

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

**«Комп'ютерна гра у жанрі метроїдванія “Dark
Watcher” - сюжетні кінематографічні сцени,
система налаштувань»**

Виконала студентка групи ІПЗ-19-1
БУХАНЕВИЧ Оксана Андріївна

Керівник роботи:
МОРОЗОВ Андрій Васильович

Рецензент:
БАКАЛЮК Тетяна Анатоліївна

Житомир – 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

В.о. зав. кафедри інженерії
програмного забезпечення
Андрій МОРОЗОВ
«14» лютого 2023 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **БУХАНЕВИЧ Оксана Андріївна**

Керівник роботи: **МОРОЗОВ Андрій Васильович**

Тема роботи: **«Комп'ютерна гра у жанрі метроїдванія “Dark Watcher” - сюжетні кінематографічні сцени, система налаштувань»,**

затверджена Наказом закладу вищої освіти від «14» лютого 2023 р. №70с

Вихідні дані для роботи: **Об'єктом дослідження є комп'ютерна гра “Dark Watcher” у жанрі метроїдванія. Предметом дослідження є сюжетні кінематографічні сцени та система налаштувань, які використовуються у грі.**

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Морозов А.В.	14.02.2023р.	14.02.2023р.
2	Морозов А.В.	14.02.2023р.	14.02.2023р.
3	Морозов А.В.	14.02.2023р.	14.02.2023р.

Календарний план

№ з/п	Назва етапів випускної кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка задачі. Пошук, огляд та аналіз аналогічних розробок. Формулювання технічного завдання. Опрацювання літературних джерел.	02 березня 2023 – 05 квітня 2023	
2	Проектування структури системи	6 квітня 2023 – 12 квітня 2023	
3	Написання програмного коду	13 квітня 2023 – 26 квітня 2023	
4	Тестування	27 квітня 2023 – 04 травня 2023	
5	Підготовка пояснювальної записки	5 травня 2023 – 09 червня 2023	

РЕФЕРАТ

Випускна робота бакалавра складається з аналізу процесу створення сюжетних катсцен та оптимізації параметрів у комп'ютерній грі у жанрі метроїдванія з метою поліпшення геймплею та збільшення емоційного занурення гравця. Пояснювальна записка до випускної роботи містить 79 сторінок, 20 ілюстрацій та 19 таблиць.

Метою роботи є створення налаштувань та сюжетних катсцен, включаючи використання діалогів, кінематографічних ефектів, анімації та взаємодії з головними персонажами у комп'ютерної гри у жанрі метроїдванія за допомогою рушія Unity.

У роботі визначено основні цілі та завдання, спрямовані на розробку системи. Проведено аналіз існуючих аналогів розробленої системи. Представлені сценарії роботи програмного комплексу, загальна структура програмного комплексу, опис алгоритмів взаємодії окремих модулів системи та об'єктна структура системи у вигляді діаграми класів. Крім того, наведені результати тестування системи, що дозволяють оцінити її функціональність та ефективність.

КЛЮЧОВІ СЛОВА: МЕТРОЇДВАНІЯ, СЮЖЕТНІ КАТСЦЕНИ, НАЛАШТУВАННЯ, АНІМАЦІЯ, КОМП'ЮТЕРНА ГРА.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		4

ABSTRACT

Bachelor's graduate work consists of an analysis of the process of creating narrative cutscenes and optimizing parameters in a computer game of the Metroidvania genre, aiming to improve gameplay and enhance player immersion. The explanatory note for the thesis contains 79 pages, 20 illustrations, and 19 tables.

The goal of the graduate work is to develop settings and narrative cutscenes, including the use of dialogues, cinematic effects, animation, and interaction with main characters in a computer game of the Metroidvania genre using the Unity engine.

The work defines the main goals and objectives directed towards the development of the system. An analysis of existing analogs of the developed system has been conducted. The scenarios of the software complex operation, the overall structure of the software complex, the description of interaction algorithms of individual system modules, and the object structure of the system in the form of class diagrams are presented. Additionally, the results of system testing are provided, allowing for an evaluation of its functionality and effectiveness.

KEYWORDS: METROIDVANIA, NARRATIVE CUTSCENES, CUSTOMIZATION, ANIMATION, COMPUTER GAME.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		5

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНОЇ ГРИ У ЖАНРІ МЕТРОЇДВАНІЯ.....	11
1.1 Постановка задачі	11
1.2 Аналіз аналогів програмного продукту.....	13
1.3 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення.....	21
Висновки до розділу 1	29
РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	31
2.1 Визначення варіантів використання та об'єктно-орієнтованої структури системи.....	31
2.2 Розробка подій та системи анімацій. Системні налаштування.....	39
2.3 Проектування та реалізація алгоритмів роботи системи.....	43
2.4 Реалізація ігрового додатку	52
Висновки до розділу 2	61
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ГРОЮ В ЖАНРІ МЕТРОЇДВАНІЯ “DARK WATCHER”	62
3.1 Порядок встановлення та налаштування системи	62
3.2 Структура інтерфейсу. Інтерфейс та порядок роботи з системою	63
3.3 Тестування роботи системи.....	69
Висновки до розділу 3	73
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	75
ДОДАТКИ.....	77

ВСТУП

Сьогодні комп'ютерні ігри користуються значною популярністю серед великої кількості людей. Безліч ігор різних жанрів, для різних платформ та цільової аудиторії існує на ринку, проте кожна з них по своєму унікальна та неповторна, несе в собі різний досвід та емоції. Гравці завжди будуть прагнути чогось нового та кращого, тому завжди буде попит на унікальні, вдосконалені та креативні ідеї.

Індустрія комп'ютерних ігор постійно еволюціонує та надихає гравців з усього світу на захоплюючі пригоди у віртуальних світах. Одним з найпопулярніших жанрів серед геймерської спільноти є метроїдванія, який поєднує в собі елементи пригодництва, екшну та головоломок.

Сучасні технології розвитку графіки, штучного інтелекту та взаємодії з гравцем відкривають безмежні можливості для створення непередбачуваних сюжетів, які спонукають гравців перемагати складних суперників та розгадувати загадки, які розташовані по усьому світу гри. Жанр метроїдванія відзначається непередбачуваністю, високим рівнем складності та можливістю відкриття нових областей та можливостей по мірі прогресу гравця. Завдяки прогресу в галузі віртуальної реальності та цікавого геймплею, комп'ютерні ігри можуть створювати неймовірні враження та переносити гравця у фантастичні світи, де він стає активним учасником захоплюючої пригоди.

Під час діалогу у грі, в залежності від сюжету, відображаються сюжетні катсцени, які надають гравцеві можливість не тільки насолоджуватися захоплюючою історією, але й впливати на розвиток сюжету та досліджувати глибинні моменти віртуального світу. Це сприяє створенню індивідуального геймплейного досвіду та розширює можливості самовираження гравця.

Значення сюжетних катсцен у комп'ютерних іграх полягає в їхній здатності створювати захоплюючий досвід для гравців та поглиблювати зацікавленість до віртуального світу гри. Сюжет відіграє важливу роль у привабливості гри, відчутті належності до персонажа та стимулює гравця до подальшого прогресу.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		7

Сюжетні катсцени можуть бути представлені у формі відео роликів, діалогів, епізодичних подій або взаємодії з неігровими персонажами. Вони розкривають історію, визначають мету гравця та надають контекст для подальших подій у грі.

Важливість сюжетних катсцен полягає в наступних аспектах:

- гра з цікавим та захоплюючим сюжетом привертає увагу гравців, спонукає їх бути активними учасниками віртуального світу. Сюжетні катсцени заохочують гравців вивчати історію гри, розгадувати загадки та виконувати завдання для продовження сюжету;
- захопливий сюжет може викликати емоції у гравця, такі як радість, смуток, напруга або захват. Гравець стає зацікавленим у долі персонажів та розвитку подій, що спонукає його до продовження гри для виявлення подальших сюжетних обертів;
- сюжетні катсцени визначають мету гравця та ставлять перед ним завдання, що спонукають до руху вперед. Вони надають контекст і ціль для подальших випробувань, пошуку артефактів чи розв'язання головоломок.

Також, сюжетні катсцени можуть мати певні характеристики та особливі моменти, у які вони доречно розповідають про ту чи іншу подію, завдяки якій гравець інтуїтивно розуміє що саме відбувається у грі.

Вони можуть мати наступні характеристики:

- сюжетні катсцени часто вражають гравців своєю величезністю та шаленими подіями. Це можуть бути епічні битви, руйнування масштабних об'єктів, захоплюючі переслідування або моменти підвищеної напруги. Ці катсцени створюють візуально захоплюючий досвід і роблять гру більш насиченою;
- Розкриття історії та персонажів: сюжетні катсцени допомагають гравцеві краще розуміти історію гри і глибше зануритися у світ гри. Вони розкривають деталі про минуле персонажів, їхні мотивації та цілі, а також розгортають ключові події сюжету. Ці катсцени впливають на емоційну зв'язок гравця з персонажами та сприяють розвитку сюжету;

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		8

- Забезпечення переходу між рівнями або локаціями: Сюжетні катсцени часто використовуються для плавного переходу між рівнями або локаціями гри. Вони можуть включати переходи через відеоролики, переміщення героя або подорожі з використанням транспорту. Ці катсцени слугують засобом збереження безперервності гри та зберігання настрою гравця;
- Відтворення важливих моментів: Сюжетні катсцени можуть слугувати для відтворення важливих моментів, які впливають на дальший розвиток сюжету або персонажів. Наприклад, це можуть бути сцени, де гравець приймає важливі рішення, зіштовхується з ключовими персонажами або вступає в битву з головними антагоністами. Ці катсцени роблять гравця активним учасником подій та впливають на подальші перипетії;

Мета випускної роботи полягає в демонстрації роботи кінематографічних сюжетів, їх взаємодію з гравцем, пересування та певні позиції супротивників, та керування налаштуваннями у грі, як вони впливають на гру та досвід гравця.

Мета даного диплому визначає ряд завдань, які потрібно виконати:

- вивчення основних характеристик та особливостей жанру, аналіз ігор у цьому жанрі для розуміння ключових елементів;
- обґрунтування та вибір засобів реалізації продукту;
- визначення і налаштування геймплейних елементів, таких як механіки переміщення, бойова система, прогрес та розвиток персонажа;
- розробка та імплементація головних геймплейних елементів, таких як дослідження локацій, головоломки, битви з ворогами;
- створення зрозумілих сюжетних катсцен, які передають ключові моменти сюжету та взаємодіють з геймплеєм;
- створення налаштувань для гнучкості та адаптованості гри
- проведення тестування, виявлення та виправлення помилок, налагодження геймплею.

Об'єктом дослідження ігровий рушій Unity та його можливості для розробки.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		9

Предметом дослідження є методи та підходи до створення ефективних сюжетних катсцен та налаштувань.

В роботі було використано методи літературного аналізу, які стосуються комп'ютерних ігор у жанрі метроїдванія, розробки концепції гри, створення сюжетних катсцен та налаштувань, імплементації геймплейних механік та функціоналу, використання середовища розробки Unity, редакторів графіки, обробки та аналізу отриманих даних, об'єктно-орієнтованого програмування за допомогою C# для реалізації ігрового контенту.

Розроблена гра буде використовуватися як розважальний сегмент геймігової індустрії та буде доступна для користувачів платформи Steam та інших популярних геймінгових платформ.

Пояснювальна записка до випускної роботи містить 79 сторінок, 20 ілюстрацій та 19 таблиць, список літературних джерел містить 16 найменувань.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		10

РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНОЇ ГРИ У ЖАНРІ МЕТРОЇДВАННЯ

1.1 Постановка задачі

Основною мета даної роботи полягає у демонстрації сюжетних катсцен під час ігрового процесу у грі “Dark Watcher”, через які гравець більш чітко розуміє події сюжету та понурюється у всесвіт гри. Також для поліпшення ігрового досвіду, робота включатиме в себе розробку налаштувань, які надають можливість індивідуалізувати гру згідно з вподобаннями кожного гравця.

У випускній роботі необхідно розробити комп'ютерну гру, у якій головний герой буде долати перешкоди, досліджувати локації, змагатися з суперниками та інше. Для реалізації поставленого завдання необхідно реалізувати такі етапи:

1. Розробка сюжетних катсцен
 - 1.1. Аналіз сюжетних елементів
 - 1.1.1. Вивчення і аналіз існуючих ігор у жанрі метроїдванія
 - 1.1.2. Визначення основних сюжетних елементів
 - 1.2. Створення концепції сюжету
 - 1.2.1. Визначення цілей головного героя
 - 1.2.2. Розробка основних подій та ключових моментів сюжету
 - 1.3. Визначення сценарію катсцен
 - 1.3.1. Розробка детального сценарію для кожної сюжетної катсцени
 - 1.3.2. Опис подій, діалогів та взаємодії персонажів
 - 1.4. Розробка локацій та об'єктів
 - 1.4.1. Створення візуальних елементів для сценаріїв
 - 1.4.2. Розробка унікальних локацій
 - 1.4.3. Створення об'єктів, анімацій та ефектів
 - 1.5. Імплементация катсцен
 - 1.5.1. Реалізація сценаріїв та сюжетних катсцен у грі

1.5.2. Використання відповідних інструментів розробки, програмування та графіки

1.6. Тестування та вдосконалення

1.6.1. Проведення тестування сюжетних катсцен

1.6.2. Виявлення та виправлення помилок

1.6.3. Поліпшення геймплею та балансу сценарію

1.7. Інтеграція і оцінка результатів

1.7.1. Інтеграція сюжетних катсцен у загальний геймплей гри

1.7.2. Оцінка впливу сюжетних катсцен на ігровий досвід та емоційну привабливість

2. Розробка налаштувань

2.1. Аналіз налаштувань у комп'ютерних іграх

2.1.1. Вивчення і аналіз налаштувань в існуючих комп'ютерних іграх

2.1.2. Визначення основних аспектів налаштувань, їх вплив на геймплей та користувацький досвід

2.2. Визначення можливостей налаштувань у грі

2.2.1. Визначення параметрів та елементів, які можна налаштувати

2.2.1.1. Графічні налаштування

2.2.1.2. Звукові налаштування

2.2.1.3. Контролі та налаштування інтерфейсу

2.2.1.4. Налаштування складності гри

2.2.1.5. Налаштування геймплею

2.2.2. Розробка концепції налаштувань для поліпшення геймплею

2.3. Реалізація налаштувань у грі

2.3.1. Розробка інтерфейсу налаштувань для користувачів

2.3.1.1. Дизайн інтерфейсу налаштувань

2.3.1.2. Реалізація взаємодії з користувачем

2.3.1.3. Збереження та завантаження налаштувань

2.3.2. Імплементация налаштувань та їх впровадження у гру

2.3.2.1. Інтеграція налаштувань з основним ігровим двигуном

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		12

- 2.3.2.2. Передача та збереження налаштувань у грі
- 2.3.2.3. Впровадження змін налаштувань у реальному часі
- 2.4. Тестування та вдосконалення налаштувань
 - 2.4.1. Проведення тестування функціональності налаштувань
 - 2.4.1.1. Перевірка працездатності налаштувань
 - 2.4.1.2. Оцінка впливу налаштувань на ігровий процес
 - 2.4.2. Виявлення та виправлення помилок у налаштуваннях
 - 2.4.2.1. Підвищення ефективності налаштувань для різних систем
 - 2.4.3. Оптимізація налаштувань для покращення користувацького досвіду
- 2.5. Інтеграція і оцінка результатів
 - 2.5.1. Інтеграція налаштувань у загальний ігровий процес
 - 2.5.1.1. Застосування налаштувань у геймплеї
 - 2.5.1.2. Вплив налаштувань на графіку, звук та інтерфейс гри
 - 2.5.2. Оцінка впливу налаштувань на геймплей та користувацький досвід

1.2 Аналіз аналогів програмного продукту

На даний момент існує багато комп'ютерних ігор у жанрі метроїдванія та кожна має індивідуальні особливості та різні механіки. Однак вони поєднують в собі елементи платформного екшну, дослідницького геймплею та складних головоломок, створюючи неповторний світ, який вабить гравців своєю загадковістю та величезними просторами для дослідження. Назва жанру "метроїдванія" походить від успішної серії ігор під назвою "Metroid", що була випущена компанією Nintendo у 1986 році.

Головна особливість ігор у жанрі метроїдванія полягає в їх нелінійній структурі та відкритому світі. Гравцеві надається велика свобода дій, де він може вільно переміщатися по різних локаціях, знаходити та розблоковувати нові області, знаходити та використовувати різноманітні предмети та здібності. У таких іграх часто

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		13

зустрічаються складні головоломки та загадки, які гравець повинен розв'язати, щоб продовжувати свою подорож.

Розглянемо деякі аналоги таких програмних продуктів, створених професіоналами та мають немалу популярність:

1. INMOST

Даний аналог є атмосферним, заснованим на історії пазл-платформер, який розповідає про трьох гравців, пов'язаних однією темною спільною історією. У старому покинутому замку гравцю потрібно досліджувати кожний куток, уникати виявлення, змагатися з ворогами і уникати жахливих пасток, щоб врятуватися від зла, що таїться всередині світу гри.

У грі INMOST дають змогу керувати трьома дуже різними персонажами: маленькою дівчинкою, яка досліджує будинок, повний таємниць і жахливих спогадів; бойовим лицарем, який прокладає собі шлях через підступні нутроці похмурого замку, слугуючи дивовижній істоті іншого світу; та занепокоєним чоловіком середнього віку, яким потрібно керувати основну частину гри, виконуючи складні платформні завдання та головоломки.

