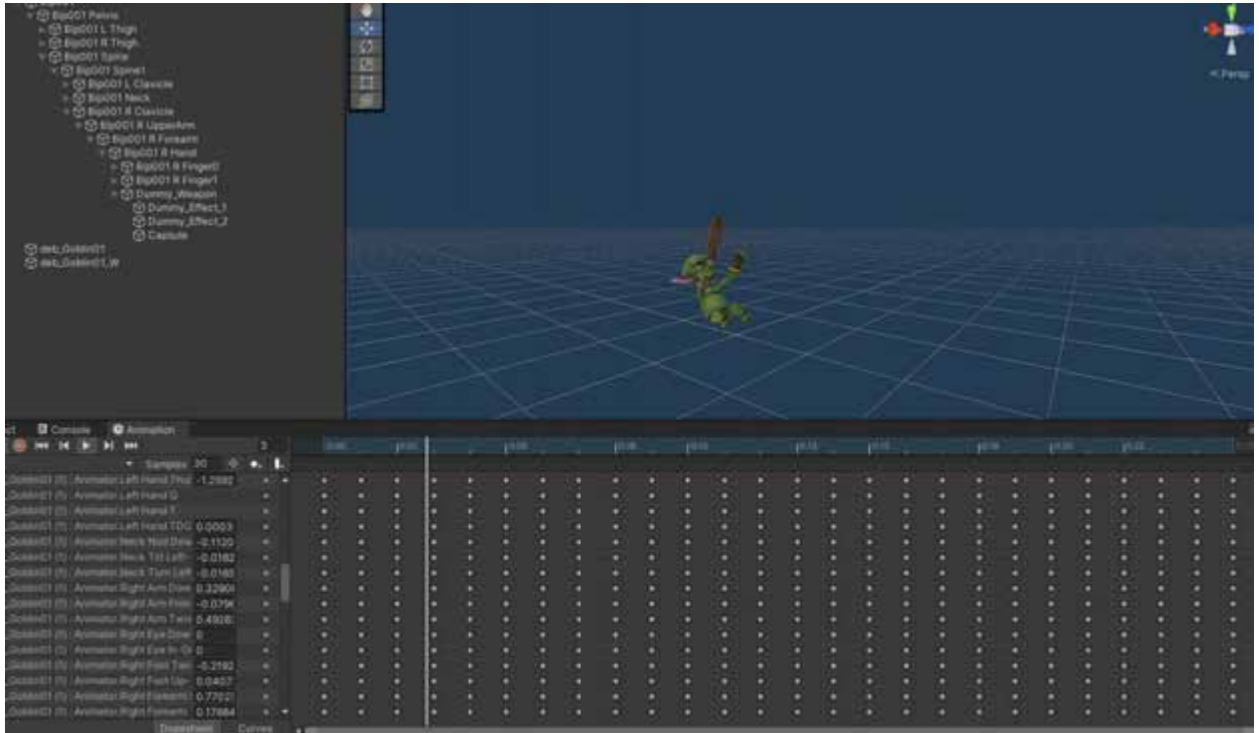


Рис 4.4.1 ілюстрація з демонстрацією анімації смерті моба

Анімація зброї та звукові ефекти:

- Кожна зброя має деталізовані анімації стрільби, перезаряджання, та вибору, що доповнюються відповідними звуковими ефектами для створення більш емоційного і занурювального досвіду користувача.

- Процес інтеграції звуку в зброю включає налаштування аудіо сорсів та імітаторів ефекту стрільби, що забезпечують синхронізацію звуку із візуальними діями.

