

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.
_____ Ю. П. Кондратенко
« ____ » _____ 2022 р.

БАКАЛАВРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

РОЗРОБКА ВІЗУАЛЬНО-ІНТЕРАКТИВНОЇ RPG
НА ПЛАТФОРМІ UNITY

Спеціальність 122 «Комп'ютерні науки»

122 – БКР – 402.21810215

Виконав студент 4-го курсу, групи 402

_____ *Д. Ю. Коверзнієв*

«20» червня 2022 р.

Керівник: канд. техн. наук, доцент

_____ *Г.В. Кондратенко*

«20» червня 2022 р.

Миколаїв – 2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет ім. Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

Рівень вищої освіти бакалавр
Спеціальність 12 «Комп'ютерні науки»
(шифр і назва)
Галузь знань 12 «Інформаційні технології»
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.

_____ Ю. П. Кондратенко

«___» _____ 2021 р.

З А В Д А Н Н Я
на виконання кваліфікаційної роботи

Видано студенту групи 402 факультету комп'ютерних наук Коверзневу Дмитру Юрієвичу.

1. Тема кваліфікаційної роботи «Розробка візуально-інтерактивної RPG на платформі Unity».

Керівник роботи Кондратенко Галина Володимирівна, канд. техн. наук, доцент.

Затв. наказом Ректора ЧНУ ім. Петра Могили від «07» 12 2021 р. № 318

2. Строк представлення кваліфікаційної роботи студентом «___» ___ 20__ р.

3. Вхідні (початкові) дані до роботи: процес розробки візуально-інтерактивного застосунку; технології розробки застосунків.

Очікуваний результат: візуально-інтерактивний додаток жанру RPG на платформі Unity.

4. Перелік питань, що підлягають розробці (зміст пояснювальної записки):

- аналіз сучасного стану задачі розробки комп'ютерних ігор;
- вибір програмного забезпечення для розробки;
- реалізація візуально-інтерактивної RPG на платформі Unity;
- оптимізація та тестування створеного за стосунку;

5. Перелік графічного матеріалу: презентація.

6. Завдання до спеціальної частини: «Охорона праці при користуванні екранними пристроями»

7. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис
Спеціальна частина з охорони праці	к.т.н, доцент А. О. Алексєєва	

Керівник роботи канд. техн. наук, доцент Кондратенко Г. В.

(наук. ступінь, вчене звання, прізвище та ініціали)

(підпис)

Завдання прийнято до виконання Коверзнев Д.Ю.

(прізвище та ініціали)

(підпис)

Дата видачі завдання « 25 » листопада 2021 р.

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

Тема: Розробка візуально-інтерактивної RPG на платформі Unity

№	Найменування роботи	Початок	Закінчення	Примітки
1	Подання заяви на затвердження теми та керівників БКР	26.10.2021	30.10.2021	Виконано
2	Отримання завдання на виконання БКР	25.11.2021	25.11.2021	Виконано
3	Складання календарного плану роботи на весь період виконання БКР	10.12.2021	10.12.2021	Виконано
4	Отримання завдання на переддипломну практику	23.05.2022	23.05.2022	Виконано
5	Проходження переддипломної практики, збір та аналіз матеріалів до БКР	23.05.2022	04.06.2022	Виконано
6	Розробка звіту з переддипломної практики	04.06.2022	06.06.2022	Виконано
7	Проведення структурнофункціонального моделювання, Реалізація базових механічних дій, процедурної генерації та візуалізація додатку	28.02.2022 та 06.06.2022	27.03.2022 та 19.06.2022	Виконано
8	Попередній захист БКР на засіданні комісії кафедри	30.05.2022	31.05.2022	Виконано
9	Доробка та остаточне оформлення БКР	02.06.2022	20.06.2022	Виконано
10	Подання БКР рецензенту	16.06.2022	18.06.2022	Виконано

Розробив студент Коверзнев Д. Ю.

(прізвище та ініціали)

(підпис)

Керівник роботи к.т.н., доцент Кондратенко Г.В.

(наук. ступінь, вчене звання, прізвище та ініціали)

(підпис)

« » 202 р.

Анотація
бакалаврської кваліфікаційної роботи студента групи 402 ЧНУ ім. Петра
Могили
Коверзєва Дмитра Юрійовича
на тему «Розробка візуально-інтерактивної RPG на платформі
Unity»

У даній роботі ставиться завдання розробки візуально-інтерактивної гри жанру RPG на ігровому рушії Unity. Було проаналізовано найпопулярніші сучасні ігрові рушії, фреймворки та конструктори ігор. **Об'єкт роботи:** процеси розробки візуально-інтерактивних застосунків. **Предмет роботи:** програмні засоби, платформи та інформаційні технології для розробки застосунків.

Популярність комп'ютерних ігор зростає з кожним роком, ніша ринку дозволяє постійно потребує поповнень новими ігровими тайтлами. Як і будь-яка інша культурна індустрія, відеоігри мають величезний ринок, повний видавців та розробників, які створюють високобюджетні продукти, що кожного року продаються мільйонами. У 2021 році глобальний ігровий ринок згенерує прибуток у розмірі \$175 млрд. Мобільні ігри 2021 року вперше «виграли першість» серед комп'ютерних ігор. Прибуток у сегменті становитиме \$90 млрд.

Результат роботи – готовий проект з візуально-інтерактивною системою, реалізований за допомогою об'єктно-орієнтованого програмування та процедурної генерації локацій.

Робота включає в себе: 84 сторінки, 6 таблиць, 25 ілюстрацій, 39 джерел та 8 додатків.

Ключові слова: Ігровий рушій, фреймворк, конструктор ігор, ігровий тайтл, процедурна генерація, чанк.

Abstract

Bachelor's degree qualification student of group 402 ChNU. Petro Mohila

Koverznev Dmitry Yurievich

"Development of visual-interactive RPG on the Unity platform"

This paper aims to develop a visual-interactive game of the RPG genre on the Unity game engine. The most popular modern game engines, frameworks and game designers were analyzed. Object of work: processes of development of visual-interactive applications. Subject of work: software, platforms and information technologies for application development.

The popularity of computer games is growing every year; the niche of the leisure market constantly needs to be replenished with new game titles. Like any other cultural industry, video games have a huge market, full of publishers and developers who create high-budget products that sell millions every year. In 2021, the global gaming market will generate revenue of \$ 175 billion. Mobile games in 2021 for the first time "won the championship" among computer games. Revenue in the segment will be \$ 90 billion.

The result is a ready-made project with a visual-interactive system, implemented using object-oriented programming and procedural generation of locations.

The work includes: 84 pages, 6 tables, 25 illustrations, 39 sources and 8 appendices.

Key words: Game engine, framework, game designer, game title, procedural generation, chunk.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	4
ВСТУП	5
1 АНАЛІЗ СУЧАСНОГО СТАНУ ЗАДАЧІ РОЗРОБКИ КОМП'ЮТЕРНИХ ІГОР	7
1.1 Перспективні напрямки розробки	7
1.2 Останні розробки та публікації.....	10
1.3 Опис існуючих аналогів RPG ігор.....	11
1.4 Постановка задачі.....	19
Висновки до розділу 1	20
2 ВИБІР ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ РОЗРОБКИ	21
2.1 Методи для вирішення задачі.....	21
2.2 Технології розробки системи	22
Висновки до розділу 2	35
3 РЕАЛІЗАЦІЯ ВІЗУАЛЬНО-ІНТЕРАКТИВНОЇ RPG НА ПЛАТФОРМІ UNITY.....	36
3.1 Створення візуальної частини додатку.....	36
3.2 Реалізація ігрової логіки.....	39
3.3 Інтеграція процедурної генерації	45
3.4 Реалізація ігрового меню.....	49
4 ОПТИМІЗАЦІЯ ТА ТЕСТУВАННЯ СТВОРЕННОГО ЗАСТОСУНКУ	52
4.1 Оптимізація ігрової логіки	52
4.2 Тестування додатку.....	54
Висновки до розділу 4	56

ВИСНОВКИ.....	72
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	73
ДОДАТОК А Код скрипту, відповідний за ігрову логіку.....	77
ДОДАТОК Б Код скрипту, відповідний за предмети	78
ДОДАТОК В Код скрипту відповідний за головне меню	79
ДОДАТОК Г Код скрипту відповідний за чанки.....	80
ДОДАТОК Д Код скрипту відповідний за генерацію чанків.....	81
ДОДАТОК Е Код скрипту відповідний за переміщення персонажа.....	82
ДОДАТОК Ж Код інтерфейсу відповідний за ініціалізацію предмету.....	83
ДОДАТОК И Код інтерфейсу відповідний за дії гравця	84

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

ПЕРЕЛІК СКОРОЧЕНЬ

AAA – високобюджетні і дорогі проекти

3D – тривимірна графіка

FPS – (frame per second) – частота зміни кадрів за секунду

RPG – жанр відеоігор

UI – користувальницький інтерфейс

ВСТУП

Ігровий рушій - це центральний програмний компонент комп'ютерних відеоігор або інших графічно-інтерактивних додатків. Його головні переваги – забезпечує основні технології для виконання поставлених цілей, спрощує розробку, та дає можливість кроссплатформеного програмування та інтеграції на різні консольні та настільні операційні системи, такі як: Microsoft Windows, Mac OS, Linux.

Для того, щоб зробити цікавий продукт, потрібна також цікава ідея. За допомогою багатofункціональних інструментів які розробники ігрових рушіїв нам надають, можна зробити все що завгодно. Однозначним плюсом розробки комп'ютерної гри на вже готовому інструменті, є швидкість розробки програмної частини. Якщо раніше програмісту потрібно було писати безліч рядків коду, щоб використовувати просту можливість перевірки зіткнення між двома об'єктами, то тепер всі ці обчислення робляться за все однією командою. Ігрові рушії прискорюють та полегшують роботу розробників. Однак, програмування це тільки одна частина розробки програмного застосунку. Створення якісного ігрового продукту буде так само включати в себе створення великого обсягу графічного матеріалу. І до всього іншого, необхідно звернути увагу на аудіо системи.

Розробкою відеоігор може займатися як компанія (ігрова студія) так і одна людина. В історії відеоігор є багато прикладів, як одна людина взяла на себе роль програміста, музиканта та художника. Розробка AAA проекту коштує від мільйона доларів і більше. Процес розробки звичайної сучасної гри займає близько року, а для AAA проектів може складати декілька років. Цикл розробки звичайних «казуальних» ігор займає близько 4-6 місяців.

Об'єкт роботи: процеси розробки візуально-інтерактивних застосунків.

Предмет роботи: програмні засоби, платформи та інформаційні технології для розробки застосунків.

Метою кваліфікаційної роботи є розробка унікальної RPG гри за допомогою ігрового рушія Unity 3D.

Оскільки даний ігровий рушій є лише конструктором, то доведеться так само використовувати професійний пакет для створення тривимірної комп'ютерної графіки та анімації "Maya" від відомої компанії Autodesk він включає в себе засоби моделювання, анімації, рендеринга і постобробки.

Основні задачі, які потрібно виконати у ході виконання роботи:

- ознайомитись з аналогічними проектами;
- ознайомитись з ігровим рушієм Unity 3D;
- ознайомитись з графічним редактором Autodesk Maya;
- спроектувати 3D-об'єкти для проекту;
- освоїти весь необхідний функціонал Unity 3D для правильного функціонування об'єктів;
- провести оптимізацію проекту Unity 3D для підвищення продуктивності програмного застосунку.

1 АНАЛІЗ СУЧАСНОГО СТАНУ ЗАДАЧІ РОЗРОБКИ КОМП'ЮТЕРНИХ ІГОР

1.1 Перспективні напрямки розробки

В еру високошвидкісного інтернету велика частина ігрової індустрії, так чи інакше, орієнтована на онлайн. На даний момент існує безліч різновидів багатокористувацьких ігор. Залежно від параметра класифікації, дані гри можна розділити на:

- браузерні і клієнтські гри. У браузерних іграх web-браузер використовується як операційна оболонка, або виступає в ролі контейнера для віртуальної машини, яка виконує код гри;

- ігри, побудовані на принципі пошуку партнерів та інтерактивні світи (або масові ігри для декількох гравців). До першої групи належать гри за типом шахів, шашок, морського бою і їм подібних і відрізняються простотою в порівнянні з другою.

На даний момент найбільш поширеними типами є інтерактивні світи і гри для соціальних мереж.

Крім технологій реалізації багатокористувацьких ігор, продукти даного сегмента ринку добре помітні за жанрами:

- MMORPG (Massively Multiplayer Online Role-Playing Game, масова розрахована на багато користувачів ролева онлайн-гра);

- MMOFPS (Massively Multiplayer Online First Person Shooter, масовий багатокористувацький онлайн-shooter від першої особи);

- MMORTS (Massively Multiplayer Online Real-time Strategy, масова розрахована на багато користувачів онлайн-стратегія в реальному часі);

- онлайн-ігри в жанрі time-managment (тайм-менеджмент, відомий також під назвою «принеси-подай»);

- інші - різні комбінації вищеописаних жанрів, а також менш поширені, «екзотичні» жанри.

Величезною популярністю серед користувачів є жанр RPG та Rouge Like. Жанр комп'ютерних ігор, характерним для якого є набір найбільшої кількості очок, перемагаючи ворогів, досліджувати підземелля, вдосконалювати характеристики персонажа. Як правило в іграх жанра RPG предметами, за допомогою яких гравець поліпшує свої характеристики – це амуніція: зброя та броня.

Однією з унікальних рис жанру RPG є те, що гравець під час гри приймає певні рішення, та робить для себе стратегії проходження локацій [1]. Сама суть рольової гри - відігрівання певної ролі. Гравець бере під контроль певного героя, з набором стандартних навичок, характеристик та вмій.

Деякі представники жанру RPG можуть бути об'єднані з представниками іншого жанру. Такі ігри як World of Warcraft, та серії ігор The Elder Scrolls мають величезну кількість прихильників по всьому світу. Стратегії гравців можуть відрізнитися від одного жанру RPG до іншого.

В іграх жанру RPG зазвичай є сюжетна лінія, в цілому це набір завдань, за які гравець отримує винагороду. Завдання які передбачають пошук або знищення певної кількості ворогів оцінюються по різному. Ранні гри жанру були браузерними та без великої кількості спец. ефектів. Їх призначення було у діалогах з іншими гравцями або комп'ютером, з задалегідь прописаними сценаріями.

Велика увага була зосереджена на інтерактивності ігор. Деякі користувачі витрачали багато часу на розвиток свого персонажу. Ці гравці користувались великою популярністю у менш розвинутих користувачів. Основою взаємодії гравців були групи(клани). У кланах усі дотримувались строгої ієрархії, новий гравець може бути прийнятий в клан за схваленням лідера клану. Клани частіше за все змагалися з іншими кланами та проходили складні підземелля великою групою, де кожен виконував свою роль. Багато ігор цього жанру використовують закадрову музику і звуки, попереджаючи

гравця про безпеку. Такі звукові підказки допомагають гравцеві в просуванні по грі.

Головні риси рольової системи ігор жанру RPG:

- гравець управляє одним (втілення, аватар) або декількома (група, партія) унікальними персонажами. Це потрібна умова для всіх рольових ігор. На відміну від чистих стратегій, його персонажі є унікальними і мають ім'я;
- гравець поступово покращує характеристики та/або навички (за допомогою внутрішньоігрових значень, найчастіше очок досвіду, одержуваних за виконання завдань, дослідження світу, діалоги, битви тощо);
- у процесі проходження відбуваються перевірки параметрів та/або можливостей і навичок;
- персонажі можуть покращувати свої характеристики та/або здібності та навички за допомогою елементів спорядження. Теоретично можна створити комп'ютерну рольову гру без спорядження. Відсутність елементів спорядження може бути розпізнавальним знаком пригодницьких ігор, які часто використовують інвентар виключно для предметів, що застосовуються при вирішенні головоломок.

Одні з найважливіших елементів рольової системи ігор жанру RPG:

- гравець може створювати своїх персонажів;
- гравець повинен планувати розвиток персонажа. Цей елемент відображає стратегічний аспект створення та розвитку персонажа, а також поєднання навичок у групі;
- основним способом вирішення проблем, взаємодії з ігровим світом та подолання перешкод є тактичне застосування навичок/здібностей персонажа/групи персонажів (навички самого гравця є вторинними).

Якщо ця умова не виконується, то швидше за все, перед нами один із видів бойовика, в якому вирішальне значення мають навички гравця, а не персонажа.

1.2 Останні розробки та публікації

За всю історію відеоігор жанр RPG був одним з найпопулярніших. Все через те, що цей жанр впливає на мозок людини. Було проведено огляд літератури про RPG як терапевтичний інструмент або стратегію профілактики в психотерапії та психічному здоров'ї, висвітлюючи популяції досліджень, форми RPG та використовувані втручання. Для цього було здійснено систематичний пошук у базах даних PubMed/MEDLINE, Embase, PsycINFO, BVS/LILACS та сірій літературі.

