

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему: «Ігровий квест-додаток з підтримкою мережевої взаємодії ігроків»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології
проектування»

Виконавець роботи: студентка групи ІТ.м-01 Каравай Ірина Станіславівна

**Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою**

_____ «__» ____ 2021 р.

Науковий керівник

_____ к.т.н., доц., Баранова І.В.
(підпис) (науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

_____ (підпис) (науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Суми-2021

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. кафедри ІТ

_____ В. В. Шендрик
«__» _____ 2021 р.

ЗАВДАННЯ

на кваліфікаційну роботу магістра студентів

Каравай Ірина Станіславівна

1 Тема роботи Ігровий квест-додаток з підтримкою мережевої взаємодії ігроків.

керівник роботи Баранова Ірина Володимірівна, к.т.н., доцент _____,

затверджені наказом по університету від «29» жовтня 2021 р. № 0787-IV

2 Термін здачі студентом закінченого проекту «13» грудня 2021 р.

3 Вхідні дані до роботи завдання на розробку ігрового квест-додатку з підтримкою мережевої взаємодії гравців

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз предметної області, постановка задачі та методи дослідження, проектування додатку, практична реалізація

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) актуальність роботи, мета та задачі, дослідження аналогів, засоби реалізації, структурно-функціональне моделювання (IDF0), діаграма варіантів використання (Use case), етапи реалізації, практична реалізація.

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

Дата видачі завдання _____.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Опис концепції та можливостей ігрового квест-додатку	01.09.21 – 21.09.21	
2	Встановлення необхідного ПЗ, перевірка ресурсомісткості робочої станції	22.09.21– 24.09.21	
3	Пошук історичних фактів	25.09.21– 15.10.21	
4	Створення макету основної логіки, розробка загадок	30.09.21– 28.10.21	
5	Розробка бібліотеки елементів 3d моделей	30.09.21 – 28.10.21	
6	Розробка основних мап додатку	20.10.21– 28.10.21	
7	Розробка меню	01.10.21– 28.10.21	
8	Розробка головного функціоналу	29.10.21– 25.11.21	
9	Оформлення документації	27.11.21– 10.12.21	

Студент _____
(підпис)

Каравай І.С.

Керівник роботи _____
(підпис)

к.т.н., доц. Баранова І.В.

РЕФЕРАТ

Тема роботи «Ігровий квест-додаток з підтримкою мережевої взаємодії ігроків».

Пояснювальна записка складається зі вступу, чотирьох основних розділів та декількох підрозділів, висновку, а також списку використаних джерел із 20 найменувань та двох додатків. Загальний обсяг пояснювальної записки складає 76 сторінки, в тому числі 64 сторінок основного тексту, 4 сторінки списку використаних джерел, 8 сторінок додатків.

Кваліфікаційну роботу магістра присвячено розробці ігрового квест-додатку з підтримкою мережевої взаємодії гравців, котрий надає можливість не лише грати в одиночному режимі, але й грати з друзями за допомогою мережевого підключення.

В роботі проведено аналіз вже існуючих подібних додатків. Окрім аналізу аналогів, проведено повне планування робіт та проектування необхідних діаграм. Виконано повний опис процесу створення ігрового квест-додатку, присутні зображення результату виконання етапів роботи.

Результатом виконаної роботи є ігровий квест-додаток з можливістю підключення гравців один до одного через мережу.

Практичне значення роботи полягає в створенні не лише цікавої та красивої гри, але й донести цікаві історичні факти та історії.

Ключові слова: КВЕСТ ДОДАТОК, МЕРЕЖЕВЕ ПІДКЛЮЧЕННЯ, UNREAL ENGINE, BLUPRINT, МОДЕЛЮВАННЯ, 3DS MAX, FUSE.

ЗМІСТ

Вступ.....	6
1 Аналіз предметної області.....	8
1.1 Огляд останніх досліджень і публікацій	8
1.2 Аналіз програмних продуктів – аналогів	10
2 Постановка задачі та методи дослідження	13
2.1 Постановка задачі	13
2.2 Методи дослідження.....	14
3 Проектування ігрового квест додатку з можливістю інтернет-взаємодії ...	18
3.1 Структурно-функціональне моделювання квест додатку	18
3.2 Моделювання варіантів використання	20
3.3 Планування робіт	21
4 Практична реалізація додатку.....	22
4.1 Проектування основних моделей, текстур та матеріалів.....	22
4.2 Створення основних мап гри.....	24
4.3 Розробка головного меню	28
4.3.1 Налаштування графіки	28
4.3.2 Налаштування збереження.....	33
4.4 Розробка завдань квесту	37
4.4.1 Розробка сувоїв	37
4.4.2 Розробка інвенторі меню.....	39
4.4.3 Розробка прихованих дверей	45
4.4.4 Розробка кодового замку.....	47
4.5 Розробка мережевої взаємодії.....	51
Висновки	63
Список використаних джерел	64
Додаток А.....	68

ВСТУП

Сучасне суспільство вже звикло до того, що без комп'ютерних ігор та інтернет-спілкування неможливо уявити теперішнє життя. Сфера комп'ютерних ігор завжди була найбільш швидкою у плані розвитку та популяризації, за рахунок чого нею зацікавилось багато людей.

При огляді ринку сучасних ігор було виявлено, що максимум 10% з них є корисними для розвитку, а всі інші зазвичай спонукають до насилля та жорстокості. У поколінні, котре зараз зростає саме на таких жорстоких іграх, немає майбутнього, адже про який розвиток може йти мова, коли замість того, аби навчитися чомусь корисному та дізнатися щось дійсно цікаве, вони просто дивляться на весь той бруд і насилля, що закладено розробниками у грі. Окрім того, в карантинних умовах, діти позбавленні повноцінного спілкування з однолітками, що є одним з важливих етапів розвитку психіки та соціалізації.

Об'єктом досліджень є застосування методів гейміфікації при розробці ігрових додатків.

Предмет досліджень – технології мережевої взаємодії гравців в ігровому додатку.

Саме тому було вирішено, що основною **метою** даної роботи є створення ігрового додатку, який буде не лише привабливим та цікавим, але й мати у собі освітньо-пізнавальну складову. Він містить історичні факти, що залучать дітей до зацікавленням історією, а відтак буде корисним для їх розвитку, а також надасть можливість спілкуватися з однолітками онлайн.

Нижче наведено список задач, виконання яких вирішено прийняти за основу проекту:

- планування поетапного виконання проекту;
- розробка контекстної діаграми та діаграм декомпозиції;
- розробка діаграми сценаріїв використання;

- планування структури проекту (створення WBS/OBS);
- побудова діаграми Ганта;
- планування ризиків проекту.

Розроблений ігровий додаток буде приносити не лише естетичне задоволення, але й надаватиме можливість розширити свій кругозір та дізнатися щось нове.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Існує велика кількість досліджень, що аналізують сферу комп'ютерних ігор, їх вплив на суспільство та інші фактори. Зазвичай такі дослідження у відсотковому відношенні показують рівень швидкості росту та популяризації даної сфери.

Якщо звернутися до конкретних публікацій, то побачимо, що комп'ютерна індустрія впливає також і на економіку та навіть політику. Седих І.А. у своїй публікації [1] зауважив: «На думку експертів, саме ігрова індустрія стає провідником ідей сучасної творчості та новаторства. Комп'ютерні ігри зараз присутні в різних сегментах масової культури, а деякі навіть стали її феноменами. У деяких країнах, наприклад в США, комп'ютерні ігри були визнані окремим видом мистецтва.» і це дійсно так.

Неможливо не погодитися з тим, що без комп'ютерних ігор наше життя вже неможливо навіть уявити. Ще одним доказом зростання «ігроманії» є глобальний графік від Newzoo, наведений на рисунку 1.1, що наочно показує ці зміни у відсотковому відношенні з 2012 року по 2021 рік [2].

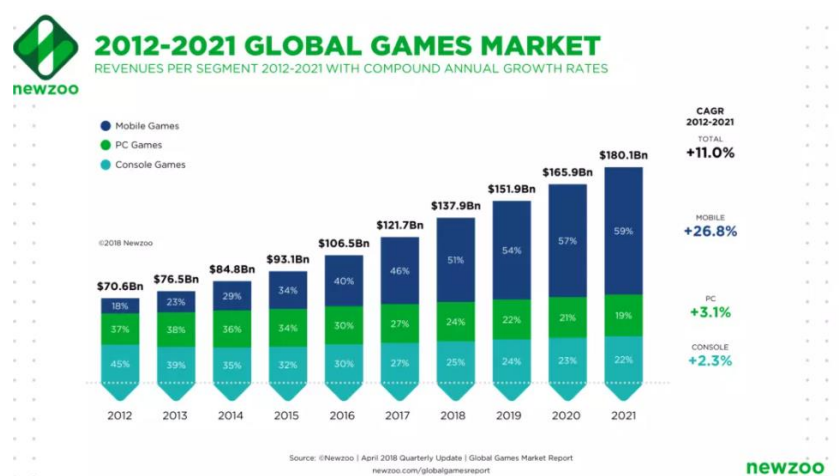


Рисунок 1.1 - Глобальний графік від Newzoo

На прикладі публікації Д.В. Галкіна [3] можна прослідити етико-філософські та етико-політичні проблеми, котрі з'являються з розвитком комп'ютерних ігор. Він зауважив, що дана сфера є деяким феноменом для сучасної культури. Доволі детально в роботі відображена та пояснена нарративна природа у відеоігор. Саме його робота спонукала розробити додаток, котрий не буде порушувати морально-етичні норми суспільства.

Якщо розглянути можливості ігрових додатків сприяти розвитку особистості, то слід звернутися до роботи Трішкіної М.С. [4]. Саме вона підкреслила те, що відеоігри – це новий інструмент, за допомогою якого, при вмілому користуванні, можна радикально змінити вплив звичайних розважальних додатків та націлити його на культурний розвиток особистості.

Також доволі велика кількість досліджень і публікації спрямована на аналіз головної причини популярності різноманітних ігор у суспільстві. У роботі Ілларіанова Г.А. та Моїсеєнка М.К. [5] проводиться достатньо конструктивний аналіз багатьох факторів впливу. Також можна побачити розгорнуту думку про корисність ігор для відпочинку, а з іншого боку розглянуто деструктивний потенціал відеоігор, що руйнує собою соціальне життя людини.

Доволі важливим фактором в сучасних додатках стала можливість мережевої взаємодії. Адже нікому вже не цікаві звичайні додатки, котрі просто встановлюються на персональний комп'ютер та надають гравцю можливість звичайної взаємодії з написаною ігровою логікою. Сьогодні користувачам хочеться мати більше взаємодії з навколишнім світом через інтернет. Як зауважив Степаненко О.О. у своїй роботі «Технології розробки мережевих додатків» [6], можливість бути пов'язаним з національною комунікаційною мережею – це один із головніших етапів розвитку людства та можна, навіть, сказати, що це дійсно прорив у деякій мірі.

Проаналізувавши ще декілька джерел інформації, наприклад статтю «Game Developers Association» [7] та роботу Седюх І.А «Computer game

industry-2020» [8], було зауважено сильні сторони та основні недоліки, які можуть бути виявлені у аналогічних розробках.

Також до уваги було взято всі аспекти життєвого циклу, щоб максимально адаптувати комп'ютерну гру під сучасне бачення та створити можливість перепрограмування сприйняття суспільством відеоігор, аби вони шукали не насилля та жорстокість, а можливість розвитку та спілкування.

1.2 Аналіз програмних продуктів – аналогів

Мережеві комп'ютерні ігри – це звісно не новинка та не відкриття. Існують сотні й навіть тисячі різноманітних додатків, квест-ігор і так далі. Є ті що підключаються за допомогою LAN-підключення, а є звичайні мережеві.

Основною проблемою уже існуючих ігрових квест-додатків стало те, що вони підтримують лише один тип мережевого зв'язку, тобто або тільки через LAN, або через Internet. Так прикладом можна навести яскраву гру, котра буде до вподоби як дітям, так і дорослим – «it takes two» [9]. Вона надає можливість взаємодії двох гравців, але лише через Internet (рис. 1.2).



Рисунок 1.2– Мережевий додаток «it takes two»

Стосовно кількості учасників також є певні проблеми, адже не всі розробники передбачили можливість вибору. Існує велика кількість квестів, котрі можна проходити лише командами і, на жаль, права вибору кількості

учасників користувачу не нададуть. Таким прикладом можна навести доволі відому у світі геймерів гру «RAID: Shadow Legends» [10]. В цій грі користувач не може обрати ні команду, в якій буде грати, ні кількість учасників (рис. 1.3).



Рисунок 1.3 – Мережевий додаток «RAID: Shadow Legends»

Доволі цікавий і важливий момент було підмічено під час перегляду гри «Call of Duty: Warzone» [11] – відносно мало ігор, котрі надають можливість користувачу змінювати графічні характеристики, адже не всі персональні комп'ютери та відеокарти можуть працювати з таким навантаженням (рис. 1.4).

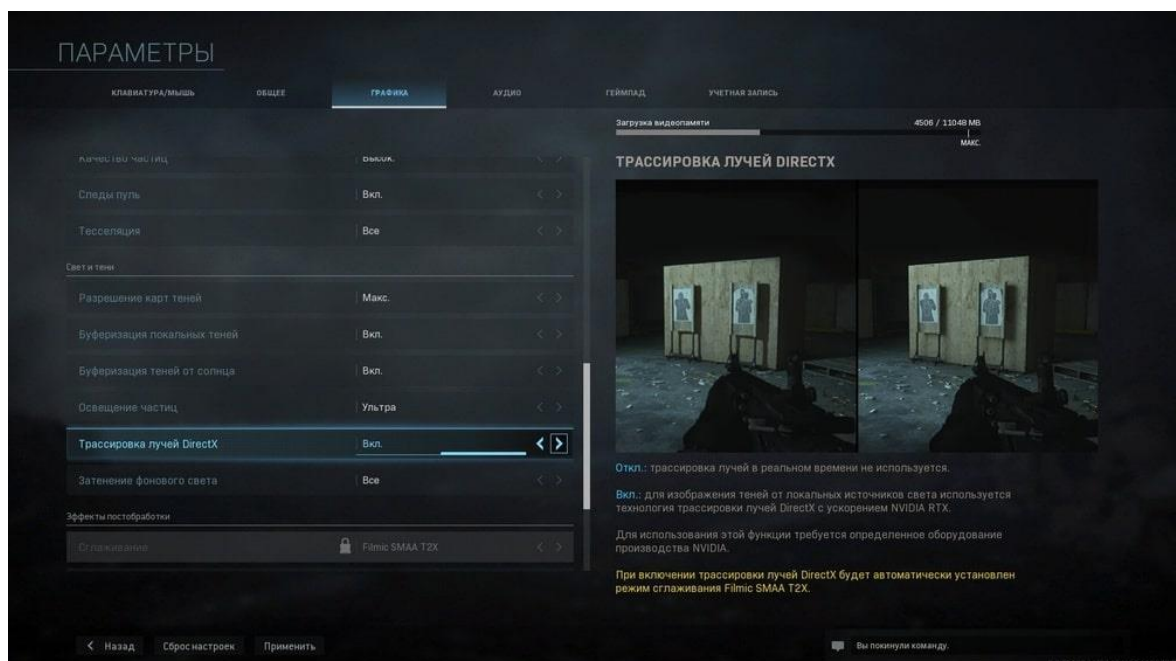


Рисунок 1.4 – Можливість налаштування графіки

Після проведеного аналізу сформульовано основні недоліки та позитивні сторони додатків-аналогів. Результати аналізу наведено в таблиці 1.1.

Таблиця 1.1 – Результати аналізу аналогічних проєктів

Недоліки	Переваги
Високі вимоги до відео-карти	Якість зображення
Зазвичай не підтримують два види мережевого зв'язку	Можливість налаштування графіки
Не надають можливості вибору команди	Швидкодія мережевого зв'язку
Корисна тематика гри	Вибір кількості персонажів в грі
	Вибір режиму гри (одиначний або ж командний)

Аналіз переваг та недоліків аналогічних проєктів допоможе створити максимально зручний и адаптивний додаток для користувачів.

2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

2.1 Постановка задачі

Отже, в результаті аналізу проведених досліджень маємо сформовану мету проекту – розробити ігровий квест-додаток з підтримкою мережевої взаємодії гравців.

Основна ціль проекту – створення квесту, який буде не лише цікавий, але й корисний, матиме підтримку онлайн-з'єднань з іншими гравцями, дозволить налаштовувати якість графіки під власні побажання та дізнатися цікаву інформацію.

Аби поставлену задачу можна було реалізувати, було прийнято рішення розбити її на певні завдання:

- провести аналіз предметної області та схожих додатків, визначити їх переваги й недоліки;
- розбити розробку проекту на етапи, а також обрати інструменти реалізації завдання;
- провести планування робіт з реалізації проекту, провести структурно-функціональне моделювання роботи додатку, визначити ризики проекту;
- розробити основні 3d моделі об'єктів ігрових локацій, призначити необхідні матеріали й текстури, налаштувати реалістичне освітлення;
- розробити структуру додатку, сценарій проходження квесту, логіки ігри;
- розробити інформативну частину додатку, підібрати цікаві історичні факти й відомості для заохочення гравців;
- реалізувати можливість мережевої підтримки гравців;

- розробити необхідний функціонал додатку.

Даний квест-додаток матиме наступні можливості:

- переміщення гравців по мапах;
- вибір режиму гри;
- можливість налаштування графіки;
- можливість вибору ігрової кімнати;
- можливість вибору рівня;
- збереження гри на різних етапах;
- можливість збереження сувоїв з інформацією для повторного використання.

2.2 Методи дослідження

Для різних етапів розробки проєкту було використано різні додатки. Спочатку розглянемо весь список інструментів (рис. 2.1), що було використано, а саме:

- Autodesk 3ds Max для проектування основних 3d моделей;
- рушій Unreal Engine для створення функціоналу;
- Adobe Photoshop для налаштування текстур та матеріалів;
- Fuse для роботи з акторами;
- Steam для перевірки можливості інтернет-з'єднання.



Рисунок 2.1 – Основні інструменти розробки додатку

Для 3d моделювання було вирішено застосовувати Autodesk 3ds Max [12] (рис. 2.2), адже він добре взаємодіє з рушієм Unreal Engine, надає можливості зберігати об'єкти в зручному форматі, налаштовувати його форми та розміри за допомогою відомих інструментів [13].

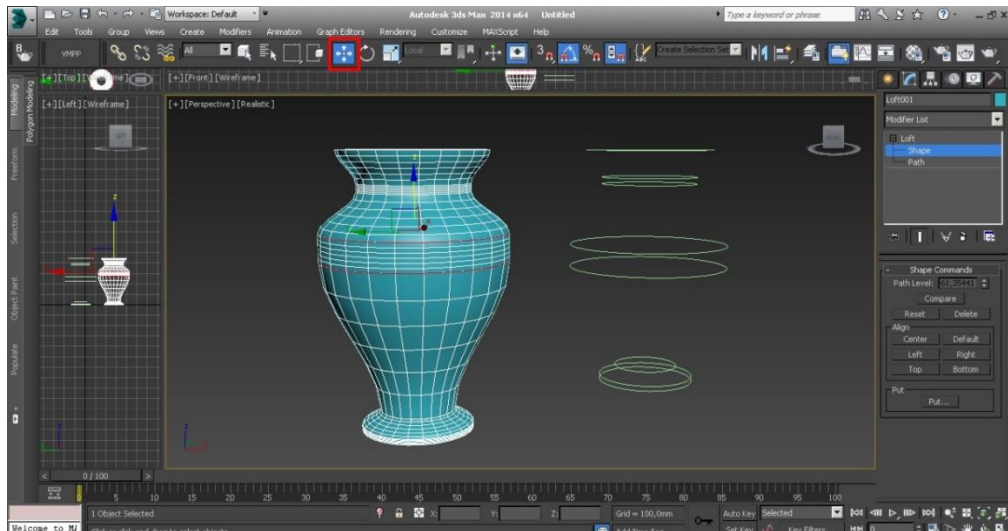


Рисунок 2.2 – Autodesk 3ds Max

Аби створити необхідні безшовні текстури, було використано всім відомий зручний Adobe Photoshop (рис.2.3), що надає можливість оброблювати зображення так, як це зручно користувачу [14].

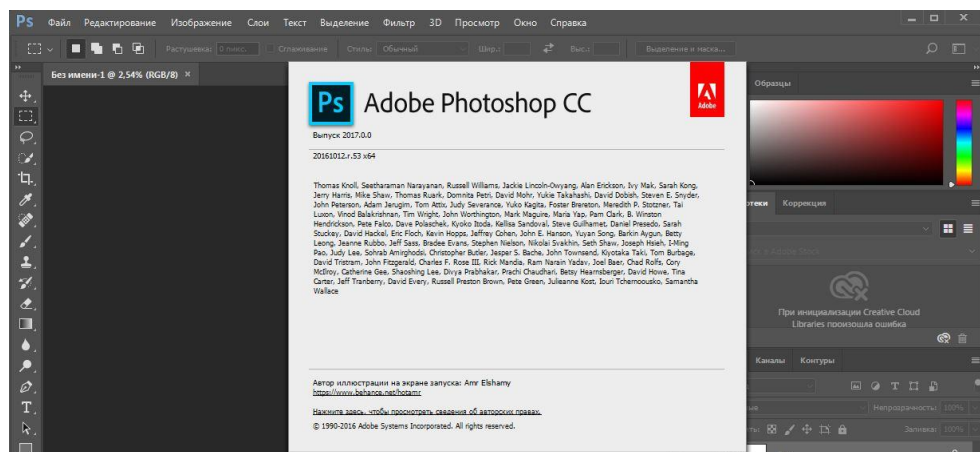


Рисунок 2.3 – Adobe Photoshop

Можливість створення самої гри та її основний функціонал надає

рушій Unreal Engine (рис.2.4) [15]. Насамперед даний рушій було обрано, адже він має велику безкоштовну бібліотеку, котру можна використовувати у власних цілях [16]. Також для Unreal Engine написана велика кількість підручників та статей, що значно полегшує пошук необхідної інформації [17].

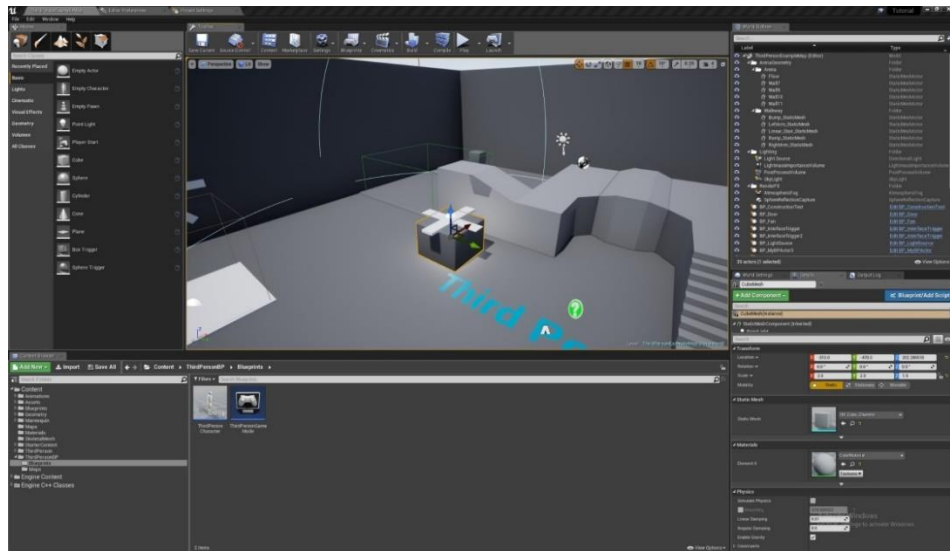


Рисунок 2.4 – Рушій Unreal Engine

Для створення додаткових акторів на другому рівні гри було використано додаток Fuse (рис. 2.5), котрий знаходиться у вільному доступі на мережевій платформі Steam [18].

