

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри

« ____ » _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Ігровий застосунок в жанрі ролевого екшену на основі
рушія Unreal Engine 4

Спеціальність 121 «Інженерія програмного забезпечення»

121 – КРБ – 408.21910826

Виконав студент 4-го курсу, групи 408 _____ **В. О. Швець**
« ____ » червня 2023 р.

Керівник канд. техн. наук, викладач _____ **К. Ю. Березний**
« ____ » червня 2023 р.

Консультант канд. техн. наук, доцент _____ **А. О. Алексєєва**
« ____ » червня 2023 р.

Миколаїв – 2023

**Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення**

ЗАТВЕРДЖУЮ

Завідувач кафедри інженерії програмного
забезпечення, канд.техн.наук, доцент,

_____ Є. О. Давиденко

« ____ » _____ 2023 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи бакалавра

Видано студенту групи 408 факультету комп'ютерних наук

_____ Швець Владислав Олександрович

(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи

«Ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine»

Затверджена наказом по ЧНУ від «17» березня 2023 р. № 60

2. Строк представлення кваліфікаційної роботи « 23» червня 2023 р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні

Вхідні дані до роботи – функціональні та нефункціональні вимоги до ігрового застосунку. Результат – функціонуючий ігровий застосунок

4. Перелік питань, що підлягають розробці

Порвести аналіз сучасних автоматизованих ресурсів для створення ігрового застосунку; специфікація вимог до ігрового застосунку; проектування та моделювання ігрового застосунку; розробка функціональних модулів ПЗ та проектування зручного та інтуїтивно зрозумілого інтерфейсу та опцій, який дозволить споживачам легко розібратися в управлінні та проходженні ігрового застосунку; тестування та апробація ПЗ.

5. Перелік графічних матеріалів:

Презентація_____.

6. Завдання до спеціальної частини

Дослідження питань охорони праці, які безпосередньо пов'язані з діяльністю розробника програмного забезпечення_____.

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
Алексєєва А.О., канд. техн. наук, доцент	Кафедра екології Медичного інституту ЧНУ ім. Петра Могили	Спеціальна частина з охорони праці

Керівник роботи _____ канд. техн. наук К. Ю. Бережний
(посада, прізвище, ім'я, по батькові)

(підпис)

Завдання прийнято до виконання

_____ Швець Владислав Олександрович
(прізвище, ім'я, по батькові студента)

(підпис)

Дата видачі завдання «___» _____ 2023 р.

КАЛЕНДАРНИЙ ПЛАН**виконання кваліфікаційної роботи**

Тема: «Ігровий застосунок в жанрі ролевого екшену на основі рушія
Unreal Engine»

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Розробка та затвердження завдання на виконання КРБ	04.10.2022	10.10.2022	Виконано
2.	Огляд літератури за темою роботи	11.10.2022	17.10.2022	Виконано
3.	Складання календарного плану КРБ	18.10.2022	24.10.2022	Виконано
4.	Аналіз предметної області	17.01.2022	30.01.2022	Виконано
5.	Розробка проєктних рішень	31.01.2023	23.02.2023	Виконано
6.	Моделювання та конструювання ПЗ	11.03.2022	17.03.2023	Виконано
7.	Кодування, тестування та апробація розробленого ПЗ, аналіз результатів тестування, розробка керівництва користувача	18.03.2023	08.05.2023	Виконано
8.	Розробка спеціальної частини з охорони праці	09.05.2023	15.05.2023	Виконано
9.	Відгук керівника КРБ	16.04.2023	17.04.2023	Виконано
10.	Оформлення КРБ та презентації	18.05.2023	22.05.2023	Виконано
11.	Попередній захист	06.06.2023	07.06.2023	Виконано
12.	Рецензування	10.06.2023	18.06.2023	Виконано
13.	Завершення оформлення КРБ та презентації	20.06.2023	22.06.2023	Виконано
14.	Захист кваліфікаційної роботи	26.06.2023	28.06.2023	Виконано

Виконав студент 4-го курсу, групи 408 _____ В. О. Швець
«__» червня 2023 р.

Керівник канд. техн. наук, викладач _____ К. Ю. Бережний
«__» червня 2023 р.

АНОТАЦІЯ

до кваліфікаційної роботи бакалавра

Ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine 4

Студент 408 гр.: Швець Владислав Олександрович

Керівник: канд. техн. наук. Бережний Кирило Юрійович

Unreal Engine 4 – це потужний ігровий рушій, який використовується розробниками для швидкого та легкого створення захопливих ігор. Він пропонує розширені можливості, такі як рендеринг графіки, фізичне моделювання та ігровий штучний інтелект, а також візуальну систему сценаріїв для непрограмістів. Маркетплейс та спільнота рушія надають доступ до ресурсів та підтримки, що робить його актуальним та популярним вибором для розробки ігор.

Об'єктом кваліфікаційної роботи є процеси пов'язані із організацією та створенням ігрового застосунку.

Предметом кваліфікаційної роботи є інструментальні засоби та інформаційні технології розробки ігрового застосунку.

Метою кваліфікаційної роботи є дослідження та вдосконалення напрямку розробки та впровадження комп'ютерних ігрових технологій в жанрі ролевого екшену.

Для досягнення поставленої мети були виконані завдання:

1. Проведено аналіз сучасних автоматизованих ресурсів для створення ігрового застосунку. Визначено переваги та недоліки даної платформи.
2. Проведено специфікацію вимог до ігрового застосунку.
3. Проведено проектування та моделювання ігрового застосунку.
4. Розроблено функціонал модулів ПЗ та проектування зручного та інтуїтивно зрозумілого інтерфейсу та опцій, який дозволить споживачам легко розібратися в управлінні та проходженні ігрового застосунку.
5. Проведено тестування ПЗ.

У першому розділі КРБ проведено аналіз автоматизованих ресурсів для створення ігрових додатків, було розглянуто переваги та недоліки платформи Unreal Engine. Визначено функціональні та нефункціональні вимоги.

У другому розділі виконано моделювання програмно забезпечення ігрового застосунку.

Третій розділ присвячено питанням проектування ігрового застосунку. Було спроектовано ігрові механіки та геймплей, ігрові активи та дизайн середовища, технічні архітектури ігрового додатку.

Четвертій розділ присвячено реалізації програмних модулів та їх функціональність, розробка, наведена інструкція користувача, елементи керування на навігації.

Результатом КРБ є функціональна гра від третьої особи, з відкритим зовнішнім світом, з лінійним сюжетом та різним функціоналом.

КРБ викладена на 76 сторінки, вона містить 4 розділи, 49 ілюстрацій, 10 таблиць, 24 джерел в переліку посилань.

Ключові слова: *Розробка ігор, рушій Unreal Engine, 3D-гра, ігрова механіка, ігрові активи, ігрова логіка, тестування продуктивності, візуалізація, забезпечення якості.*

ABSTRACT

of the Bachelor's Thesis

"A role-playing action game application based on the Unreal Engine 4"

Student of 408 group: Vladyslav Shvets

Supervisor: Ph. D. Berezhnyi K. Yu.

Unreal Engine 4 is a powerful game engine used by developers to quickly and easily create immersive games. It offers advanced features such as graphics rendering, physical modeling, and game artificial intelligence, as well as a visual scripting system for non-programmers. The engine's marketplace and community provide access to resources and support, making it a relevant and popular choice for game development.

The object of qualification work is the processes associated with the organization and creation of a game application.

The subject of the qualification work is the tools and information technologies for developing a game application.

The purpose of the qualification work is to study and improve the direction of development and implementation of computer game technologies in the genre of role-playing action.

To achieve this goal, the following tasks were performed:

1. The analysis of modern automated resources for creating a game application is carried out. The advantages and disadvantages of this platform are determined.
2. The specification of requirements for the game application is carried out.
3. The design and modeling of the game application was carried out.
4. The functionality of software modules and the design of a user-friendly and intuitive interface and options that will allow consumers to easily understand the management and passage of the gaming application.
5. Software testing was carried out.

The first section of the BT analyzes automated resources for creating gaming applications, and considers the advantages and disadvantages of the Unreal Engine platform.

The second section describes the gaming application, functional and non-functional requirements.

The third section includes design and modeling. It includes game mechanics and gameplay, game assets and environment design, and technical architecture of the game application.

The fourth chapter covered the implementation of program modules and their functionality, the development and implementation of the user interface, and navigation controls.

In the fifth section, some types of testing, approbation, and final evaluation are carried out.

The last section concludes and reflects on the development process.

The result was a functional third-person game with an open world, a linear storyline, and a variety of features.

The BT is 76 pages long and contains 4 sections, 49 illustrations, 10 tables, 24 references.

Keywords: *Game development, Unreal Engine, 3D game, game mechanics, game assets, game logic, performance testing, visualization, quality assurance.*

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	10
ВСТУП	11
1 АНАЛІЗ ПРЕДМЕТНОЇ СФЕРИ РОЗРОБКИ КОМП'ЮТЕРНИХ ІГОР.....	14
1.1 Опис предметної сфери розробки комп'ютерних ігор	14
1.2 Жанри комп'ютерних ігор	15
1.3 Аналіз ігрових застосунків у жанрі ролевого екшену	16
1.4 Специфікація вимог до програмного забезпечення комп'ютерної гри	21
Висновки до розділу 1	24
2 МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ ГРИ.....	25
2.1 Діаграма прецедентів	25
2.2 Діаграма варіантів використання	26
2.3 Опис дійових осіб	27
2.2 Діаграма станів	31
Висновки до розділу 2	34
3 ПРОЄКТУВАННЯ ІГРОВОГО ЗАСТОСУНКУ	36
3.1 Опис ігрового циклу	36
3.2 Розробка ігрових механік	39
3.3 Проектування ігрової локації	40
3.4 Проектування ігрового інтерфейсу	43
Висновки до розділу 3	45
4 ПРОГРАМНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНОЇ ГРИ.....	47
4.1 Вибір технологій розробки комп'ютерної гри	47
4.2 Діаграма класів програмного забезпечення	50
4.3 Опис функціоналу	51
4.4 Тестування комп'ютерної гри	67
4.5 Інструкції користувача комп'ютерної гри	69
Висновки до 4 розділу	71
ВИСНОВКИ.....	73
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	74

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

VR – Viretual Reality

AR – Augmented Reality

UE4 – Unreal Engine 4

ПЗ – Програмне забезпечення

RTS – Real-Time Strategy

TBS – Turn-Based Strategy

FPS – First-Person Shooter

RPG – Role-Playing Game

MMO – Massively Multiplayer Online

NPC – Non-Player Character

UML – Unified Modeling Language

UI – User Interface

БД – База даних

ВСТУП

Розробка ігрового застосунку в жанрі ролевого екшену на основі рушія Unreal Engine є актуальною темою, тому що, UE4 продовжує набувати все більшого поширення у світі розробки віртуальної реальності (VR) та доповненої реальності (AR). З виходом Unreal Engine 5, Epic Games зробила сильний акцент на розробці для цих імерсивних технологій, з новими можливостями, такими як здатність передавати величезні обсяги даних у реальному часі для підтримки великомасштабних середовищ і динамічного освітлення. Нарешті, Epic Games продовжує забезпечувати потужну спільноту та систему підтримки для розробників UE4. Unreal Marketplace надає широкий вибір ресурсів, плагінів та інструментів, які допомагають розробникам створювати високоякісні ігри більш ефективно, а Unreal Developer Network пропонує ресурси та форуми для співпраці та усунення несправностей [16].

Розробка гри на платформі Unreal Engine 4 має як наукове, так і практичне значення.

Наукове значення:

1) Unreal Engine 4 - це потужний та складний ігровий рушій, який дозволяє створювати якісну та реалістичну 3D графіку, фізику та анімацію.

2) Використання Unreal Engine 4 дозволяє розробникам ігор використовувати новітні технології та методики в розробці ігор, такі як трасування променів в реальному часі та глобальне освітлення, що може призвести до значного прогресу в галузі розробки ігор.

Практичне значення:

1) Unreal Engine 4 є широко використовуваним ігровим рушієм, з великою та активною спільнотою розробників, які обмінюються знаннями та ресурсами [16].

2) Рушій також підтримується широким спектром платформ, включаючи ПК, консолі, мобільні та VR/AR пристрої, що дозволяє розробникам охопити своїми іграми широку аудиторію [20].

3) Рушій також є дуже гнучким і налаштовуваним, розробники можуть адаптувати рушій до своїх конкретних потреб, а також використовувати готові ресурси та креслення, які надає рушій [15].

4) Unreal Engine 4 також пропонує широкий спектр інструментів та функцій, які спрощують процес розробки ігор, таких як візуальна система сценаріїв, вбудована система анімації та потужний фізичний рушій.

5) Це також економічно вигідне рішення для розробки ігор, оскільки рушій доступний безкоштовно. Тому ця тема є актуальною [12].

Цілі та завдання розробки гри на платформі Unreal Engine 4 можуть включати в себе наступні

1) Створення якісної та реалістичної графіки, фізики та анімації: Unreal Engine 4 - це потужний і складний ігровий рушій, який дозволяє створювати якісну і реалістичну 3D-графіку, фізику та анімацію.

2) Охоплення широкої аудиторії: Оскільки Unreal Engine 4 підтримується широким спектром платформ, включаючи ПК, консолі, мобільні пристрої та VR/AR-пристрої, це дозволяє розробникам охопити своїми іграми широку аудиторію.

3) Налаштування гри під конкретні потреби: Unreal Engine 4 дуже гнучкий і налаштовується, розробники можуть адаптувати рушій під свої конкретні потреби, а також використовувати готові ресурси та креслення, які надає рушій.

4) Спрощення процесу розробки гри: Unreal Engine 4 пропонує широкий спектр інструментів та функцій, які спрощують процес розробки ігор, таких як візуальна система сценаріїв, вбудована система анімації та потужний фізичний рушій [14].

5) Економічно ефективне рішення: Unreal Engine 4 є економічно ефективним рішенням для розробки ігор, оскільки рушій доступний безкоштовно з моделлю розподілу доходу, що робить його доступним для невеликих розробників та інді-студій [14].

6) Створюйте захопливий та цікавий досвід: Завдяки новітнім технологіям і методам, які підтримує Unreal Engine 4, таким як трасування променів у реальному часі та глобальне освітлення, розробники можуть створювати захопливий та цікавий ігровий досвід для гравців [14].

7) Досягнення конкретних цілей і вимог проєкту: Розробка гри на Unreal Engine 4 дозволяє розробникам відповідати конкретним цілям і вимогам проєкту, наприклад, створювати гру в певному жанрі, в даному випадку це action/RPG [11].

Об'єктом кваліфікаційної роботи є процеси пов'язані із організацією та створенням ігрового застосунку.

Предметом кваліфікаційної роботи є іструментальні засоби та інформаційні технології розробки ігрового застосунку.

Метою кваліфікаційної роботи є дослідження та вдосконалення напрямку розробки та впровадження комп'ютерних ігрових технологій в жанрі ролевого екшену.

Для досягнення цієї мети необхідно вирішити наступні завдання:

1. Провести аналіз сучасних автоматизованих ресурсів для створення ігрового застосунку. Визначити переваги та недоліки даної платформи.
2. Специфікація вимог до ігрового застосунку.
3. Проєктування та моделювання ігрового застосунку.
4. Розробка функціональних модулів ПЗ та проєктування зручного та інтуїтивно зрозумілого інтерфейсу та опцій, який дозволить споживачам легко розібратися в управлінні та проходженні ігрового застосунку.
5. Тестування та апробація ПЗ.

1 АНАЛІЗ ПРЕДМЕТНОЇ СФЕРИ РОЗРОБКИ КОМП'ЮТЕРНИХ ІГОР

1.1 Опис предметної сфери розробки комп'ютерних ігор

Розробка комп'ютерних ігор у жанрі рольового екшену на основі рушія Unreal Engine – це захоплива предметна область, що поєднує творчість ігрового дизайну з технічними навичками програмування, художньої та анімації, а також аудіовиробництва. Unreal Engine – популярний інструмент розробки ігор, який дозволяє розробникам створювати високоякісні, захоплюючі ігри для різних платформ, включаючи консолі, ПК і мобільні пристрої.

Розробка ігор у жанрі рольового екшену зазвичай передбачає створення складної ігрової механіки, захопливого сюжету та динамічного, інтерактивного світу. Геймдизайнери тісно співпрацюють з програмістами для створення правил і механік гри, таких як кастомізація персонажів, бойові системи та проходження квестів. Вони також співпрацюють з художниками та аніматорами для створення візуально приголомшливих середовищ і персонажів, які допомагають оживити ігровий світ [12].

Unreal Engine – це передовий інструмент для розробки ігор, який надає розробникам доступ до широкого спектру функцій та інструментів, включаючи просунуту фізику, штучний інтелект та системи рендерингу. Програмісти використовують Unreal Engine для створення ігрової механіки та оптимізації продуктивності, а художники та аніматори – для створення візуально приголомшливих ігрових ресурсів та середовищ [12].

Окрім ігрової механіки та арт-активів, важливим аспектом розробки ігор у жанрі рольового екшену є також аудіовиробництво. Звукові ефекти, музика та озвучка допомагають створити багатий та захоплюючий досвід для гравців, посилюючи їхнє відчуття занурення та залучення до гри.

Загалом, розробка ігор у жанрі рольового екшену на основі рушія Unreal Engine – це складна та захоплива предметна область, що вимагає високого рівня технічної майстерності та творчого підходу. Він пропонує розробникам

ігор можливість створювати захоплюючі, інтерактивні світи, які залучають і захоплюють гравців [17].

1.2 Жанри комп'ютерних ігор

Екшн-ігри це динамічні ігри, які вимагають швидких рефлексів та зорово-моторної координації. Гравці зазвичай керують персонажем, який бере участь у бою або інших інтенсивних діях, таких як перегони або платформери.

Пригодницькі ігри часто містять елементи дослідження та розв'язання головоломок, а гравці беруть на себе роль протагоніста, який просувається по сюжету, керованому історією [11].

Рольові ігри (RPG) гравці беруть на себе роль персонажа і контролюють його дії та рішення. РПГ часто вирізняються глибокою кастомізацією персонажів, захопливими світами та складними сюжетними лініями.

Стратегічні ігри ці ігри вимагають від гравців тактичного мислення та планування для досягнення цілей. Прикладами є стратегічні ігри в реальному часі (RTS) та покрокові стратегічні ігри (TBS).

Імітаційні ігри дозволяють гравцям імітувати реальний життєвий досвід або сценарії, наприклад, управління бізнесом, польоти на літаку або управління містом [11].

Спортивні ігри імітують популярні види спорту, такі як футбол, баскетбол і регбі. Гравці керують командою або окремим спортсменом і змагаються з іншими командами або гравцями.

Головоломки вимагають від гравців розв'язання різноманітних головоломок або завдань, часто з обмеженими ресурсами або часовими обмеженнями.

Файтинги – це бої між персонажами один на один, де гравці використовують різні прийоми та комбінації для перемоги над опонентами.

Шутери від першої особи (FPS) гравці керують персонажем, який вступає в бій, використовуючи різноманітну зброю. Ці ігри часто мають багатокористувацькі онлайн-режими [12].

Масові багатокористувацькі онлайн-ігри (ММО) – ігри дозволяють гравцям взаємодіяти один з одним у постійному онлайн-світі. Гравці часто беруть участь у квестах, битвах та інших заходах разом [11].

1.3 Аналіз ігрових застосунків у жанрі ролевого екшену

Приклади аналогічних систем. Для кожного визначено наступні характеристики:

Назви: Elder Scrolls 3, Neverwinter Nights, Anachronox, The Elder Scrolls II: Daggerfall.

На рис. 1.1 – 1.4 наведено іконки ігр [9].

