

Міністерство освіти і науки України
Харківський національний університет Радіоелектроніки

Факультет _____ Комп'ютерних наук _____

(повна назва)

Кафедра _____ Системотехніки _____

(повна назва)

УДК 004.946

АТЕСТАЦІЙНА РОБОТА

Пояснювальна записка

_____ другий (магістерський) _____

(рівень вищої освіти)

_____ ГЮИК.502640.012 ПЗ _____

(позначення документа)

Методи та технології розробки навчальних програм із застосуванням
віртуальної реальності

(Тема)

Виконав: студент 2 курсу, групи СПРМ-18-2 _____

спеціальності _____ 122 Комп'ютерні науки _____

(код і повна назва спеціальності)

освітньо-наукова програма Системне проектування _____

(повна назва освітньої програми)

_____ Орлов О.К _____

(прізвище, ініціали)

Керівник _____ проф к.т.н Калита Н.І _____

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри системотехніки _____

(підпис)

_____ Гребеннік І.В _____

(прізвище, ініціали)

2020 р.

Харківський національний університет радіоелектроніки

Факультет _____ *комп'ютерних наук* _____

Кафедра _____ *системотехніки* _____

Рівень вищої освіти _____ *другий (магістерський)* _____

Спеціальність _____ *122 Комп'ютерні науки* _____
(код і повна назва)

освітньо-наукова програма _____ *Системне проектування* _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові _____ *Орлову Олександрю Костянтиновичу* _____
(прізвище, ім'я, по батькові)

1. Тема роботи Методи та технології розробки навчальних програм із застосуванням віртуальної реальності

затверджена наказом по університету від «22» травня 2020р. № 477

2. Термін подання студентом роботи 21 травня 2020р

3. Вихідні дані до роботи Навчальна програма типу "симулятор", вимоги до навчальної гри, функція: навчання. Використовувати ОС Microsoft Windows, HTC VIVE, середовище програмування Visual Studio, Unreal Engine 4

4. Перелік питань, що потрібно опрацювати в роботі: 4.1 Вступ, 4.2 Аналіз предметної області, 4.2.1 Поняття віртуальної реальності, 4.3 Вимоги до навчальних програм, 4.4 Огляд програмних засобів жанру «симулятор», 4.5 Постановка задачі дослідження та обґрунтування проектних рішень, 4.5.1 Аналіз проблеми ігор аналогів, 4.5.2 Постановка задачі, 4.5.3 Опис gameplay та прийнятих програмних рішень, 4.5.4 Методи та технології реалізації віртуальної реальності, 4.6 Архітектура та проектування програмного забезпечення віртуальної реальності, 4.6.1 UML моделювання, 4.6.2 Створення інтерфейсу, графіки та анімації, 4.6.3 Реалізація ігрових механік, 4.6.4 Blueprints, 4.6.5 Реалізація інтерфейсу HUD, 4.7 Етапи проектування гри та створення ігрових рівнів, 4.7.1 Можливості розробки гри в середовищі Unreal Engine 4, 4.7.2 Крива складності, 4.8 Аналіз результатів досліджень, 4.8.1 Проведення аналізу даних, 4.8.2 Аналіз результатів, 4.9 Висновок

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) _ порівняння ігрових аналогів, порівняння продуктів-аналогів за використанням методів vr, uml-моделювання системи, схеми візуального проектування, візуальний скриптинг blueprint, аналіз фактору можливостей, порівняння ігрових додатків, середні значення отриманих результатів

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		Підпис	Дата
Спецчастина	Проф. к.т.н Калита Н.І		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів	Примітка
1.	Аналіз предметної області дослідження	15 березня 2020 р.	
2.	Постановка задачі досліджень та обґрунтування проектних рішень	05 квітня 2020 р.	
3.	Архітектура та проектування програмного забезпечення віртуальної реальності	20 квітня 2020 р.	
4.	Етапи проектування гри та створення ігрових рівнів	01 травня 2020 р.	
5.	Аналіз результатів дослідження	8 травня 2020 р.	
6.	Підготовка презентації та доповіді	14 травня 2020 р.	
7.	Подання атестаційної роботи у ЕК до захисту	22 травня 2020 р.	

Дата видачі завдання 30 березня _____ 2020 р.

Студент _____  _____

(підпис)

Керівник роботи _____ проф. к.т.н. Калита Н.І. _____

(підпис)

(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до атестаційної роботи магістра, 97 с., 42 рис., 6 табл., 26 джерел інформації.

ТЕХНОЛОГІЇ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ, НАЧАЛЬНІ ПРОГРАМИ, РОЗВИВАЮЧІ ІГРИ, UNREALENGINE 4, C++, BLUEPRINTS, VIRTUALREALITY, GAMESIMULATOR, VRІГРИ, AI.

Об'єктом дослідження є віртуальна реальність.

Предметом дослідження є методи та технології розробки віртуальної реальності.

Мета дослідження – визначення особливостей застосування віртуальних технологій у навчальному процесі, аналіз існуючих прикладів програмних засобів з віртуальною реальністю, створення і застосування віртуальних технологій у навчанні.

Методи дослідження – системний підхід, технології проектування великих програмних систем, сучасні парадигми програмування і моделювання складних систем.

У роботі виконано аналіз сучасних методів розробки та засобів реалізації віртуальної реальності у навчальних програмах, визначені вимоги до віртуальної реальності, які забезпечують заохочення та успішність навчання. Апробація результатів дослідження виконана при розробці програмного засобу жанру гри, який створено на платформі UnrealEngine 4, мовою C++, blueprints з урахуванням виконаних досліджень.

Результати досліджень магістерської атестаційної роботи можуть бути використані при розробці навчальних програм будь-якого спрямування з використанням віртуальної реальності, що значно підвищує ефективність навчання.

ABSTRACT

Explanatory note to the certification work of the master's thesis, 97 pp., 42 figs., 6 tables, 26 titles.

VIRTUAL REALITY TECHNOLOGIES, INITIAL PROGRAMS, DEVELOPING GAMES, UNREALENGINE 4, C ++, BLUEPRINTS, VIRTUALREALITY, GAMESIMULATOR, VR GAMES, AI.

The object of research is virtual reality.

The subject of research is the methods and technologies of virtual reality development.

The purpose of the study is to determine the features of the use of virtual technologies in the educational process, analysis of existing examples of software with virtual reality, creation and application of virtual technologies in education.

Research methods - systems approach, technologies for designing large software systems, modern paradigms of programming and modeling of complex systems.

The analysis of modern methods of development and means of realization of virtual reality in educational programs is carried out in the work, requirements to virtual reality which provide encouragement and success of training are defined. Approbation of the research results was performed during the development of the game genre software, which was created on the UnrealEngine 4 platform, in C ++, blueprints, taking into account the performed research.

The results of research of master's attestation work can be used in the development of curricula in any field with the use of virtual reality, which significantly increases the effectiveness of training.

ЗМІСТ

ВСТУП.....	8
1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	10
1.1 Поняття віртуальної реальності.....	10
1.1.1 Віртуальна реальність у початковому процесі	15
1.1.2 Використання віртуальної реальності у симулятора.....	16
1.2 Вимоги до навчальних програм і симуляторів та їх структура	18
1.3 Огляд програмних засобів жанру «симулятор».....	22
1.3.1 Crazy Fishing	22
1.3.2 Fancy Fishing VR.....	23
1.3.3 Amigo Fishing.....	24
2 ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ.....	26
2.1 Аналіз проблеми ігор аналогів	26
2.2 Постановка задачі.....	28
2.3 Опис gameplay та прийнятих програмних рішень.....	29
2.4 Методи та технології реалізації віртуальної реальності	30
3 ЗАРХІТЕКТУРА ТА ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ.....	38
3.1 UML-моделювання системи	38
3.1.1 Use Case diagram.....	38
3.1.2 Class diagram	39
3.1.3 Sequence diagram.....	41
3.2 Створення UI/UX дизайну, графіки та анімації.....	42
3.3 Реалізація ігрових механік	43
3.4 Blueprints	44
3.4.1 Взаємодія різних елементів рівня з водою.....	49
3.4.2 AI та поведінка риби	49
3.4.3 Логіка та поведінка вудки	51
3.4.4 Алгоритм генерації риби.....	52
3.4.5 Реалізація управління персонажем через контролер.....	52
3.4.6 Створення телепорту для руху персонажа.....	54

3.5 Реалізація інтерфейсу HUD	56
4 ЕТАПИ ПРОЕКТУВАННЯ ГРИ ТА СТВОРЕННЯ ІГРОВИХ РІВНІВ	59
4.1 Можливості розробки гри в середовищі Unreal Engine 4.....	60
4.2 Крива складності	62
4.3 Перший рівень	63
4.4 Другий рівень	63
4.5 Третій рівень.....	63
4.6 Четвертий рівень	64
5 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ.....	65
5.1 Проведення аналізу даних	65
5.2 Аналіз результатів	67
ВИСНОВКИ	71
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	73
Додаток А. Графічні матеріали атестаційної роботи	74
Додаток Б. Текст програми.....	78

ВСТУП

У сучасному світі нові моделі навчання спрямовані на те, щоб навчити людину користуватися не тільки теоретичною інформацією, а й використовувати здобуті знання та навички. Потрібно не лише їх формувати, але й розуміти навіщо і як люди навчаються, користуючись при цьому різноманітними гаджетами. На перший план сьогодні у навчанні виходить завдання — зацікавити та мотивувати до навчання, до пізнання нового, через реальні приклади.

Інтерактивні комп'ютерні ігри засновані на взаємодії гравця з створюваними ними віртуальним світом [1]. Багато з них засновані на ототожненні гравця з персонажем гри, видимим або імовірним. Існує цілий клас ігор-симуляторів будь-якого роду діяльності. Поширені авіасимулятори, автосимулятори, різного роду економічні і спортивні симулятори, ігровий світ яких моделює важливі для даного роду фізичні закони, створюючи наближену до реальності модель.

Об'єкти віртуальної реальності зазвичай поведуть себе близько до поведінки аналогічних об'єктів матеріальної реальності. Користувач може впливати на ці об'єкти згідно з реальними законами фізики (гравітація, властивості води, зіткнення з предметами, відображення і т.д.). Однак часто в розважальних цілях користувачам віртуальних світів дозволяється більше, ніж можливо в реальному житті (наприклад: літати, створювати будь-які предмети і т.д.) [1].

Актуальність роботи обумовлена тим, що у наш час умови та спосіб життя відрізняється новими характеристиками інформаційного суспільства, і; це призводить, зокрема, до змінення парадигми освіти, до нового розуміння того, як люди можуть навчатися. Більш актуальною стає неформальна освіта, яку люди отримують поза інституціональних структур, використовуючи відкриті освітні ресурси та платформи електронного навчання.

На наш час вже накопичений достатній досвід як у розробці навчальних програм, так і віртуальної реальності, але потреби користувачів програмного забезпечення вимагають подальшого його вдосконалення.

Метою магістерської атестаційної роботи є дослідження методів та технологій розробки навчальних програм із застосуванням віртуальної реальності.

Для досягнення поставленої мети необхідно:

- визначити основні проблеми, які мають місце у продуктах-аналогах віртуальної реальності;
- розглянути сучасні середовища, інструментарій та засоби розробки застосунків віртуальної реальності;
- проаналізувати сучасний стан розвитку навчальних програм, їх типи, засоби реалізації;
- сформулювати постановку задачі дослідження; – розробити навчальну програму із застосуванням віртуальної реальності.

Об'єктом дослідження є віртуальна реальність.

Предметом дослідження є методи та технології розробки віртуальної реальності.

Методи дослідження – системний аналіз продуктів-аналогів різних провідних компаній у сфері віртуальних освітніх та ігрових середовищ, технології проектування великих програмних систем, сучасні парадигми програмування і моделювання складних систем.

Наукова новизна одержаних результатів. Порівняно продуктами- аналогами, у ігровому застосунку шляхом використання механік, елементів та штучного інтелекту зростаючої складності забезпечено ненав'язливе навчання на початкових рівнях, демонстрація усіх механік відповідно рівню, а на кінцевому рівні використовуються одразу усі. Завдяки поступовому ускладненню рівнів користувач має змогу навчитися усьому, що пропонується, та використати свої знання та навички для просування на більш складні рівні.

Результати досліджень магістерської атестаційної роботи можуть бути використані при розробці навчальних програм будь-якого спрямування з використанням віртуальної реальності, що значно підвищує ефективність навчання. На даний час ведеться впровадження розробленого програмного застосунку на Viverport майданчик. Після надання дозволу для розміщення розробки, усі бажаючі зможуть використовувати гру Fisherman.

Публікація. Матеріали досліджень доповідалися на тему «Створення алгоритму поведінки риб на основі реальних умов рибалки» в 24-му Міжнародному молодіжному форумі «Радіоелектроніка і молодь у XXI столітті» [2].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ

1.1 Поняття віртуальної реальності

Перш за все необхідно дати визначення терміну віртуальна реальність. Віртуальний можна визначити як «існуючий в сутності або впливі, але не фактично» [1]. Таке використання було застосовано до ранніх концепцій в обчисленнях; наприклад, коли комп'ютерна система потребує більшої кількості оперативної пам'яті (первинного сховища), ніж їй доступно, пам'ять розширюється віртуально за допомогою дискового сховища (вторинне та дешевше сховище). Отримана в результаті, як може здатися, збільшена оперативна пам'ять називається віртуальною пам'яттю.

Те, що розуміється під реальністю, є більш складним. Для заданих цілей можемо означити реальність як «місце, яке існує, і яке ми можемо відчувати»[1].