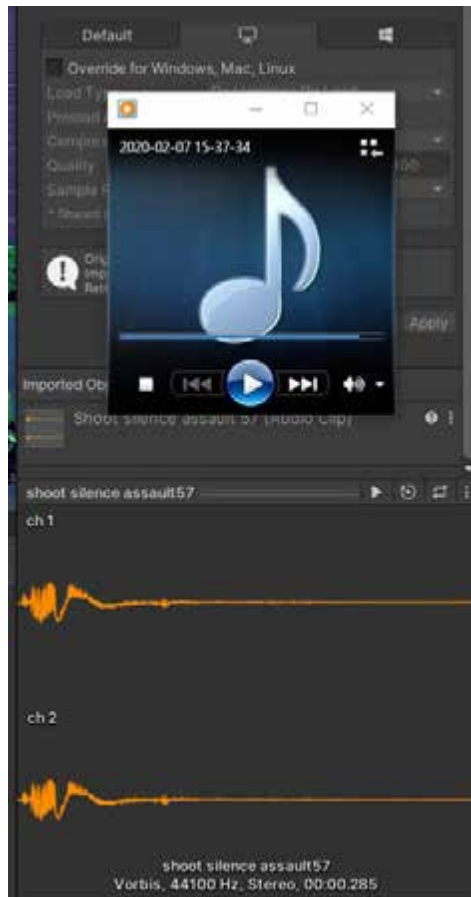


Рис. 4.4.2 Звуковий ефект стрільби

Цей підхід до аудіовізуальної інтеграції дозволяє створити ігровий простір, де кожен елемент має свій звук та рух, що сприяє глибшому зануренню в ігрове середовище та створенню враження "живого" світу.

4.4.2 Комплексна структура скриптів

Блок-схема взаємодії скриптів у ігровій системі:

1) Код на спавн мобів:

- Активує "Код на патрулювання для мобів", якщо моб не виявив гравця.
- Активує "Код на переслідування для мобів", якщо моб помітив гравця.

2) Код на патрулювання для мобів:

- Передає управління "Коду на переслідування для мобів", якщо гравець виявлений.

3) Код на переслідування для мобів:

- Активує "Код на атаку для мобів", якщо моб досить близько до гравця.

- Повертається до "Коду на патрулювання для мобів", якщо гравець загублений з поля зору.

4) Код на атаку для мобів:

- Викликає "Дати Урон", щоб завдати урону гравцеві при успішній атаці.

5) Дати Урон:

- Передає дані про урон "Коду на головного героя" або іншим мобам.

6) Урон прийняти:

- Отримує дані від "Дати Урон" і змінює параметри здоров'я.

7) Код на головного героя:

- Керує взаємодією гравця з ігровим світом та реагує на отриманий урон.

- Використовує "Код на зброю" для атаки мобів.

8) Код на зброю:

- Генерує кулі під час стрільби, використовуючи "Код на створення кулі та її фізику".

- Активує "Дати Урон" при попаданні в моба.

9) Код на створення кулі та її фізику:

- Керує рухом куль та їх взаємодією з об'єктами ігрового світу.

10) Код на спавн предметів (бонусів):

- Створює предмети у ігровому світі, які гравець може використовувати для відновлення здоров'я або отримання бонусів.

4.4.3 Інноваційний підхід до системи аналізу

У цьому підрозділі акцентується увага на нестандартному підході до розробки системи аналізу даних, який був прийнятий у рамках розробки ігрової системи. Ця система була створена з урахуванням потреби в адаптації до різноманітних стилів гри, забезпечуючи унікальний геймплейний досвід для кожного користувача.

Нетрадиційні методи аналізу:

- Система аналізу була розроблена з нуля, з використанням фундаментальних принципів штучного інтелекту для створення гнучкої та ефективної моделі поведінки НПС.

- Відмова від використання готових рішень AI дозволила кастомізувати алгоритми під конкретні завдання та цілі ігрового процесу.

Адаптація до стилів гри:

- Розроблені алгоритми здатні аналізувати поведінку гравців в реальному часі та адаптувати рівень складності НПС, що робить ігровий процес цікавішим та більш динамічним.

- Система аналізу враховує різні підходи гравців до проходження ігрових рівнів, від стратегічного підкрадання до агресивного штурму, і змінює поведінку НПС відповідно до цього.

Переваги інноваційної системи:

- Гнучкість у налаштуванні параметрів НПС без потреби в переписуванні великих блоків коду.

- Здатність системи швидко вчитися та адаптуватися до непередбачених стратегій гравців, завдяки чому кожна гра є унікальною.

- Зниження витрат на обчислення порівняно з традиційними методами AI, що дозволяє покращити продуктивність ігрової системи на широкому спектрі апаратного забезпечення.

Імплементация та тестування:

- Ретельне тестування було здійснене з метою відшліфувати алгоритми реакції НПС на дії гравців, забезпечуючи при цьому баланс та справедливість геймплею.

- Аналіз даних, отриманих від тестових гравців, використовувався для постійного вдосконалення системи, що підтримує високий рівень інтерактивності та занурення у ігровий світ.



Рис.4.4.1 Гоблін при наближенні використовує анімацію атаки

4.5 Перспективи розвитку та вдосконалення ігрової системи

4.5.1 Можливі шляхи розширення ігрових можливостей та геймплею

Ігрова система пропонує солідний фундамент для подальшого розширення ігрових можливостей. Це може включати додавання нових рівнів, персонажів, предметів інвентарю та бойових навичок, які збагатять ігровий досвід та збільшать повторну грабельність. Також можливе створення відкритого світу для дослідження гравцями та додавання елементів сюжету, які поглиблюють іммерсивність геймплею.