З 4069 розглянутих досліджень було включено 50 джерел доказів. Більшість з них була опублікована станом на 2011 рік (78%) у журналах (62%) і цільове терапевтичне використання РПГ (84%). Більшість втручань використовували комп'ютерні (50%) або настільні RPG (44%), переважно з когнітивними та/або поведінковими (52%) терапевтичними підходами та орієнтованими на підлітків (70%) [2, 3].

Результати свідчать про потенційне використання RPG як додаткового інструменту в психотерапії. Однак лише 16% включених досліджень були експериментальними. Ми виявили значну неоднорідність у визначеннях РПГ, результатах та використаних заходах, що перешкоджає систематичному перегляду. Таким чином, необхідні більш емпіричні та добре продумані дослідження щодо застосування RPG у психічному здоров'ї.

Рольові ігри можуть задовольняють потреби. Хоча RPG ігри можуть здатися іграми з простими вигаданням, але Обрі С. Адамс з Каліфорнійського університету, у Санта-Барбарі, провів дослідження, що емоції, товарицькість та досягнення, які переживають гравці, є реальними, що свідчить про те, що RPG мають реальні наслідки для гравців [3].

1.3 Опис існуючих аналогів RPG ігор

1.3.1 Опис гри World of Warcraft

World of Warcraft — багатокористувацька рольова онлайн-гра, розроблена компанією Blizzard Entertainment. Це четверта відеогра в серії Warcraft, не враховуючи доповнень та відхиленої Warcraft Adventures: Lord of the Clans. Події ігор серії Warcraft відбуваються у фентезійному всесвіті Warcraft, що вперше був зображений у грі Warcraft: Orcs & Humans в 1994 році. Події World of Warcraft відбуваються через п'ять років після фіналу Warcraft III: The Frozen Throne.

Згідно з книгою рекордів Гіннеса, World of Warcraft є найпопулярнішою MMORPG в світі. В прес-релізі Blizzard від 23 січня 2007 року повідомлялося, що кількість передплатників гри склала більше 8 млн. осіб по всьому світу, станом на 7 жовтня 2010 року кількість передплатників перевищила вже 12 млн. World of Warcraft таким чином є найбільш популярною MMORPG в світі, і тримає місце в Книзі рекордів Гіннеса як найпопулярніша MMORPG за кількістю передплатників. Станом на липень 2012 року, У січні 2014 було оголошено, що протягом всієї роботи гри створено понад 100 млн. акаунтів.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity



Рисунок 1.1 – Скріншот із гри World of Warcraft

Таблиця 1.1 – Оцінка проекту World of Warcraft різними видавниками:

Назва ресурсу	Оцінка
IGN	9.1 з 10
Eurogamer	8 з 10
Game Informer	9.5 з 10
GamePro	4.5 з 5
GameSpot	9.5 з 10
PC Gamer (UK)	94%

Гра отримала наступні нагороди:

- гра року за версією таких видань як: Gamespot, Macworld, XSages, Fragland, Cinescape, Gaming Illustrated;
- найкраща багатокористувацька онлайн-гра за версією ресурсу GameSpot;
- найкраща рольова гра (RPG або MMORPG) за версією таких видань як : GameSpy, VoodooExtreme;

Кафедра інтелектуальних інформаційних систем

Розробка візуально-інтерактивної RPG на платформі Unity

- найкраща RPG для ПК за версією ресурсу FileFront;
- переможець конкурсу Best of Show за версією ресурсу E3 2003;
- найкраща гра Persistent Online за версією ресурсу E3 2003;
- лауреат конкурсу Best Graphics (найкраща графіка) за версією ресурсу E3 2003;
- найкращий вступний відеоролик за версією ресурсу Game Chronicles [5].

1.3.2 Опис Albion Online

Albion Online — кросплатформова масова розрахована на багато користувачів рольова онлайн-гра в середньовічній стилістиці, розроблена і випущена Sandbox Interactive.

Особливості гри:

- великий світ поділений на зони, у кожній зоні від 6 до 15 територій під захоплення гільдіями. Загалом у грі наразі близько 800 територій під захоплення;
- ігрова економіка регулюється гравцями;
- всі предмети та будівлі у грі створюються гравцями;
- можливість будівництва своїх будівель на власних островах;
- можливість здавати свій острів/побудови іншим гільдіям;
- PvP & Full Loot. Повна втрата речей при смерті в PvP (у «червоних» та «чорних» зонах);
- унікальна система кастомізації персонажа.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity



Рисунок 1.2 – Скріншот із гри Albion Online

Таблиця 1.2 – Оцінка проєкта Albion Online різними видавниками:

Назва ресурсу	Оцінка
GameRevolution	4/5
GameStar	75/100
StopGame	80/100
Metacritic	73/100

Джоел Пітерсон з Destructoid відзначив схожість гри з іншою популярною RPG грою - Ultima Online , але порахував свої ігрові враження більш схожими з EVE Online, з його точки зору, Albion Online пропонує гравцеві цікаву економічну систему але необхідність доробити інтерактивність ігрового всесвіту [6].

1.3.3 Опис The Elder Scrolls

The Elder Scrolls V: Skyrim - мультиплатформенна комп'ютерна рольова гра з відкритим світом, розроблена студією Bethesda Game Studios та випущена компанією Bethesda Softworks. Це п'ята частина серії The Elder Scrolls. Гра була випущена 11 листопада 2011 для Windows, Playstation 3 і Xbox 360. Для гри були випущені три додатки, що завантажуються під назвами Dawnguard, Hearthfire і Dragonborn, пізніше об'єднаних у виданні The Elder Scrolls V: Skyrim — Legendary Edition. Восени 2016 року було випущено перевидання для Windows, Playstation 4 і Xbox One, що включає ті ж доповнення, оновлену графіку та підтримку користувацьких модифікацій



Рисунок 1.3 – Скріншот із гри The Elder Scrolls V

Таблиця 1.3 – Оцінка проекту The Elder Scrolls V різними видавниками:

Назва ресурсу	Оцінка
Famitsu	40/40

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

Game Informer	9.5/10
GameSpot	9/10
GameTrailers	9.3/10
IGN	9.5/10
OPM (UK)	7/10

Концепція гри виникла незабаром після випуску попередньої гри The Elder Scrolls IV: Oblivion у 2006 році, але робота над грою не почалася до випуску іншого відомого проекту Bethesda Game Studios - Fallout 3 у 2008. Розробники вважали, що гра повинна бути духовним спадкоємцем як Fallout 3, так і попередніх ігор із серії The Elder Scrolls. Над грою працював колектив із приблизно 100 людей, до числа яких входили як нові таланти, так і ветерани серії. Керував розробкою Тодд Говард, виконавчий продюсер Bethesda Game Studios [7].

1.3.4 Опис Dark Souls III

Dark Souls III — рольова відеогра, розроблена FromSoftware і видана Bandai Namco Entertainment для PlayStation 4, Xbox One і Microsoft Windows. Вийшла 24 березня 2016 року в Японії та 12 квітня 2016 по всьому світу. Є заключною частиною серії Dark Souls та оповідає про повсталого з мертвих героя, який шукає в занепалому й помираючому світі Володарів Попелу, яким під силу відновити джерело життя.



Рисунок 1.4 – Скріншот із гри Dark Souls III

Таблиця 1.4 – Оцінка проєкта Dark Souls III різними видавниками:

Назва ресурсу	Оцінка
Famitsu	38/40
Game Informer	9.3/10
GameSpot	8/10
IGN	9.5/10
PC Gamer (UK)	94/100
PlayUA	89/100

Dark Souls III отримала загальне схвалення і дуже високі оцінки критиків. Крім візуальної складової та механіки боїв багатьма позитивно зазначалася подібність цієї гри до попередньої гри трилогії та схожість з грою Bloodborne від компанії FromSoftware. Dark Souls III – один з видатних

Кафедра інтелектуальних інформаційних систем

Розробка візуально-інтерактивної RPG на платформі Unity

шедеврів складної механіки боїв та гарний приклад середньовікової реалізації зовнішнього світу [8].

1.3.5 Опис Gothic

Gothic — відеогра в жанрі RPG, розроблена німецькою компанією Piranha Bytes для платформи Windows. Спочатку гра вийшла в Німеччині 15 березня 2001, потім в США 15 листопада 2001.

Історія гри розгортається на острові Хорініс, де знаходяться великі родовища магічної руди, куди потрапляє головний герой за якусь невизначену провину. Героєві, ім'я якого в грі так і не називається, доводиться завоювати репутацію серед місцевих таборів, вижити серед місцевих злочинців і чудовиськ, та в підсумку врятувати весь острів від стародавнього демона.

Гра отримала досить високі оцінки критиків та швидко здобула популярність. У 2002 році вийшло продовження Gothic II, а в 2006 було видано третю та останню частину серії.



Рисунок 1.4 – Скріншот із гри Gothic

Таблиця 1.4 – Оцінка проекту Gothic різними видавниками:

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

Назва ресурсу	Оцінка
Absolute Games	70/100
Eurogamer	8/10
<i>PC Gamer (US)</i>	75/100
GameSpot	7.2/10
GameSpy	83/100
IGN	8.6/10

Gothic отримала схвальні оцінки і відгуки, зібравши на агрегаторі Metacritic 81 бал зі 100 баллів. Ресурс Gamespot оцінив гру в 7,2/10 баллів з висновком: «Чи грати вам у Gothic? Ні, якщо ви хочете RPG з глибоким розвитком персонажа чи бойової hack-and-slash гри, або коли ви боїтеся зіткнутися з численними проблемами гри. І ні, якщо ви не маєте хорошого комп'ютера, істотно кращого, ніж вказано в системних вимогах. Однак, якщо жодна з цих умов не турбує вас і ви зацікавлені грою, більше присвяченою стосункам персонажів і квестам, ніж боям, вам варто спробувати Gothic». Можна зробити з цього такий висновок: це не просто гра, одна з тисячі схожих по жанру, це одна з тих, рідкісних, найінтерактивніших проєктів, де головне - розвиток персонажа як співрозмовника, простої людини, а не тільки лютого воїна [9].

1.4 Постановка задачі

Тому згідно з актуальністю теми розробку візуально-інтерактивної RPG на платформі Unity **Об'єктом роботи** вибрано процеси розробки візуально-інтерактивних застосунків. Згідно з цим, **предметом роботи** обрано програмні засоби, платформи та інформаційні технології для розробки застосунків.

є технології розробки застосунків.

Висновки до розділу 1

Як видно з зазначених матеріалів за останні роки жанр RPG зайняв чималу нішу на ринку відеоігор. А в деяких країнах навіть досягли статусу «Найкращого жанру усіх часів» та мистецтва. Компаній з розробки ігор стає дедалі більше. З кожним роком з'являється усе більше розробників RPG проектів, кожен з яких додає щось унікальне. В свою чергу з'явився чималий попит на інструментальні засоби що полегшують розробку, покращують графіку, оптимізують фізику тощо. Проте кожна гра має особливості залежно від типу, жанру, платформи тощо. А отже потреба в їх проектуванні та розробці виглядає досить актуальною.

2 ВИБІР ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ РОЗРОБКИ

2.1 Методи для вирішення задачі

Для розробки RPG застосунку можуть використовуватись різні бібліотеки, фреймворки та інші інструменти та toolkits. Більшість RPG ігор написані за допомогою ігрових рушіїв, але це не єдиний спосіб написати власний проект. Ігрові рушії поділяються на ті, що з візуальним програмуванням, та без цієї функції.

Окрім цього існує безліч різних рушіїв які кардинально відрізняються один від одного. Деякі більш орієнтовані на дво-вимірні ігри, інші на три-вимірні. Най популярніші ігрові рушії орієнтовані на обидва ці напрямки пропонуючи дуже великий функціонал для якісного проектування, розробки та аналізу.

Для цієї роботи я обрав ігровий рушій Unity, тому що від надзвичайно ефективний та не потребує сильних навичок та великого досвіду у розробці ігор. Unity більш орієнтований на мобільні ігри для платформ Android та IOS, але також має багато прикладів високоякісних продуктів на настільні комп'ютери та ігрові приставки (консолі).

За допомогою власної програми, я порівняв усі най популярніші ігрові рушії та обрав найкращий за допомогою **методу аналізу ієрархій в умовах нечіткості та невизначеності** [10]. Такий підхід був обраний тому, що на власну думку – це один з найефективніших методів для вибору альтернативи. Порівнювались такі ігрові рушії як: Unity, Unreal Engine, Godot та CryEngine.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

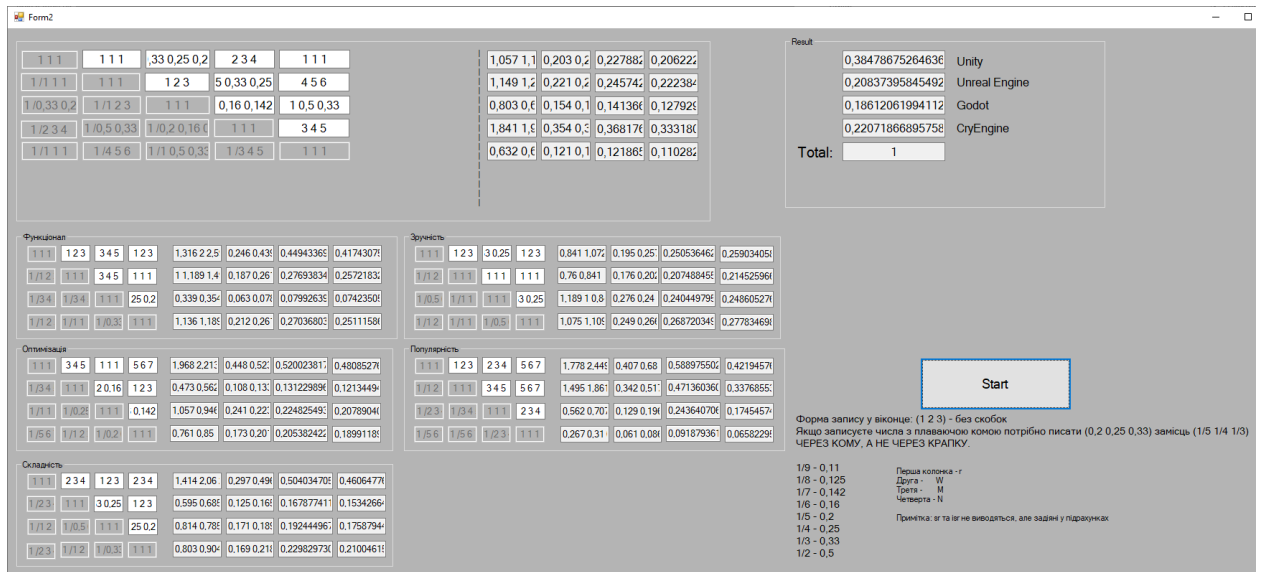


Рисунок 2.1 – порівняння альтернатив методом **FANP**

За результатами програми, на мою думку, ігровий рушій Unity обрано як найефективніший, набравши 0,38 балів.

2.2 Технології розробки системи

2.2.1 Характеристика візуального програмування

Деякі графічні рушії мають функцію візуального програмування. Візуальне програмування – вид програмування, який передбачає використання спеціальних блоків (скриптів), для програмування ігрової логіки. Використання даної технології повністю позбавить користувача написання коду, але такий вид програмування не забезпечує гнучкості програми.

Візуальне програмування – це головний інструмент реалізації взаємодій у проєкті. За допомогою графів дій та значень можна розробляти будь-яку логіку — як реалізовану покадрово, так і ту, що спрацьовує за умови.

Ідеальний варіант – використання програмування для важких частин логіки програми. За бажанням програміста дозволено використовувати графічне проектування, але не за для складної логіки.

На сьогодні є багато функцій та бібліотек для реалізації будь-яких завдань з програмування використовуючи візуальний скриптинг. Технологія передбачає можливість внесення змін до графів у режимі реального часу. Такий підхід значно прискорює ітерацію та тестування ідей без необхідності компілювати зміни у проекті.

Використовуючи візуальний скриптинг значно легше можна виконати процедуру знаходження помилок (Debug). Скрипти, які виконують задані процедури (такі як додавання до змінної, або тернальний оператор), з'єднані між собою вузлами. Ці вузли підсвічуються, коли програма обробляє даний вузол. За допомогою вузлів логіка програми виглядає як єдиний граф. Якщо програма зупиняється через помилку, підсвічується вузол, на якому зупинилась програма. Таким чином можна з легкістю виконати Debug.