Ключовими елементами віртуальної реальності є:

- віртуальний світ – опис колекції об'єктів у просторі, а також правила та відносини, що регулюють ці об'єкти;

- занурення – відчуття перебування у середовищі; може бути суто психічним станом або може здійснюватися фізичними засобами; фізичне занурення є визначальною характеристикою віртуальної реальності; психічне занурення, мабуть, є метою більшості розробників медіа;

- сенсорний зворотний зв'язок (відповідь на введення користувача) – система віртуальної реальності забезпечує безпосередній сенсорний зворотній зв'язок користувачам з урахуванням їх фізичного положення;

- інтерактивність – відповідь на дії користувача, його здатність впливати на віртуальний світ та фізично рухатися у ньому.

Враховуючи це, можемо дати визначення віртуальної реальності. Віртуальна реальність – середовище, яке складається з інтерактивного комп'ютерного моделювання, яке враховує положення користувача та його дії, та замінює або збільшує зворотний зв'язок з одним або декількома органами чуття, що дає змогу користувачу відчувати психічне занурення або присутність у віртуальному світі [1].

Початком розвитку технологій Virtual Reality (VR) вважається створення Мортоном Хейлінгом у 1957 році пристрою під назвою «Сенсоррама», який є першим у світі віртуальним симулятором. Цей громіздкий апарат дозволяв глядачеві відчувати досвід занурення у віртуальну реальність, наприклад, покататися на мотоциклі. Пристрій впливав на всі основні органи чуття, і окрім зображення та звуку користувач відчував вібрацію від сидіння та запахи. Незважаючи на провал проекту «Сенсоррама» є прототипом сучасних VR пристроїв.

У 1965 році Айвен Сазерленд запропонував остаточне рішення віртуальної реальності [3]: концепцію конструювання штучного світу, що включає в себе інтерактивну графіку, силовий зворотній зв'язок, звук, запах та смак.

Першою апаратною реалізацією VR системи на основі цієї концепції є «The Sword of Damocles». Айвен Сазерленд сконструював пристрій, який вважається першим дисплеєм, що встановлюється на голову (Head Mounted Display або HMD) з відповідним відстеженням голови. Він підтримував стереозображення відповідне до позиції та орієнтації голови користувача.

У 1975 році Майрон Крюгер створив штучну реальність, яка називалася «VIDEOPLACE». У цій системі камери захоплювали силуети користувачів, які потім проектувалися на великий екран. Учасники могли взаємодіяти один з одним завдяки технологіям обробки зображень, що визначали їхні позиції в двовимірному просторі екрану [3].

Томас Фернесс розробив у 1982 році симулятор візуально з'єднаних бортових систем (Visually Coupled Airborne Systems Simulator або VCASS) – передовий симулятор польоту.

У 1984 році в NASA Ames готовою технологією було сконструйовано дисплей віртуального візуального середовища (Virtual Visual Environment Display або VIVED) – стереоскопічний монохромний HMD.

Далі компанія VPL виробляє популярні Data Glove (1985) і Eye phone HMD (1988) – перші комерційно доступні VR пристрої.

Комерціалізований у 1989 році пристрій «BOOM» являв собою маленьку коробку, що містив два монітори на основі електронно-променевої трубки (CRT). Користувач тримав коробку на рівні очей, а механічна рука вимірювала позицію та орієнтацію коробки.

Презентований у 1992 році «CAVE» був системою віртуальної реальності та наукової візуалізації. Замість того, щоб використовувати HMD, він проектував стереоскопічні зображення на стінах кімнати, а користувач надягав спеціальні окуляри. Такий підхід забезпечував чудову якість зображень і більш широке поле зору в порівнянні з HMD-системами [3].

На початку 1990-х років розвиток у сфері VR став більш стрімким і термін «віртуальна реальність» сам по собі став надзвичайно популярним, і вже в наш час технології віртуальної реальності широко застосовуються у різних сферах людської діяльності.

В останні кілька років технології VR стають все більш популярними в розробці ігрового програмного забезпечення. VR є відносно новою галуззю у цій сфері, тому, на відміну від інших платформ, ринок додатків під VR технології не є перенасиченим, що робить дану платформу цікавою для освоєння, особливо для нових та невеликих команд розробників, які ще не мають достатньої кількості досвіду та фінансування. Найбільш розповсюдженими є комп'ютерні та мобільні VR.

Серед комп'ютерних VR можна виділити наступні гарнітури віртуальної реальності(шоломи):

–HTC Vive

–Oculus Rift

До мобільних VR відносяться:

–Gear VR

–Google Cardboard

Для мобільних VR базовим пристроєм може бути смартфон з підтримкою віртуальної реальності. Перевагою мобільних VR є відносно невелика ціна та мобільність. Комп'ютерна VR працює на базі персонального комп'ютера, що надає більше можливостей для розробки програмного забезпечення та взаємодії користувача з ним.

Розробка програмного забезпечення для віртуальної реальності підтримується гральними рушіями, серед яких найбільш популярними є Unity та Unreal Engine 4. Unity вважається інтуїтивно зрозумілим для нових користувачів, в той час як Unreal Engine 4 краще оптимізований з точки зору обчислень.

Для розробки гри-симулятора найбільше підходить комп'ютерна система віртуальної реальності HTC Vive через наявність необхідних контролерів та

потужностей, а також ігровий рушій Unreal Engine 4 через кращу оптимізацію серед аналогів.

Залежно від способу і режиму взаємодії з користувачем, існують такі системи VR [4]:

1) системи типу «Вікно в світ» (Desktop VR) – користувач сидить перед монітором комп'ютера і взаємодіє з ним за допомогою комп'ютерної миші;

2) відео накладання – за допомогою відеокамери силует користувача накладається на двовимірне зображення, яке створюється комп'ютером, у результаті чого користувач дивиться на екран, бачить свій силует у кіберпросторі, і взаємодіє з віртуальним світом;

3) системи занурення – повністю занурюють користувача у віртуальний світ, створюючи при цьому відчуття присутності; можуть включати взаємодію між двома або більше аватарами, контрольованими людьми.

4) системи дистанційної присутності – з'єднання віддалених сенсорів, розташованих на будь-якому об'єкті в реальному світі з оператором-людиною;

5) змішана реальність – об'єднання систем дистанційної присутності й системи, що ґрунтується на віртуальній реальності, які використовують поєднання реального світу, що розглядається безпосередньо або через камеру, з контентом, накладеним комп'ютером.

Незважаючи на те, що ці системи є відносно новими, вони мають потенціал для підготовки студентів у галузі медицини, техніки, іноземних мов.

Виділяються п'ять основних переваг застосування технологій VR в освіті [4]:

1. Наочність. Використовуючи 3D-графіку, можна деталізовано показати хімічні процеси аж до атомного рівня. Причому ніщо не забороняє заглибитися ще далі і показати, як всередині самого атома відбувається поділ ядра перед ядерним вибухом. Віртуальна реальність здатна не тільки дати відомості про саме явище, а й продемонструвати його з будь-яким ступенем деталізації.

2. Безпека. Операція на серці, управління надшвидкісним поїздом, космічним кораблем, техніка безпеки під час пожежі – можна занурити глядача в будь-які з цих обставин без найменших загроз для життя.

3. Залучення. Віртуальна реальність дозволяє змінювати сценарії, впливати на хід експерименту або вирішувати задачу в ігровій та доступній для розуміння

формі. Під час віртуального уроку можна побачити світ минулого очима історичного персонажа або вирушити в подорож.

4. Фокусування. Віртуальний світ, який оточить глядача з усіх боків на всі 360 градусів, дозволить повністю зосередитися на матеріалі і не відволікатися на зовнішні подразники.

5. Віртуальні уроки. Сприйняття від першої особи і відчуття своєї присутності в намальованому світі – одна з головних особливостей віртуальної реальності.

Різновиди елементів віртуальної реальності (кіберпростір, тривимірний графік, симуляція, 3D-тур, віртуальна панорама, доповнена реальність) дозволяють:

- потрапити в інтерактивне мовне середовище, яке функціонує за допомогою комп'ютерних систем;
- побудувати тривимірну модель міста, штату, вулиці, додати написи з підказками;
- подорожувати, використовуючи спеціальні переходи, з одного приміщення в інше, орієнтуючись за картою (навігатором);
- переміщатися у віртуальному просторі з ілюзією присутності в тривимірному просторі.

Використовуючи окуляри і шоломи VR, студенти можуть занурюватися в атмосферу симулятора, гри, навчальної програми і т. д. На даний момент найбільш перспективним напрямком використання віртуальних програм є відтворення соціокультурної реальності країн мови, що вивчається, що дозволяє студенту стати учасником мовної, культурної, соціокультурної або комунікативної ситуації. Інструменти віртуальної реальності істотно впливають не лише на аудіовізуальне сприйняття, але й дають можливість інтерактивної взаємодії, що сприяє ефективнішому засвоєнню мовного матеріалу.

У віртуальній реальності можна спробувати робити щось багаторазово, не боячись помилитися, і саме тому такою високою є ефективність програм, заснованих на використанні технологій віртуальної реальності. Крім того, зниження цінності об'єктів віртуального світу внаслідок того, що всі вони є лише ілюзією, спонукає людей творити, пробувати, досліджувати.

Отже, віртуальна реальність може здійснювати дійсний, актуальний вплив на реальність константну, реальність даного нам світу. Перенесення чогось,

наприклад, знань, умінь і навичок чи нового досвіду з однієї реальності в іншу може відігравати значну роль у розвитку людини. Варто зазначити, що така технологія не є альтернативою академічному навчанню. Її слід розглядати як одну з багатьох ігрових методик в рамках навчальної програми та використовувати лише для підвищення ефективності та активізації учбового процесу.

1.1.1 Віртуальна реальність у навчальному процесі

У навчальному процесі вищих навчальних закладів навчальні інформаційні системи, що використовують комп'ютерні технології, відіграють дуже важливу роль, оскільки повне залучення в навчальний процес за допомогою спостереження за максимально реалістичною картинкою підвищує мотивацію й успіхи в отриманні знань, стимулює мозкову діяльність. В якості одного з перспективних освітніх методів сучасні інформаційні технології пропонують нове освітнє середовище - віртуальну реальність (VR), яка моделюється комп'ютером і розглядається в якості особливого інформаційного середовища, в якому всі об'єкти представлені в трьох вимірах. Відмінною рисою цього середовища є зміна зображень в режимі реального часу і переживання ефекту присутності. VR імітує як вплив, так і реакції на цей вплив [4].

За останні п'ять років технології віртуальної реальності розвинулися від сумнівно перспективних до повсюдно використовуваних і впроваджуваних. Незважаючи на поширення даної технології в якості одного з експериментальних методів навчання, а також засобів і технологій навчання одночасно, реальних досліджень віртуальної реальності в педагогіці здійснюється вкрай мало. Це можна пояснити складністю, високими матеріальними витратами даних досліджень, причому не лише в нашій країні, а й за кордоном. Більше того, сучасна система загальної та професійної освіти демонструє посилене протиріччя між використанням педагогічних технологій на основі віртуальної реальності та інерцією колишніх стереотипів освітньої практики [5].

Однак, попри очевидні переваги застосування технологій віртуальної реальності в освіті, на сьогодні є чимало потенційних проблем з технологічної точки зору [6].

По-перше, будь-яка дисципліна є досить об'ємною, що вимагає значних ресурсів для створення контенту на кожен тему заняття – у вигляді повного курсу

або десятків і сотень невеликих додатків. Компанії, які створюватимуть такі матеріали, повинні бути готові займатися розробкою досить тривалий час без можливості її окупати до виходу повноцінних наборів занять.

По-друге, уразі дистанційного навчання навантаження з покупки пристрою віртуальної реальності лягає на користувача, або цим пристроєм може бути його телефон. Але освітнім установам необхідно буде купувати комплекти обладнання для аудиторій, в яких проходять заняття, що також вимагає істотних інвестицій.

По-третє, віртуальна реальність, як і будь-яка технологія, вимагає використання своєї, специфічної мови. Важливо знайти правильні інструменти для того, щоб зробити контент наочним. Крім того, складно буде оновити вже наявні освітні програми [6].

1.1.2 Використання віртуальної реальності у симуляторах

Симуляція – це розміщення людей в «фіктивні, що імітують реальні» ситуації для навчання або одержання оцінки виконаної роботи, інакше це навчання дією або в дії. Цей режим створення проекту не записує рухів маніпулятора миші. Симуляцію в освітньому процесі розрізняють за такими видами: когнітивна інтеграція; інтеграція психомоторна; швидке прийняття рішень; закріплення навичок через повторення; дозволяє розібрати ситуацію по частинах; навчання через навчання інших [4].

Освітня симуляція – це структурований сценарій з детально розробленою системою правил, завдань і стратегій, які створені з певною метою: сформувати специфічні компетенції, які можуть бути безпосередньо перенесені в реальний світ. Комп'ютерні симуляції – це моделювання навчальної ситуації і послідовне її виконання з метою вирішення на комп'ютері.

Симуляції представляють деяку частину навколишньої дійсності, вони дозволяють вивчати ті аспекти дійсності, які не можуть бути вивчені іншим способом з міркувань безпеки, етики, високої вартості, необхідного технічного забезпечення або масштабу досліджуваного явища. Симуляції допомагають наочно уявити абстрактні поняття. Студенти розуміють суть досліджуваного явища завдяки можливості маніпуляції з його параметрами. Комп'ютерна симуляція як інтерактивна форма навчання володіє величезними можливостями:

створює образ реальних атрибутів діяльності; виступає як віртуальний аналог реального взаємодії; створює умови заміщення реального виконання соціальних або професійних ролей; є формою контролю ефективності професійного навчання.

В комп'ютерній симуляції виділяються основні компоненти:

- робоча модель професійного середовища або структурно-організаційна схема, в якій закладені можливі варіанти поведінки і взаємодії людей один з одним;
- сценарій (сюжет) процесу симуляції, спрямований на застосування знань, розвиток інтуїції, пошуку альтернативного нестандартного шляху вирішення проблеми.

Одним із сильних переваг комп'ютерних симуляцій полягає в тому, що вони можуть давати точну оцінку конкретних дій учня, оскільки технологія контролю вбудована в інструментальні засоби симуляцій.