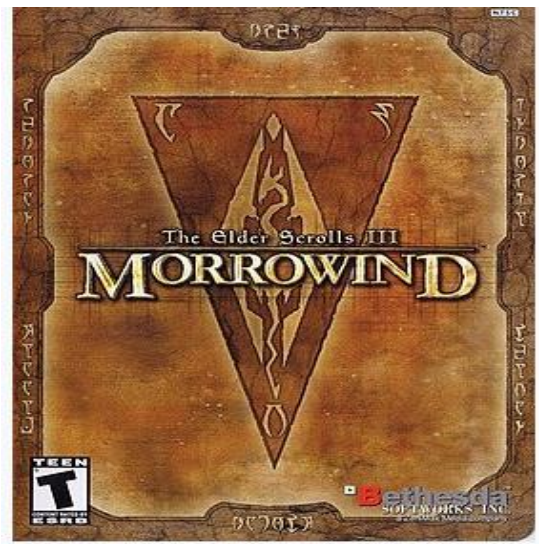


Рисунок 1.1 – Гра Elder Scrolls 3



Рисунок 1.2 – Гра Neverwinter Nights



Рисунок 1.3 – Гра Anachronox

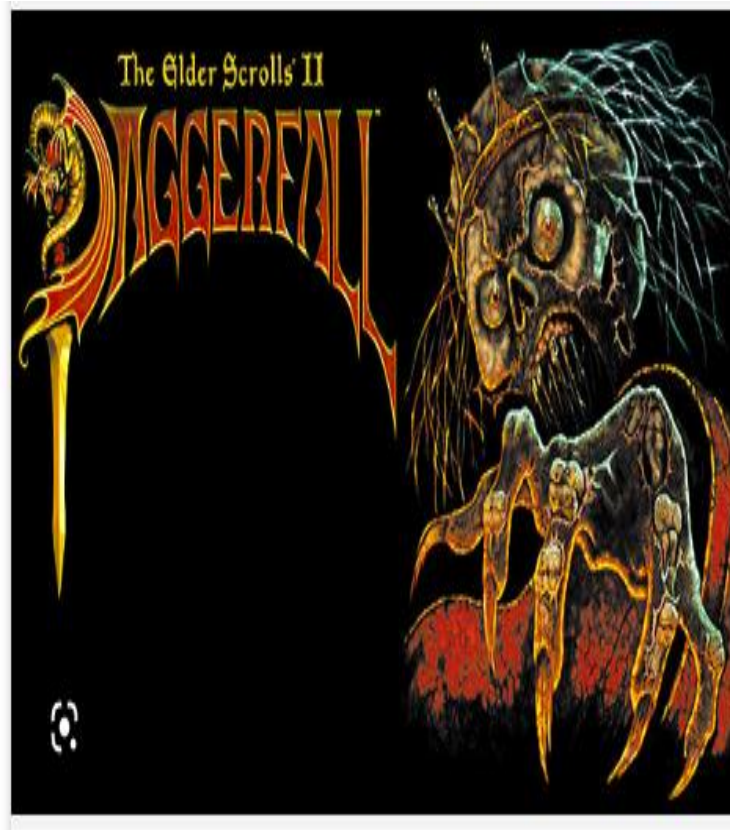


Рисунок 1.4 – Гра The Elder Scrolls II: Daggerfall

Розробники: Elder Scrolls: 3 розроблено компанією Bethesda Game Studios. Neverwinter Nights: розроблено компанією BioWare. Anachronox: розроблено компанією Ion Storm. The Elder Scrolls II: Daggerfall: розроблено компанією Bethesda Softworks.

Архітектура: Всі ці ігри базуються на ігровому рушію, який забезпечує основну функціональність гри, включаючи графіку, фізику та звук. Наприклад, Elder Scrolls 3 використовує рушія Gamebryo, а Neverwinter Nights – Aurora. Ці ігри мають велике, відкрите середовище, яке створюється за допомогою поєднання процедурних і ручних технік. Світ населений неігровими персонажами (NPC), які взаємодіють з гравцем і надають квести та інформацію. Ці ігри мають бойову систему в реальному часі або покрокову бойову систему, яка дозволяє гравцям вступати в битви з ворогами. Боротьба зазвичай базується на поєднанні навичок і статистики та може включати використання зброї, заклинань чи інших здібностей [22].

Інтерфейс користувача

Мова реалізації: Найімовірніше, вони реалізовані на мові C++ та Python.

Перелік функцій:

1) Середовища з відкритим світом всі ці ігри мають велике, захоплююче середовище відкритого світу, яке гравець може вільно досліджувати, з різним ступенем вибору та нелінійності [3].

2) Неігрові персонажі (NPC) у всіх цих іграх є ряд NPC, з якими гравець може взаємодіяти, зокрема торговці, квестери та компаньйони. NPC можуть мати власні цілі та характери, а іноді можуть надавати важливу інформацію або допомогу гравцеві [9].

3) Захоплюючий звук та візуальні ефекти ці ігри вирізняються захоплюючим звуковим оформленням, музикою та озвученням, а також деталізованою та реалістичною графікою, що посилює відчуття занурення в ігровий світ [9].

Аналіз переваг та недоліків даного ПЗ:

Переваги:

- різноманітність ігрового процесу;
- реіграбельність;
- підтримка модифікацій;

– можливість обирати рівень складності;

Недоліки:

– технічні проблеми;

– швидка сюжетна лінія;

– витрати часу;

Джерело інформації (вебсайт):

Elder Scrolls III: Morrowind – <https://elderscrolls.bethesda.net/en/morrowind>

Neverwinter Nights – <https://www.beamdog.com/products/neverwinter-nights-enhanced-edition>

Anachronox – офіційного вебсайту для Anachronox немає, але фанати можуть обговорювати гру на таких форумах, як RPG Codex або Steam.

Elder Scrolls II: Daggerfall – Як для старої гри, для Daggerfall немає офіційного вебсайту, але фанати можуть отримати доступ до таких ресурсів, як утиліта DaggerfallSetup та фанатські модифікації на таких сайтах, як Nexus Mods.

Аналіз системи, що розробляється

- призначення ПЗ;

Мета ігри – надати гравцям захопливий і цікавий досвід у віртуальному світі. Вони дозволяють гравцям досліджувати величезні та деталізовані світи, а також брати участь у різноманітних ігрових активностях, таких як битви та рольові ігри. Ігри часто мають складні сюжетні лінії та персонажів, що запам'ятовуються, які заохочують гравців вкладати час і зусилля в їхні пригоди. Зрештою, мета ігри – надати гравцям багатий і корисний досвід, яким вони зможуть насолоджуватися і запам'ятовувати на довгі роки.

- опис структури системи

Ігровий рушій – ігровий рушій слугує ядром ігрової системи та надає розробникам ряд інструментів і функцій для створення ігрового світу, механіки гри та користувацького інтерфейсу [13].

Ігровий світ – це віртуальне середовище, в якому відбувається гра. Він може складатися з різних місцевостей, ландшафтів, будівель та об'єктів, кожен з яких має свій власний унікальний дизайн та характеристики.

Персонажі – це сутності, якими гравці керують або з якими взаємодіють в ігровому світі. Вони можуть бути заздалегідь спроектовані розробниками або створені гравцями. Вони мають власні унікальні атрибути, такі як здоров'я, сила, спритність та інші здібності, які впливають на їхню поведінку в грі [16].

Неігрові персонажі (NPC) – це персонажі, які контролюються ігровою системою, а не гравцями. Вони можуть бути ворогами, союзниками або нейтральними персонажами в ігровому світі та надавати гравцям квести, предмети або інші винагороди [16].

Бойова система – це набір правил і механік, які визначають, як гравці та NPC беруть участь у битвах і конфліктах. Вона включає такі елементи, як зброя, обладунки, заклинання та інші здібності, які гравці можуть використовувати для перемоги над супротивниками [16].

Квести та місії – це завдання або цілі, які гравці повинні виконати для проходження гри. Вони можуть включати в себе бій, дослідження, вирішення головоломок або інші завдання, які вимагають від гравців використання їхніх навичок та вмінь.

Користувацький інтерфейс – це засіб, за допомогою якого гравці взаємодіють з ігровою системою. Він включає в себе такі елементи, як меню, HUD і елементи керування, які дозволяють гравцям отримати доступ до різних можливостей гри [9].

- засоби апаратної та програмної реалізації

Апаратне забезпечення:

Комп'ютер: Для запуску рушія Unreal Engine, а також для створення та відтворення ігор потрібен високопродуктивний комп'ютер з потужною відеокартою та достатньою кількістю оперативної пам'яті.

Консоль: Гра може бути розроблена для таких консолей, як Xbox або PlayStation, які вимагають специфічних апаратних конфігурацій та інструментів розробки [16].

Програмне забезпечення:

Unreal Engine є основним програмним інструментом, який використовується для створення ігор цього жанру. Він надає розробникам цілий ряд інструментів і функцій для створення ігрових світів, персонажів і механік ігрового процесу.

Програмне забезпечення для 3D-моделювання створення та проектування ігрового світу та персонажів можна використовувати програмне забезпечення для 3D-моделювання, наприклад, Blender або Maya [11].

Мови програмування щоб додати грі інтерактивності та функціональності, можна використовувати такі мови програмування, як C++ або Blueprint (візуальна мова сценаріїв, що використовується в рушії Unreal Engine) [21].

Аудіо та музичне програмне забезпечення звукові ефекти та музика є важливими елементами ігрового досвіду, і для створення та редагування аудіофайлів можна використовувати такі програми, як Adobe Audition або Pro Tools [21].

Програмне забезпечення для контролю версій, таке як Git або SVN, можна використовувати для управління базою коду та співпраці з іншими розробниками над проєктом.

Загалом, апаратні та програмні засоби реалізації, необхідні для ігрового додатку в жанрі рольового екшену на базі рушія Unreal Engine, покликані забезпечити розробників інструментами та ресурсами, необхідними для створення багатого та захоплюючого ігрового досвіду [12].

1.4 Специфікація вимог до програмного забезпечення комп'ютерної гри

1. Функціональні вимоги

1.1 Ігрова механіка

Гра повинна надавати гравцеві можливість зануритися у рольову гру з повним зануренням. Вона повинна мати відкритий світ, який дозволяє гравцеві вільно досліджувати ігровий світ, взаємодіяти з NPC, виконувати квести та брати участь у боях. Гра також повинна включати систему кастомізації персонажа, яка дозволяє гравцеві обирати зовнішність і здібності свого героя.

Бойова система повинна бути інтуїтивно зрозумілою та захоплюючою, з різноманітною зброєю, заклинаннями та здібностями на вибір. Гравець повинен мати можливість вступати в бої з NPC та іншими гравцями, з можливістю об'єднуватися з іншими гравцями для вирішення складніших завдань [12].

Гра повинна включати систему рівнів, яка дозволяє гравцеві розвиватися і покращувати здібності свого персонажа. Виконуючи квести та битви, гравець має заробляти очки досвіду, які він може використати для підвищення рівня та розблокування нових здібностей.

1.2 Інтерфейс користувача:

Користувацький інтерфейс повинен бути зручним та інтуїтивно зрозумілим, дозволяючи гравцям легко орієнтуватися в ігровому світі та меню. Він повинен містити опції для налаштування ігрових параметрів, таких як якість графіки та звуку. Гра також повинна мати систему навчання, яка допоможе новим гравцям вивчити механіку гри [12].

1.3 Аудіо та візуальні ефекти

Гра повинна мати високоякісну графіку та візуальні ефекти для створення захоплюючого середовища. Ігровий світ повинен бути добре деталізованим, з реалістичним освітленням і динамічними погодними ефектами. Гра також повинна включати різноманітні звукові ефекти та фонову музику, щоб покращити досвід гравця.

2. Нефункціональні вимоги

2.1 Продуктивність

Гра має бути оптимізована для безперебійної роботи на різних апаратних конфігураціях. Вона повинна завантажуватися швидко та ефективно, з

мінімальними затримками та часом завантаження. Гра також має бути оптимізована для роботи з постійною частотою кадрів, навіть під час інтенсивних боїв або в місцях великого скупчення людей [12].

2.2 Безпека

Гра повинна передбачати заходи для запобігання шахрайству та злому. Це може включати в себе програмне забезпечення для боротьби з шахрайством, перевірку дій гравців на стороні сервера та шифрування даних гравців для запобігання несанкціонованому доступу.

2.3 Сумісність

Гра повинна бути сумісною з різними операційними системами та конфігураціями обладнання, включаючи ПК та ігрові консолі. Гра також повинна бути сумісною з популярними ігровими платформами, такими як Steam, Xbox Live та PlayStation Network [5].

Обмеження

Розробка гри повинна бути завершена в рамках встановленого бюджету і термінів. Це може вимагати використання гнучких методологій розробки для забезпечення гнучкості та швидкої ітерації. Гра також повинна відповідати всім відповідним законодавчим та нормативним вимогам, включаючи закони про захист даних та права інтелектуальної власності. Нарешті, гра повинна бути розроблена з використанням ігрового рушія Unreal Engine та інших супутніх програмних інструментів [21].

Специфікація вимог до програмного забезпечення слугує планом для розробки ролевого екшену на основі рушія Unreal Engine. Визначивши функціональні та нефункціональні вимоги, а також будь-які обмеження, команда розробників може гарантувати, що кінцевий продукт відповідатиме потребам гравців і буде завершений в рамках встановленого бюджету та термінів [21].

Висновки до розділу 1

На основі проведеного аналізу предметної сфери розробки ігрового застосунку в жанрі ролевого екшену на основі рушія UE, було виявлено, що розробка ігор – це складний процес, який вимагає поєднання технічних знань і творчого бачення. Особливо популярним і складним є жанр рольового екшену, прикладами якого є такі ігри, як Elder Scrolls 3, Neverwinter Nights, Anachronox та The Elder Scrolls II: Daggerfall. Ці ігри побудовані на тривірневій архітектурі і потребують широкого спектру апаратних і програмних засобів для реалізації.

Розробка успішного рольового екшену вимагає ретельного планування та уваги до деталей. Специфікація вимог до програмного забезпечення відіграє вирішальну роль в окресленні обсягу проєкту та визначенні необхідних функцій і можливостей. Вона також слугує дорожньою картою для команди розробників, гарантуючи, що всі працюють над досягненням однакових цілей і завдань.

Загалом, процес розробки ігор постійно розвивається, оскільки з'являються нові технології та змінюються вподобання споживачів. Однак одне залишається незмінним: важливість надання високоякісного, захопливого ігрового досвіду, який резонує з гравцями і змушує їх повертатися за новими іграми.

2 МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ ГРИ

2.1 Діаграма прецедентів

Діаграми використання зазвичай відображають дані за допомогою комбінації лінійних графіків, гістограм та інших візуалізацій. Дані можуть бути представлені у різний спосіб, залежно від конкретних потреб користувача та типу інформації, що відстежується. Загалом, діаграми використання можуть надати цінну інформацію про моделі поведінки та допомогти користувачам приймати більш обґрунтовані рішення щодо споживання ресурсів.

Діаграма станів, також відома як діаграма автомата або діаграма станів, – це графічне зображення поведінки системи або об'єкта у відповідь на зовнішні стимули. Вона широко використовується в програмній інженерії, інженерії управління та інших галузях для моделювання поведінки складних систем [22].

Діаграма станів складається з набору станів, які представляють різні умови або режими, в яких може перебувати система або об'єкт, і переходів, які представляють події або умови, що змушують систему переходити з одного стану в інший. Діаграма може також включати дії, які представляють дії, що виконуються, коли відбувається перехід [22].

Діаграми станів корисні для моделювання складних систем, оскільки вони дозволяють проєктувальникам розбити систему на менші, більш керовані частини і чітко визначити поведінку кожної частини. Їх також можна використовувати для виявлення потенційних проблем або вузьких місць у системі, а також для тестування і доопрацювання проєкту перед його реалізацією. Діаграми станів можуть бути створені за допомогою різних інструментів, в тому числі спеціально розроблених для цього програм [22].

На табл. 2.1 наведено глосарій проєкту

Таблиця 2.1 – Словник проєкту

Розробники	Спеціалісти, які розробляють гру
Гейм-дизайнер	Спеціалісти, які розробляють графіку для гри
Менеджери	Люди які просувають гру за допомогою реклами
Гравці	Люди які використовують продукт

2.2 Діаграма варіантів використання

Діаграма використання – це тип графіка або діаграми, що відображає дані про використання або споживання певного ресурсу чи продукту протягом певного періоду часу. Це може включати інформацію про те, як часто використовується продукт або ресурс, скільки його використовується кожного разу і як змінюються моделі використання з часом. На рисунку 2.1 наведено діаграма варіантів [22].

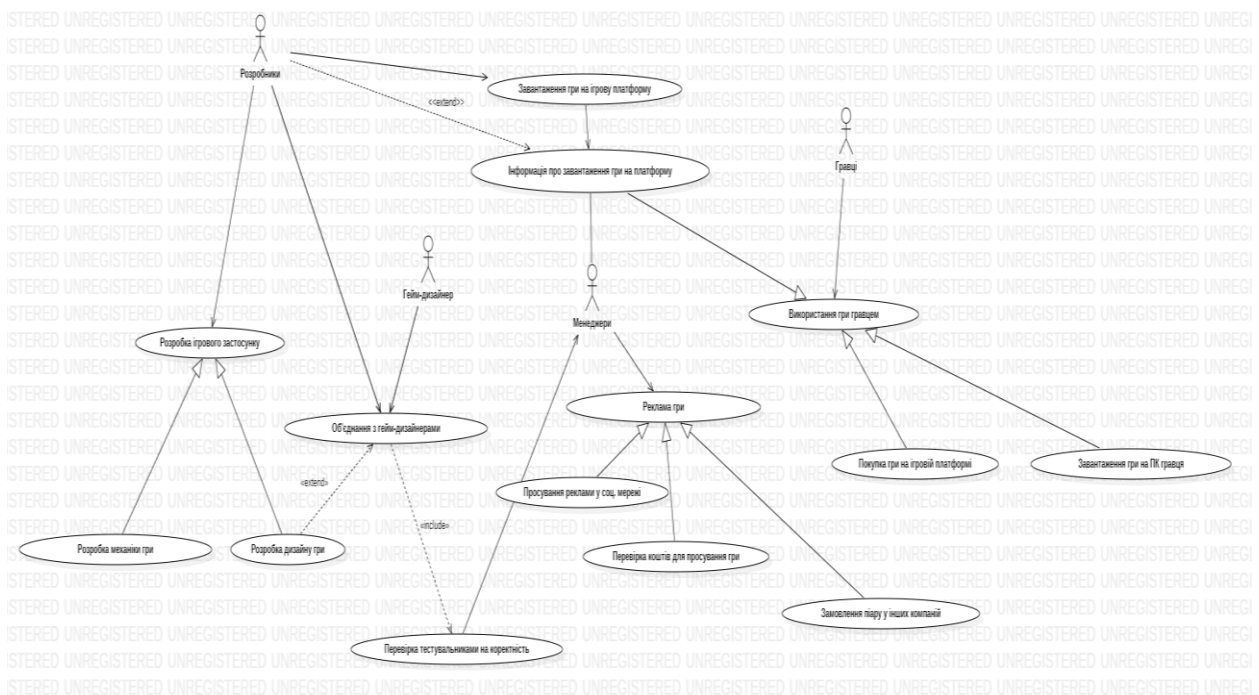


Рисунок 2.1 – Діаграма варіантів використання для розробки ігрового застосунку

Діаграма має в собі 4 actors, такі як розробники, гейм-дизайнери, менеджери та гравці. Розробник має 3 use cases, один из них головний, два

інших дочірні. З дочірнього use case(розробка дизайну гри), йде розширення до кейсу об'єднання з гейм дизайнерами, після цього через include включення використовується кейс(перевірка тестувальниками на коректність). Після цього, астор менеджери мають головний кейс(реклама гри), та 3 дочірніх від нього. Після цього астор розробники, виконують кейс(завантаження гри на платформу), де астор гравці, можуть встановити собі гру на ПК, через два дочірніх кейса [22].

2.3 Опис дійових осіб

Розробники – починає розробку ігрового застосунку.

Гейм-дизайнери – розробляє графіку ігрового застосунку.

Менеджери – роблять та просувають рекламу для ігрового застосунку.

Гравці – використовують вже повністю готовий продукт.

Варіант використання «Розробка ігрового застосунку»

На таблиці 2.2 наведено варіант використання «Розробка ігрового застосунку».