Кожен з цих трьох персонажів має свій власний унікальний стиль гри. Маленька дівчинка не висока, тому доведеться тягти предмети та прокладати шляхи, щоб вона могла дістатися до об'єктів і точок доступу, щоб продовжити свою історію. Чоловік середнього віку може стрибати, лазити і знаходити різні предмети, щоб допомогти собі в подорожі, а лицар може розкидатися гарпуном, рухаючись дуже обмежено, і використовувати меч та ухилятися, проходячи через містичних монстрів, які сидять у лабіринті гри.

Переваги INMOST:

- Сюжетні катсцени. Гра має добре розроблену і глибоку історію, яка розкривається через захоплюючі сюжетні катсцени. Ці катсцени допомагають поглибити сприйняття головних персонажів, розкрити їхні мотивації та з'єднати окремі фрагменти історії в єдину картину.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		14

- Налаштування і атмосфера. INMOST має структуровані налаштування освітлення та звуку, щоб створити потрібну атмосферу для кожної сцени. Це допомагає поглибити емоційне враження від гри і поринути у світ загадковості та темряви.
- Зв'язок з персонажами. Гра вдало розбиває бар'єр між гравцем і персонажами, створюючи емоційний зв'язок. Через взаємодію з персонажами, гравець може співчувати їхнім стражданням, відчувати радість за них і стати частиною їхньої подорожі.
- Розмаїття геймплею. INMOST пропонує різноманітні види геймплею, включаючи платформи, головоломки, боротьбу та розвідку. Це розширює можливості гравця і додає варіативності до гри, утримуючи її цікавою та непередбачуваною.



Рисунок 1.1 – INMOST

Недоліки INMOST:

- Коротка тривалість. Деякі гравці вважають, що гра триває занадто короткий час. Залежно від стилю гри та швидкості прогресування, історія може бути завершена всього за кілька годин.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		15

- Нерівномірний розподіл геймплею. Інколи можна відчути нерівномірність у розподілі геймплею між трьома головними персонажами. Наприклад, можуть виникати ситуації, коли один персонаж має більше цікавих завдань та розв'язування головоломок, ніж інші.
- Розмита ціль. Деякі гравці вважають, що певні частини сюжету або ціль гри можуть бути не настільки чіткими або зрозумілими. Це може призвести до плутанини або неоднозначності щодо напрямку історії.
- Обмеженість геймплею. Хоча гра пропонує різноманітні види геймплею, деякі гравці можуть відчувати обмеженість у можливостях і варіантах взаємодії з оточуючим світом. Деякі гравці бажають більшої відкритості і свободи в розвідуванні та виконанні завдань.

2. Dead Cells

Даний аналог – це спрощений екшн-платформер, натхненний Castlevania, від студії Motion Twin. Він був випущений 7 серпня 2018 року. Гра дозволяє дослідити просторий замок, що постійно змінюється, за умови, що гравець зможе подолати його оборонців. Щоб перемогти в грі, потрібно опанувати двовимірну бойову систему з завжди присутньою загрозою перманентної смерті без місць для збереження гри.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		16

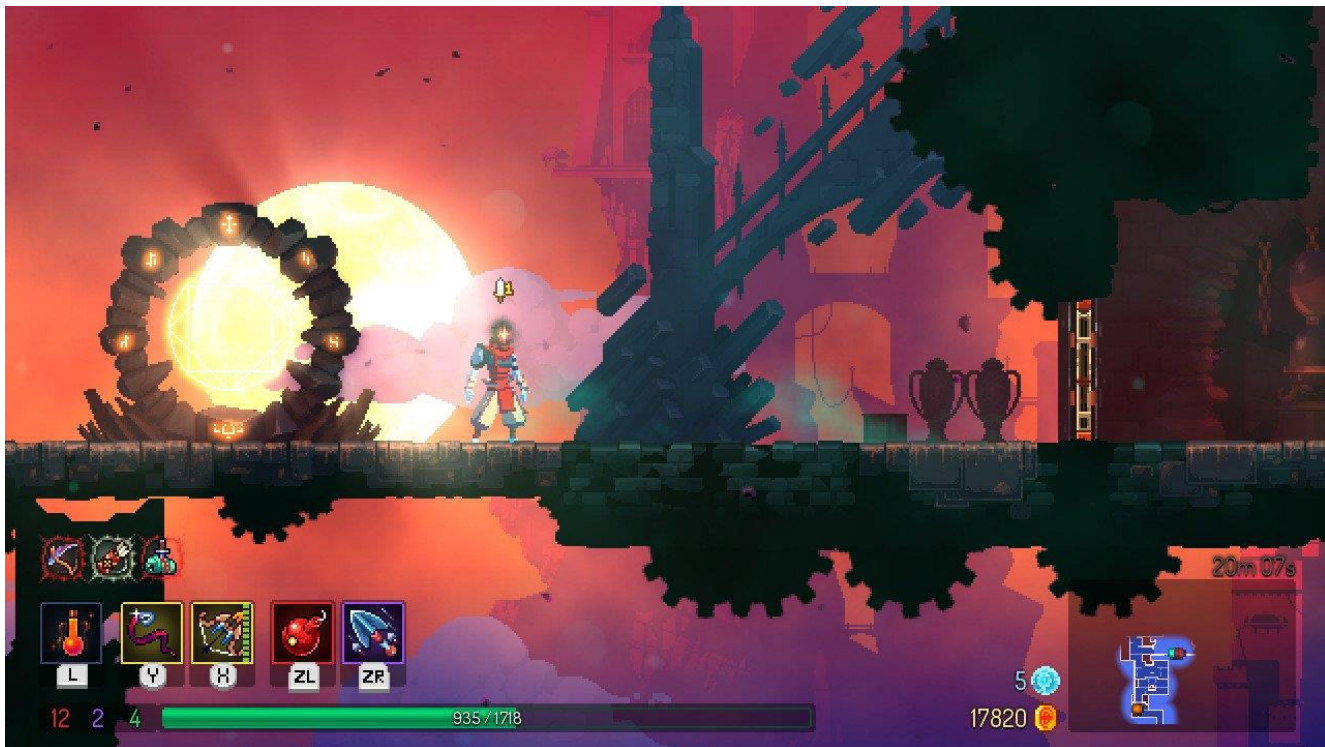


Рисунок 1.2 – Dead Cells

Переваги Dead Cells:

- Захоплюючий сюжет. Гра має захоплюючий сюжет, який розкривається через сюжетні катсцени та взаємодію з персонажами в грі. Таким чином гравець дізнається більше про світ гри та його історію.
- Налаштування персонажа. Гра надає широкі можливості для налаштування головного героя. Персонаж може натягти різні зброї, закляття та уміння, щоб створити унікальний стиль гри, який відповідає уподобанням гравця.
- Різноманітні механіки гри. Dead Cells пропонує багато різних механік гри, що додають різноманітності та глибини геймплею. Гравець може використовувати різні комбінації атак, блокування та ухилення, щоб ефективно боротися з ворогами. Крім того, гра пропонує різноманітні пастки, загадки та секрети, які треба розгадати та використовувати для прогресування.
- Рівень генерації. Головний замок генерується випадковим чином, що означає, що кожна нова гра буде унікальною. Це виклик для гравців, оскільки вони повинні адаптуватися до нових обставин на кожному перезапуску рівня.

- Швидкий темп гри. Dead Cells відома своїм швидким темпом гри, що створює високий рівень напруження та екшену. Битви з ворогами вимагають швидкої реакції та точності.

Недоліки Dead Cells:

- Висока складність. Гра відома своїм високим рівнем складності, особливо для новачків. Вимоглива бойова система та безперервна загроза перманентної смерті можуть викликати тривожність у гравців, які шукають більш доступні шляхи проходження.
- Відсутність історичної вітки. Хоча гра має захоплюючий сюжетний задній план, історія розкривається загалом через атмосферу та довідковий матеріал. Деякі гравці можуть прагнути більш прямої структури або глибшої розробки персонажів.
- Повторення контенту. У деяких випадках, коли гравець переграє рівень кілька разів для досягнення кращих результатів, може виникнути відчуття монотонності або повторення. Нестача достатнього рівня різноманітності може призвести до втрати інтересу після тривалої гри.
- Відсутність багатокористувацького режиму. Гра Dead Cells пропонує лише одиночний режим гри, що означає, що відсутні можливості для багатокористувацької гри або спільного досвіду з друзями.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		18

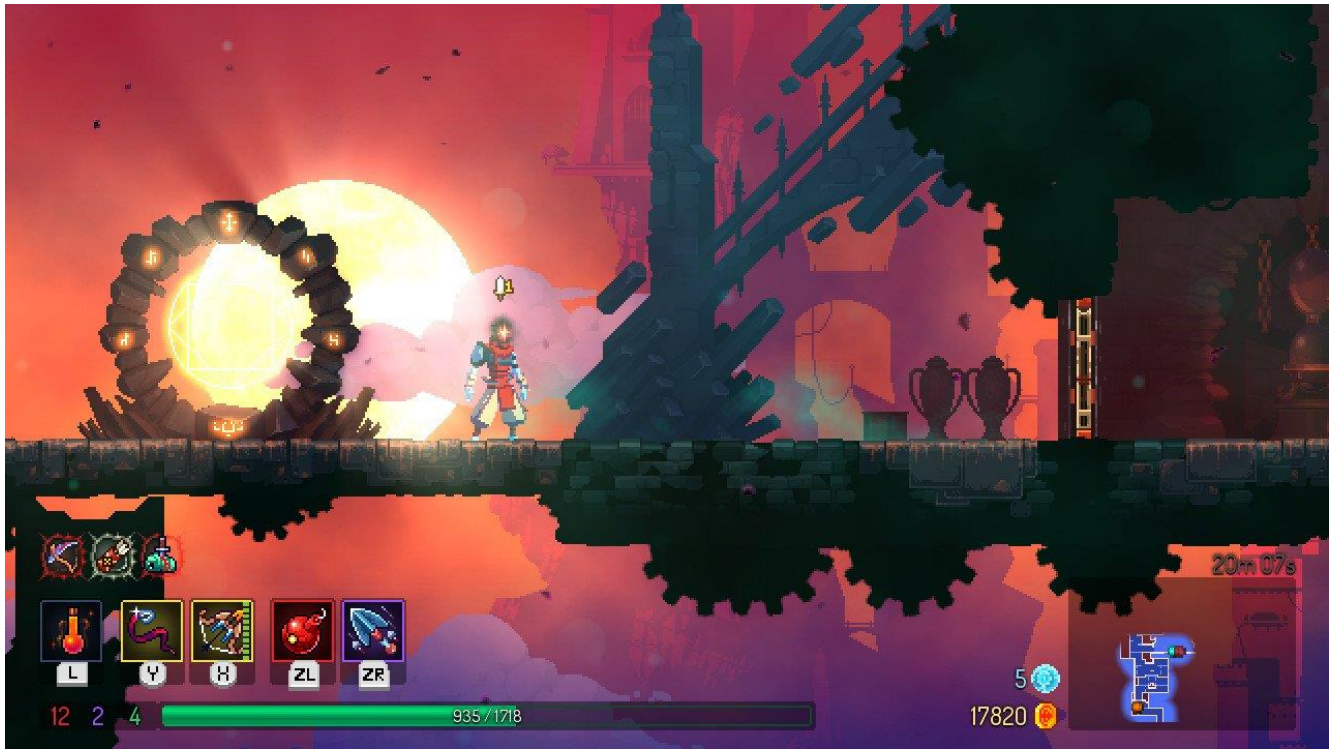


Рисунок 1.3 – Blasphemous

3. Blasphemous

Даний аналог має назву «Blasphemous», що перекладається як «Богохульний». Це відеогра у жанрі екшн-платформера та метроїдванії, розроблена іспанською ігровою студією The Game Kitchen і випущена командою Team17.

Сюжет гри розгортається в Кустоді, країні з сильним впливом римо-католицизму, іконографії та іспанської культури, зокрема регіону Андалусія в Іспанії. Цей святий край відомий своєю особливою силою, відомою як "Диво", яка проявляється у дивних та іноді жорстоких способах, перетворюючи мешканців Кустоді на спотворені чудовиська або благословляючи їх.

Головний герой, відомий як Спокутник, єдиний хто вижив з Братства Тихої Скорботи, яке володіє обітницею мовчання. Він носить маску та гостроверхий шолом, огорнутий терновим вінцем, а його меч прикрашений трояндовими шипами. Під час свого паломництва він зустрічає різних персонажів, таких як Деограсіас, який відіграє важливу роль у сюжеті та діє як оповідач гри, Реденто, паломник, який носить як знак покаяння схилену спину та зв'язані руки, Канделарія, стара торговка, що продає різноманітні предмети, і Вірідіана, яка пропонує допомогу

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		19

гравцю у битвах з босами. У Кустоді править Церква під керівництвом Його Святості Ескрібара, який воскрес як Останній Син Дива задовго до подій гри та встановив тиранічне правління.

Переваги Blasphemous:

- Захоплюючий сюжет. Гра має захоплюючий сюжет, який розкривається через сюжетні катсцени та взаємодію з персонажами в грі. Таким чином гравець дізнається більше про світ гри та його історію.
- Світ та рівні проходження. Blasphemous пропонує розмаїття місць та рівнів, які вражають своєю деталізацією та атмосферою. Кожне місце має свою унікальну естетику, створюючи враження живого світу, повного загадкових та небезпечних місць для дослідження.
- Зміна та прогресія персонажа. Гра дозволяє гравцям налаштовувати свого героя за допомогою різних здібностей та кольорів обладунків. Гравець може вдосконалювати свої навички, отримувати нові атаки та здібності, що робить героя сильнішим у боротьбі зі своїми ворогами.
- Вимогливий бойовий геймплей. Blasphemous пропонує вимогливий та стратегічний бойовий геймплей. Гравцям потрібно вивчити рухи ворогів, уникати атак та вчасно реагувати для успішної боротьби. Бойова система відчувається задовільною та виглядає ефектно.
- Чудова атмосфера та візуальний стиль. Blasphemous вражає своїм готичним та макабричним візуальним стилем. Підкреслені деталі та розкішні анімації додають грі особливу атмосферу та чарівність.

Недоліки Blasphemous:

- Високий рівень виклику. Гра вважається досить складною, особливо для гравців, які не знайомі з жанром метроїдванії. Це може здатися забагато для деяких гравців і може призвести до втрати інтересу.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		20

- Повторність. Після деякого часу гра може викликати відчуття повторюваності, оскільки багато областей і ворогів повторюються в процесі прогресу. Це може зменшити відчуття свіжості та новизни гри.
- Візуальне перенасичення. Дизайн різних ворогів, рівнів та кровавих ефектів може бути дуже насиченим і навіть перевантаженим для деяких гравців. Це може вплинути на читабельність екрану та загальний візуальний досвід.
- Помітні баги та технічні проблеми. У деяких випадках гравці можуть зіткнутися з помітними багами та технічними проблемами, такими як падіння кадрів, несправні анімації або проблеми зі збереженням гри. Це може негативно вплинути на загальний досвід гри.
- Обмежений розмах інтерактивності. Гра може відчуватися дещо лінійною і обмеженою у плані інтерактивності. Існує обмежена можливість впливу гравця на оточуючий світ та розвиток сюжету.

Таким чином, було розглянуті деякі аналоги комп'ютерних ігор за жанром метроїдванія. Наявність аналогів свідчить про популярність даної теми випускної роботи, жанру та концепцій серед розробників та гравців.

Проте, хоча аналоги важливі для вивчення та прийняття успішних рішень, важливо, щоб гра мала свою унікальність. Розробники повинні працювати над створенням унікальних механік гри, сюжетних поворотів або візуального стилю, щоб виділитися серед конкурентів і привернути увагу гравців.

Присутність аналогів також вказує на наявність конкуренції на ринку комп'ютерних ігор. Розробники повинні свідомо підходити до ринкової ситуації і докладати зусиль, щоб залучити увагу гравців та забезпечити конкурентоспроможність своєї гри. Це може бути досягнуто шляхом вдосконалення геймплею, технічних аспектів та надання унікального та захопливого досвіду.

1.3 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

Перш за все, необхідно визначити набір інструментальних засобів, які найкраще відповідають поставленим цілям і завданням випускної роботи. Це повинні

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		21

бути програмні засоби для розробки комп'ютерної гри та для створення унікального стилю, які забезпечують зручну та ефективну реалізацію розроблених алгоритмів і систем. При виборі таких інструментів необхідно враховувати їхню функціональність, надійність, зручність використання та можливості інтеграції з іншими інструментами, що використовуються в процесі роботи.

Крім того, необхідно встановити вимоги до апаратного забезпечення, необхідного для виконання розробки та тестування програмних рішень. Це включає комп'ютерну систему з певними параметрами, такими як оперативна пам'ять, обсяг дискового простору та графічні можливості. Вимоги до апаратного забезпечення повинні враховувати потреби в ресурсах, що виникають при виконанні розробки, а також специфіку роботи з обраними програмними засобами.

Важливим критерієм при обґрунтуванні вибору інструментальних засобів та вимог до апаратного забезпечення є їхня доступність і вартість. Вибір таких засобів повинен бути обґрунтованим з точки зору ефективності та економічної доцільності.

Спочатку необхідно розглянути популярні інструментальні засоби, один з яких є об'єктно-орієнтована мова C#.

C# є сучасною об'єктно-орієнтованою мовою програмування, розробленою компанією Microsoft. У поточному 2023 році C# продовжує підтримувати свою популярність серед розробників завдяки своїм універсальним можливостям, продуктивності та легкості використання. Вона знайшла широке застосування в розробці різноманітних додатків, починаючи від настільних програм до веб та мобільних додатків.

Особливу популярність C# здобула в галузі розробки ігор, де вона використовується для створення високопродуктивних ігрових проектів для консолей, персональних комп'ютерів та мобільних пристроїв. Крім того, C# активно використовується при створенні корпоративних додатків, що дозволяють розробникам створювати програмне забезпечення для потреб підприємств і організацій.