Система спроектована таким чином, щоб користувач без технічних навичок міг з легкістю знайти той інструмент, який потрібен для вирішення задачі. Це забезпечується завдяки правильним правилам найменувань, угруповання та коментування. Розділ Fuzzy Finder допомагає легко знайти потрібний інструмент у разі виникнення труднощів.

Система візуального скриптингу підтримує використання у графах будь-яких методів, полів, атрибутів або подій Unity, а також сторонніх плагінів або навіть ваших власних скриптів. Завдяки прямому доступу системи до вашої кодової бази всі зміни коду відразу відображаються в інтерфейсі. Таким чином можна з легкістю об'єднувати використання візуального програмування за допомогою скриптів, та кодування на мові програмування.

Ігрові рушії які сьогодні забезпечують можливість візуального програмування:

- Unreal Engine;
- Unity;
- Godot;
- Armory 3D.

2.2.2 Опис ігрового рушія Unity

Unity — інструмент для розробки програмних застосунків і рушій, на якому вони працюють. Unity – візуально-інтерактивний багатоплатформовий інструмент, застосунки на якому можуть бути запущені на різних платформах, таких як:

- Windows;
- IOS;
- Android;
- Linux;
- WebGL;
- Universal Windows Planform;
- PlayStation 4;
- PlayStation 5;
- Xbox;
- Oculus;
- AndroidTV;
- tvOS;
- Nintendo Switch;
- ARCore;
- Stadia;
- Microsoft HoloLens;
- Magic Leap.

Ігровий рушій може проектувати ігри у дво- та тривимірній графіці, на пристроях віртуальної та доповненої реальності. Програми, які були розроблені на платформі Unity, підтримують DirectX та OpenGL.

Unity забезпечує використання технології Computer Vision, використовуючи синтетичні дані. Такий тип даних використовується у

навчанні систем комп'ютерного зору для поліпшеної ітерації та прискореного розгортання.

Використовуючи синтетичні дані можна вирішити проблему у навчанні моделей штучного інтелекту, орієнтованих на людину. Unity надає генератор даних про людину з гнучно-настрайовальною системою та готовою базою 3D-асетів людей. Таким чином можна розробляти неупереджені моделі штучного інтелекту для віртуальної та доповненої реальності, оцінки людських поз, безпілотних автомобілів, розпізнавання дій, та інших систем аналізу поведінки людей без використання реальних зображень людей.

Unity надає велику та зручну документацію, де кожен розробник різного рівня навичок може дізнатися про нові функції та робочі процеси Unity. Документація надає інформацію що до скриптів з прикладами та посиланнями на схожі функції. Усе організовано досвідченими розробниками платформи [11].

2.2.3 Опис Microsoft Visual Studio

Microsoft Visual Studio — інтегроване середовище для редагування, програмування та компіляції програмних застосунків, розроблених компанією Microsoft. Visual Studio надає можливість писати код на різних мовах програмування, а також використовувати різні бібліотеки та інструменти. Середовище надає змогу розробляти програми з графічним інтерфейсом, включно з підтримкою технології Windows Forms, а також веб-сайти, веб-застосунки, веб-служби. Visual Studio забезпечує кодування на всіх платформах, які підтримує компанія Microsoft. Фреймворк надає все необхідне для кодування, збірки проекту, пошуку помилок, аналізу, розробки бібліотек та навіть навчання.

Одна з версій Visual Studio Code – базується на платформі Microsoft Windows, та являє собою автономний редактор коду. Це один з найкращих виборів для написання веб-додатків та веб-сайтів для веб-розробників з розширеннями. Code підтримує будь-які мови програмування. Для початку

розробки необхідно завантажити відповідний фреймворк на якому необхідно написати програму. Однією з переваг Visual Studio Code є те, що у додаток вбудовано команди Git та інших SCM.

Git — розподілена система керування версіями файлів та спільної роботи. Проєкт створив Лінус Торвальдс для керування розробкою ядра Linux, а сьогодні підтримується Джуніо Хамано. Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні і злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів [12].

На сьогоднішній день, остання версія Visual Studio 2022 року. Швидше і надійне 64-розрядне середовище розробки. Поставляється з .NET 6, включаючи підтримку MAUI, програм Blazor та Гаряче перезавантаження. Покращені завершення IntelliCode. Останній ланцюжок інструментів для C++20 [13].

Visual Studio дає можливість кастомізувати середовище за вподобанням користувача, надаючи декілька кольорових схем. За допомогою Visual Studio Marketplace, користувач може розробити свій власний дизайн редактору та зробити його по-справжньому персональним.

Редактор коду Visual Studio – найпопулярніше інтегроване середовище для редагування коду. Співробітники та розробники кожен день оголошують новини про майбутнє та нові інтегровані технології редактору коду. Кожні 2-3 роки виходять нові, найсучасніші версії середовища які полегшують, прискорюють та оптимізують розробку нових додатків.

2.2.4 Опис графічних редакторів

Autodesk Maya — професійний графічний редактор, для моделювання тривимірних об'єктів, ефектів, реалістичних персонажів, анімації, композитингу та візуалізації. В даний час є стандартом для розробки

3D графіки для кіно і телебачення. Спочатку розроблена для ОС IRIX (платформа SGI), потім була портована під ОС Linux, Microsoft Windows і Mac OS. З 2013 року, версії програми випускаються тільки для 64-бітових систем.

Maya – стандарт індустрії, він настільки популярний, що з яким працюють майже всі студії світу. Архітектура софту зручна для побудови пайплайну завдяки високотехнологічному API (від англ. application programming interface — набір бібліотек, що відповідають різним функціям Maya). Свого часу це дозволило софт ще більше зміцнити свої позиції в індустрії. Сьогодні видавець, Autodesk, а до цього - Alias, постачає софт до навчальних закладів, таким чином більшість фахівців навчаються робити анімацію саме у цій програмі. Монополізація поряд із гнучкістю функціоналу та централізованою підтримкою – те, що робить Maya популярною серед гігантів медіа-індустрії. Програму використовували у фільмах «Аватар», «У пошуках Немо», франшизах «Трансформери» та «Льодовиковий період», «Холодне серце» та багатьох інших [14].

Maya має багаторічну репутацію надійного професійного софту, що дуже важливо для ААА проектів.

Maya має такі переваги:

- відкритість вихідного коду;
- високоякісні анімації;
- можливість підтримувати найпопулярніші мультимедійні формати;
- можна створювати плаваючі заголовки та логотипи;
- редактор вузлів;
- велика кількість відео-уроків, які можуть переглянути всі бажаючі;
- оптимізація до роботи серед Windows 10.

Також має такі недоліки:

- інтерфейс англійська, відсутність російськомовної версії;
- 30-денна пробна версія програми;

- високий рівень системних вимог;
- не підтримує роботу із 32-розрядними системами.

Але Autodesk Maya – це не єдиний відомий графічний редактор для моделювання тривимірних об'єктів. Такі редактори як 3D Max, Zbrush, Blender можуть скласти конкуренцію.

3D Max — тривимірний графічний редактор, повнофункціональний професійний застосунок, система для створення і редагування об'єктів та створення візуалізацій, розроблена компанією Autodesk. Містить найсучасніші засоби для архітекторів, дизайнерів, художників і фахівців в області мультимедіа. Працює в операційних системах Microsoft Windows і Windows NT (як в 32-бітових, так і в 64-бітових). 3ds Max використовують для візуалізації моделей будівель, комп'ютерних ігор, тривимірних анімаційних мультфільмів, рекламних роликів тощо. За допомогою цього редактора зроблено безліч анімованих моделей для кінофільмів [15].

3D Max має такі переваги:

- 3D анімація та динаміка;
- загальні інструменти анімації;
- інструменти анімації персонажів та оснащення;
- багатоканальний трекінг;
- редактор потоку частинок;
- 3Ds Max Fluids;
- максимальне створення графіків контролерів;
- просте моделювання імпорту даних;
- геодезична та Теплова Карта (технологія Skinning);
- 3D-рендерінг;
- покращений рендеринг ActiveShade;
- фізичні камери;
- технологія "Арнольд" для 3Ds Max;

- рендеринг у A360;
- Autodesk Raytracer Renderer (ART);
- інтерфейс, робочий процес і конвеєр;
- конвертер Сцен;
- бібліотека активів;
- розумна упаковка активів;
- настроювані робочі простори;
- покращена інтеграція інструментів конвеєра;
- пряме посилання з ігровим рушієм Stingray;
- підтримка дисплея з високою роздільною здатністю;
- 3D моделювання, текстурювання та ефекти;
- моделювання сітки та поверхні;
- призначення та редагування текстури;
- модифікатор даних;
- модифікатор волосся та хутра.

Також має такі недоліки:

- **складність.** Оскільки програма професійна, вона потребує навичок та знань;
- **високі вимоги.** Комп'ютер має бути по-справжньому потужним;
- **висока ціна.** Для того, щоб отримувати прибуток із продуктів програми, потрібно купити ліцензію, а вона коштує недешево [16].

ZBrush — програма для тривимірного моделювання, створена компанією Pixologic. Відмінною особливістю даного ПЗ є імітація процесу «ліплення» 3d-скульптури, посиленого рушієм тривимірного рендерингу в реальному часі, що істотно спрощує процедуру створення необхідного 3d-об'єкта. Кожна точка (звана піксель Архівовано 28 листопада 2010 у Wayback Machine.) містить інформацію не тільки про свої координати XY і значеннях кольору, але також й глибину Z, орієнтацію та матеріал. Це означає, що ви не тільки можете «ліпити» тривимірний об'єкт, але і «розфарбувати» його,

малюючи штрихами з глибиною. Тобто вам не доведеться малювати тіні і відблиски, щоб вони виглядали натурально — ZBrush це зробить автоматично [17].

Також швидко працює з стандартними 3d об'єктами, використовуючи пензлі для модифікації геометрії матеріалів і текстур. Дозволяє домогтися інтерактивності при неймовірній кількості полігонів. Використовуючи спеціальні методи, можна підняти деталізацію до десятків (а то й сотень) мільйонів полігонів. Також є безліч модулів (робота з текстурами, геометрією, безліч нових пензлів, швидка інтеграція з професійними пакетами 2d графіки та багато іншого).

Zbrush має такі переваги:

- наявність деталізованого скульптингу;
- широкий вибір кистей;
- авторитетність на ринку (більшість 3D аніматорів використовують Zbrush для створення персонажів);
- наявність можливості самостійного створення власних пензлів або додавання скачаних пакетів;
- велика кількість кистей, що дозволяє опрацьовувати найдрібніші ділянки шкірного покриву.

Також має такі недоліки:

- високу вартість;
- складнощі, що виникають при освоєнні програми у людей без досвіду;
- слабку палітру та рендер [18].

Blender — програмний пакет для створення тривимірної комп'ютерної графіки, що включає засоби моделювання, анімації, рендерінгу, після-обробки відео. До версії 2.80 містив рушій Blender Game Engine для створення відеоігор. Пакет є вільним програмним забезпеченням та розповсюджується під ліцензією GNU GPL [19].

Особливостями пакету є малий розмір, висока швидкість рендерінгу, наявність версій для багатьох операційних систем — FreeBSD, GNU/Linux, Mac OS X, SGI Irix 6.5, Sun Solaris 2.8 (sparc), Microsoft Windows, SkyOS, MorphOS та Pocket PC. Пакет має такі функції, як симуляція динаміки твердих тіл (Rigid Body), рідин (Liquid simulation) та м'яких тіл (Soft body), систему гарячих клавіш (hot key), велику кількість легко доступних розширень, написаних мовою Python. Починаючи з версії 2.61 з'явилися функції «відстеження камери», та «захоплення руху».

Blender має такі переваги.

Швидкість. Програма запускається набагато швидше за своїх конкурентів і до того ж практично моментально реагує на всі команди навіть на не дуже потужних системах. Саме невибагливість до комп'ютерного заліза найчастіше є визначальним фактором для вибору на користь Blender. Використовувати якийсь більш просунутий софт на слабких збірках не вдасться.

Функціональність. На відміну від великої кількості аналогів, Blender заздалегідь включає всі необхідні інструменти для вирішення різних завдань. Тут можна робити майже все, що завгодно, включаючи створення тривимірних об'єктів, накладання текстур, налаштування шейдингу, рендеринг і композитинг. На ряді інших програм подібної функціональності можна досягти лише встановленням великої кількості додаткових плагінів.

Універсальність. За допомогою інструментів Blender користувач може створювати об'єкти для ігор, мультфільмів або кінострічок. Навіть передбачена деяка подоба анімації, яка, втім, перебуває на не дуже високому рівні.

Швидкий внутрішній рендерер Cycles. Ця підсистема чудово функціонує як на GPU, так і CPU. Розробники Blender впровадили цю можливість задовго до того, як вона стала визнаним стандартом.

Рушій Eevee, який дозволяє в режимі реального часу переглядати всі текстури та накладені ефекти.

Розширений UI. У Blender цей компонент дійсно більш приємний і зручний, тоді як інтерфейс більшості конкурентів виглядає як мінімум застарілим. Ця проблема пояснюється тим, що розробники старих систем просто не хочуть нічого змінювати, щоб не змушувати користувачів звикати до нових елементів управління.

Також має такі недоліки.

Інструменти можуть бути не такими потужними, як вимагає робочий процес. Цей недолік вимагає деякого уточнення: деякі інструменти не працюють належним чином тільки тому, що користувач не знайомий з їх правильним застосуванням. Насправді виявляється, що багато вбудованих функцій можна без проблем налаштувати під свої потреби.

Часті оновлення. Регулярне доопрацювання та зміни, безсумнівно, підвищують актуальність програми. Однак вони постійно змінюють інтерфейс або вводять нові, не до кінця зрозумілі функції. В результаті користувачам доводиться щоразу заново звикати до системи. До того ж регулярне поповнення функціоналу відбивається на кількості багів та недоробок, які виявляються вже у процесі використання [20].

2.2.5 Опис Microsoft Visio

Microsoft Visio — редактор діаграм для Windows і редактор векторної графіки. Доступний в двох версіях, стандартний і професійний.

Стандартна і професійна версія використовують той самий інтерфейс, але друга має додаткові шаблони для більш просунутих діаграм і вихідних форматів, а також унікальну функціональність для простішого приєднання користувача до даних різних джерельних форматів, які мають бути відображені діаграмою [21].

Додаток був уперше створений у 1992 році компанією Шейпвере (Shapeware corporation). Microsoft придбав Visio Corporation в 2000 році,

назвавши його застосунком з офісного пакету Microsoft Office. Тим не менше, Visio жодного разу не з'явився в Microsoft Office. Але Visio for Enterprise Architects був частиною деяких редакцій Visual Studio .NET 2003 та Visual Studio 2005.