Таблиця 2.2 – Головний розділ сценарію виконання варіанта використання «Розробка ігрового застосунку»

Варіант використання	Розробка ігрового застосунку
Актори	Розробник гри, тестувальники, дизайнери, маркетологи
Короткий опис	Описується детальну розробку ігрового застосунку
Мета	Розробити ігровий застосунок
Тип	Базовий

На таблиці 2.3 наведено хід подій варіанту використання «Розробка ігрового застосунку».

Таблиця 2.3 – Типовий хід подій сценарію виконання варіанта використання «Розробка ігрового застосунку»

Дії актора	Відгук системи
1. Концептуалізація: Розробник гри придумує ідею для гри, яка сподобається широкій аудиторії та матиме потенціал для отримання прибутку.	4. Запуск: Гра запускається і стає доступною для публіки через магазини додатків або інші платформи.
2. Дизайн та розробка: Команда дизайнерів створює зовнішній вигляд гри, команда програмістів пише код, а команда тестувальників гарантує, що гра не містить помилок і працює безперебійно.	5. Монетизація: Гра отримує дохід за допомогою різних методів монетизації, таких як внутрішні покупки, реклама або моделі на основі підписки.
3. Маркетинг та просування: Команда маркетологів просуває гру через різні канали, такі як соціальні мережі, реклама та зв'язки з громадськістю, щоб охопити широку аудиторію.	6. Постійні оновлення та підтримка: Гра постійно оновлюється для виправлення помилок і додавання нових функцій, щоб гарантувати, що гра залишається актуальною і приємною для геймерів.

На таблиці 2.4 наведено винятки варіанту використання «Розробка ігрового застосунку».

Таблиця 2.4 – Винятки сценарію виконання варіанта використання «Розробка ігрового застосунку»

Дії актори	Відгук системи
Виняток №1. Команда тестувальників знайшла помилку	

Кінець таблиці 2.4.

	7.Гра відображає помилку про неправильну загрузку у головне меню.
8. Розробники заново перевіряють код та виправляють помилку.	

Варіант використання «Просування гри через різні канали»

На таблиці 2.5 наведено варіант використання «Просування гри через різні канали».

Таблиця 2.5 – Головний розділ сценарію виконання варіанта використання «Просування гри через різні канали»

Варіант використання	Просування гри через різні канали
Актори	Маркетологи
Мета	Просунути гру на різні платформи
Короткий опис	Просунути гру на різні ігрові платформи
Тип	Підлеглий

На таблиці 2.6 наведено хід подій варіант використання «Просування гри через різні канали».

Таблиця 2.6 – Типовий хід подій сценарію виконання варіанта використання «Просування гри через різні канали»

Дії актора	Відгук системи
1. Маркетологи виділяють гроші на маркетинг	3. Система дає повідомлення про збільшення завантаження гри

Кінець таблиці 2.6

2. Маркетологи завантажують гру на різні платформи	
--	--

На таблиці 2.7 наведено винятки варіанту використання «Просування гри через різні канали».

Таблиця 2.7 – Винятки сценарію виконання варіанта використання «Просування гри через різні канали»

Дії актора	Відгук системи
Виняток №2. Брак маркетологу	
5. Маркетологи виділяють більше коштів, та покупають більше реклами для гри	4. Система відображає інформацію про те, що не збільшується завантаження гри

Варіант використання «Розробка дизайну»

На таблиці 2.8 наведено варіант використання «Розробка дизайну».

Таблиця 2.8 – Головний розділ сценарію виконання варіанта використання «Розробка дизайну»

Варіант використання	Розробка дизайну
Актори	Дизайнери
Короткий опис	Зробити сучасну графіку
Мета	Розробити графіку гри за ідеєю розробника
Тип	Підлеглий

На таблиці 2.9 наведено хід подій варіанту використання «Розробка дизайну».

Таблиця 2.9 – Типовий хід подій сценарію виконання варіанта використання «Розробка дизайну»

Дії актора	Відгук системи
1. Дизайнери починають розробку	3. Прийшло повідомлення, про завантаження
2. Завантажують свій проєкт	

На таблиці 2.10 наведено винятки варіанту використання «Розробка дизайну».

Таблиця 2.10 – Винятки сценарію виконання варіанта використання «Розробка дизайну»

Дії актора	Відгук системи
Виняток №3. Дизайнери не правильно завантажили проєкт у гру	
2. Перезавантаження проєкту	1. Повідомлення про помилку

Варіанти використання дали розуміння того як саме розробка гри повинна працювати та які головні сценарії має, та розглянувши винятки, для того щоб виправити заздалегідь якісь помилки [22].

2.2 Діаграма станів

Діаграма станів – це графічний засіб моделювання системи, що показує різні стани, в яких може перебувати ця система, а також переходи між цими станами. Діаграми станів широко використовуються для моделювання поведінки об'єктів, програмних систем, протоколів та інших систем.

У діаграмі станів стан системи відображається у вигляді овалів або прямокутників з назвою стану всередині. Переходи між станами показуються стрілками, які показують напрямок переходу. Кожен перехід може мати певні умови або події, що спричиняють перехід, та/або дії, які відбуваються при переході [22].

Діаграми станів допомагають розібратися в складній системі, зрозуміти її можливі стани і взаємодії між ними. Вони можуть бути використані для аналізу та проєктування систем, виявлення помилок в логіці програм або уточнення вимог до системи.

На рисунках 2.2 – 2.4 наведено діаграми станів.

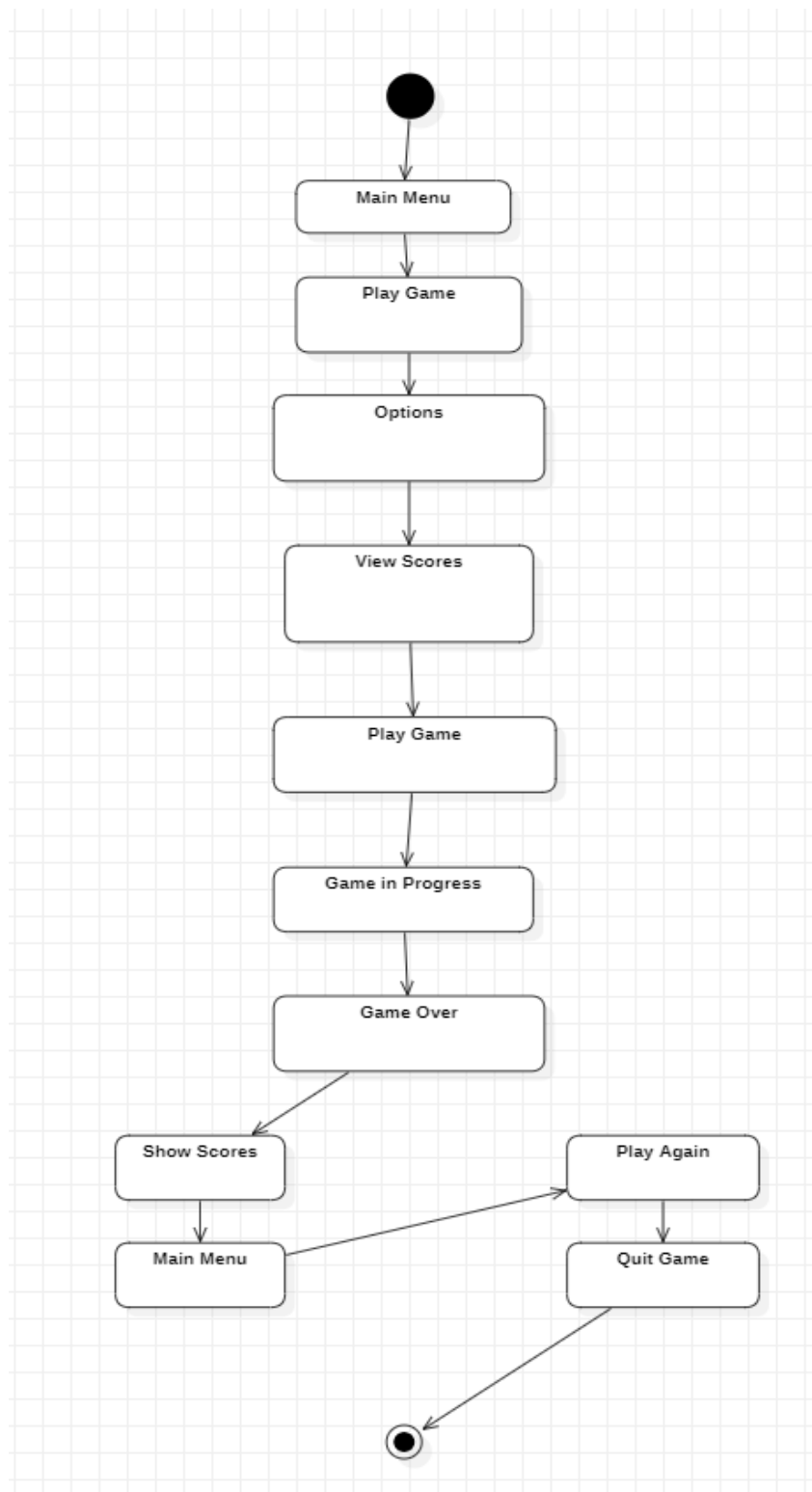


Рисунок 2.2 – Діаграма стану Main Menu

На першій діаграмі описується головне меню, діаграма виконана послідовно, має старт та фінал. Має 11 Simple State які послідовно з'єднанні друг за другом.

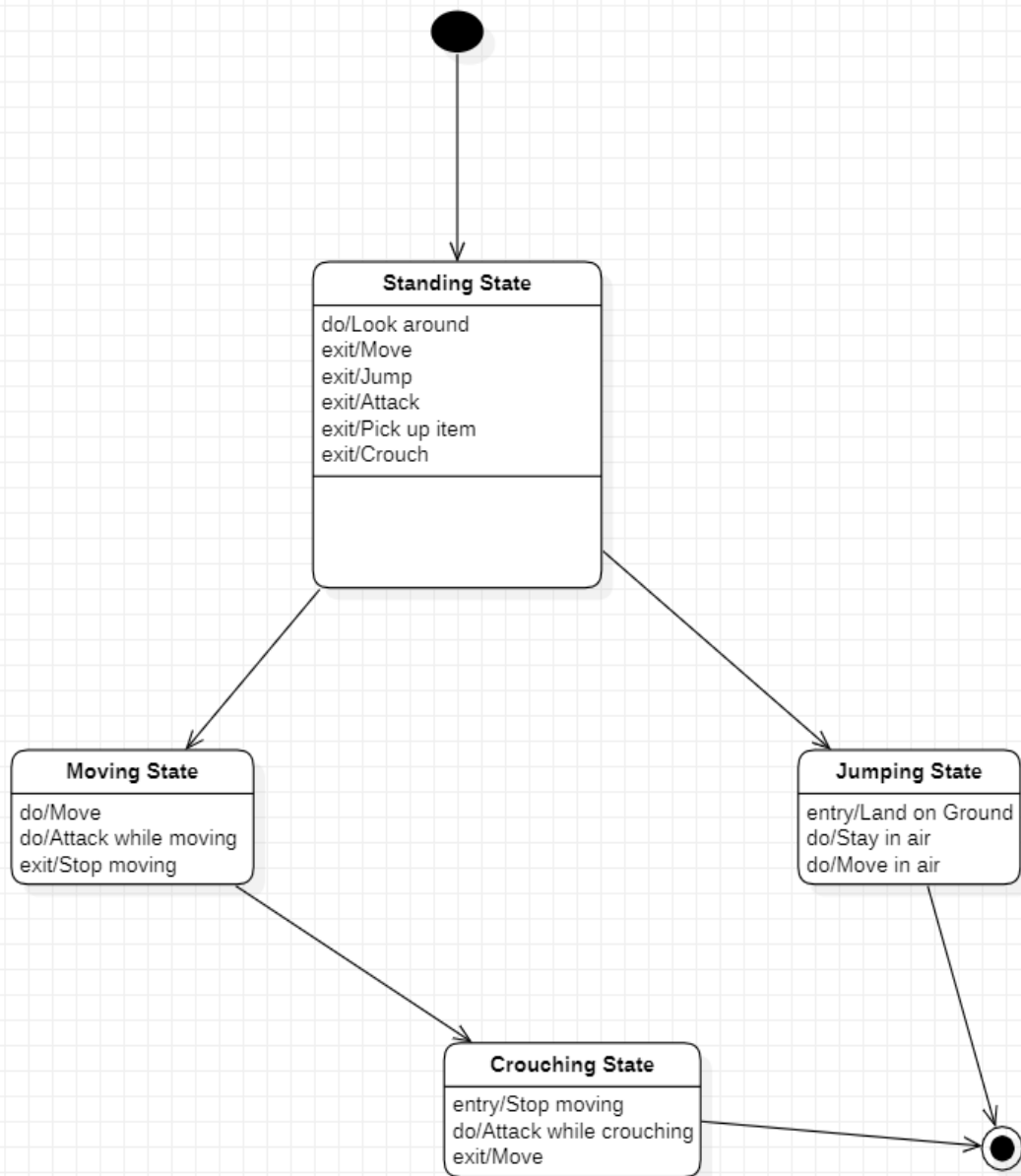


Рисунок 2.3 – Діаграма стану Standing

На другій діаграмі описується Standing State, діаграма має 4 State, у standing є 6 активностей, один do та 5 exit, standing переходить до 2 інших які називаються Moving та Jumping. Moving має один do та два exit, Jumping має один entry та два do. 3 State Moving переходить до Crouching State, який має 3

активності(entry, do, exit) та фінал йде як від Crouching State так, і Jumping State.

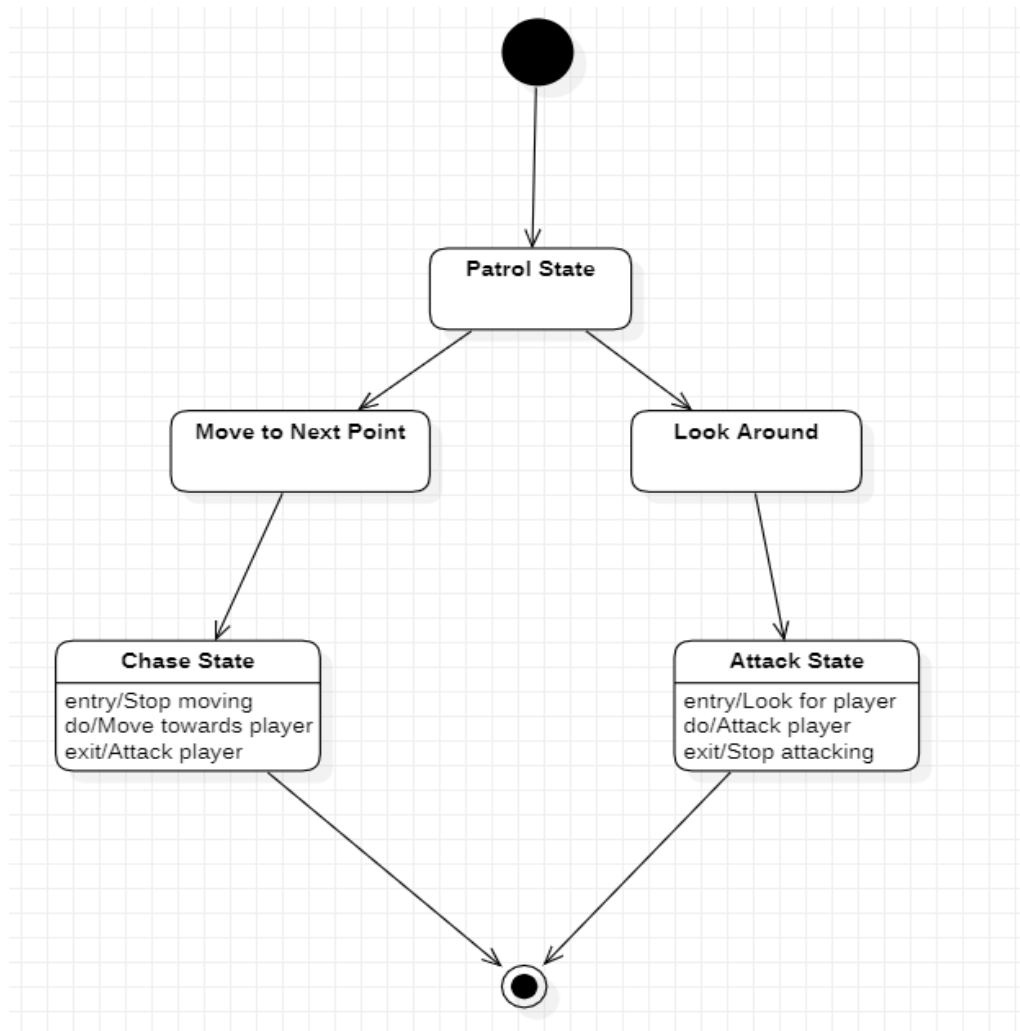


Рисунок 2.4 – Діаграма стану Patrol

На третій діаграмі описується Patrol State, який переходить до Move to next point та Look around. З Move to next point переходить до Chase State який має 3 активності(entry, do, exit). З Look around переходить до Attack State який має теж 3 активності(entry,do,exit). Та фінал йде як від Chase State, так і Attack State.

Висновки до розділу 2

У другому розділі було проведено аналіз Use case для кожного з акторів, включаючи їх дії при розробці гри. Було проведено до кожного Use case свої винятки, та їх виправлення. Було наведено три діаграми станів, такі як,

діаграма Main Menu, Standing та Patrol. У кожній діаграмі було створено початкову та фінальну точку та мають різні активності.

Розроблено діаграма використання, яка відображає дані про використання або споживання певного ресурсу чи продукту протягом певного періоду часу. Вони можуть надати цінну інформацію про моделі поведінки і допомогти користувачам приймати більш обґрунтовані рішення щодо споживання ресурсів.

Розроблено діаграма станів, яка моделює поведінку системи або об'єкта у відповідь на зовнішні стимули. Вони складаються з набору станів, переходів та дій і корисні для розбиття складних систем на менші, більш керовані частини, а також для тестування та вдосконалення дизайну перед впровадженням. Загалом, як діаграми використання, так і діаграми станів є важливими інструментами для розуміння та моделювання складних систем.

За допомогою цих етапів, було показано, як саме проходить розробка відео гри , показано ефективність та за допомогою діаграм було показано зручність розробки ігрового застосунку.

3 ПРОЄКТУВАННЯ ІГРОВОГО ЗАСТОСУНКУ

3.1 Опис ігрового циклу

Ігровий додаток, що розробляється, є рольовим екшеном, який має на меті занурити гравців в інтерактивний та динамічний віртуальний світ. Гравці беруть на себе роль персонажа і вирушають в епічну подорож, наповнену квестами, битвами, розвитком персонажа та дослідженнями [12].

Саме ролевої екшен є одним з найпопулярніших жанрів у ігровій індустрії.

На рисунок 3.1 наведено діаграма найпопулярніших ігрових жанрів [3].