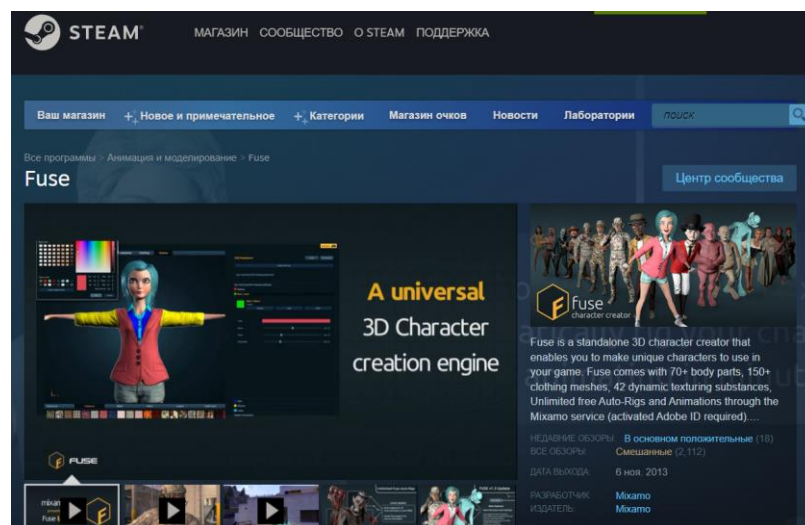


Рисунок 2.5 – Додаток Fuse

За допомогою обраних інструментів реалізовано функціональні вимоги проєкту: створені всі необхідні текстури, бібліотека 3d об'єктів, додаткові персонажі, а також сам функціонал ігрового квест-додатку.

3 ПРОЕКТУВАННЯ ІГРОВОГО КВЕСТ ДОДАТКУ З МОЖЛИВІСТЮ ІНТЕРНЕТ-ВЗАЄМОДІЇ

3.1 Структурно-функціональне моделювання квест додатку

Основні процеси в проекті можна описати за допомогою методології функціонального моделювання, або якщо скорочено методологією IDEF0 [19-21]. Дана методологія використовується, аби схематично представити всі інструменти, а також ресурси, що використовуються у проекті. Окрім того на IDEF0 діаграмі також відображаються і бізнес-процеси [22].

Контекстна діаграма ігрового квест-додатку та його декомпозиція, що створена у вигляді блоків та описує роботу створеного додатку, зображено на рисунках 3.1-3.2.

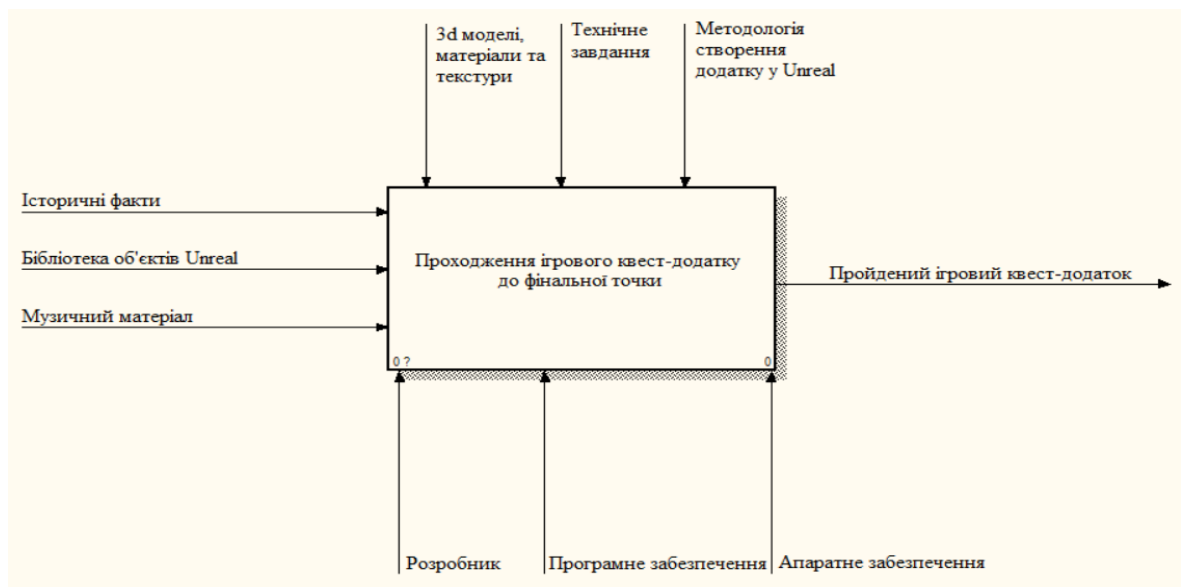


Рисунок 3.1 – Контекстна діаграма

Завдяки декомпозиції контекстної діаграми добре видно основні етапи проходження ігрового-квест додатку. Окрім основних блоків наведено

інформаційні та матеріальні ресурси, вони позначені за допомогою стрілок [21].

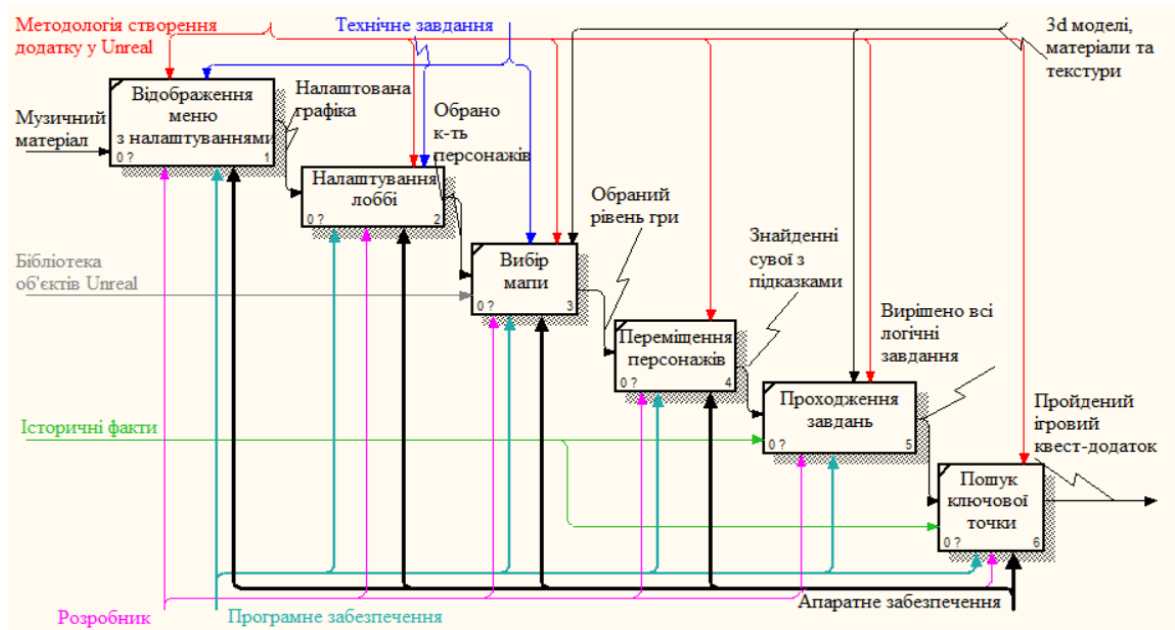


Рисунок 3.2 – Діаграма декомпозиції

Наступним кроком є розробка декомпозиції головних блоків створеної діаграми. На рисунку 3.3 наведено приклад декомпозиції одного з блоків – «Налаштування лоббі».

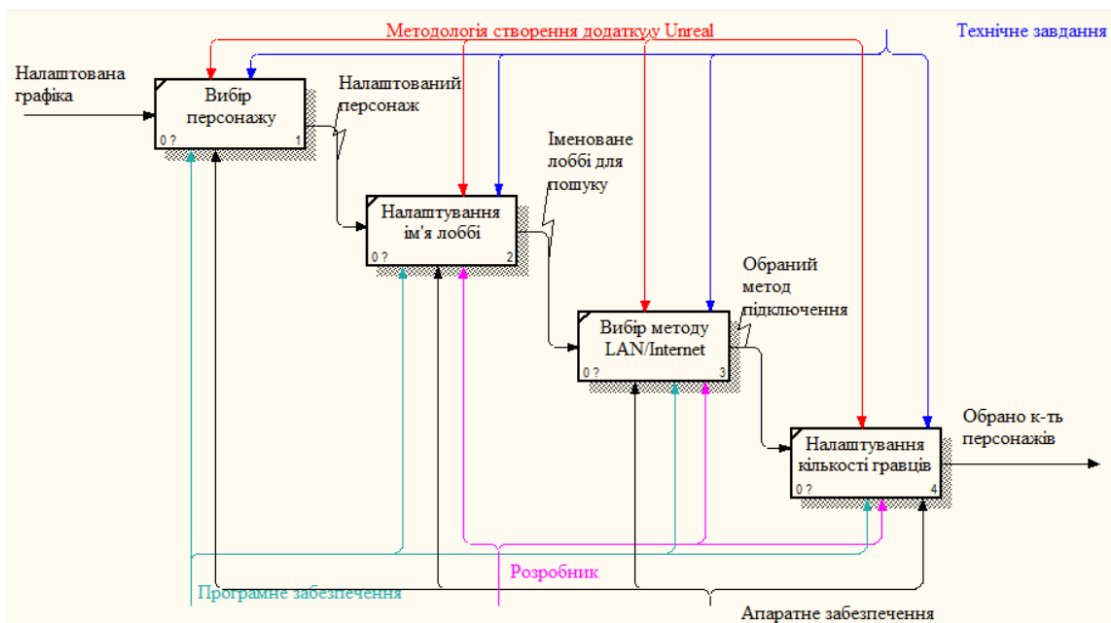


Рисунок 3.3 – Декомпозиції блоку «Налаштування лоббі»

3.2 Моделювання варіантів використання

Останнім кроком проектування додатку є побудова діаграми класів. Вона необхідна для того аби описати поведінку системи, відображення акторів та їх діяльності [22]. Дана Use Case diagram зображена на рисунку 3.4.

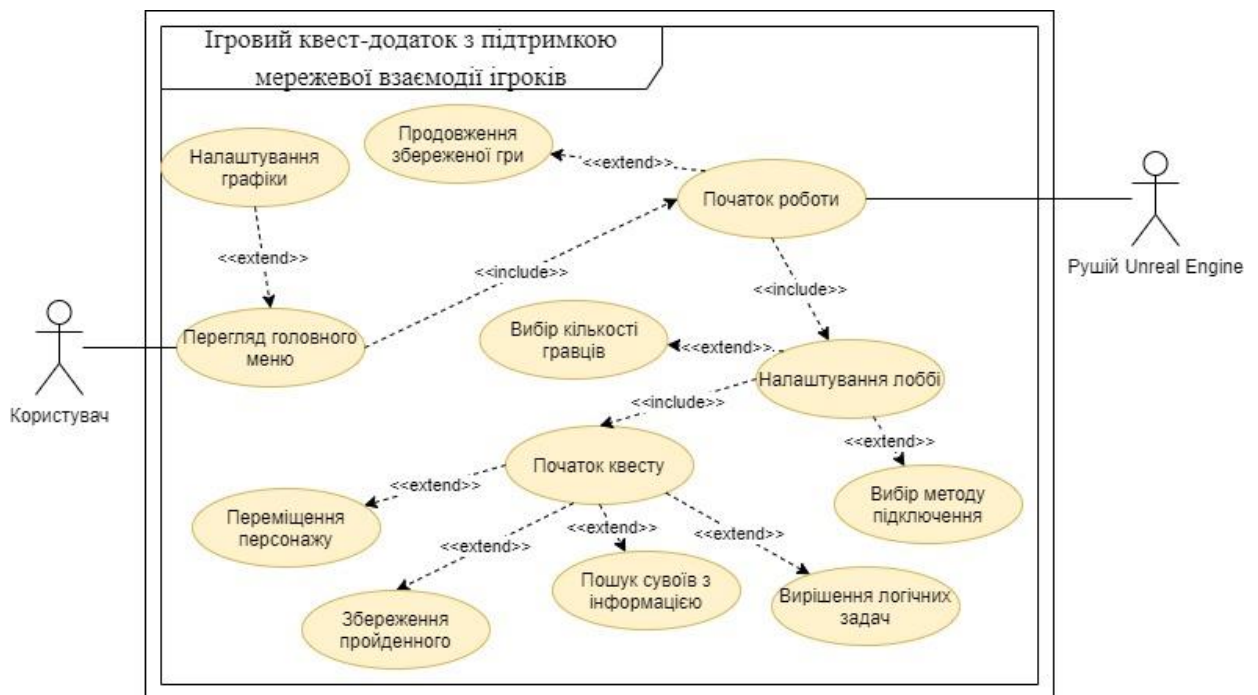


Рисунок 3.4 – Діаграма сценаріїв використання

В наведеній системі будуть діяти декілька або один актор – користувач або ж користувачі – залежить від вибору гри (одиначне проходження чи в команді). Для початку користувачеві доступне меню, де він може налаштувати графіку під можливості своєї відеокарти, створити лоббі, розпочати одиночну гру або продовжити гру з моменту попереднього збереження.

Після старту додатку користувач матиме можливість переміщатися по вибраній ним карті (рівню гри) та шукати сувої з історичними фактами, котрі є підказками.

Кожний сувій приховуватиме в собі одну цифру, необхідну для відкриття головного кодового замку. Також користувач може зберегти свої досягнення, переглянути інвентар (тобто всі знайдені сувої) та звісно покинути гру в будь-який момент часу.

3.3 Планування робіт

Для коректного планування проєкту було проведено деталізацію мети методом SMART, створені WBS та OBS структури для відображення головних ролей на проєкті та задач, за котрі вони відповідають.

Окрім того побудована діаграма Ганта для візуалізації календарного плану роботи. Для того аби розрахувати всі наявні в проєкті ризики, було окремо створено таблицю ризиків для зручного сприйняття, а також побудовано матрицю ризиків. Повний бюджет розраховано в самому кінці виконання роботи.

Вся описана інформація наведена в додатку А для зручного пошуку та структуризації пояснювальної записки.

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ДОДАТКУ

В роботі над проектом можна виділити такі основні етапи:

- проектування основних моделей, текстур та матеріалів;
- створення основних мап гри;
- розробка головного меню;
- розробка завдань гри;
- розробка мережевої взаємодії.

Перший етап є найпростішим, адже там ми розробляємо основні 3d моделі для гри, матеріали з текстурами для даних моделей і допоміжні елементи, наприклад, такі як сувої.

Другий етап є більш кропітким, адже треба створити мапи на котрих гравці будуть грати. В даній роботі наявно 3 мапи – це перший рівень (підземелля замку), другий рівень (у дворі замку) та рівень лоббі для красивої картинки завантаження гри.

Третій етап відповідає за створення меню, по більшій мірі – це створення можливостей налаштування графіки, а також початок гри.

На четвертому етапі було розроблено інвенторі-меню, сувої із завданнями, кодовий замок та приховані двері за шафою.

Ну і найскладніша частина проекту – це створення можливості інтернет взаємодії гравців.

4.1 Проектування основних моделей, текстур та матеріалів

Для початку за допомогою Adobe Photoshop було створено безшовні текстури [23] для подальшого накладання їх на майбутні 3d-моделі (рис. 4.1).

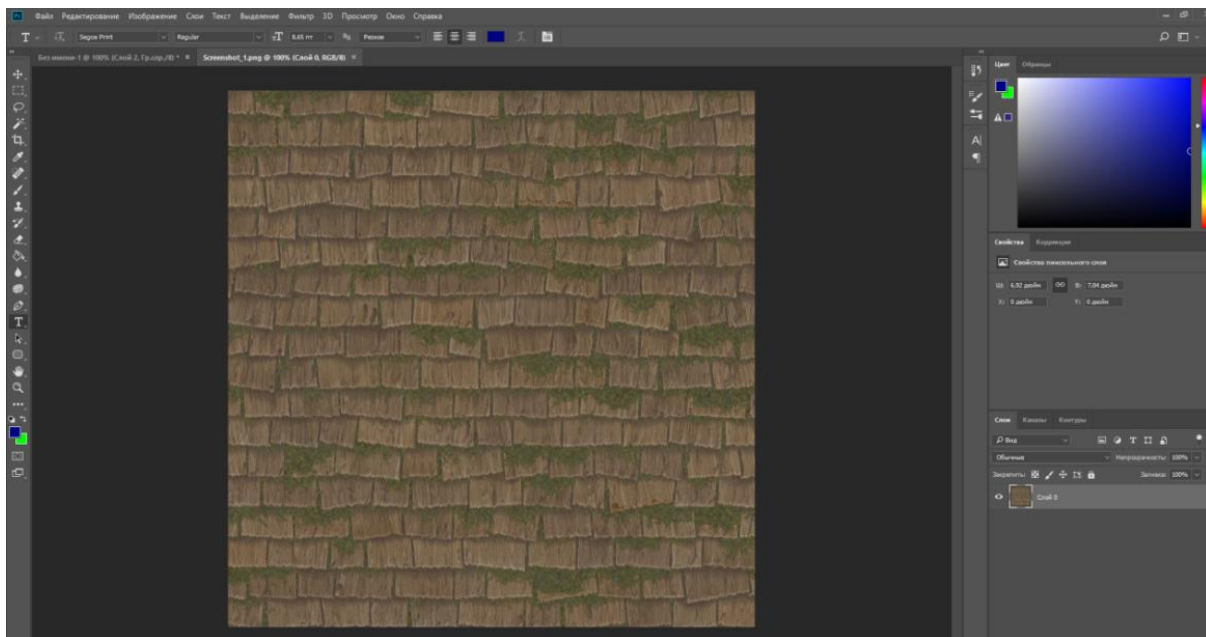


Рисунок 4.1 – Створення текстур

Наступним кроком, було створення необхідних 3d моделей для мап, аби вони були наповнені об'єктами та створювали певний історичний антураж. Так, наприклад, на рисунку 4.2 наведено проектування моделі шафи, яка буде використовуватися як потаємні двері в грі.

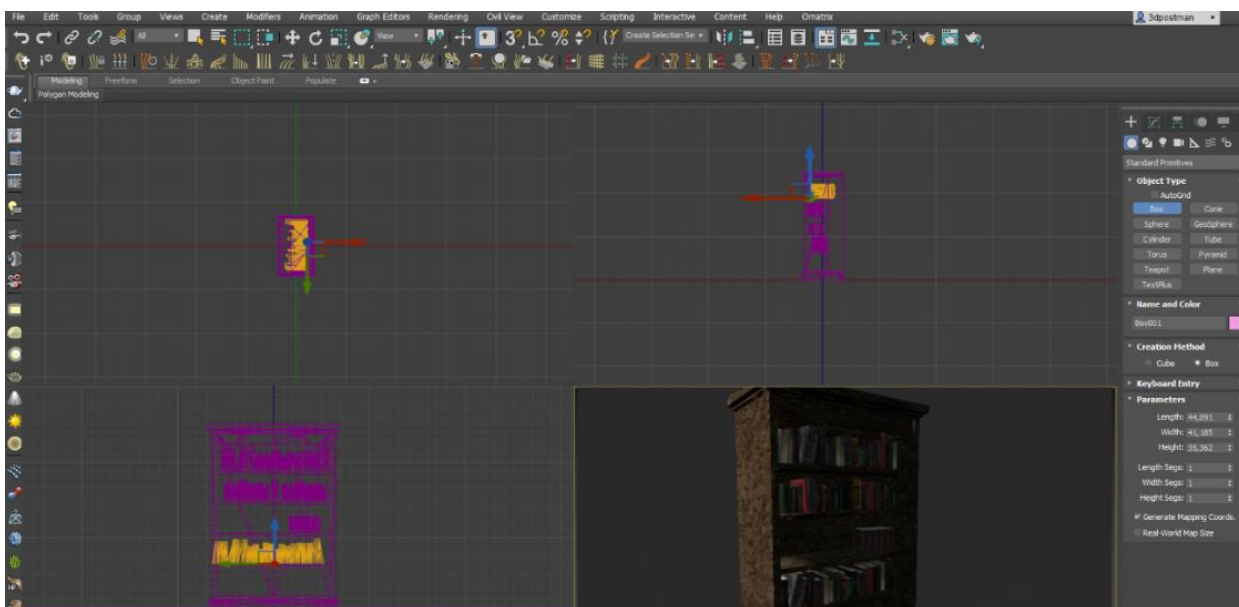


Рисунок 4.2 – Проектування шафи

На другому рівні гри в додатку задіяні додаткові актори, котрі надають певні підказки гравцям для пошуку сувоїв. Для створення даних акторів було використано безкоштовний додаток Fuse [24], на рисунку 4.3 наведено приклад роботи з даним додатком, створено стражі.



Рисунок 4.3 – Робота над створенням додаткових акторів

4.2 Створення основних мап гри

Звісно одним із найважливіших етапів роботи є створення мап. Для цього було розроблено три різних рівня, один із них є транзитним: це той, що з'являється, поки гравець очікує інших учасників під час створення лоббі (рис. 4.4).



Рисунко 4.4 – Транзитний рівень

Для першого рівня було розроблено підземний рівень, від знизу першого рівня наведено на рисунку 4.5. Вид зсередини наведено на рисунку 4.6, на першому рівні наведено два кодових замків, а також шафа, що є потаємними дверями.

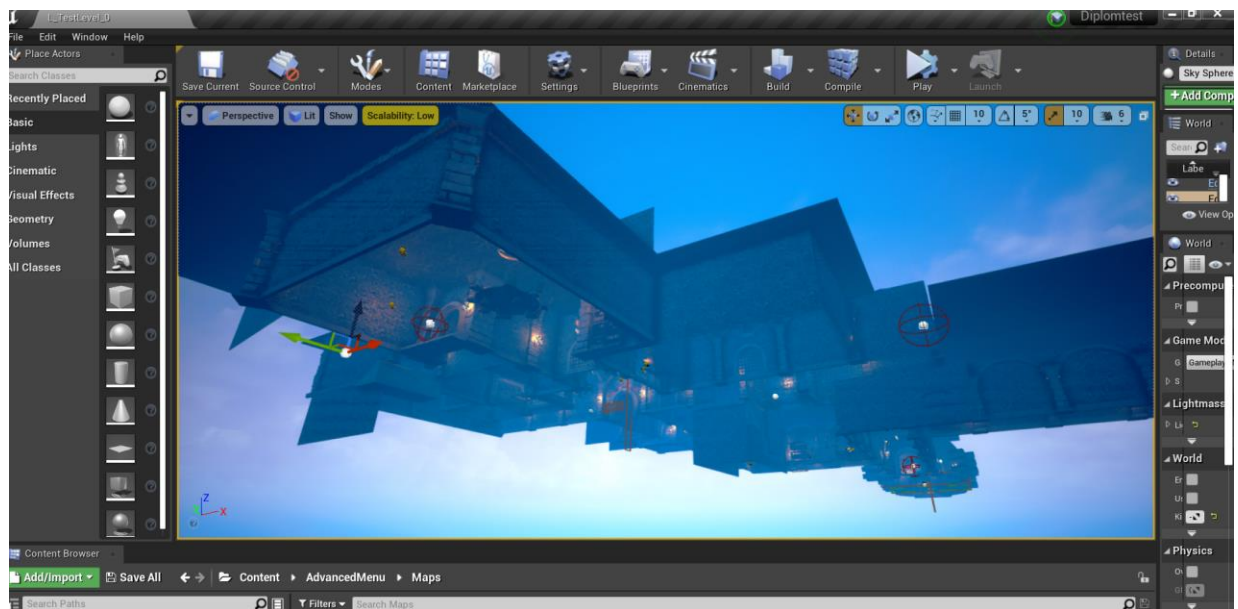


Рисунок 4.5 – Вид мапи знизу

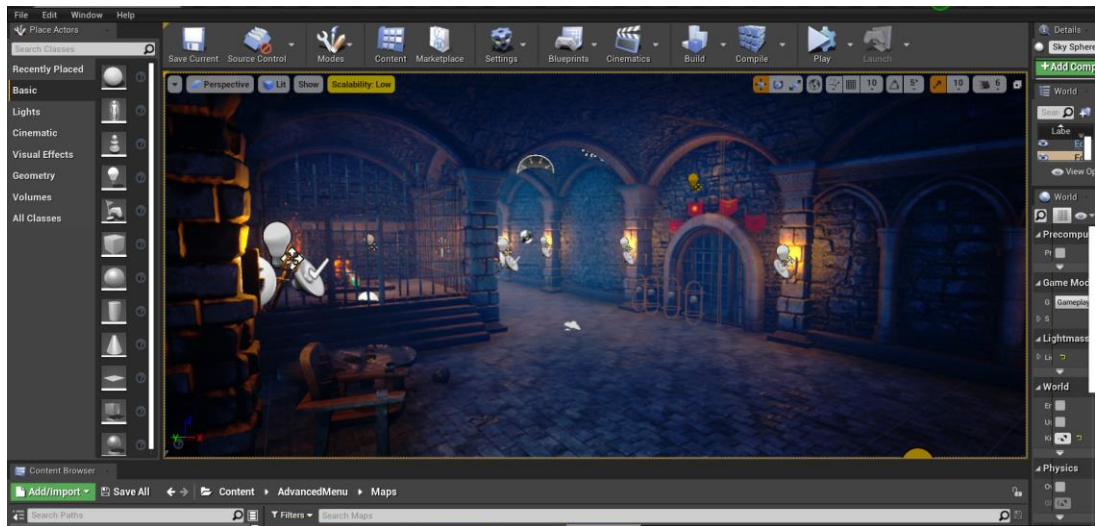


Рисунок 4.6 – Вид мапи першого рівня зсередини

Другий рівень переносить гравців на другу мапу, а саме на поверхню замку (рис. 4.7).