Основні характеристики C# полягають у наступному:

- є мовою програмування, що базується на об'єктно-орієнтованому підході, де об'єкти використовуються для представлення даних і функцій. Це дозволяє

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		22

зручно використовувати код у різних частинах програми та логічно структурувати дані;

- особливість в автоматичному зборі сміття, що означає, що розробникам не потрібно безпосередньо керувати пам'яттю. Ця функція сприяє покращенню надійності та спрощенню обслуговування коду на C#;
- є кросплатформенною мовою програмування, що означає, що її можна використовувати для розробки програмного забезпечення, яке працює на різних операційних системах, включаючи Windows, MacOS та Linux;
- є універсальною, що дозволяє створювати різноманітні додатки, починаючи від настільних програм і закінчуючи веб та мобільними додатками;
- може ефективно обробляти складні завдання та великі обсяги даних;
- є офіційною мовою сценаріїв для Unity, що означає, що вона має глибоку інтеграцію з двигуном Unity та надає доступ до його API (Application Programming Interface) [5]. Це дозволяє розробникам створювати скрипти, контролювати об'єкти, обробляти вхідні дані, реалізовувати логіку гри та багато іншого;
- має широкий набір функцій, бібліотек та фреймворків, які дозволяють розробникам розширювати функціонал своїх ігор та додатків. Додаткові бібліотеки можуть бути використані для графіки, фізики, штучного інтелекту, мережевого взаємодії та багато іншого.

Своїми можливостями C# зарекомендувала себе як популярна мова програмування. Вона використовується в найкращих світових компаніях, що займаються розробкою ігор. За рейтингами, вона входить до переліку 10 найкращих мов програмування для розробки ігор у 2023 році.

C# має репутацію складної та багаторівневої мови програмування. Давайте заглибимось трохи глибше і розглянемо деякі програми, що були написані з використанням C#.

Одним з найвідоміших прикладів використання C# є розробка Microsoft Windows - найпопулярнішої операційної системи у світі. Хоча розробка Windows

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		23

включає в себе й інші мови програмування, С# відіграє ключову роль у створенні основного функціоналу, доводячи надійність та великі обчислювальні можливості цієї мови [15].

Ще одним прикладом використання С# є розробка Microsoft Office - набору програм, з якими багато людей працюють щодня. Програми, такі як Word, Excel, PowerPoint та Outlook, були розроблені з використанням С#, що дозволило створити зручні та функціональні інструменти для користувачів.

Також варто зазначити Adobe Photoshop - найпопулярніший фото редактор, який також був розроблений з використанням С#. Вибір мови програмування обумовлений багатозадачністю та багаторівневістю програми, а С# надає можливість реалізувати такі складні функції.

Ці приклади ілюструють широкий спектр застосування С# у великих та складних програмах, що підтверджує його потужність та гнучкість в розробці програмного забезпечення.

Після випуску .NET-фреймворку компанією Microsoft з'явилося безліч можливостей для створення потужного, гнучкого та надійного програмного забезпечення. Проте варто відзначити, що цей фреймворк був спрямований лише на платформу Windows. Це викликало питання про можливість його використання на інших платформах. У 2001 році виник відкритий проект під назвою Mono, який став рішенням цієї проблеми. Mono надав можливість використовувати .NET-сумісні класи фреймворку на інших платформах та постачав свій власний компілятор С# та CLR. Зараз Mono підтримує майже всі функціональні можливості .NET і продовжує активно розширювати свої можливості.

У випадку цього фреймворку виконання коду проходить за аналогічними принципами, як в .NET-фреймворку:

- при написанні коду використовується мова програмування С#;
- код на С# компілюється вбудованим компілятором С#, що генерує байт-код або проміжну мову;

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		24

- збірка, що включає цей скомпільований код, запускається на цільових пристроях з використанням вбудованого середовища виконання Mono (або Загального Середовища Мови);
- Mono відповідає за виконання завдань, таких як керування збиранням сміття, забезпечення безпеки, обробка виключень та інші аспекти виконання програми.

Двигун виконання коду в Mono, відомий як Runtime Mono, працює в трьох режимах, які забезпечують оптимальне виконання програмного коду:

- компіляція Just-in-Time (JIT): При виконанні код перетворюється в машинний код;
- компіляція Ahead-of-Time (AOT): У цьому режимі більшість коду компілюється перед виконанням. Проте, деякі частини коду, наприклад, трапліни та інші, все ще потребують JIT-компіляції для належної функціональності;
- повна статична компіляція: Цей режим підтримується лише деякими пристроями, такими як IOS від Apple, PlayStation 3 та Xbox. Він виходить за межі AOT-компіляції, оскільки генерує всі трапліни, обгортки та проксі, повністю усуваючи необхідність у JIT-компіляції.

Unity 3D, відомий ігровий рушій, використовує фреймворк Mono, щоб реалізувати можливість його переносу на різні платформи. Завдяки інтеграції з Mono, Unity 3D здатний створювати ігри, які працюють на різних платформах, включаючи Windows, MacOS, Linux, Android, IOS та інші.

Враховуючи можливість переносу та гнучкість фреймворку Mono, а також здатність створювати потужні, гнучкі та кросплатформенні додатки, які працюють на різних пристроях, було прийнято рішення використовувати мову C# Mono для розробки випускного проекту.

Розглянемо ігрові рушії на мові C#, які надають розробникам потужні інструменти та функціональність для процесу створення ігор. Вибір певного рушія залежить від потреб розробки, особистих вподобань та цілей проекту.

Unity є одним з найпопулярніших ігрових рушіїв, що використовують мову C#. Він пропонує широкий спектр функціональності та інструментів для розробки

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		25

ігор, включаючи візуальний редактор, систему фізики, частинок, анімації та багато іншого [4]. Unity також забезпечує підтримку кросплатформенності, що дозволяє розробникам створювати ігри для різних платформ, таких як Windows, MacOS, IOS, Android і багато інших.

Godot є безкоштовним та відкритим ігровим рушієм, який також використовує мову C#. Він має простий у використанні інтерфейс та набір інструментів для розробки ігор. Godot надає гнучкість у роботі з різними типами ігрових об'єктів, включаючи 2D та 3D графіку, анімацію, фізику, звук та інше. Цей рушій також дозволяє розробникам створювати ігри для різних платформ.

Monogame є відкритим рушієм для розробки ігор на основі XNA Framework та використовує мову C#. Він надає розробникам доступ до функцій низького рівня ігрового рушія, що дозволяє створювати швидкі та ефективні ігри. Monogame підтримує різні платформи, включаючи Windows, MacOS, Linux, Android, IOS та інші, і надає можливості для розробки як 2D, так і 3D ігор.

Хенко: Хенко є безкоштовним ігровим рушієм з відкритим вихідним кодом, який використовує мову C#. Він надає інтегроване середовище для створення як 2D, так і 3D ігор. Хенко пропонує багато функцій, включаючи систему частинок, штучний інтелект, анімацію, фізику та інші, що допомагають розробникам створювати захоплюючі ігрові візуальні ефекти. Він також підтримує кросплатформенність і дозволяє розробникам розгортати ігри на різних пристроях та операційних системах.

Хоча кожен з цих ігрових рушіїв на мові C# має свої унікальні переваги та особливості, Unity є найбільш відповідним для створення ігри у жанрі метроїдванія, задовольняючи потреби випускної роботи найкращим чином [8].

Щоб реалізувати сюжетні катсцени у Unity, необхідно звернути увагу на такий інструмент як система подій Unity (Unity Event System). Ця система дозволяє створювати та керувати подіями та поведінкою, яка відбувається в ігровому середовищі. І саме завдяки системі подій Unity відкриваються безліч можливостей для ефективної реалізації сюжетних катсцен.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		26

Система подій - це інструмент, який надається ігровим двигуном Unity і дозволяє розробникам створювати та керувати подіями у своїх іграх або програмах. Вона розроблена для полегшення взаємодії між об'єктами гри та компонентами, що дозволяє їм взаємодіяти та реагувати на різні вхідні дані, тригери та дії користувача [16].

Дана система використовує шаблон "публікація-підписки", що реалізує механізм передачі повідомлень, в якому відправники повідомлень, що називають видавцями, не здійснюють пряме відправлення повідомлень приймачами, які називаються підписниками. Коли відбувається певна подія, наприклад, натискання кнопки або зіткнення, відповідна подія спрацьовує, і всі підписані слухачі отримують повідомлення.

Розробники можуть створювати та налаштовувати події відповідно до своїх конкретних вимог. Unity надає можливість прикріплювати слухачів подій до об'єктів гри або компонентів, визначаючи, як гра повинна реагувати на спрацювання події. Це дозволяє реалізувати різні механіки гри, взаємодію з користувацьким інтерфейсом, керування анімацією та багато іншого.

Також вона сприяє модульному та роз'єднаному дизайну, що полегшує управління та розширення функціональності ігор. За допомогою подій розробники можуть створювати більш інтерактивні взаємодії для гравців.

Загалом, система подій Unity спрощує впровадження поведінки, орієнтованої на події в іграх Unity, надаючи гнучкий та інтуїтивно зрозумілий спосіб обробки введення користувача, взаємодії та ігрових подій [9].

Не найменш важливим для реалізації сюжетних катсцен є система анімації Unity (Unity Animation System). Дана система надає інструменти та можливості для створення живих та реалістичних анімаційних переходів, що додають глибину до сюжету гри.

Система анімації Unity пропонує широкий набір інструментів для роботи з анімаціями об'єктів, персонажів та інших елементів гри. Таким чином є можливість використовувати ключові кадри, криві руху, переходи, зміну розміру та багато іншого для створення реалістичних анімаційних ефектів.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		27

Система анімації Unity інтегрується з системою подій Unity, що дозволяє створювати синхронізовані анімаційні переходи під час відтворення сценаріїв. Це забезпечує плавність та точність у відтворенні сюжетних катсцен, додаючи реалістичності та емоцій до гри.

Обидві системи Unity є хорошими інструментами для реалізації анімації та подій у грі, проте вони також мають деякі обмеження і недоліки.

Недоліками Unity Animation System можуть бути:

- не інтегрується напряду з фізичним двигуном Unity, тобто анімації можуть бути обмеженими у взаємодії з фізичними об'єктами та впливом фізичної симуляції на анімацію;
- не має прямої підтримки синхронізації анімацій з аудіо. Для досягнення точної синхронізації звуків з анімацією можуть знадобитися додаткові налаштування та зусилля.

Недоліки Unity Event System включають:

- може бути важко відслідковувати та налагоджувати події. На відміну від інших інструментів, Unity Event System не надає детальних засобів для перевірки підписки на події, встановлення точок зупинки або налагодження логіки подій;
- більше спрямований на синхронне виконання подій, тобто події обробляються в тому ж потоці, де вони спрацювали. Якщо потрібно обробити асинхронні події або використовувати багатопоточну логіку, доведеться шукати додаткові бібліотеки;
- не існує природного механізму для регулювання порядку обробки подій при їх перекритті. Якщо дві або більше події спрацювають одночасно, можуть виникнути складнощі з визначенням порядку, в якому вони повинні бути оброблені.

Все ж таки, з врахуванням досвіду та вміння використовувати ці інструменти, Unity Animation System і Unity Event System можуть значно полегшити процес створення анімацій та реалізації подій у сюжетних катсценах гри Unity.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		28

Для створення унікального стилю у комп'ютерній грі, було вирішено використовувати програму Aseprite. Aseprite є інструментом для створення піксельної графіки та анімацій, який дозволяє докладно налаштовувати кожен аспект візуального вигляду гри.

Одним з головних переваг використання Aseprite є його спеціалізація на роботі з піксельною графікою. Це дозволяє створювати деталізовані та стилізовані піксельні малюнки, які додають унікальності та характеру до гри [6]. У програмі є різні інструменти, такі як пензлі, заливка, фільтри та інші, щоб створити багатий асортимент графічних ефектів і деталей.

Крім того, Aseprite надає можливість анімувати графіку. З його допомогою можна створювати рухомі спрайти, анімовані переходи та ефекти, що дозволить надати персонажам та об'єктам динаміки. Aseprite має зручний інтерфейс, який дозволяє працювати зі шкалою часу, ключовими кадрами та іншими інструментами для створення плавної та креативної анімації.

Також даний інструмент надає можливість експортувати графічні ресурси у різні формати, що відповідають вимогам гри у Unity. Експортування спрайтів, анімацій та інші елементів будуть використовуватися у процесі розробки.

Загалом, використання програми Aseprite дозволяє створити унікальний стиль гри, завдяки його спеціалізації на піксельній графіці, зручному інтерфейсу та можливості анімації [10]. Він допомагає надати грі виразність.

Висновки до розділу 1

У першому розділі кваліфікаційної роботи було розглянуто декілька важливих аспектів, пов'язаних з підготовкою до розробки комп'ютерної гри у жанрі метроїдванія, з сюжетними катсценами та налаштуваннями.

Початково, була поставлена задача визначення цілей роботи. Головним завданням було створення геймплею, привабливого візуального середовища та інтерактивних функцій для користувачів.

Далі був проведений аналіз аналогів програмного продукту для детального вивчення існуючих ігрових рішень та визначення їх сильних та слабких сторін. Цей

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		29

аналіз допоміг зрозуміти, як можна покращити гру кваліфікаційної роботи та надати їй конкурентні переваги.

Крім того, було обґрунтовано вибір необхідних інструментальних засобів для розробки гри. Була використана мова програмування C#, яка є потужною та гнучкою, ідеально підходить для створення ігор. Unity, відомий ігровий рушій, було обрано для розробки гри з метою отримання багатофункціональності та кросплатформенності. Для керування подіями та анімацією в грі використовувались Unity Event System та Unity Animation System. Відповідно, для створення унікального візуального стилю була використана програма Aseprite, яка дозволяє створювати деталізовані піксельні малюнки.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		30

РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Визначення варіантів використання та об'єктно-орієнтованої структури системи

Перш ніж розпочати розробку кваліфікаційної роботи, необхідно провести загальне проектування, розділивши його на конкретні етапи. Ці етапи охоплюють ключові компоненти, що сприяють ефективному розвитку програмного продукту. У цьому розділі буде проведено детальне планування та опис роботи, забезпечуючи його високу якість та успішну реалізацію.

Етапи проектування:

- технічне завдання;
- структура дизайну гри.

Технічне завдання включає в себе опис вимог до гри, її частин які розробляються у даній кваліфікаційній роботі та її функціональності. Воно визначає, які можливості повинна мати гра, які механіки повинні бути реалізовані, які особливості повинні бути вбудовані в гру. Технічне завдання також включає системні вимоги до гри, такі як платформи, на яких вона повинна працювати, мінімальні вимоги до апаратного забезпечення, мережеві можливості та інші технічні аспекти.

Структура дизайну гри включає у себе ряд елементів, які допомагають створити світ гри. Один з таких елементів - покадрова анімація головного героя та неігрових персонажів. Це означає, що для кожної дії чи руху персонажів створюються окремі кадри анімації, які послідовно відтворюються для створення руху. Це дозволяє персонажам виглядати живими та деталізованими, додаючи грі більшу реалістичність та естетичну привабливість.

Наступний елемент включає постановку сюжетних катсцен. Це означає, що розгортання історії та відтворення важливих подій відбувається через спеціально створені сцени, діалоги та взаємодію персонажів. Постановка сюжетних катсцен допомагає підтримувати структуру гри та залучати гравця до історії.

Крім того, структура дизайну гри включає генерацію палітри кольорів. Кольори в грі мають велике значення для передачі настрою, атмосфери та сприйняття сцен гравцем. Генерація палітри кольорів допомагає підібрати правильні відтінки і комбінації кольорів, щоб вони відображали задуманий настрій та естетичну концепцію гри.

Ці елементи структури дизайну гри - покадрова анімація, постановка сюжетних катсцен та генерація палітри кольорів - співпрацюють, щоб створити візуальний світ гри, який захоплює гравця, передає настрій та додає глибини до геймплею.

Також важливо виділити ролі в команді, щоб кожен учасник виконував конкретні завдання, які наведені в таблиці 2.1. Кожний учасник команди має свої власні обов'язки, що сприяють ефективній співпраці та досягненню цілей кваліфікаційної роботи.

Таблиця 2.1 Ролі команди

Учасник	Ролі
Харченко Віталій Андрійович	Team Leader, Lead Developer
Леут Назарій Віталійович	Developer, QA
Козакевич Наталія Олександрівна	Developer, Director
Буханевич Оксана Андріївна	Developer, Designer

2.1.1 Технічне завдання

У даній кваліфікаційній роботі розробляється частина комп'ютерної гри, яка містить сюжетні катсцени та системні налаштування. Таким чином, необхідно скласти технічне завдання, яке відображатиме подальшу роботу розробника.

1. Опис проекту

Гра “Dark Watcher” є комп'ютерною грою у жанрі метроїдванія, тобто має простору карту для дослідження з нелінійним сюжетом. Головний герой опиняється у деякій сфері, де йому доведеться позмагатися зі своїми важкими періодами життя. Разом з тим його час від часу супроводжує Олень, який вказує на деякі цікаві

моменти його сфери та допомагає у вирішенні проблем, які виникають на шляху головного героя.