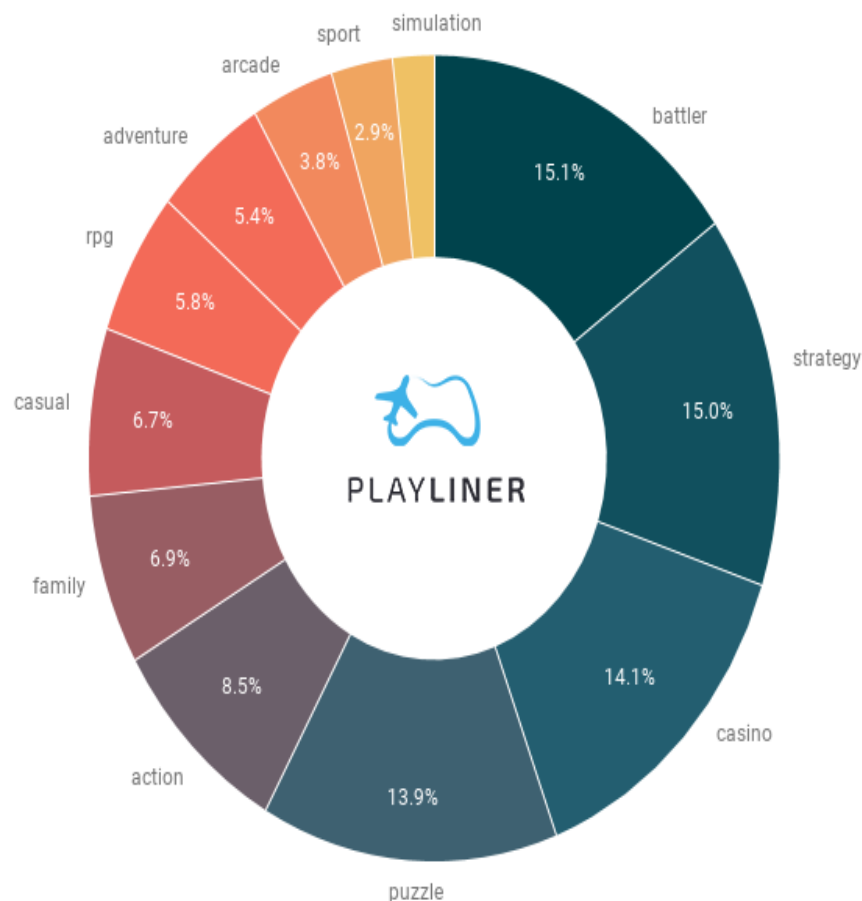


Рисунок 3.1 – Діаграма найпопулярніших жанрів

Екшен містить в собі багато захоплюючі процеси для гравця такі як:

Квести та розповідь:

Гра має багатий та захопливий сюжет, який розгортається через динамічну систему квестів. Гравці виконують різноманітні завдання, які

розвивають сюжет, розкривають таємниці та ведуть до захоплюючих зустрічей і випробувань. Ці квести варіюються від основних сюжетних ліній до додаткових побічних квестів, які пропонують додатковий контент і можливості для розвитку персонажа [12].

Дослідження та відкритий світ:

Ігровий світ є широким і відкритим, запрошуючи гравців досліджувати його різноманітне середовище. Вони можуть відправитися в густі ліси, підступні гори, галасливі міста та моторошні підземелля. Дослідження винагороджується захищеними скарбами, цінними предметами та зустрічами з унікальними неігровими персонажами (NPC), які можуть надавати квести, цінну інформацію або брати участь у торгівлі [11].

На рисунку 3.2 наведено відкритий світ ігрового застосунку.



Рисунок 3.2 – Відкритий світ ігрового застосунку

Боротьба та екшн:

Основна механіка ігрового процесу обертається навколо бою в реальному часі, пропонуючи гравцям захопливі битви проти різних ворогів, включаючи могутніх босів та інших гравців у PvP-сценаріях. Боротьба вимагає прийняття стратегічних рішень, дій на основі навичок та ефективного використання здібностей персонажа, щоб досягти успіху в динамічних і складних сутичках.

На рисунку 3.3 наведено бій з мобом.



Рисунок 3.3 – Демонстрація бойових дій с мобом

Розвиток персонажа:

Виконуючи квести та перемагаючи ворогів, гравці заробляють очки досвіду, валюту та цінну здобич, що сприяють розвитку та прогресу їхнього персонажа. Очки досвіду дозволяють гравцям підвищувати рівень, відкриваючи нові здібності та навички. За валюту можна придбати зброю, броню та витратні матеріали, щоб покращити свої бойові можливості [12].

На рисунку 3.3 наведено отримання ігрового рівня у грі.



Рисунок 3.3 – Демонстрація отримання досвіду у грі

Реіграбельність та ендгейм-контент:

Щоб забезпечити довготривале залучення, гра пропонує реіграбельність завдяки численним варіантам створення персонажів, розгалуженим сюжетним лініям та складному контенту в кінці гри.

3.2 Розробка ігрових механік

Цей розділ присвячений розробці ігрової механіки для ролевого екшену. Ігрова механіка охоплює правила, взаємодії та системи, які керують ігровим процесом. Наступні ігрові механіки були розроблені для покращення занурення, прийняття стратегічних рішень та загального задоволення для гравців [12].

Розвиток персонажа:

Механіка розвитку персонажа дає гравцям відчуття зростання та розвитку в міру проходження гри. Гравці заробляють очки досвіду, виконуючи квести, перемагаючи ворогів та досягаючи віх. Накопичені бали досвіду дозволяють персонажам підвищувати рівень, відкриваючи нові здібності, навички та можливості кастомізації. Ця механіка гарантує, що гравці постійно отримують винагороду за свої зусилля і мотивують продовжувати гру [12].

Бої на основі навичок:

Бойова механіка в грі зосереджена на взаємодії на основі навичок, щоб забезпечити захопливі та динамічні битви. Гравці мають доступ до різноманітної зброї, заклинань та здібностей, які вимагають точного розрахунку часу, стратегічного позиціонування та швидких рефлексів. Успішне виконання цих бойових навичок дозволяє гравцям ефективно перемагати ворогів, тоді як погане виконання може призвести до несприятливого результату. Ця механіка заохочує гравців брати активну участь у боях і розвивати свої бойові навички [21].

Дослідження та відкриття:

Механіка дослідження відіграє життєво важливу роль у грі, заохочуючи гравців подорожувати величезним відкритим світом і відкривати приховані скарби, таємні зони та інтригуючі сюжетні лінії. Гравці можуть мандрувати різноманітними середовищами, розв'язувати головоломки, взаємодіяти з NPC

та знаходити цінну здобич. Механіка дослідження додає відчуття дива та відкриття, винагороджуючи гравців унікальним досвідом та додатковим контентом за межами основної квестової лінії [21].

Система квестів:

Надійна система квестів є основою сюжету гри та розвитку ігрового процесу. Гравці беруть участь у різноманітних квестах, включаючи основні сюжетні лінії та побічні квести, кожен з яких має свої цілі, виклики та винагороди. Квести можуть включати битви, розв'язання головоломок, діалогові взаємодії або дослідження, забезпечуючи різноманітний ігровий досвід. Система квестів проводить гравців через ігровий світ, розкриває історію та пропонує можливості для розвитку персонажа і занурення в гру [16].

Ітемізація:

Механіка крафтингу та деталізації дозволяє гравцям збирати ресурси, та налаштовувати своє спорядження. Гравці можуть збирати матеріали з переможених ворогів, збирати з навколишнього.

Розвиток цих ігрових механік покращує досвід ролевого екшену, надаючи гравцям захоплюючий розвиток персонажа, бої на основі навичок, дослідження, осмислене прийняття рішень, можливості для крафтингу та багатокористувацької взаємодії. Ці механіки працюють в гармонії, створюючи захоплюючий і приємний ігровий досвід, який захоплює гравців, дозволяючи їм прокладати свій власний шлях в динамічному і захоплюючому віртуальному світі [16].

3.3 Проєктування ігрової локації

Створення захопливої ігрової локації має вирішальне значення для створення захопливого та цікавого ігрового процесу. Цей розділ зосереджується на ключових аспектах і міркуваннях, пов'язаних зі створенням ігрової локації, яка приваблює гравців, надає можливості для дослідження і покращує загальний наратив та ігровий процес [10].

Тема та декорації:

Першим кроком у створенні ігрової локації є визначення теми та оточення. Тема визначає загальний настрій, атмосферу та стиль локації, в той час як сеттінг визначає її фізичні та географічні характеристики. Незалежно від того, чи це постапокаліптичне місто, містичний ліс чи футуристична космічна станція, тема і сеттінг повинні відповідати жанру гри, її сюжетній лінії та очікуваному досвіду гравця [10].

Дизайн середовища:

Дизайн середовища охоплює візуальні елементи, архітектуру та планування ігрової локації. Він передбачає створення унікальних і візуально привабливих ландшафтів, споруд і пам'яток, які передають бажану атмосферу і занурюють гравців у ігровий світ. Увага до деталей, послідовність та узгодженість елементів дизайну допомагають створити правдоподібне та захоплююче середовище [10].

Дизайн локації:

Дизайн локації передбачає створення просторового планування та потоку ігрової локації. Він включає в себе розміщення перешкод, шляхів, інтерактивних елементів та цікавих місць. Дизайн локації повинен заохочувати до дослідження, пропонувати стратегічний вибір і давати відчуття прогресу в міру того, як гравці переміщуються по локації. Баланс між викликом і винагородою має важливе значення для підтримання залученості гравця та задоволення від ігрового процесу [10].

Інтерактивні елементи:

Щоб покращити взаємодію та занурення гравців у гру, в ігрову локацію слід включити інтерактивні елементи. Це можуть бути рухомі об'єкти, приховані проходи, середовище, що руйнується, головоломки та інтерактивні NPC. Інтерактивні елементи надають гравцям можливості для відкриттів, вирішення проблем та змістовних взаємодій, які поглиблюють їхній зв'язок зі світом гри [10].

Аудіо та звуковий дизайн:

Аудіо та звуковий дизайн відіграють важливу роль у створенні настрою та атмосфери ігрової локації. Звуки навколишнього середовища, фонова музика та діалоги персонажів повинні доповнювати візуальний дизайн і викликати бажану емоційну реакцію у гравців. Звуковий супровід має бути ретельно продуманий, щоб посилити занурення, підвищити напругу та надати звукові підказки, які допоможуть гравцям пройти локацію [10].

Інтеграція розповіді:

Ігрова локація повинна бути органічно інтегрована в загальний наратив і слугувати інструментом розповіді. Саме середовище може передавати інформацію про історію, культуру та легенди ігрового світу за допомогою технік розповіді про навколишнє середовище. Важливі сюжетні події, взаємодії персонажів і завдання квесту можуть бути прив'язані до конкретних локацій у ігровому світі, створюючи цілісний наративний досвід [10].

Ігрова агенція та дослідження:

Дизайн ігрової локації повинен давати гравцям відчуття самостійності та свободи у дослідженні. Наявність кількох шляхів, прихованих зон і додаткового контенту заохочує гравців взаємодіяти з навколишнім середовищем і заохочує їхню допитливість. Нелінійний дизайн, розгалуження шляхів та альтернативні маршрути дають гравцям відчуття вибору та контролю над ігровим процесом.

На рисунку 3.4 наведено ігрова локація море.



Рисунок 3.4 – Ігрова локація море

Розробка ігрової локації передбачає ретельний баланс між візуальним дизайном, плануванням рівнів, інтерактивними елементами, аудіо та інтеграцією розповіді. Враховуючи тему, оточення, активність гравця та

занурення, добре продумана ігрова локація може перенести гравців у захопливий віртуальний світ. Створення середовища, яке спонукає до дослідження, пропонує цікаві завдання та збагачує загальний ігровий досвід, сприяє незабутній та приємній ігровій подорожі для гравців [10].

3.4 Проектування ігрового інтерфейсу

Проектування ігрового інтерфейсу є критично важливим аспектом створення захопливого та зручного ігрового досвіду. Цей розділ зосереджується на ключових міркуваннях і принципах, пов'язаних з проектуванням інтуїтивно зрозумілого і візуально привабливого ігрового інтерфейсу, який покращує користувацький досвід і доступність [10].

Дизайн, орієнтований на користувача:

Підхід, орієнтований на користувача, має вирішальне значення при розробці ігрового інтерфейсу. Розуміння цільової аудиторії, її вподобань та шаблонів взаємодії має важливе значення для створення інтуїтивно зрозумілого та приємного у використанні інтерфейсу. Проведення досліджень користувачів, юзабіліті-тестування та врахування відгуків гравців протягом усього процесу проектування гарантує, що інтерфейс відповідатиме потребам та очікуванням гравців [10].

Рисунок 3.4 платформа з безкоштовними ресурсами для дизайну локації



Рисунок 3.4 - Платформа Epic Games для заохочення ресурсів

Візуальна ієрархія та організація:

Встановлення чіткої візуальної ієрархії та організація елементів інтерфейсу є життєво важливими для зручності використання. Найважливіші та найчастіше використовувані елементи повинні бути помітними та

легкодоступними, тоді як другорядні та рідше використовувані елементи можуть бути розміщені в менш помітних місцях. Послідовні візуальні підказки, такі як кольорове кодування, піктограми та написи, допомагають користувачам швидко зрозуміти інтерфейс та орієнтуватися в ньому.

Адаптивний і масштабований дизайн:

Зі збільшенням різноманітності ігрових платформ і пристроїв, розробка адаптивного і масштабованого інтерфейсу має вирішальне значення. Інтерфейс повинен адаптуватися до різних розмірів екрану, роздільної здатності та співвідношення сторін, щоб забезпечити оптимальну зручність використання на різних пристроях, включаючи настільні комп'ютери, консолі та мобільні пристрої. Масштабований дизайн інтерфейсу також пристосований до майбутніх оновлень, розширень і нових функцій без шкоди для загального користувацького досвіду [10].

Чітке та лаконічне відображення інформації:

Чітке та лаконічне представлення інформації є життєво важливим для ефективної комунікації в ігровому інтерфейсі. Використовуйте відповідну типографіку, розміри шрифтів та інтервали для покращення читабельності. Використовуйте лаконічні та змістовні написи, підказки та сповіщення, щоб спрямовувати гравців і надавати необхідну інформацію, не перевантажуючи їх. Візуалізація складних даних за допомогою діаграм, графіків або візуальних індикаторів також може допомогти ефективніше донести інформацію [10].

На рисунку 3.5 наведено відображення інформації про здоров'я, ману та досвід.



Рисунок 3.5 – Відображення інформації про здоров'я, ману та досвіду

Інтуїтивно зрозуміла навігація та керування:

Зручна навігація та інтуїтивно зрозумілі елементи керування сприяють плавному та приємному ігровому процесу. Елементи навігації, такі як меню,

кнопки та піктограми, повинні бути логічно організовані та надавати чіткі вказівки щодо їхньої функціональності. Інтуїтивно зрозумілі схеми керування, чуйний ввід і настроювані параметри керування дозволяють гравцям персоналізувати свій ігровий досвід і адаптуватися до свого улюбленого стилю гри [10].

Міркування щодо доступності:

Розробка інклюзивного ігрового інтерфейсу передбачає врахування потреб у доступності та пристосування до різних гравців. Включення таких функцій, як регульований розмір тексту, режими для дальтоніків та настроювані прив'язки клавіш, підвищує доступність для гравців з вадами зору, кольорового зору або руховими порушеннями. Надання альтернативних звукових сигналів, субтитрів і варіантів локалізації також покращує доступність гри для ширшої аудиторії.

Зворотній зв'язок та анімація:

Надання оперативного та змістовного зворотного зв'язку за допомогою анімації та візуальних підказок покращує користувацький досвід. Анімація може показувати прогрес, підтверджувати дії або надавати зворотній зв'язок про результати взаємодії гравця. Крім того, використання звукових підказок і тактильного зворотного зв'язку (якщо це можливо) ще більше занурює гравців у гру і посилює їхню зацікавленість у ній [23].

Розробка ігрового інтерфейсу вимагає ретельного врахування потреб користувачів, візуальної ієрархії, організації, адаптивного дизайну, інтуїтивної навігації та доступності. Добре продуманий інтерфейс покращує користувацький досвід, ефективно скеровує гравців і сприяє загальному зануренню в гру та отриманню задоволення від неї.

Висновки до розділу 3

У третьому розділі розробка ігрового додатку в жанрі ролевого екшену охоплює різні важливі елементи, включаючи ігровий цикл, ігрову механіку, дизайн ігрової локації та дизайн ігрового інтерфейсу. Кожен аспект сприяє створенню захопливого та цікавого ігрового досвіду для гравців.

Було описано ігровий цикл який забезпечує гравців структурованою та корисною системою розвитку, що охоплює створення персонажа, квести, бої, дослідження та соціальну взаємодію. Цей комплексний ігровий цикл гарантує, що гравці мають різноманітні види діяльності, в яких вони можуть брати участь, утримуючи їх захопленими та вмотивованими протягом усієї ігрової подорожі.

Було розглянуто розвиток ігрової механіки є життєво важливим для забезпечення глибини та складності ігрового процесу. Бойова механіка, що базується на навичках, змушує гравців опановувати стратегічне прийняття рішень, розрахунок часу та здібності персонажа, щоб досягти успіху в бою. Механіка розвитку персонажа стимулює гравців вкладати час і зусилля у розвиток свого героя, відкриваючи нові здібності та можливості кастомізації. Крім того, дослідження, прийняття рішень та багатокористувацька механіка збагачують ігровий досвід, сприяючи відчуттю занурення, активності та спільноти.

Було опрацьовано з проєктуванням ігрової локації передбачає створення візуально привабливого середовища, яке відповідає темі, сеттингу та наративу гри. Увага до деталей, дизайн рівнів та інтерактивні елементи сприяють дослідженню, відкриттю та залученню гравців до ігрового світу. Крім того, інтеграція ігрової локації в оповідь додає глибини та сенсу взаємодії та досвіду гравців.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНОЇ ГРИ

4.1 Вибір технологій розробки комп'ютерної гри

Unreal Engine 4 (UE4) – це потужна і широко використовувана платформа для розробки ігор, яка надає розробникам широкий набір інструментів і можливостей для створення високоякісних комп'ютерних ігор. У цьому технологічному огляді ми обговоримо ключові компоненти та можливості UE4 і дослідимо процес розробки комп'ютерної гри з використанням цієї платформи [19].

Система ігрового дизайну та створення сценаріїв:

UE4 пропонує візуальну систему сценаріїв, відому як система Blueprint, яка дозволяє розробникам створювати ігрову логіку без написання коду. Вона надає інтерфейс на основі вузлів, який дозволяє дизайнерам визначати механіку ігрового процесу, поведінку штучного інтелекту та інтерактивні елементи. Ця інтуїтивно зрозуміла система дає дизайнерам і художникам можливість швидко впроваджувати та ітеративно змінювати ігрові функції.

Візуалізація та графіка в реальному часі:

Unreal Engine 4 використовує свої передові можливості рендерингу для створення приголомшливих візуальних ефектів та захоплюючих середовищ. Він підтримує фізично заснований рендеринг (PBR) для реалістичного представлення матеріалів, динамічного освітлення і тіней, просунуті системи частинок і ефекти пост-обробки. Рушієм також включає потужний редактор матеріалів, який дозволяє художникам створювати складні мережі шейдерів [14].

Фізика та моделювання:

UE4 містить потужний фізичний рушієм, який дозволяє розробникам імітувати реальну фізику та взаємодії. Він підтримує динаміку твердого тіла, імітацію тканин і рідин, а також середовище, що руйнується. Цей фізичний рушієм забезпечує реалістичну основу для ігрових механік, таких як взаємодія об'єктів, рух персонажів та ефекти навколишнього середовища.

Аудіо та звуковий дизайн:

Звук є невід'ємною частиною розробки ігор, і UE4 надає комплексну аудіосистему. Він підтримує високоякісний просторовий звук, динамічне мікшування та обробку ефектів. Рушій також пропонує візуальний редактор звукових сигналів, що дозволяє дизайнерам створювати складну звукову поведінку та запускати звукові події на основі ігрових подій та умов.

Анімація та керування персонажами:

Unreal Engine 4 надає потужну систему анімації для створення реалістичних рухів та взаємодій персонажів. Вона підтримує скелетну анімацію, зворотну кінематику (ІК) та проміжки змішування для плавних переходів між анімаціями. Система Animation Blueprint дозволяє дизайнерам визначати складну поведінку анімації та керувати рухом персонажів за допомогою інтерфейсу на основі вузлів [14].

Багатолатформенна підтримка:

UE4 пропонує широку багатолатформну підтримку, що дозволяє розробникам орієнтуватися на широкий спектр платформ, включаючи ПК, консолі, мобільні пристрої та гарнітури віртуальної реальності (VR). Рушій надає специфічні для кожної платформи оптимізації та набори інструментів для забезпечення оптимальної продуктивності та сумісності на різних пристроях [20].

Мережеві та онлайн-функції:

Для багатокористувацьких та мережевих ігор UE4 включає мережевий фреймворк, який підтримує як клієнт-серверну, так і однорангову архітектуру. Він надає вбудовані системи реплікації та прогнозування для синхронізації ігрових станів між мережевими гравцями. Крім того, UE4 пропонує інтеграцію з популярними онлайн-сервісами, такими як матчмейкінг, таблиці лідерів та хмарні сховища [14].