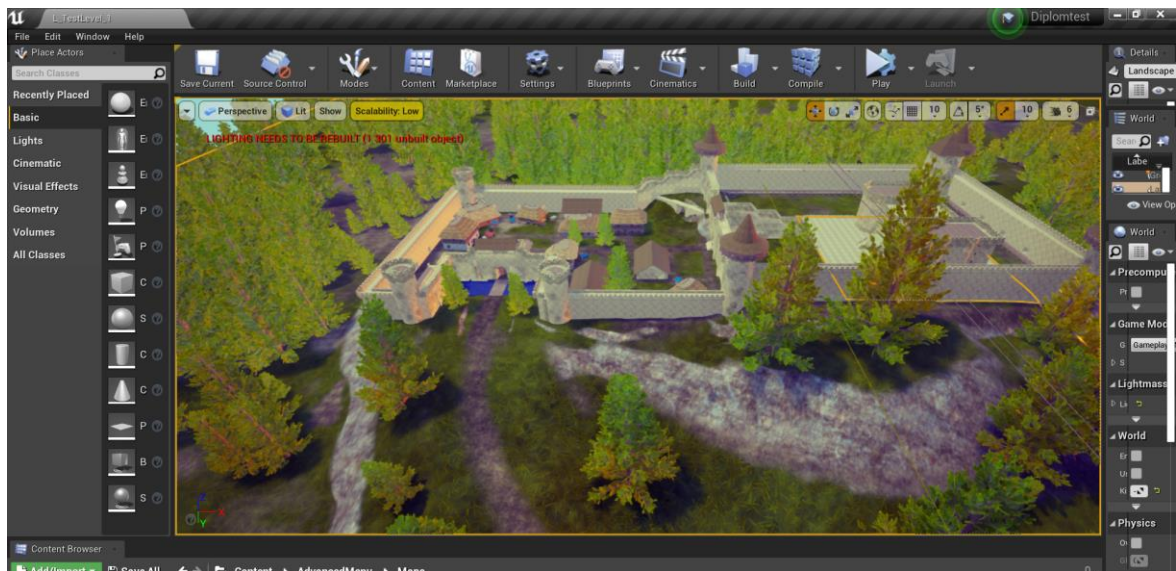


Рисунок 4.7 – Вид мапи зверху

Дана мапа продовжує логіку гри, але просто представляє собою інший вид замку (рис. 4.8 – рис. 4.9), де наявні додаткові актори для наповнення мапи, а також є сувої з завданнями та кодовий замок.

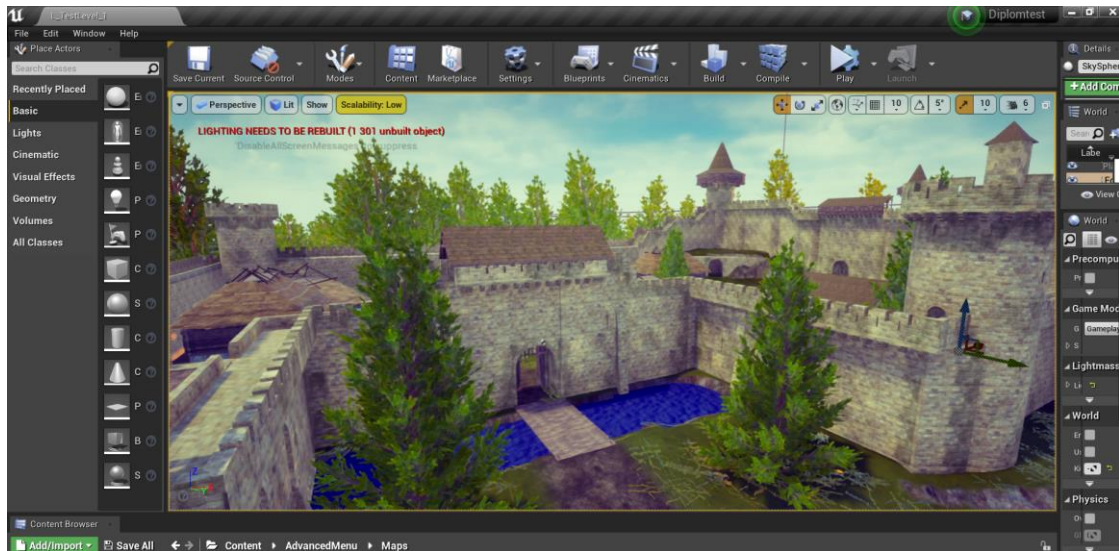


Рисунок 4.8 – Один вид замку зблизу



Рисунок 4.9 – Другий вид двору зсередини

Сувої із завданнями є головними допоміжними елементами гри. На них наведена певна історична інформація, що приховує в собі дату. Остання цифра дати є однією частиною коду до замку. На рисунку 4.10 наведено один із сувоїв, як приклад.

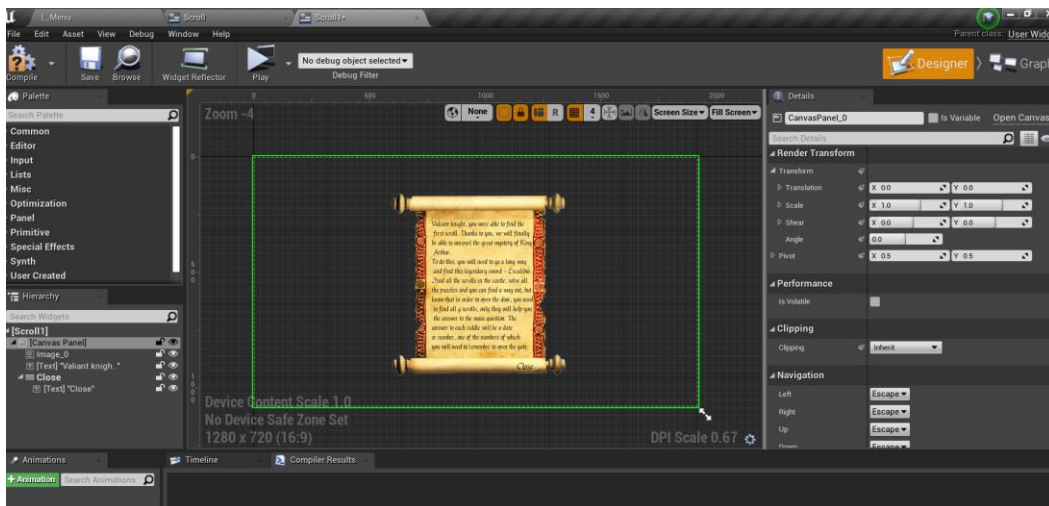


Рисунок 4.10 – Приклад сувою

4.3 Розробка головного меню

4.3.1 Налаштування графіки

Для реалізації можливості налаштування графіки необхідно створити віджет, за допомогою якого будемо це робити. Налаштування віджету зображено на рисунку 4.11.

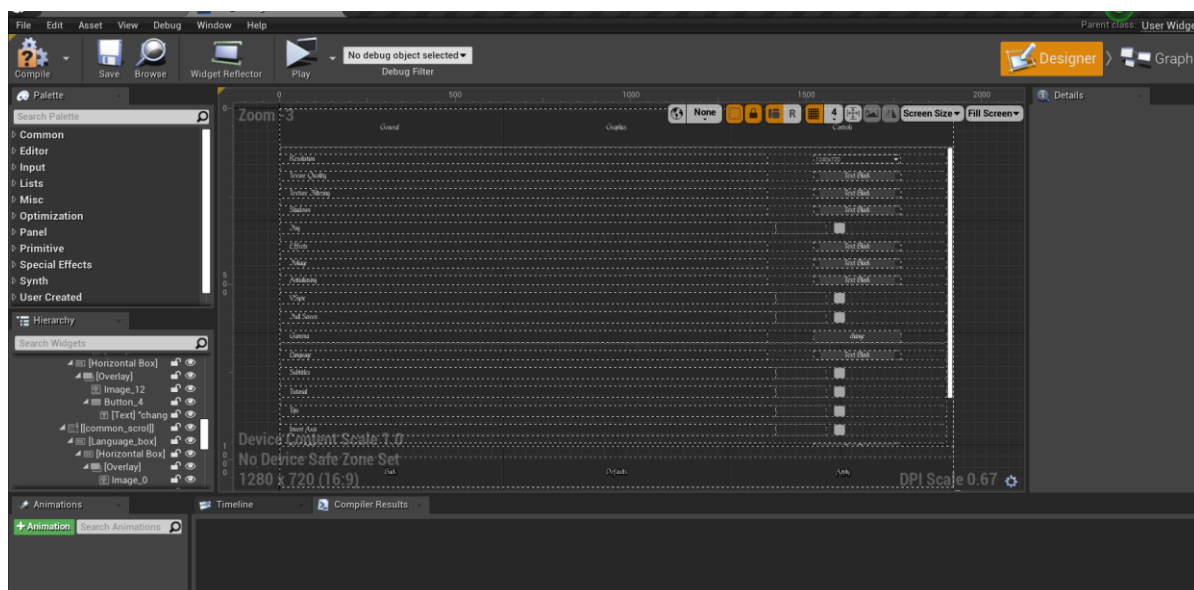


Рисунок 4.11 – Створення віджету налаштувань

Щоб віджет з'явився після старту гри, необхідно створити відповідний блупрінт (рис. 4.12).

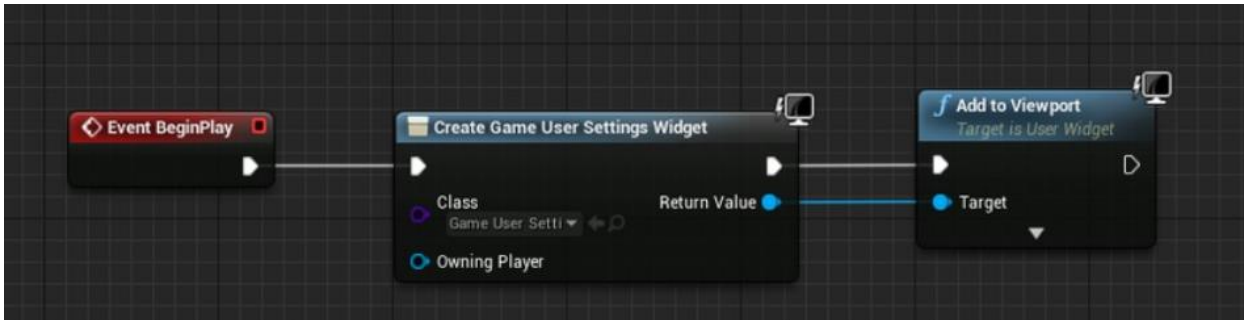


Рисунок 4.12 – Додавання віджету до гри

Користувач має змогу зберегти існуючі налаштування гри в пам'ять ПК у файлі, для цього використовуємо відповідний Blueprint, зображений на рисунку 4.13.

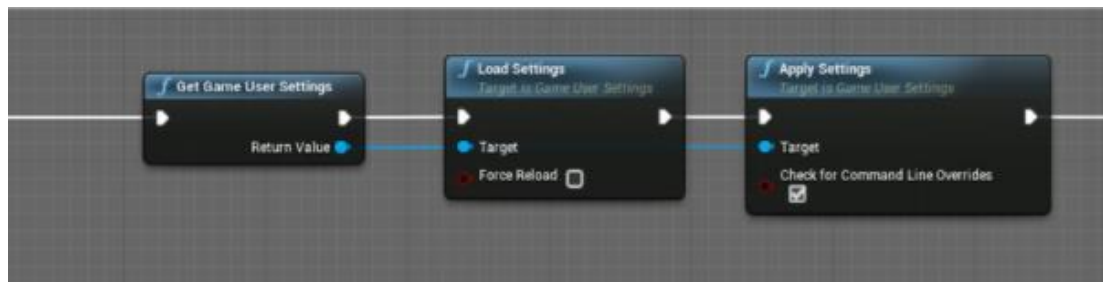


Рисунок 4.13 – Збереження налаштувань до файлу

Для налаштування варіантів зміни роздільної здатності екрану було використано функцію `GetSupportedFullscreenResolution`. За допомогою даної функції викликається масив з усіма можливими роздільними здатностями моніторів у типі `IntPoint`, що відповідатиме певній роздільній здатності у пікселях (рис. 3.14).

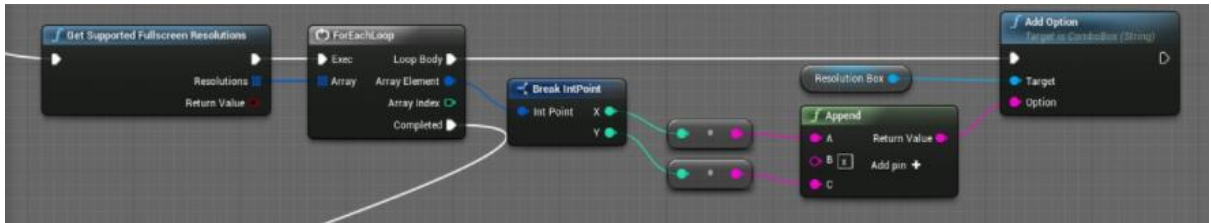


Рисунок 4.14 – Додавання параметрів у список

Далі необхідно створити подію, яка відповідатиме за встановлення роздільної здатності, обраної гравцем (рис. 4.15)

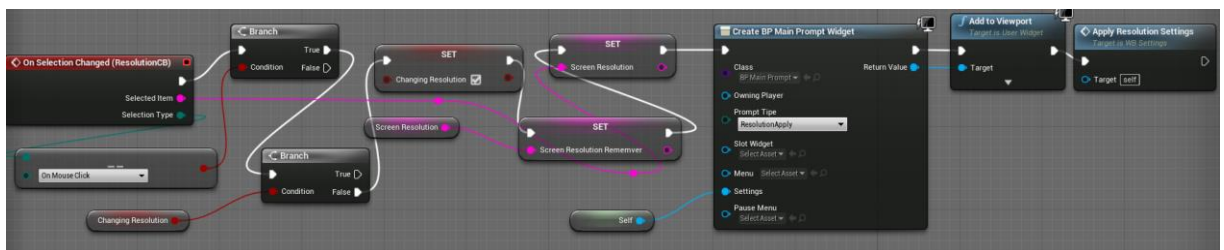


Рисунок 4.15 – Встановлення налаштувань роздільної здатності екрану

Розглянемо також налаштування якості згладжування. На рисунку 4.16 наведено Blueprint, котрий відповідає за зменшення та збільшення показника згладжування – це відбувається за допомогою функцій Clamp Settings Decrease та Clamp Settings Encrease (рис. 4.17).

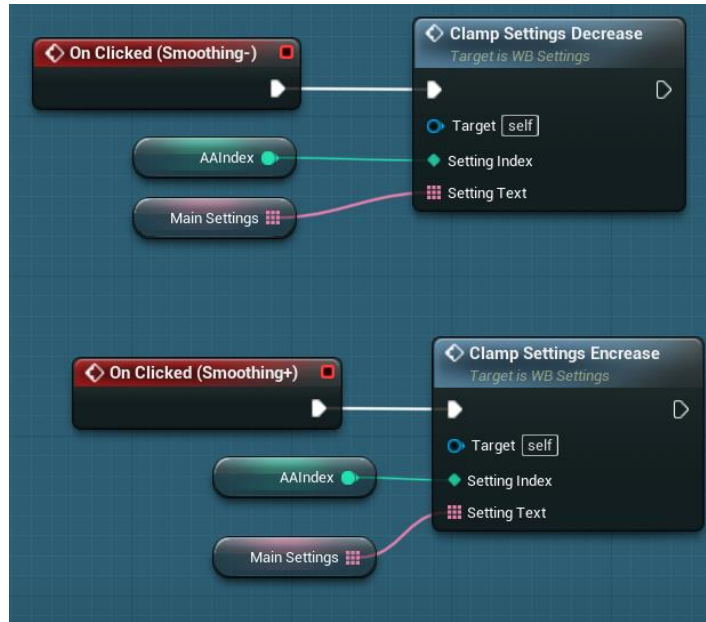


Рисунок 4.16 – Налаштування показника згладжуваності

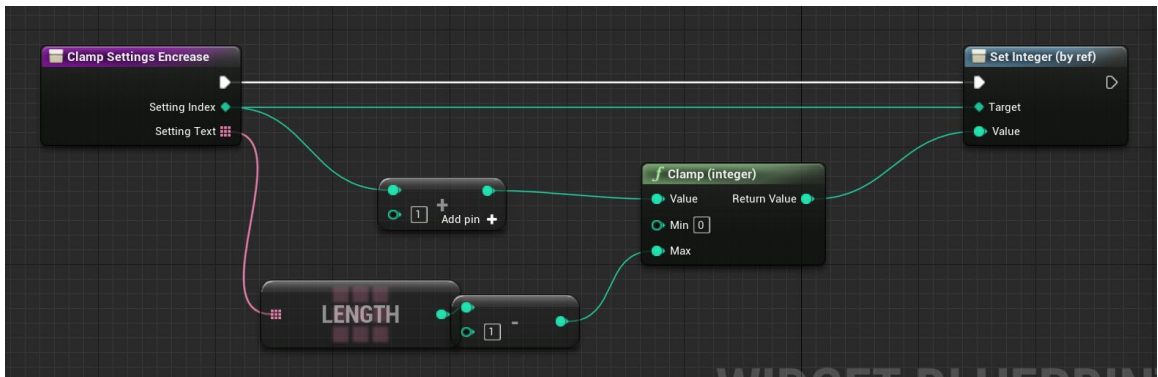


Рисунок 4.17 – Функція Clamp Settings Encrease

Інші подібні налаштування відбувалися за аналогічним принципом роботи (рис. 4.18).



Рисунок 4.18 – Розробка основних налаштувань

За кількість освітлення в локації відповідає змінна гамма. Для налаштування гамми в грі використовується додаткова кнопка. При натисканні на неї викликається подія (рис. 4.19), яка запускає додатковий віджет для зміни гамми (рис. 4.20).

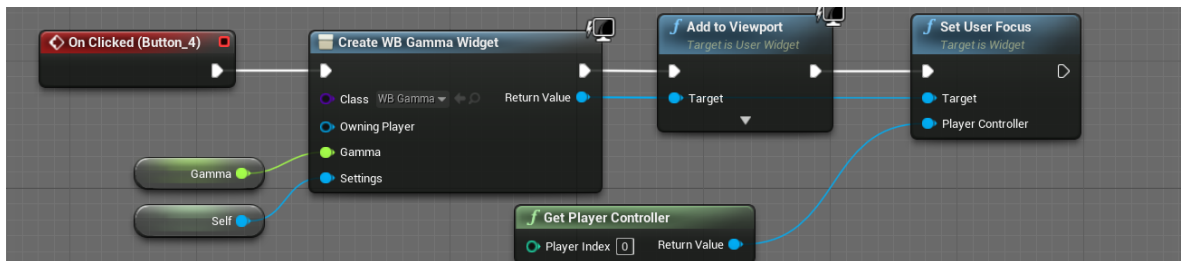


Рисунок 4.19 – Відкриття вікна налаштування гамми

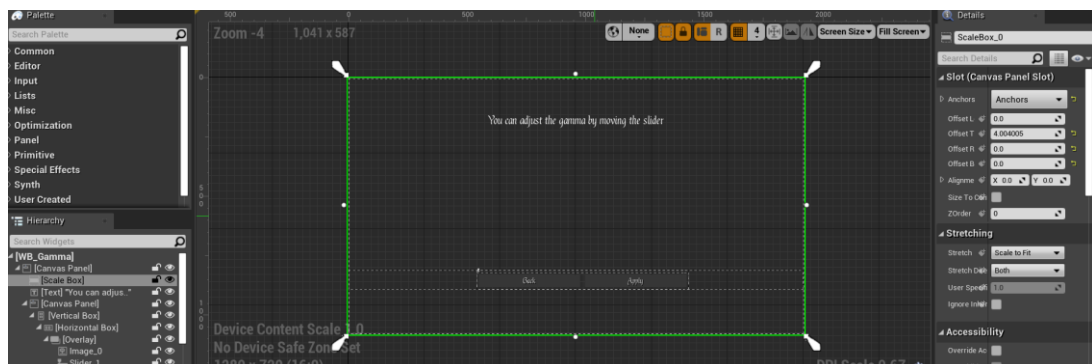


Рисунок 4.20 – Віджет для налаштування гамми

Після натискання на кнопку «Apply» створено наступні можливості для користувача – запам'ятовування налаштованої гамми, встановлення налаштувань, зміна гамми та повернення до попереднього стану, всі ці налаштування наведені на рисунку 4.21.

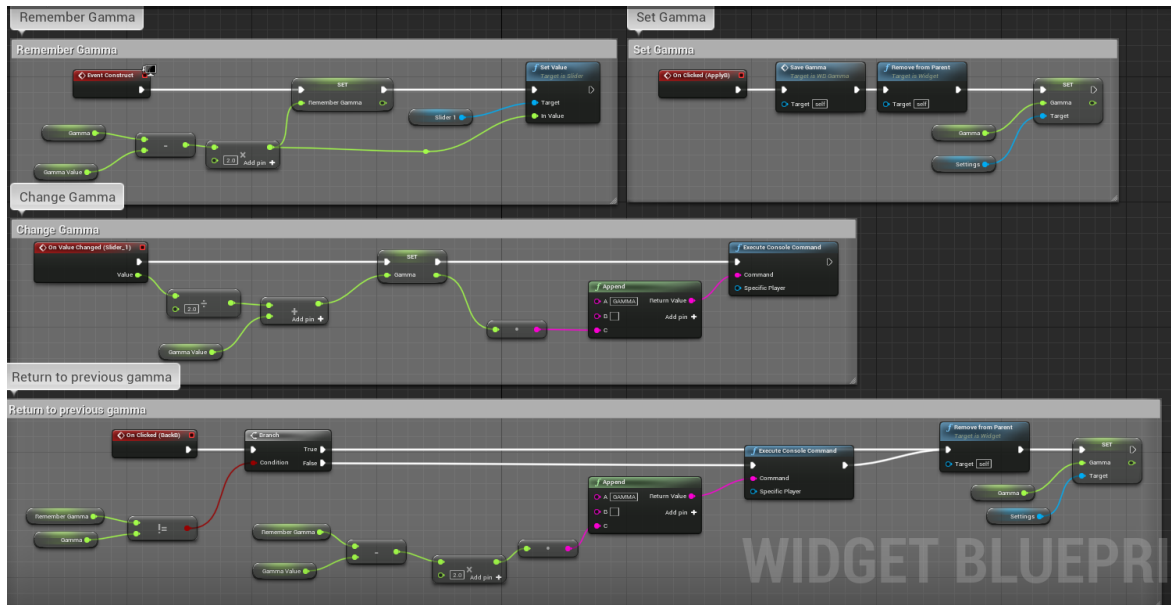


Рисунок 4.21 – Налаштування гамми

4.3.2 Налаштування збереження

У кожній грі є певний прогрес гравців, тому вирішено створити можливість збереження. Для цього було розроблено окремий віджет, наведений на рисунку 4.22. На ньому відображається місце збереження, тобто перший або другий рівень квесту, а також дата та час збереження [25].