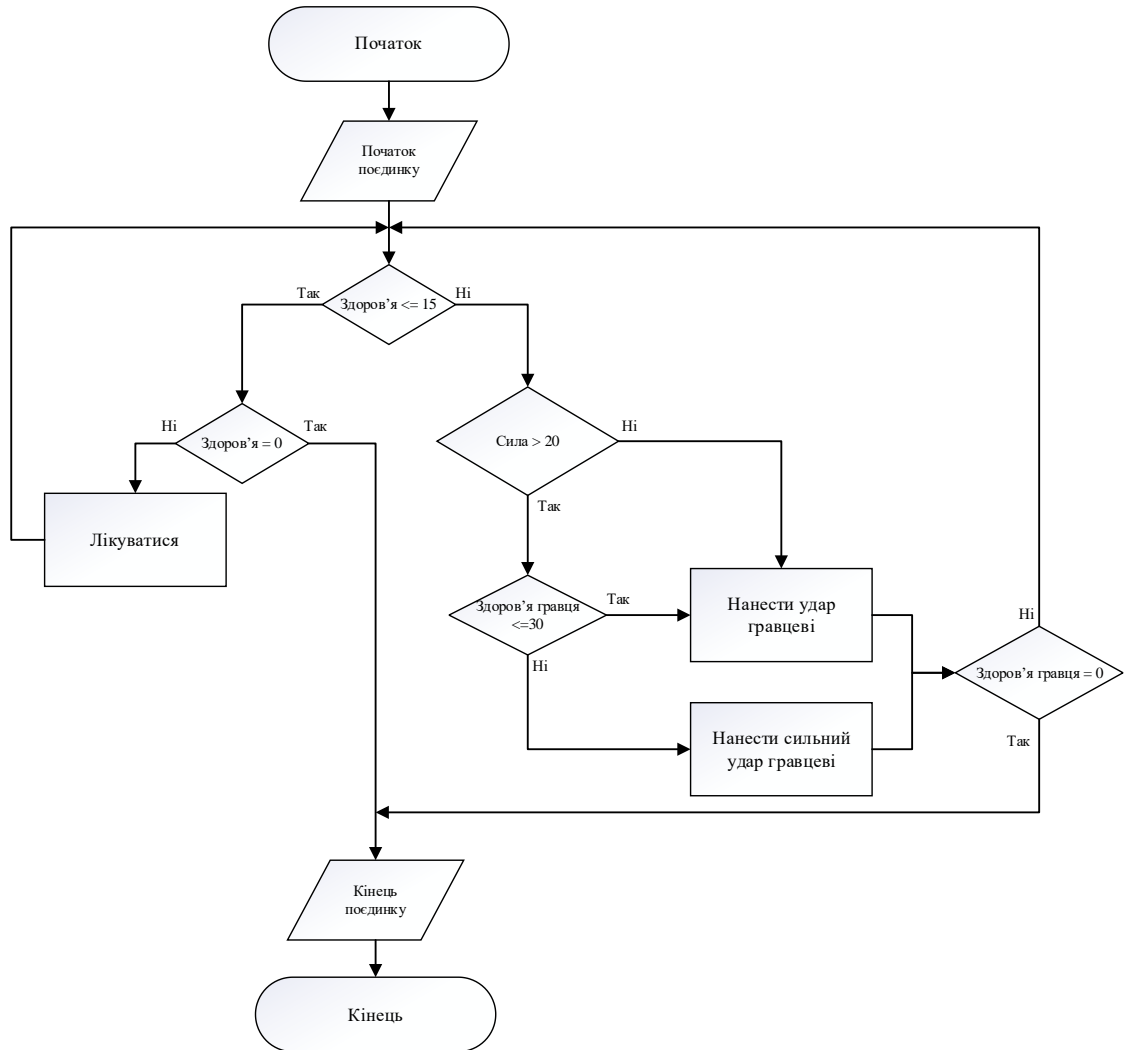


Рисунок 2.1 – Дерево рішень для ворога, виконано у Visio

Visio підтримує можливість створення мережових схем. Таким чином можна спростити процес побудови системи й ілюструйте взаємодію пристроїв і мереж, використовуючи різноманітні мережові шаблони, фігури та з'єднувачі. Слід також відзначити, що Visio також надає змогу редагувати та створювати професійні схеми, візуалізувати данні, і виконувати процеси разом

з іншими користувачами у реальному часі [22]. Це допомагає збільшити продуктивність команди.

2.2.6 Опис Trello

Trello — безкоштовна багатоплатформна система менеджменту та управління проектами, розроблена Trello Enterprise. Створена у 2011 році компанією Fog Creek Software для створення окремої компанії в Нью-Йорку у 2014 році й продана компанії Atlassian в січні 2017 року [23].

Вона використовує парадигму керування проектами, відому як канбан. Проекти зображуються графами та дошками, які містять інформацію. Списки містять картки, якими зображуються задачі. Картки повинні переходити з попереднього списку до наступного (за допомогою перетягування), таким чином зображаючи рух якоїсь функції від ідеї, до тестування. Картці може бути присвоєно відповідальних за неї користувачів. Користувачі та дошки можуть об'єднуватись в команди. У 2017 році, журнал «PC» оцінив сервіс Trello у 3,5 з 5 зірок, назвавши його гнучкою та креативною системою менеджменту проектів.

Trello має обмежену підтримку тегів у вигляді шести кольорових міток. Картки можуть містити коментарі, вкладення, дату завершення та переліки (списки підзадач). Форматуються картки розміткою Markdown. Також Trello має API [24].

Інтерфейс програми працює в форматі drag-and-drop, всі дані оновлюються динамічно на фоні. Серед недоліків: система не працює в режимі офлайн, безплатні плани мають обмеження на розмір приєднаних файлів у 10 Мб (250 Мб для платних акаунтів), відсутня можливість редагувати коментарі.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

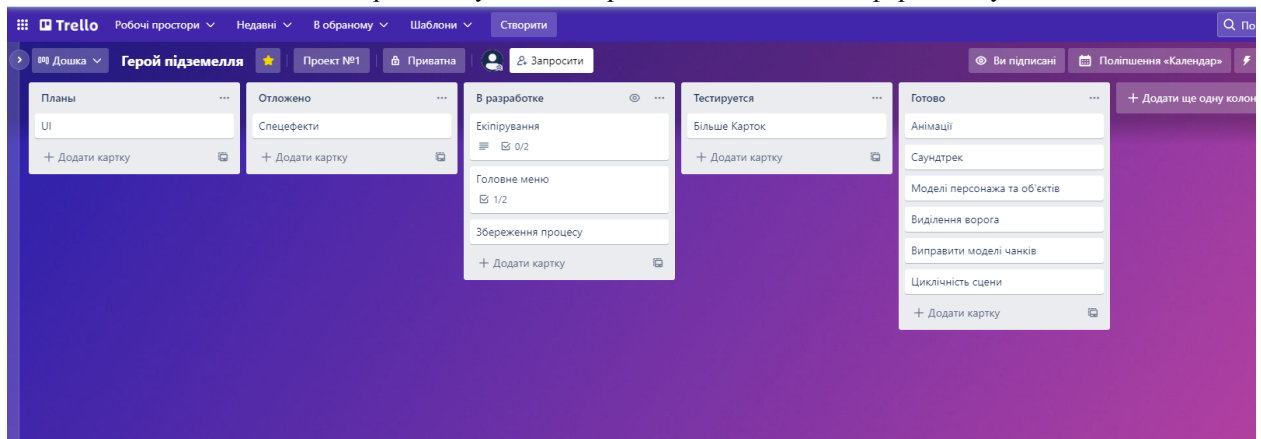


Рисунок 2.2 – Скріншот із Trello

Trello має можливість інтеграції інструментів зберігання даних, таких як Google Drive, One Drive, Dropbox, Evernote, та інших не менш відомих інструментів. Система підтримує дуже зручні, ефективні та популярні діаграми Ганта.

Також для зручності Trello не вимагає спеціальних навичок з автоматизації. Завдяки роботам, можна делегувати прості завдання, що звільнить користувача від:

- автоматизації популярних завдань, як переміщення колонок;
- нагадування про deadline;
- створення кнопок для більше ефективної роботи.

Висновки до розділу 2

Як видно з зазначених матеріалів інструменти для вирішення поставленого завдання підібрані чітко та якісно. Вибір програмних застосунків та методів повністю обґрунтовані. Усе необхідне для виконання завдання описано у розділі 2, Бакалаврської Кваліфікаційної Роботи. Виникає необхідність у щодільному моніторингу оновлень до наведених інструментів, за для якісного виконання роботи найсучаснішими технологіями. Це не тільки скоротить час виконання задачі, але й вирішить питання оптимізації.

3 РЕАЛІЗАЦІЯ ВІЗУАЛЬНО-ІНТЕРАКТИВНОЇ RPG НА ПЛАТФОРМІ UNITY

3.1 Створення візуальної частини додатку.

3.1.1 Розробка 3D моделей в застосунку Autodesk Maya

Один з найголовніших елементів кожної гри – його зовнішній вигляд. Усі моделі та текстури гри повинні бути найвищої якості. Розробку додатка необхідно почати створення 3D моделей, які у свою чергу підвищать продуктивність та впевненість розробника гри. Використовуючи середу розробки 3D моделей Autodesk Maya, можна створити бажані локації та персонажів маючи лише хорошу уяву.

Для застосунку було створено декілька локацій, які служать полем бою для головного героя. На рис. 3.1 зображено одна з спроектованих локацій – «Міст», який буде інтегрований у проект.

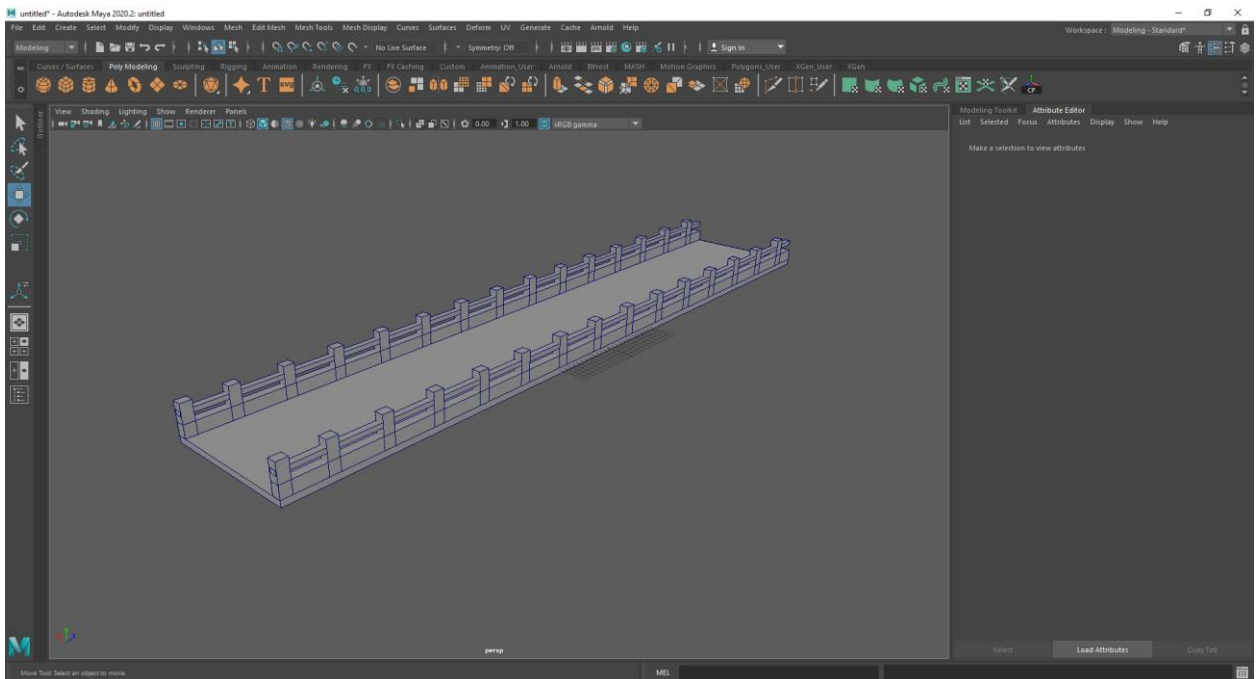


Рисунок 3.1 – Спроектований міст у середовищі Autodesk Maya

Під час розробки даної локації було застосовано такі інструменти як:

- Move tool – переміщення об'єкту;
- Scale tool – змінення розміру об'єкту;
- Copy – копіювання об'єкту;
- Paste – Вставка скопійованого об'єкта
- Rotate tool – обертання об'єкту;
- Bevel components – створення граней на місці обраного, або обраних ребер або кутів;
- Union – об'єднання об'єктів.

Слід необхідно зробити UV розгортку для нашого об'єкту. UV mapping — процес в 3D моделюванні, який полягає в накладанні двовимірного зображення на тривимірну модель. Літерами U і V позначають осі координат площини розгортки, оскільки літери X, Y і Z використовуються для позначення просторових координат. Це необхідно для того, щоб текстури, які ми накладатимемо на саму модель відображалися особисто для кожної сторони об'єкту, а не на всю деталь одночасно.

Для реалізації UV розгортки необхідно перейти до вкладки UV у шапці середовища, потім для відображення текстур так, як необхідно. Результат представлений на рис. 3.2.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

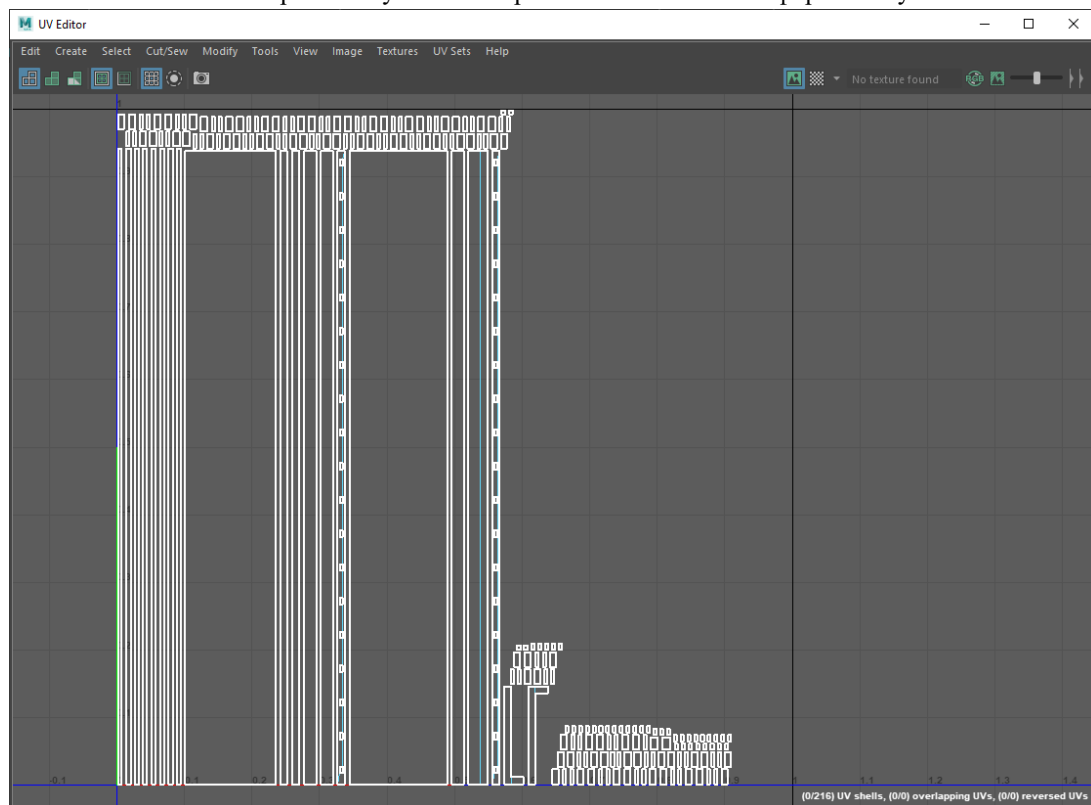


Рисунок 3.2 – Розкладена локація «Міст» у меню UV Editor

Для переносу 3D моделі у середовище Unity необхідно перейти до вкладки File на шапці середовища, потім вибрати «Export All...», або «Export Selection» та вибрати необхідну папку для зберігання у спливаючому вікні. Також для зручності можна використовувати функцію «Send To Unity» або «Send To Unreal» в залежності від обраного ігрового рушія.

Після цього необхідно відкрити середовище розробки Unity, натиснути на меню проекту, обрати функцію «Import new Asset...» та вибрати нашу локацію. Також є альтернативний варіант, можна просто перетягти у меню проекту. Тепер можна розміщувати локацію у середовищі Unity.

3.1.2 Розробка UI інтерфейсу для застосунку

У середовищі розробки Unity є можливість реалізації UI інтерфейсу. Для цього необхідно натиснути на вікно ієрархії в Unity, навести на надпис UI та обрати необхідний елемент інтерфейсу. Елементи інтерфейсу, які були застосовані:

- Canvas – єдиний клас, де зберігаються усі елементи інтерфейсу;
- Panel – задній фон для елементів інтерфейсу;
- Button – кнопка;
- Text – надпис;
- Image – картинка.

3.2 Реалізація ігрової логіки

3.2.1 Опис ігрової логіки

Процес самої гри починається з пересування персонажа від початкової точки появи персонажа, до тригера перед ворогами. Після того, як персонаж зупинився, з'являються картки. Гравець обирає одну з карток, яку хоче використати. Після завершення ходу гравця, необхідно натиснути кнопку «Завершити хід», після цього вороги атакуючи гравця. Гравця можуть атакувати лише живі вороги.

Використані картки замінюються на нові, а не використані залишаються. Тепер гравець може здійснити свій хід знову. Щоб використовувати картки гравець витрачає енергію. Енергія – це ресурс, для використання карток. Кожен новий раунд, енергія гравця дорівнює фіксованому значенню, наприклад 10 одиниць. За допомогою картки Boost, гравець може збільшити свій запас енергії до 14-ти одиниць, але після натискання на кнопку закінчення ходу, енергія гравця буде дорівнювати за замовчуванням, навіть якщо він не використав її.

Протистояння завершується, якщо вороги перемогли персонажа опустивши його показник здоров'я до нуля, або якщо гравець переміг ворогів. У останньому випадку гравець проходить далі, на наступну локацію, де його зустрічають наступні вороги. За кожного вбитого ворога, гравець отримує деяку кількість балів. Ціль гри – набрати найбільшу кількість балів, використовуючи різні елементи екіпірування та різні картки вмінь.

Особливістю даної гри – є необхідність швидко приймати рішення. Гравець може накласти на себе щит, або вилікуватися перед тим, як ворог нанесе удар по гравцеві. Для цього необхідно встигнути завдати удар, або використати захисну картку перед анімацією ворога.