Ринок та спільнота:

Unreal Engine 4 має жваву спільноту розробників, художників та ентузіастів, які підтримують його. Unreal Marketplace пропонує величезну

колекцію готових до використання ресурсів, включаючи 3D-моделі, текстури, анімацію та звукові ефекти, які можуть значно прискорити процес розробки ігор. Спільнота активно ділиться знаннями та ресурсами через форуми, навчальні посібники та офіційну документацію.

На рисунку 4.1 наведено ігрова платформа для розробки ігрового застосунку Unreal Engine 4 [10].



Рисунок 4.1 – Ігрова платформа Unreal Engine 4

Unreal Engine 4 надає розробникам ігор комплексну та багатофункціональну платформу для створення комп'ютерних ігор. Потужна система візуальних скриптів, розширені можливості рендерингу, симуляція фізики, аудіосистема та багатоплатформенна підтримка роблять його чудовим вибором як для невеликих інді-проектів, так і для великомасштабних виробництв. Завдяки підтримці спільноти та безлічі доступних ресурсів, Unreal Engine 4 пропонує надійну основу для розробки ігор та розкриває творчий потенціал розробників та художників по всьому світу.

4.2 Діаграма класів програмного забезпечення

Діаграма класів - це тип структурної діаграми в програмній інженерії та уніфікованій мові моделювання (UML), яка представляє структуру та взаємозв'язки класів у системі або програмному додатку. Вона надає огляд об'єктно-орієнтованого дизайну системи, ілюструючи класи, їхні атрибути, методи та зв'язки між ними. Основна мета діаграми класів - візуально зобразити статичну структуру системи та її компонентів, полегшуючи спілкування і розуміння між зацікавленими сторонами, включаючи розробників, дизайнерів і менеджерів проєктів [22].

На рисунку 4.2 наведено діаграма класів ігрового застосунку.

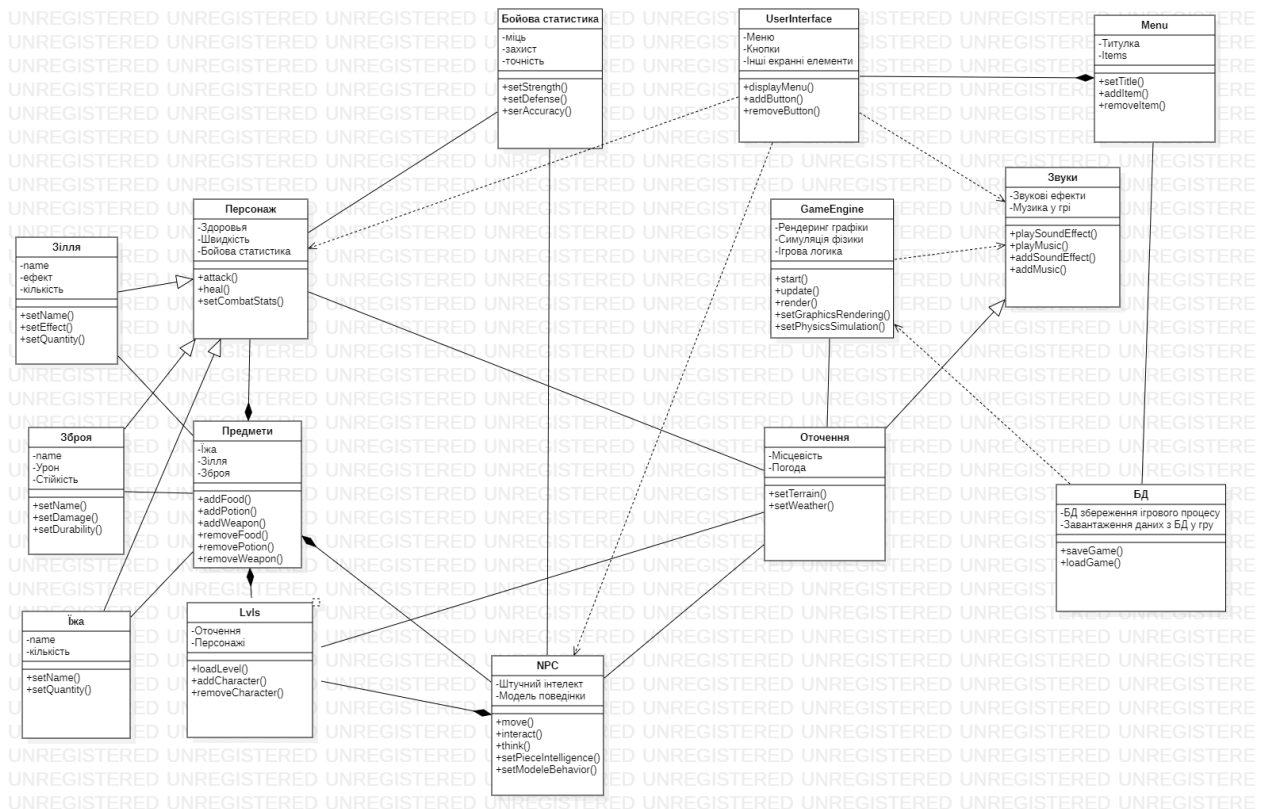


Рисунок 4.2 – Діаграма класів

Клас Персонаж має відношення композиції з класом Предмети, оскільки персонаж гравця може мати у своєму інвентарі різні предмети, такі як їжа, зілля та зброя.

Клас `Lvls` має відношення композиції з класами `NPC` та `Предмети`, оскільки рівень може містити різних персонажів та об'єкти, з якими гравець може взаємодіяти.

Клас `GameEngine` має відношення залежності з класом `Звуки`, оскільки рушієм може відтворювати звукові ефекти або музику під час ігрового процесу.

Клас `UI` має відношення композиції з класом `Меню`, оскільки інтерфейс може містити різні кнопки, на які гравець може натискати.

Клас `Database` має відношення залежності з класом `GameEngine`, оскільки рушієм може зберігати та завантажувати дані гри з бази даних.

4.3 Опис функціоналу

При вході до гри з'являється головне меню, яке містить в собі набір функцій.

На рисунку 4.3 наведено головне меню гри.



Рисунок 4.3 – Головне меню

Спочатку йде створення нової гри.

На рисунку 4.4 наведений алгоритм коду створення нової гри.



Рисунок 4.4 – Код для створення нової гри

Наступне меню налаштувань.

На рисунку 4.5 наведений алгоритм коду виклику меню налаштувань.

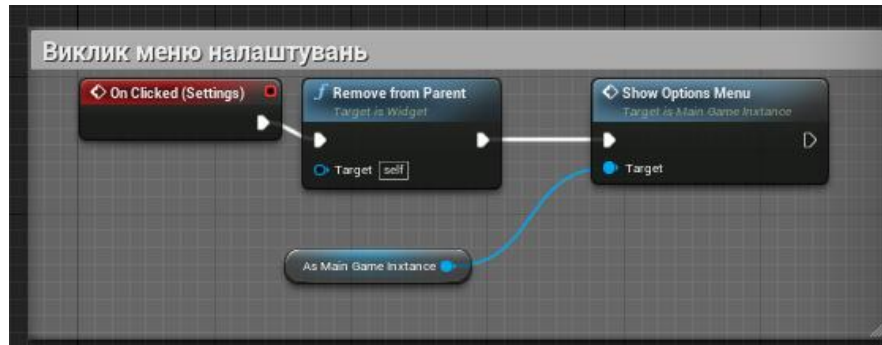


Рисунок 4.5 – Код виклику меню налаштувань

Вихід з гри.

На рисунку 4.6 наведений алгоритм виходу з гри.

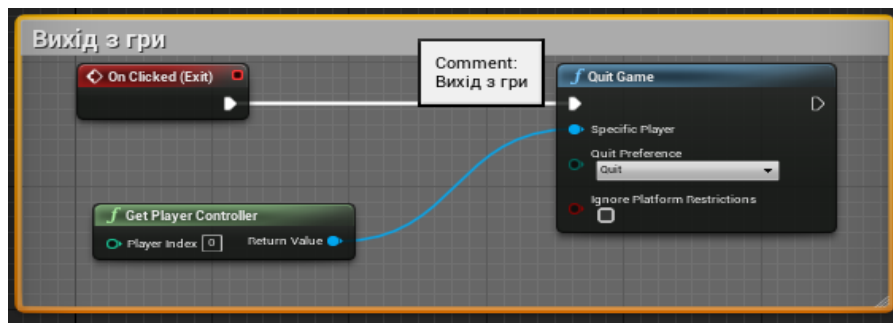


Рисунок 4.6 – Код виходу з гри

Останнє це завантаження до гри, містить в собі 3 кнопки завантаження, видалення та назад.

На рисунку 4.7 наведено меню завантаження до гри.



Рисунок 4.7 – Меню завантаження до гри

Програмний код для початку гри з завантаження.

На рисунку 4.8 наведений алгоритм початку гри з завантаження.

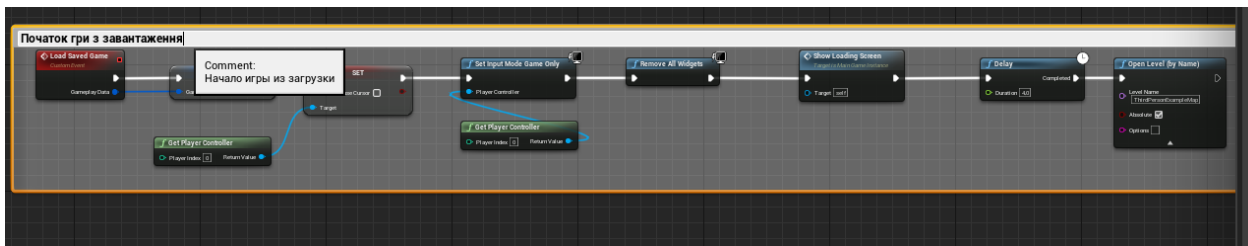


Рисунок 4.8 – Код початку гри з завантаження

Біг головного персонажа.

На рисунку 4.9 наведено демонстрація бігу головного персонажа.



Рисунок 4.9 – Демонстрація бігу

Програмний блупрінт код для бігу головного персонажа.

На рисунку 4.10 наведений алгоритм бігу головного персонажа.

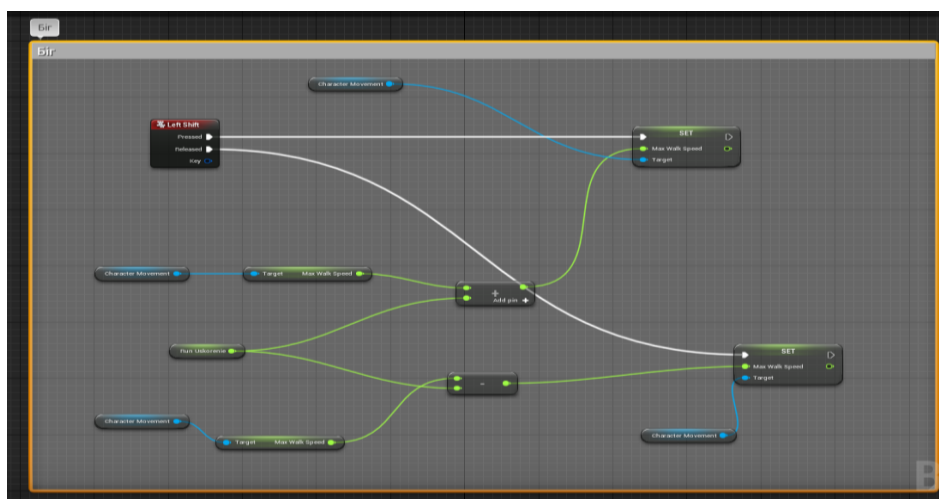


Рисунок 4.10 – Код бігу головного персонажа

Стрибок головного персонажа.

На рисунку 4.11 наведено стрибок головного персонажа.



Рисунок 4.11 – Демонстрація стрибка головного персонажа

Програмний блупрінт код для стрибку головного персонажа.

На рисунку 4.12 наведений алгоритм стрибка головного персонажа.

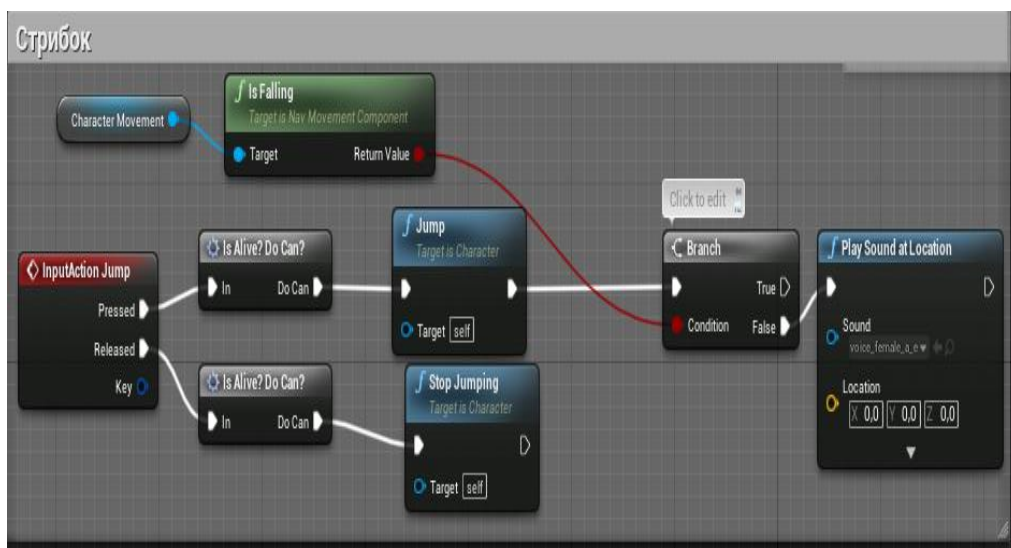


Рисунок 4.12 – Код стрибка головного персонажа

Демонстрація бойової системи.

На рисунку 4.13 наведено демонстрація бойової системи.



Рисунок 4.13 – Демонстрація бойової системи

Програмний блупрінт код для бойової системи, код містить в собі атаку з правої руки та з лівої зі шансом у 50%.

На рисунку 4.14 наведений алгоритм бойової системи.

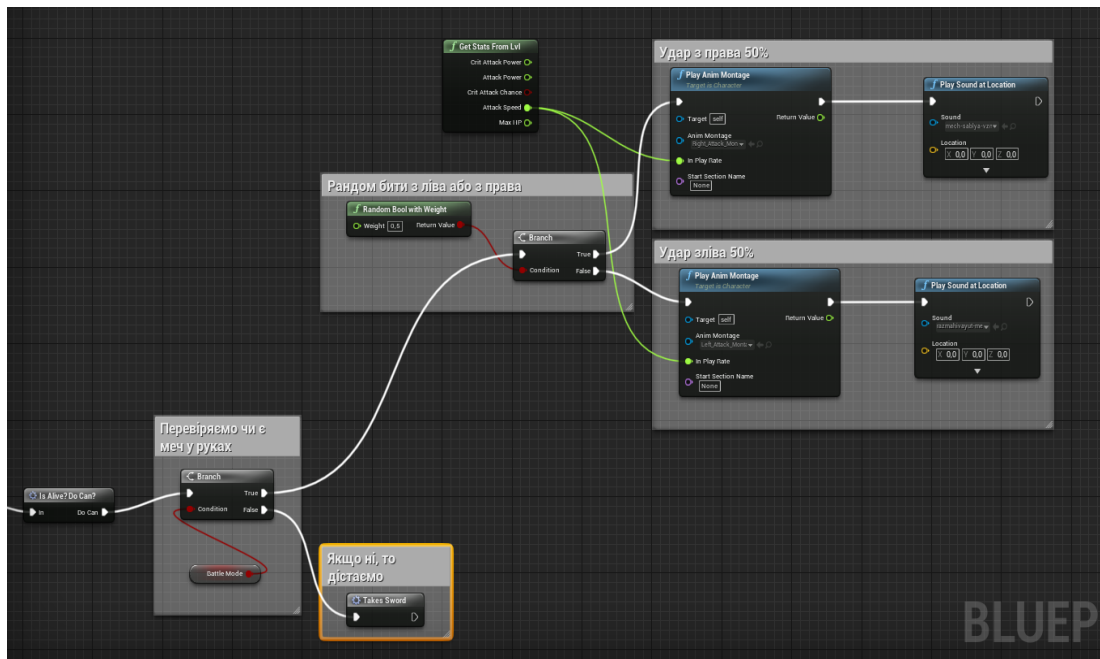


Рисунок 4.14 – Код бойової системи

Інтерфейс здоров'я, мани, досвіду головного персонажа та його поточний рівень.

На рисунку 4.15 наведено демонстрація здоров'я, мани та досвіду.

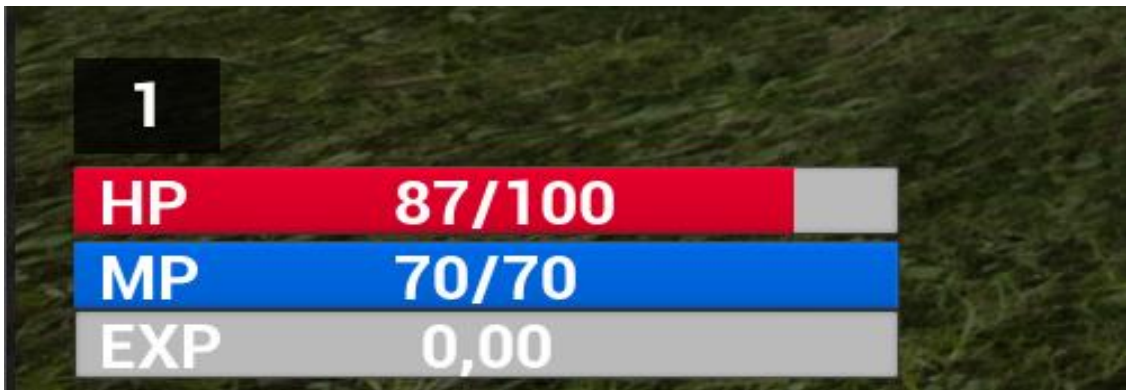


Рисунок 4.15 – Демонстрація здоров'я, мани та досвіду

Програмний блупрінт код для здоров'я та мани. Код містить в собі алгоритм максимальної кількості здоров'я, мани та при шкоді від максимальної кількості здоров'я віднімається частка та переводить у поточну кількість.

На рисунку 4.16 наведений алгоритм реалізації віджета здоров'я, мани та досвіду.

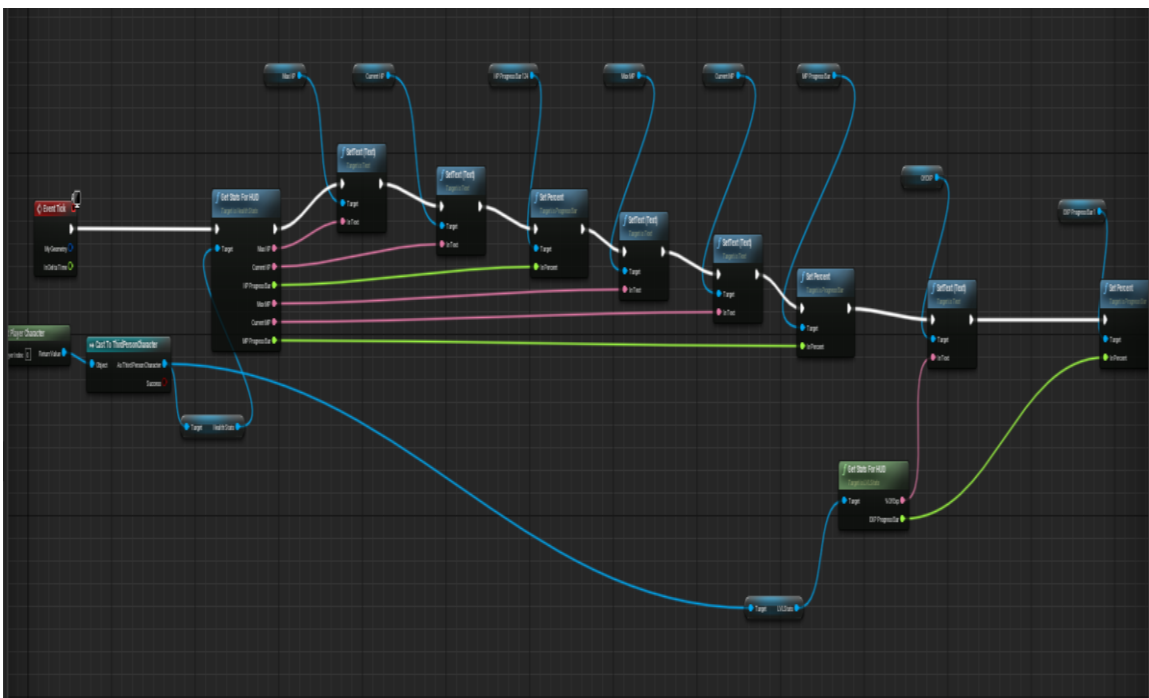


Рисунок 4.16 – Код віджета здоров'я, мани та досвіду.