2. Функціональні вимоги

2.1. Сюжетні катсцени

- 2.1.1. Створити послідовність сюжетних катсцен, що відповідають сюжету гри;
- 2.1.2. Розробити детальний сценарій для кожної катсцени, включаючи діалоги, дії персонажів та події;
- 2.1.3. Визначити вимоги до візуальної презентації катсцен, включаючи анімацію, ефекти та камерні ракурси;
- 2.1.4. Забезпечити плавні переходи між катсценами та взаємодію гравця з ними;
- 2.1.5. Врахувати можливість пропущення катсцен гравцем за бажанням;

2.2. Системні налаштування

- 2.2.1. Визначити параметри графіки та візуальних ефектів для досягнення бажаного стилю та атмосфери гри;
- 2.2.2. Розробити систему налаштувань гри, що дозволяє гравцеві змінювати настройки графіки, звуку та управління;
- 2.2.3. Забезпечити оптимізацію гри для різних конфігурацій комп'ютерів та платформ;
- 2.2.4. Визначити вимоги до музичного супроводу та звукових ефектів в катсценах та грі загалом;
- 2.2.5. Забезпечити сумісність гри з різними операційними системами та пристроями;

3. Технічні вимоги

- 3.1. операційна система: Windows 7;
- 3.2. процесор: Intel Core 2 Duo E5200;
- 3.3. оперативна пам'ять: 4 ГБ;
- 3.4. відеокарта: GeForce 9800GTX+ з 1 ГБ пам'яті;

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		33

3.5. directX: версія 10;

3.6. місце на диску: 2 ГБ доступного місця;

4. Завдання для розробника

- 4.1. Створення скриптів для сюжетних катсцен, включаючи введення персонажів, візуальних ефектів та діалогів;
- 4.2. Налаштування правильного відтворення анімацій персонажів катсцени та їх тривалість, кількість фреймів;
- 4.3. Розробка трьох різних видів катсцен (сюжетна, на полотні та звичайна), які мають різне смислове навантаження для гравця;
- 4.4. Створити завантаження нової або створеної гри з головного меню;
- 4.5. Розробити всі необхідні системні налаштування для зміни звуку, роздільної здатності, графіки, зміни управління;
- 4.6. Виконання регулярного тестування гри, включаючи перевірку функціональності, стабільності та виявлення помилок;
- 4.7. Виконання тестових проходжень гри з метою перевірки балансу, геймплейного досвіду та коректності роботи сюжетних катсцен.

Дане технічне завдання допоможе структурно розробляти комп'ютерну гру та завдяки ньому було визначено вимоги до функціональності, апаратного забезпечення та інструментів розробників.

2.1.2 Структура дизайну гри

Структура дизайну у грі складається з покадрової анімації різних персонажів, генерації палітри для визначення кольорів гри та постановки сюжетних катсцен.

У кожній грі жанру метроїдванія є головний герой та персонажі або вороги, які з ним взаємодіють. У всіх цих персонажів є певний набір анімацій, який створюється на допомогою малювання кожного кадру. Гра "Dark Watcher" не виключення і вона теж має певний набір персонажів з анімаціями. Таким чином, саме малювання здійснюється у програмі Aseprite, яка має спеціальні налаштування під піксельну 2D графіку.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		34

Також програма дає змогу одразу ж анімувати необмежену кількість кадрів. Проте, оскільки рухи персонажа повинні відображатися плавно та інтуїтивно зрозуміло для гравця, необхідно виставити певну кількість кадрів, які потрібно промалювати. Протестувавши на декількох прикладів, можна сказати що оптимальна кількість кадрів у анімації містить 6-10 кадрів, оскільки якщо поставити замало, то гравець не встигне зрозуміти, що відбулося на екрані, але якщо промалювати надто багато кадрів, гравець може втратити зацікавленість у грі, бо замість активного бою з ворогом, він буде чекати коли анімація закінчиться, щоб натиснути кнопку атаки. Таким чином, 6-10 кадрів вистачає для гарної анімації.

Для експортування кадрів у широкий формат використовується також програма Aseprite для подальшого використання у Unity, оскільки пікселі мають надто малий оригінальний розмір [11]. Після експортування кожного кадру у Unity, необхідно програмувати анімаційні контролери та виставити кадрів окремо, з урахуванням правильної послідовності та їх тривалість. Кожен кадр має свою певну тривалість, яка визначається з урахуванням бажаного ефекту та характеристик гри. Цей процес забезпечує плавний та реалістичний рух персонажів у грі, надаючи їм життєвості та виразності.

Щодо інтеграції у Unity, для покадрової 2D анімації персонажа використовуються наступні інструменти та компоненти:

- Анімаційні контролери (Animation Controllers) - це компонент Unity, який дозволяє керувати анімацією об'єкта [7]. Він містить стани (States), переходи (Transitions) та логіку, яка визначає, яка анімація повинна відтворюватися в певний момент гри. Є можливість створити новий анімаційний контролер та додавати до нього стани, які відповідають різним діям або рухам персонажа.
- Анімаційні кліпи (Animation Clips) - це окремі анімаційні файли, які містять послідовність кадрів [12]. Таким чином створюється анімаційний кліп для кожної дії або руху персонажа, наприклад, ходьби, стрибка чи атаки. Кожен анімаційний кліп може мати свою тривалість та налаштування.

- Спрайтові рендери (Sprite Renderers) - це компонент Unity, який дозволяє відтворювати спрайти (кадри анімації) на екрані [13]. Таким чином можна додати спрайтовий рендер до об'єкта персонажа і вказати, який спрайт (кадр) повинен відображатися на даний момент.
- Параметризовані анімації (Parameterized Animations) - можна використовувати параметризовані анімації для керування анімацією персонажа в залежності від зовнішніх факторів або вхідних даних. Наприклад, використовувати параметр швидкості для зміни анімації ходьби персонажа в залежності від його руху.
- Інспектор анімації (Animation Inspector): Інспектор анімації - це інструмент Unity, який дозволяє налаштовувати параметри анімації, такі як тривалість кадрів, швидкість відтворення та зв'язки між анімаційними кліпами та станами.

Таким чином, у Unity створюється анімаційний контролер, додається до нього анімаційні кліпи та визначаються переходи між ними. Далі, прикріплюється спрайтовий рендер до персонажа та налаштовується, щоб відтворювати відповідні спрайти з анімаційних кліпів. З використанням анімацій з параметрами та інших функцій Unity, можна контролювати процес відтворення анімації персонажа, забезпечуючи плавність та реалістичність руху у грі.

Ці інструменти та компоненти в Unity дозволяють створювати покадрову 2D анімацію персонажа та інтегрувати її у гру з високою якістю.

Наступна складова структури дизайну гри становить постановка сюжетних катсцен:

- використання Timeline дозволяє створювати та управляти послідовностями подій, анімаціями та звуками в сюжетних катсценах. Час відтворення кожної події може бути налаштований для забезпечення синхронізації елементів катсцен;
- Animator Controller контролює анімацію персонажів та об'єктів у сюжетних катсценах. Його використання дозволяє визначати стани та переходи, керуючи анімацією і створюючи різні рухи та дії;

- скрипти C# програмують поведінку об'єктів та дії, що відбуваються в сюжетних катсценах. З їх допомогою можна контролювати послідовність подій, анімацію, параметри об'єктів та виконувати інші дії, які роблять катсцени більш сюжетними та динамічними;
- Cinemachine є інструментом для керування камерою у грі. Використовуючи Cinemachine, можна створювати рухи камери, зуми, змінювати позицію та орієнтацію камери для досягнення відповідної атмосфери та настрою в сценах [14]. Можна налаштувати камеру таким чином, щоб вона слідувала за персонажем або панорамувала по заданій траєкторії, що робить сюжетні катсцени більш динамічними та виразними;
- Particle System дозволяє створювати ефекти частинок, такі як дим, вогонь, вибухи та багато іншого. Використання Particle System додає деталізацію та реалізм до сюжетних катсцен. Можна контролювати властивості частинок, такі як швидкість, розмір, кольори та траєкторії руху, щоб створити бажаний візуальний ефект;
- додавання відповідних звукових ефектів та музики до сюжетних катсцен додає атмосфери та емоційності. Можна використовувати аудіо компоненти Unity, щоб відтворювати звуки руху, голоси персонажів, фонову музику та інші аудіо елементи для створення повноцінного звукового досвіду.

Ці інструменти у Unity дозволяють створювати дизайн сюжетних катсцен. Вони надають можливість створювати послідовності подій, керувати анімацією персонажів та об'єктів, а також програмувати логіку та дії, що відбуваються під час катсцен, створюючи синхронізований геймплей.

Остання складова структури дизайну гри даної кваліфікаційної роботи становить генерація палітри кольорів для гармонічного вигляду локацій, персонажів, меню, налаштувань та інших складових гри. Для формування палітри потрібно застосувати формулу, яка буде генерувати монохромну гармонію кольорів.

Формула для генерації монохромної палітри може бути наступною:

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		37

Необхідно обрати початковий колір, який буде базовим для палітри. Це може бути будь-який колір, представлений у форматі RGB або HEX.

Застосуємо зміну насиченості до початкового кольору, щоб створити варіації монохромних кольорів. Зазвичай, для монохромних кольорів насиченість зменшується, але можна вибрати будь-який метод зміни насиченості, який задовольняє потребам та стилістиці гри.

Формула зміни насиченості може бути представлена таким чином:

Новий колір = $(1 - k) * \text{початковий колір} + k * \text{колір\_позаду}$

У цій формулі:

- k - коефіцієнт зміни насиченості (зазвичай від 0 до 1);
- початковий колір - початковий колір палітри;
- колір_позаду - колір, який використовується як фон або основний колір, від якого змінюється насиченість;
- застосування цієї формули до початкового кольору для кожного кольору палітри допоможе створити монохромну палітру з різними варіаціями одного кольору.

Дана формула є загальною, тому можна налаштувати коефіцієнт зміни насиченості та використовувати додаткові методи для отримання бажаної монохромної палітри.

Для прикладу, щоб згенерувати палітру з трьох кольорів зеленого монохромності, слід застосувати формулу до початкового зеленого кольору тричі, використовуючи різні значення коефіцієнта зміни насиченості (k) для кожного кольору.

Нехай наш початковий зелений колір буде RGB (0, 255, 0) або HEX #00FF00, а колір позаду буде відтінком сірого, наприклад, RGB (128, 128, 128) або HEX #808080.

Припустимо, що необхідно отримати палітру з трьох кольорів зі збереженням основного відтінку зеленого.

Перший колір:

$k_1 = 0.3$ (довільне значення від 0 до 1)

Новий колір₁ = $(1 - 0.3) * (0, 255, 0) + 0.3 * (128, 128, 128)$

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		38

$$\begin{aligned}
&= (0.7 * 0, 0.7 * 255, 0.7 * 0) + (0.3 * 128, 0.3 * 128, 0.3 * 128) \\
&= (0, 178.5, 0) + (38.4, 38.4, 38.4) \\
&= (38.4, 216.9, 38.4) \\
&\approx (38, 217, 38)
\end{aligned}$$

Другий колір:

$$k_2 = 0.6 \text{ (довільне значення від 0 до 1)}$$

$$\begin{aligned}
\text{Новий колір}_2 &= (1 - 0.6) * (0, 255, 0) + 0.6 * (128, 128, 128) \\
&= (0.4 * 0, 0.4 * 255, 0.4 * 0) + (0.6 * 128, 0.6 * 128, 0.6 * 128) \\
&= (0, 102, 0) + (76.8, 76.8, 76.8) \\
&= (76.8, 178.8, 76.8) \\
&\approx (77, 179, 77)
\end{aligned}$$

Третій колір:

$$k_3 = 0.9 \text{ (довільне значення від 0 до 1)}$$

$$\begin{aligned}
\text{Новий колір}_3 &= (1 - 0.9) * (0, 255, 0) + 0.9 * (128, 128, 128) \\
&= (0.1 * 0, 0.1 * 255, 0.1 * 0) + (0.9 * 128, 0.9 * 128, 0.9 * 128) \\
&= (0, 25.5, 0) + (115.2, 115.2, 115.2) \\
&= (115.2, 140.7, 115.2) \\
&\approx (115, 141, 115)
\end{aligned}$$

Таким чином, було отримано палітру з трьох кольорів зеленого монохромності: (38, 217, 38), (77, 179, 77) та (115, 141, 115). Потрібно змінювати значення k для отримання різних варіацій зеленого в монохромній палітрі.

2.2 Розробка подій та системи анімацій. Системні налаштування

При розробці сюжетних катсцен у грі "Dark Watcher" у Unity було задіяно дві головні системи їх створення: Unity Animation System та Unity Event System.

Unity Animation System є інструментом, що дозволяє реалізувати різноманітні анімаційні ефекти та рухи персонажів. Він пропонує різні підходи до анімації, включаючи ключові кадри, інтерполяцію та ієрархічне керування скелетною анімацією. Завдяки Unity Animation System є можливість створити плавні та реалістичні

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		39

рухи персонажів у сюжетних катсценах. Таким чином, можна контролювати параметри анімації, встановлювати часові лінії та створювати переходи між анімаційними станами.

Unity Event System був використаний для управління послідовністю подій та активації різних елементів сценарію відповідно до специфічних умов. Ця система дозволяє налаштовувати тригери, реагувати на дії гравця та автоматично активувати певні події у сюжетних катсценах. Наприклад, використання Unity Event System дозволяє запускати анімації після досягнення певної точки в сценарії, активувати діалогові вікна або змінювати стан оточуючого середовища.

Поєднання Unity Animation System та Unity Event System у грі "Dark Watcher" дозволяє створити сюжетні катсцени з реалістичними анімаціями та взаємодією з гравцем [1]. Ці системи надають гнучкість та контроль над відтворенням подій та анімацій, що сприяє створенню незабутнього геймплею та емоційного занурення гравців у світ гри "Dark Watcher".

2.2.1 Сюжетні катсцени

Сюжетні катсцени є основною складовою розробки гри "Dark Watcher" у середовищі Unity, які характеризуються складністю та унікальністю. Кожна сюжетна катсцена в грі прописується окремим кодом для створення розмаїтості інтерактивних сценаріїв, анімацій, діалогів та подій.

У процесі розробки гри в Unity для реалізації складних сюжетних катсцен було використано різноманітні інструменти та можливості, доступні у цьому середовищі. Один з ключових інструментів - Unity Scripting, що дозволяє програмувати різні аспекти гри за допомогою мови програмування C# [3]. За допомогою Unity Scripting є можливість створювати складні логічні сценарії, керувати поведінкою персонажів, анімаціями та взаємодією об'єктів у сюжетних катсценах. Це дає можливість реалізувати унікальні та потаємні моменти в грі, які розкривають сюжет та поглиблюють емоційний досвід гравців.

Крім Unity Scripting, для створення складних сюжетних катсцен у грі також використовуються інші інструменти, такі як Unity Animator та Unity Timeline [2]. Unity Animator надає можливість створювати й керувати анімаціями персонажів,

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		40

об'єктів та інших елементів, які з'являються в сюжетних катсценах. Також даний інструмент дозволяє надати рухам та виразності цих елементів реалістичну якість. За допомогою Unity Timeline формуються послідовності подій, анімацій та діалогів, що контролюють час, порядок та взаємодію компонентів гри всередині катсцен. Використання цих інструментів сприяє створенню сцен, які передають настрій гри й збільшують зацікавленість для гравців.

2.2.2 Катсцени на полотні

Катсцени на полотні дозволяє створювати та відображати графічні елементи на екрані. Canvas у Unity [2] - це графічний контейнер, який використовується для розміщення та організації різних об'єктів, таких як зображення, кнопки, текст і інші елементи інтерфейсу користувача, на сцені гри. Canvas надає можливість керувати розміщенням, розміром, анімацією та взаємодією цих графічних елементів.

Для роботи з катсценами на полотні застосовується Unity UI, який є набором компонентів інтерфейсу користувача, який дозволяє швидко створювати і налаштовувати елементи інтерфейсу. За допомогою Unity UI можна додавати кнопки, текстові поля, індикатори прогресу та багато іншого. Крім того, Unity також надає можливість створювати власні компоненти та скрипти для керування поведінкою катсцен.

Використання катсцен на полотні із використанням canvas дозволяє створювати ефективні та зручні інтерфейси для гравця, забезпечуючи йому зручну навігацію та взаємодію з грою. Контроль над розміщенням та виглядом графічних елементів, а також можливість взаємодії з ними, допомагають створювати привабливе ігрове середовище.

2.2.3 Звичайні катсцени

Звичайні катсцени відтворюються без перенесення сцени на інший екран та залишаються в межах поточної сцени гри. Вони використовуються для створення послідовного розгортання подій та геймплею, де гравець може безперервно взаємодіяти з оточенням без необхідності переходу на нову сцену.

Для досягнення цього використовується камера, яка контролює, що відображається на екрані гравця і як навігація відбувається по сцені. Вона визначає положення та орієнтацію до об'єкта до якого камера рухається.

Також взаємодія з ігровими об'єктами в звичайних катсценах реалізується за допомогою скриптів та компонентів Unity. Скрипти дозволяють контролювати поведінку об'єктів, забезпечуючи їм анімацію, фізичну модель та інші функціональні можливості.

2.2.4 Налаштування гри

Реєстр Windows - це ієрархічна база даних, яка зберігає налаштування системи та програм у операційній системі Windows. Він зазвичай знаходиться у файловій системі Windows, зокрема у директорії «%SystemRoot%\System32\Config». Реєстр складається з ключів, підключів та значень, які містять конфігураційні дані.

Для отримання даних про системні налаштування з реєстру важливо визначити конкретні ключі та значення, що відповідають потрібним налаштуванням. Залежно від вимог гри ці налаштування можуть включати роздільну здатність екрану, якість графіки, гучність звуку, налаштування введення тощо.

Unity, як середовище розробки ігор, не має прямого доступу до реєстру, проте надає можливості для реалізації крос-платформенного коду. Таким чином потрібно створити скрипти у Unity, які викликають відповідні функції або класи для отримання даних з реєстру. Важливо розробити чітку архітектуру та забезпечити взаємодію між кодом доступу до реєстру та іншими скриптами та системами гри.

Також, необхідно провести тестування функціональності, щоб переконатися, що дані налаштувань системи вірно завантажуються до гри. Потрібно оптимізувати реалізацію, щоб забезпечити ефективність та продуктивність завантаження даних з реєстру.