3.2.2 Реалізація карток

Картки поділяються на 2 категорії – атакуючі, та захищаючі. До атакуючих карток відносяться ті картки, які здатні завдати шкоди ворогові. Перед тим, як застосовувати карту такого типу, необхідно вибрати ворога, на якому буде застосована ця картка. Після того, як картка була використана, вона помічається характерним надписом «Used».

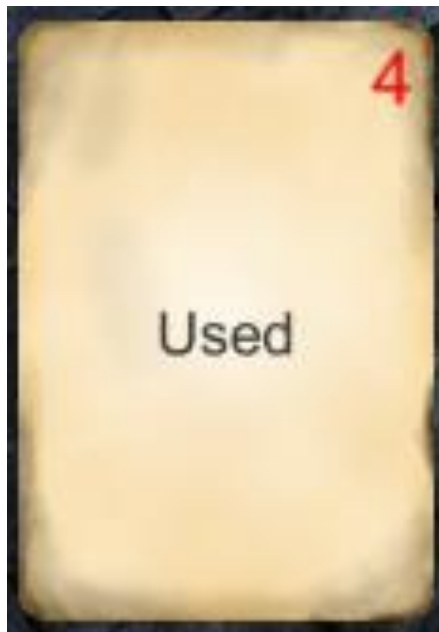


Рисунок 3.3 – Зразок використаної картки

До реалізованих атакуючих карток відносяться такі картки:

- Light attack – передбачає нанесення ворогові урону, який дорівнює показнику сили героя;
- Heavy attack – передбачає нанесення ворогові урону, який дорівнює показнику сили героя помноженому на 1,5;

- Shock – нанесення половини від показника сили героя та блокування дій обраного противника на 1 хід.

До захищаючих карток відносяться усі картки, які надають додаткові одиниці здоров'я, додатковий захист у вигляді щита, або різноманітні ефекти.

До реалізованих захищаючих карток відносяться такі картки:

- Healing – збільшення здоров'я персонажа;
- Shield – утворення захисного щита, який залежить від недостатнього здоров'я;
- Boost – додання додаткових одиниць енергії;
- Rampage – ефект, який триває 1 хід. Додає здоров'я гравцеві та підвищує його урон;
- Health Regeneration – ефект, який діє 3 ходи. Збільшує здоров'я гравця на деяку величину, яка завжди менша за Healing.



Рисунок 3.4 – Зображення усіх розроблених карток

Після закінчення гри, гравець має шанс отримати нові картки. Картки не можна придбати у магазині. Гравець отримує нові картки за свої заслуги.

Зроблено це для того, щоб у гравця була додаткова ціль – отримати усі картки, або бажану для колекції.

3.2.3 Реалізація персонажів

Персонажі поділяються на 2 категорії: ігрові, та неігрові. Ігрові персонажі – це ті, за яких може грати гравець. Відповідно неігрові персонажі – це боти, які запрограмовані виконувати заздалегідь прописані розробником команди. При програмуванні ігрової логіки неігрових персонажів було розроблено особисту логіку для атакуючих персонажів та для персонажів підтримки.

Персонажі підтримки – це персонажі, які не можуть нанести шкоди гравцеві, але можуть вилікувати своїх союзників. Ігрова логіка таких та атакуючих персонажів заснована на деревах рішень. Це прості, але дуже ефективні алгоритми, які використовуються в галузі статистики та аналізу даних для прогнозних моделей та машинному навчанні.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

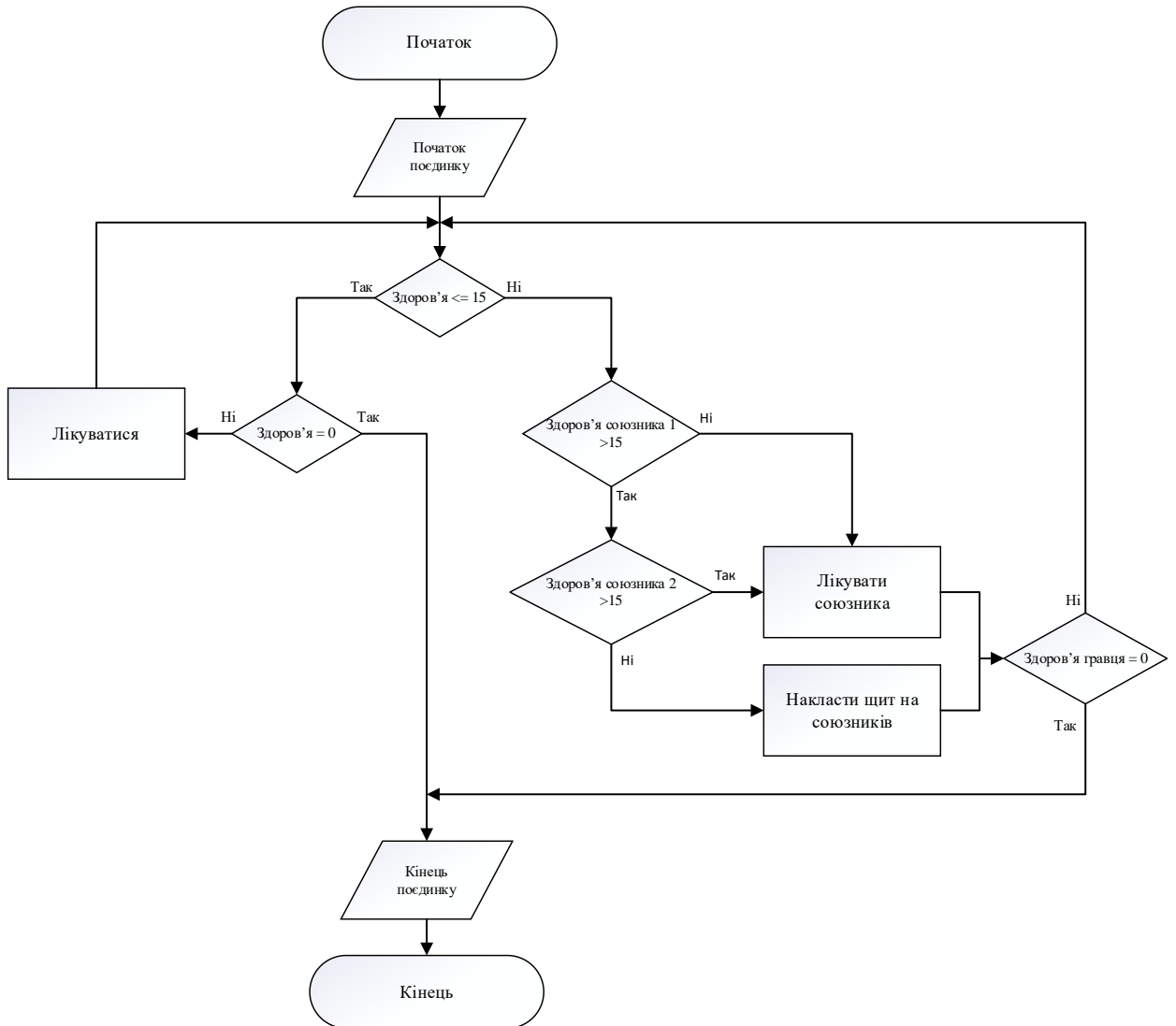


Рисунок 3.5 – Дерево рішень для неігрових персонажів підтримки

Ігрові та неігрові персонажі мають свої анімації, зовнішній вигляд та зброю, яку вони використовують. В залежності від локації, вороги мають різний зовнішній вигляд.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity



Рисунок 3.6 – Скріншот з проекту: гравець зустрів ворогів

Відповідно до рис. 3.6 кожен ворог має свій показник здоров'я. Персонаж на цьому скріншоті підсвічений світлом. Коли гравець вбиває ворога, останній зникає. Коли у гравця закінчується здоров'я, гра закінчується та виводить рахунок, який гравець набрав під час гри. Таблицю лідерів та найкращий результат гравця можна побачити у відповідному розділі у головному меню.



Рисунок 3.7 – Скріншот з проекту: ворог б'є героя

3.3 Інтеграція процедурної генерації

Процедурна генерація рівнів – один з найважливіших елементів гри. Тому необхідно приділяти достатню увагу цьому елементу. Процедурна генерація реалізується за допомогою об'єктів які позначають початок та кінець рівня та допоміжних скриптів.

Для реалізації процедурної генерації рівнів необхідно створити 2 скрипта: `Chunk` та `ChunkPlacer`. `Chunk` – скрипт який відповідає за координати точок старту та закінчення локації (чанків), див. рис 3.8. Цей скрипт необхідно розмістити на кожній локації.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

```

1  using System.Collections.Generic;
2  using UnityEngine;
3
4  public class Chunk : MonoBehaviour
5  {
6      public Transform Begin;
7      public Transform End;
8
9      public List<Enemy1Script> EnemyList = new List<Enemy1Script>();
10
11     public MovementStopTrigger mST;
12
13     public Fight fight;
14
15     void Start()
16     {
17
18         for (int i = 0; i < EnemyList.Count; i++)
19         {
20             EnemyList[i].gameLogic.resetButton.onClick.AddListener(EnemyList[i].EnemyFight);
21         }
22     }
23
24
25     void Update()
26     {
27
28     }
29 }
30

```

Рисунок 3.8 – Екземпляр коду зі скрипта Chunk.cs

ChunkPlacer – скрипт який містить інформацію про позицію гравця та перелік локацій, які необхідно створити. Усі операції по створенню та з'єднанню локацій також необхідно реалізувати у скрипті ChunkPlacer, див. рис 3.9. Цей скрипт необхідно розмістити на пустому об'єкті.

Згідно наведеного коду, на початку у динамічний масив заноситься початковий чанк з якого гравець починає грати. Це робиться за для того, щоб у майбутньому можна було реалізувати видалення пройдених локацій, які займають місце в оперативній пам'яті.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

```
Скрипт Unity (1 ссылка на ресурсы) | Ссылка: 0
public class ChunksPlacer : MonoBehaviour
{
    public Transform Player;
    public Chunk[] ChunkPrefabs;
    public Chunk FirstChunk;

    private List<Chunk> spawnedChunks = new List<Chunk>();

    public GameLogic gameLogic;

    Сообщение Unity | Ссылка: 0
    void Start()
    {
        spawnedChunks.Add(FirstChunk);
    }

    Сообщение Unity | Ссылка: 0
    void Update()
    {
        if (Player.position.x < spawnedChunks[spawnedChunks.Count - 1].End.position.x)
        {
            SpawnChunk();
        }
    }

    Ссылка: 1
    private void SpawnChunk()
    {
        Chunk newChunk = Instantiate(ChunkPrefabs[Random.Range(0, ChunkPrefabs.Length)]);
        for (int i = 0; i < newChunk.EnemyList.Count; i++)
        {
            newChunk.EnemyList[i].gameLogic = gameLogic;
            newChunk.EnemyList[i].gameLogic.Enemys.Add(newChunk.EnemyList[i]);
            newChunk.EnemyList[i].fight = gameLogic.fight;
            newChunk.EnemyList[i].Cards = gameLogic.cardsSkript;
        }

        newChunk.mST.Cards = gameLogic.cards;

        newChunk.transform.position = spawnedChunks[spawnedChunks.Count - 1].End.position;
        spawnedChunks.Add(newChunk);

        for (int i = 0; i < spawnedChunks[spawnedChunks.Count - 1].EnemyList.Count; i++)
        {
            gameLogic.resetButton.onClick.RemoveListener(spawnedChunks[spawnedChunks.Count - 1].EnemyList[i].EnemyFight);
        }
    }
}
```

Рисунок 3.9 – Екземпляр коду зі скрипта Chunk.cs

У методі Update ми перевіряємо позицію гравця відносно позиції закінчення локації по координаті x. Якщо координати гравця по x більше, ніж координати по x кінцевої точки локації, спрацьовує метод SpawnChank(). У цьому метод ми створюємо новий чанк, заносимо данні до ворогів та додаємо ворогів у динамічний масив. Якщо масив буде пустий, це означатиме, що всі вороги на локації переможені, та гравець може йти далі.



Рисунок 3.10 – Приклад роботи процедурної генерації

Згідно з рис. 3.10, нова локація створюється на кінці попередньої локації. Точка Begin нової локації приєднується до точки End минулої локації. Таким чином локації будуть генеруватися рівно, щоб не було видимих прогалів.

Також є необхідність додати функцію, щоб локація генерувалася наприклад за 10 кроків до кінця чанку, щоб створити уявлення, що це один

великий шлях. Гравець не повинен побачити, як створюється локація. На рис 3.11 зображено результат роботи, на якому можна побачити перехід між двома різними локаціями.



Рисунок 3.11 – Результат роботи процедурної генерації.

3.4 Реалізація ігрового меню

Для реалізації ігрового меню використано внутрішню систему UI інтерфейсу Unity. Коли гравець запускає гру, одне з перших, що він має побачити – ігрове меню. Ігрове меню – вказівник по грі. Ігрове меню складається з:

- назва гри;
- кнопка «Грати» – переходить до гри;
- кнопка «Магазин» – переходить до магазину, у якому можна купувати речі;
- кнопка «Інвентар», у якому можна налаштовувати екіпірування;
- кнопка «Налаштування» – відповідає за налаштування гри;
- вікно з новинами;
- кнопка «Вихід» – виходить з гри.

На рис. 3.12 можна побачити розміщення кнопок. Після кожного оновлення гри та додання нових функцій адміністратор або розробник гри повинен оновлювати вікно з новинами для швидкого інформування користувачів.



3.12 – Ігрове меню

У меню інвентаря гравець обирає необхідні картки та екіпірування. У цьому меню наведені всі картки, які має гравець. Натисканням на картку він обирає ті, які хоче взяти з собою. Якщо карток які взяв гравець забагато, це ускладнює тактику гравця, та може призвести до короткої гри. Якщо карток замало – гра буде одноманітною, також гравець не буде достатньо підготовлений до всіх випадків. Слід дотримуватися балансу, брати тільки те, що необхідно. Меню з картками можна побачити на рис. 3.4.

У вкладці екіпірування є кнопка «Предмети». Переходячи у цю вкладку, користувач може побачити список предметів, схожий на список карток. Кожен предмет має свої характеристики відповідно до його категорії. Так наприклад не кожен меч не може мати характеристики, які надають гравцеві додаткового здоров'я, так і не кожен наколінник може надавати додатковий урон. Деякі предмети високої якості можуть мати спеціальні унікальні здібності, на

приклад помножувати урон гравця вдвічі. Усього можна взяти з собою не більше трьох предметів. У майбутньому планується зробити платне видання, яке б надавало гравцю змогу носити з собою до шести предметів.

У меню налаштувань користувач виставляє особисті вподобання до гри, такі як: гучність музики, мову інтерфейсу, налаштування графіки та інші параметри. Ці параметри зберігається у файлі на пристрої, а данні гравця повинні зберігатися на спеціальному орендованому сервері.

На сторінці магазину можна придбати як предмети для посилення ігрового персонажа так і косметичні прикраси для героя, наприклад новий зовнішній вигляд, або кольорову схему дня нього. Ці косметичні предмети не впливають на гру, вони зроблені за для краси. Більшість з косметичних предметів можна купити тільки за спеціальну валюту. Цю валюту можна купити за реальні гроші.

На відміну від ігровий та косметичних предметів, картки не можна купити у магазині. Картки можна отримати лише за виконані завдання, досягнення нового рівня, або з невеликим шансом нова картка може випасти з переможеного ворога. Якщо гравцю випадає картка, яка у нього уже є, він отримує винагороду, автоматично продаючи цю картку.

4 ОПТИМІЗАЦІЯ ТА ТЕСТУВАННЯ СТВОРЕННОГО ЗАСТОСУНКУ

4.1 Оптимізація ігрової логіки

4.1.1 Оптимізація ігрового середовища

Під час створення процедурної генерації локацій слід пам'ятати про оптимізацію. За допомогою скрипта, нові чанки створюються, але непотрібні не видаляються.

Для цього необхідно створити динамічний масив, до якого закосяться ті чанки, які зараз на сцені ігрового рушія. Після цього, якщо кількість активних чанків перевищує три, або більше, необхідно видалити перший, потім другий стає першим, та коли ми переходимо до наступної локації, їх знову стає три, та перший видаляється.

Щоб не виникло помилок, необхідно перший чанк, який у нас завжди один й той самий одразу занести у масив.

```
if (spawnedChunks.Count >= 3)
{
    Destroy(spawnedChunks[0].gameObject);
    spawnedChunks.RemoveAt(0);
}
```

Рисунок 4.1 – Код оптимізації генерації локацій

Перш за все необхідно позбутися асетів, які наразі не потрібні. Додаткові матеріали, моделі, анімації та інші асети, які не використовуюся, займають місце на диску. Внаслідок цього проект буде довше збиратися та буде займати більше пам'яті.