Інтеграція додаткових аналітичних інструментів та механізмів машинного навчання

Ігрова система може бути покращена за допомогою інтеграції передових аналітичних інструментів та алгоритмів машинного навчання для збору глибших інсайтів про поведінку гравців. Це дозволить створити більш персоналізовані та адаптивні ігрові сценарії, які відреагують на стиль гри

кожного гравця, підвищуючи таким чином задоволеність та залученість користувачів.

4.5.2 Впровадження нових функцій на основі аналізу даних

Опираючись на аналіз ігрових даних, розробники можуть виявити і впровадити нові функції, які відповідатимуть конкретним вподобанням і потребам гравців. Це може охоплювати покращення існуючих механік, введення нових способів взаємодії в ігровому світі, та більш гнучкі налаштування геймплею для забезпечення відчуття новизни та виклику

Усі ці напрямки розвитку та вдосконалення вимагатимуть постійного тестування та аналізу даних, щоб забезпечити, що нововведення покращують ігровий досвід та зберігають баланс гри.

ВИСНОВОК

У цій магістерській роботі було розглянуто комплексний підхід до розробки інноваційної ігрової системи, яка включає в себе глибокий аналіз ігрової статистики, поведінки гравця, адаптивності ігрового процесу, а також детальну проробку ключових ігрових механік. Дослідження виявило важливість кожного з цих елементів у створенні цілісного та занурюючого геймплею.

Значення дослідження:

Розробка ігрової системи з акцентом на деталізацію та індивідуальний підхід до кожного гравця має велике значення для ігрової індустрії. Наша робота відображає потенціал адаптивних систем у створенні унікальних ігрових досвідів та підвищенні залученості гравців.

Сила роботи:

Сила цієї магістерської роботи полягає у впровадженні та інтеграції комплексних рішень, що об'єднують технічну складність та творчу інновацію. Це включає розробку власних методів аналізу поведінки гравців, створення гнучких ігрових механік та вдосконалення аудіовізуального супроводу.

Вплив на геймдизайн:

Результати дослідження можуть відіграти важливу роль у подальшому розвитку геймдизайну, пропонуючи нові способи для створення ігор, які би відповідали індивідуальним перевагам і стилям гри широкого кола гравців. Індивідуальний аналіз та адаптація геймплею може зміцнити взаємодію між гравцем та ігровим світом, роблячи кожную ігрову сесію неповторною.

Перспективи розвитку:

Проект демонструє великі перспективи для розвитку, включаючи інтеграцію більш складних систем штучного інтелекту, розширення ігрових можливостей та покращення інтерактивності ігрового світу. Забезпечення балансу між технічною оптимізацією та ігровими нововведеннями залишається ключовим пріоритетом для подальшої роботи.