У грі “Dark Watcher” передача реєстру до Unity було реалізовано за допомогою вбудованих класів:

- AudioMixer в Unity є потужним інструментом для керування звуковими налаштуваннями в грі. Він дозволяє налаштовувати гучність, еквалайзер

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		42

та ефекти звуку. Крім того, ви можете підключати звукові джерела до відповідних груп в AudioMixer і контролювати їх параметри окремо або в межах групи. Це дозволяє створювати збалансований та цікавий звуковий досвід для гравців;

- клас Screen в Unity надає доступ до інформації про екран та дозволяє керувати параметрами відображення. З його допомогою ви можете отримати розмір екрану, налаштувати роздільну здатність, встановити повноекранний режим та змінити орієнтацію екрану. Це особливо корисно для розробки мобільних ігор, де важливо адаптувати графіку та інтерфейс до різних розмірів та орієнтацій екрану пристроїв;
- клас QualitySettings в Unity дозволяє керувати налаштуваннями якості графіки в грі. Ви можете встановити рівень деталізації, якість текстур, розмір тіней та інші параметри, що впливають на візуальний вигляд гри. Це дозволяє розробникам пристосовувати графіку до можливостей пристроїв і забезпечувати оптимальну продуктивність. QualitySettings також дозволяє переключати рівні якості в реальному часі, щоб гравці могли налаштовувати графічні налаштування під свої потреби і можливості своєї системи.

2.3 Проектування та реалізація алгоритмів роботи системи

Одним з основних аспектів проектування алгоритмів системної роботи є визначення загальної структури та організації кодової бази гри. Unity надає гнучку платформу, яка дозволяє створювати модульні та ще раз використані компоненти. За допомогою модульного підходу можна проектувати алгоритми, які не лише ефективні, але й легкі у підтримці та розширенні. Це включає розбиття логіки гри на менші, керовані компоненти, які можуть бути незалежно протестовані та модифіковані, покращуючи загальну якість коду.

На даному етапі розробки наведемо загальний алгоритм роботи катсцен, що зображений на рисунку 2.1.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		43

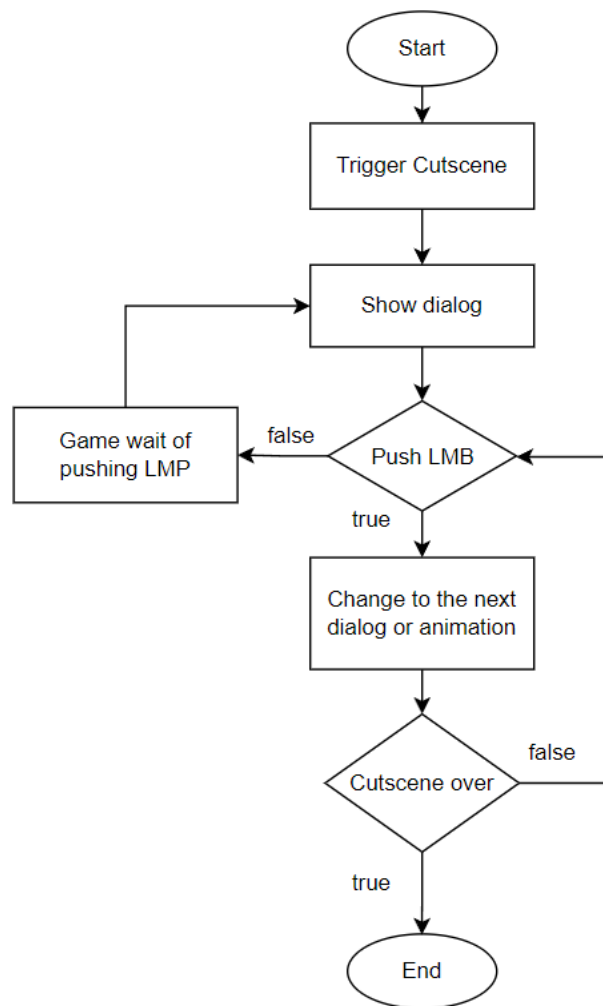


Рисунок 2.1 – Схема загального алгоритму роботи катсцен

Дана схема відображає процес роботи гравця з катсценами, який починається з задіяння гравцем тригера катсцени:

- гравець підходить до області задіяння тригера, спричиняючи початок катсцени;
- при задіянні тригера, Unity встановлює спеціальну сцену та віддалену позицію камери для початку катсцени;
- після початку катсцени, різні елементи, такі як діалоги, анімації або переміщення камери, можуть послідовно відтворюватися згідно з заданою послідовністю;

- гравець може вплинути на подальший хід катсцени шляхом натискання лівої кнопки миші (ЛКМ). Залежно від натискання кнопки, катсцена може продовжуватися або припинятися;
- після завершення катсцени, камера повертається до попередньої позиції та гравець отримує здатність знову керувати своїм персонажем.

Таким чином, кожна з трьох типів катсцен (сюжетна катсцена, катсцена на полотні, звичайна катсцена) має у загальному один алгоритм роботи, при цьому зміст катсцен завжди різний та кожен раз виставляється згідно сюжету.

Перейдемо до наступної діаграми, яка містить системні налаштування.

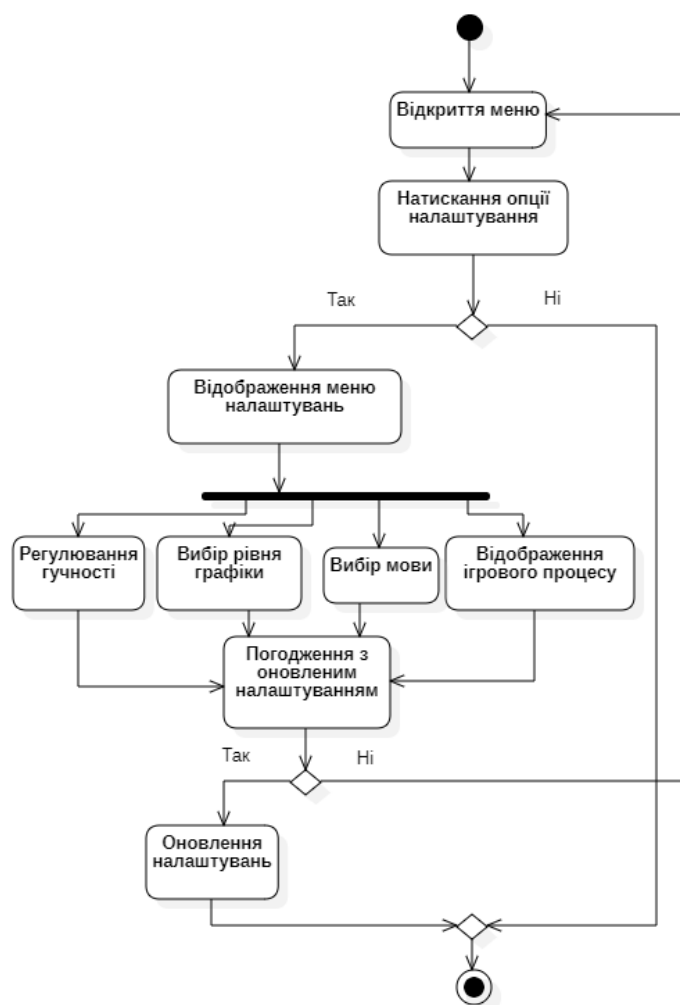


Рисунок 2.2 – Діаграма активності «Налаштування»

Кожен гравець повинен мати змогу підлаштувати комп'ютерну гру під себе задля комфортного досвіду під час її проходження. Діаграма «Налаштування» (див.

рис. 2.2) відображає усі можливі функції та регулювання, які спрямовані на покращення даного досвіду.

Пояснення діаграми «Налаштування»:

- спочатку гравець відкриває меню комп'ютерної гри через запуск лаучеру або за допомогою паузи у процесі гри;
- гравець має можливість обрати опцію налаштувань;
 - якщо гравець натискає опцію налаштувань, відображається певний перелік для змін;
 - якщо гравець не натискає опцію налаштування, діаграма виходить на кінцевий вузол;
- якщо обрана опція налаштувань, відображається меню налаштувань;
- у меню налаштувань гравець має можливість змінити такі опції:
 - регулювання гучності;
 - вибір рівня графіки;
 - вибір мови;
 - відображення ігрового процесу;
- якщо гравець змінив якісь опції налаштувань, він має підтвердити їх зміну:
 - якщо гравець погодився з оновленням налаштувань, то гра змінює своє відображення згідно оновлень;
 - якщо гравець не погодився з оновленнями налаштувань, то гра не змінює своє відображення та гравцю відображається меню гри або меню паузи в залежності від того, як він заходив у налаштування;
- виходження діаграми на кінцевий вузол.

Під час гри підбуваються короткі анімаційні сюжети для детальнішого роз'яснення гравцю про ті чи інші предмети або події. Діаграма «Катсцени» (див. рис. 2.3) відображає дані анімаційні сюжети.

Пояснення діаграми «Катсцени»:

- задіяння тригера катсцени гравцем під час взаємодії з певним предметом або певною областю з тригером;

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		46

- після задієння триггеру, відбувається активація катсцени;
- під час відтворення катсцени, гравцю відображається анімаційний сюжет;
- діаграма виходить на кінцевий вузол.



Рисунок 2.3 – Діаграма активності «Катсцена»

Розглянемо діаграму класів. Діаграма класів є інструментом у процесі проектування та розробки програмного коду. Вона дозволяє визначити структуру програми, ідентифікувати класи та їх взаємозв'язки, а також встановити функціональність кожного класу.

Діаграма класів на рисунку 2.4 дає змогу візуалізувати архітектуру програми та зрозуміти, як класи взаємодіють між собою. Вона допомагає уникнути зайвого зв'язку та забезпечити чистоту та структурованість коду. Крім того, діаграма класів є важливим засобом комунікації між розробниками, допомагаючи зрозуміти інтеграцію та спільну роботу над проектом.

У процесі розробки гри “Dark Watcher” можуть виникати нові вимоги або змінюватися вимоги до системи, що потребує внесення змін у структуру класів. Орієнтована діаграма класів забезпечує зручний спосіб внесення цих змін, дозволяючи додавати нові класи або модифікувати існуючі.

Ця гнучкість орієнтованої діаграми класів дозволяє ефективно реагувати на зміни ітераційної розробки програмного продукту. При необхідності, вносяться

зміни в структуру класів, не порушуючи функціональність системи та забезпечуючи належну модульність та розширюваність коду.

На діаграмі класів класи представлені у вигляді блоків, які містять назву класу, а також його атрибути та методи. Атрибути перераховані всередині поля класу та зазвичай позначені типом даних. Методи також перераховані всередині поля класу та зазвичай позначені типом, які повертають значення.

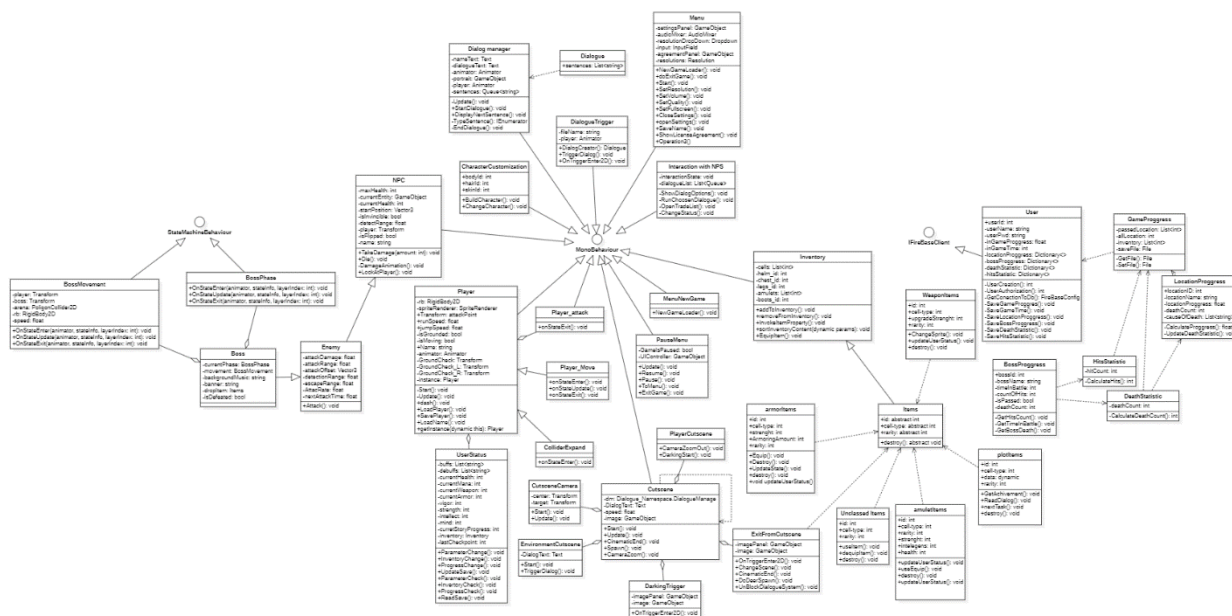


Рисунок 2.4 – Орієнтована діаграма класів

Діаграма послідовності — це тип діаграми, який показує взаємодію між об'єктами або компонентами протягом певного періоду часу. Він використовується для моделювання поведінки системи або для опису конкретної взаємодії між об'єктами в системі.

Діаграми послідовності зазвичай використовуються для моделювання взаємодії між об'єктами в системі в контексті конкретного сценарію або випадку використання.

На рисунку 2.5 представлена діаграма послідовності, яка відображає послідовність подій та взаємодій між гравцем та системою в процесі активації катсцени, вибору налаштувань та початку гри. Вона надає зрозуміння про хід взаємодії та поведінку системи під час цього сценарію, допомагаючи отримати чітку картину процесу та забезпечити правильну послідовність дій у грі "Dark Watcher".

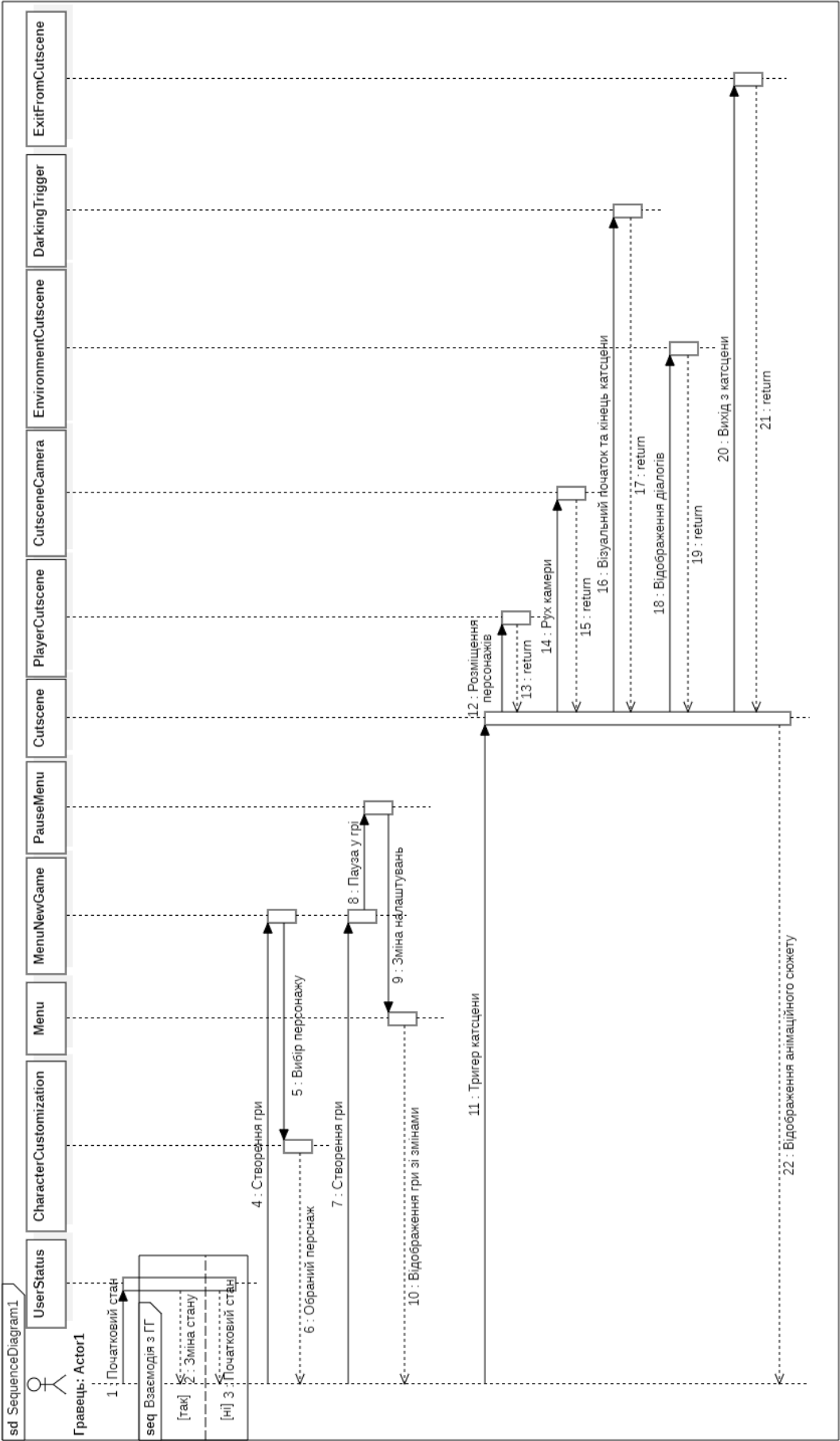


Рисунок 2.5. Діаграма послідовності

На діаграмі компонентів залежності дозволяють зв'язати разом компоненти та інтерфейси, які використовуються конкретними компонентами та компонентами різних типів. У цьому випадку стрілка прямує від компонента клієнта до інтерфейсу. Така стрілка означає, що компонент не реалізує інтерфейс, а використовує під час виконання. При цьому на тій самій діаграмі може існувати інший компонент, що реалізує цей інтерфейс. Відносини реалізації інтерфейсу показані на діаграмах компонентів звичайними лініями без стрілок.