Для цього усі непотрібні асети необхідно видалити. Перед цим необхідно зробити план. Якщо проект буде й надалі оновлюватися, необхідно

залишити ті асети, які необхідні для наступного оновлення додатку, щоб уникнути їх повторного завантаження.

Щоб перевірити продуктивність додатка можна скористатися спеціальною утилітою profiler, яка інтегрована в Unity. Щоб скористатися даною утилітою, необхідно перейти до вкладки Window > Analysis > Profiler. Після цього відкриється вікно profiler, яке можна переміщувати та закріпити біля інспектора. Щоб оцінити продуктивність додатка необхідно просто натиснути кнопку запустити проект та відслідковувати продуктивність FPS на екрані. На рис. 4.2, можна побачити результат який повністю задовольняє. Результат тесту на продуктивність 100 FPS при активних діях, та 200 FPS у стані спокою. Рекомендована частота кадрів для проектів на мобільних пристроях – 30 FPS, для консольних та настільних додатків – 60 FPS, а для проектів, які використовують технологію VR – необхідно як найменш 90 FPS. Для останнього це необхідно, щоб не постраждав вестибулярний апарат. Якщо FPS буде замало, у гравця з'являться побічні ефекти, такі як: нудота, запаморочення, головна біль та біль в очах [25].

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity

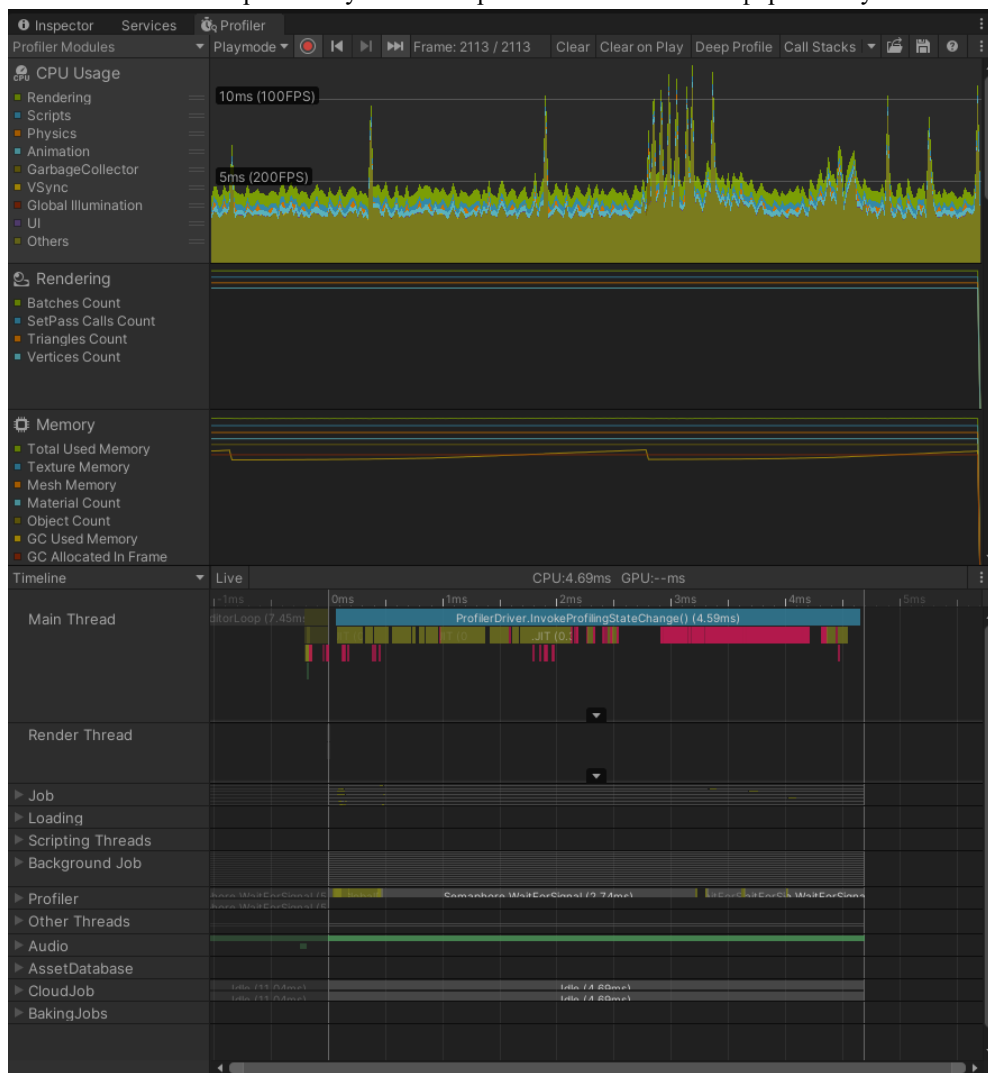


Рисунок 4.2 – Результат роботи утиліти profiler

Якщо буде замало FPS у настільних, консольних та мобільних додатках, користувач відчує незручність; це може привести до дратівливості. Ось, чому так важливо приділяти особливу увагу до оптимізації проєктів.

4.2 Тестування додатку

Тестування проєкту, є не менш важливим етапом, як і оптимізація. Тестування продукту необхідно за для запобігання наявності дефектів додатку. Тестування починається з головного меню. Саме головне меню бачить користувач, коли починає грати у гру. У меню є кнопки, натиснувши на які, можна переміститися до іншого меню, наприклад кнопка «Settings»

відкриє меню з налаштуваннями, в якому можна змінити гучність саундтреку.

Переходячи до меню інвентаря, гравець може побачити список карток, які він може використовувати. Натискаючи на картку можна побачити, що вона міняє колір та стає більш сірою – це означає що картка не належить до активної колоди, та її не буде, коли гравець почне грати.

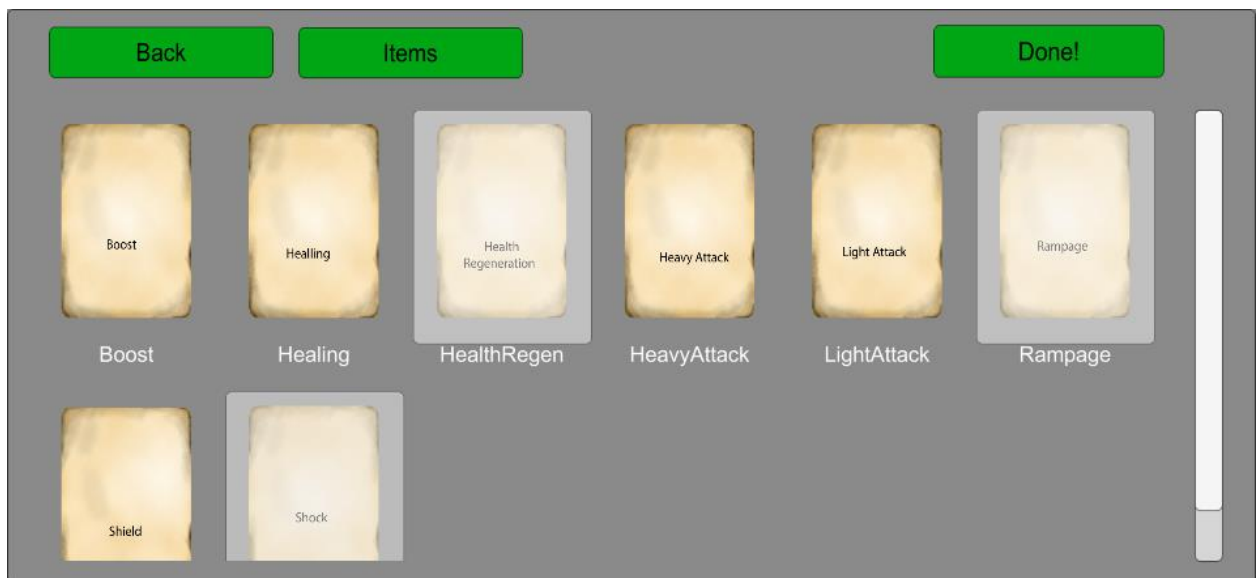


Рисунок 4.3 – Вибір карток

У меню предмети можна вибрати ті предмети, які гравець хоче взяти з собою, під час гри. Обрати можна тільки 3 предмети.

Після того, як предмети були обрані та налаштування вдовольняють побажанням гравця, можна почати гру, натиснувши на кнопку «Start». Після цього гравець переноситься на першу локацію, звідки він починає гру. Гравець переміщується до ворогів, перемагає їх, та пересувається далі. Якщо у гравця закінчиться здоров'я, гра закінчиться, гравець побачить результат, та зароблені кошти. Після таблиці він повернеться у меню. Переходячи до меню магазину, гравець може витратити зароблені монети на придбання деяких речей. До цих речей входить:

- Ігрові предмети:

- Екіпірування;
- Особливі акційні предмети з обмеженою пропозицією.
- Косметичні предмети для героя;
- Бонуси та прискорювачі прогресу;
- Особливі статуси «Преміум»;
- Інші предмети без категорії.

Усі помилки та баги, які були знайдені під час тестування додатку були швидко вирішені. Натиснувши на кнопку «Exit», гра вимикається. Таким чином гравець виходить з гри.

Висновки до розділу 4

Під час виконання кваліфікаційної роботи було задіяно дуже багато зусиль. Була проведена велика та складна робота. Ця робота створена за для перевірки, та покращення навичок, здобутих під час навчання. Дуже важливою частиною роботи – є оптимізація та тестування зробленого проекту, тому що проект, який не пройшов перевірку не може бути експлуатованим. Такі продукти загалом містять чимало багів, та мають погану оптимізацію, що означає повну або частичну несправність проекту.

Спеціальний розділ

ОХОРОНА ПРАЦІ

до кваліфікаційної роботи

на тему:

«РОЗРОБКА ВІЗУАЛЬНО-ІНТЕРАКТИВНОЇ RPG НА ПЛАТФОРМІ UNITY»

Спеціальність 122 «Комп'ютерні науки»

122 – БКР – 402.21810215

Виконав студент 4-го курсу, групи 402

Д. Ю. Коверзнев

(підпис, ініціали та прізвище)

«20» червня 2022 р.

Консультант А. О. Алексєва

(наук. ступінь, вчене звання)

(підпис, ініціали та прізвище)

«20» червня 2022 р.

Миколаїв – 2022

ВСТУП

Жодне підприємство сьогодні не обходиться без використання інформаційних технологій, а саме роботи технічних пристроїв. Персональний комп'ютер використовується для збору, обміну та аналізу інформації всередині підприємства та між іншими установами. Використання персонального комп'ютера дуже полегшує різні процеси на підприємстві. Попри позитивні наслідки комп'ютер може бути шкідливим здоров'ю людини. Використання комп'ютера загостило проблеми суспільного, а насамперед власного здоров'я, що вимагає удосконалення існуючих робочих місць та розробки нових, більш позитивних до здоров'я робочих місць та вимагає проведення профілактичних заходів до запобігання розвитку негативних наслідків впливу персонального комп'ютера на здоров'я користувача.

Надмірне використання комп'ютера може зробити досить багато проблем. При роботі з комп'ютером слід дотримуватись правил, які значно запобігають шкідливості спини, зору, суглобів та електро-магнітного випромінювання.

При роботі з комп'ютером людина повністю залежить від положення дисплея. Крім того, зображення на екрані динамічно оновлюється, а низька частота оновлення викликає його мерехтіння. При цьому очні і внутрішньо очні м'язи, фокусують погляд, втомлюються від надмірного навантаження. Розвивається зорове стомлення, що сприяє виникненню короткозорості. Тривала робота з комп'ютером вимагає також підвищеної зосередженості, що призводить до появи головного болю, дратівливості, нервової напруги і стресу. Низька чистота оновлення кадрів або низька швидкість роботи комп'ютера може викликати нервовий розлади.

5.1 Нормативна документація щодо забезпечення охорони праці під час використання екранними пристроями

Існують правила, які працівники підприємств, установ повинні дотримуватися під час використання електронно-обчислювальних машин. Дотримання цих правил знизить наслідки несприятливої дії на працівників шкідливих та небезпечних факторів, які можуть супроводжуватися під час роботи з електронно-обчислювальними машинами. По перше, це стосується зорових та нервово-емоційних перевантажень, серцево-судинних захворювань та психічної складової працівника.

Нижче наведені загальні вимоги до приміщень.

Облаштування робочих місць, обладнаних відеотерміналами, повинно забезпечувати:

- належні умови освітлення приміщення і робочого місця, відсутність відблисків;
- оптимальні параметри мікроклімату (температура, відносна вологість, швидкість руху, рівень іонізації повітря);
- належні ергономічні характеристики основних елементів робочого місця.

А також враховувати такі небезпечні і шкідливі фактори:

- наявність шуму та вібрації;
- м'яке рентгенівське випромінювання;
- електромагнітне випромінювання;
- ультрафіолетове і інфрачервоне випромінювання;
- електростатичне поле між екраном і оператором;
- наявність пилу, озону, оксидів азоту й аероіонізації.

Умови праці користувачів ПК мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПіН 3.3.2.007-98, затверджених

постановою Головного державного санітарного лікаря України від 10.12.1998 № 7 (далі - ДСанПіН 3.3.2.007-98), та Правил № 65 [2].

Вид трудової діяльності, робочі місця	Рівні звукового тиску в дБ									
	в октавних смугах із середньгеометричними частотами, Гц									
	31,5	63	125	250	500	1000	2000	4000	8000	Рівні звуку, еквівалентні рівні звуку, дБА/дБАекв.
Програмісти ЕОМ	86	71	61	54	49	45	42	40	38	50
Оператори в залах обробки інформації на ЕОМ та оператори комп'ютерного набору	96	83	74	68	63	60	57	55	54	65
В приміщеннях для розташування шумних агрегатів ЕОМ	103	91	83	77	73	70	68	66	64	75

Рисунок 5.1 – Допустимі рівні звуку, еквівалентні рівні звуку і рівні звукового тиску в октавних смугах частот

ДСанПіН 3.3.2.007-98 поширюються на умови й організацію праці під час робіт з ВДТ усіх типів вітчизняного та зарубіжного виробництва на основі електронно-променевих трубок (ЕПТ), що використовуються в ЕОМ колективного та персонального використання.

Сьогодні ж у більшості офісних приміщень встановлено нові сучасні комп'ютери з рідкокристалічними моніторами, для яких не прописано гігієнічні вимоги у ДСанПіН 3.3.2.007-98. Водночас вимоги ДСанПіН

3.3.2.007-98 до виробничих приміщень, параметрів виробничого середовища приміщень, організації і обладнання робочих місць, режимів праці та відпочинку є актуальними під час робіт з будь-якими комп'ютерами та моніторами.

Відповідно до пунктів 4.1, 4.2 Правил № 65 ЕОМ з ВДТ і ПП під час експлуатації мають відповідати вимогам стандартів, вказаних Правил і пройти державну санітарно-епідеміологічну експертизу згідно з Порядком проведення державної санітарно-епідеміологічної експертизи, затвердженим наказом МОЗ України від 09.10.2000 № 247 (у редакції наказу МОЗ України від 14.03.2006 № 120) [3].

ЕОМ з ВДТ і ПП закордонного виробництва мають відповідати вимогам національних стандартів держав-виробників і мати відповідну позначку на корпусі, у паспорті або в іншій експлуатаційній документації. Зазначені вище нормативні документи - обов'язкові для виконання Вони визначають критерії безпечного використання комп'ютерної техніки та призначені для запобігання несприятливої дії на працівників шкідливих факторів, що супроводжують роботу з візуальними дисплейними терміналами.

Керівники державних органів, підприємств, установ, організацій незалежно від форми власності й підпорядкування у порядку забезпечення контролю зобов'язані облаштувати робочі місця користувачів ЕОМ та персональних ЕОМ (ПЕОМ) з ВДТ відповідно до вимог Правил № 65. Керуючись зазначеними вище нормативними документами можна дійти висновку, що для забезпечення комфортної роботи оператора ПК необхідно звернути увагу на такі основні вимоги:

- стан виробничих приміщень;
- характеристики та стан техніки (ПК, монітори, принтери, наявність заземлення тощо);
- організація робочого місця;
- дотримання режимів праці та відпочинку;

- моніторинг стану здоров'я працівника (стан органів зору, показники захворювань з тимчасовою втратою працездатності, ін.);
- профілактичні заходи.

Так робоче місце оператора має відповідати вимогам ГОСТ 12.2.032-78 [4].

Вимоги до приміщення з використанням екранних пристроїв.

Усі приміщення, де робітники використовують персональні комп'ютери, повинні відповідати ряду вимог. Ці вимоги перевіряються лише спеціальними службами, які призначені для нагляду за дотриманням вимог охорони праці на підприємствах.