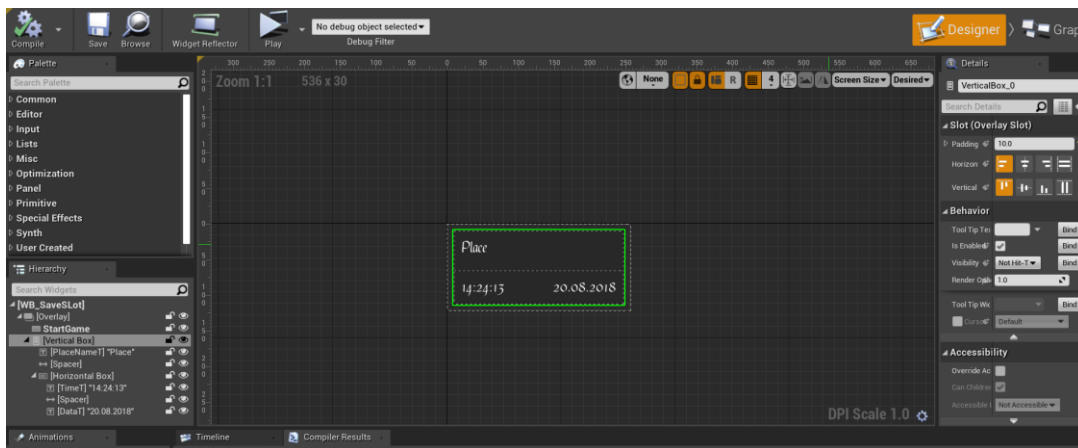


Рисунок 4.22 – Віджет слоту збереження

Дані дати та часу отримуються з системних налаштувань персонального комп'ютера, на якому встановлений додаток, за допомогою звичайної функції `GetTime` та `GetDate` (рис. 4.23). Рівень гри отримуємо за допомогою коду, зображеного на рисунку 4.24.

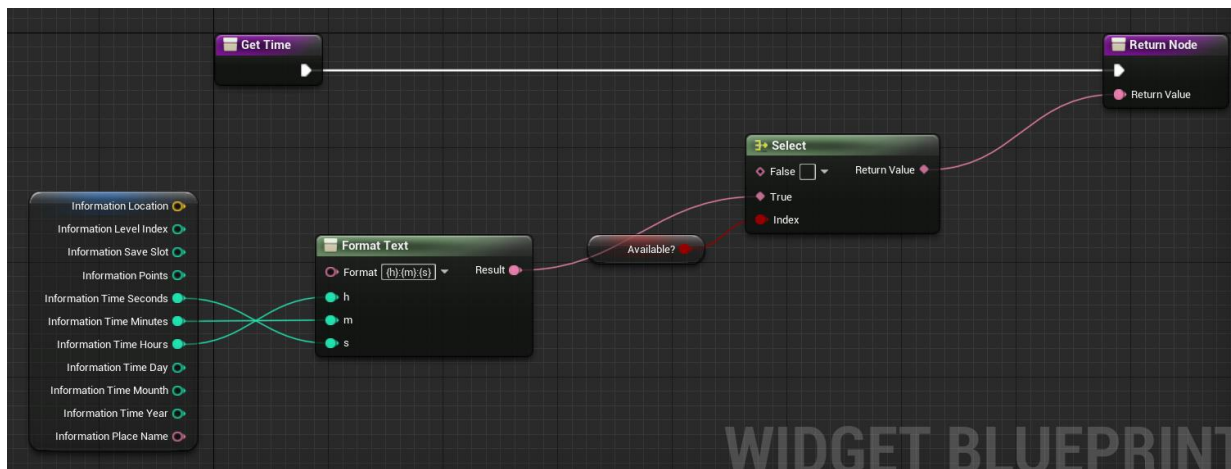


Рисунок 4.23 – Приклад функції `GetTime`

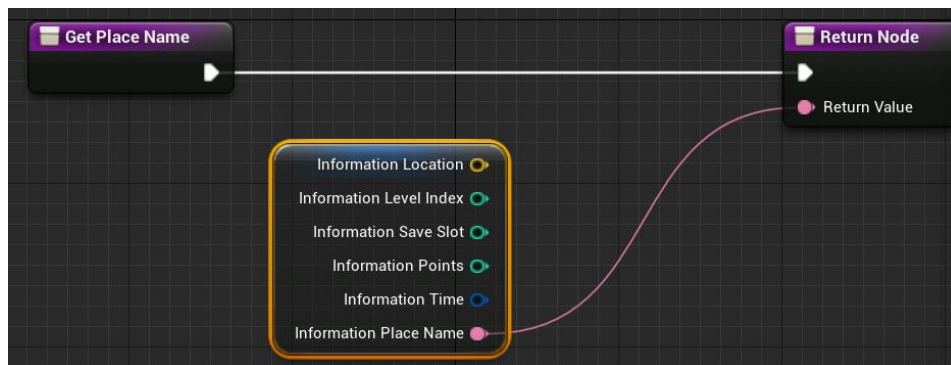


Рисунок 4.24 – Отримання інформації про рівень гри

Після початку гри система перевіряє, чи перший раз відбувається вхід до гри (рис. 4.25), тобто обираємо slot, що має вже певну інформацію про гру, або ж пустий slot.

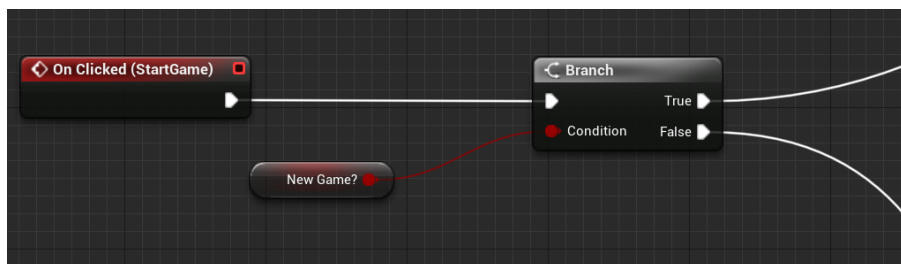


Рисунок 4.25 – Початок гри

Якщо обрано нову гру, то гравець переміщається на перший рівень гри з індексом нуль (рис. 4.26). Якщо обрано продовжити гру, то викликається Blueprint, наведений на рисунку 4.27, за допомогою якого завантажуються збережений в slotі індекс мапи та гравець продовжує існуючу гру.

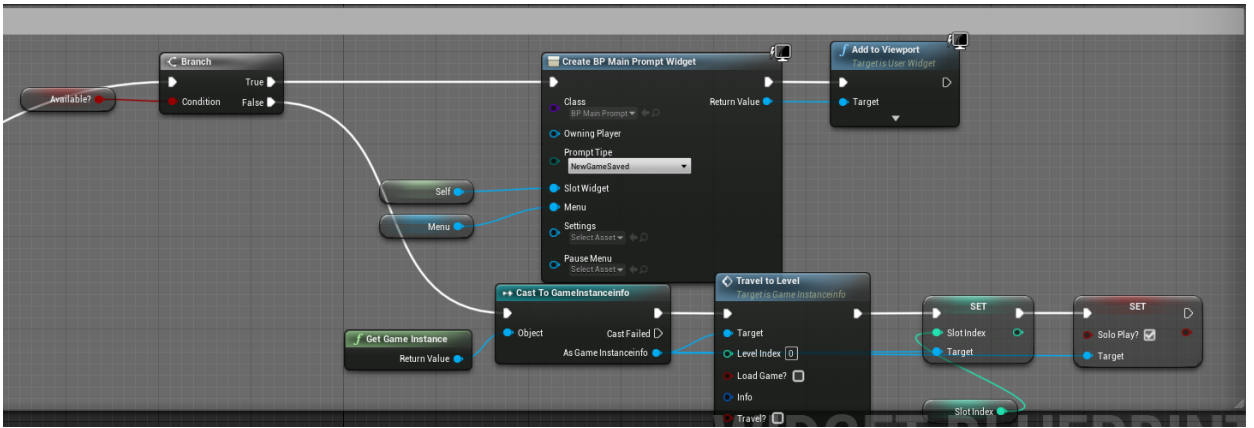


Рисунок 4.26 – Початок нової гри

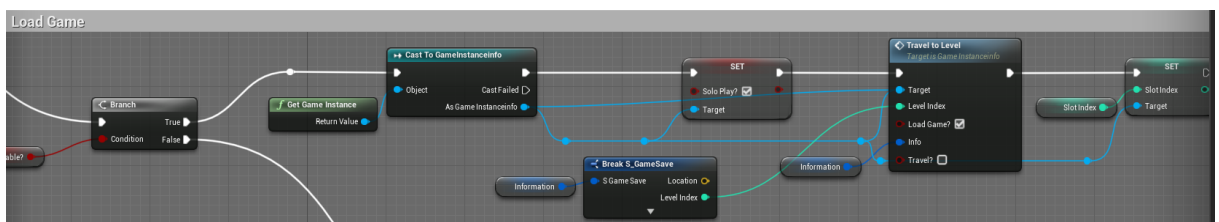


Рисунок 4.27 – Завантаження існуючої гри

На рисунку 4.28 показано, як відбувається подія збереження гри. Після натискання кнопки збереження, поточний прогрес гравця витягується зі змінної Game Instance, займається вільний слот з меню, в котрий додатково записується інформація про час, дату та місце збереження персонажу.

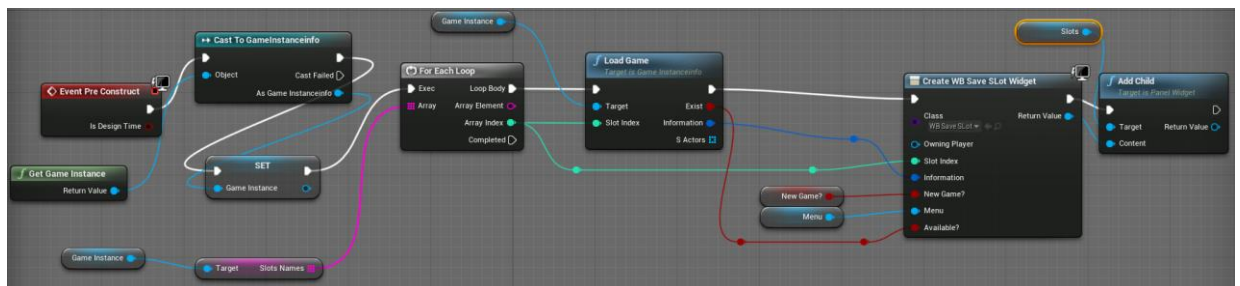


Рисунок 4.28 – Збереження гри

4.4 Розробка завдань квесту

4.4.1 Розробка сувоїв

Розглянемо створення сувоїв. Для початку на сувої маємо два основні елементи – це кнопка закриття, а також заданий основний текст (рис. 4.26).

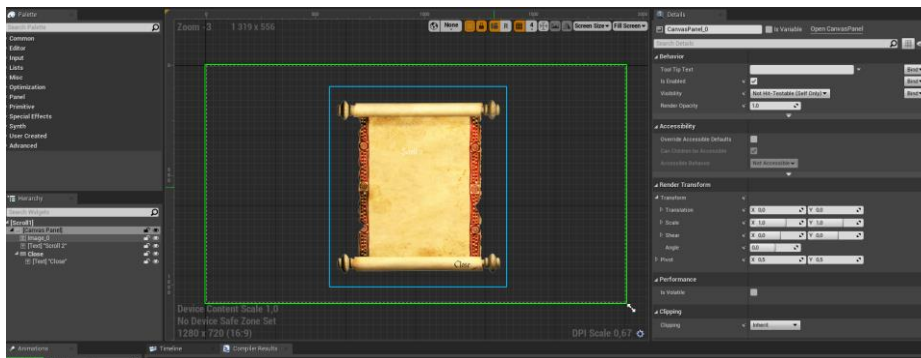


Рисунок 4.26 – Віджет сувою

На мапі кожен сувій супроводжується декількома кулями, які парять навколо даного об'єкту, аби вони були більш помітні та привертали більше уваги. На рисунку 4.27 наведено приклад одного з них.

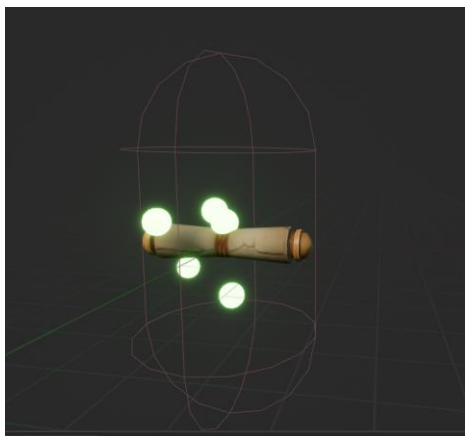


Рисунок 3.27 – Сувій з кулями навколо

Добре помітно, що навколо об'єкту існує колайдер (контейнер). При контакті гравця з ним даний сувій потрапляє до інвенторі-меню. Під час гри

сувої обертаються. Віджет реалізації такої його можливості зображено на рисунку 4.28, де ми бачимо, що обертання предмету задано по осі z.

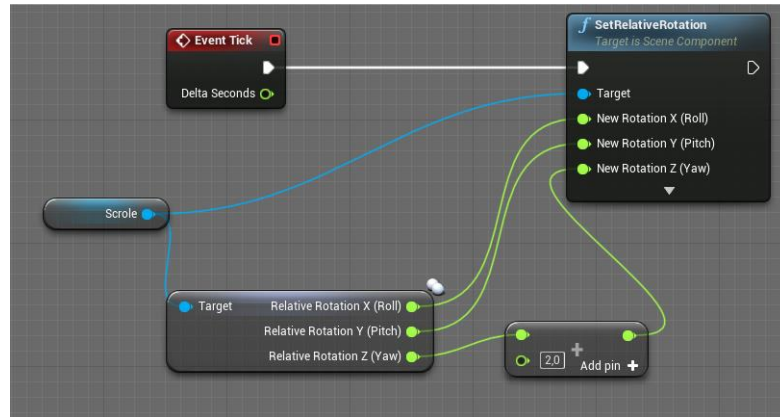


Рисунок 4.28 – Обертання сувою

Кулі навколо сувою також мають певну анімаційну логіку, аби вони також переміщалися разом з об'єктом, а не просто висіли у повітрі. На рисунку 4.29 зображено анімаційну логіку, а саме випадкова кількість кульок плавно переміщаються у просторі. Для плавних рухів створено PlayMeshAnimation (рис. 4.30), який також реалізує реверсійний рух для безперервного переміщення.

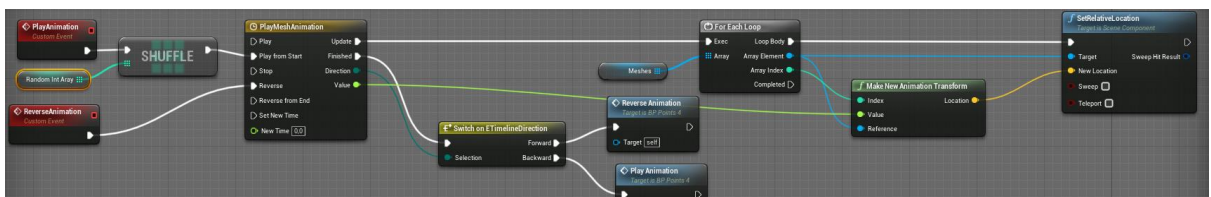


Рисунок 4.29 - Анімаційна логіка кульок

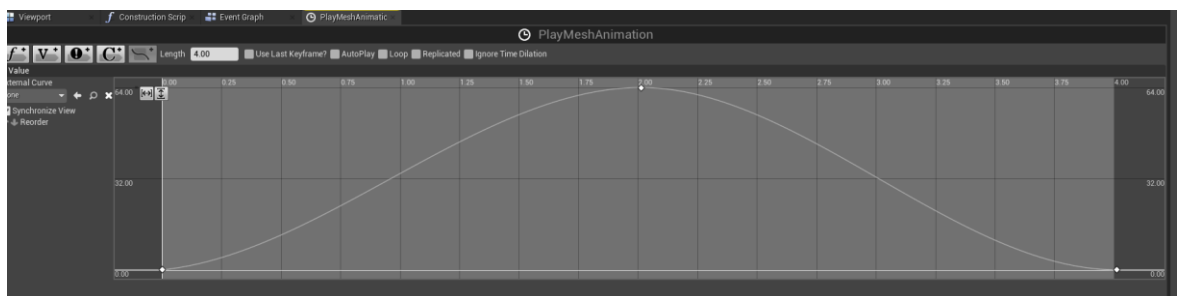


Рисунок 4.30 – PlayMeshAnimation

Початок задавання кількості кульок та запуск їх анімації зображено на рисунку 4.31.

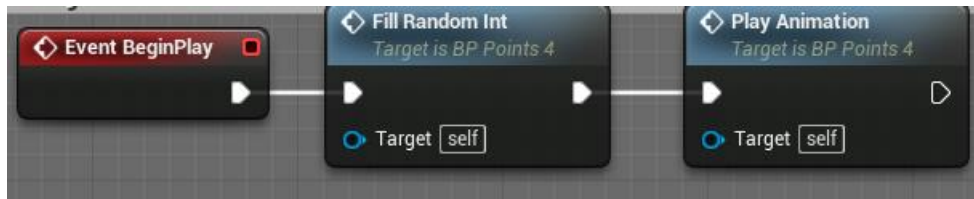


Рисунок 4.31 – Початок анімації кульок

Щоб кількість кульок була випадковою (рандомною), створено функцію, зображену на рисунку 4.32. З її допомогою встановили максимальну і мінімальну кількість об'єктів.

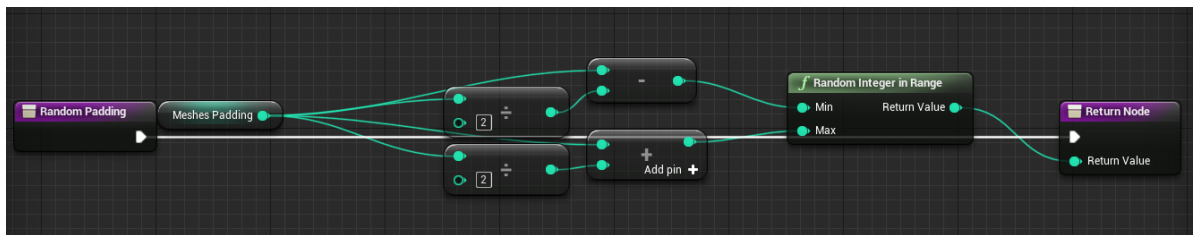


Рисунок 4.32 – Функція рандому

Додавання сувою до інвенторі-меню, після того, як персонаж перетне колайдер, зображено на рисунку 4.33.

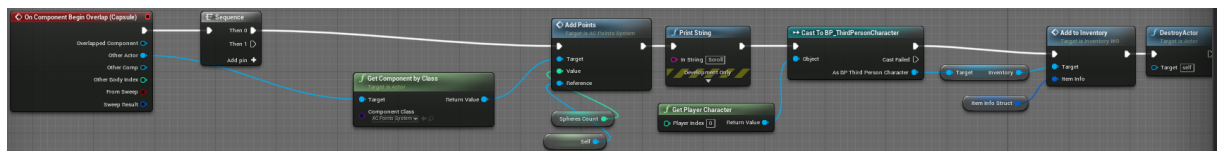


Рисунок 4.33 – Додавання сувою до інвенторі-меню

4.4.2 Розробка інвенторі меню

Для початку розглянемо, як можливо зберігати знайдені предмети (такі як сувої), та мати до них доступ протягом всієї гри. Аби реалізувати дану можливість, вирішено створити меню з інвентарем, яке можна викликати за

допомогою натискання кнопки «I» під час гри. Налаштування даної та інших гарячих клавiш зображено на рисунку 4.34.

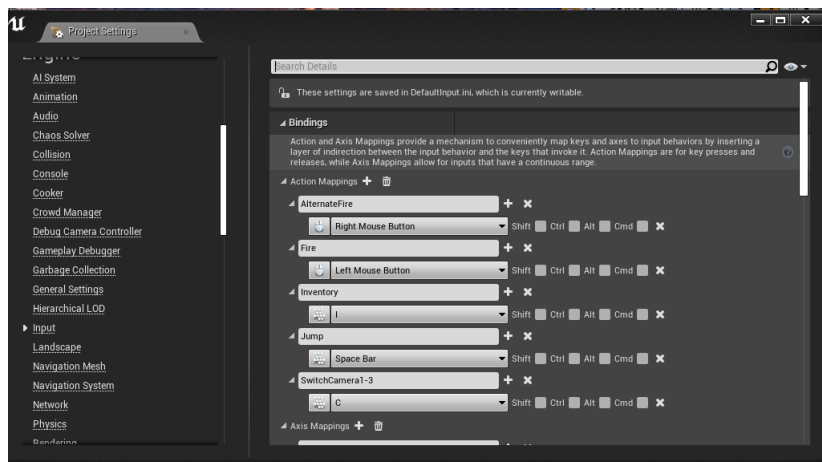


Рисунок 4.35 – Налаштування гарячих клавiш

Вiджет, вiдповiдальний за iнвенторi-меню (рис. 4.36), складається з декiлькох самостiйних вiкон (рис. 4.37). Разом вони створюють певний масив, в якому вже будуть всi предмети для зберiгання.

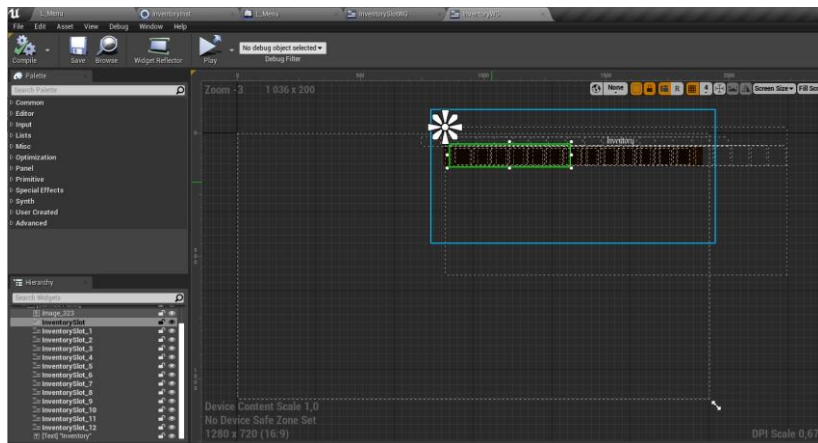


Рисунок 4.36 – Вiджет iнвенторi-меню

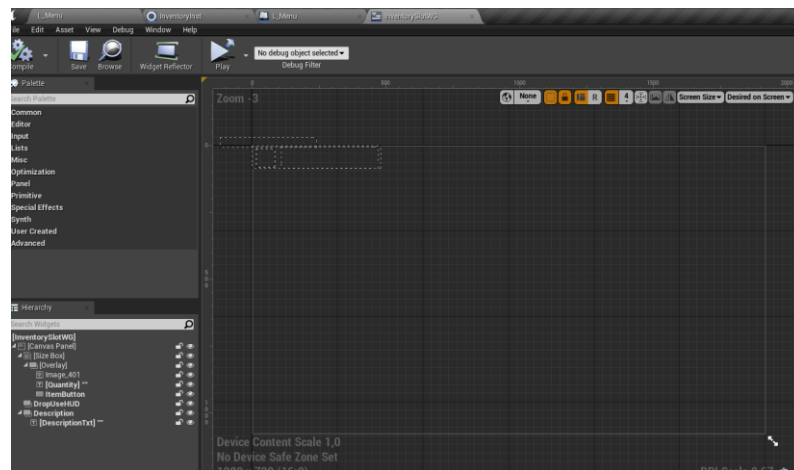


Рисунок 4.37 – Віджет, відповідальний за одну секцію інвенторі-меню

За створення упорядкованого масиву в інвенторі-меню відповідає наступний код, зображений на рисунку 4.38.