Заключна частина цієї магістерської роботи не лише піддає сумніву традиційні методики в ігровому дизайні, але й відкриває нові перспективи для інновацій, що здатні перетворити розробку ігор. Вона демонструє, як оригінальний підхід може збагатити ігровий досвід, вносячи свіжі ідеї та рішення, які можуть змінити індустрію ігор. Ця робота виступає як промінь світла, що вказує на необхідність відходу від звичних парадигм та відкриття нових горизонтів для творчості, де кожен елемент геймплею може бути витончено та вдосконалено для досягнення найвищої залученості гравців. Усе це веде до розвитку галузі, де інновації не лише вітаються, але й стають новим стандартом для наступних поколінь розробників ігор, які прагнуть створювати значущі та запам'ятовуючі ігрові світи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. "Artificial Intelligence and Games" - Georgios N. Yannakakis, Julian Togelius, 2018, Springer, ISBN 978-3-319-63519-4, [springer.com](https://www.springer.com/).
2. "The Art of Game Design: A Book of Lenses" - Jesse Schell, 2014 (2nd Edition), CRC Press, ISBN 978-1466598645, [crcpress.com](https://www.crcpress.com/).
3. "Level Up! The Guide to Great Video Game Design" - Scott Rogers, 2014, Wiley, ISBN 978-1118877166, [wiley.com](https://www.wiley.com/).
4. "Rules of Play: Game Design Fundamentals" - Katie Salen Tekinbaş, Eric Zimmerman, 2003, MIT Press, ISBN 978-0262240451, mitpress.mit.edu.
5. "Game Programming Patterns" - Robert Nystrom, 2014, Genever Benning, ISBN 978-0990582908, gameprogrammingpatterns.com.
6. "Unity in Action: Multiplatform Game Development in C# with Unity 5" - Joe Hocking, 2015, Manning Publications, ISBN 978-1617292323, [manning.com](https://www.manning.com/).
7. "Unreal Engine 4 Game Development in 24 Hours, Sams Teach Yourself" - Aram Cookson, Ryan DowlingSoka, Clinton Crumpler, 2016, Sams Publishing, ISBN 978-0672337628, [pearson.com](https://www.pearson.com/).
8. "Virtual Reality for Game Developers" - Michelle Menard, 2019, CRC Press, ISBN 978-0367191584, [crcpress.com](https://www.crcpress.com/).
9. "Data Analytics for Game Development" - Gregor Schiele, 2020, CRC Press, ISBN 978-0367896222, [crcpress.com](https://www.crcpress.com/).
10. "Audio for Games: Planning, Process, and Production" - Alexander Brandon, 2005, New Riders, ISBN 978-0735714137, [peachpit.com](https://www.peachpit.com/).
11. "The Psychology of Video Games" - Jamie Madigan, 2015, ISBN 978-1498705697, [psychologyofgames.com](http://www.psychologyofgames.com/).
12. "Game AI Pro: Collected Wisdom of Game AI Professionals" - Steve Rabin, 2013, CRC Press, ISBN 978-1466565961, [crcpress.com](https://www.crcpress.com/).
13. "Players Making Decisions: Game Design Essentials and the Art of Understanding Your Players" - Zack Hiwiler, 2016, New Riders, ISBN 978-0134396750, [peachpit.com](https://www.peachpit.com/).

14. "Game Mechanics: Advanced Game Design" - Ernest Adams, Joris Dormans, 2012, New Riders, ISBN 978-0321820273, [peachpit.com](https://www.peachpit.com/).
15. "AI for Games, Third Edition" - Ian Millington, 2019, CRC Press, ISBN 978-0367009298, [crcpress.com](https://www.crcpress.com/).
16. "The Ultimate Guide to Video Game Writing and Design" - Flint Dille, John Zuur Platten, 2008, Lone Eagle, ISBN 978-1580650663, [penguinrandomhouse.com](https://www.penguinrandomhouse.com/).
17. "Building Virtual Reality with Unity and Steam VR" - Jeff W. Murray, 2017, CRC Press, ISBN 978-1498785006, [crcpress.com](https://www.crcpress.com/).
18. "Unity 2020 By Example: A project-based guide to building 2D, 3D, augmented reality, and virtual reality games" - Robert Wells, 2020, Packt Publishing, ISBN 978-1838984717, [packtpub.com](https://www.packtpub.com/).
19. "Understanding Video Games: The Essential Introduction" - Simon Egenfeldt-Nielsen, Jonas Heide Smith, Susana Pajares Tosca, 2019, Routledge, ISBN 978-1138325441, [routledge.com](https://www.routledge.com/).
20. "Game Feel: A Game Designer's Guide to Virtual Sensation" - Steve Swink, 2008, Morgan Kaufmann, ISBN 978-0123743282, [elsevier.com](https://www.elsevier.com/).
21. "Games, Design and Play: A detailed approach to iterative game design" - Colleen Macklin, John Sharp, 2016, Addison-Wesley Professional, ISBN 978-0134392073, [pearson.com](https://www.pearson.com/).
22. "Game Design Workshop: A Playcentric Approach to Creating Innovative Games" - Tracy Fullerton, 2014, AK Peters/CRC Press, ISBN 978-1482217162, [crcpress.com](https://www.crcpress.com/).
23. "The Gamer's Brain: How Neuroscience and UX Can Impact Video Game Design" - Celia Hodent, 2017, CRC Press, ISBN 978-1498775502, [crcpress.com](https://www.crcpress.com/).
24. "Games as Service: How Free to Play Design Can Make Better Games" - Oscar Clark, 2014, Routledge, ISBN 978-0415707137, [routledge.com](https://www.routledge.com/).
25. "Serious Games: Games That Educate, Train, and Inform" - Michael R. Macedonia, 2005, Course Technology PTR, ISBN 978-1592006229, [cengage.com](https://www.cengage.com/).
26. "Game Development Essentials: An Introduction" - Jeannie Novak, 2011, Cengage Learning, ISBN 978-1111307653, [cengage.com](https://www.cengage.com/).