Комп'ютерне моделювання, яке застосовується в освіті, може бути розділено на наступні категорії:

- комп'ютерні текстові симулятори;
- комп'ютерні графічні симулятори;
- симулятори з використанням манекенів;
- симулятор віртуальної реальності.

Комп'ютерні текстові симулятори створюють словесний опис ситуації, в якій користувач вибирає один з декількох готових відповідей. Ґрунтуючись на отриманій відповіді, комп'ютер генерує наступну ситуацію. Отримавши інформацію про дії студента, програма створює наступний бланк, де може бути представлено вже набагато більше варіантів подальшого вибору. Графічні симулятори відтворюють на дисплеї графічне зображення ситуації. Хоча такі симуляції сприяють розумінню і засвоєнню матеріалу зазвичай вони не розвивають у студентів практичних навичок.

Головна мета їх використання полягає в поясненні якихось абстрактних понять у доступній і недорогій формі. Симулятори з використанням манекенів можуть бути виконані з різним ступенем складності та реалістичності, але зазвичай завжди дорогі. Основу для створення симуляцій як елементу віртуальної реальності складають сценарії завдань, які б потім користувач мультимедійного навчального комплексу без труднощів міг виконати і засвоїти. Сценарій – це

детально розроблений план виконання будь якого завдання у віртуальному середовищі. Він складається зі списку дій та відповідних ним віртуальних сцен (зображень). Чим більший список, тим реальніше виглядає віртуальна реальність, але тим самим ускладнюється процес створення симуляцій[4].

1.2 Вимоги до навчальних програм і симуляторів та їх структура

На даний момент VR в галузі освіти переважно використовується в якості різного роду тренажерів, а також для демонстрації процесів, явищ та об'єктів, які неможливо або вкрай важко показати у справжній реальності. Такі технології електронного навчання мають багато переваг, але водночас мають певні недоліки. Використання навчальних програм поступово зменшує частку безпосереднього спілкування викладача і студента, тому їх застосування може бути виправдане лише отриманням нової якості навчання і нових можливостей для студентів .

Використання систем віртуальної реальності в галузі освіти-це новий підхід до подачі і засвоєння наукового і методичного матеріалу в школах і вузах. Школярі і студенти можуть попрацювати в унікальних експериментальних лабораторіях, поспостерігати за історичними подіями та навіть взяти участь в них, побувати в космосі, відправитися в подорож в будь-яку точку земної кулі, будувати об'ємні діаграми і проводити хімічні досліди. Учасники віртуальної системи можуть перебувати в різних містах і країнах і взаємодіяти один з одним в науковій сфері, разом спостерігати за експериментами і брати участь в наукових розробках.

Оскільки природа систем VR є емпіричною, то навчальні системи повинні походити від трьох джерел:

- 1) занурення;
- 2) інтерактивність;
- 3) багато сенсорний зворотній зв'язок.

«Занурення» означає бути охопленим або оточеним навколишнім середовищем. Перевага занурення полягає в тому, що воно забезпечує відчуття присутності або відчуття, що людина дійсно знаходиться у зображеному світі. Інтерактивність – це можливість контролювати події в моделюванні, використовуючи ті рухи тіла, які в свою чергу ініціюють відповіді в симуляції.

Метою VR є замінити реальний світ віртуальним і дозволити користувачеві поводити себе так, наче він перебуває у реальному світі.

Емпірична природа VR підтримує конструктивістський підхід до навчання. Конструктивізм – це теорія набуття знань, яка стверджує, що люди набувають знань в результаті досвіду. Теорія стверджує, що учень намагається асимілювати новий досвід у рамках вже встановленої і зрозумілої моделі. Якщо учень не може успішно асимілювати нові деталі, він змінює свій світогляд, щоб пристосуватися до нового досвіду. Коли ми діємо згідно очікувань, що світ побудовано відповідно до нашої світової моделі, і виявляємо, що це не так, ми повинні враховувати новий досвід, змінювати нашу модель того, як світ працює, тобто вчитися з досвіду. Звідси випливає, що навчання є формою активної перевірки гіпотез, отже навчання у VR має забезпечувати середовище для такого активного тестування гіпотез і бути потужним середовищем для навчання.

У моделюванні VR комп'ютер моделює і відображає середовище, де ми можемо ходити і взаємодіяти з об'єктами і імітованими людьми (зазвичай їх називають «агентами» або «аватарами»). Віртуальне середовище зображується зазвичай як тривимірний світ, і часто віртуальні світи намагаються відтворити реальний світ як за зовнішнім виглядом, так і в тому, як поведуться об'єкти (наприклад, симуляція гравітації). Однак слід зазначити, що немає необхідності, щоб цей віртуальний простір був схожий на реальний світ.

Однією з переваг віртуальних середовищ є те, що вони можуть бути використані для відображення абсолютно нереальних сценаріїв. Проте, для навчальних цілей віртуальні середовища моделюють середовище, у якому студент (або інший, хто навчається) працюватиме, і створюють безпечне середовище, у якому можна перевірити сценарії, що будуть або занадто складними, або небезпечними для виконання в реальному житті. «Сценарій – це детально розроблений план виконання будь-якого завдання у віртуальному середовищі. Він складається зі списку дій та відповідних їм віртуальних сцен (зображень). Чим більший список, тим реальніше виглядає віртуальна реальність, але водночас ускладнюється процес створення симуляцій».

В педагогічній діяльності особлива увага приділяється:

- механізмам впливу на мотивацію навчання,
- свідомість і поведінку студентів,

– розробці ефективної педагогічної технології виховання учнів в навчальному процесі.

Ці завдання можуть бути вирішені при застосуванні ігрових форм навчання, оскільки ігри дозволяють ефективно організувати творчу взаємодію викладача і студентів, створюють умови для формування особистісних якостей [7, 8].

Навчальна гра (освітня гра) – програмне забезпечення, яке тренує і навчає людину в ігровому режимі. Може застосовуватися як для навчання, так і для розваги.

У категорію навчальна гра входять жанри:

- квест;
- аркада;
- 3D-шутер;
- симулятор;
- комп'ютерний тренажер;
- інтерактивний курс з якої-небудь дисципліни.

Вибір жанру, в якому буде здійснюватися подача матеріалу навчальної гри є дуже важливим. Не будь-який жанр гри може підійти для конкретних цілей навчальної гри для конкретної дисципліни. Сучасні навчальні програми виглядають однаково. Різні тести і методичні роботи, навіть віртуальні лабораторні роботи виглядають не завжди цікаво і не збуджують уяву та зацікавленість користувачів.

Розробники вважають так: головне - це функціонал і адаптивність, а все інше відсувається на другий план. Важливо не тільки подати найбільш повні і опосередковані знання, а й зробити так, щоб студенти проявляли інтерес до самого процесу вивчення і пізнання. Викликати зацікавленість і мотивацію можна за допомогою використання ігрової форми подачі матеріалу [9].

Саме це і роблять вихователі і вчителі в школах з маленькими дітьми. Прийнято вважати, що студент у вищому навчальному закладі більш зібраний, самостійний і серйозний, але це не завжди так, тому більша частина всіх навчальних програм сприймаються студентами якщо не з негативом, то знизькоюзацікавленістю. На ринку мобільних навчальних програм існує величезний спектр різних продуктів і сервісів. Мабуть, одним з найпопулярніших для вивчення іноземної мови є - DuoLingo. Інтерфейс програми зображено на рисунку 1.1

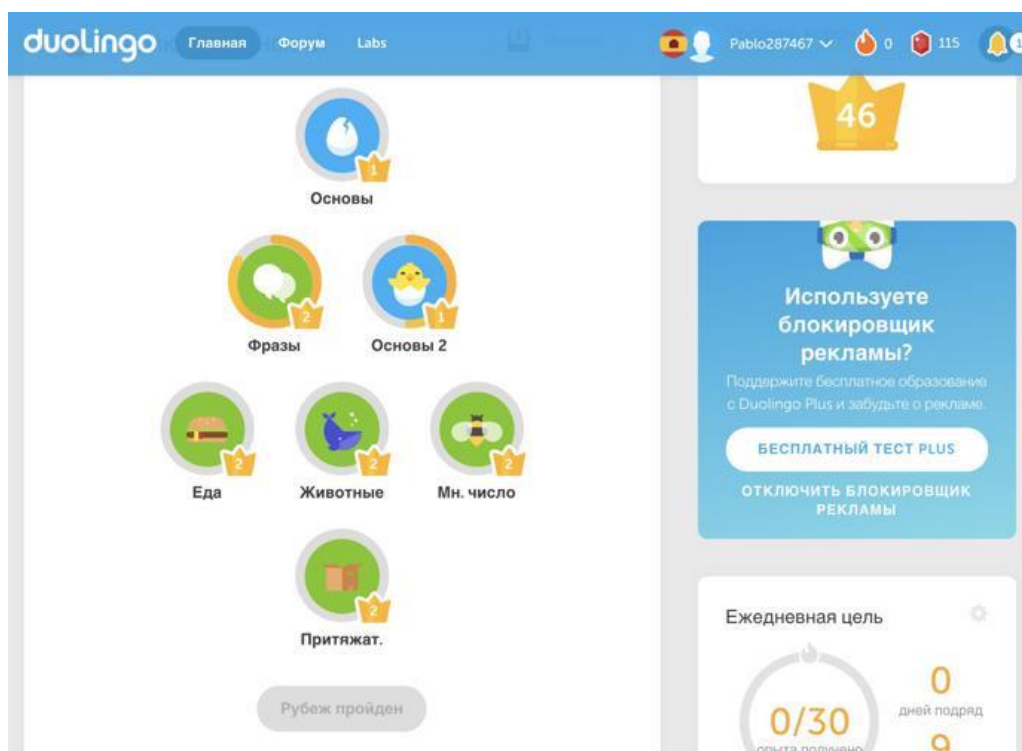


Рисунок 1.1 – Основний інтерфейс програми «DuoLingo»

Ця програма має низку важливих переваг – вона не ускладнена різними текстовими поясненнями. Весь навчальний матеріал подається в легкій і ненав'язливій формі питання-відповідь, де треба вибрати правильну відповідь із списку доступних або ж підставити потрібне слово, або картинку. Так само тут присутня адаптація завдань під рівень користувача. Всі теми жорстко фіксовані, але всередині теми завдання складені таким чином, щоб цікаво було їх вирішувати як і тим, у кого низький рівень володіння мовою, так і бажаючим закріпити або повторити якусь тему.

Основою ігрового процесу будь-якої відеогри є повторюваний ланцюжок дія-реакція-зворотний зв'язок [1].

Дія – інформація, яку гравець надає грі з ігрових контролерів, як геймпад чи клавіатура.

Реакція – відповідь гри на дії гравця, сформована на основі наданої ним інформації, але ще не виражена для нього в доступній формі.

Відгук – інформація, отримувана з гри гравцем. Поділяється на два основних види: явний(є очевидною відповіддю на дії гравця) і неявний(служить фоном чи інформує гравця, наприклад, довідка з гри). Зворотний зв'язок має свої різновиди в межах явного і неявного: візуальний(бачене гравцем на екрані), звуковий (чуте через пристрої відтворення звуку), дійова подія в самій грі –

відповідь на дії гравця, зазвичай поєднання зображення зі звуком), неігрових персонажів(відповідь персонажів, що «населюють» гру, на дії гравця), акумулятивний(виражений певним чином прогрес гри), емоційний (емоції, які викликає гра), здійснення(вираження сенсу окремих завдань і всієї гри, їх завершення), інформативний(відомості, які гравець отримує в контексті гри). Отриманий зворотний зв'язок спонукає до нових дій і так до завершення гри.

1.3 Огляд програмних засобів жанру «симулятор»

На даний момент кількість ігор для віртуальної реальності є невеликою відносно інших платформ. Серед них вже є представники ігор жанру simulator та під жанру fishing simulator.

Симулятор - програмні і апаратні засоби, що створюють враження дійсності, відображаючи частина реальних явищ і властивостей у віртуальному середовищі. Симулятор-гра – ігри, що надають можливість симуляції і управління тим чи іншим процесом з реального життя, наприклад, ловлі риби. Гравцю надається вудка і гра генерує рибу за певним правилом. У деяких іграх гравцю надається можливість вибирати наживку та ставити сітку для риби. Ігровий процес може ставити різні цілі для гравця і в залежності від них оцінювати швидкість, кількість та вид риби.

Наведемо приклади та короткий опис існуючих аналогів ігрового програмного забезпечення з цієї тематики.

1.3.1 Crazy Fishing

Гра вийшла у 2017 році під видавництвом ManaWings. Сам розробник описує гру як казуальну екшен гру, де користувач сам вирішує де там що ловити. Це перша гра яку розробник створив.

У грі є декілька видів риби яку гравець може зловити та продати. Окрім насолоди рибалки користувач може досліджувати ігровий простір через телепортацію. Також може прикрашати свій дім трофеями. Коли користувач бере в руки вудку йому потрібно зачепити приманку на гачок, даль він повинен закинути у воду, в залежності від сили замаху наживка може відчепитися від гачка. Коли риба піймається за гачок, вудка зігнеться – це сигнал що риба на

гачку. Щоб її достати треба крутити котушку. Є можливість пересуватися тільки за допомогою контролера, що є зручним для користувачів з обмеженою ігровою площею.

Відгуки у грі в основному дуже позитивні через графічну складову та взаємодією із ігровим простором. На рисунку 1.2 зображено процес гри у «CrazyFishing».



Рисунок 1.2 – CrazyFishing

Цю гру можна вважати найближчим аналогом для даного ігрового програмного забезпечення. При створенні ПО було використано 2 механіки з цієї гри.

Було позичено механіку «Намотування волосінь». Гравцю необхідно розмотувати волосінь для того щоб далеко закинути вудку, а для того щоб витягнути рибу, треба замотати та дістати із води рибину рукою. Також взяли механіку «Наживки», для того щоб риба частіше клювала, гравець можеш вибрати 1 із декілька видів наживки яку він покупає за зароблені гроші.