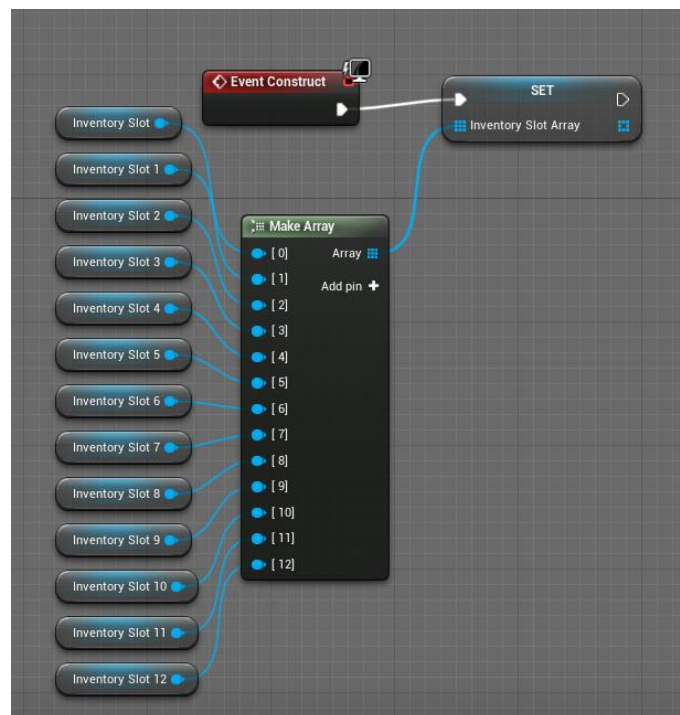


Рисунок 4.38 – Створення масиву із слотів

Також додана валідація на перевірку вільного місця з наступною логікою – перевірка, чи існує предмет, якщо є, додається стек, якщо він не існує, пошук вільного місця для розміщення нового предмета в інвенторі-меню (рис. 4.39).

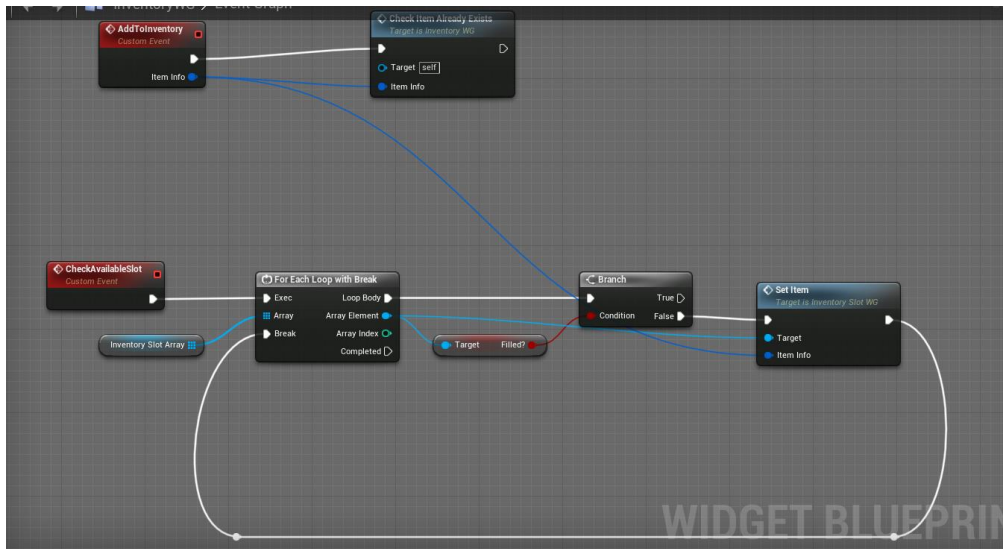


Рисунок 4.39 – Валідація в меню

Так як даний квест-додаток має два рівні, то додана валідація на перевірку рівня, в якому відкривається інвенторі. Це зроблено, аби не було випадків, що залишився непотрібний інвентар з попередньої гри. Дана валідація має у собі створену власноруч функцію – Check Item Already Exists (рис. 4.40), котра перевіряє слоти, вже зайняті первинним інвентарем. Зі змінної InventorySlotArray витягуємо інформацію стосовно слоту, що цікавить, далі за допомогою булевої операції перевіряємо заповнений даний слот чи ні.

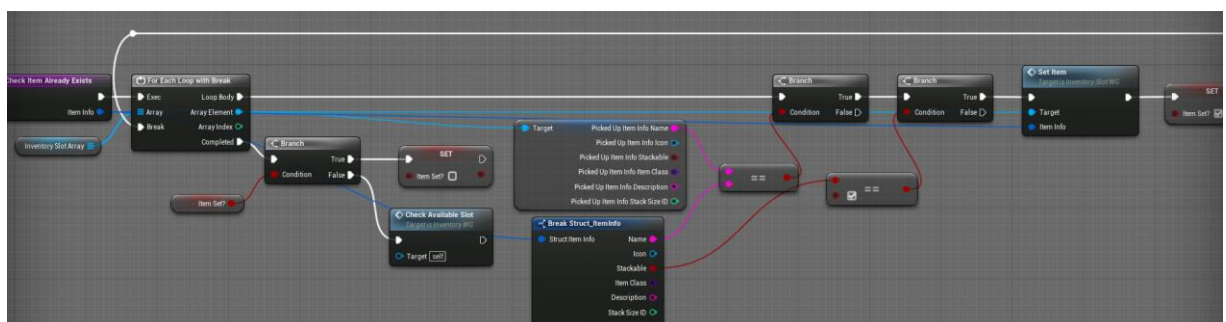


Рисунок 4.40 – Функція Check Item Already Exists

Звісно, передавати дані про об'єкт напряму до функції було б дуже незручно, адже кожен предмет унікальний та має свій набір даних. Саме тому вирішено упорядкувати дану інформацію, та створити структуру, щоб потім з

її допомогою структурувати дані про предмет і в окремій змінній зберігати дані про нього (рис. 4.41).

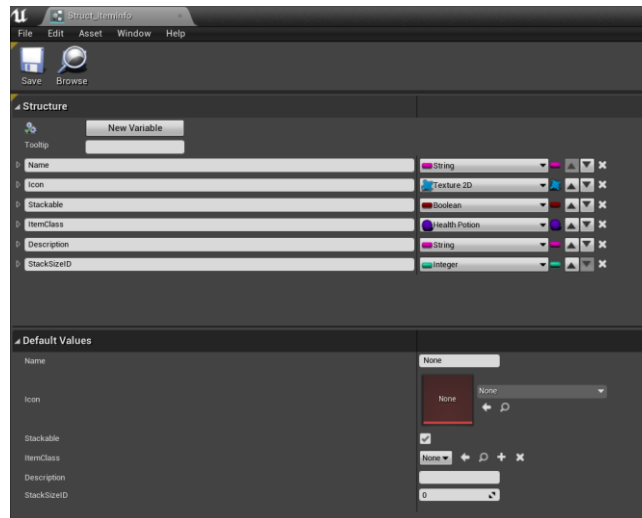


Рисунок 4.41 – Структура предмету

За допомогою цієї структури набагато зручніше додавати, та видаляти предмети з інвенторі-меню (рис. 4.42).

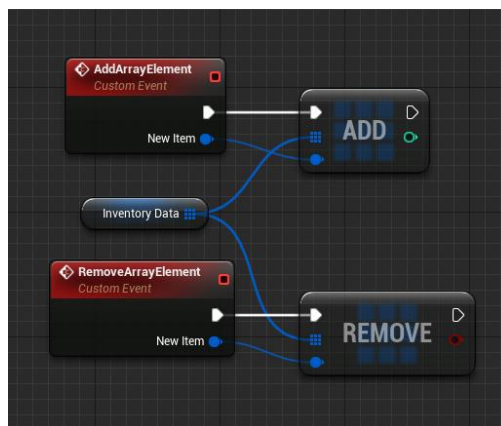


Рисунок 4.42 – Додавання та видалення предметів

Виклик інвенторі-меню персонажем наведено на рисунку 4.43, де на початку гри гравець бачить, що існує таке контекстне меню, може його закрити. На рисунку 4.44 зображено код, що відповідає за відкриття інвенторі-меню за допомогою натискання клавіші «I».

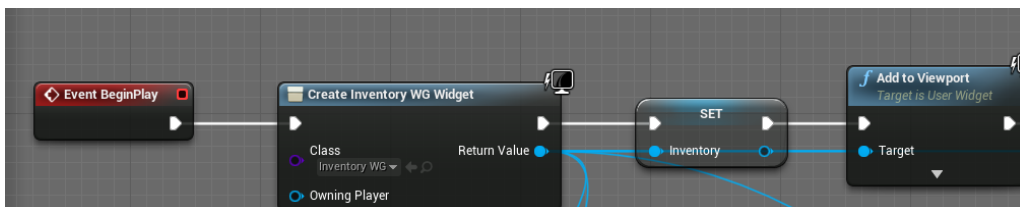


Рисунок 4.43 – Відкриття меню на початку гри

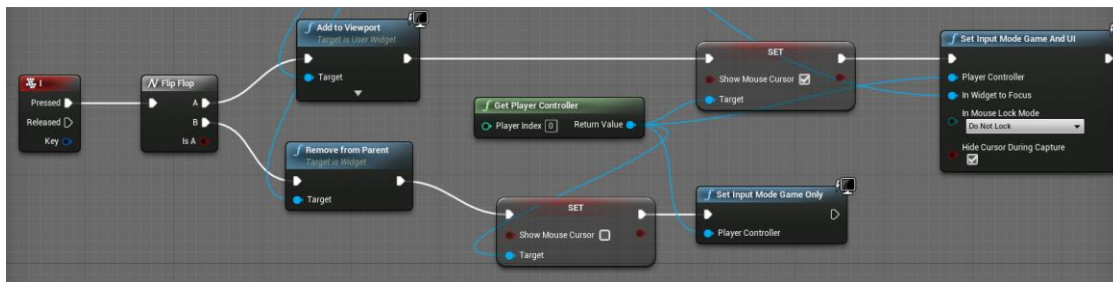


Рисунок 4.44 – Відкриття інвенторі-меню за допомогою клавіші

Щоб усю інформацію, збережену до інвенторі-меню, можна було відкрити, створено кнопку Open для кожного об'єкту. Інформація, представлена гравцю, повинна відповідати обраному сувою, за що відповідає код, наведений на рисунку 4.45. Основна логіка даного блупринта в тому, що при натисканні в інвенторі-меню на певний сувій, з нього вивантажується інформація до раніше створеної структури. Дана інформація одночасно порівнюється з усіма існуючими конструкціями, поки не буде знайдено потрібну, яка і виведеться на екран користувачу.

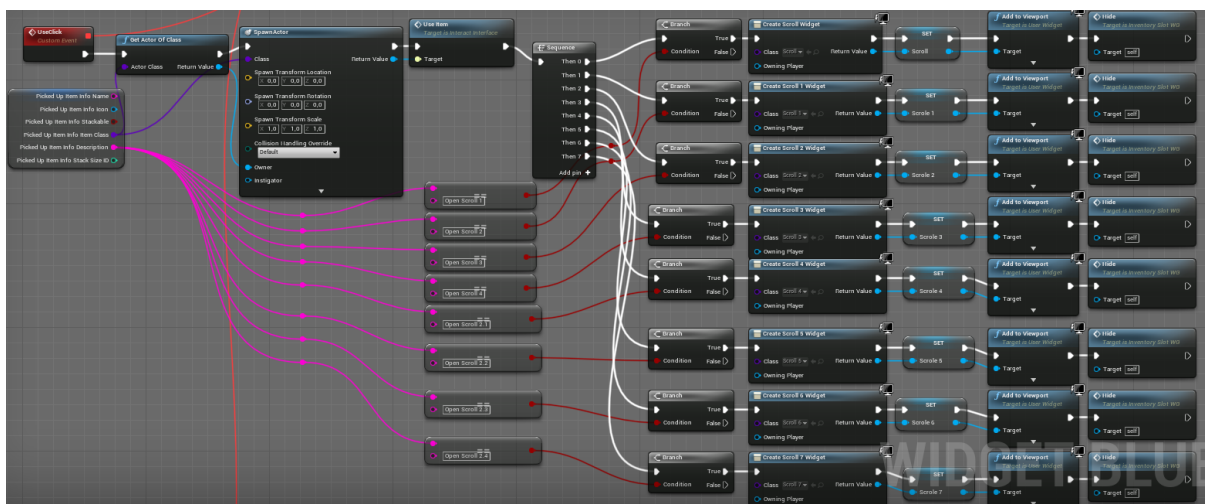


Рисунок 4.45 – Перебір інформації з сувоїв

Щоб закріпити інформацію певного сувою за конкретним слотом в меню, створено блупрінт, зображений на рисунку 4.46. В ньому реалізовано наступне – інформація передається для зберігання та подальшого використання до структури об'єкту, слот меню заповнюється зображенням, щоб було зрозуміло, який об'єкт там знаходиться, та закріплюється текст, аби при наведенні він з'являвся як підказка назви збереженого об'єкту.

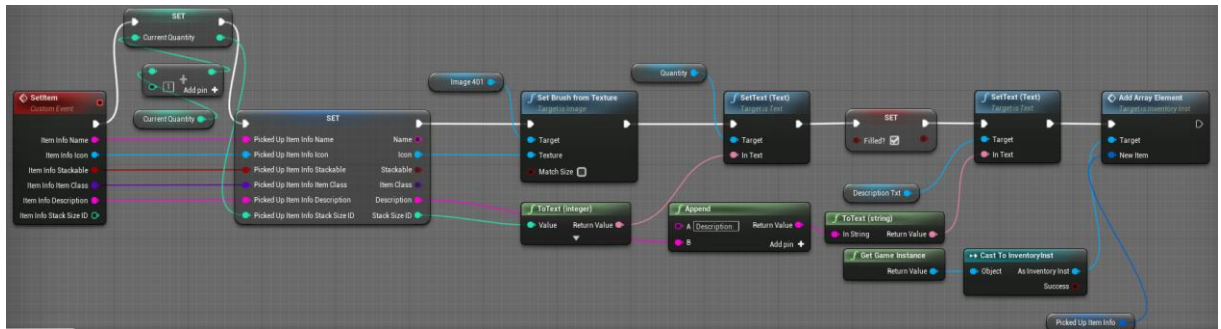


Рисунок 4.46 – Закріплення інформації в меню

4.4.3 Розробка прихованих дверей

В квесті наявні двері, замасковані під звичайну шафу з книжками. Щоб гравець зміг відкрити дані двері, йому достатньо натиснути правою кнопкою миші по спеціальній книжці, яка є таємним ключем даної шафи (рис 4.47).

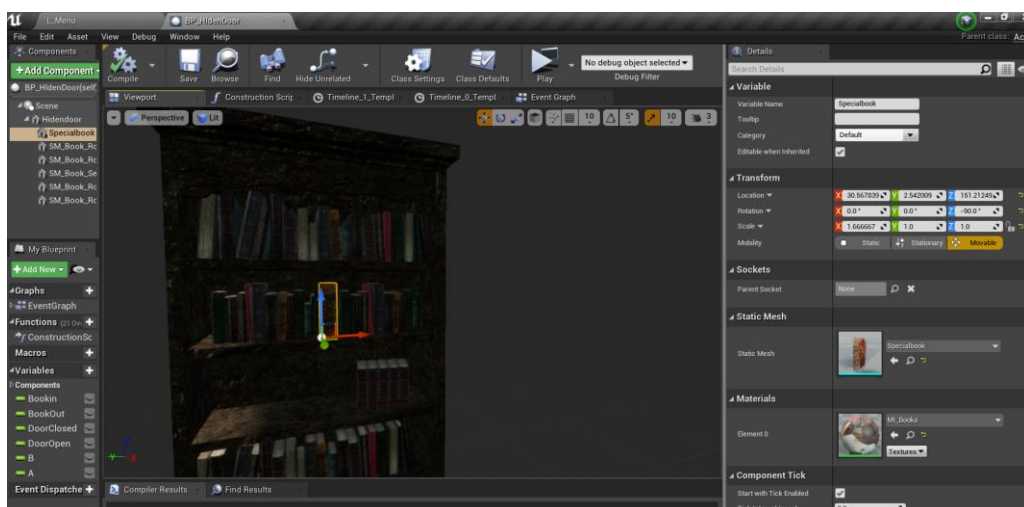


Рисунок 4.47 – Спеціальна книга-ключ

Сам механізм роботи даної шафи є доволі простим. Для початку ми створюємо тайм-лайн відхилення книги, аби після натискання вона відхилялася на декілька градусів, та тайм-лайн для відкриття самої шафи. На рисунку 4.48 наведено один із тайм-лайнів як приклад.

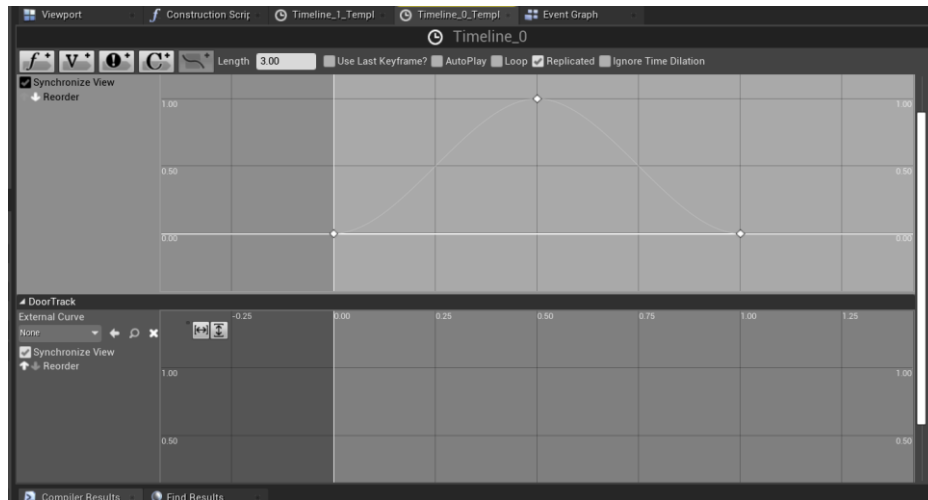


Рисунок 4.48 – Приклад тайм-лайну

На рисунку 4.49 наведено блупринт, котрий відповідає за відкриття дверей. Після натискання на книгу, вона відхиляється на декілька градусів, викликається функція встановлення повороту, та ми проставляємо нові параметри розташування шафи таким чином, аби вона відкрила прохід. За допомогою налаштованого тайм-лайну це відбувається плавно.

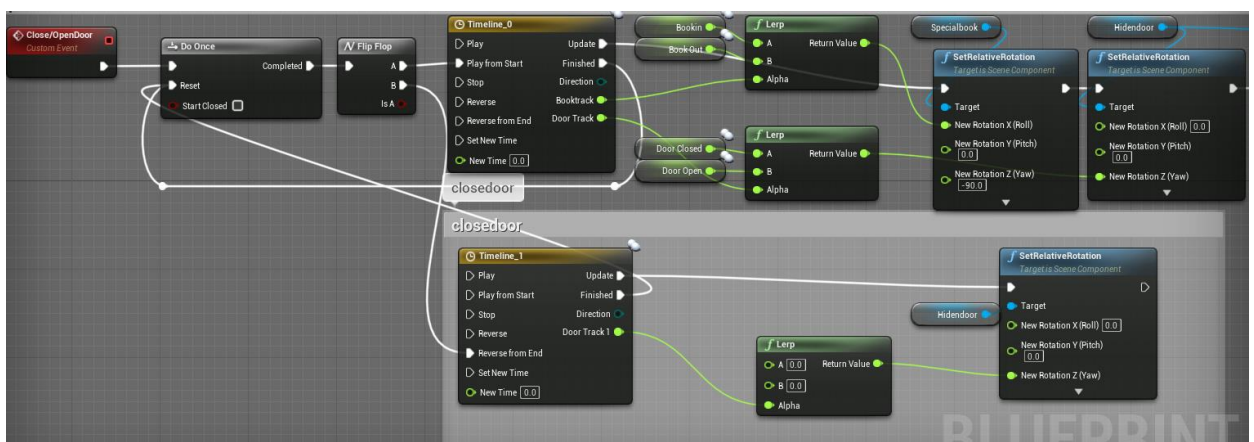


Рисунок 4.49 – Блупринт налаштування відкривання шафи

Налаштування відповідних трейс ліній (уявні лінії, котрі дозволяють взаємодіяти з предметом при попаданні точки даної лінії на необхідний об'єкт) для взаємодії гравця з книгою наведено на рисунку 4.50. Це зроблено, щоб шафа відкривалася лише при натисканні на певну книгу, а не при перетині триггеру-перемикача даної книги.



Рисунок 4.50 – Налаштування трейс ліній

4.4.4 Розробка кодового замку

Спочатку необхідно створити модель замку та модель тумблера, який гравець буде крутити аби ввести необхідну комбінацію. На рисунку 4.51 наведено візуалізація одного із тумблерів, з котрими відбувається взаємодія гравців. Також кожній грані тумблера відповідає певний градус повороту.

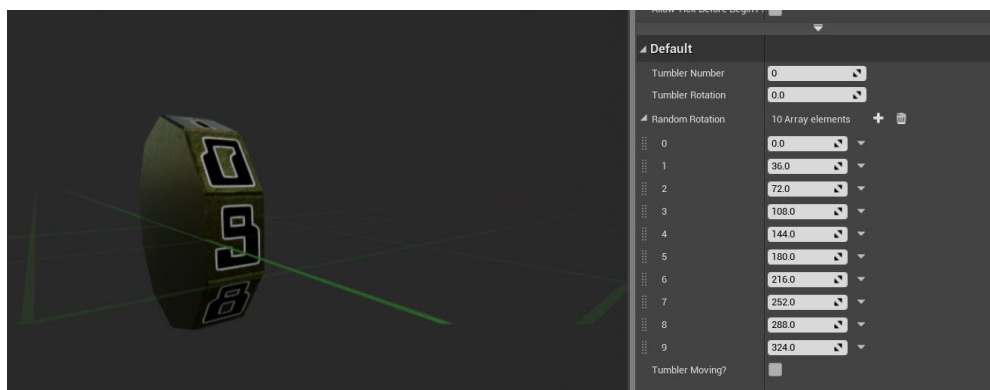


Рисунок 4.51 – Візуалізація тумблера

Далі написана логіка, яка надає можливість обертати даний об'єкт. Для цього створено два Custom Event, що відповідають за обертання тумблеру вперед та назад, та відповідний тайм-лайн для анімації обертання (рис. 4.52).



Рисунок 4.52 – Анімація обертання тумблеру

На рисунку 4.53 наведено повний блупрінт код для обертання тумблеру, де для початку по координаті X додається хід обертання об'єкту відповідно до дій користувача.