Алгоритм для показу рівня головного персонажа на екрані.

На рисунку 4.17 наведений алгоритм для показу рівня.

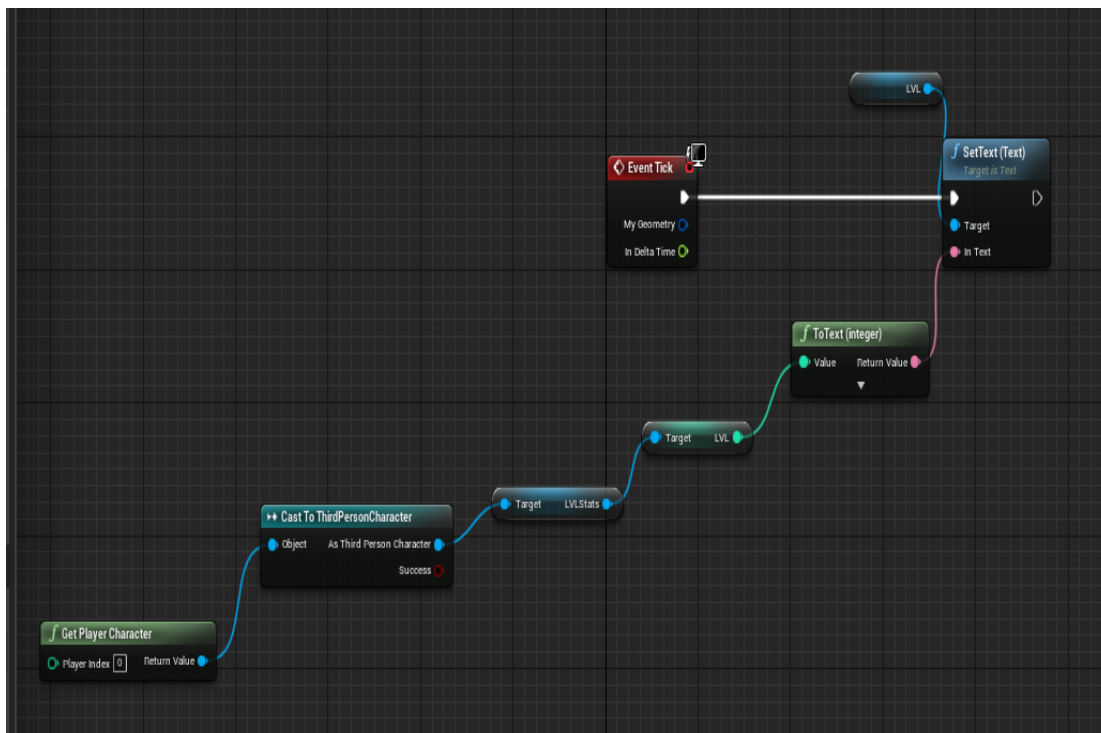


Рисунок 4.17 – Код для показу рівня

Система отримання бонусів до здоров'я, швидкості атаки та шкоди головного персонажа при отриманні нового рівня.

На рисунку 4.18 наведений алгоритм система отримання бонусів за підвищення рівня персонажа.

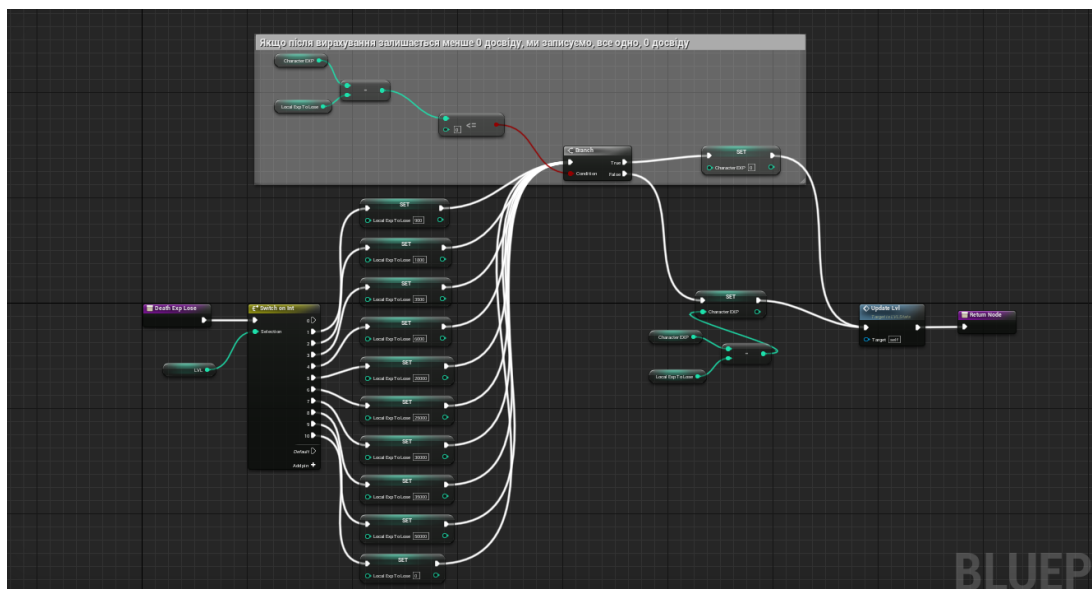


Рисунок 4.18 – Код отримання бонусів за підвищення рівня

Отримання шкоди головного персонажа від моба.

На рисунку 4.19 наведено демонстрація отримання шкоди від моба.

Ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine



Рисунок 4.19 – Демонстрація отримання шкоди від моба

Алгоритм отримання шкоди від моба.

На рисунку 4.20 наведений алгоритм отримання шкоди.

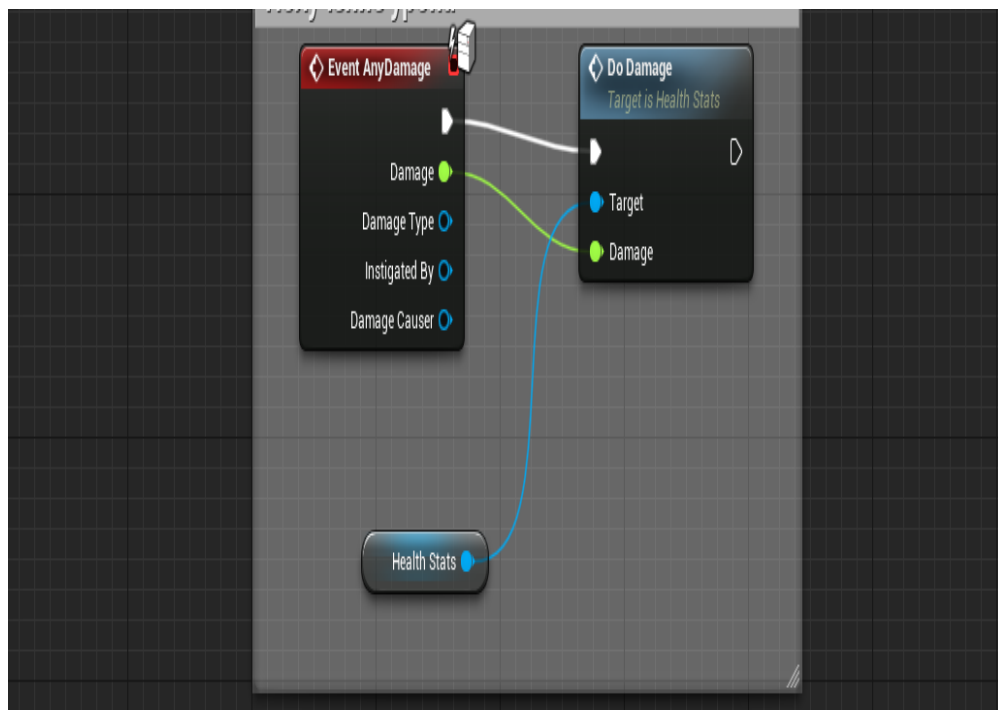


Рисунок 4.20 – Код отримання шкоди

На рисунку 4.21 наведено смерть головного персонажа.



Рисунок 4.21 – Демонстрація смерті головного персонажа

Програмний блупрінт код смерті головного персонажа та появи після натискання “Respawn in Forest”.

Анімація смерті та поява віджета для появи головного персонажа.

На рисунку 4.22 наведений алгоритм реалізації смерті головного персонажа.

Ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine

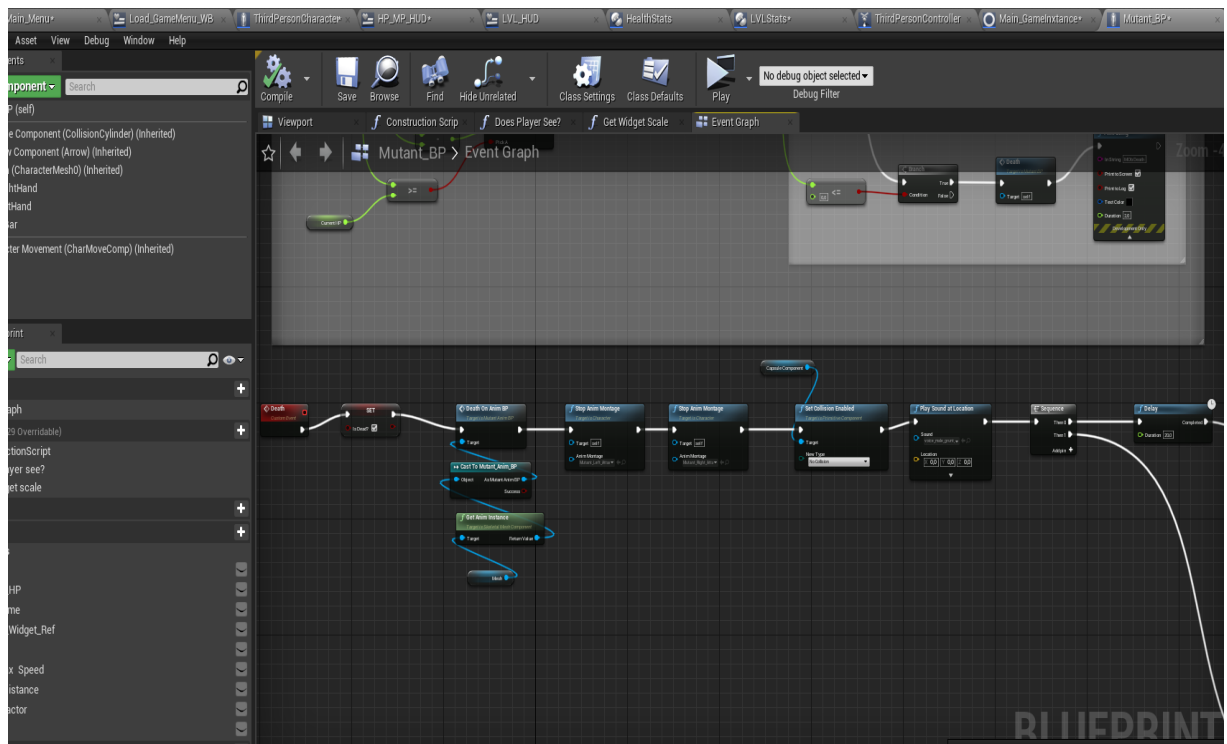


Рисунок 4.22 – Код реалізації смерті

Вигляд ворожого моба у ігровому застосунку.

На рисунку 4.23 наведено демонстрація вигляду ворожого моба.

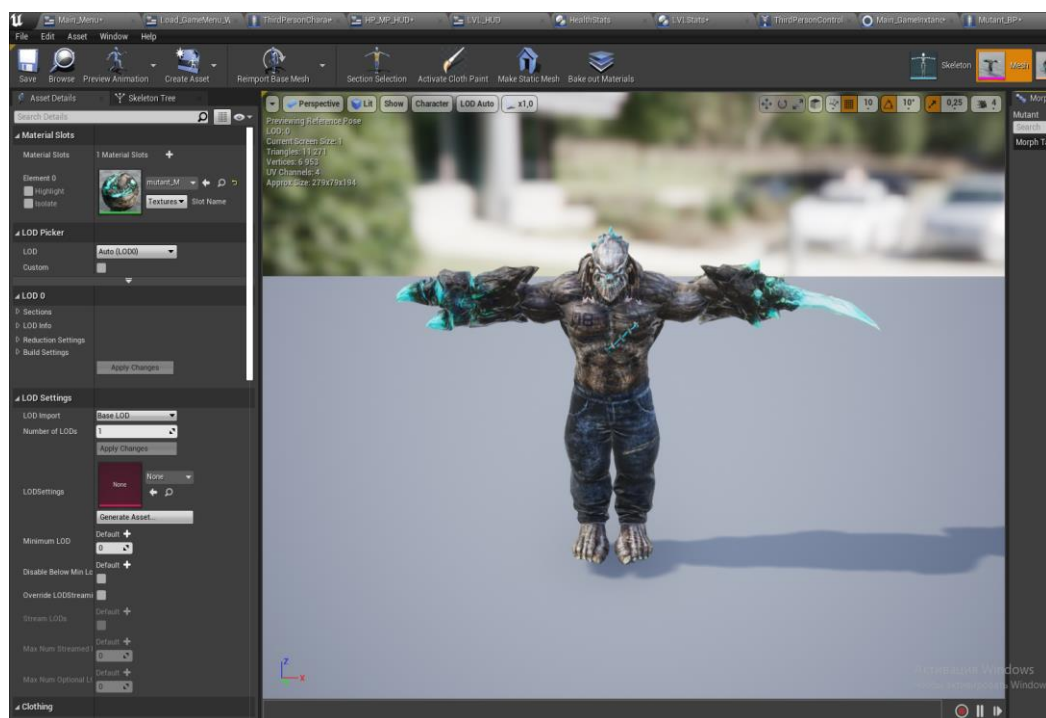


Рисунок 4.23 – Демонстрація вигляду ворожого моба

Анімація атаки лівою рукою ворожого моба в ігровому застосунку.

На рисунку 4.24 наведено анімація атаки лівою рукою.

Ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine



Рисунок 4.24 – Анімація атаки лівою рукою

Анімація атаки правою рукою ворожого моба в ігровому застосунку.

На рисунку 4.25 наведено анімація атаки правою рукою.

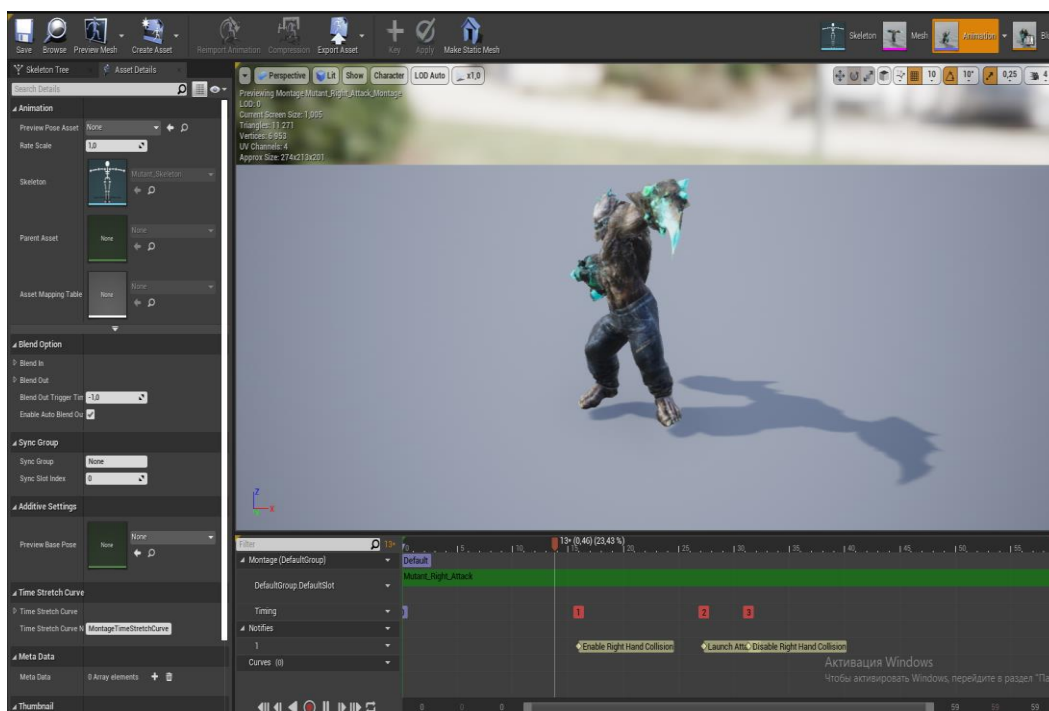


Рисунок 4.25 – Анімація атаки правою рукою

Система атаки ворожого моба, яка містить в собі відновлення швидкості після атаки, на час атаки максимальна швидкість 0.

На рисунку 4.26 наведений алгоритм коду атаки ворожого моба.

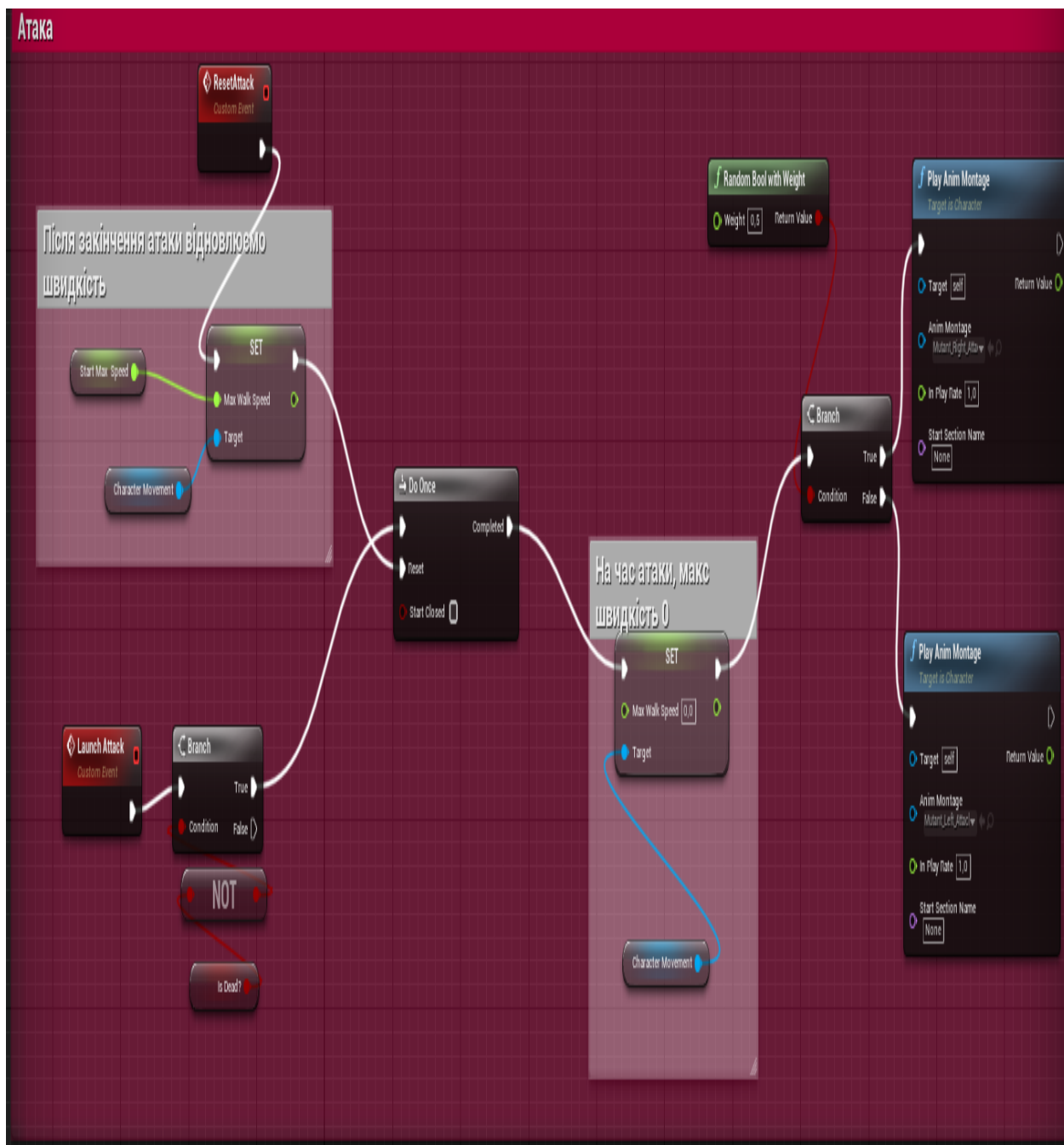


Рисунок 4.26 – Код атаки ворожого моба

Система атаки ворожого моба лівою та правою рукою з нефіксованою шкодою зі шансом у 50%, та після кожного удару ворожий моб оновлює свою атаку.