Побудовану діаграму основних компонентів проекрованої системи можна побачити на рисунку 3.2.

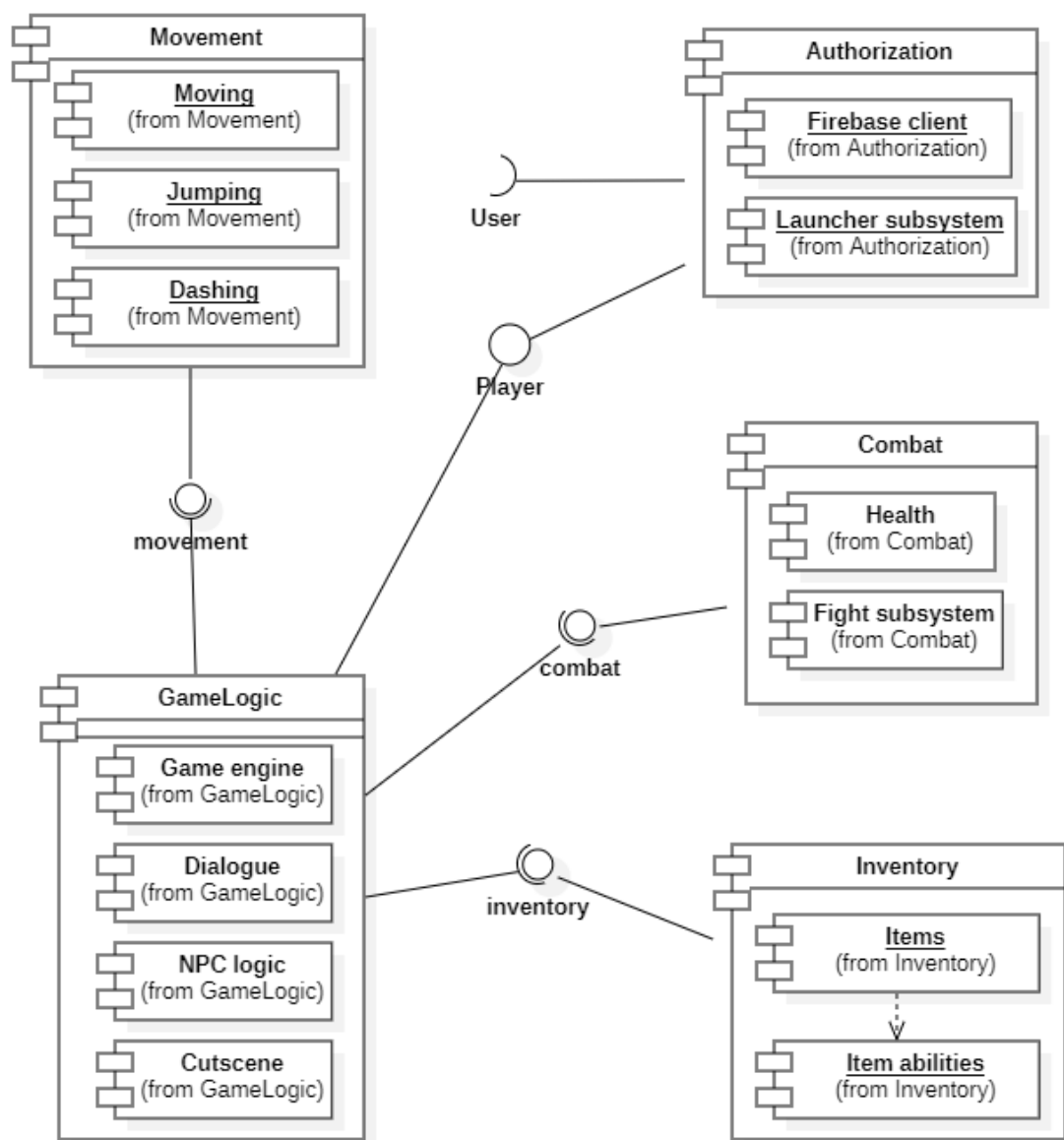


Рисунок 2.6 – Діаграма компонентів

Діаграми розгортання використовуються в рамках UML для відображення конфігурації та топології програмної системи. Вони є останнім кроком у визначенні моделі програмного комплексу гри "Dark Watcher".

Діаграми розгортання використовуються для представлення компонентів та елементів програми, які існують лише на етапі виконання, тобто під час реального запуску гри. Ці діаграми надають графічне уявлення пристроїв, процесів та взаємозв'язків між ними, що становить важливу частину архітектури програми.

На рисунку 3.3 зображена діаграма розгортання для модельованого програмного комплексу гри "Dark Watcher". Ця діаграма надає візуальне уявлення про топологію системи та взаємозв'язки між її компонентами.

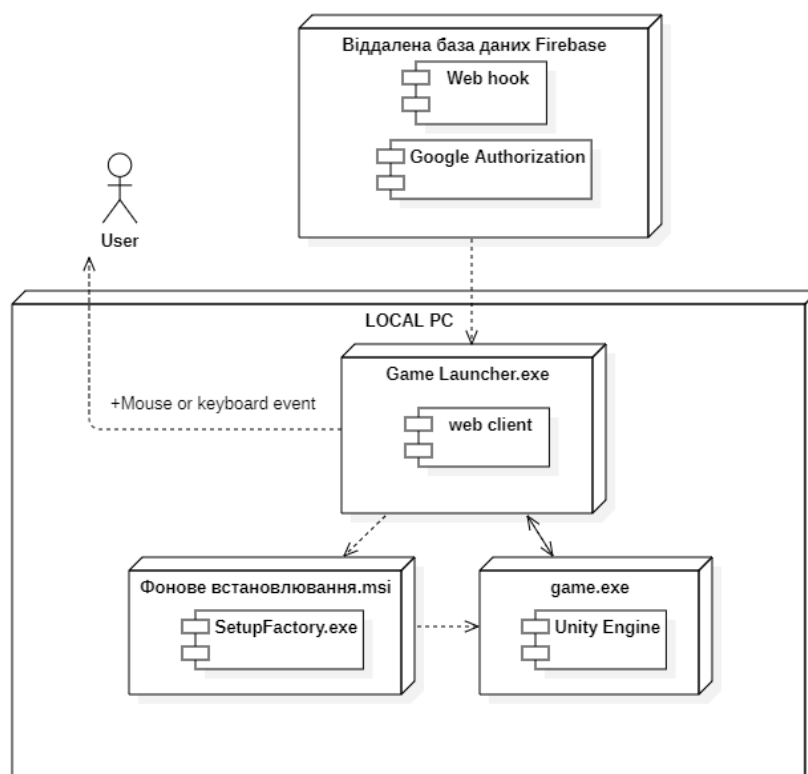


Рисунок 2.7 – Діаграма розгортання

Система, яка використовує архітектурну парадигму SEC (System-Entity-Component), є основою на специфічних принципах. Ця парадигма розподіляє функціональні компоненти на три основні типи: системи (Systems), сутності (Entities) і компоненти (Components). В контексті розробки великих ігрових систем, використання парадигми SEC є широко поширеним.

У рамках парадигми SEC, системи відповідають за виконання конкретних функцій та операцій в системі гри. Вони керують компонентами і сутностями, виконуючи необхідні обчислення та логіку гри.

Сутності відображають реальні об'єкти або ігрові об'єкти в системі. Вони мають характеристики, стан та поведінку, які можуть бути змінюваними в процесі гри. Сутності зазвичай містять компоненти, які додають до них специфічну функціональність.

Використання парадигми SEC дозволяє забезпечити чітку структуру та організацію великих ігрових систем. Цей підхід допомагає зробити код більш модульним, повторно використовуваним та піддаємым змінам, спрощуючи розробку та підтримку гри "Dark Watcher".

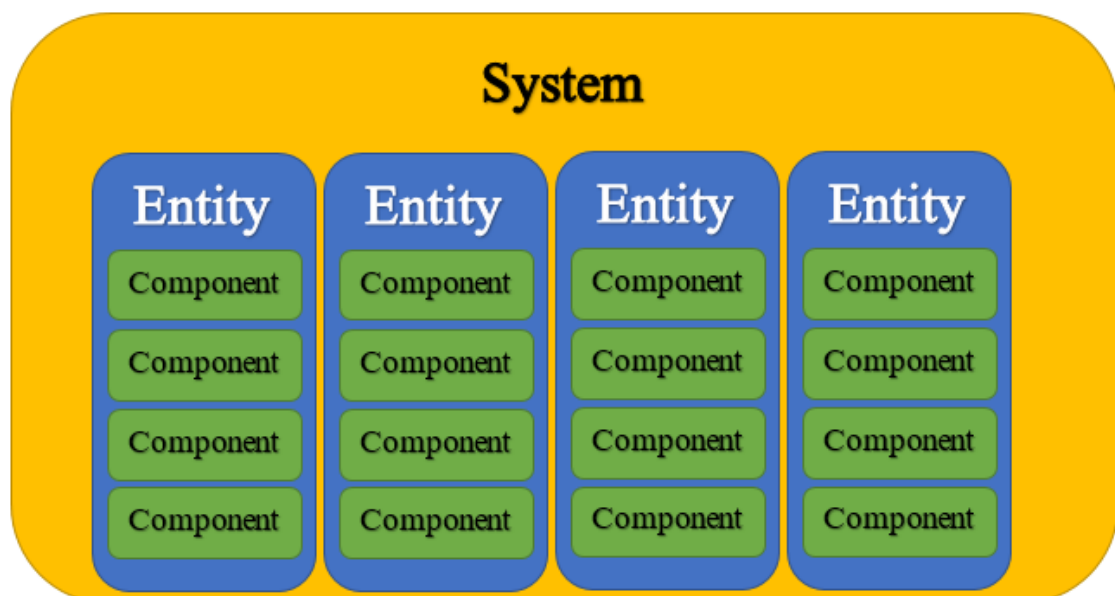


Рис 2.8. SEC-патерн проектування

2.4 Реалізація ігрового додатку

Реалізація ігрового додатку є процесом перетворення концепції та дизайну гри в функціонуючий ігровий продукт. Це включає в себе програмування, створення графіки, звуку, анімації та інших елементів, необхідних для створення ігрового досвіду.

Реалізація ігрового додатку "Dark Watcher" базується на використанні інструменту розробки ігор - Unity. Unity надає інтегровану середу розробки (IDE), яка

дозволяє створювати ігрові об'єкти, реалізовувати логіку гри та взаємодію з гравцем.

Процес реалізації ігрового додатку включає такі етапи, як:

- створюють графічних об'єктів, які використовуються у грі, такі як персонажі, оточення, предмети тощо. Вони моделюють геометрію та встановлюють текстури для кожного об'єкта;
- використання мови програмування C# для реалізації логіки гри, обробки введення гравця, керування рухом об'єктів, обробки колізій та багато іншого.
- створення анімації для персонажів та інших об'єктів у грі. Це включає рухи, взаємодію з об'єктами, ефекти та інші анімаційні дії.
- додавання звукових ефектів, фонові музики та голосового озвучення до гри. Це допомагає створити атмосферу та підсилює зацікавлення гравця.

Реалізація гри включає в себе ітеративний процес тестування, виявлення помилок та налагодження програмного коду, щоб забезпечити належну функціональність ігрового додатку.

В процесі реалізації ігрового додатку "Dark Watcher" було враховано вимоги геймплею, дизайну рівнів, візуального оформлення та оптимізації для досягнення бажаного ігрового досвіду. Було зосереджено на реалізації захоплюючої геймплейної механіки, використанні ефектів та атмосфери, які поглиблюють взаємодію гравця з грою.

Було реалізовано клас «Menu», відповідальний за основні системні налаштування. Цей клас керує та зберігає налаштування, які впливають на поведінку і функціональність системи. Він забезпечує можливість зміни налаштувань та їх збереження для подальшого використання. Клас включає в себе різноманітні параметри, такі як роздільна здатність, рівень складності, налаштування звуку та графіки. Цей клас дозволяє легко змінювати налаштування системи, що допомагає відповідати вимогам і потребам гравців.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		53

Таблиця 2.2 Основні методи класу Menu

Назва методу	Відповідальність методу
NewGameLoader()	Створення екземпляру нової гри.
doExitGame()	Завершення процесу гри. Закриття Exit файлу.
Start()	Зчитання з операційної системи списку роздільної здатності та виведення їх у викидне меню.
SetResolution (int index)	Встановлення обраної зі списку роздільної здатності.
SetVolume(float volume)	Встановлення обраного рівня гучності.
Set Quality(int index)	Вибір якості промальовки графіки.
SetFullscreen(bool isFullscreen)	Зміна повноекранного режиму.
CloseSettings()	Завершення налаштувань.
openSettings()	Відкриття налаштувань.
SaveName()	Встановлення імені персонажа для лінійних діалогів.
ShowLicenseAgreement()	Відображення вікна для введення імені.

Було реалізовано клас «MenuNewGame», який відповідає за створення нової гри. Цей клас містить необхідну логіку та функціональність, пов'язані з процесом створення гри. Він взаємодіє з іншими компонентами системи, щоб забезпечити безперебійне створення нових ігрових сесій. Клас отримує вхідні дані від гравця, які включають налаштування гри, і виконує необхідні дії для створення гри з відповідними параметрами. Крім того, клас забезпечує перехід до режиму гри після успішного створення, що дозволяє гравцю насолоджуватись новою грою.

Таблиця 2.3 Основні методи класу MenuNewGame

Назва методу	Відповідальність методу
NewGameLoader()	Навантажувач нової гри.

Було реалізовано клас «PauseMenu», який відповідає за обслуговування меню паузи з логікою та роботою з інтерфейсом користувача (UI). Цей клас забезпечує функціональність, пов'язану з відображенням та управлінням меню паузи в грі. Він взаємодіє з компонентами інтерфейсу користувача, такими як кнопки, панелі та текстові поля, для забезпечення відповідного візуального відображення та реагування на взаємодію користувача. Клас також містить логіку, пов'язану з обробкою подій та виконанням дій при виборі пунктів меню паузи. Наприклад, він може відповідати за призупинення гри, зміну налаштувань, відкриття додаткових панелей тощо. Цей клас допомагає забезпечити зручне та інтуїтивно зрозуміле користування меню паузи в грі.

Таблиця 2.4 Основні методи класу PauseMenu

Назва методу	Відповідальність методу
Update()	Зміна стану паузи при натисканні клавіші.
Resume()	Відновлення гри.
Pause()	Активація стану паузи
ToMenu()	Вихід до головного меню.
ExitGame()	Вихід з гри.

Було реалізовано клас «CutsceneCamera», який відповідає за управління міні-катсценами та переміщення камери до об'єкту катсцени. Цей клас забезпечує функціональність, пов'язану зі зміною активних катсцен та відображенням їх на екрані. Клас також включає функціональність, що дозволяє контролювати рух камери та зміну її позиції та орієнтації. Наприклад, він може виконувати зумування камери, слідування за рухом об'єкту чи зміну кута огляду. Цей клас допомагає забезпечити

плавну та контрольовану навігацію між міні-катсценами та покращує взаємодію гравця з грою.

Таблиця 2.5 Основні методи класу CutsceneCamera

Назва методу	Відповідальність методу
Update()	Переміщення камери до об'єкта фіксації.

Було реалізовано клас «DarkingTrigger», який відповідає за управління подією зіткнення з об'єктом активації виходу з катсцени. Цей клас забезпечує функціональність, пов'язану з обробкою взаємодії гравця з об'єктом активації виходу, що вказує на перехід в ігровий режим.

Таблиця 2.6 Основні методи класу DarkingTrigger

Назва методу	Відповідальність методу
OnTriggerEnter2D()	При зіткненні з об'єктом виконується анімація затемнення.

Було реалізовано приклад класу «DeerCutscene», який відповідає за управління основним сюжетним ігровим об'єктом під час катсцени. Цей клас забезпечує функціональність, пов'язану з рухом, поведінкою та взаємодією гравця з основним сюжетним об'єктом. Він включає логіку, що керує рухом об'єкта, оброблює взаємодію з іншими об'єктами в середовищі гри, а також виконує обробку введення гравця для управління об'єктом. Клас також може включати додаткову логіку, пов'язану з анімаціями, звуковими ефектами та іншими аспектами пов'язаними з основним сюжетним об'єктом. Цей клас допомагає забезпечити плавний рух, взаємодію та керування основним сюжетним об'єктом під час катсцени.

Таблиця 2.7 Основні методи класу DeerCutscene

Назва методу	Відповідальність методу
Start()	Встановлення станів діалогу катсцени.
Update()	В залежності від станів діалогу запуск наступного діалогу та анімацій.
CinematicEnd()	Анімаційне віддалення камери.
DeerSpawn()	Телепортація об'єкта на задану точку.
DeerRun()	Пересування об'єкта по карті.
CameraZoomDeer()	Активація наближення віртуальної камери до ігрового об'єкту.

Було реалізовано приклад класу «EnvironmentCutscene», який відповідає за управління оточенням для катсцени. Цей клас забезпечує функціональність, пов'язану зі створенням та керуванням оточенням, яке оточує гравця під час катсцени. Він включає логіку, що керує розміщенням та розміром об'єктів усередині оточення, відтворенням звукових ефектів, реалістичною обробкою світла та тіней, а також відображенням інших елементів, що створюють атмосферу катсцени. Клас також може включати додаткову логіку, пов'язану зі зміною оточення відповідно до дій гравця або на основі зміни контексту катсцени.

Таблиця 2.8 Основні методи класу EnvironmentCutscene

Назва методу	Відповідальність методу
Start()	Отримання ігрового об'єкту.
TriggerDialog()	Запуск непроговореного діалогу.