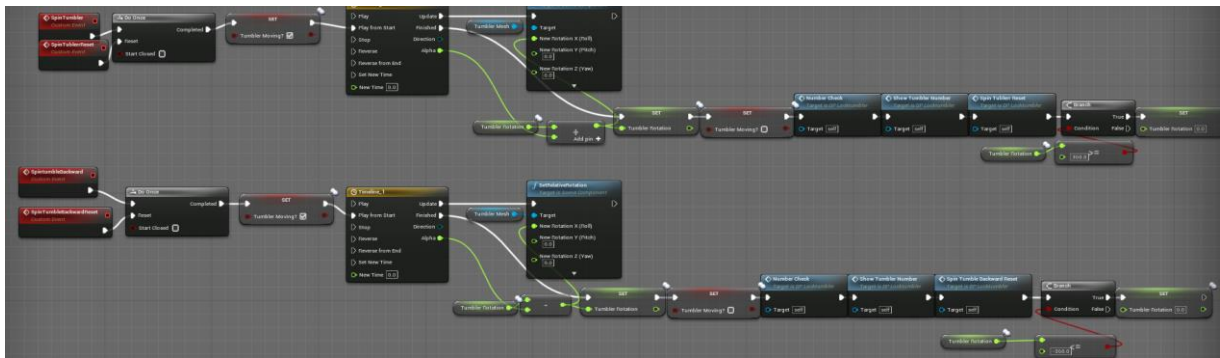


Рисунок 4.53 – Обертання тумблеру

Аби цифра на тумблері відповідала певному градусу обертання об'єкту та певному значенні для вводу коду, створено функція Number Check. На рисунку 4.54 наведено повний блупрінт даної функції, і видно як

встановлюється певне значення конкретній грані тумблеру. Збільшений фрагмент даного коду наведено на рисунку 4.55.

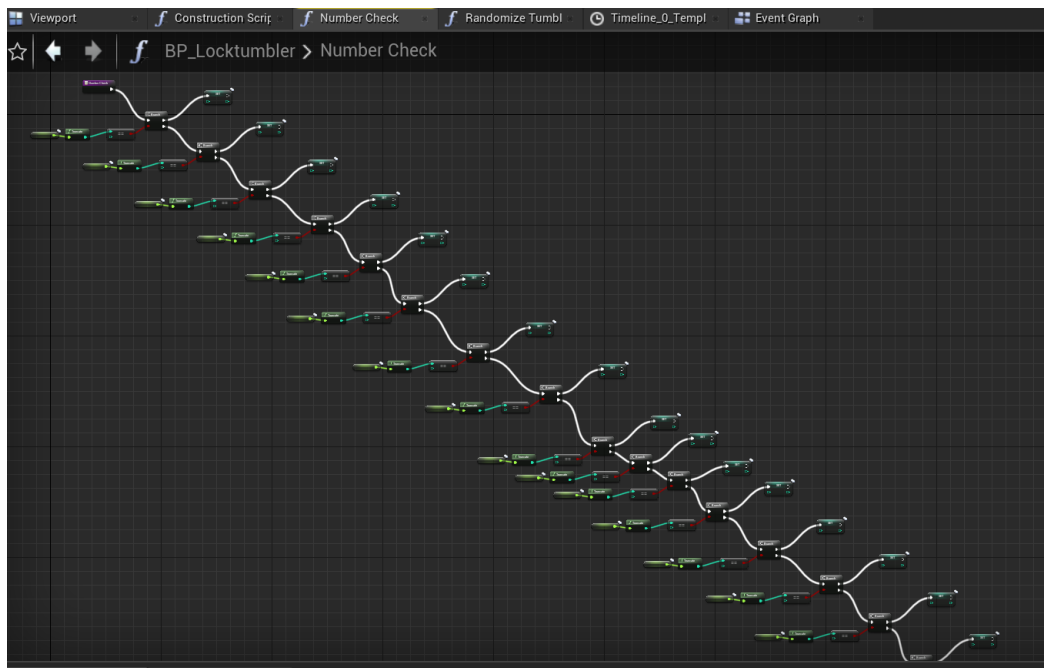


Рисунок 4.54 – Загальний вигляд функції

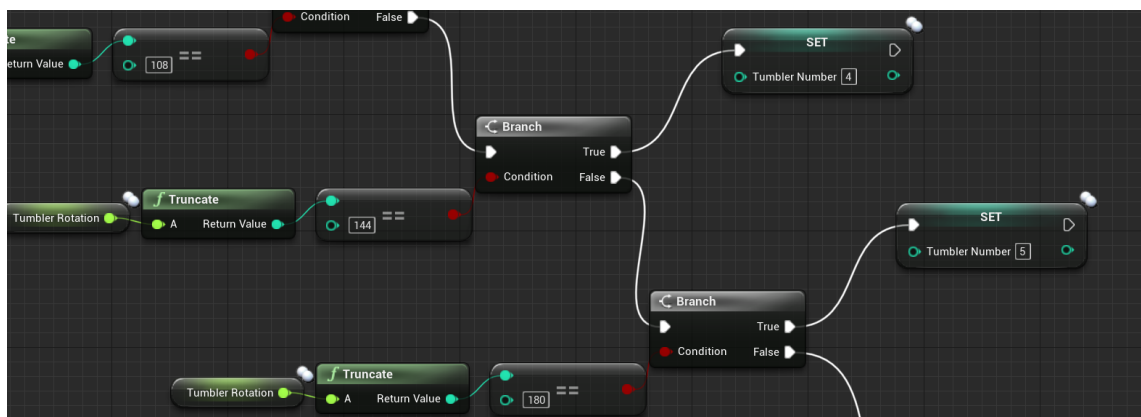


Рисунок 4.55 – Фрагмент коду функції

Після налаштування тумблеру можна зібрати сам замок, він матиме в собі декілька обертальних об'єктів (рис. 4.56).

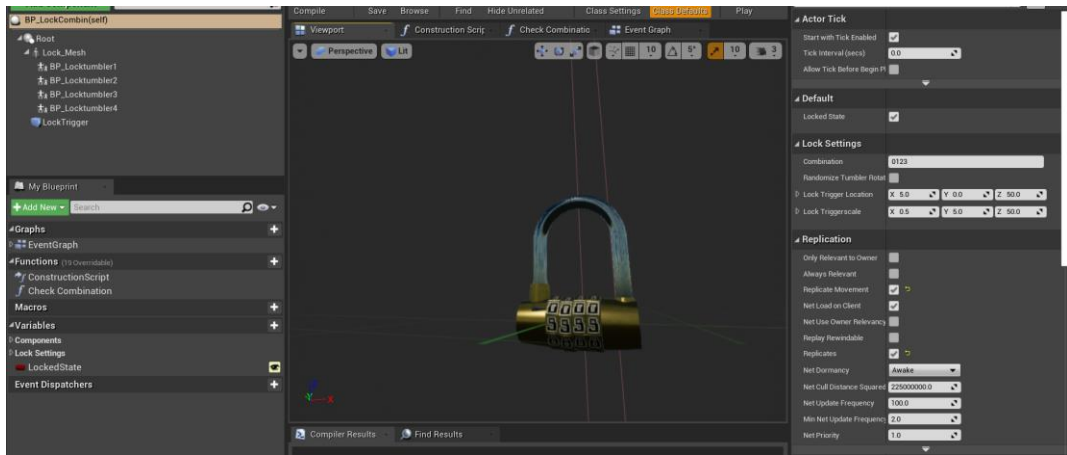


Рисунок 4.56 – Зібраний замок

Щоб положення кожного тумблера на момент запуску нової гри було завжди різним, створено функцію випадкового встановлення положення об'єктів (рис. 4.57).

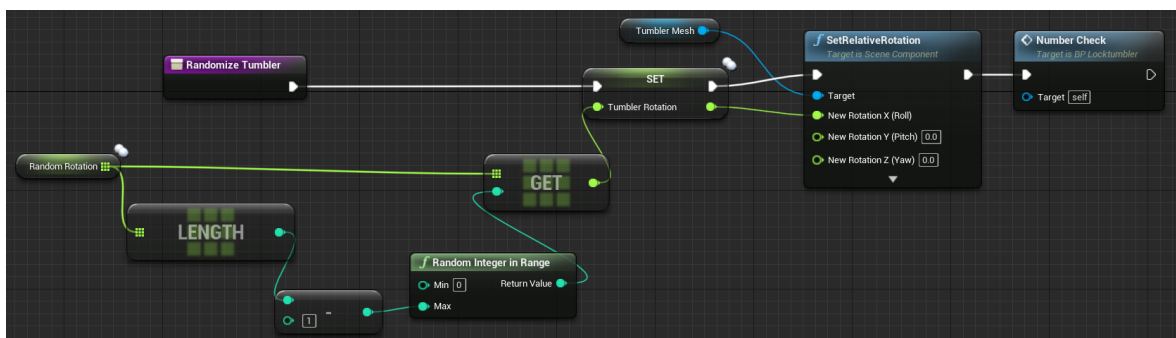


Рисунок 4.57 – Рандомізація об'єктів

Щоб замок та двері відчинилися, необхідно перевірити введену комбінацію гравцем (рис. 4.58) та створити окремий блупринт, який відповідатиме за металеві двері (рис. 4.59), закриті об'єктом (кодним замком) (рис.60).

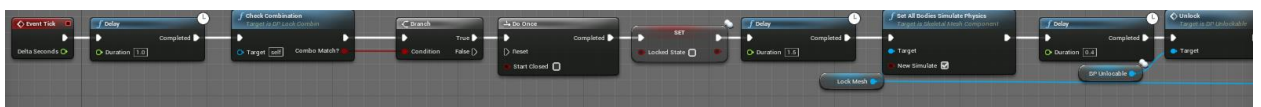


Рисунок 4.58 – Перевірка комбінації

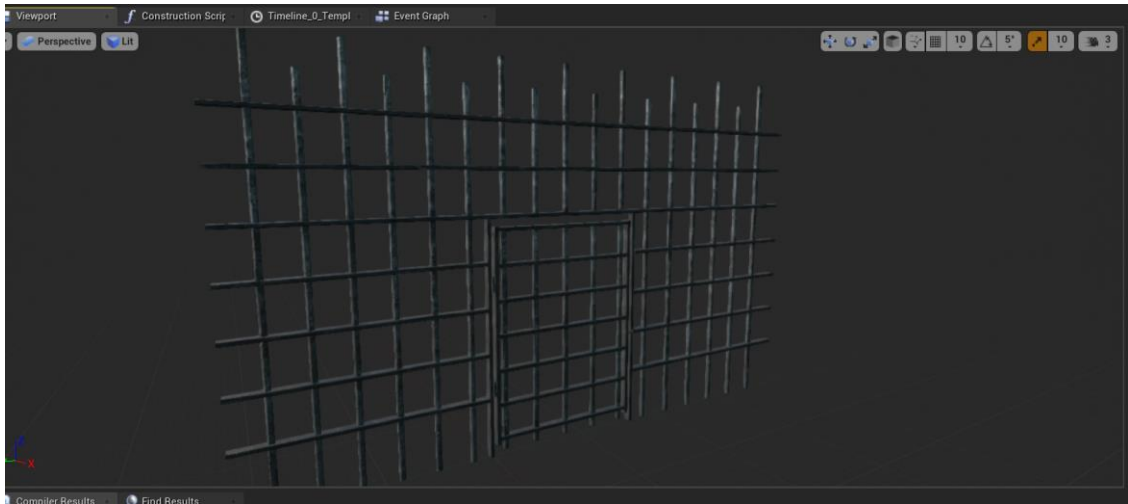


Рисунок 4.59 – Металеві двері

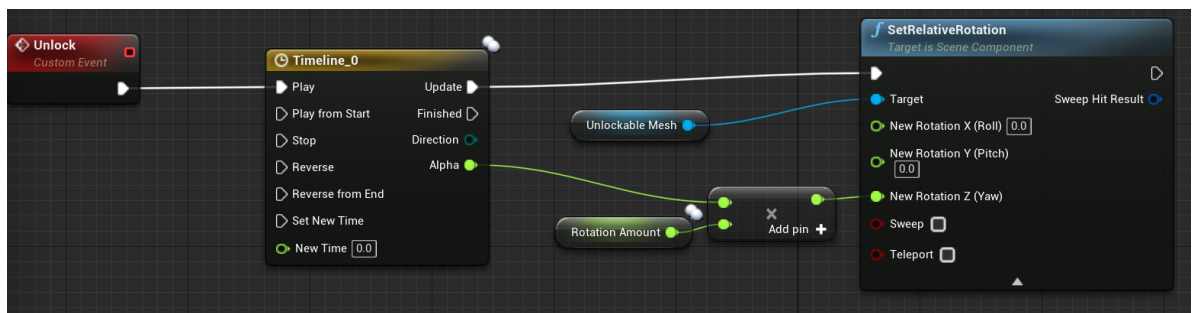


Рисунок 4.60 – Функція відкривання дверей

4.5 Розробка мережевої взаємодії

Для реалізації даної функції розглянемо спроможність підключення користувачів один до одного через мережу. Для початку необхідно встановити плагіни advanced steam sessions (рис. 4.61), котрий відповідає за можливість підключення до steam та online subsystem (рис. 4.62), та надає можливість створювати LAN-з'єднання [26].

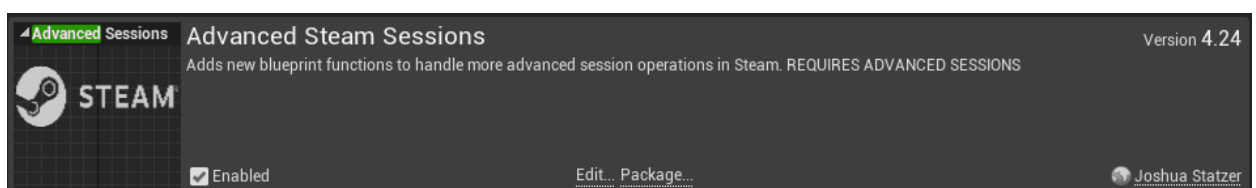


Рисунок 4.61 – Плагін advanced steam sessions

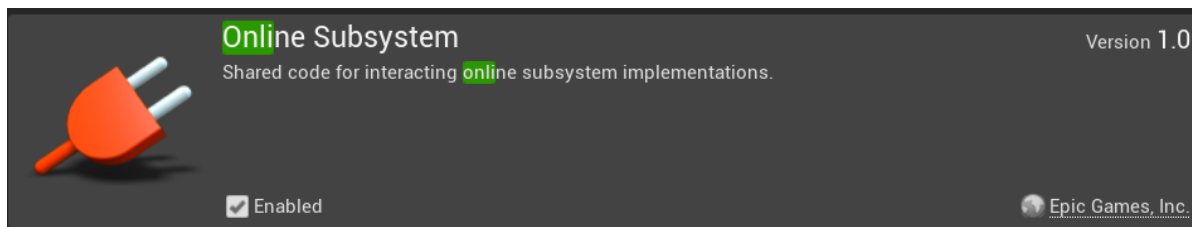


Рисунок 4.62 – Плагін online subsystem

В налаштуваннях проекту налаштовуємо мапи, а саме призначаємо, яка буде картою по замовчуванню, а яка буде транзитною, також вказуємо Game Mode та Pawn Class (рис. 4.63).

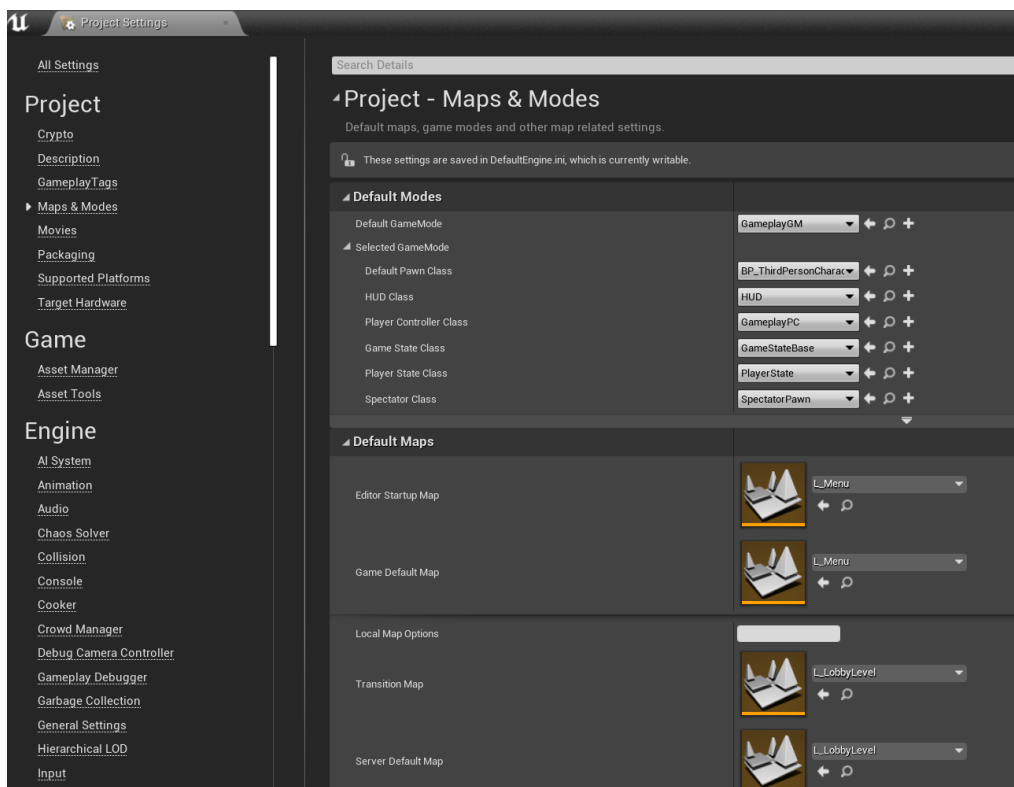


Рисунок 4.63 – Project settings

Щоб інформація про підключених гравців оновилася, та одразу розділити їх на клієнта та сервер, використовується блупринт, наведений на рисунку 4.64. Гравець, котрий створює лоббі, буде виступати сервером, і відповідно на його ПК буде більше навантаження. Вся інформація про гравців зберігається в змінній Player Settings (його ім'я, картинка в лоббі, готовність до гри і т.д.)

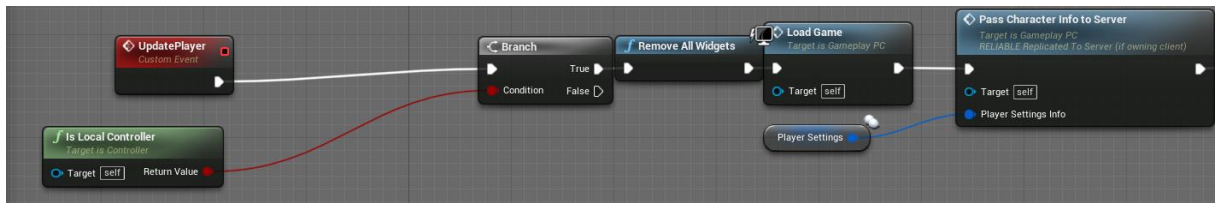


Рисунок 4.64 – Оновлення гравців

Для отримання Steam Nick name гравця використовуємо відповідну функцію, де йде зберігання імені до файлу та до слоту гри, а також всіх налаштувань користувача. При повторному підключення інформація підтягується зі збережених даних (рис. 4.65). В цих даних також зберігається прогрес гравця, обрані ним налаштування для графіки. Збережені файли можна знайти в корні проекту по шляху ...\\Saved\\SaveGames.

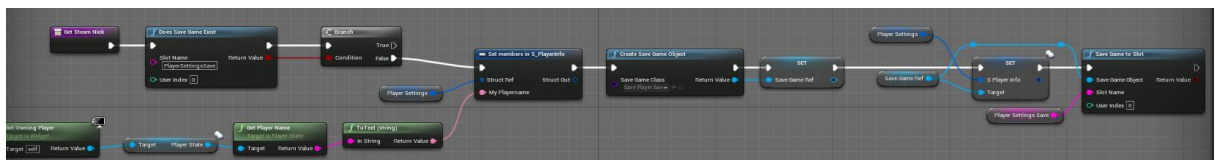


Рисунок 4.65 – Функція оновлення мережевої інформації

Саме збереження інформації до файлів наведено на рисунку 4.66, де видно, що збереження відбувається до файлу PlayerSettingsSave.sav.

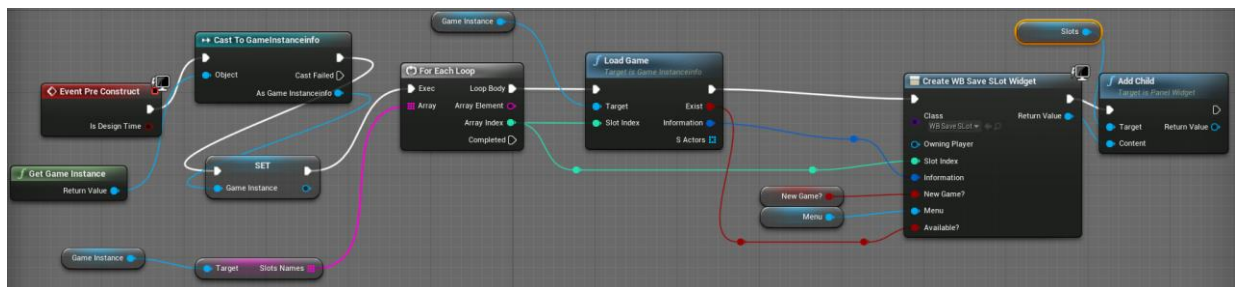


Рисунок 4.67 – Збереження до файлу

Якщо запустити гру при мережевому підключенні без додаткових налаштувань, то такі дії, як обертання голови, повертання тулуба іншого

гравця, не буде видно, лише повороти вліво та вправо. Щоб це виправити додатково описана логіка контролеру обертання (рис. 4.68-4.69).

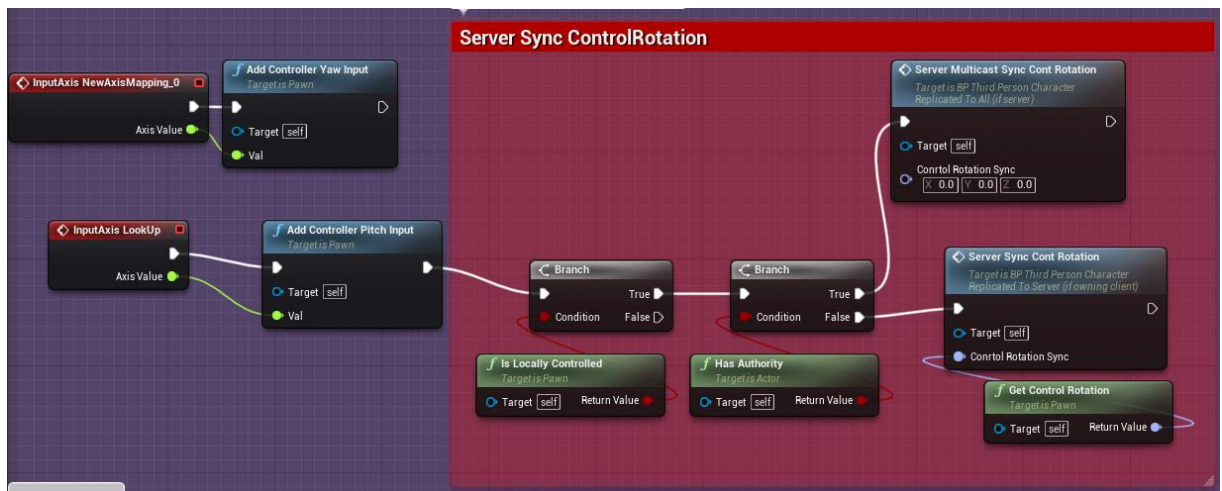


Рисунок 4.68 – Додатковий функціонал для обертання в мережі (перша частина)

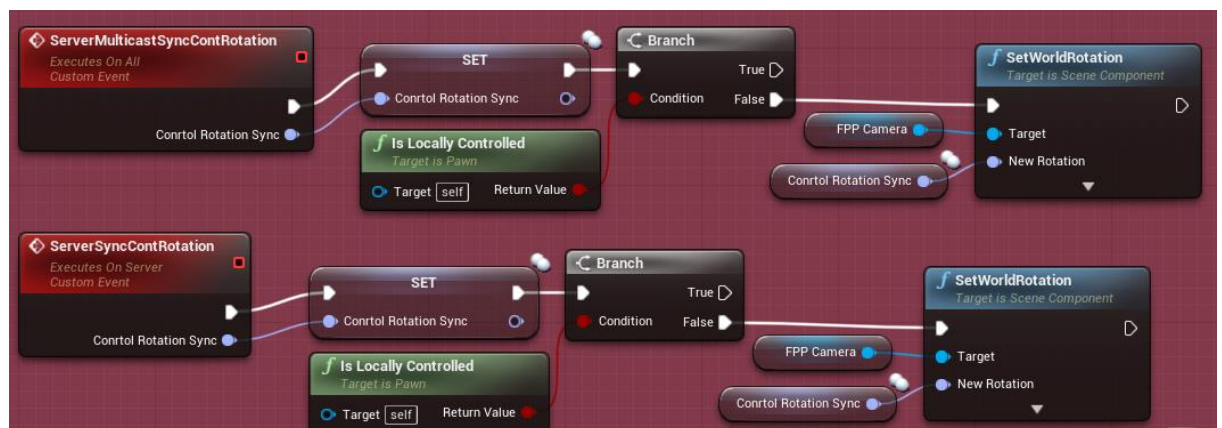


Рисунок 4.69 – Додатковий функціонал для обертання в мережі (друга частина)

Розглянемо налаштування можливості входу до серверу. Для початку необхідно створити відповідний віджет, для зручного візуального представлення (рис. 4.70).