1.3.2 Fancy Fishing VR

Гра вийшла у 2017 році під видавництвом китайської студії. На відміну від попереднього аналогу гра має інший жанр та менше відгуків. У грі є декілька режимів, також користувач обмежений в переміщенні.

У грі подали нестандартний вид на рибалку, суміщено рибалку та тир. Гравець знаходиться під водою в якому має інвентар в якому може вибрати

зброю. Ігровий процес складається з попаданням зброєю у рибу, яка плаває навколо гравця. У грі реалізовано різні режими. Наприклад в одному з них на гравця можуть напасти акули але можна відбитися яка в її поцілити, гра сповіщає звуком про те, що з'явилася нова акула. В іншому режими гравець має обмежені патрони та має поцілити якомога більше риби

Графіка у грі більш реалістична, ніж у першого аналогу, але досі примітивна. У грі добре пророблена механіка взаємодії зі зброєю. Ігровий процес «FancyFishingVR» зображено на рисунку 1.3.



Рисунок 1.3 – FancyFishingVR

Рахунок користувача зберігається та відображується на спеціальній дошці серед попередніх його результатів, що дає змогу гравцю оцінювати свій поточний результат.

1.3.3 Amigo Fishing

Гра з'явилася у 2018 році. Видавцем гри є Amigo VR. Гра безкоштовна. Гравець знаходиться глибоко під водою у субмарині. Треба ловити рибу за допомогою пушок які зтягують рибу до підводної човен. Фактично, ця гра поєднує у собі рибалку та shooter. За пійману рибу отримує гроші для покращення зброї. Після ловлі певної кількості риби гравець отримує рівень і відкриваються нові пушки у магазині. Гра є нескінченною та після виходу зберігається ігровий процес.

На даний момент гра знаходиться у ранньому доступі і періодично оновлюється. Додаються нові локації та режими гри. Оскільки гра досить нова, відгуків має мало. До переваг гри можна віднести цікавий спосіб ловлі, хорошу анімацію та фізику. Загалом графіка у грі досить непогана. Є недоліки з управління, наприклад, при перезарядці зброї гравцю заважає розмір контролерів, або якщо користувач впустив зброю на підлогу, її складно підняти.

Ігровий процес у режимі класичного тирю зображено на рисунку 1.4.



Рисунок 1.4 – Amigo Fishing VR

Хочу відмітити те що в цієї гри дуже гарний AIриби. При попаданні у найближчих риб інші намагаються упливати від місця куди влучив гравець.

2 ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ ТА ОБГРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

2.1 Аналіз проблем ігор-аналогів

Загалом, виконавши огляд ігор-аналогів, можна виділити такі проблеми, як відсутність підказок для гравця та насиченість багатьма непотрібними для гри деталями. Тобто відсутні дуже важливі складові для користувачів які у перший раз знайомляться з технологією віртуальної реальності. Є великий ризик що користувачеві набридне гра і він почне втомлюватися від неї.

Крім того у таблиці 2.1 наведено порівняння цих розробок за основними вимогами до ігрових програм.

Отже, для уникнення існуючих проблем необхідно розробити гру у якій не буде зайвих деталей і які будуть допомагати користувачеві у проходженні своєї основної місії.

Таблиця 2.1 – Порівняння обраних аналогів

Характеристика	Crazy Fishing	Fancy Fishing VR	Amigo Fishing VR
Особливі навички гравця	Вміння вибирати місце ловлі та наживку.	Вміння швидко реагувати на екстремальну ситуацію	Вміння швидко реагування на зміну у рівні.
Звукове супроводження	Присутнє стандартне супроводження притаманне даному жанру гри.	Присутнє похвала після вдалого проходження рівня. Стандартне супроводження	Має різноманітну карту звуків, які можуть збити гравця з мети.

Ігрові механіки	Має усі потрібні ігрові механіки для даного жанру гри.	Має тир механіку що не підходить для жанру.	Має тир-подібну механіку, перенасичення ігрових механік
-----------------	--	---	---

Можна зробити висновки по характеристикам, які були наведені у таблиці вище. Отже, звукове супроводження у грі CrazyFishing дуже насичене, бо гра заснована на жанрі simulator який має бути насиченим. Саме тому у грі присутнє стандартне звукове супроводження. Гра має багато деталізованих речей які б потребували максимальної деталізації. Насиченість у грі заважають, та тримають користувача у напруженні. Динамічність гри, для цього жанру має відповідну динамічність, не вимагає від користувача швидку реакцію.

Наступна гра FancyFishing, дуже велика насиченість ігрових механік, які перенасичують цей жанр. Має жанр тир, що не дуже підходить для жанру симуляторів. Графічні показники, вони грають велику роль у даному жанрі гри, із за великої кількості об'єктів вимагає детальної прорисовки. Звукове супроводження робить дану гру яскравою. Динамічність цієї гри дозволяє гравцю постійно будить у напрузі.

Останньою грою в порівнянні аналогів залишається AmigoFishingVR. Гра має певні недоліки та переваги. Завдяки багатій кількості речей у грі є можливість заплутатися гравцеві у пошуку потрібної речі. Саме тому графічний показник вимагає великої кількості навантаження на відео карту і із за цього гра потребує великої кількості місця для зберігання пройдених рівнів. Як у попередній грі, вона має спокійний характер який дозволить користувачеві спокійно проходити рівні. Кольорова насиченість, переважають яскраві кольори тим самим заохочують розглядати усі предмети які знаходяться у грі. Викликають дуже великий інтерес у користувачів.

Отже, можна зробити висновок, що серед аналогів є свої недоліки та переваги саме тому потрібно ураховувати їх під час проектування та розробки власного ігрового програмного додатку, щоб уникнути певних недоліків потрібно підібрати гарну насиченість кольорів, максимально позбавитися від зайвих об'єктів на сцені[10].

2.2 Постановка задачі

З огляду на наведені вище недоліки, можна зробити висновок, що жодне з наведених ігрових програмних забезпечень для систем віртуальної реальності не відповідає потребам користувача у повній мірі.

Складність, висока або низька, це компроміс доступності. Це комбінація здібностей гри до позначення цілей (гра з величезною порожньою картою без єдиної підказки буде складною, але не цікавою), способів їх досягнення (гра, в якій кнопка пострілу вибирається випадково після кожного натискання, буде складною, але не цікавою) і створення завдання, пропорційного \ можливостям гравця.

Відповідно до теорії потокового стану, щоб гравець відчув **«потік»** - стан зацікавленості в грі, коли час в ній пролітає непомітно, потрібно кинути йому **виклик**, відповідний його рівню.

Якщо завдання надто важке, гравець буде відчувати фрустрацію, якщо легке - нудьгу. Невідповідна складність вбиває віру в запропоновані обставини.

Складнощі як об'єктивної заходи не існує. Це відносне поняття, різниця між гравцем і викликом, який пропонує гра. З точки зору розробників, складність існує тільки щодо гравця і його зразкового розуміння ігрових механік, цілей і способів їх досягнення.

Метою магістерської роботи є визначення особливостей застосування віртуальних технологій у навчальному процесі, які можуть бути використані для покращення навчання та придбанні навичок у симуляторі.

З вищесказаного випливає, що задача полягає у дослідженні методів, засобів та технологій віртуальної реальності, які забезпечують виконання вимог до навчальних (ігрових) програм:

- складність, відповідна рівню підготовки користувача;
- виклик та занурення, що обумовлює зацікавленість у грі;
- навчання та заохочування.

Для виконання завдання дослідження методів віртуальної реальності в атестаційній роботі необхідно розробити навчальну гру типу «стимулятор». Для реалізації обрано тематику рибалки.

2.3 Опис gameplay та прийнятих програмних рішень

Ігровий процес в цілому типовий для даного виду ігор. Ідея гри полягає в тому, щоб гравець, занурившись у віртуальну реальність зміг зловити рибу за відведений час.

Гравцеві потрібно зловити якомога більше риби за відведенні грою(розробником) 7 хвилин ігрового часу. За кожну рибу отримує певну кількість балів. У кожної риби є своя рідкість, звідси йде різна кількість балів за рибу. Після завершення таймеру, гравцю показують його результат, рахунок записується до «Таблиці рекордів».

Гравець може закинути вудку у воду та чекати поки риба піймається за гачок. Після того як риба з'їла гачок, гравцю необхідно почати крутити катушку. Коли риба буде біля гравця необхідно рибу зняти з качка. Як тільки кнопка на контролері буде відпущена, риба автоматично добавиться до «Кошика»

Гравець може пересуватися по платформі у кімнаті використовуючи телепорт, займаючи потрібне місце. Також на деяких рівнях буде присутній «Очерет», у якому може застрягнути крючок, його можна буде зняти через кнопку на контролері.

Будуть присутні різні об'єкти такі як:

- вудка,
- 7 видів риби;
- ускладнюючий елемент «Очерет»;
- управління персонажем здійснюється за допомогою контролера (телепорта).

Для взаємодії з грою потрібно тримати кнопку закидання, зрегулювати силу кидку та очікувати коли риба схопить гачок.

Щоб витягнути рибу треба крутити контролером як крутиш справжню катушку.

До програмного продукту існує ряд вимог, які необхідно дотримати при розробці. Так як вимоги до програмного продукту різняться за типом їх потрібно розглянути окремо.

2.4 Методи та технології реалізації віртуальної реальності

Технологія без занурення — це найменш захоплююча реалізація технології віртуальної реальності. У такому випадку стимулюється лише деяка множина відчуттів користувача, що дозволяє периферійне усвідомлення дійсності поза моделюванням віртуальної реальності. Користувачі опиняються у таких тривимірних віртуальних середовищах через портал або вікно, використовуючи стандартні монітори високої роздільної здатності, які працюють на процесорній потужності, зазвичай це зустрічається на звичайних настільних робочих станціях [11].

Часткове занурення забезпечує більш захоплюючий досвід, в якому користувач частково, але не повністю занурений у віртуальне середовище. В частковому зануренні використовують багато однакових технологій, отриманих, наприклад, при моделюванні польотів. Моделювання такого занурення забезпечується високопродуктивними графічними обчислювальними системами, які часто з'єднуються з велико-екранними проектними системами або кількома телевізійними системами для правильного стимулювання користувацьких зображень.

Повне занурення забезпечує найбільш захоплююче використання технологій віртуальної реальності. При симулюванні з повним зануренням, такі апаратні засоби, як дисплеї, що вмонтовані в шоломи та пристрої відслідковування руху, використовуються для стимулювання всіх відчуттів користувача. Повністю інтегровані, такі технології здатні забезпечити дуже реалістичний досвід для користувача завдяки широкій області перегляду, високій роздільній здатності, підвищеній частоті оновлення та високому рівню контрастності на головному дисплеї (HMD) [11].

Щоб людський мозок сприйняв штучне, віртуальне середовище як реальне, воно повинно виглядати не тільки як реальне, але й давати можливість відчутти себе реальним. Таких відчуттів можна досягти, одягнувши на голову шолом (HMD), який відображає відтворені в натуральну величину об'єкти, 3D-віртуальне середовище без кордонів, яке зазвичай можна побачити на екрані телевізора чи комп'ютера. Відчуття реальності може бути досягнуто за допомогою портативних пристроїв введення, таких як відслідковуючі рух пристрої. Крім того дані можуть передаватися і безпосередньо через нервові закінчення або в головний мозок за

допомогою мозкових інтерфейсів. Подібна технологія застосовується в медицині, але поки вона занадто дорога для повсякденного застосування і не досягає якості передачі даних, прийнятної для застосування віртуальної реальності.

Для відтворення віртуальної реальності необхідні декілька ключових компонентів, а саме:

- гарнітура віртуальної реальності, настільки ж важлива, як і те, що користувачі бачать всередині гарнітури;
- значна обчислювальна потужність, для того, щоб керувати цими інтерактивними тривимірними середовищами: це персональні комп'ютери, консолі та смартфони;
- пристрої введення – одна з категорій компонентів, які надають користувачам почуття занурення (тобто переконання людського мозку сприймати штучне середовище як справжнє).

Пристрої введення забезпечують користувачів більш природним способом навігації та взаємодії в середовищі віртуальної реальності. Найбільш поширені форми пристроїв вводу віртуальної реальності включають в себе:

- джойстики;
- відслідковуючі сфери;
- контролери Wands;
- спеціальні рукавички;
- трек пади;
- кнопки керування на пристрої;
- трекери руху;
- боді-сумки;
- бігові доріжки;
- платформи руху (VirtuixOmni).

Окуляри віртуальної реальності (також називаються HMD, Headset, шоломи віртуальної реальності) — це тип пристрою, який має дисплей або дисплеї, встановлені перед очима користувача. Ці дисплеї, як правило, охоплюють повну область перегляду користувача та відображають вміст віртуальної реальності. Є декілька типів окулярів віртуальної реальності. Одні працюють за допомогою комп'ютера, наприклад OculusRift, HTC Vive чи під'єднуються до ігрової консолі, наприклад, PlayStation VR. Інші ж під'єднуються до смартфона Android, iPhoneили Windows Phone, наприклад VR Box, GoogleCardboard. Також є

незалежні окуляри віртуальної реальності, в яких вже є вбудований дисплей та процесор з іншими частинами, наприклад Sulon Q, Deeroon чи OmimoUranus VR One [10].

Для створення віртуальної реальності використовуються такі методи:

- механіка;
- анімація;
- графіка;
- поведінка об'єктів(фізика);
- штучний інтелект;
- звукове супроводження.

Механіка – це метод, який реалізує набір правил і способів інтерактивної взаємодії гравця і гри. Уся множина ігрових механік гри формують конкретну реалізацію її ігрового процесу. Механіка реалізується засобами мов програмування (середовищами).