Був реалізований клас «ExitFromCutscene», який відповідає за управління виходом з катсцени. Клас також включає відображенням анімаційних ефектів та повернення камери у попереднє положення.

Таблиця 2.9 Основні методи класу ExitFromCutscene

Назва методу	Відповідальність методу
OnTriggerEnter2D(Collider2D collider)	Активація виходу з катсцени.
ChangeScene()	Зміна сцени.
CinematicEnd()	Анімаційне віддалення камери.
DoDeerSpawn()	Телепортація ігрового об'єкта.
UnBlockDialogueSystem()	Розблокування діалогової системи, якщо вона заблокована.

Було реалізований приклад класу «PlayerCutscene», який керує гравцем під час катсцени. Під час катсцени камера віддаляється для створення спеціальної сцени та може відбуватися затемнення для передачі сюжету гравцю.

Таблиця 2.10 Основні методи класу PlayerCutscene

Назва методу	Відповідальність методу
CameraZoomOut()	Віддалення камери.
DarkingStart()	Анімаційне затемнення екрану.

Клас «CanvasCutsceneManager» має функціональність, пов'язану з запуском різних катсцен на полотні гри. Він включає логіку, що керує відображенням та зміною активних катсцен, забезпечує плавні переходи між ними та управляє потоком подій та дій, які відбуваються під час різних катсцен. Клас також може включати додаткові функції, такі як збереження та завантаження стану катсцен, обробка звукових ефектів, анімація переходів та інші аспекти, що сприяють покращенню взаємодії з грою.

Таблиця 2.11 Основні методи класу CanvasCutsceneManager

Назва методу	Відповідальність методу
Start()	Деактивація полотна для катсцени.
IEnumerator PlayCutscene()	Асинхронний метод для програвання катсцени на полотні.
FadeScreen(float targetAlpha, float duration)	Функція для управління затемнення.
IEnumerator Fade(Image image, float startAlpha, float targetAlpha, float duration)	Асинхронний метод для програвання анімацій затемнення.
PlayAudio()	Відтворення звуку

Клас «CutsceneScript» відповідає за ініціалізацію та запуск катсцен на полотні, забезпечуючи їх коректне відображення та взаємодію з гравцем.

Таблиця 2.12 Основні методи класу CutsceneScript

Назва методу	Відповідальність методу
OnTriggerEnter2D(Collider2D collision)	Запуск катсцени при зіткненні з об'єктом
OnButtonClick()	Запуск катсцени по натисканню на об'єкт.
CutsceneStarter(CanvasKeeper obj)	Запуск катсцени та її налаштування.
CutsceneStarter(Button obj)	

Було розроблено клас «CheckPoint», який виконує функцію створення об'єктів для файлового збереження. Цей клас дозволяє зручно створювати та керувати об'єктами, які здатні зберігати дані в файловій системі. Він зчитує та записує дані у ці файли.

Таблиця 2.13 Основні методи класу CheckPoint

Назва методу	Відповідальність методу
Checkpoint(PlayerHealth playerHealthScript, Transform userTransform, Inventory userInventory)	Створення об'єкту файлового збереження.

Було розроблено клас «CheckpointTrigger», що виконує функцію управління анімаціями знака збереження. Цей клас забезпечує можливість створення та відтворення анімаційного ефекту, та функції збереження прогресу гри гравця пов'язаного зі знаком.

Таблиця 2.14 Основні методи класу CheckpointTrigger

Назва методу	Відповідальність методу
OnTriggeerEnter2D()	Запуск перевірки знаходження гравця.
OnTriggerStay2D()	При знаходженні зони, при натисканні кнопки взаємодії, відбувається збереження персонажу.
OnTriggerExit2D()	При виході з зони прибирається підказки.
FlipButton()	Перевірка знаходження гравця.
MenuLoadButton()	Завантаження кнопок меню.

Була реалізована функція «LookAtPlayer», що відповідає за управління поворотами та вибором різних підтипів знаку анімації. Ця функція дозволяє змінювати орієнтацію знаку відносно вказаної осі або точки повороту до гравця.

Таблиця 2.15 Основні методи класу LookAtPlayer

Назва методу	Відповідальність методу
SwitchSide()	Розвертає спрайт.
Blink()	Анімація кліпання.
Update()	Запуск різних анімацій кліпання.

Висновки до розділу 2

У даному розділі були розглянуті основні аспекти створення ігрового додатку. Перш за все, проведено аналіз та визначено варіанти використання та об'єктно-орієнтовану структуру системи. Це дозволило визначити ключові компоненти та їх взаємозв'язки, що були використані для подальшого проектування.

На основі отриманих результатів було сформульоване технічне завдання, яке визначало вимоги та функціональність системи. Це стало основою для розробки структури дизайну гри, включаючи інтерфейс гравця, систему анімацій та системні налаштування. Важливою частиною проектування була розробка подій, що визначали взаємодію гравця з катсценою та реагування системи на ці події.

Для ефективної роботи гри “Dark Watcher” було проведено проектування та реалізацію алгоритмів, які забезпечували потрібну функціональність, таку як керування рухом об'єктів, зіткнення та взаємодія з об'єктами гри. Застосування відповідних алгоритмів дало змогу забезпечити плавність та реалістичність геймплею.

В результаті проведених досліджень та розробок було успішно реалізовано ігровий додаток. Його реалізація включала в себе роботу з подіями, системою анімацій та системними налаштуваннями. Отриманий додаток відповідає поставленим завданням та вимогам технічного завдання.

Таким чином, проектування та реалізація ігрового додатку були успішно завершені, а отримані результати дозволяють стверджувати про досягнення поставлених цілей. Подальші можливості розвитку та удосконалення додатку включають розширення функціональності, оптимізацію алгоритмів та поліпшення візуального вигляду.

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ГРОЮ В ЖАНРІ МЕТРОЇДВАННЯ “DARK WATCHER”

3.1 Порядок встановлення та налаштування системи

Гра "Dark Watcher" розроблена на рушії Unity з підтримкою будь-якої операційної системи. Користувач може знайти її у онлайн-сервісі Steam, яка підтримує операційні системи Windows і MacOS та інші. Процес встановлення та налаштування системи забезпечує оптимальний ігровий досвід для гравців.

Для встановлення "Dark Watcher" на Windows або MacOS користувачі можуть перейти в магазин Steam та знайти гру через стрічку пошуку за назвою гри. Після придбання гри вона буде додана до бібліотеки Steam. Встановлення починається з вибору "Dark Watcher" у бібліотеці та натискання кнопки "Встановити". Steam автоматично завантажить та встановить необхідні файли.

Перед запуском гри важливо перевірити, чи система відповідає мінімальним вимогам, включаючи версію операційної системи, швидкість процесора, обсяг пам'яті (RAM) та вимоги до графічної карти. Це забезпечує ефективну роботу системи під час гри.

Приклад оптимальних вимог з операційною системою Windows для досягнення мінімум 120 кадрів в секунду (fps):

1. Процесор (CPU): Intel Core i7 або AMD Ryzen 7 з високою частотою роботи.
2. Графічна карта (GPU): NVIDIA GeForce GTX 1660 Super або AMD Radeon RX 5700 з великим обсягом відеопам'яті.
3. Оперативна пам'ять (RAM): мінімум 8 гігабайт (ГБ) оперативної пам'яті.
4. Жорсткий диск (HDD) або твердотільний накопичувач (SSD): рекомендовано використовувати SSD для швидкого завантаження ігор та швидкого доступу до даних.
5. Монітор: роздільна здатність Full HD (1920x1080) з частотою оновлення 120 Гц або вище для показу 120 fps.
6. Операційна система: Windows 11 64-біт.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		62

7. Живлення: достатньо потужний блок живлення для живлення всієї системи, з урахуванням споживання енергії від компонентів.

Також слід врахувати, що вказані вимоги є лише рекомендацією і фактичні або мінімальні вимоги були вказані раніше, на етапі проектування. Слідуючи інструкціям щодо встановлення, перевіряючи вимоги системи та налаштовуючи параметри гри, можна насолоджуватися гладким і захоплюючим ігровим досвідом.

3.2 Структура інтерфейсу. Інтерфейс та порядок роботи з системою

У даному підрозділі буде описана частина роботи, яка була розроблена під тему даної кваліфікаційної роботи, інші частини проекту можна побачити у кваліфікаційних роботах решти команди.

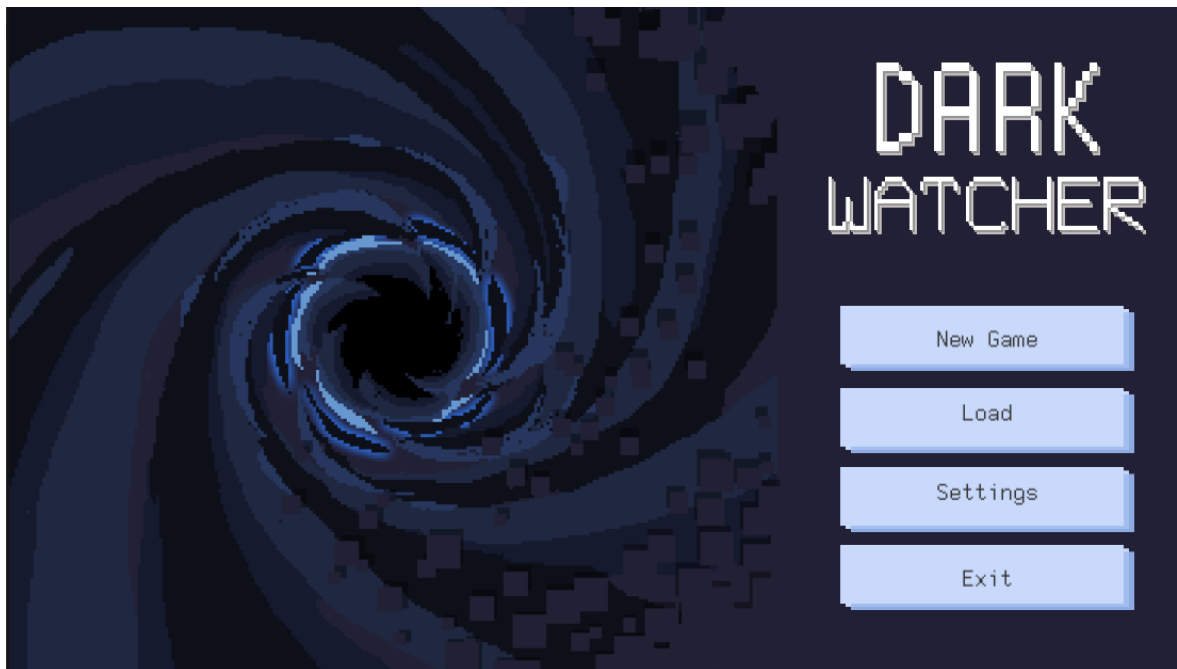


Рисунок 3.1 – Демонстрація вигляду головного меню

Після виконання входу у лаунчері, гравцю відображається головне меню гри “Dark Watcher” (див. рис.3.1). Головне меню надає різні опції для гравця, які включають:

- кнопка «Нова гра» дозволяє гравцеві розпочати нову гру з початку (див. рис.3.2). Щоб розпочати гру, потрібно ввести ім'я головного героя та натиснути кнопку «Почати».
- кнопка «Завантаження створеної гри»: якщо гравець вже розпочав гру і зберіг свій прогрес, він може скористатися цією опцією, щоб завантажити

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		63

збережену гру та продовжити грати з місця, на якому він зупинився раніше.

- кнопка «Налаштування» дозволяє гравцю налаштувати різні параметри гри, такі як графіка, звук, роздільна здатність та інші.
- кнопка «Вихід»: Якщо гравець бажає завершити гру і повернутися до лаунчера або вийти з гри повністю, він може скористатися цією опцією. Вибір цієї опції вимкне гру та поверне гравця до вихідного екрану.

Головне меню гри "Dark Watcher" надає гравцеві зручний спосіб управління грою, дозволяючи йому легко навігувати та користуватися різними функціями гри.



Рисунок 3.2 – Демонстрація початку нової гри

Після натиснення кнопки "Налаштування" гравець може побачити додаткові опції (див. рис.3.3), які дозволяють налаштувати гру під свої потреби і вподобання. Деякі з цих опцій включають:

- опція «Роздільна здатність»: гравець може вибрати оптимальну роздільну здатність (див. рис.3.4), яка відповідає його монітору або екрану. Це дозволяє забезпечити чітке відображення гри.
- опція «Гучність звуку»: гравець може регулювати гучність звукових ефектів та музики у грі. Це дозволяє налаштувати аудіо так, щоб воно було комфортним і відповідало особистим вподобанням гравця.

- опції «Графіка»: в налаштуваннях графіки гравець може змінити якість відображення графіки гри, аби не навантажувати свій пристрій.
- опція «Повноекранний режим»: гравець може вибрати, чи грати в гру "Dark Watcher" в повноекранному режимі або в віконному режимі. Повноекранний режим може забезпечити більш глибокий досвід, тоді як віконний режим може бути зручним для мультитаскінгу.

Ці опції налаштувань дозволяють гравцеві персоналізувати гру "Dark Watcher" на основі його власних вподобань і можливостей системи. Це допомагає забезпечити оптимальний ігровий досвід, відповідний потребам кожного гравця.



Рисунок 3.3 – Демонстрація системних налаштувань

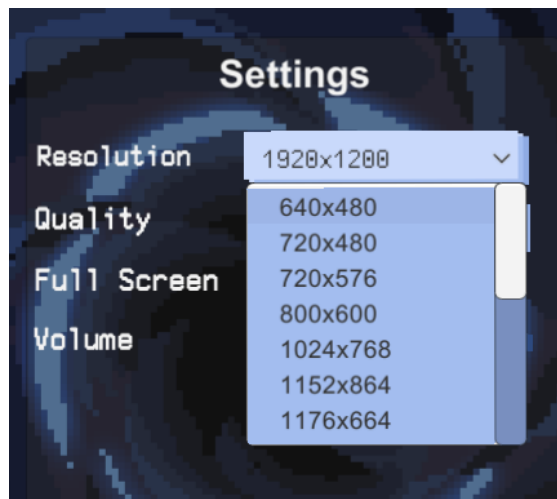


Рисунок 3.4 – Демонстрація зміни однієї з опцій налаштувань

На рисунку 3.5 представлене діалогове вікно, яке використовується для позначення катсцени. Цей елемент дозволяє гравцю слідкувати за подіями під час катсцени.

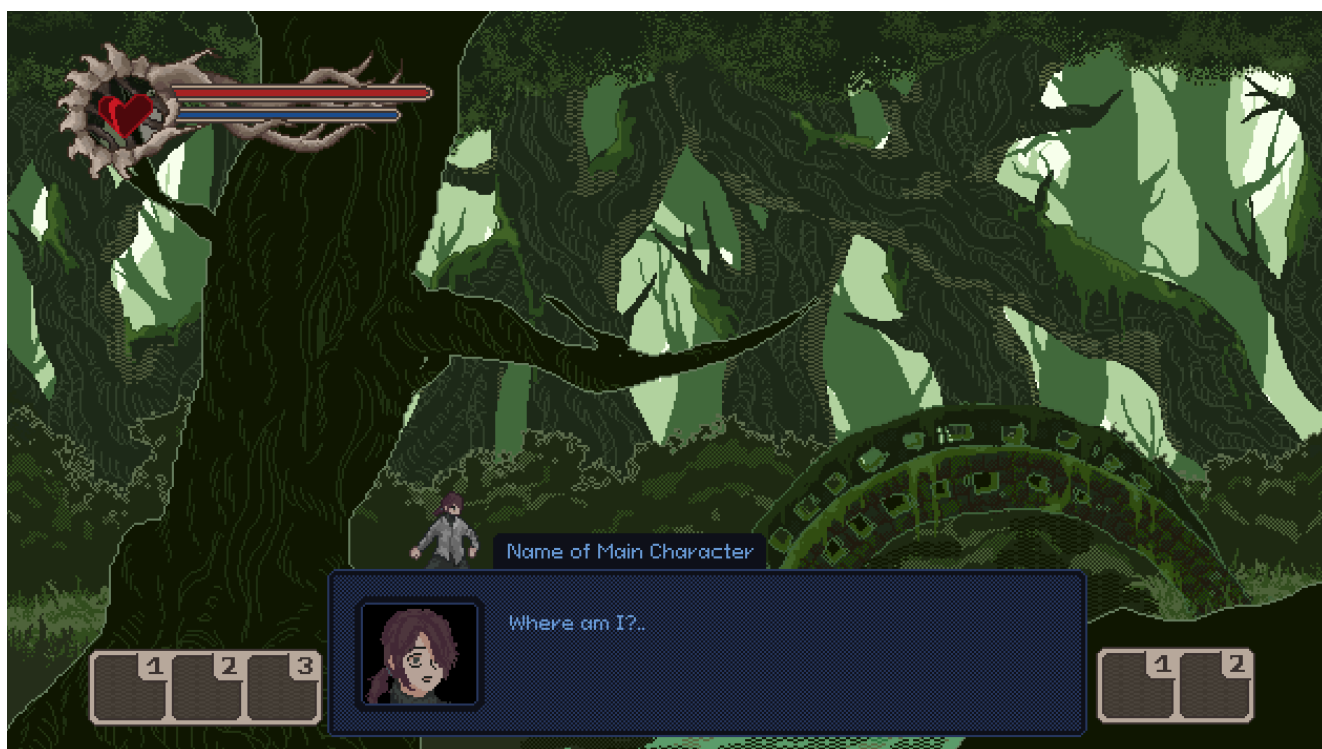


Рисунок 3.5 – Демонстрація вигляду звичайної катсцени з діалогом

На рисунку 3.6 представлений початок спеціальної сюжетної катсцени з появою нового персонажу. На катсцені можна помітити, що камера віддалена від основного персонажа та фокусується на сцені, де розгортається подія. Це дозволяє створити ефект глибини та показати широкий кадр, щоб гравець міг сприйняти всю обстановку.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		66



Рисунок 3.6 – Демонстрація сюжетної катсцени

На рисунку 3.7 та 3.8 відображається головний екран ігрового процесу та подальше його затемнення, яке свідчить про завершення катсцени.