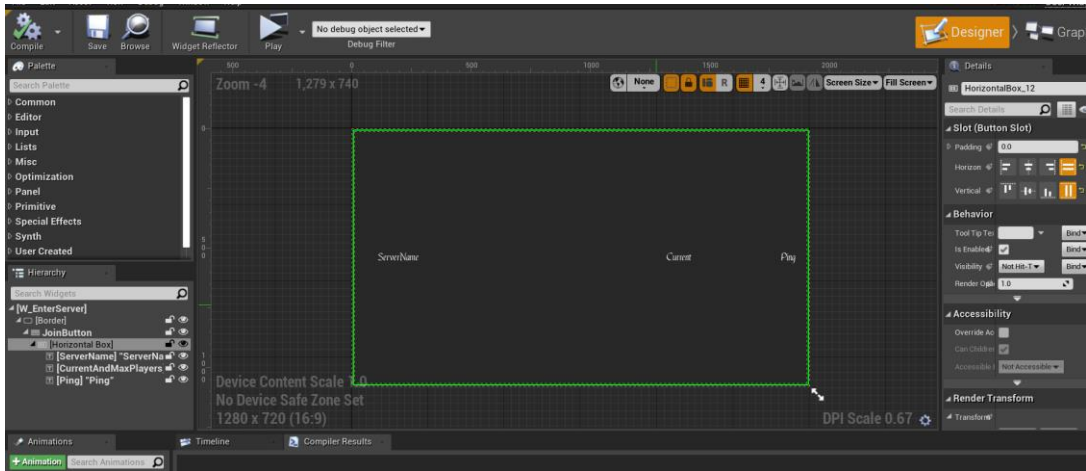


Рисунок 4.70 – Віджет для входу на сервер

Коли гравець натискає на створений сервер, викликається блупринт, зображений на рисунку 4.71, для підключення до сесії. Сама інформація про сервер, що створює гравець за допомогою лоббі, витягується з використанням функції, наведеної на рисунку 4.72.

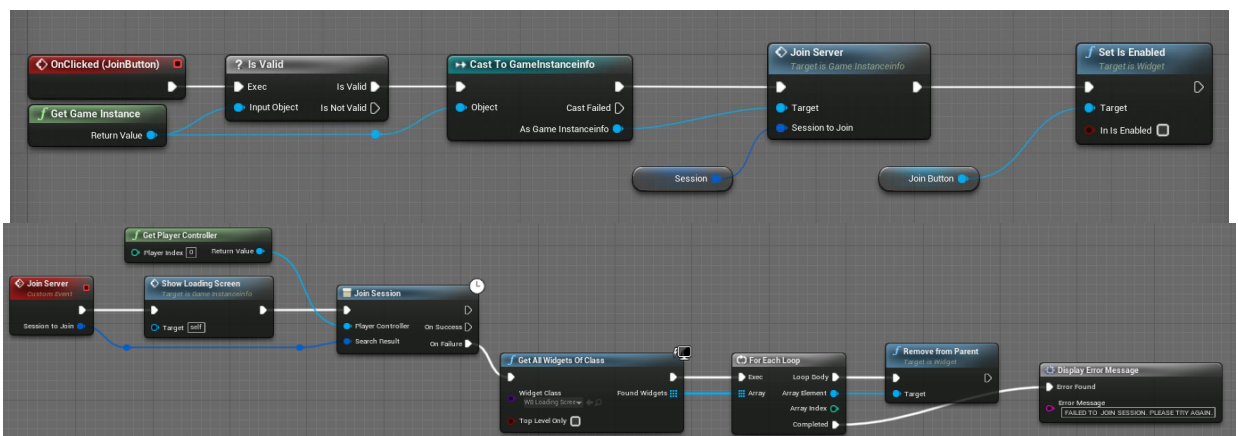


Рисунок 4.71 – Підключення до серверу

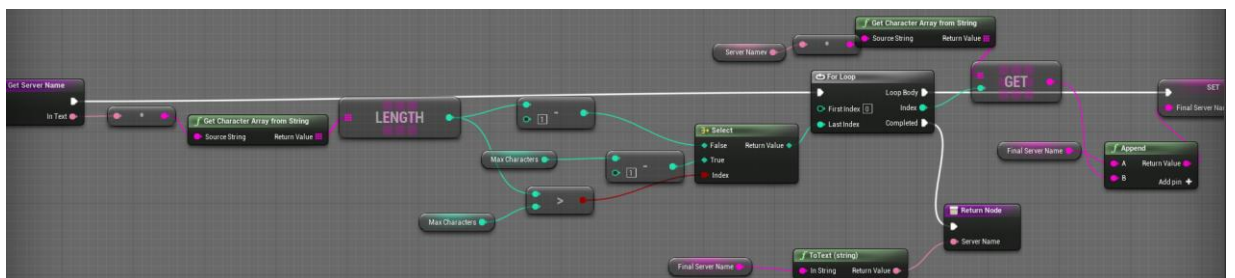


Рисунок 4.72 – Отримання інформації про сервер

Для постійного оновлення списку доступних серверів у відповідному вікні (рис. 4.73) окремо створено код, що наведено на рисунку 4.74.

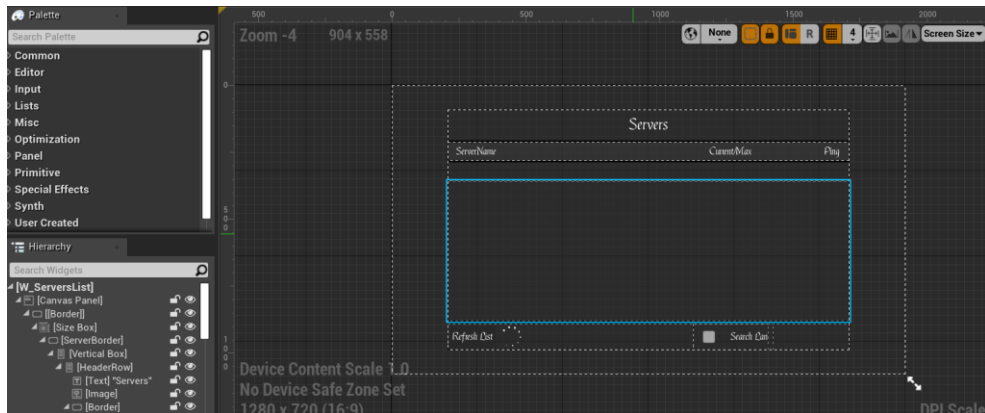


Рисунок 4.73 – Віджет вікна з серверами

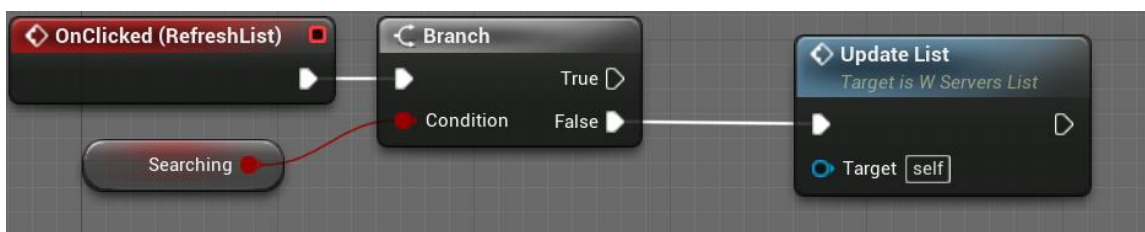


Рисунок 4.74 – Оновлення списку серверів

Щоб реалізувати пошук сесій, спочатку видаляємо весь старий список серверів (рис. 4.75), через функцію Find Sessions знаходимо сесії, котрі були нещодавно створені (рис. 4.76) та створюємо слоти з поточною інформацією про доступні сервери (рис. 4.77).

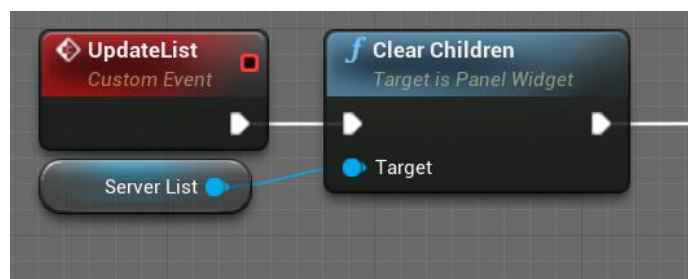


Рисунок 4.75 – Видалення старих записів

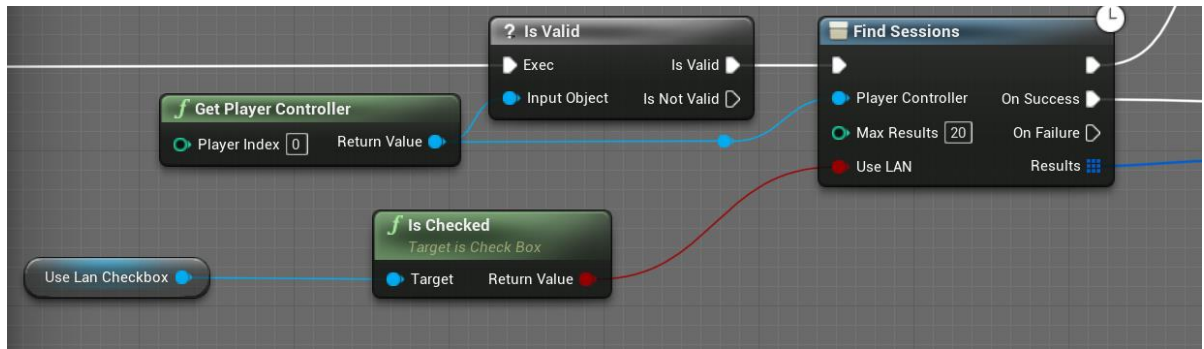


Рисунок 4.76 – Пошук сесії

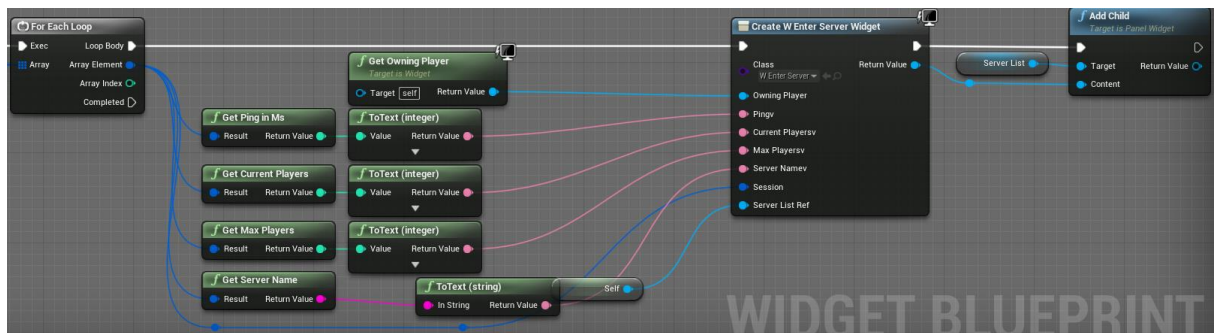


Рисунок 4.77 - Створення слотів з поточною інформацією

Тепер розглянемо, як відбувається налаштування серверу. Щоб створити нову сесію, необхідно вказати налаштування гри, а саме кількість гравців, мережеве з'єднання та ім'я серверу(рис. 4.78).

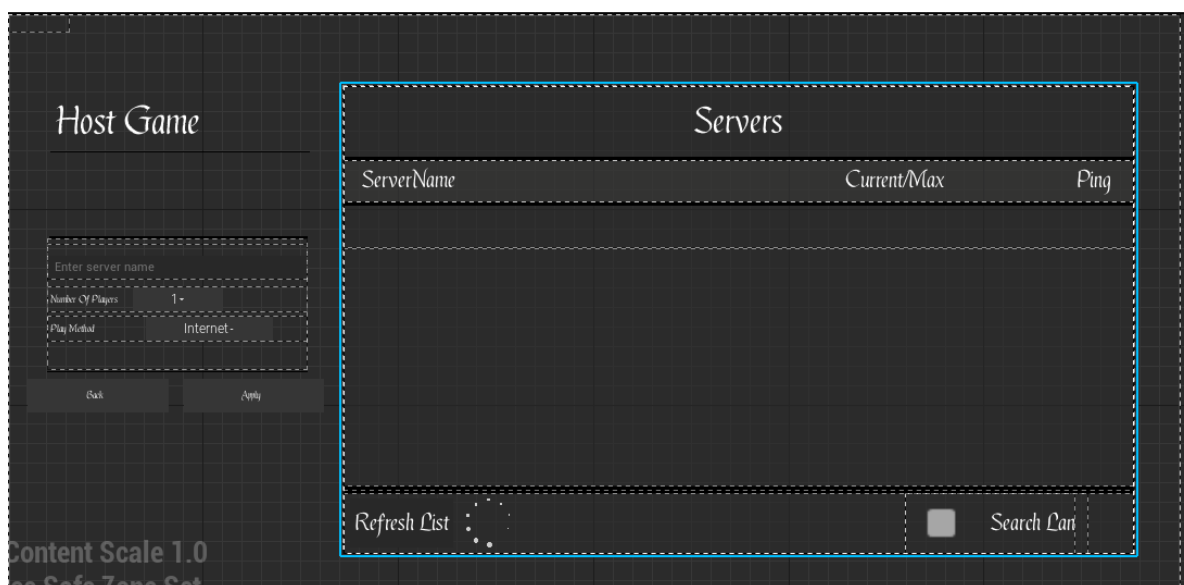


Рисунок 4.78 – Віджет налаштування серверу

Максимальна кількість гравців на одному сервері може бути 6 чоловік. Для вибору кількості гравців в одній сесії створено окремий блупринт (рис. 4.79).

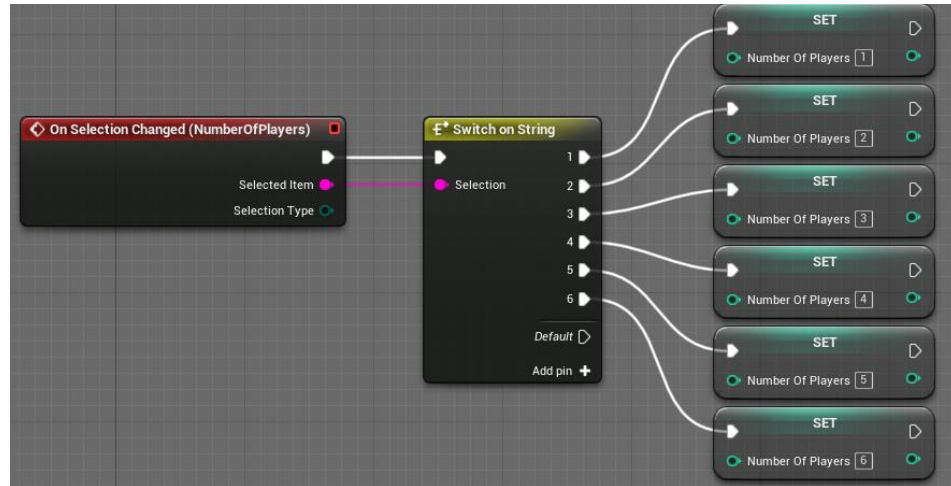


Рисунок 4.79 – Вибір кількості гравців

Також необхідно повторити перевірку на обраний метод гри, тобто через який мережевий спосіб відбуватиметься підключення (рис. 4.80).

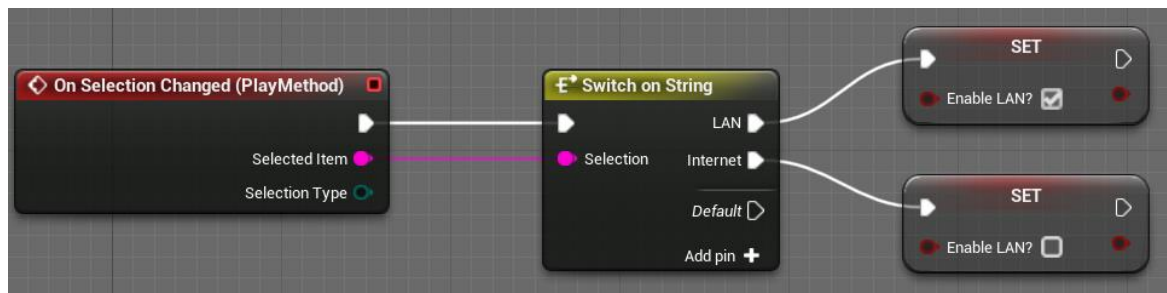


Рисунок 4.80 – Вибір мережевого підключення

Всі відповідні налаштування, вказані гравцем, необхідно передати до Game Instance, аби вся інформація про створений сервер відображалася надалі (рис. 4.81).

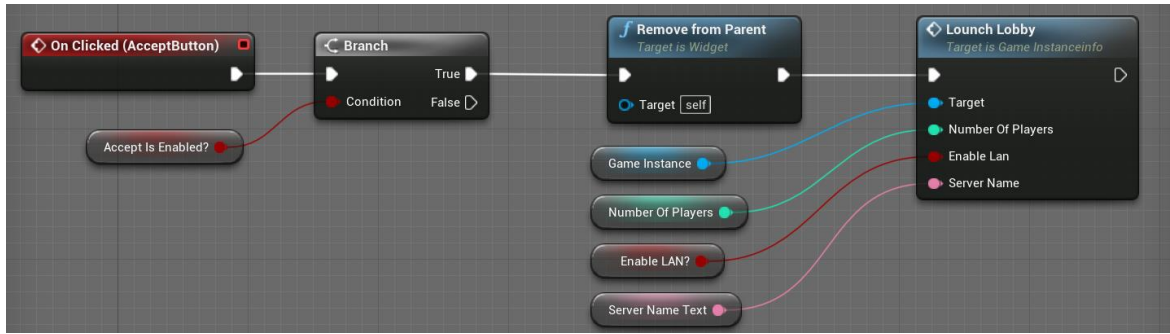


Рисунок 4.81 – Інформація про сервер

Також гравець-хост може встановити мапу, на якій будуть грати користувачі. Для цього створено окремий віджет (рис. 4.82) та відповідний блупринт (рис. 4.83).

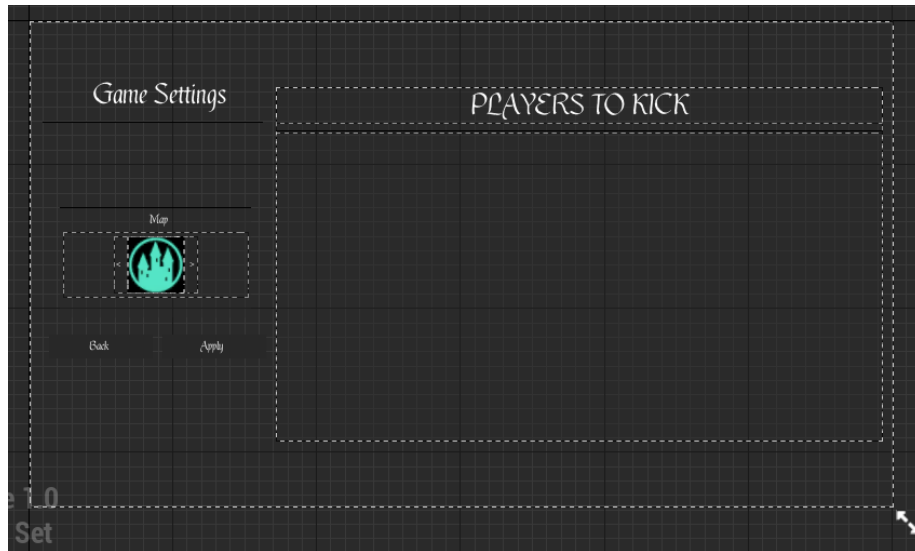


Рисунок 4.82 – Віджет вибору мапи

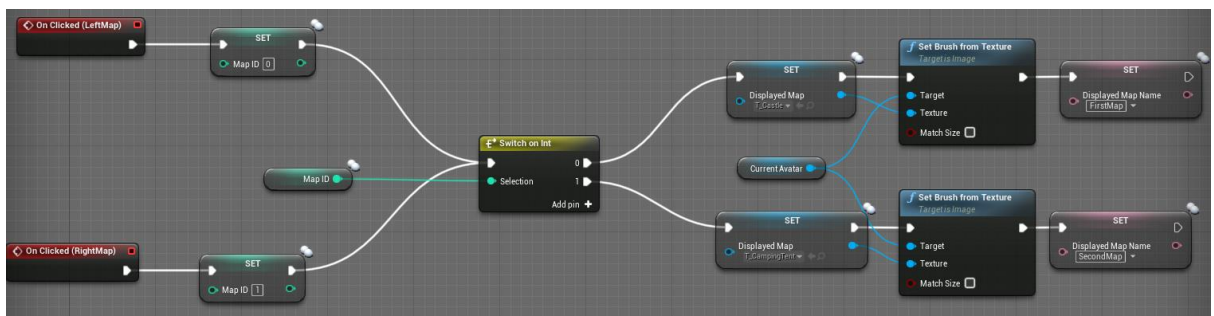


Рисунок 4.83 – Блупринт вибору мапи

За створення серверу відповідає функція LaunchLobby (рис. 4.84), яка не лише створює сервер, але й виводить помилку типу «FAILED TO CREATE SESSION. PLEASE TRY AGAIN.», якщо відбулася помилка з'єднання.

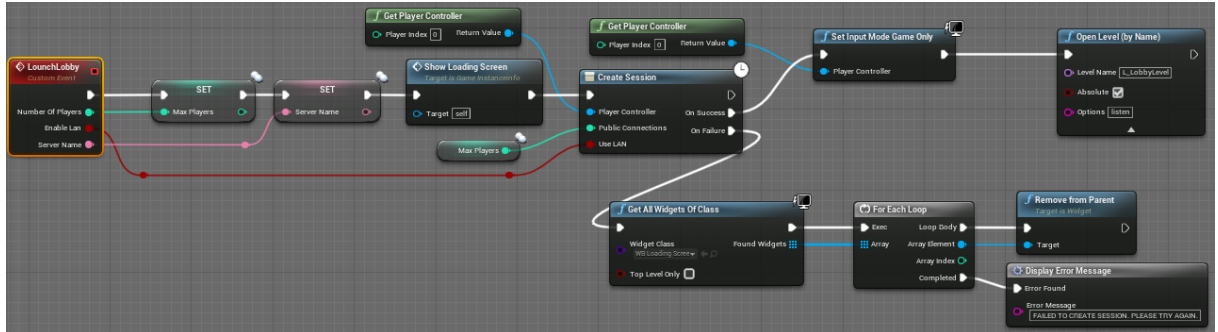


Рисунок 4.84 – Створення серверу

Після того як створено сервер з усіма можливими налаштуваннями, гравець-хост очікує на підключення інших гравців. Усі гравці, які підключаються до серверу, будуть відображатися у віджеті (рис. 4.85), а також їх статус (готові вони чи ні почати сесію). Код оновлення статусу гравців наведено на рисунку 4.86.