Анімація та графіка забезпечує обраний розробниками стиль та дизайн гри, а також глибину та деталізацію віртуальності. Комп'ютерна графіка - область діяльності, в якій комп'ютери поряд зі спеціальним програмним забезпеченням використовуються в якості інструменту як для створення (синтезу) і редагування зображень, так і для оцифрування візуальної інформації, отриманої з реального світу, з метою подальшої її обробки і зберігання. За способами подання зображень графіку можна розділити на категорії двовірної графіки(2D) ті тривимірної графіки(3D). До двовірної графіки відноситься[1]:

- векторна графіка;
- растрова графіка;
- фрактальна графіка;

Тривимірна графіка оперує з об'єктами в тривимірному просторі. Зазвичай результати являють собою плоску картинку, проекцію. Комп'ютерна графіка широко використовується в кіно, комп'ютерних іграх.

Тривимірна графіка буває полігональною і воксельною. Воксельна графіка аналогічна растровій. Об'єкт складається з набору тривимірних фігур, найчастіше кубів. А в полігональній комп'ютерній графіці всі об'єкти зазвичай представляються як набір поверхонь, мінімальну поверхню називають полігоном. Як полігон зазвичай вибирають трикутники.

3D-графіка в порівнянні з 2D-графікою - це графіка, яка використовує тривимірне уявлення геометричних даних. З метою продуктивності це зберігається в комп'ютері. Це включає в себе зображення, які можуть бути для подальшого відображення або для перегляду в реальному часі.

CGI (Computer-generated imagery) - зображення, одержувані комп'ютером на основі розрахунку і використовуються в образотворчому мистецтві, друку, кінематографічних спецефектах, на телебаченні і в симуляторах. Створенням рухомих зображень займається комп'ютерна анімація, що представляє собою більш вузьку область графіки CGI[1].

На відміну від більш загального поняття «графіка CGI», що відноситься як до нерухомих, так і до рухомих зображень, комп'ютерна анімація має на увазі тільки рухомі. На сьогодні набула широкого застосування як в області розваг, так і у виробничій, науковій та діловій сферах

Анімація створюється у таких популярних середовищах як середовищах:

- Adobe Photoshop;
- GIMP;
- Blender;
- Cinema 4D;
- Vectorian Giotto;
- Source Filmmaker;
- Anime Studio.

На даний момент використовується тривимірна комп'ютерна анімація, і її можна розділити на два основних напрямки: фотореалістична і не фотореалістичний рендеринг. Фотореалістична комп'ютерна анімація сама по собі може бути розділена на дві підкатегорії: реальний фотореалізм (де захоплення руху використовується в створенні віртуального персонажа-людини) і стилізований фотореалізм.

Ігровий штучний інтелект (ШІ) - набір програмних методик, які використовуються в комп'ютерних іграх для створення ілюзії інтелекту в поведінці персонажів, керованих комп'ютером. Ігровий ШІ, крім методів традиційного штучного інтелекту, включає також алгоритми теорії управління, робототехніки, комп'ютерної графіки та інформатики в цілому.

Реалізація ШІ сильно впливає на геймплей, системні вимоги і бюджет гри, і розробники балансують між цими вимогами, намагаючись зробити цікавий і

невимогливий до ресурсів ШІ малою ціною. Тому підхід до ігрового ШІ серйозно відрізняється від підходу до традиційного ШІ: широко застосовуються різного роду спрощення, обмани і емуляції. Наприклад: з одного боку, в шутерах від першої особи безпомилкове рух і миттєве прицілювання, властиве роботам, не залишає жодного шансу людині, так що ці здібності штучно знижуються. З іншого боку - боти повинні робити засідки, діяти командою і т. і. Для цього застосовуються «милиці» у вигляді контрольних точок, розставлених на рівні. Персонажів комп'ютерних ігор, керованих ігровим штучним інтелектом, ділять на[1]:

- неігрові персонажі – як правило, ці персонажі є дружніми або нейтральними до персонажу, за якого грає людина;
- боти – ворожі до гравця персонажі, що наближаються за можливостями до ігрового персонажу. Боти найбільш складні в програмуванні;
- моби – ворожі до гравця «низько інтелектуальні» персонажі. Моби вбиваються гравцями в великих кількостях заради очок досвіду, артефактів або проходження території.

Евристичні алгоритми ігрового штучного інтелекту використовуються в широкому розмаїтті в багатьох галузях всередині гри. Найочевидніше застосування ігрового ШІ проявляється в контролюванні неігрових персонажів, хоча скриптинг теж є дуже поширеним способом контролю. Пошук шляху є іншим широко поширеним застосуванням ігрового ШІ, він особливо проявляється в стратегіях реального часу. Пошук шляху є методом для визначення того, як неігрового персонажу перейти з однієї точки на карті до іншої: потрібно враховувати ландшафт, перешкоди і, можливо, «туман війни». Ігровий ШІ також пов'язаний з динамічним ігровим балансуванням.

Звук як джерело інформації. Головна відмінність ігор від, наприклад, кіно – в інтерактивності. Звукове середовище повинне інформувати дії гравця, які, в свою чергу, теж на нього впливають. Тому в іграх гіпертрофовані звуки, які допомагають гравцеві розробити стратегію.

Музика і атмосфера. Саундтреки відеоігор вже давно можна поставити в один ряд з музикою до кіно за якістю і емоційного впливу на слухача. У поєднанні з інтерактивними елементами вони створюють дивовижний ефект занурення в штучний світ гри.

У хоррорах гнітючі, скрегочучи, що шарудять мелодії з різкими вибухами струнних тримають гравця в напрузі і змушують вслухатися в кожен найменший скрип. Несподівана поява противника, що супроводжується раптовим гучним звуком, лякає помітно сильніше, ніж без озвучування

Об'ємний звук це ще один спосіб занурити гравця в уявний світ використавши звуки гри і музику. Ця техніка особливо важлива для набирають популярність VR- і AR-ігор. У них гравець фізично займає місце персонажа, і нереалістичне аудіо здатне зруйнувати цю ілюзію. Для створення об'ємного звуку розробники використовують бінауральні і багатоканальні записи, а також спеціальні бібліотеки, що дозволяють динамічно обробляти джерела ігрового звуку [1].

2.4 Аналіз використання технологій віртуальної реальності у продуктах-аналогах

В якості продуктів аналогів для дослідження в роботі обрані ігри CrazyFishing , FancyFishingVR ,AmigoFishingVR. У таблиці 2.2. наведено результати аналізу використання методів віртуальної реальності:

Таблиця 2.2 Порівняння продуктів-аналогів за використанням методів VR

	CrazyFishing	FancyFishingVR	AmigoFishingVR
Механіка	<ul style="list-style-type: none"> - Закидання вудки у воду - Насадження наживки - Згинання вудки коли риба тягне крючок - Продаж улову - Прикрашання дому трофеями - Міні ігри 	<ul style="list-style-type: none"> - Стріляти у рибу - «Вбита» риба дає монети - Вибір зброї за монети. - Можна вмерти від акули 	<ul style="list-style-type: none"> - Контролерами можна взяти або положити пістолети. - Стріляти у рибу - Вдосконалювати пістолети, вставляючи різні модифікації
Анімація	<ul style="list-style-type: none"> - У воді анімацію не можна побачити 	<ul style="list-style-type: none"> - Усі риби мають однакові рухи. 	<ul style="list-style-type: none"> - Анімація старту гри, гравця

	- Кожна риба має свою унікальну анімацію при	- Акули та деякі морські жителі мають свої рухи у воді - Монети летять до таблиці з рахунком	запускають з корабля у спеціальній капсулі - Риба довільно рухаються, також є косяки риб. - При смерті падають монети
Графіка	Гра зроблена в мультяшному стилі	Виконано в реалістичному стилі	Виконано у реалістичному стилі
Штучний інтелект	У воді рибу не видно. Гравець зрозуміє що риба на гачку коли вудка зігнеться. Коли рибу витягують, вони може використовувати унікальні звуки.	Риби рухаються у випадковому напрямку. Коли у рибу влучає снаряд, то вона и усі риби поруч починають швидше рухатись.	Риби рухаються зліва на право, або навпаки. На постріли не як не реагують

Можна зробити висновки по характеристикам які були наведені у таблиці вище. Отже, звукове супроводження у грі CrazyFishing дуже насичене, бо гра заснована на жанрі simulator який має бути насиченим. Саме тому у грі присутне стандартне звукове супроводження. Гра має багато деталізованих речей які б потребували максимальної деталізації. Насиченість у грі заважають, та тримають користувача у напруженні. Динамічність гри, для цього жанру має відповідну динамічність, не вимагає від користувача швидку реакцію.

Наступна гра FancyFishing, дуже велика насиченість ігрових механік, які перенасищують цей жанр. Має жанр тирю, що не дуже підходить для жанру симуляторів. Графічні показники, вони грають велику роль у даному жанрі гри, із за великої кількості об'єктів вимагає детальної прорисовки. Звукове супроводження робить дану гру яскравою. Динамічність цієї гри дозволяє гравцю постійно будить у напрузі.

Останньою грою в порівнянні аналогів залишається AmigoFishing VR . Гра має певні недоліки та переваги. Завдяки багатій кількості речей у грі є можливість

заплутатися гравцеві у пошуку потрібної речі. Саме тому графічний показник вимагає великої кількості навантаження на відео карту і із за цього гра потребує великої кількості місця для зберігання пройдених рівнів. Як у попередній грі, вона має спокійний характер який дозволить користувачеві спокійно проходити рівні. Кольорова насиченість, переважають яскраві кольори тим самим заохочують розглядати усі предмети які знаходяться у грі. Викликають дуже великий інтерес у користувачів.

ЗАРХІТЕКТУРА ТА ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ

3.1 UML-моделювання системи

Моделювання розроблюваної навчальної гри проводиться з використанням мови UML для побудови діаграм, що допоможуть відобразити функціональність та внутрішню структуру системи. UML – мова графічного опису для об’єктного моделювання в галузі розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, який використовує графічні позначення для створення абстрактної моделі системи, що називається UML-моделлю. При розробці системи були розроблені діаграми наступних типів [13]:

- діаграма варіантів використання;
- діаграма класів;
- діаграма послідовності;
- діаграма станів.

3.1.1 Use case diagram

На рисунку 3.1 приведена діаграма варіантів використання. Діаграма використання показує, що в системі існує тільки один користувач. Він може почати нову гру, відкрити правила гри, змінити налаштування або вийти з ігрового додатку. В налаштуваннях користувач може відключити звуки або змінити руку, з якої ведеться ловля.

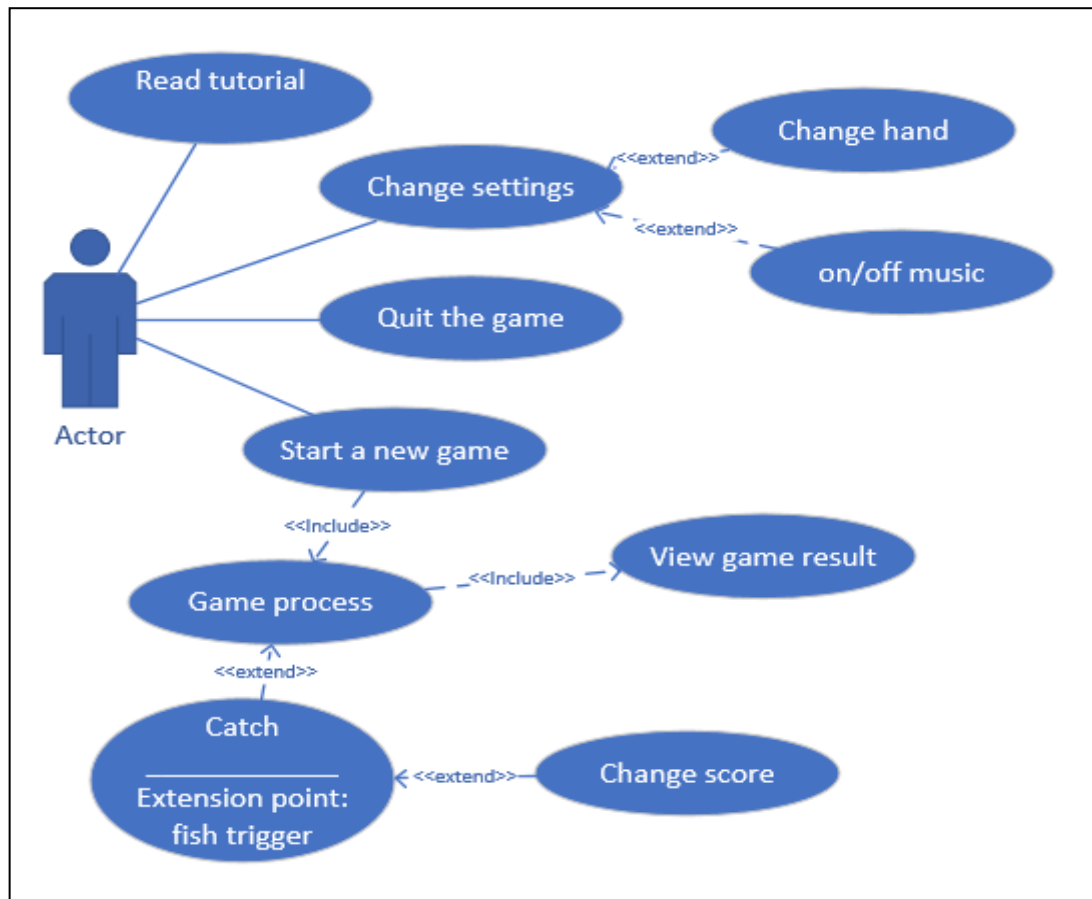


Рисунок 3.1 – Діаграма варіантів використання

При виборі нової гри починається ігровий процес, який включає в себе безпосередньо ігровий процес та перегляд результатів після завершення ігрової сесії. Якщо під час ігрового процесу користувач спіймав рибу його рахунок змінюється.