На рисунку 4.27 наведений алгоритм коду атаки ворожого моба лівою та правою рукою.

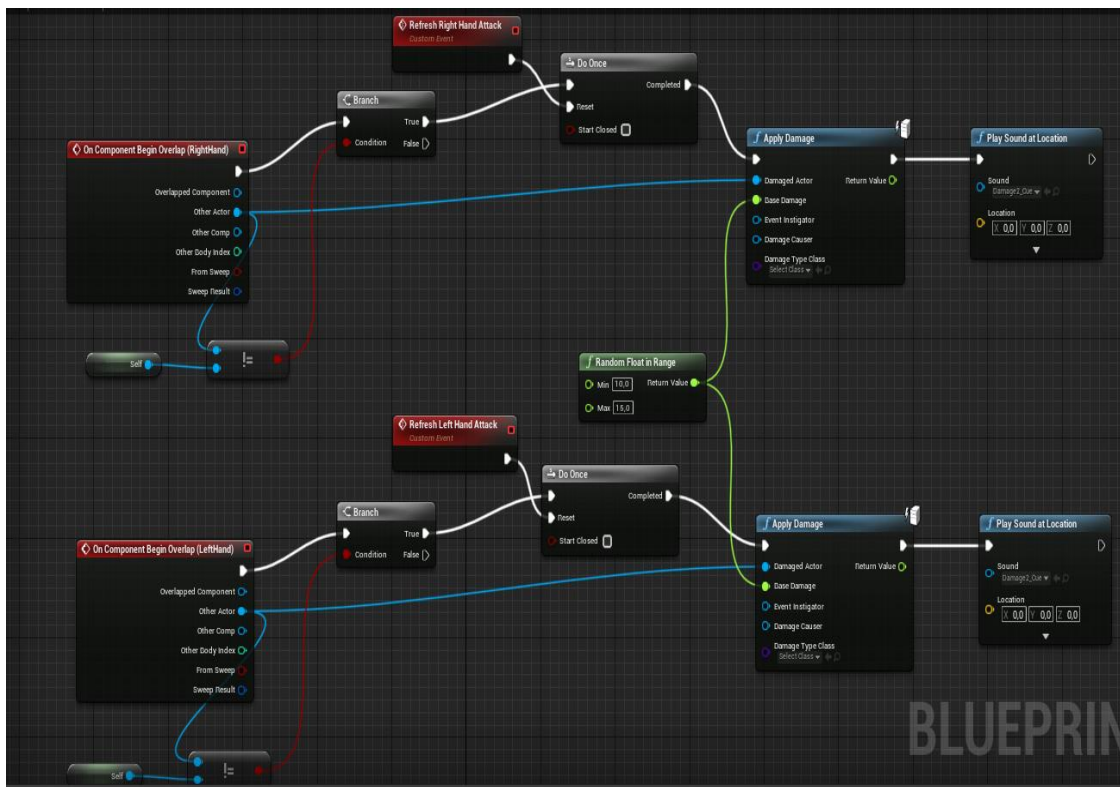


Рисунок 4.27 – Код атаки лівою та правою рукою

Демонстрація віджету здоров'я ворожих мобів.

На рисунку 4.28 наведено демонстрація віджету здоров'я ворожих мобів.

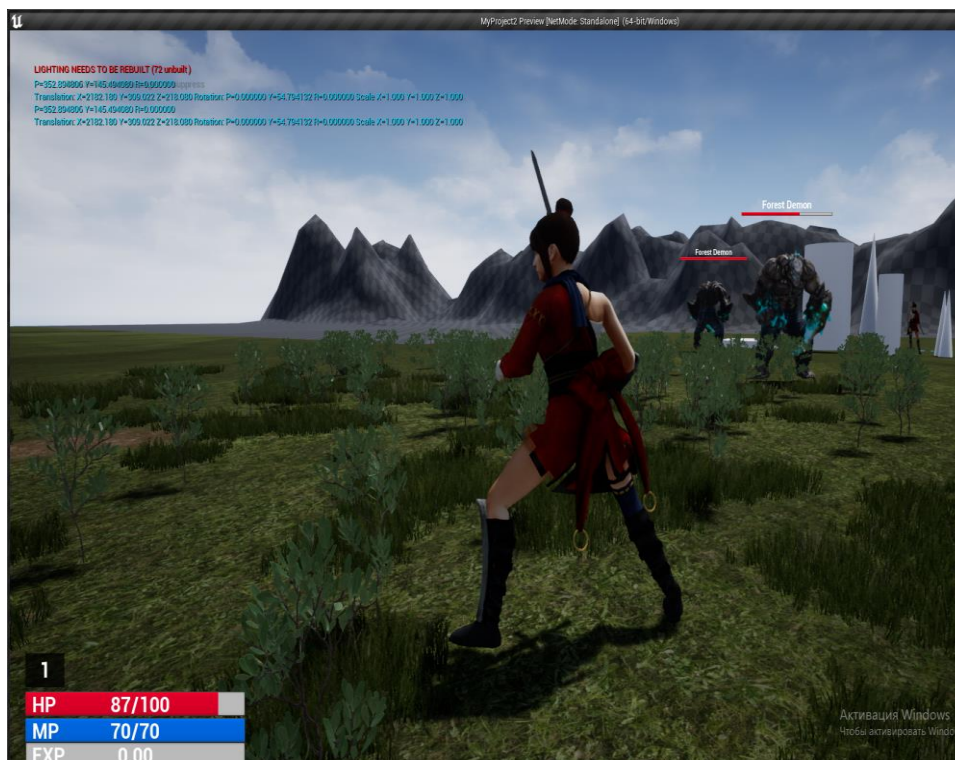


Рисунок 4.28 – Демонстрація здоров'я ворожих мобів

Програмний блупрінт код для передачі здоров'я до віджету моба та його назву.

На рисунку 4.29 наведений алгоритм коду передачі здоров'я до віджету.

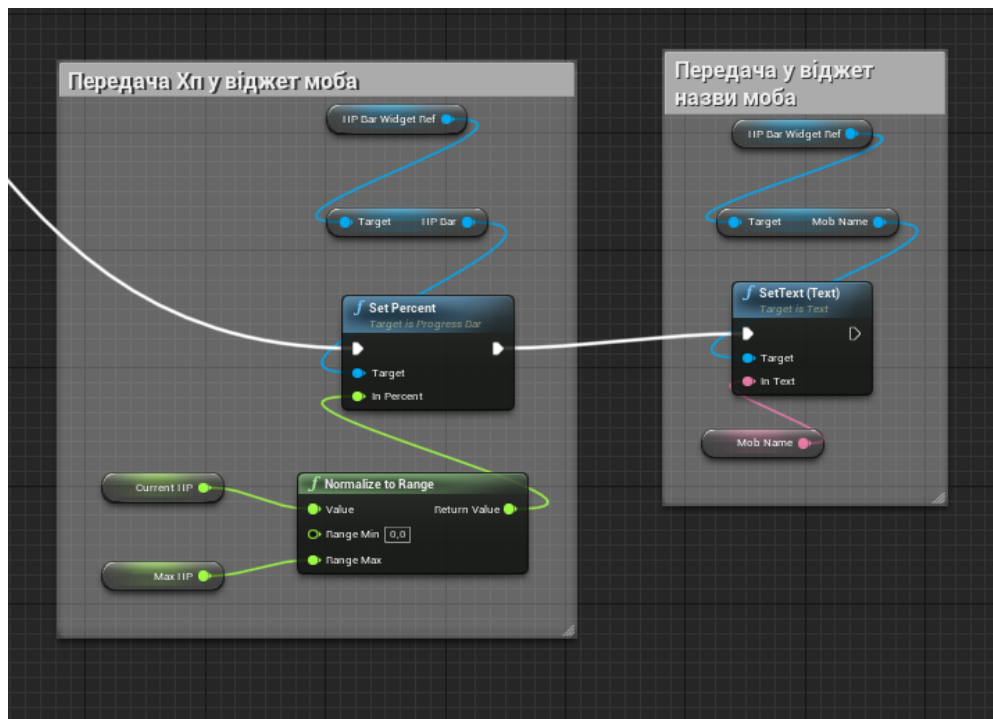


Рисунок 4.29 – Код передачі здоров'я до віджету

Демонстрація інвентаря головного персонажа.

На рисунку 4.30 наведено демонстрація інвентарю головного персонажа.

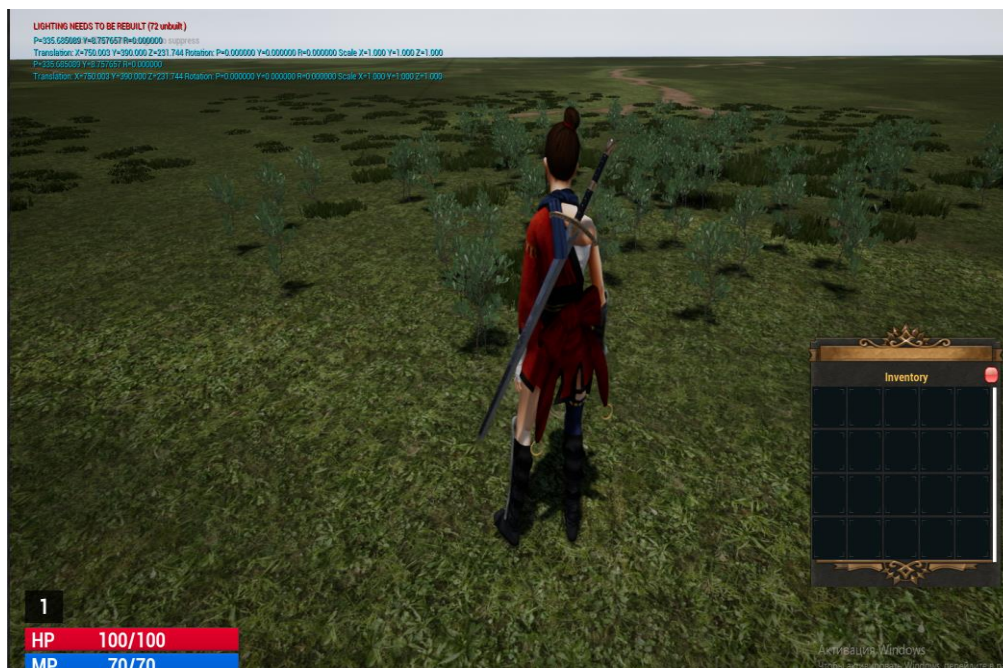


Рисунок 4.30 – Демонстрація інвентарю

Програмний блупрінт код який містить в собі створення віджету інвентаря у зміну, вимикає видимість інвентарю та додає віджет на екран гравців.

На рисунку 4.31 наведений алгоритм коду створення віджету інвентаря та виведення його на екран.

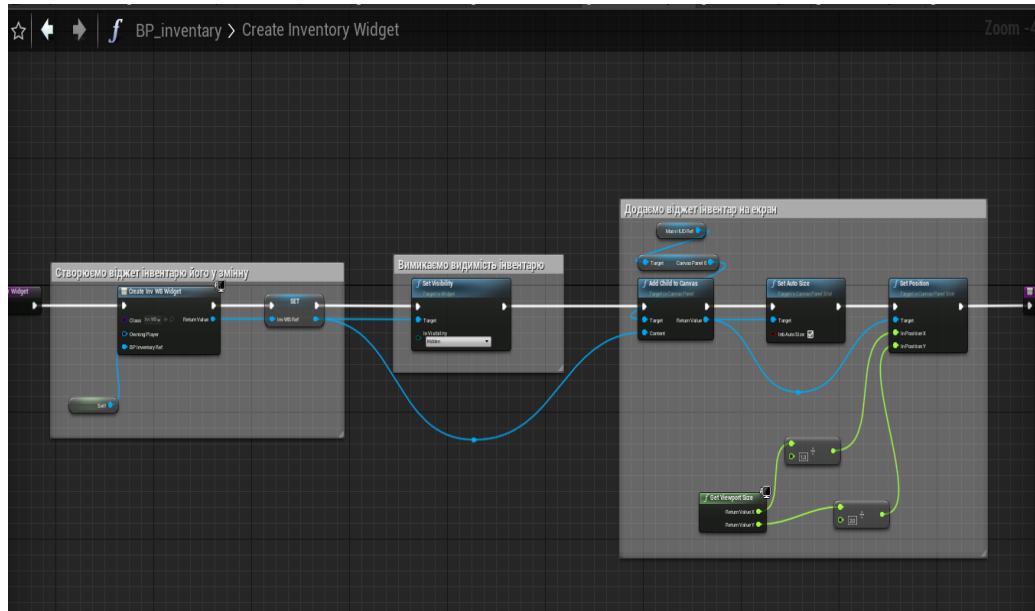


Рисунок 4.31 – Код віджету створення та виведення інвентаря на екран

Алгоритм цього коду містить в собі перевірку статусу інвентаря, увімкнення видимості інвентаря та увімкнення миші і фокусу на віджеті.

На рисунку 4.32 наведений алгоритм перевірки статусу інвентаря.

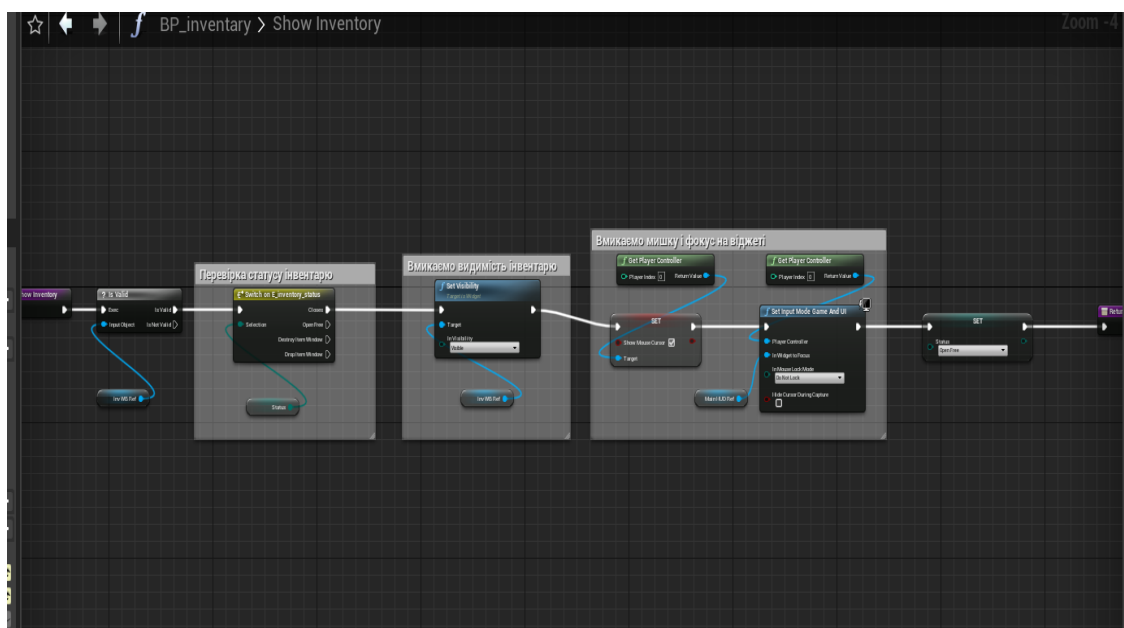
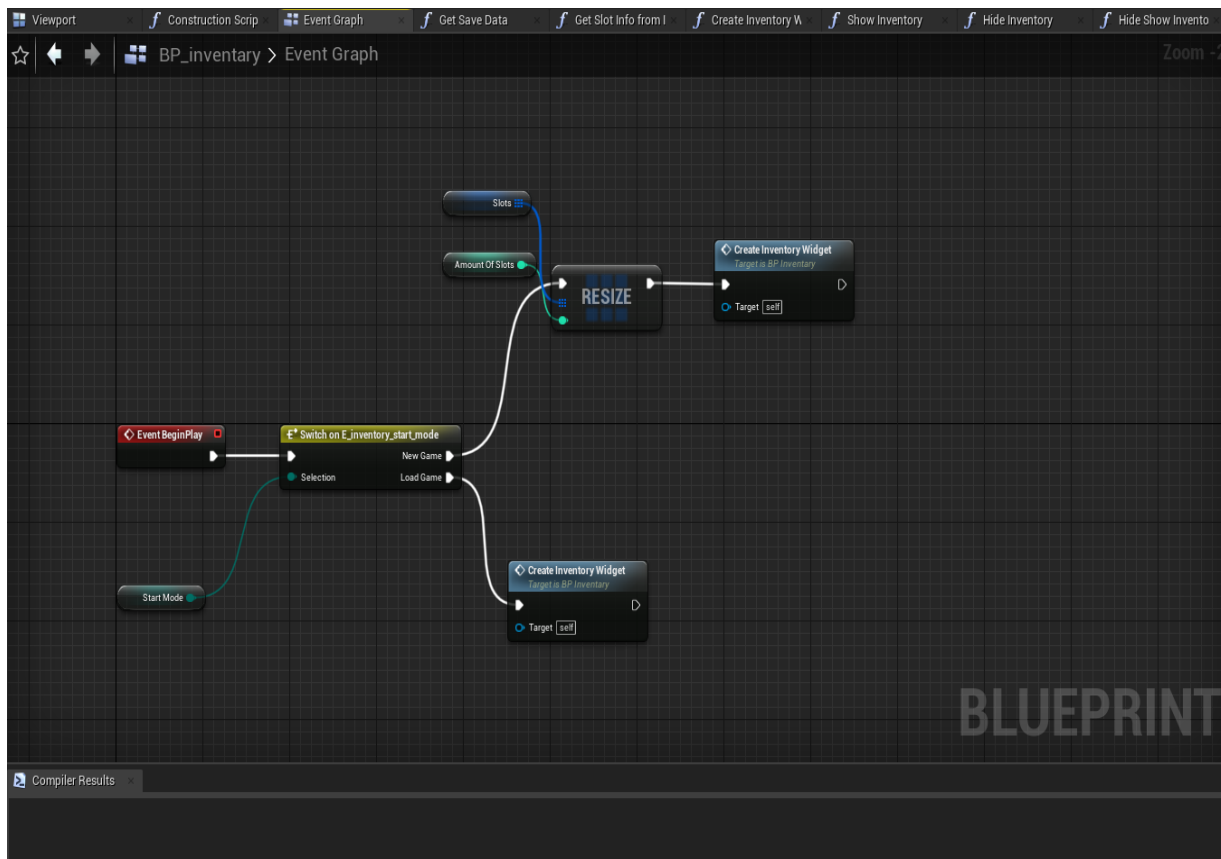


Рисунок 4.32 – Код перевірки статусу інвентаря

Ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine

Підключення всіх функцій інвентаря до ігрового застосунку.