Робочі місця працівників з екранними пристроями мають бути:

- спроектовані так і мати такі розміри, щоб працівники мали простір для зміни робочого положення та рухів;
- для забезпечення безпеки та захисту здоров'я працівників усе випромінювання від екранних пристроїв має бути зведене до гранично допустимого рівня з погляду безпеки та охорони здоров'я працівників;
- організація робочого місця працівника з екранними пристроями має забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним, антропологічним, психофізіологічним вимогам, а також характеру виконуваних робіт;
- освітлення робочого місця працівника з екранними пристроями має створювати відповідний контраст між екраном і навколишнім середовищем (з урахуванням виду роботи) та відповідати вимогам ДСанПІН 3.3.2.007-98;
- мікроклімат виробничих приміщень з робочими місцями працівників з екранними пристроями має підтримуватись на постійному рівні та відповідати вимогам Санітарних норм мікроклімату виробничих приміщень ДСН 3.3.6.042-99 [5];

– робочий стіл або робоча поверхня повинні бути достатнього розміру та мати поверхню з низькою відбивною здатністю, допускати гнучкість під час розміщення екрана, клавіатури, документів і відповідного устаткування;

– робоче крісло має бути стійким і дозволяти працівнику з екранними пристроями легко рухатися та займати зручне положення. Сидіння має регулюватися по висоті, спинка сидіння - як по висоті, так і по нахилу. Слід передбачати підніжку для тих, кому це необхідно для зручності.

Мінімальні вимоги безпеки під час роботи:

- щодня перед початком роботи необхідно очищати екранні пристрої від пилу та інших забруднень;
- після закінчення роботи екранні пристрої слід відключати від електричної мережі;
- у разі виникнення аварійної ситуації необхідно негайно відключити екранний пристрій від електричної мережі.

Не допускається:

- виконувати технічне обслуговування, ремонт і налагодження екранних пристроїв безпосередньо на робочому місці працівника під час роботи з екранними пристроями;
- відключати захисні пристрої, самочинно проводити зміни у конструкції та складі екранних пристроїв або їх технічне налагодження;
- працювати з екранними пристроями, у яких під час роботи виникають нехарактерні сигнали, нестабільне зображення на екрані та інші несправності.

Мінімальні вимоги безпеки до екранних пристроїв:

- екранні пристрої не мають бути джерелом ризику для працівників;
- усе випромінювання, за винятком видимої частини електромагнітного спектра, має бути зведене до незначного рівня з погляду безпеки і охорони здоров'я працівників;

- символи на екранних пристроях мають бути чіткими, відповідного розміру. Між символами і рядками символів має бути належна відстань;
- зображення на екрані має бути стабільним, без миготінь або інших видів нестабільності;
- яскравість та/або контрастність символів має легко регулюватися працівником під час роботи з екранними пристроями, а також швидко адаптуватися до навколишніх умов;
- вибираючи екрани, слід надавати перевагу таким екранам, які легко та вільно повертаються і нахиляються відповідно до потреби працівника;
- за необхідності може використовуватись окрема підставка або регульований стіл для розміщення екрана;
- екран не має відблискувати або відбивати світло, щоб не викликати дискомфорту у працівника під час роботи з екранними пристроями;
- вибираючи клавіатуру, слід надавати перевагу такій клавіатурі, яка відкидається і є автономною (відокремленою від екрана), щоб працівник міг вибрати зручну робочу позу й уникнути втоми рук (кисті і верхньої частини руки);
- поверхня клавіатури має бути матовою, щоб уникнути віддзеркалювання. Розташування клавіш і самі клавіші мають полегшувати роботу із клавіатурою. Позначення клавіш повинно бути достатньо контрастним і розбірливим;
- устаткування, яке входить до робочої станції, не має виділяти надлишкового тепла, що може спричинити незручності працівникам під час роботи з екранними пристроями;
- під час розробки, вибору, замовлення та модифікації програмного забезпечення, а також під час розробки завдань, що передбачають використання устаткування з екранними пристроями, роботодавець має керуватися таким програмним забезпеченням, яке відповідає

розв'язуванням завданням і є простим у використанні, а де необхідно - адаптованим до рівня знань і досвіду працівника [1].

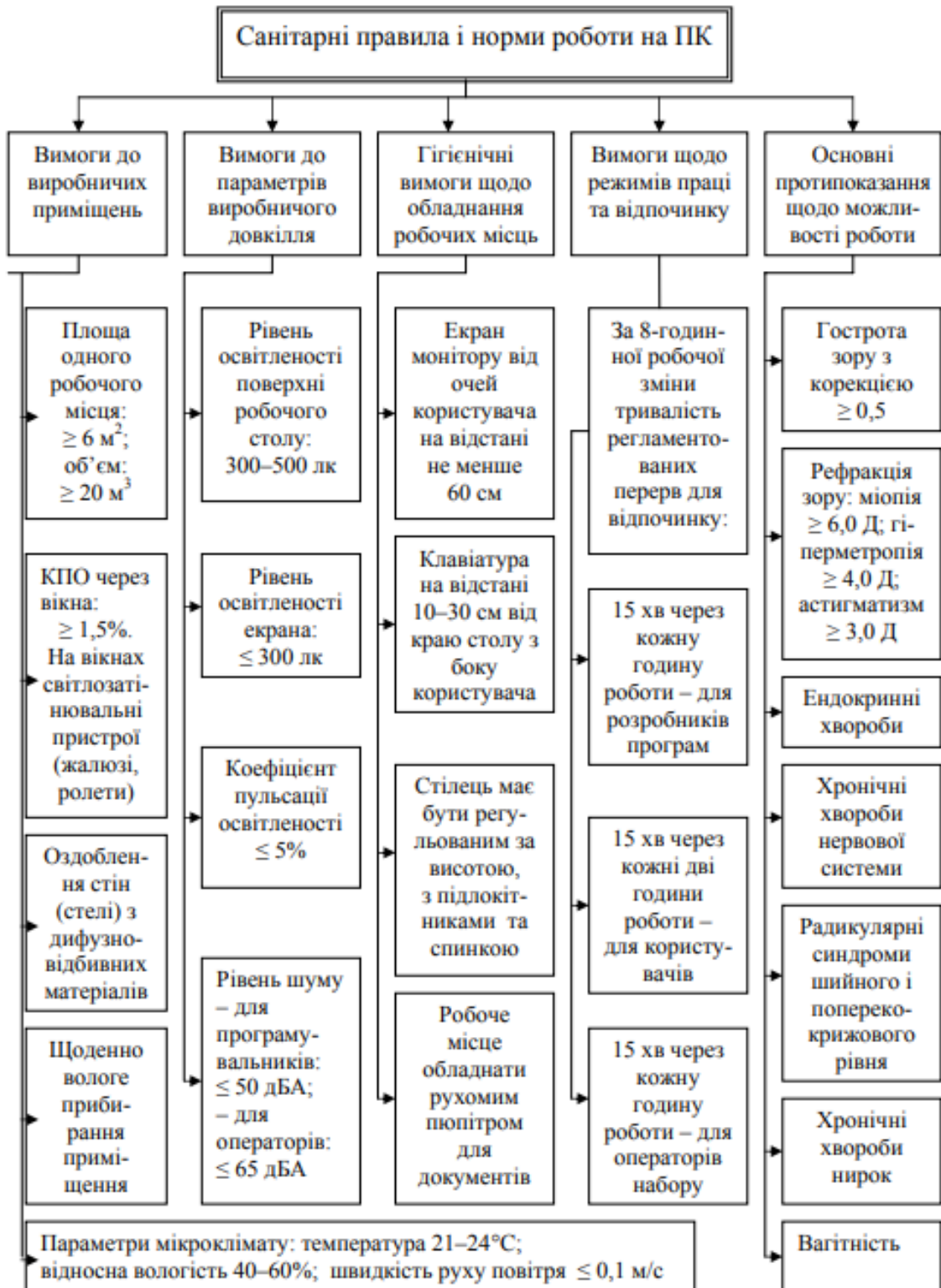


Рисунок 5.2 – Блок-схема реалізації робіт на ПК

Вимоги до роботи з екранними пристроями.

На сьогодні існує ряд вимог, які повинні використовувати працівники на підприємствах, основним місцем роботи яких є місце за персональним комп'ютером. Нижче наведено частину вимог з наказу Міністерством соціальної політики України «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» на стан 18 травня 2018 року:

- робочий стіл або робоча поверхня повинні бути достатнього розміру і мати низьку відбивну здатність, допускати гнучкість при розміщенні екрану, клавіатури, документів і відповідного обладнання;
- робоче крісло повинно бути стійким і дозволяти працівнику легко рухатися і займати зручне положення. Сидіння має регулюватися по висоті, спинка сидіння – як по висоті, так і по нахилу;
- варто передбачати підніжку для тих, кому це необхідно для зручності. Щодня перед початком роботи необхідно очищати екранні пристрої від пилу і інших забруднень;
- не допускається робота з екранними пристроями, в яких під час роботи з'являються нехарактерні сигнали, нестабільне зображення на екрані та інші несправності;
- також встановлено мінімальні вимоги безпеки до екранних пристроїв. При будь-якій роботі потрібно дотримуватися певних правил поведінки і безпеки, щоб зберегти своє здоров'я і уберегтися від можливих травм або будь-яких захворювань;
- профілактика краще лікування, тому правила роботи за комп'ютером необхідно знати всім, адже ми все більше і більше часу проводимо саме за комп'ютером - за ним працюємо на роботі, і за ним же сидимо вдома.

У відділі з питань гігієни праці Головного управління Держпраці у Київській області повідомляють, що вимоги щодо безпеки та захисту здоров'я

працівників під час роботи з екранними пристроями затверджені наказом Мінсоцполітики від 14 лютого 2018 року № 207. Вимоги поширюються на всіх суб'єктів господарювання незалежно від форм власності, організаційно-правової форми і видів діяльності та встановлюють мінімальні вимоги безпеки та захисту здоров'я під час здійснення роботи, пов'язаної з використанням екранних пристроїв незалежно від їхнього типу та моделі.

Ці вимоги не обмежують права роботодавця встановлювати більш жорсткі та/або спеціальні вимоги безпеки і захисту здоров'я та життя працівників під час роботи з екранними пристроями, якщо це не суперечить чинному законодавству.

Встановлено мінімальні вимоги до безпеки здоров'я працівників при роботі з екранними пристроями.

Вимоги не стосуються комп'ютерних класів (кабінетів, аудиторій тощо) навчальних закладів та робочих місць співробітників, що займаються обслуговуванням, ремонтом і налагодженням екранних пристроїв. Встановлено обов'язок роботодавця проінформувати працівників під розписку про умови праці та наявність на їх робочих місцях небезпечних і шкідливих виробничих факторів (фізичних, хімічних, біологічних, психофізіологічних), які виникають при роботі з екранними пристроями та ще не усунуті, а також про можливі наслідки їх впливу на здоров'я.

Серед найбільш небезпечних факторів при роботі з комп'ютером-це фактори важкості та напруженості трудового процесу – малорухливий характер праці. Більшу частину робочого часу працівники перебувають на робочому місці в положенні сидячи, виконують однотипну роботу (монотонність праці).

Тривала робота з документами та комп'ютером спричинює перенапруження зорового аналізатора. А робота, пов'язана з високим ступенем відповідальності, може призводити до нервово-емоційного

виснаження. Розумова праця передбачає розв'язання великої кількості завдань, а отже, загрожує розумовим перенапруженням.

Державними санітарними нормами та правилами «Гігієнічна класифікація праці за показниками шкідливості та небезпечності факторів виробничого

середовища, важкості, затвердженими наказом Міністерства охорони здоров'я України від 8 квітня 2014 р. № 248 визначено, що умови праці - це сукупність факторів виробничого середовища і трудового процесу, які впливають на здоров'я і працездатність людини під час виконання нею трудових обов'язків [7]. А шкідливими умовами праці є стан умов праці, за якого рівень впливу одного або більше факторів виробничого середовища та/або трудового процесу перевищує допустимий.

Роботодавець має забезпечити за свій рахунок проведення медичних оглядів працівників відповідно до вимог Порядку проведення медичних оглядів працівників певних категорій, затвердженого наказом МОЗ України від 21 травня 2007 року № 246 і при необхідності – виконання відповідних оздоровчих заходів [8].

5.2 Забруднення повітря на робочих місцях з використанням екранних пристроїв.

Чимало досліджень було присвячено визначенню хімічного складу повітря на робочих місцях операторів ВДТ. Багатьма дослідниками було відмічено, що до кінця робочого дня в повітря робочої зони різко зростає концентрація CO₂ яка сягала від 0,12-0,13 до 0,19% (в атмосферному повітрі CO₂ міститься 0,03%) [9].

В Німеччині опубліковані дані про те, що на робочих місцях операторів ВДТ виявлені діоксин та фуран, які викликають ракові захворювання. Як виявилось, ці шкідливі для здоров'я сполуки входили на правах, так званих,

полібромованих протипожежних речовин до складу пластмас, з яких виготовляли електронні плати та корпуси дисплеїв.

Слід зазначити, що в окремих публікаціях вказується на те, що користувачі комп'ютерів, які носять окуляри більш схильні до розладів функції зору. Це пояснюється тим, що для нормальної роботи користувача за дисплеєм комп'ютера, як правило, необхідні інші окуляри ніж ті, які вони використовують для читання. Останні мають фокусну відстань 30 см, а при роботі за дисплеєм комп'ютера фокусна відстань окулярів повинна бути [10].

В інших дослідженнях було виявлено, що концентрації поліхромованих біфенілів (ПХБ) в приміщеннях (офісах) з ВДТ в кілька разів перевищували значення ПХБ для приміщень без ВДТ. Дослідники припускають, що ПХБ можуть виділятися конденсаторами та трансформаторами ВДТ. Важливо підкреслити, що концентрації вищезазначених речовин лише у рідкісних випадках перевищували гранично допустимі концентрації (ГДК). В той же час, у більшості досліджень серед речовин, у яких було виявлено перевищення ГДК у повітрі біля робочих місць з ВДТ найчастіше називались озон, оксиди азоту, пил.

Особливу небезпеку щодо впливу на здоров'я представляє підвищена концентрація озону - високотоксичного подразнюючого газу. З цієї причини він був внесений у список речовин, максимальні значення концентрації яких на робочих місцях обмежені та строго визначені. Надзвичайна небезпека озону для здоров'я людини пов'язана з тим, що він належить до так званих радіоміметичних речовин – хімічних сполук, що викликають в живих організмах зміни, схожі з тими, які виникають після дії іонізуючого випромінювання. Тому озон вважається не лише подразнюючою, а й канцерогенною речовиною.

Основними джерелами озону на комп'ютеризованих місцях є ЕПТ ВДТ та лазерні принтери. З огляду на це, необхідно виключати ВДТ у випадках, коли він не використовується, а лазерний принтер бажано розташовувати

подали від робочого місця оператора. Однак, це додаткові заходи, основним же заходом щодо запобігання несприятливого впливу озону та інших шкідливих речовин на здоров'я операторів є забезпечення функціонування припливно-витяжної вентиляції. Для того, щоб шкідливі речовини не проникали із сусідніх приміщень в приміщеннях з ВДТ необхідно створити деякий надлишковий тиск.

Відповідно до ГОСТ 12.1.005-88 [12] вміст озону в повітрі робочої зони не повинен перевищувати 0,1 мг/м³; вміст оксидів азоту - 5 мг/м³; вміст пилу – 4 мг/м³.

Таблиця 5.1 – Гранично допустимі концентрації (ГДК) шкідливих речовин у повітрі робочої зони

Найменування речовини	Величина ГДК, мг/м	Переважаючий агрегатний стан в умовах виробництва	Клас небезпеки	Особливості дії на організм
Озон	0,1	п	I	О

Де:

п - пари та/або газу;

Під час виконання спеціальної частини з охорони праці було проаналізовано умови праці людини на підприємствах та установах. Також було досліджено роботу при користуванні електронними пристроями.

На кожному підприємстві є працівники, які працюють з електронними пристроями, тому вказані у роботі нормативні правила є актуальними. Під час роботи з екранним дисплеєм людина може відчувати втоми, запаморочення, біль в очах, спині, напруженість та стрес, порушення роботи суглобів.

На сьогодні існує багато шкідливих умов, які погіршують процес трудової діяльності людини, такі як: шум, вібрація та інші умови.

Підприємство зобов'язано виконувати усі вимоги та нормативи охорони праці під час трудової діяльності працівника. Дотримання поставлених вимог до працівників та власників підприємств дозволить мінімізувати шкідливі наслідки, які мають вплив на здоров'я людини.

ВИСНОВКИ

На початку розробки даної кваліфікаційної роботи був проведений детальний аналіз предметної області проекту. Були проаналізовані аналоги, та зазначені перспективні напрямки розробки. Базуючись на отриманих даних, було прийнято рішення про розробку даного програмного застосунку, через високу актуальність.