3.1.2 Class diagram

На рисунку 3.2 зображено діаграму класів. На ній можна побачити, що існує інтерфейс Catch, що реалізується класами

Класи реалізують однакові методи, єдина різниця в них – змінна Score, в якій зберігається кількість балів за ловлю риби. Також хочу відмітити про AIController який дозволяє рибам динамічно пересуватися по рівню. AIController - базовий клас контролерів для керованих AI в UnrealEngine. У той час як PlayerController покладається на гравця людини для прийняття рішень «що робити», AIController більш орієнтований на реагування на вклад з навколишнього середовища та ігрового світу. Завдання AIController - спостерігати

за навколишнім світом і приймати рішення та реагувати відповідно діям гравця-людини. Для AIController було створено два скрипти на мові C++ які відповідають за рух риби та поведінку на гачку(коли у воді, та коли витянули)

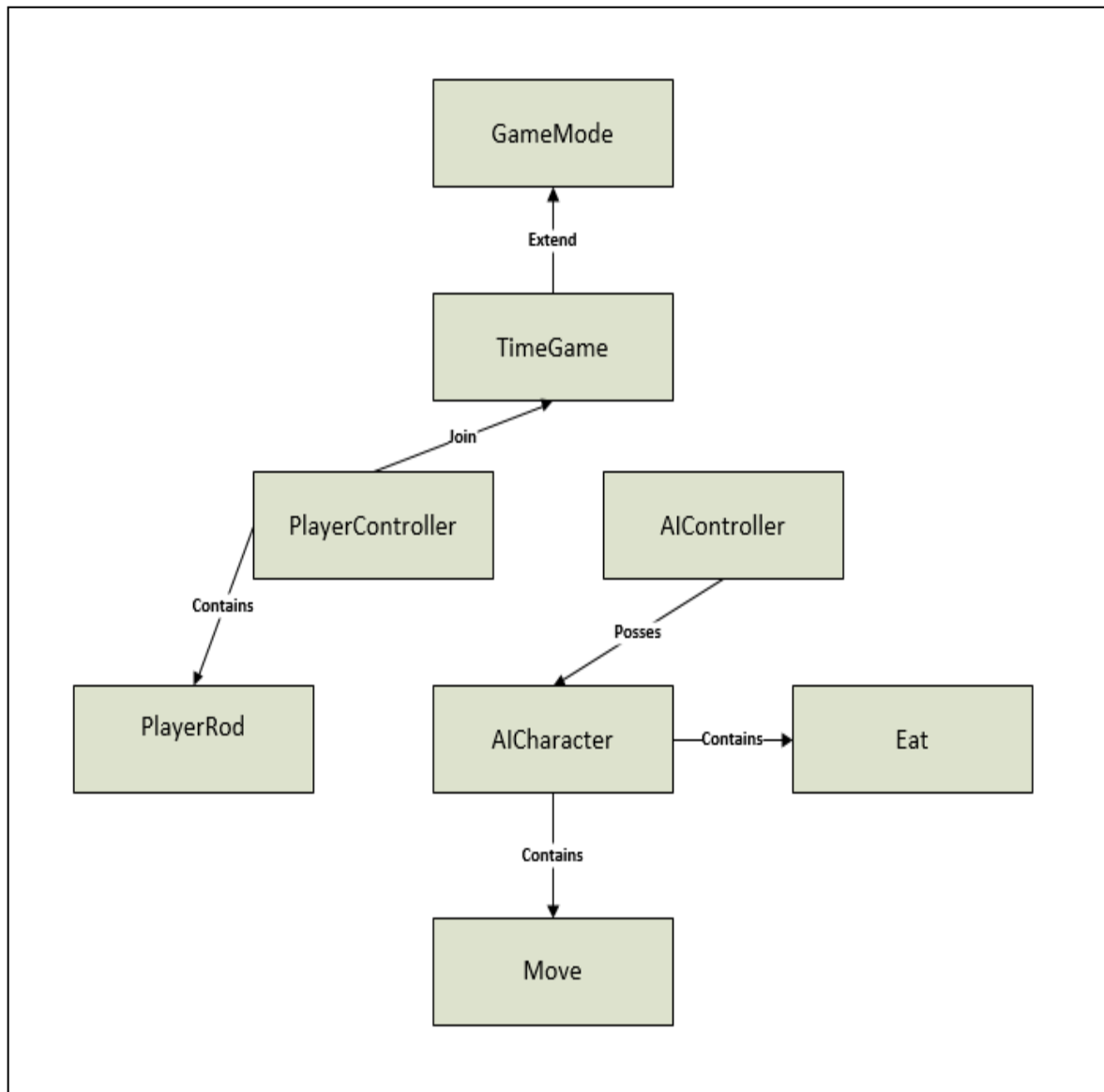


Рисунок 3.2 – Діаграма класів

Існує високорівневий клас Player, що викликає класи які відповідають за взаємодію користувача з грою та реалізує методи для ловлі. Клас InGamePlayerScore відповідає за підрахунок балів під час гри, а клас InGamePlayerTime відповідає за контроль часу ігрової сесії. Клас TargetSpawner відповідний за генерацію риб на ігровому просторі.

Під час ловлі викликається подія WasHit(), якщо риба, яка знаходиться на зоні триггеру, реалізує інтерфейс Target.

3.1.3 Sequence diagram

На рисунку 3.3 зображено діаграму послідовності. У ній відображено впорядковану в часі взаємодію об'єктів під час ігрової сесії.

Коли користувач розпочинає нову гру об'єкт, що відповідає за ігровий процес (GameMode на діаграмі), сповіщає об'єкти Score, Time та TargetSpawner про початок ігрової сесії. Time викликає TimeWidget, що відповідає за відображення таймеру у грі, та періодично оновлює час на ньому. Аналогічним чином працюють Score та ScoreWidget, але відображають рахунок гравця.

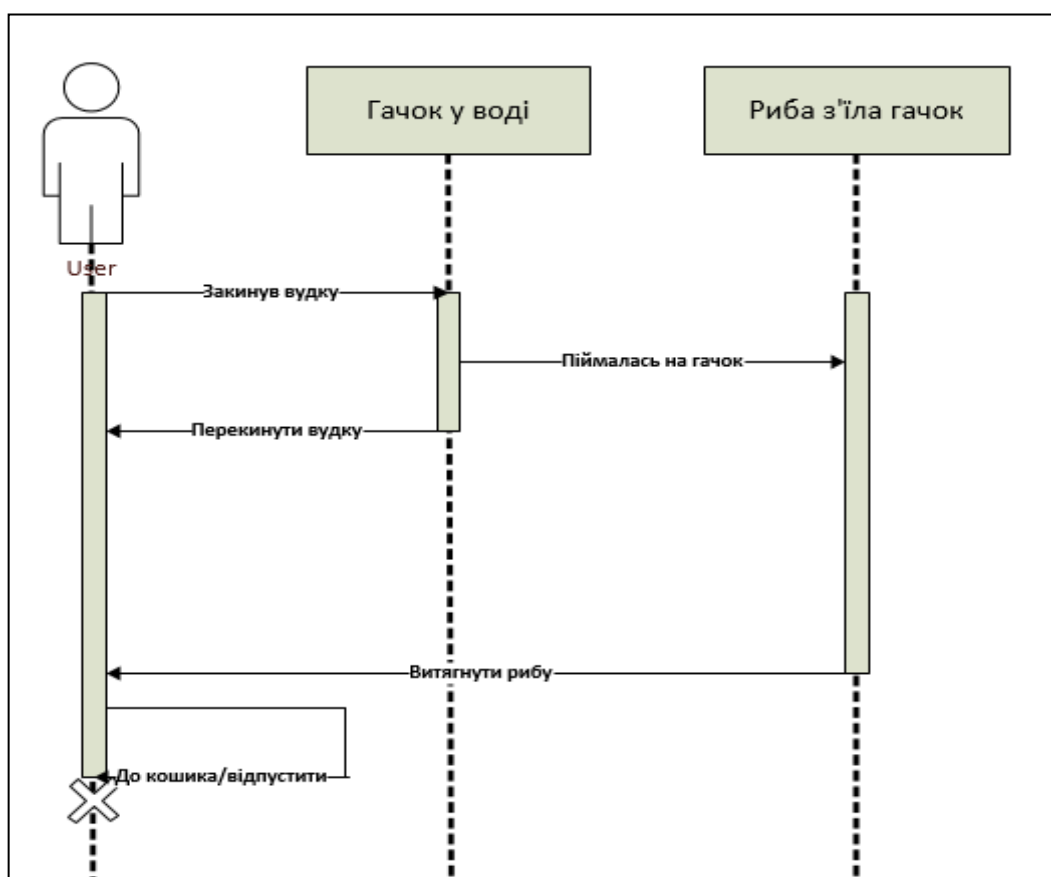


Рисунок 3.3 – Діаграма послідовності

Далі TargetSpawner генерує рибу одного з трьох типів. Якщо користувач впіймав рибу Target сповіщає GameMode про це. GameMode інформує Score про зміну рахунку, і він в свою чергу оновлює ScoreWidget.

3.2 Створення UI / UX дизайну, графіки та анімації

Інтерфейс - надтонка плівка, яка відокремлює гравця від гри. Коли з цією плівкою щось не так, крихкий каркас досвіду, створений завдяки взаємодії гравця з грою, може зруйнуватися [14].

З цієї причини надзвичайно важливо розуміти те, як працює ігровий інтерфейс, щоб робити цей каркас настільки міцним, сильним і непомітним, наскільки це можливо.

Але спочатку потрібно визначити цілі хорошого інтерфейсу. Він не повинен просто "добре виглядати" або "плавно працювати", хоча це теж важливі властивості. Мета інтерфейсу - дати гравцеві відчуття того, що саме він керує своїм досвідом.

Інтерфейси є основою взаємодії в сучасних інформаційних системах. Якщо інтерфейс якого-небудь об'єкту (комп'ютера, програми, функції) не змінюється (стабільний, стандартизований), це дає можливість модифікувати сам об'єкт, не перероблюючи його принципи взаємодії з іншими об'єктами. Це дозволяє масштабувати інтерфейс, створювати інші пристрої чи функції, які будуть використовувати такий спосіб взаємодії [13].

Інтерфейс вводу-виводу вимагає управління процесором кожного пристрою. Інтерфейс повинен мати відповідну логіку для інтерпретації адреси пристрою, що генерується процесором. Встановлення контакту повинно бути реалізовано інтерфейсом за допомогою відповідних команд типу (ЗАЙНЯТИЙ, ГОТОВИЙ, ЧЕКАЮ), щоб процесор міг взаємодіяти з пристроєм вводу-виводу через інтерфейс. Якщо існує необхідність передачі розрізнених форматів даних, то інтерфейс повинен вміти конвертувати послідовні (впорядковані) дані в паралельну форму і навпаки.

Комп'ютер, що використовує введення-виведення з розподілом пам'яті, звертається до апаратного забезпечення за допомогою читання і запису в певні осередки пам'яті, використовуючи ті ж самі інструкції мови асемблера, які комп'ютер зазвичай використовує при зверненні до пам'яті.

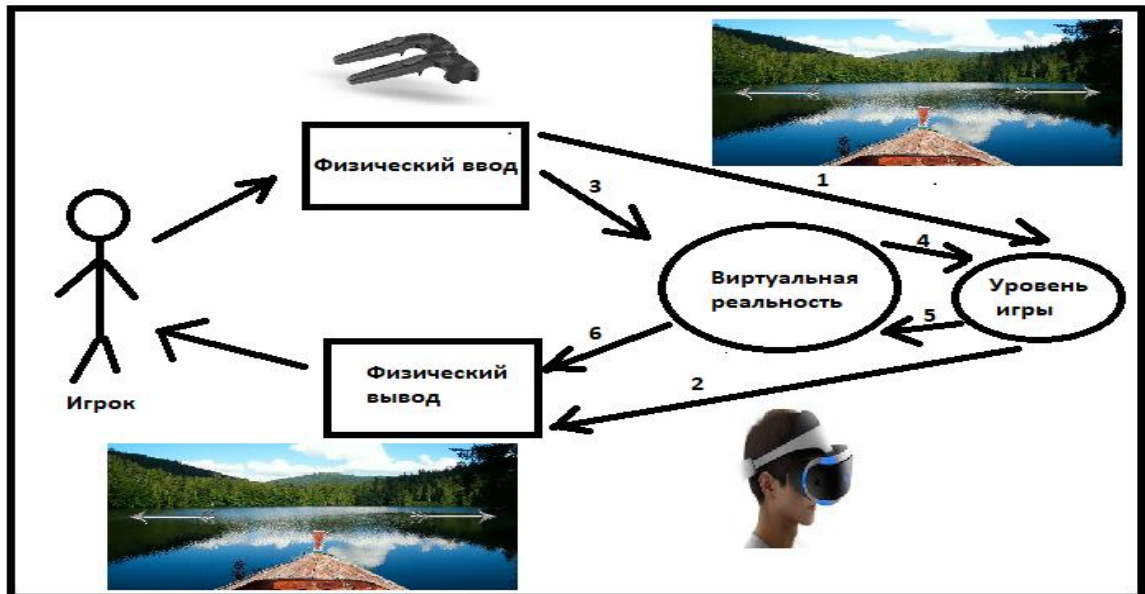


Рисунок 3.4 – Взаємодії людина-світ через фізичний та віртуальний інтерфейси

Інформація передається по колу: від гравця до гри і від гри до гравця, знову і знову. Це нагадує потік води, який крутить колесо водяного млина, генеруючи ігровий досвід. Але це стосується не всієї інформації, що знаходиться в цьому колі. Інформація, яка повертається до гравця від гри, в значній мірі впливає на те, якими будуть його наступні дії. Цю інформацію зазвичай називають фідбек (англ. - Feedback: відповідна реакція), а від якості цього фідбека залежить те, як добре гравець розуміє те, що відбувається в грі, і то, як сильно йому це подобається.