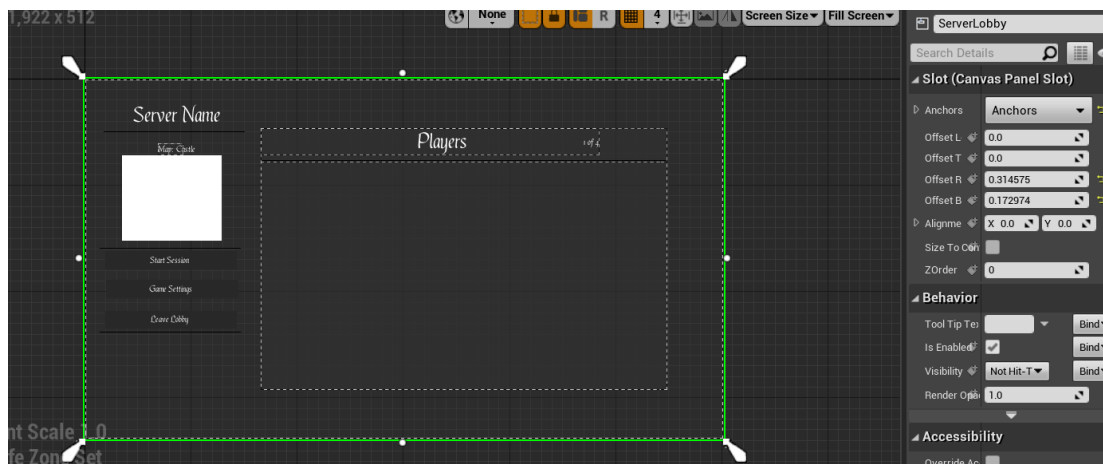


Рисунок 4.85 – Віджет підключення

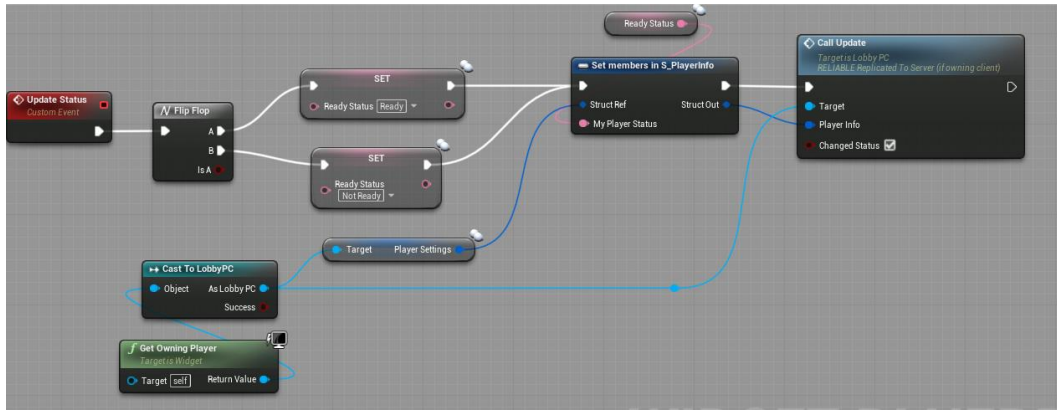


Рисунок 4.86 – Оновлення статусу гравців

Як тільки гравець-хост бачить готовність всіх підключених гравців, він запускає початок сесії (рис. 4.87).

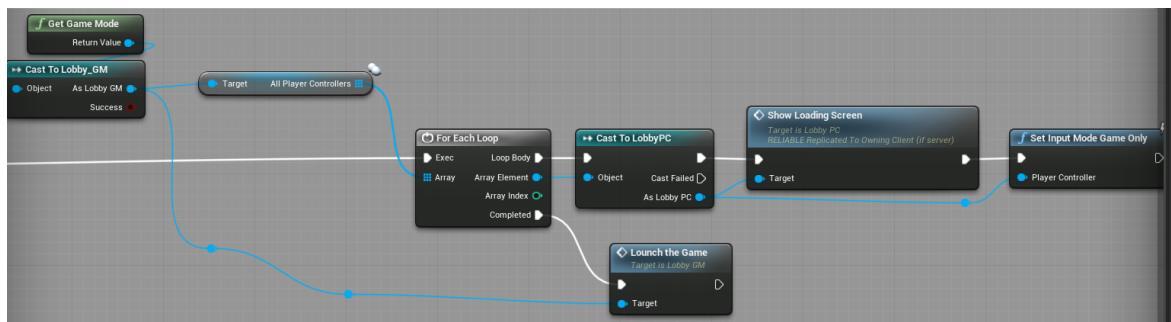


Рисунок 4.87 – Запуск сесії

Додатково була створена валідація, яка не дасть почати сесію, поки всі гравці не будуть готові. Кнопка, що відповідає за запуск даної сесії, буде недоступною, поки не вповняться всі задані підключення. Повний код початку сесії зображено на рисунку 4.88.

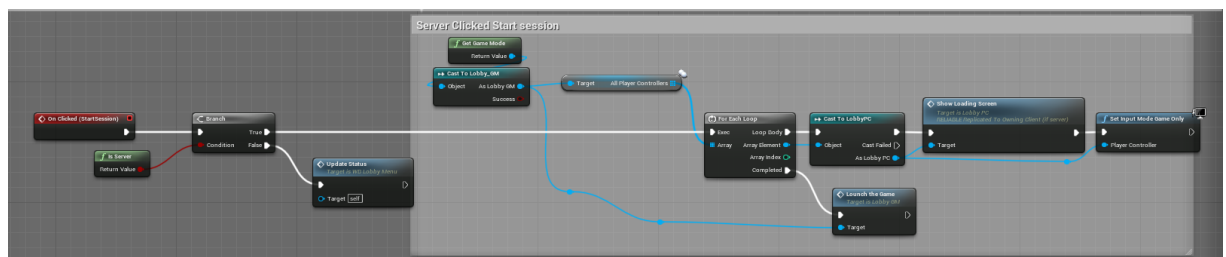


Рисунок 4.88 – Початок онлайн сесії

У підсумку, можна зробити висновок, що всі поставлені задачі проєкту виконано. Реалізовано:

- мережеве з'єднання гравців;
- збереження гри;
- можливість налаштування графіки;
- можливість одиночного режиму проходження квесту.

ВИСНОВКИ

Під час виконання дипломного проєкту було:

- проведено ретельний аналіз предметної області;
- розглянуто аналогічні додатки, основні їх недоліки та переваги;
- сформована головна задача проєкту та вимоги до нього.

Для реалізації поставленої мети обрано інструменти розробки, та на основі аналізу всіх існуючих додатків обрано найбільш практичні та найзручніші варіанти.

Під час структурного проєктування проєкту розроблено діаграму варіантів використання, а також проведено структурно-функціональне моделювання з необхідними декомпозиціями основних блоків.

Реалізацію проєкту розбито на основні етапи, котрі були виконані згідно календарного плану.

У додатку реалізовано такі основні можливості для користувача:

- можливість налаштування графіки;
- можливість мережевого з'єднання;
- можливість збереження прогресу;
- можливість грати в одиночному режимі.

В результаті виконання дипломного проєкту створений ігровий квест-додаток з підтримкою мережевої взаємодії гравців.

Практичне значення роботи полягає в створенні ігрового додатку, який буде не лише красивою грою, але й цікавою та пізнавальною. Додаток навчить гравців пошуку інформації за рахунок завдань на знання історичних дат, що є дуже корисним навиком.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Седых И. А. Индустрия компьютерных игр / И. А. Седых. // Национальный исследовательский университет. – 2020. – 74 с.
2. Ерман Г. Аналіз і тенденції індустрії мобільних ігор в 2020 році [Електронний ресурс] / Георгій Ерман. – 2020. – Режим доступу до ресурсу: <https://vc.ru/life/114169-analiz-i-tendencii-industrii-mobilnyh-igr-v-2020-godu>.
3. Галкин Д. В. Компьютерные игры как феномен современной культуры: опыт междисциплинарного исследования / Д. В. Галкин // Открытый междисциплинарный электронный журнал "Гуманитарная информатика". - Вып. 4. [Електронний ресурс]. - Режим доступу: <http://huminf.tsu.ru/e-jurnal/magazine/4/gal2.htm>.
4. Трішкіна, М. С. Комп'ютерні ігри як феномен сучасної культури: дипломна робота бакалавра / М. С. Трішкіна; наук. кер. О.Ю.Білянська; ОНУ ім. І.І. Мечникова, Філософ. ф-т, каф. культурології. – Одеса, 2017. – 47 с.
5. Илларионов, Г. А. Социально – философский анализ «традиционалистского проекта»: автореф. дисс. ... кандидата философских наук / Г. А. Илларионов. — Красноярск : СФУ, 2018.
6. Степаненко О. О. Конспект лекцій з дисципліни “Технології розробки мережевих додатків” для студентів спеціальності 121 "Інженерія програмного забезпечення" (всіх форм навчання) / уклад. О. О. Степаненко – Запоріжжя : ЗНТУ, 2017. – 81 с..
7. Czech Game Developers Association. Czech video game industry PC, console and mobile game developers in Czech Republic 2020. Режим доступу: https://gda.cz/wpcontent/uploads/2020/07/GDACZ_Study_2020.pdf.
8. Sedykh I. A. Computer game industry-2020 /National Research University Higher School of Economics, Development center. Режим доступу: <https://ict.moscow/research/industriia-kompiuternykh-igr-2020/>.

9. Сведения об игре "It Takes Two" и юридическая информация [Электронный ресурс] // EA Swiss Sarl. – 2021. – Режим доступа: https://store.playstation.com/ru-ru/product/EP0006-PPSA02343_00-ITTAKESTWORETAIL.
10. RAID: Shadow Legends [Электронный ресурс] // Plarium Global Ltd. – 2020. – Режим доступа: <https://plarium.com/ru/game/raid-shadow-legends/>.
11. CALL OF DUTY WARZONE. [Электронный ресурс] // Activision Publishing. – 2019. – Режим доступа: <https://www.callofduty.com/ru/warzone>.
12. Pradeep Mamgain. Autodesk 3ds Max 2021: A Detailed Guide to Modeling, Texturing, Lighting, and Rendering, 3rd Edition / Pradeep Mamgain., 2020. – 672 с.
13. Using Datasmith with 3ds Max [Электронный ресурс] // Unreal Engine documentation. – 2020. – Режим доступа до ресурсу: <https://docs.unrealengine.com/4.27/en-US/WorkingWithContent/Importing/Datasmith/SoftwareInteropGuides/3dsMax/>.
14. Robin Nichols. Mastering Adobe Photoshop Elements: Boost your image-editing skills using the latest tools and techniques in Adobe Photoshop Elements / Robin Nichols., 2021.
15. Sathesh PV. Unreal Engine 4 Game Development Essentials / Sathesh PV., 2016. – 266 с.
16. Game Development Projects with Unreal Engine: Learn to build your first games and bring your ideas to life using UE4 and C++ / Hammad Fozi, Gonçalo Marques, David Pereira, Devin Sherry., 2020. – 822 с.
17. Slate UI Framework (UE4/5) — standalone application [Электронный ресурс]. – 2021. – Режим доступа до ресурсу: <https://habr.com/ru/post/591135/>.
18. Database of everything on Steam [Электронный ресурс] // Steam – Режим доступа до ресурсу: <https://steamdb.info/>.

19. Розробка функціональної моделі. Методологія IDEF0. URL:https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6_2.
20. IDEF0 діаграма: приклади і правила побудови [Електронний ресурс] // ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ. – 2021. – Режим доступу до ресурсу: <https://ukr.kagutech.com/3929706-idef0-diagram-examples-and-construction-rules>.
21. МЕТОДОЛОГІЯ IDEF0 [Електронний ресурс] // Підручники для студентів онлайн (infostud.com.ua). – 2019. – Режим доступу до ресурсу: https://stud.com.ua/87184/ekonomika/metodologiya_idef0.
22. Шьонталер, Ф.А. Бизнес-процессы. Языки моделирования, методы, инструменты / Ф.А. Шьонталер, Фоссен Г.Д., Обервайс А.В., Карле Т.Е – М.: Альпина, 2019. - 264 с.
23. SHIANNE EDELMAYER. How to Create Textures in Photoshop CC [Електронний ресурс] / SHIANNE EDELMAYER. – 2019. – Режим доступу до ресурсу: <https://www.makeuseof.com/tag/create-textures-photoshop/>.
24. Fuse is a standalone 3D character creator [Електронний ресурс] // Міхамо – Режим доступу до ресурсу: <https://store.steampowered.com/app/257400/Fuse/>.
25. Unreal Engine. Saving and Loading Your Game [Електронний ресурс] / Unreal Engine. – 2020. – Режим доступу до ресурсу: <https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/SaveGame/>.
26. How to Make a Multiplayer Game in Unreal Engine [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://game-ace.com/blog/multiplayer-game-in-unreal/>.
27. Kupe Kupersmith. How to Use SMART Objectives to Clarify Your Business Analysis-2016 / Business Analysis For Dummies. Режим доступу: <https://www.dummies.com/article/business-careers-money/business/general-business/how-to-use-smart-objectives-to-clarify-your-business-analysis-162510> .

28. Mong M. What is a Work Breakdown Structure? [Электронный ресурс] / Matt Mong. – 2020. – Режим доступа до ресурсу: <https://www.adeaca.com/blog/faq-items/what-is-a-work-breakdown-structure/>.

29. OBS СТРУКТУРИ ПРОЕКТУ [Электронный ресурс] // SideToDo – Режим доступа до ресурсу: <https://present5.com/obs-strukturi-proektu-1-obs-struktura/>.

30. Марта Быстрова. Диаграмма Ганта — инструмент для тех, кто не любит срывать сроки [Электронный ресурс] / Марта Быстрова // Работа и учёба. – 2018. – Режим доступа до ресурсу: <https://lifehacker.ru/diagramma-ganta/>.

ДОДАТОК А

Планування робіт

А.1 Деталізація мети методом SMART

Метою проекту є розробка ігрового квест-додатку з підтримкою мережевої взаємодії гравців. Головна ціль даної роботи – створення квесту, який буде не лише цікавий, але й корисний, матиме підтримку онлайн-з'єднань з іншими гравцями, дозволить налаштовувати якість графіки під власні побажання та дізнатися цікаву інформацію. Деталізація мети проекту методом SMART [27] наведена в таблиці А.1.

Таблиця А.1 – Деталізація мети методом SMART

Specific (конкретна)	Розробити ігровий квест-додаток з підтримкою мережевої взаємодії гравців
Measurable (вимірювання)	Головним результатом даного проекту, адже він є не комерційним, є оцінка від дипломного керівника та комісії, що буде приймати дипломну роботу
Achievable (досяжна, узгоджена)	Досяжність цілі даного проекту – цілковито реальна, так як усі необхідні навички та знання, розробник проекту має, аби створити даний додаток, а також всі моменти погоджені з дипломним керівником.
Relevant (реалістична)	Все програмне та апаратне забезпечення, котре необхідне для реалізації даного проекту, а також доступ до Інтернет-мережі розробник має.
Time-framed (обмежена в часі)	Виконання проекту відбувається поступово, дотримуючись строків, котрі зазначені в календарному плані

А.2 Планування змісту робіт та структури виконавців

Найкращий інструмент для зручного планування робіт – це WBS, що розшифровується як Work Break Structure [28]. Саме за допомогою нього можна в зручному вигляді представити проект у вигляді структури, побудованої в ієрархічному порядку. Зазвичай верхня частина даної структури це сам проект, його назва, а вже нижче розташовані деталізовані результати роботи .

Під час планування розробки додатку було створено WBS для даного проекту, де перший рівень структури має назву – «Ігровий квест-додаток з підтримкою мережевої взаємодії ігроків», що в подальшому розбивається на декілька гілок декомпозиції. Повна WBS структура наведена на рисунку А.1.



Рисунок А.1 – WBS структура проекту

Після побудови WBS структури необхідно створити OBS – це структура виконавців, тобто хто та за який етап розробки буде відповідати [29]. Список виконавців, котрі брали участь у розробці даного проекту наведено на рисунку А.2.



Рисунок А.2 – OBS структура проекту

А.3 Побудова календарного графіку

Після розбиття проекту на основні етапи розробки, необхідно визначитися з датами, протягом яких відбуватиметься реалізація вже визначених кроків. Для зручного представлення даного плану вирішено побудувати його у вигляді діаграми Ганта [30]. Завдяки даній діаграмі часові проміжки розробки етапів проекту можна з легкістю відслідкувати. На рисунку А.3 наведена діаграма Ганта з усіма етапами розробки ігрового додатку.

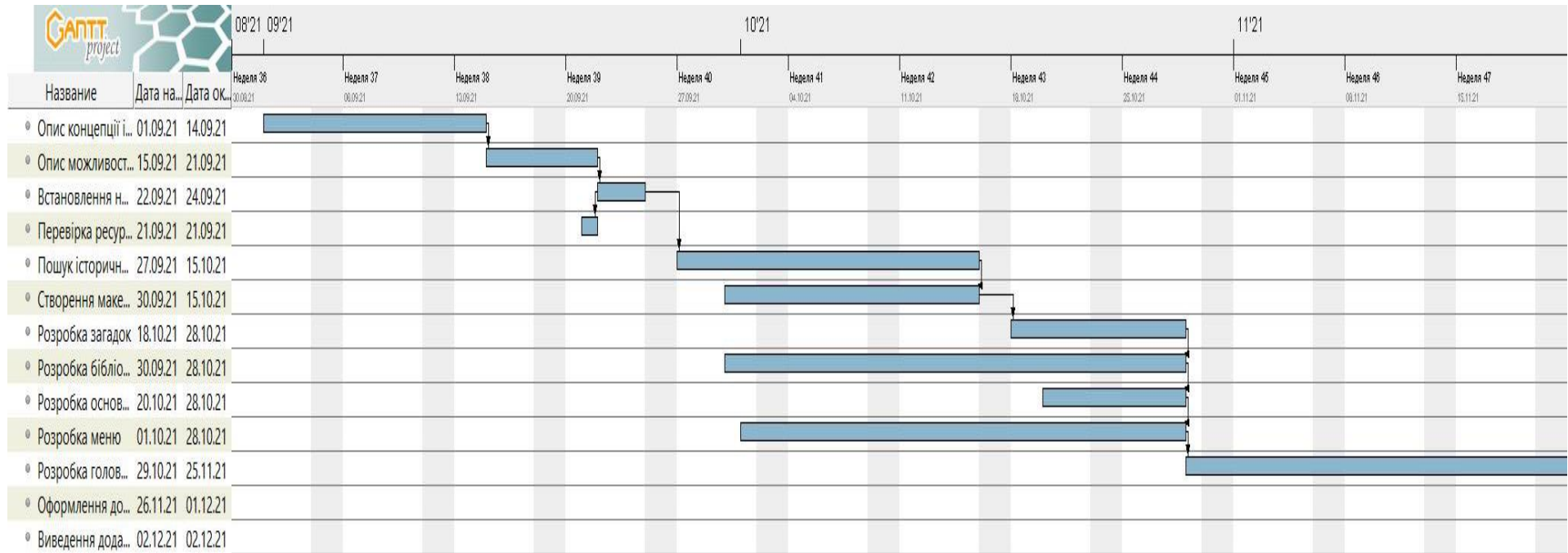


Рисунок А.3 – Діаграма Ганта виконання комплексного проєкту

А.4 Управління ризиками проекту

Кожен проект має свої певні ризики, адже причиною їх виникнення є невизначеність. Ризики можна класифікувати за двома основними видами, а саме відомі та невідомі, тобто ті, які вже оцінені або ж визначені, і ті, котрі неможливо спрогнозувати.

Після аналізу виявлено деякі ризики для даного проекту:

- R1 – Невідповідність у ТЗ;
- R2 – Нехтування календарним планом;
- R3 – Збої в апаратно-програмному забезпеченні;
- R4 – Тестування продукту неякісно;
- R5 – Певний людський фактор (погане самопочуття, сімейні обставини і т.д.);
- R6 – Цілі змінилися під час виконання проекту;
- R7 – Непередбачувані серйозні (форс-мажорні) обставини.

Таблиця А.2 – Класифікація ризиків

№	Назва ризику	Ймовірність	Величина втрат
1	с	3	2
2	Нехтування календарним планом	4	5
3	Збої в апаратно-програмному забезпеченні	3	5
4	Тестування продукту неякісно	1	3
5	Певний людський фактор	4	5
6	Цілі змінилися під час виконання проекту	1	4
7	Непередбачувані серйозні обставини	3	5

Дану інформацію було використано для побудови матриці ризику, котра представлена в таблиці А.3.

Таблиця А.3 – Матриця ризиків

Ймовірність виникнення	5						Неприпустимі ризики	
	4					R2, R5		
	3		R1				R3, R7	Виправдані ризики
	2							
	1			R4	R6			Допустимі ризики
		1	2	3	4	5		
	Величина втрат							

Визначимо рівні ризиків та ступінь їх дії.

Рівні можуть бути:

- допустимі $1 < R < 4$;
- виправдані $5 < R < 10$;
- недопустимі $11 < R < 25$.

Ступінь дії ризиків:

- ті, що можна проігнорувати $1 < R < 4$;
- незначні $5 < R < 8$;
- помірні $9 < R < 10$;
- істотні $11 < R < 16$;
- критичні $17 < R < 25$.

На основі цих даних виконана оцінка ступенів та рівнів для кожного ризику в проекті. Результати роботи представлені в таблиці А.4.

Таблиця А.4 – Визначення ступенів та рівнів ризиків

№	Ризик	Ймовірність	Величина втрат	Індекс ризику	Рівень ризику	Ступінь дії
1	Некоректно складене ТЗ	3	2	6	Виправданий	Незначний
2	Відставання від календарного плану	4	5	24	Неприпустимий	Істотний
3	Апаратно-програмні збої	3	5	15	Неприпустимий	Істотний
4	Низька якість тестування	1	3	3	Допустимий	Проігнорувати
5	Людський фактор	4	5	20	Неприпустимий	Істотний
6	Зміна вимог	1	4	4	Допустимий	Проігнорувати
7	Форс-мажорні обставини	3	5	15	Неприпустимий	Істотний

А.5 Формування бюджету проєкту

Завершальним етапом планування робіт є етап формування бюджету для даного проєкту.

В управлінні проєктами використовують декілька термінів, пов'язаних з фінансуванням проєкту: кошторис проєкту, бюджет проєкту, план фінансування продукту проєкту. При цьому їх можуть застосовувати стосовно:

- всього життєвого циклу проєкту;
- моменту отримання продукту проєкту;
- фази реалізації проєкту.

Кошторис продукту проєкту – це загальні майбутні витрати, які необхідні безпосередньо для створення продукту проєкту. Тобто це витрати на фінансування всіх робіт, передбачених WBS - структурою проєкту.

Бюджет продукту проєкту – це кошторис продукту проєкту, розподілений в часі на основі календарного плану реалізації робіт або за окремими WBS елементами. План фінансування – це кошторис продукту проєкту в розрізі основних джерел фінансування робіт з проєкту. Розрахуємо бюджет заробітної плати.

На початку реалізації даного процесу необхідно визначити учасників проєктів, які безпосередньо приймали участь в етапах реалізації задачі. На основі структури OBS було виділено таких працівників\виконавців:

- розробник;
- менеджер проєкту;
- консультант проєкту;
- тестувальник.

В таблиці А.6 приведена заробітна плата кожного учасника проєкту з урахуванням різної кількості робочих годин кожного з них.

Таблиця А.6 – Заробітна плата учасників проєкту

Посада	За 1 год	Робочих годин	Заробітна плата
Розробник	150	120	18000
Менеджер проєкту	145	110	15950
Консультант проєкту	130	105	13650
Тестувальник	90	50	4500

Бюджет заробітної плати складає 52100 грн.