Рисунок 3.7 – Демонстрація вигляду сцени до затемнення

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		67



Рисунок 3.8 – Демонстрація вигляду сцени після затемнення

На рисунку 3.9 відображено героя, який підійшов достатньо близько до триггеру катсцени, тому відповідно катсцена розпочалася і її учасники почали рух згідно прописаного сценарію.



Рисунок 3.9 – Демонстрація задіяння триггеру катсцени

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		68

3.3 Тестування роботи системи

В процесі розробки програмного коду, тестування проекту стає необхідним етапом, що допомагає переконатися в правильному та безперебійному функціонуванні системи.

У випадку тестування катсцен та налаштувань гри "Dark Watcher", цей процес відрізняється від тестування веб-ресурсів, мобільних додатків та інших продуктів, оскільки вимагає специфічного підходу.

Тестування гри "Dark Watcher" включає наступні основні види:

- тестування функціональності катсцен: цей вид тестування включає перевірку коректності та взаємодії різних функцій катсцен, таких як відтворення анімації, діалогів, перехід між сценами, реакція на події гравця.
- тестування ефектів та візуального контенту: цей вид тестування передбачає перевірку відтворення ефектів, візуальних елементів та контенту, що присутній у катсценах. Ефекти повинні відображатися правильно, об'єкти виглядати реалістично, освітлення відповідає налаштуванням та інші візуальні аспекти.
- тестування логіки та взаємодії: цей вид тестування спрямований на перевірку логіки катсцен та взаємодії об'єктів. Він включає перевірку правильності послідовності подій, правильності виконання завдань та інших аспектів, що впливають на геймплей.
- тестування налаштувань та конфігурації: цей вид тестування спрямований на перевірку правильності роботи налаштувань і конфігурацій. Він включає перевірку налаштувань роздільної здатності екрану, гучності звуку, графічних параметрів, повноекранного режиму та інших параметрів, які можуть бути налаштовані гравцем.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		69

– Тестування функціональності катсцен:

Таблиця 3.1 Тест-кейс функціональності катсцен

Test name: Cutscene sequence (Issue id #21)			
Status	Passed		
Created date	2023-06-07	Tested at	2023-06-08
Reporter	Leut Nazarii	Assigned to	Oksana Bukhanevych
Pre Conditions: Authorization/Registration, game started			
Steps	Description	Expected results	
1	Launch the game and navigate to a scene with a cutscene.	The game successfully loads the scene with the cutscene.	
2	Trigger the start of the cutscene by performing the required action or reaching the specific area.	The cutscene begins playing as intended.	
3	Observe the animations, dialogues, and transitions between scenes during the cutscene.	The animations are smooth and visually appealing. The dialogues are displayed correctly with proper timing. Transitions between scenes are seamless.	
4	Interact with the cutscene by performing various player events, such as pressing specific buttons.	The cutscene reacts appropriately to player events, triggering the intended actions or dialogues.	
5	Verify the correctness of the accuracy of animations, dialogue sequences, and scene transitions.	All animations are displayed accurately, dialogues follow the correct sequence.	
6	Complete the cutscene and ensure the game returns to the normal gameplay state.	After the cutscene ends, the game transitions back to the regular gameplay.	

– Тестування ефектів та візуального контенту:

Таблиця 3.2 Тест-кейс ефектів та візуального контенту

Test name: Cutscene visual effects (Issue id #22)			
Status	Passed		
Created date	2023-06-05	Tested at	2023-06-06
Reporter	Leut Nazarii	Assigned to	Oksana Bukhanevych
Pre Conditions: Authorization/Registration, game started			
Steps	Description	Expected results	
1	Launch the game and navigate to a scene with visual effects and content, such as a cutscene.	The game successfully loads the scene with the desired visual elements and effects.	
2	Observe the visual effects used in the cutscene, including character-specific effects.	The visual effects are displayed accurately, matching the intended design.	
3	Inspect the visual elements and objects present in the cutscene, such as characters, objects, and environments.	The visual elements appear with proper textures, shading, and proportions.	

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		70

4	Assess the lighting and ambiance in the cutscene to ensure it aligns with the desired setting and atmosphere.	The lighting effects match the scene's mood and environment, enhancing the overall visual presentation. Shadows, reflections, and other lighting effects should be appropriately rendered.
5	Check for any visual glitches, graphical artifacts, or inconsistencies in the visual content.	The visuals should be free from glitches, artifacts, or any visual abnormalities that may detract from the immersive experience. All visual elements should appear as intended without any graphical distortions.
6	Verify that the visual content and effects are synchronized properly with other components, such as animations, sound effects, and dialogues.	The visual effects and content should seamlessly integrate with other elements of the cutscene, creating a cohesive and engaging experience. All components should be synchronized accurately and in harmony.

– Тестування логіки та взаємодії:

Таблиця 3.3 Тест-кейс логіки та взаємодії

Test name: Cutscene logic (Issue id #23)			
Status	Passed		
Created date	2023-06-07	Tested at	2023-06-08
Reporter	Leut Nazarii	Assigned to	Oksana Bukhanevych
Pre Conditions: Authorization/Registration, game started			
Steps	Description	Expected results	
1	Launch the game and navigate to a scene with a cutscene that involves logic and object interaction.	The game successfully loads the scene with the cutscene and related objects.	
2	Observe the sequence of events in the cutscene and verify that they occur in the intended order.	The events unfold in a logical and coherent manner, following the desired storyline or scenario.	
3	Interact with the objects present in the cutscene according to the given tasks or objectives.	The objects respond appropriately to the player's interactions, triggering the expected actions or reactions.	
4	Verify the correctness of the tasks and objectives presented within the cutscene.	The tasks or objectives are clear, achievable, and aligned with the overall gameplay and narrative.	

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		71

5	Check for any logical inconsistencies or errors in the cutscene's flow or object interactions.	The cutscene should not contain any illogical or contradictory elements that disrupt the immersion or gameplay experience.
6	Ensure that the cutscene's logic and object interactions align with the larger game mechanics and systems.	The cutscene's logic and object interactions should integrate seamlessly with the overall gameplay, maintaining consistency and coherence throughout the player's experience.

– Тестування налаштувань та конфігурації:

Таблиця 3.4 Тест-кейс налаштування та конфігурації

Test name: Configuration (Issue id #24)			
Status	Passed		
Created date	2023-06-01	Tested at	2023-06-03
Reporter	Leut Nazarii	Assigned to	Oksana Bukhanevych
Pre Conditions: Authorization/Registration			
Steps	Descripiron	Expected results	
1	Launch the game and access the settings or configuration menu.	The game successfully loads the settings or configuration menu, providing options for adjustment.	
2	Verify the functionality of the screen resolution setting by selecting different resolution options.	The game's display adjusts accordingly, maintaining the correct aspect ratio and visual quality for the selected resolution.	
3	Adjust the sound volume settings and verify that the changes are reflected in the game's audio output.	The sound volume should increase or decrease as per the adjustments made, ensuring a comfortable and balanced audio experience.	
4	Test the graphics settings by selecting different options, such as low, medium, and high graphics quality.	The game's graphics should adjust accordingly, reflecting the chosen settings with changes in visual fidelity, texture details, and special effects.	
5	Toggle the full screen mode setting on and off to verify its functionality.	The game should seamlessly switch between full screen and windowed mode, with the visuals properly adapting to the selected mode without any display issues.	

Висновки до розділу 3

У цьому розділі описано процес встановлення та налаштування гри "Dark Watcher" на операційних системах Windows та MacOS. Також вказано системні вимоги, рекомендовані для забезпечення комфортної гри.

Гра "Dark Watcher" має реалізовану систему відтворення катсцен. Система відтворення катсцен включає анімації, діалоги, переходи між сценами та реакції на події гравця. Ці елементи були ретельно розроблені та впроваджені для забезпечення захоплюючого ігрового досвіду.

Крім того, гра має можливість налаштування різних системних параметрів, таких як роздільна здатність екрану, гучність звуку, налаштування графіки та режим повноекранного відображення. Гравець може змінювати ці налаштування, щоб відповідати своїм уподобанням та потребам.

Під час розробки було реалізовано всі необхідні функції та функціональність системи відтворення катсцен та налаштувань. Весь код був перевірений та протестований, щоб переконатися в його правильному функціонуванні. Результатом цього процесу є гра "Dark Watcher", яка працює належним чином і задовольняє поставлені завдання.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		73

ВИСНОВКИ

У кваліфікаційній роботі було проведено аналіз різних напрямків використання інформаційних технологій для реалізації комп'ютерної гри в жанрі метроїдванія. Дослідження охоплювало широкий спектр технологій, включаючи графічний двигун Unity, мову програмування C# Mono, а також інструмент Aseprite для розробки анімацій. Цей аналіз дозволив визначити оптимальний набір технологій для реалізації гри "Dark Watcher" у жанрі метроїдванія.

Було розглянуто проектування програмного забезпечення гри. В результаті була створена архітектура програми, яка включала різні модулі та компоненти для керування різними аспектами гри, такими як керування сюжетними катсценами трьох типів, взаємодію з системним налаштуванням, обробка взаємодії з об'єктами катсцени, тощо. Це забезпечувало ефективне проектування коду та зручність у розробці нових функцій.

Окрема увага була приділена розробці структури дизайну та порядку роботи з грою в жанрі метроїдванія "Dark Watcher". Був розроблений інтуїтивно зрозуміле та зручне меню налаштувань, що дозволяє гравцям легко підлаштовувати візуальне відтворення гри на власне бачення. Було створено послідовність подій та розроблений процес відтворення катсцен та забезпечено коректну роботу відтворення анімацій усіх об'єктів.

Отже, результати кваліфікаційної роботи підтверджують успішне використання інформаційних технологій для реалізації комп'ютерної гри в жанрі метроїдванія "Dark Watcher". Розроблене програмне забезпечення демонструє ефективну архітектуру, зручний інтерфейс та надійну роботу, що задовольняють потреби гравців.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		74

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Dean J. Unity Character Animation with Mecanim / Jamie Dean. – New York: Packt, 2015. – 290 с. – (9781849696364).
2. Halpern J. Developing 2D Games with Unity / Jared Halpern. – New York: Apress, 2019. – 398 с. – (978-1-4842-3771-7).
3. Learning C# by Developing Games with Unity – Birmingham: Packt Publishing, 2019. – 532 с. – (4). – (978-1-78953-205-0).
4. Price M. C# 11 and .NET 7 – Modern Cross-Platform Development Fundamentals: Start building websites and services with ASP.NET Core 7, Blazor, and EF Core 7, 7th Edition / Mark J. Price. – New York: Packt, 2022. – 818 с. – (1803237805).
5. Unity Technologies. Unity Manual [Електронний ресурс] / Unity Technologies // 1. – 2022. – Режим доступу до ресурсу: <https://docs.unity3d.com/Manual/index.html>.
6. Azzi M. Pixeland Logic: A guide to pixel art / Michael Azzi., 2019. – 242 с. – (1).
7. Thorn A. Unity Animation Essentials / AlanThorn. – NEW YORK: PacktPublishing, 2016. – 176 с. – (978-5-97060-377-2).
8. Okita A. Learning C# Programming with Unity 3D / AlexOkita. – 6000 BrokenSoundParkway NW, Suite 300: Taylor&FrancisGroup, 2015. – 686 с. – (978-1-4665-8653-6).
9. Ferrone H. Learning C# by Developing Games with Unity 2020 / HarrisonFerrone. – LiveryPlace 35 LiveryStreetBirmingham B3 2PB, U: PacktPublishingLtd., 2020. – 455 с. – (978-1-80020-780-6).
10. Костер Р. Книга Теорія розваг для ігрового дизайну / Раф Костер. – Київ: ArtHuss, 2023. – 288 с. – (9786178025311).
11. Aseprite Help [Електронний ресурс] // – 2022. – Режим доступу до ресурсу: <https://www.aseprite.org/docs/>.
12. Falzon E. Unity for Absolute Beginners - Getting Started [Електронний ресурс] / Edward Falzon // Coursera Plus. – 2023. – Режим доступу до ресурсу: <https://www.coursera.org/projects/unity-absolute-beginners-getting-started>.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		75

13. Ager R. Unity 2D Dungeon Gunner Roguelike Development Course [Электрон-
ный ресурс] / Rob Ager // Udemу. – 2023. – Режим доступа до ресурсу:
[https://www.udemy.com/course/unity-2d-dungeon-gunner-roguelike-
development-course/](https://www.udemy.com/course/unity-2d-dungeon-gunner-roguelike-development-course/).
14. Ager R. Unity 2D Game Developer Course Farming RPG [Электронный ресурс]
/ Rob Ager // Udemу. – 2023. – Режим доступа до ресурсу:
<https://www.udemy.com/course/unity-2d-game-developer-course-farming-rpg/>.
15. Ghukasyan A. Learn C# By Building Applications [Электронный ресурс] /
Avetis Ghukasyan // Udemу. – 2023. – Режим доступа до ресурсу:
<https://www.udemy.com/course/learn-csharp-by-building-applications/>.
16. Unity Tutorials [Электронный ресурс] // Meetras. – 2023. – Режим доступа до
ресурсу:
[https://www.youtube.com/watch?v=UyUogO2DvwY&list=PLa8GtFRCBhkjsu8
w_0sgacti7HvcSZxM7&ab_channel=Meetras](https://www.youtube.com/watch?v=UyUogO2DvwY&list=PLa8GtFRCBhkjsu8w_0sgacti7HvcSZxM7&ab_channel=Meetras).

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		76

ДОДАТКИ

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		77

Додаток А. Відгуки на кваліфікаційну роботу



Вулиця Довженка, 64, Житомир, Житомирська область, 10000.

Тел. +380 067 411 5422.

email: info@inp-software.com

Шановна Оксано Буханевич,

Після детального огляду вашої роботи, я хотів би висловити свої враження та думки про вашу професійну діяльність. Ваші навички у розробці катсцен демонструють ваше зацікавлення та знання у галузі геймдевелопменту.

Ваші роботи в області катсцен вражають своєю деталізацією, естетикою та вмінням створювати вражаючі візуальні ефекти. Ви проявили глибоке розуміння наративу та геймплею, що дозволяє вам створювати незабутні ігрові досвіди для гравців.

Ваш досвід у розробці катсцен також допомагає вам у ролі Junior Unity Developer. Ви здатні ефективно впроваджувати свої дизайнерські ідеї за допомогою Unity та створювати функціональні ігрові сцени. Ваша здатність поєднувати дизайнерську естетику з технічним виконанням є цінним активом у галузі геймдевелопменту.

На основі представленої роботи, я можу охарактеризувати ваш рівень як Senior Game Designer та Junior Unity Developer. Ваша креативність та знання дизайну ігор підтверджують ваш досвід та експертизу у цій області. З одного боку, ви можете бути ведучим дизайнером гри, здатним керувати процесом розробки та надавати напрямок команді. З іншого боку, ваші навички у розробці на платформі Unity дають вам можливість самостійно реалізовувати свої ідеї у конкретних ігрових проектах.

Бажаю вам подальших успіхів у розвитку вашої кар'єри та надалі творчо реалізовувати свої ідеї у галузі геймдевелопменту.

Рекомендую оцінити роботу на високому рівні.

З повагою,

Представник компанії INP-SOFTWARE



Рудницький С. Г.

		О.А. Буханевич			ІПЗ.КР.Б – 121 – 23 – ПЗ	Арк.
		А.В. Морозов				
Змн.	Арк.	№ докум.	Підпис	Дата		78



КОМП'ЮТЕРНА АКАДЕМІЯ IT STEP

вул. Перемоги, буд. 29, м. Житомир, Україна, 10001

тел.: +38 (067) 303-05-33; +38(093) 087-30-33

e-mail: HYPERLINK "mailto:itstep.zhitomir@gmail.com" itstep.zhitomir@gmail.com

Шановна Оксано Буханевич,

Після детального огляду вашої роботи, ми хотіли б висловити наші враження та думки про вашу професійну діяльність. Наша команда визнає ваші навички у галузі розробки ігор та ваш внесок у створення катсцен.

Ваші знання та досвід у галузі геймдизайну вражають своєю глибиною та креативністю. Ви проявилися як справжній професіонал, здатний створювати захоплюючі та неповторні катсцени. Ваші талант та уважність до деталей стали ключовими елементами успішного втілення ідей у візуальний досвід.

Крім того, ми відзначаємо ваші навички у розробці на Unity. Ваш внесок у створення катсцен свідчить про вашу здатність ефективно використовувати цей інструментарій та реалізовувати комплексні ігрові механіки. Ваша робота свідчить про ваш потенціал у галузі розробки на Unity та вашу готовність до подальшого росту.

Узагалі, ваша робота підтверджує ваші здібності та професійну підготовку. Ви проявилися як творчий та досвідчений геймдизайнер, здатний досягти високих результатів у своїй галузі. На основі представленої вами роботи, ми можемо охарактеризувати ваш рівень як Senior Game Designer та Junior Unity Developer.

Ви володієте потенціалом стати цінним активом у галузі розробки ігор, і ми раді мати вас серед наших талановитих фахівців. Бажаємо вам успіху в подальшій кар'єрі та надаємо підтримку у вашому професійному зростанні.

Рекомендую оцінити роботу на високому рівні, у літерованій міжнародній системі, як 90A+.

З повагою,
Гармаш Олександр
Представник компанії ITSTEP Academy



		О.А. Буханевич		
		А.В. Морозов		
Змн.	Арк.	№ докум.	Підпис	Дата

ІПЗ.КР.Б – 121 – 23 – ПЗ

Арк.

79