У більшості випадків достатньо одного правила: якщо інтерфейс не відповідає гравцеві протягом десяти секунд після введення, гравець буде думати, що з інтерфейсом щось не так. Типовий приклад подібної проблеми часто виникає в розробці гри з кнопкою "стрибок". Якщо художник, що працює над анімацією стрибків, не має досвіду створення відеоігор, він, швидше за все, додасть елемент "очікування" до анімації стрибка, коли персонаж присідає, готуючись до стрибка, що може зайняти від чверті до пів секунди. Можна подумати, що це стосується лише анімації, але через ті долі секунди, на які затримується стрибок (я натискаю на кнопку стрибка, але персонаж стрибає тільки через пів секунди), гравці іноді просто божеволіють.

«Хороша гра - це предмет, який приносить фан. Фан - це задоволення з сюрпризами.» [14]



Рисунок 3.5 – Головне меню гри

Головне меню в Fisherman буде виглядати як на рисунку 2.5. С меню гравець може зайти до 2 режимів гри та переглянути свої результати або по всьому світу.

Графіка в грі буде цілком фотореалістичною, нижче буде наведено моделі риб.



Рисунок 3.6 – Модель риби №1

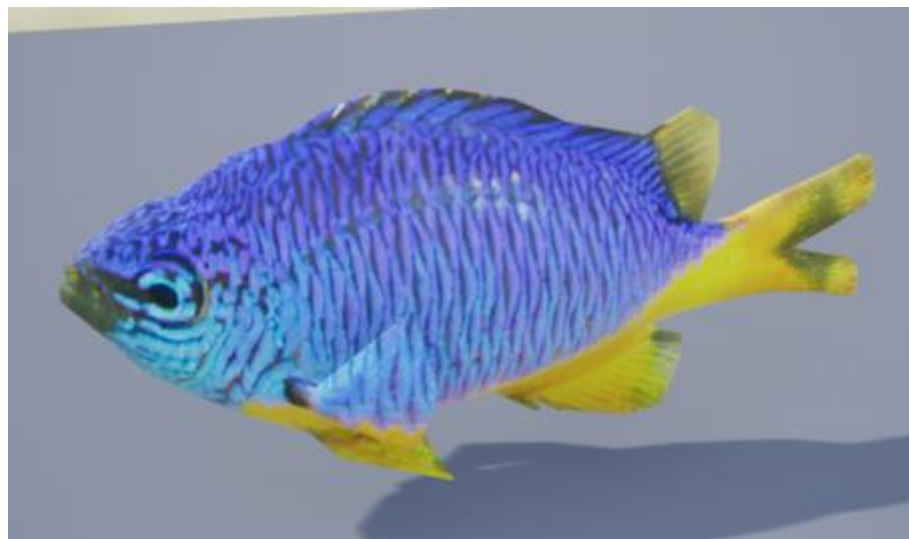


Рисунок 3.7 – Модель рыбы №2



Рисунок 3.8 – Модель рыбы №3



Рисунок 3.9 – Модель рыбы №4

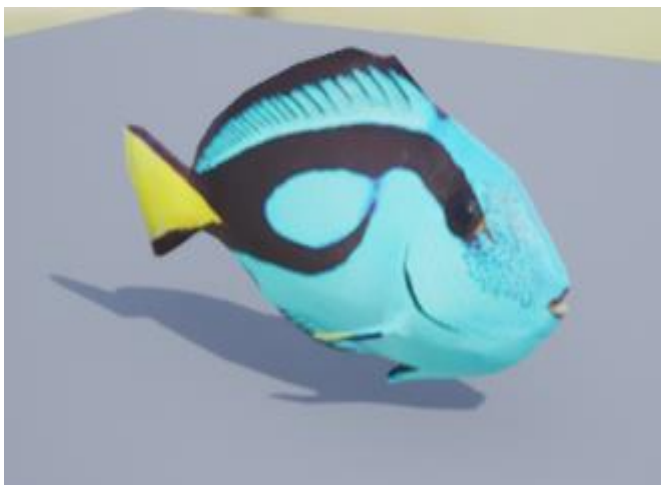


Рисунок 3.10 – Модель рыбы №5

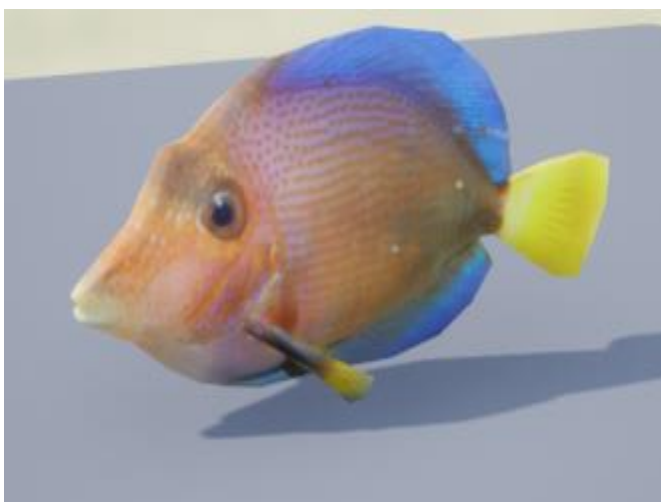


Рисунок 3.11 – Модель рыбы №6



Рисунок 3.12 – Модель рыбы №7

3.3 Реалізація ігрових механік

Побудуємо діаграму (рис 3.13) виділивши при цьому основі дії гравця. Гравець починає утримувати кнопку та закидати контролер щоб гачок полетів у воду. Далі треба дочекатися поки риба схопить гачок. Коли риба схопила гачок, гравцю потрібно крутити гачок завдяки контролеру і при витягуванні риби із води гравець вибере або «до кошика» або «відпустити». При закінченні часу гравця кидає до головного меню.

Ігровий цикл або Gameplayloop – зациклений набір ігрових механік, які визначають базовий ігровий досвід гравця.

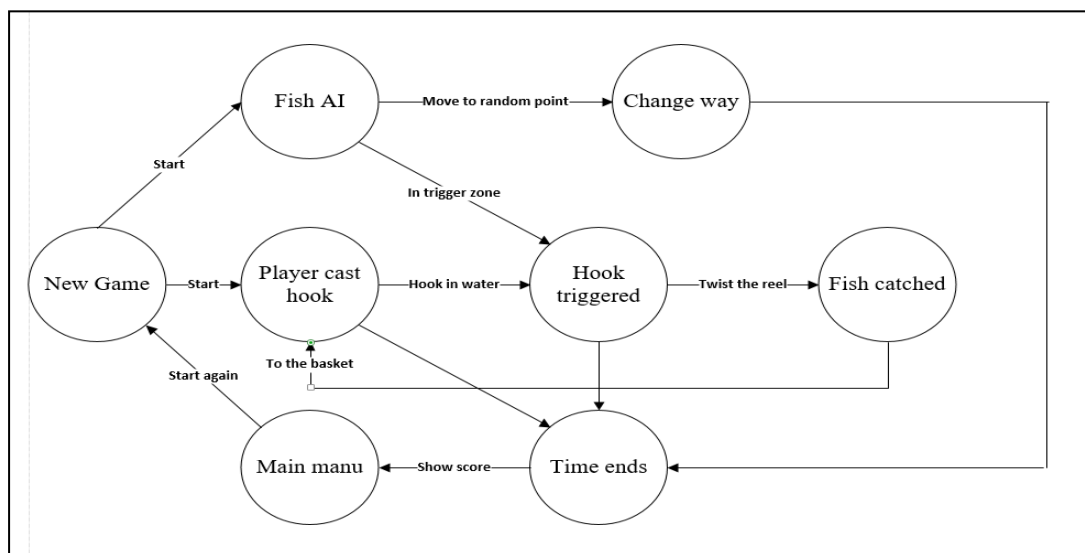


Рисунок 3.13 – Ігровий цикл

GameplayLoop – це ядро гри яку геймдизайнер може доповнювати та розвивати. Можна налаштовувати нові ключові механіки, розвиваючи їх до більш складних. Чим складніша гра, тим більше виходить циклів. Рисунок 3.14. приклади алгоритмів та методів Blueprints

Блупрінти – це візуальна скрипкова мова, який дозволяє написати логіку гри без застосування мов програмування. Яким би складними або простим він не здавався, він залишається досить таки потужним інструментом, на якому можна створити майже що завгодно, від простенького персонажа або відкриття дверцят до процедурної генерації рівня[12].

Повторення є одним з фундаментальних аспектів гри. Коли люди отримують задоволення від чогось, то вони хочуть це повторити. Наприклад, діти

надзвичайно люблять переглядати мультфільми знову і знову, або постійно запитують, щоб їм прочитали одну і ту ж казку на ніч. Аналогічний процес відбувається для більш старшого віку і даний аспект переноситься на побудову ігрової механіки геймдизайнером. Під час розробки гри основний ігровий цикл (англ. Coregameplayloop) є центральним блоком, на підставі якого будується гра. Зазвичай він представлений у вигляді дієслів: вистрілити, стрибнути, подивитися стан і т. П. В ідеальному випадку ігровий цикл може бути описаний кількома словами, які відображають той досвід, який отримує гравець [1].

З наукової точки зору ігровий цикл може бути розглянуто як послідовність «гіпотеза, експеримент і спостереження результатів». Іншими словами, гравець має у себе деяку ментальну модель ігрового світу і спосіб, за допомогою якого можна вивчати цей світ. Відповідно, для вивчення і розуміння необхідно зіставляти наявну у нього модель світу з реальним ігровим світом за допомогою ігрових дій (як експерименту), і для більш докладного вивчення необхідно виконати безліч ітерацій, тим самим виконуючи ігровий цикл. При цьому якщо гравець все знає про ігровий світ, то той йому стає нецікавим (важливо зберігати деяку невизначеність). Таким чином, розглядаючи ігровий цикл можна управляти тим досвідом, який отримує гравець.

3.4 Blueprints

Blueprints– це система візуального скриптинга Unreal Engine 4. Вона є швидким способом створення прототипів ігор. Замість порядкового написання коду все можна робити візуально: перетягувати Ноди (вузли), задавати їх властивості в інтерфейсі і з'єднувати їх «проводи» [15].

Наша система буду повістю побудована на візуальному скриптіngu. Ігровий движок Unreal Engine 4 дозволяю побудувати повноцінну гру та не використовувати навик програмування.

За допомогою Блупрінтов, розробники можуть створювати такі речі, як:

Ігрові режими. Встановлювати правила гри, змінити те, як ігри в загальному плані

Гравці. Призначати гравців, надавати їм особливі рис і вид.

Камери. Створювати види для огляду і змінювати властивості камер в реальному часі

Управління. Призначати кнопки, для управління персонажем, автомобілем, або зовнішньому рівнем

Речі. Зброя, підбираються предмети та інше

Оточення. Створення випадково генерованого оточення

3.4.1 Взаємодія різних елементів рівня з водою

На рисунку 3.14 можна побачити взаємодію з водою на рівні. OnComponentBeginOverlap дозволяє Actor закинути вудку у воду. Якщо гачок потрапляє у воду починає програвати звук попадання у воду, а також є звуки розмотування катушки.

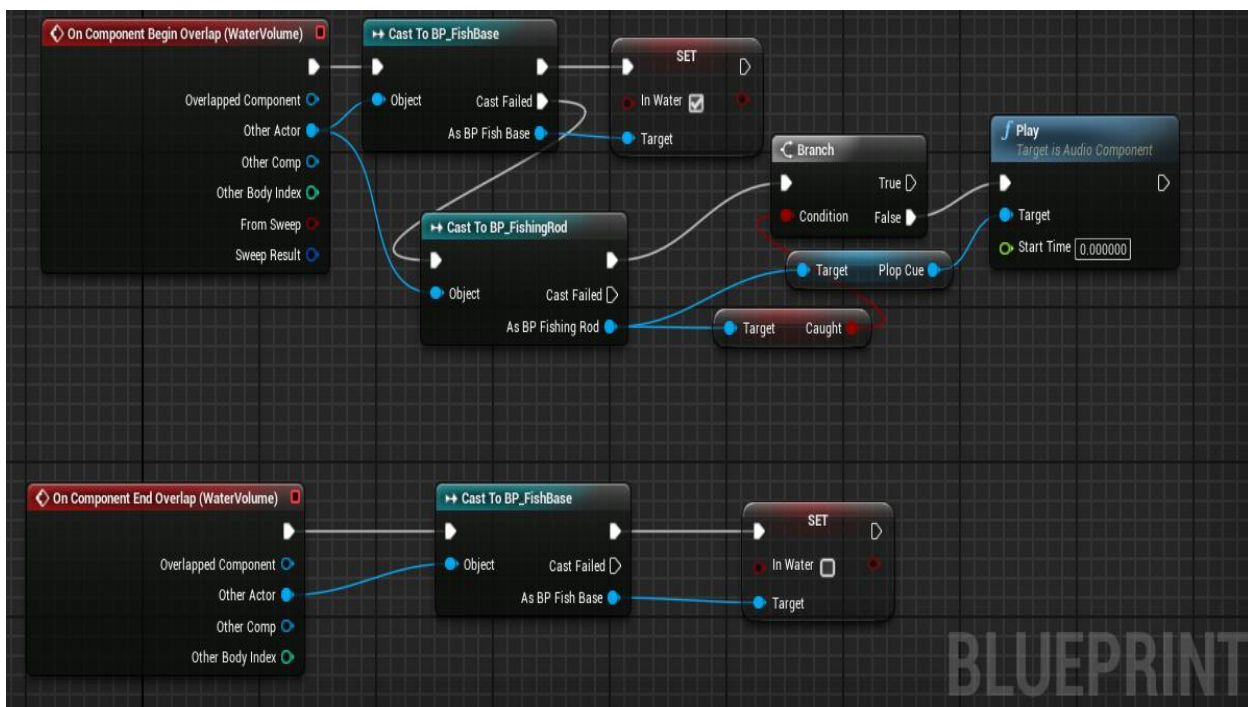


Рисунок 3.14 – Взаємодія з водою

Коли риба потрапляє на гачок починає програватись звук того що риба на гачку, це можна побачити на CastToBPFishBase.