На рисунку 4.33 наведений алгоритм підключення всіх функцій інвентаря.



На рисунку 4.33 – Код підключення всіх функцій інвентаря
Демонстрація меню паузи у грі.

На рисунку 4.34 наведено демонстрація меню паузи

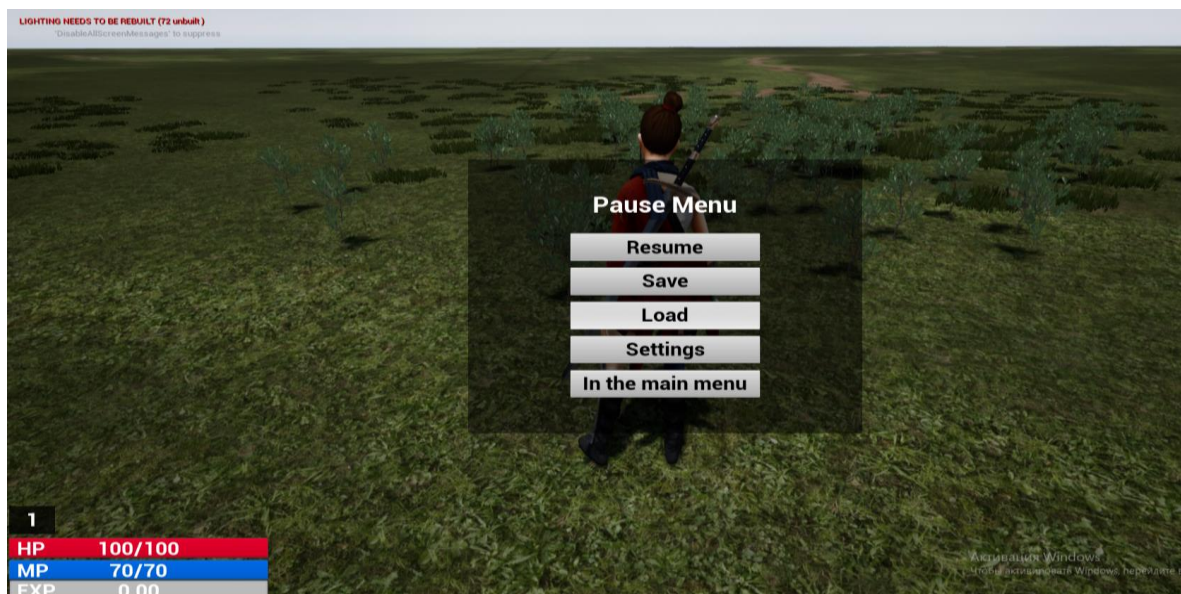


Рисунок 4.34 – Демонстрація меню паузи

Алгоритм коду меню паузи, який містить в собі відновлення гри, вихід у головне меню, меню налаштувань паузи, меню збереження гри, відкриття меню завантаження гри.

На рисунку 4.35 наведений алгоритм коду меню паузи.

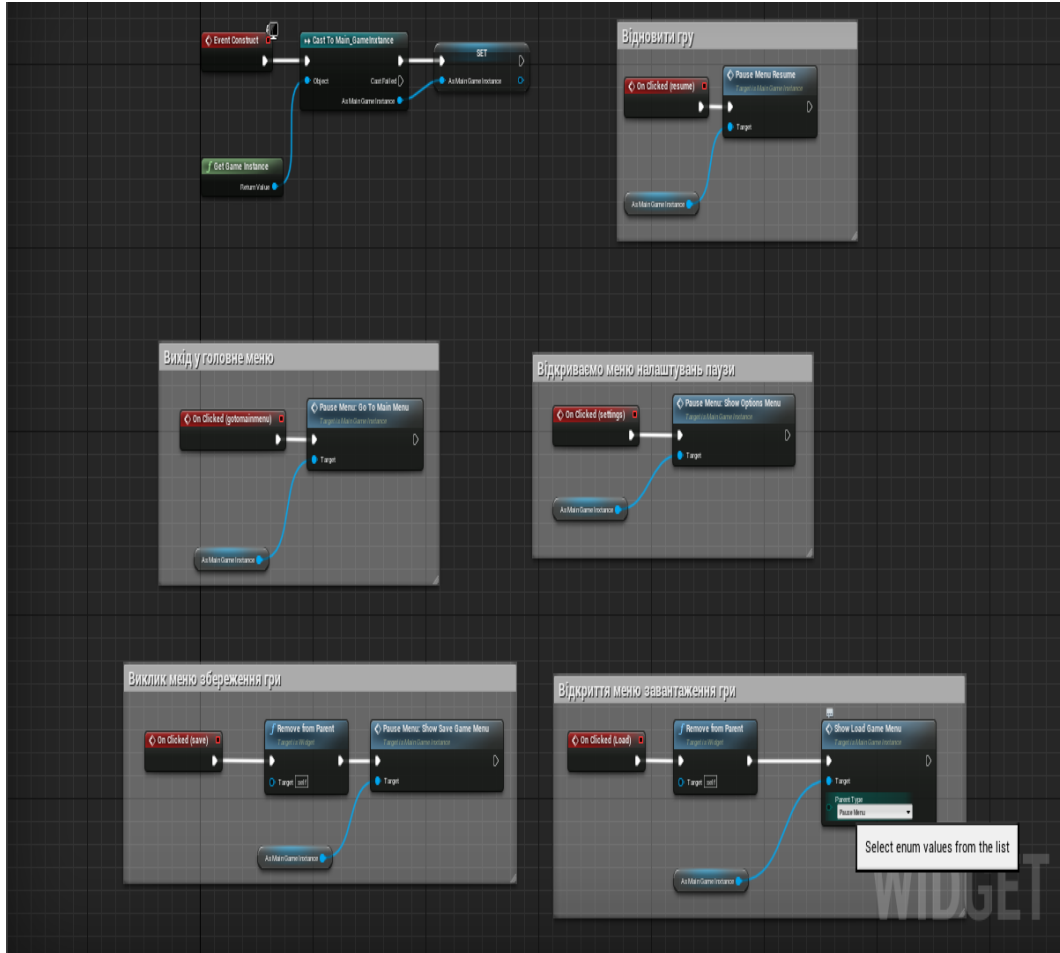


Рисунок 4.35 - Код меню паузи

4.4 Тестування комп'ютерної гри

Тестування комп'ютерної гри є процесом перевірки, оцінки та перевірки функціональності, якості та відповідності вимогам комп'ютерної гри перед її випуском на ринок або після випуску з метою виявлення та виправлення помилок, проблем та недоліків.

Тестування комп'ютерних ігор є важливою частиною процесу розробки, оскільки дозволяє виявити та виправити помилки, які можуть впливати на користувацький досвід, функціональність, стабільність та інші аспекти гри. Тестування гри включає в себе різні аспекти, такі як перевірка гри на

відповідність специфікаціям, перевірка роботи гри на різних платформах (наприклад, ПК, консолі, мобільні пристрої), перевірка функціональності гри, тестування мережевої гри та інші види тестування [18].

Тестування комп'ютерних ігор може проводитися як вручну, так і за допомогою автоматизованих інструментів тестування. Вручне тестування включає в себе виконання різних дій та сценаріїв у грі, щоб перевірити їх роботу і виявити помилки. Автоматизоване тестування використовує спеціальні програми та скрипти для виконання тестових сценаріїв і автоматичного виявлення помилок.

Основні етапи тестування комп'ютерних ігор включають планування тестування, створення тестової документації, виконання тестових сценаріїв, виявлення та документування помилок, повторне тестування після виправлення помилок, а також функціональне, продуктивне, сумісне та інші види тестування [18].

Однак, загальний процес тестування може включати наступні кроки:

1) Планування тестування: Визначення цілей, обсягу і стратегії тестування. В цьому етапі визначаються тести, які потрібно провести, встановлюються критерії успішності тестування та створюється план тестування.

2) Розробка тестової документації: Створення тестових сценаріїв, тест-кейсів та інших документів, що описують послідовність дій, які потрібно виконати під час тестування. Це можуть бути переліки функцій, які потрібно перевірити, та очікуваного результату [18].

3) Виконання тестів: Здійснення тестування згідно з розробленою тестовою документацією. Тестери виконують тестові сценарії, проводять експерименти з грою, перевіряють функціональність, стабільність, здатність до взаємодії з різними платформами, виявляють та документують помилки.

4) Документування помилок: Якщо тестери виявляють помилки, вони детально описують їх в спеціальній системі відстеження помилок. Кожна помилка повинна містити опис проблеми, кроки для відтворення,

специфікацію системи, на якій вона була виявлена, та іншу відповідну інформацію [18].

5) Повторне тестування: Після виправлення помилок розробники здійснюють повторне тестування, щоб перевірити, чи були помилки виправлені і чи виникли нові проблеми в процесі виправлення.

6) Тестування продуктивності: Випробування гри на масштабі та продуктивність для перевірки, як гра працює при великому навантаженні, зокрема в мультиплеєрному режимі або на слабшому апаратному забезпеченні.

7) Тестування сумісності: Перевірка гри на різних платформах, операційних системах та пристроях, щоб забезпечити, що гра працює належним чином із різними конфігураціями.

8) Фінальне тестування та звітність: Останній етап тестування, де проводиться повне перевірка всіх аспектів гри перед випуском. Після завершення тестування складається звіт, який містить результати тестування, виявлені проблеми та рекомендації для подальшого вдосконалення [12].

4.5 Інструкції користувача комп'ютерної гри

Початок роботи:

1. Головне меню.
2. Елементи керування.
3. Режими гри.

Ігровий процес:

1. Мета.
2. Управління гравцем.
3. Ігрова механіка.
4. Посилення та покращення.
5. Система підрахунку очок.

Налаштування гри:

1. Графіка.
2. Звук.

3. Керування.

1) Головне меню:

Після запуску гри привітає головне меню. Звідси можливо отримати доступ до різних функцій гри, включаючи запуск нової гри, завантаження збереженої гри, зміну налаштувань, доступ до підручників тощо. Використовування миші або клавіатури для навігації по пунктах меню [13].

2) Елементи керування:

Ознайомлення зі схемою керування грою, яка буде пояснена в меню налаштувань або опцій керування. Керування може включати введення з клавіатури, рухи миші або входи геймпада/контролера. Налаштуйте керування відповідно до ваших уподобань, якщо це дозволяє гра.

3) Режими гри:

Гра може пропонувати різні режими, такі як однокористувацька, багатокористувацька або мережева гра. Виберіть потрібний режим гри та дотримуйтесь підказок, щоб розпочати свою пригоду. Кожен режим може мати певні особливості та варіанти гри [12].

4) Мета:

Основна мета гри, яка може полягати у проходженні рівнів, вирішенні головоломок, перемозі над ворогами, отриманні високих результатів або досягненні певних етапів. Мета може відрізнятися залежно від жанру або режиму гри.

5) Керування гравцем:

Опанування елементів керування гравця, щоб переміщатися ігровим світом, взаємодіяти з об'єктами, виконувати дії та керувати своїм персонажем. Елементи керування можуть включати клавіші руху, кнопки дій, керування камерою та навігацію по меню. Практика в управлінні, щоб покращити свої ігрові навички [12].

6) Ігрова механіка:

Ознайомлення з ігровою механікою, таку як бойові системи, механіку розв'язання головоломок, управління ресурсами, розвиток персонажа та будь-

які унікальні функції, що сприяють ігровому процесу. Розуміння механіки допоможе вам розробляти стратегію та ефективно просуватися вперед.

8) Система підрахунку очок:

Якщо гра включає систему підрахунку очок, ознайомлено з тим, як вони заробляються. Ігри на основі балів часто винагороджують гравців за досягнення, виконання завдань за певний час або збирання предметів. Досягнути високих результатів і прагнути очолити таблицю лідерів [12].

9) Налаштування гри:

1. Графіка:

Налаштування графічних параметрів, такі як роздільна здатність, якість текстур, згладжування та візуальні ефекти, щоб оптимізувати продуктивність гри на вашому комп'ютері та пристосувати візуальні ефекти до ваших уподобань.

2. Звук:

Налаштування параметрів звуку, зокрема рівень гучності музики, звукові ефекти, голос за кадром, а також налаштуйте будь-які специфічні аудіоопції, такі як об'ємний звук або субтитри.

3. Елементи керування:

Зміна налаштувань елементів керування, відображення кнопок, чутливості або інвертування осей відповідно до ваших уподобань або для пристосування до певних пристроїв введення [12].

Висновки до 4 розділу

У четвертому розділі було розглянуто процес розробки комп'ютерної гри в жанрі ролевого екшену включає кілька ключових аспектів, серед яких вибір технологій, діаграма класів програмного забезпечення, опис функціональності, тестування та інструкції для користувача.

Було проаналізовано та обрано стек технологій реалізації комп'ютерної гри, до якого увійшли вибір технологій, яке впливає на загальний процес розробки та продуктивність гри. Важливо враховувати такі фактори, як сумісність

платформи, інструменти розробки та мови програмування, які відповідають вимогам гри. Вибираючи відповідні технології, розробники можуть забезпечити безперебійний та ефективний процес розробки, надаючи при цьому високоякісний ігровий досвід.

Розроблено діаграма класів програмного забезпечення яка слугує планом для організації та структурування кодової бази гри. Вона візуально представляє взаємозв'язки між різними класами та їхніми атрибутами і методами. Ця діаграма допомагає підтримувати модульність коду, покращує повторне використання коду та сприяє кращій співпраці між членами команди розробників. Вона забезпечує чітке розуміння архітектури гри, що полегшує реалізацію та управління різними функціями.

Запропоновано різноманітну функціональність ролевого екшену яка повинна відповідати очікуванням та вподобанням гравців. Зазвичай вона включає кастомізацію персонажа, цікавий сюжет, захоплююче середовище, бойові механіки, управління предметами, квести та інтерактивних NPC. Розробники повинні приділяти першочергову увагу створенню захопливих елементів ігрового процесу, які пропонують баланс викликів і задоволення, гарантуючи, що гра забезпечує гравцям захопливий і приємний досвід.

Проведено тестування, яке дозволяє виявити та вирішити будь-які помилки, збої чи проблеми з продуктивністю.

ВИСНОВКИ

Кваліфікаційна робота присвячена питанням дослідження та вдосконалення напрямку розробки та впровадження комп'ютерних ігрових технологій в жанрі ролевого екшену. Аналіз ролевих ігор є особливо цікавим, оскільки він дає уявлення про те, як розробляються ці типи ігор і що робить їх привабливими для гравців.

Результатом виконання кваліфікаційної роботи є функціонуючий ігровий застосунок в жанрі ролевого екшену на основі рушія Unreal Engine.

Розробка ігрового застосунку в жанрі ролевого екшену на основі рушія Unreal Engine 4 є актуальною прикладною задачею, оскільки UE4 продовжує набувати все більшого поширення у світі розробки віртуальної реальності (VR) та доповненої реальності (AR).

В процесі виконання кваліфікаційної роботи було вирішено наступні задачі:

- проаналізовані існуючі аналоги ігрових застосунків в жанрі ролевого екшену; визначено їх сильні та слабкі сторони;

визначено основні функціональні вимоги до програмного забезпечення ігрового застосунку. Специфікація вимог до ігрового програмного забезпечення є важливим етапом, оскільки підкреслює важливість планування та уваги до деталей у процесі розробки ігор;

- спроектовано та змодельовано програмне забезпечення ігрового застосунку;

- розроблено функціональні модулі ПЗ та зручний, інтуїтивно зрозумілий інтерфейс ігрового застосунку;

- проведено тестування застосунку;

- розроблено інструкцію користувача.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ігри для програмистів CodeCombat. URL: <https://my-gamezone.ua/igrydlja-programmistov/code-combat-igrat-onlajn/> (дата звернення: 06.01.2023).
2. Комп'ютерні ігри як важливий фактор формування життєвих навичок. URL: <https://osvita.ua/vnz/reports/psychology/28614/> (дата звернення: 06.11.2021).
3. Костер Р. Розробка ігор та теорія розваг. URL: <https://foxminded.ua/knygy-dlya-zanurennya-v-industriyu-igor/> (дата звернення 09.05.2023).
4. Москаль В. Р. Метод проектування веб-застосунків: пояснювальна записка до валіфікаційної роботи здобувача вищої освіти на другому (магістерському) рівні, спеціальність 123 Комп'ютерна інженерія; МОН України, Харків. нац. ун-т радіоелектроніки. Харків, 2021. 69 с.
5. Платформи для створення ігор: поняття, види платформ, рейтинг кращих, класифікація ігор, огляди, відгуки і поради фахівців. URL: <https://ukr.kagutech.com/3933450-platforms-for-creating-games-concept-types-ofplatforms-rating-of-the-best-classification-of-games-reviews-reviews-and-expertadvice> (дата звернення: 06.11.2021).
6. П'ять ігор, які допоможуть дитині програмувати. URL: <https://osvitoria.media/experience/p-yat-igor-yaki-dopomozhudytniprogramuvaty/> (дата звернення: 06.01.2023).
7. Шелл Дж. Геймдизайн. Як створити гру, в яку гратимуть усі: URL: <https://foxminded.ua/knygy-dlya-zanurennya-v-industriyu-igor/> (дата звернення 26.04.2023).
8. Creating immersive, photorealistic VR experiences with the High Definition Render Pipeline. URL: <https://blog.unity.com/technology/creatingimmersive-photorealistic-vr-experiences-with-the-high-definition-render> (дата звернення: 21.02.2023).

9. Edmonds M. Mastering Game Development with Unreal Engine 4. Packt Publishing, 2018. 356 p.
10. Frost B. Atomic Design: A Methodology for Creating Design Systems. Pittsburgh: Self-published, 2016. 200p.
11. Garcia M. Building Engaging Action RPGs with Unreal Engine: A Practical Approach. Cambridge: O'Reilly Media, 2023. 420p.
12. Goss, C. Unreal Engine and Role-Playing Action Games: Techniques and Best Practices. San Francisco: No Starch Press, 2023. 390p.
13. Kim S. Unreal Engine and Role-Playing Action Games: Techniques and Best Practices. San Francisco: No Starch Press, 2023. 390p.
14. Lee J., Doran J. P., Misra N. Unreal Engine: Game Development from A to Z. Packt Publishing, 2016. 837 p.
15. Lee J. Learning Unreal Engine Game Development. Packt Publishing, 2016. 274 p.
16. Moniem M. A. Mastering Unreal Engine 4.X. Packt Publishing, 2016. 384 p.
17. Patel R. Action RPG Development with Unreal Engine: From Concept to Launch. London: CRC Press, 2023. 410p.
18. Sewell B. Blueprints Visual Scripting for Unreal Engine. Packt Publishing, 2015. 188 p.
19. Tavakkoli A. Game Development and Simulation with Unreal Technology. Peters/CRC Press, 2018. 736 p.
20. Thompson J. Mastering Unreal Engine: Creating Immersive Role-Playing Action Games. New York: Packt Publishing, 2023. 450p.
21. Unreal Engine 4 розробка VR гри, введення. URL: https://gamemod.ua/story/unreal_engine_4_rozrobka_3d__vr_igr_1_vedenya_v_ue_4_i_pershiy_skript_na_blueprint (дата звернення: 21.02.2023).
22. Valcasara N. Unreal Engine Game Development Blueprints. Packt Publishing, 2015. 352 p.

23. Weilkiens T., Lamm J., Roth S., Walker M. Model-Based System Architecture A. Comprehensive Guide to UML and SysML. Hoboken: Wiley, 2016. 400p.
24. Williams L. Unreal Engine for Role-Playing Action Games: A. Comprehensive Guide. Boston: Addison-Wesley, 2023. 375p.