У наступному розділі наведені інструменти планування та розробки проекту. Була доведена доцільність використання таких інструментів як: Autodesk Maya, Unity, Visual Studio, Microsoft Visio та Trello.

Після проектування, була розпочата безпосередньо розробка проекту. На початку, були зроблені тривимірні декоративні моделі, після цього базова механіка пересування вздовж локації. На наступному кроці було реалізовано взаємодію гравця з ворогами, а після того процедурну генерацію рівнів та саму систему RPG. Наприкінці було створено ігрове меню, проведена оптимізація та тестування проекту.

Результатом роботи над проектом є готова візуально-інтерактивна RPG, яка створена на платформі Unity. Додаток створений для операційних систем Android, та IOS. Також є можливість запуску гри на операційній системі Windows. MacOS та Linux.

Кафедра інтелектуальних інформаційних систем
Розробка візуально-інтерактивної RPG на платформі Unity
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Метт Бартон. Історія комп'ютерних рольових ігор Частина 1: Ранні роки монографія (1980-1983) (дата звернення: 28.05.2022).
2. Адамс Е., Ендрю Р., Andrew Rollings and Ernest Adams on game design: Ендрю Роллінгс та Ернест Адамс про ігровий дизайн: навч. посіб. — New Riders Publishing 2003. (дата звернення: 28.05.2022).
3. Needs Met Through Role-Playing Games: A Fantasy Theme Analysis of Dungeons & Dragons: наукова стаття. URL: <https://opensiuc.lib.siu.edu/kaleidoscope/vol12/iss1/6/> (дата звернення: 29.05.2022).
4. Мур М. Основи ігрового дизайну: навч. посіб. CRC Press, 2016. (дата звернення: 29.05.2022).
5. World of Warcraft [Електронний ресурс] – Режим доступу: https://uk.wikipedia.org/wiki/World_of_Warcraft (дата звернення: 29.05.2022).
6. Albion Online [Електронний ресурс] – Режим доступу: https://en.wikipedia.org/wiki/Albion_Online (дата звернення: 30.05.2022).
7. The Elden Scrolls V: Skyrim [Електронний ресурс] – Режим доступу: https://uk.wikipedia.org/wiki/The_Elder_Scrolls_V:_Skyrim (дата звернення: 30.05.2022).
8. Dark Souls III [Електронний ресурс] – Режим доступу: https://ru.wikipedia.org/wiki/Dark_Souls_III (дата звернення: 30.05.2022).
9. Gothic [Електронний ресурс] – Режим доступу: [https://uk.wikipedia.org/wiki/Gothic_\(%D0%B2%D1%96%D0%B4%D0%B5%D0%BE%D0%B3%D1%80%D0%B0\)](https://uk.wikipedia.org/wiki/Gothic_(%D0%B2%D1%96%D0%B4%D0%B5%D0%BE%D0%B3%D1%80%D0%B0)) (дата звернення: 30.05.2022).
10. Кюблер С., Робер Ж., Дерижан У., Вуазен А., Ив Ле Траон. ФАНР [Електронний ресурс] – Режим доступу: <https://www.sciencedirect.com/science/article/abs/pii/S0957417416304638> (дата звернення: 30.05.2022).

11. Unity [Електронний ресурс] – Режим доступу: <https://unity.com/> (дата звернення: 30.05.2022).
12. Git [Електронний ресурс] – Режим доступу: <https://git-scm.com/> (дата звернення: 30.05.2022).
13. Visual Studio [Електронний ресурс] – Режим доступу: <https://visualstudio.microsoft.com/> (дата звернення: 30.05.2022).
14. Autodesk Maya [Електронний ресурс] – Режим доступу: <https://www.autodesk.com/products/maya/overview> (дата звернення: 30.05.2022).
15. 3D Max [Електронний ресурс] – Режим доступу: <https://www.autodesk.com/products/3ds-max/overview?term=1-YEAR&tab=subscription> (дата звернення: 30.05.2022).
16. Wikipedia 3D Max [Електронний ресурс] – Режим доступу: https://uk.wikipedia.org/wiki/Autodesk_3ds_MAX (дата звернення: 30.05.2022).
17. Wikipedia Zbrush [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/Zbrush> (дата звернення: 30.05.2022).
18. Zbrush [Електронний ресурс] – Режим доступу: <https://pixologic.com/> (дата звернення: 30.05.2022).
19. Wikipedia Blender [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/Blender> (дата звернення: 30.05.2022).
20. Blender [Електронний ресурс] – Режим доступу: <https://www.blender.org/> (дата звернення: 30.05.2022).
21. Microsoft Visio [Електронний ресурс] – Режим доступу: <https://www.microsoft.com/uk-ua/microsoft-365/visio/flowchart-software> (дата звернення: 30.05.2022).
22. Wikipedia Microsoft Visio [Електронний ресурс] – Режим доступу: https://uk.wikipedia.org/wiki/Microsoft_Visio (дата звернення: 30.05.2022).
23. Wikipedia Trello [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/Trello> (дата звернення: 30.05.2022).

24. Trello [Електронний ресурс] – Режим доступу: <https://trello.com/uk> (дата звернення: 30.05.2022).
25. Альтон К. Experience, 60 Frames Per Second: Virtual Embodiment and the Player/Avatar Relationship in Digital Games. Наукова стаття [Електронний ресурс] – Режим доступу: <https://journals.sfu.ca/loading/index.php/loading/article/view/178> (дата звернення: 30.05.2022).
26. Андреев Д. Real-time frame rate up-conversion for video games: or how to get from 30 to 60 fps for "free" [Електронний ресурс] – Режим доступу: <https://dl.acm.org/doi/abs/10.1145/1837026.1837047> (дата звернення: 30.05.2022).
27. Хагстром Р. Frames That Matter: The Importance of Frames per Second in Games [Електронний ресурс] – Режим доступу: <https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A857949&dswid=-1861> (дата звернення: 30.05.2022).
28. Офіційний сайт Верховної ради України [Електронний ресурс] – Режим доступу: <https://zakon.rada.gov.ua/go/z0508-18>
29. ДСанПІН 3.3.2.007-98 [Електронний ресурс] – Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0007282-98#Text> - Загол. з екрану.
30. Наказ МОЗ України від 09.10.2000 № 247 [Електронний ресурс] – Режим доступу: <https://www.buh24.com.ua/> - Загол. з екрану.
31. ГОСТ 12.2.032-78.ССБП [Електронний ресурс] – Режим доступу: <https://docs.cntd.ru/document/1200003913> - Загол. з екрану.
32. ДСН 3.3.6.042-99 [Електронний ресурс] – Режим доступу: <https://zakon.rada.gov.ua/rada/show/va042282-99#Text> - Загол. з екрану.
33. Зеркалов Д. В. Охорона праці в галузі. Навчальний посібник. Київ «Основа» 2011 р.

34. Наказ Міністерства охорони здоров'я України від 8 квітня 2014 р. № 248 [Електронний ресурс] – Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0472-14#Text> - Загол. з екрану.
35. Наказ МОЗ України від 21 травня 2007 року № 246 [Електронний ресурс] – Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0846-07#Text> - Загол. з екрану.
36. Забруднення повітря на робочих місцях ВДТ [Електронний ресурс] – Режим доступу: <https://kazedu.com/referat/170221/1> - Загол. з екрану.
37. Опорний конспект лекцій з дисципліни “Охорона праці в галузі” спеціальностей “Комп’ютерні системи та мережі”, “Спеціалізовані комп’ютерні системи” освітньо-кваліфікаційних рівнів “Спеціаліст”, “Магістр”. [Електронний ресурс] – Режим доступу до ресурсу: http://otipb.at.ua/_ld/33/3328_fkit_kiosu_dopg.pdf. – 13 с.
38. Кепич Т.Ю. Охорона праці в галузі / Т.Ю. Кепич, І.Ю. Семенова, М.В. Лавренюк. – К.: КНУ ім.Т. Шевченка, 2013. – 255 с.
39. ГОСТ 12.1.005-88.ССБП [Електронний ресурс] – Режим доступу: <http://docs.cntd.ru/document/1200003608> - Загол. з екрану.

ДОДАТОК А

Код скрипту, відповідний за ігрову логіку

```
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
public class GameLogic : MonoBehaviour
{
    public bool IsEnemyActive;
    public Fight fight;
    public MainMenu mainMenu;
    public List<Enemy1Script> Enemys = new List<Enemy1Script>();
    public GameObject canvas;
    public GameObject cards;
    public Cards cardsSkript;
    public Button resetButton;
    public Button btn;
    void Start()
    {
        resetButton = resetButton.GetComponent<Button>();
    }
    public void Endgame()
    {
        if (fight.PlayerHealth <= 0)
        {
            mainMenu.EndGame();
            Debug.Log("Game Over!");
        }
    }
}
```

ДОДАТОК Б

Код скрипту, відповідний за предмети

```
using System;
using UnityEngine;
using UnityEngine.UI;

[CreateAssetMenu(menuName = "Item")]
public class AssetItem : ScriptableObject, IItem
{
    public string Name => _name;
    public Sprite UIIcon => _uiIcon;

    Text newText;
    [SerializeField] private string _name;
    [SerializeField] private Sprite _uiIcon;
}
```

ДОДАТОК В

Код скрипту відповідний за головне меню

```
public void StartButton()
{
    SceneManager.LoadScene("Game");

    Time.timeScale = 1;
    MainPanel.SetActive(false);
}

public void EquipmentButton()
{
    EquipmentPanel.SetActive(true);
    MainPanel.SetActive(false);
    panel = EquipmentPanel;
    CardListFunk();
}

public void ShopButton()
{
    ShopPanel.SetActive(true);
    MainPanel.SetActive(false);
    panel = ShopPanel;
}

public void SettingsButton()
{
    SettingsPanel.SetActive(true);
    MainPanel.SetActive(false);
    panel = SettingsPanel;
}

public void BackButton()
{
    if (panel != ItemsPanel)
    {
        MainPanel.SetActive(true);
        panel.SetActive(false);
    }
    else
    {
        panel.SetActive(false);
        EquipmentPanel.SetActive(true);
        panel = EquipmentPanel;
    }
}

public void ItemsButton()
{
    ItemsPanel.SetActive(true);
    EquipmentPanel.SetActive(false);
    panel = ItemsPanel;
}
```

ДОДАТОК Г

Код скрипту відповідний за чанки

```
using System.Collections.Generic;
using UnityEngine;

public class Chunk : MonoBehaviour
{
    public Transform Begin;
    public Transform End;

    public List<Enemy1Script> EnemyList = new List<Enemy1Script>();

    public MovementStopTrigger mST;

    public Fight fight;

    void Start()
    {
        for (int i = 0; i < EnemyList.Count; i++)
        {
            EnemyList[i].gameLogic.resetButton.onClick.AddListener(EnemyList[i].EnemyFight);
        }

        void Update()
        {
        }
    }
}
```

ДОДАТОК Д

Код скрипту відповідний за генерацію чанків

```

using System.Collections.Generic;
using UnityEngine;

public class ChunksPlacer : MonoBehaviour
{
    public Transform Player;
    public Chunk[] ChunkPrefabs;
    public Chunk FirstChunk;
    private List<Chunk> spawnedChunks = new List<Chunk>();
    public GameLogic gameLogic;
    void Start()
    {
        spawnedChunks.Add(FirstChunk);
    }
    void Update()
    {
        if (Player.position.x < spawnedChunks[spawnedChunks.Count -
1].End.position.x + 7)
        {
            SpawnChunk();
        }
    }
    private void SpawnChunk()
    {
        Chunk newChunk = Instantiate(ChunkPrefabs[Random.Range(0,
ChunkPrefabs.Length)]);
        for (int i = 0; i < newChunk.EnemyList.Count; i++)
        {
            newChunk.EnemyList[i].gameLogic = gameLogic;
            newChunk.EnemyList[i].gameLogic.Enemys.Add(newChunk.EnemyList[i]);
            newChunk.EnemyList[i].fight = gameLogic.fight;
            newChunk.EnemyList[i].Cards = gameLogic.cardsSkript;
        }
        newChunk.mST.Cards = gameLogic.cards;
        newChunk.transform.position = spawnedChunks[spawnedChunks.Count -
1].End.position;
        spawnedChunks.Add(newChunk);
        for (int i = 0; i < spawnedChunks[spawnedChunks.Count - 1].EnemyList.Count;
i++)
        {
            gameLogic.resetButton.onClick.RemoveListener(spawnedChunks[spawnedChunks.Count -
1].EnemyList[i].EnemyFight);
        }
        if (spawnedChunks.Count >= 4)
        {
            Destroy(spawnedChunks[1].gameObject);
            spawnedChunks.RemoveAt(1);
        }
    }
}

```

ДОДАТОК Е

Код скрипту відповідний за переміщення персонажа

```
using UnityEngine;

public class PlayerMovemant : MonoBehaviour
{
    private float speed = 1.4f;
    public Rigidbody rb;
    void Start()
    {
        Move();
    }
    public void Move()
    {
        rb.velocity = new Vector3(-speed, 0, 0);
        //Debug.Log(-speed * Time.deltaTime);
        Debug.Log("Run!");
    }
}
```

ДОДАТОК Ж

Код інтерфейсу відповідний за ініціалізацію предмету

```
using UnityEngine;
using UnityEngine.UI;

public interface IItem
{
    string Name { get; }
    Sprite UIIcon { get; }
}
}
```


ДОДАТОК II

Код інтерфейсу відповідний за дії гравця

```

using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

public class Fight : MonoBehaviour
{
    [SerializeField]
    Rigidbody player;
    public GameObject ResetButton;
    public int TotalScore = 0;
    public int PlayerHealth = 100;
    public int PlayerEnergy;
    public int PlayerShield = 0;
    public int PlayerDamage = 30;
    // Effects //
    public int rampageBonus;
    public bool rampage = false;
    public bool healthRegeneration = false;
    //
    public PlayerMovemant playerMovemant;
    [SerializeField]
    Cards cards;
    [SerializeField]
    GameObject card;
    [SerializeField]
    Color colorEnableFalse;
    [SerializeField]
    GameLogic gameLogic;
    [SerializeField]
    GameObject GameCardLeft;
    [SerializeField]
    GameObject GameCardMiddle;
    [SerializeField]
    GameObject GameCardRight;
    public int countCard = 3;
    int DMG;
    public List<Texture> Cards = new List<Texture>();
    public Enemy1Script enemyTarget;
    public Animator anim;
    [SerializeField]
    Text playerHealthOnUI;
    [SerializeField]
    Text playerEnergyOnUI;
    [SerializeField]
    Text playerShieldOnUI;
    [SerializeField]
    Text playerScoreOnUI;
    public List<GameObject> Enemys = new List<GameObject>();
    public List<GameObject> projectors = new List<GameObject>();
    public List<int> Cardprices = new List<int>();
    public void Start()
    {
        anim = GetComponent<Animator>();
        anim.SetBool("IsRun", true);
    }
}

```

```

new Equipment();
playerShieldOnUI.text = "";
PlayerEnergy = 10;
Enemys.Clear();
playerMovemant.Move();
}
void Update()
{
    playerScoreOnUI.text = $"Total score: {TotalScore}";
    Debug.Log("Enemys: " + Enemys.Count);
    if (player.velocity == Vector3.zero)
    {
        anim.SetBool("IsRun", false);
    }
}
public void RemoveFromListEnemy()
{
    gameLogic.Enemys.Clear();
    projectors.Clear();
    Debug.Log("Removed! " + Enemys.Count);
    card.SetActive(false);
    anim.SetBool("IsRun", true);
    playerMovemant.Move();
}
public void PlayerStatus(){
    playerHealthOnUI.text = $"Health: {PlayerHealth}";
    playerEnergyOnUI.text = $"Energy: {PlayerEnergy}";
    if (PlayerShield > 0)
    {
        playerShieldOnUI.text = $"Shield: {PlayerShield}";
    }
    else
    {
        playerShieldOnUI.text = "";
    }
    gameLogic.Endgame();
}
public bool SetPlayerEnergy(int cost, Text textCard)
{
    if (PlayerEnergy < cost)
    {
        textCard.text = "You no have enough energy";
        return false;
    }
    else
    {
        textCard.text = "Used";
        return true;
    }
}
public void CardPrices(Image card, int i, int count) {
    int cost;
    if (cards.CardsListTexture.IndexOf(card.sprite) == count){
        cost = Cardprices.IndexOf(count);
        cards.Prices[i].GetComponent<Text>().text = $"{cost}";
    }
}
public void RemoveUsedCards(Sprite card) {
    cards.CardsListActive.Remove(card);
    Debug.Log("Card Remove");
    PlayerStatus();
}
}

```