Звуки припиняють програватись коли у воді.

3.4.2 AI та поведінка риби

Під час EventBeginPlay починається гра та риба починає рухатись до випадкової точки у радіусі 2000.

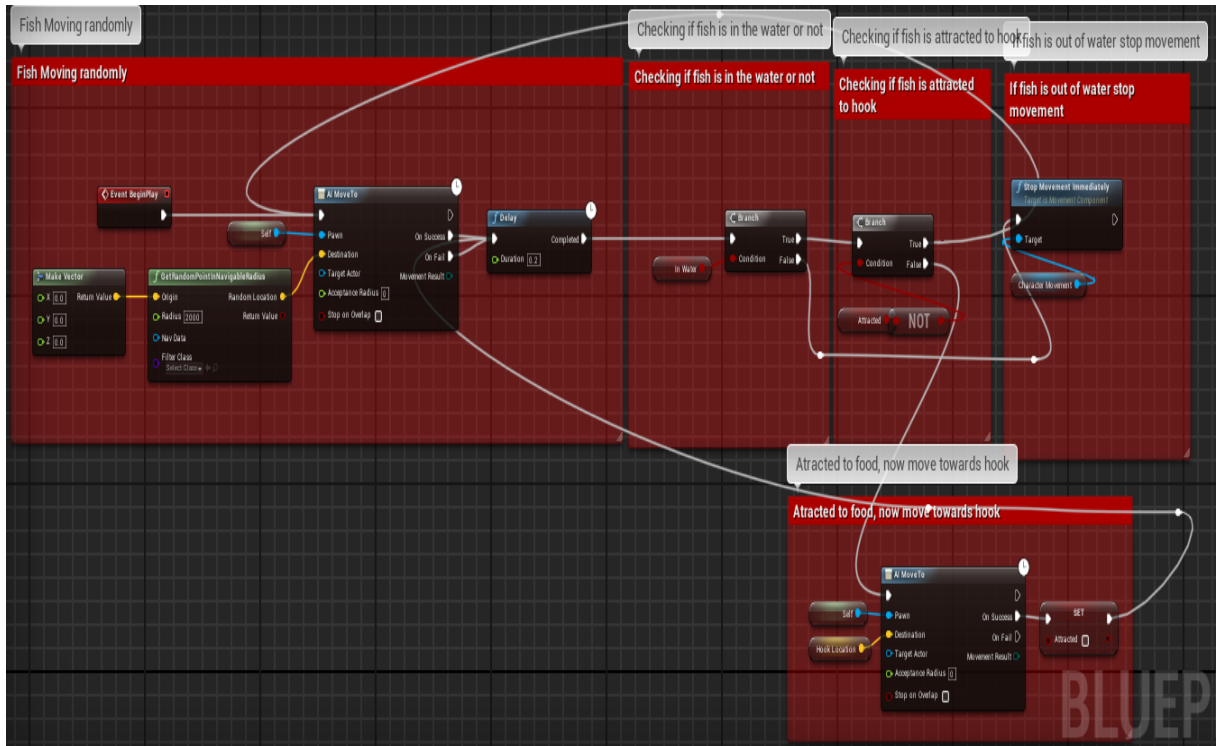


Рисунок 3.15 – Штучний інтелект риби

Далі іду перевірка чи є риба у воді, якщо так вони рухаються далі, якщо ні то зупиняються. Коли риба у воді та з'їла гачок, лестка починає рухатись за рибою, зображено на рисунку 3.15.

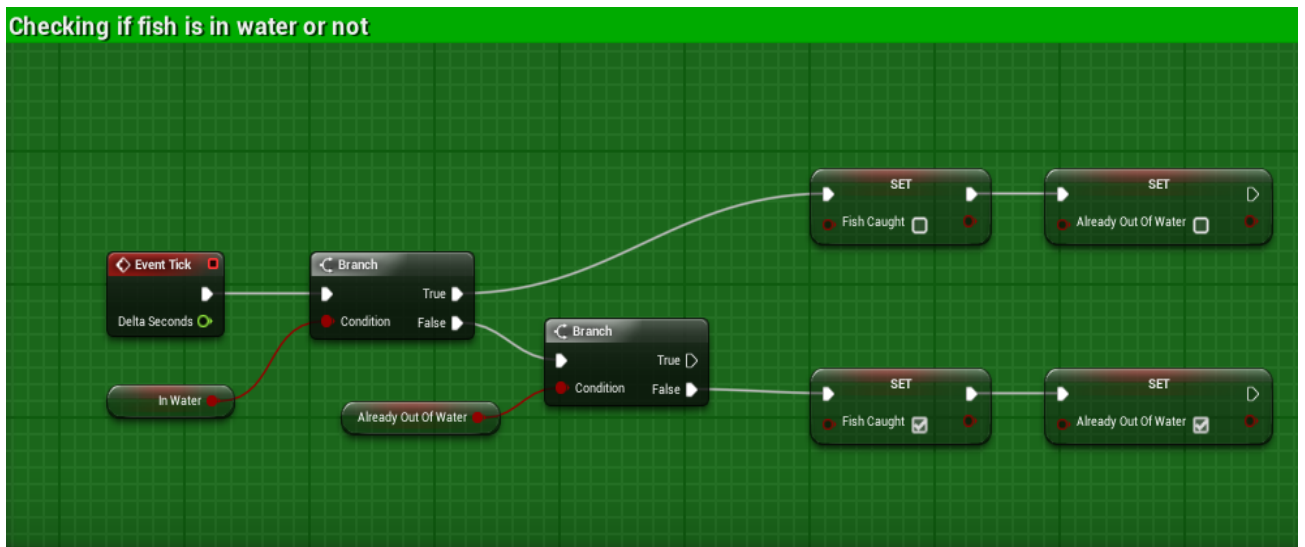


Рисунок 3.16 – Перевірка риби «У воді чи ні»

На малюнку 3.16 зображено як проходить перевірка у воді чи ні, кожен кадр гри перевіряється та встановлює зміни для поведінки у AIController.

3.4.3 Логіка та поведінка вудки

Під час початку гри до ігрового персонажу підключається CharacterController завдяки якому гравець зможе пересуватися та виконувати дії. Під час Cast починається закидання вудочки, максимальна дальність 1000.

EventsReelUp та ReelDown це події які закручують або спускають лестку відповідно. Коли відбувається подія ReelUp програвється звук, далі іде перевірка чи є риба на гачку, коли вона на гачку швидкість закручування прораховується на основі ваги риби яку піймали. Також на рисунку 3.17 видно FishCaught та FishAttachedtowardshook.

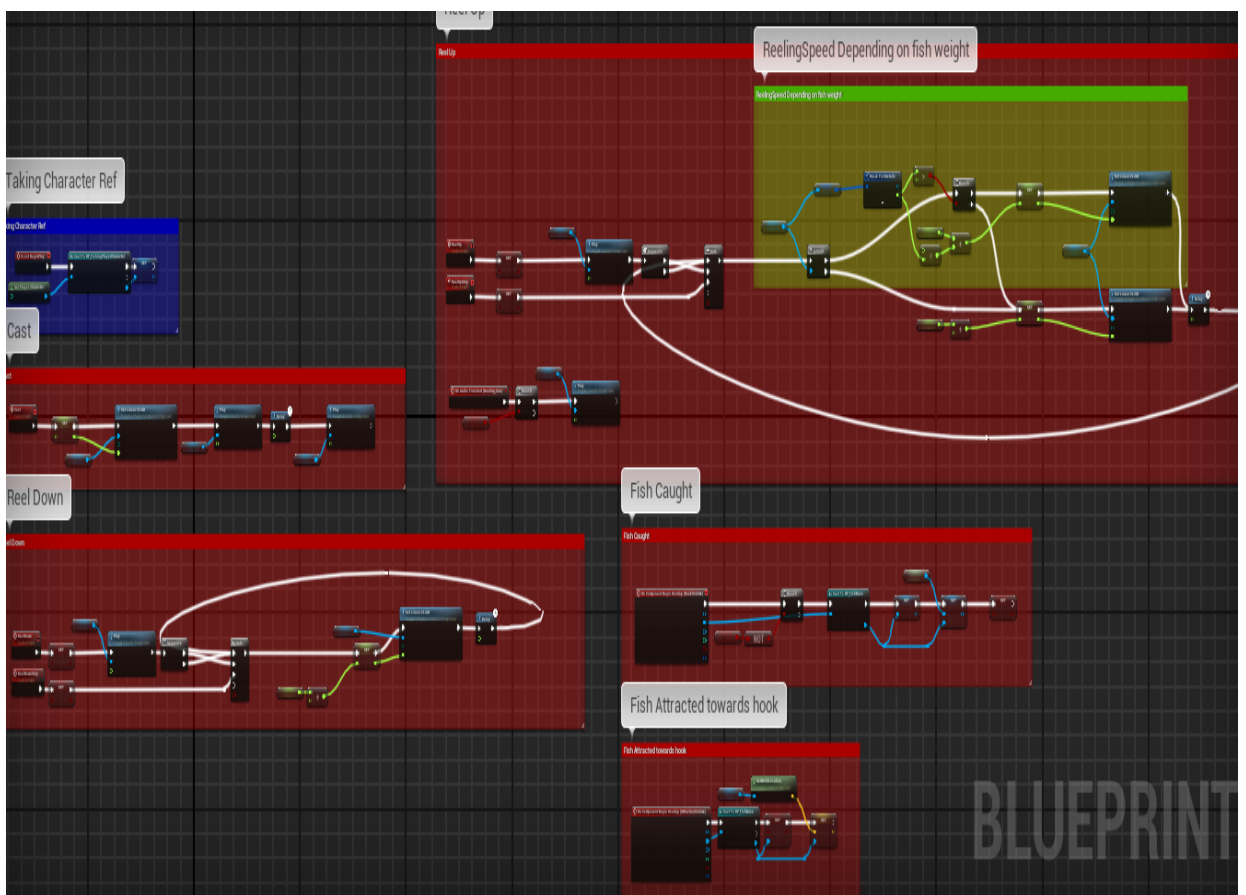


Рисунок 3.17 – Логіка вудки

Коли риба хвтає гачок іде перевірка положення риби та гачка, що дозволяє створити ефект того що риба на гачку. Цю функцію виконує FishAttachedtowardshook. FishCaught починається коли риба уже на гачку та гравець її витягнув, функція показує інформацію про рибу – вага, рідкість, назву та кількість балів яку вона дасть.

3.4.4 Алгоритм генерації риби

Алгоритм буде генерувати на рівні випадкову кількість риби які будуть відрізнятись одна від одної, їх шлях по рівню прораховується іншим алгоритмом який дозволить гравцю побачити реальну рибалку. Кожна риба буде приносити різну кількість балів

```
class AFishManager : public AActor
{
    GENERATED_BODY()

public:
    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    TArray<UClass*> flockTypes;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    TArray<float> numInFlock;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    float minZ = -9000.0f;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    float maxZ = -950.0f;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    float underwaterBoxLength = 10000;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    bool attachToPlayer = false;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    bool DebugMode = false;

    UPROPERTY(EditAnywhere, BlueprintReadWrite, Category = Config)
    UClass* playerType;

    AActor* player;
    // get min z
    float getMinZ();
    // get max z
    float getMaxZ();
    /** Constructor */
    AFishManager(const FObjectInitializer& ObjectInitializer);
};
```

Рисунок 3.18 Приклад алгоритму генерації риби

У якості приклада реалізації системи розглянемо ігровий мір у жанрі симулятора. Користувач перед початком відповість де він хоче рибалити. Під час завантаження рівню виконається алгоритм. На основі цих даних ігровий мір заповниться різною кількістю різноманітних риб. Частота появи буде повністю випадковою

3.4.5 Реалізація управління персонажем через контролер.

Викличемо подію "MotionControllerTrigger" для правої руки і поставимо логічне значення (true / false) для натискання і відпускання кнопки спуску, рисунок 3.19:

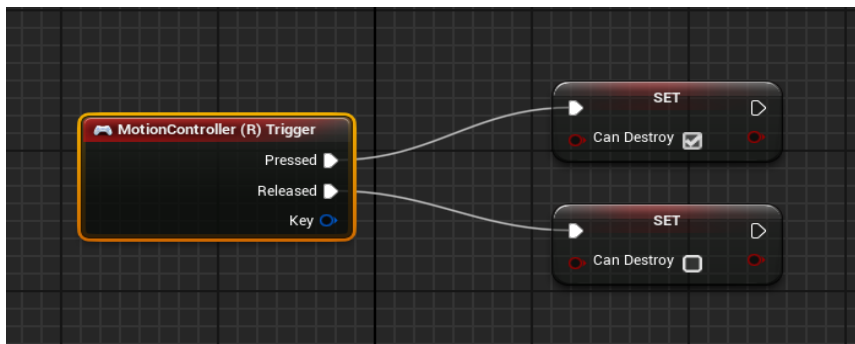


Рисунок 3.19 – Реалізація однієї руки

Для події Tick створимо лінію прицілювання (linetrace), щоб знати, на який об'єкт ми наводили. Для цього потрібно задати початкову і кінцеву точки. Для початкової точки ми пов'яжемо RightController, MotionController і отримуємо GetWorldLocation.

Також з MotionController пов'яжемо GetForwardVector. Таким чином ми будемо отримувати напрямок прицілювання контролером. Множимо цей вектор на необхідну дистанцію (скажімо, на 1000 = 10 метрів) і додаємо до початкового місця розташування, тим самим отримавши кінцеву точку.

Додаємо модуль LineTraceByChannel, з'єднуємо початкову і кінцеву точки, у вкладці DrawDebugType вибираємо OneFrame. Цей параметр означає відрисовку лінії між початковим і кінцевим пунктом.

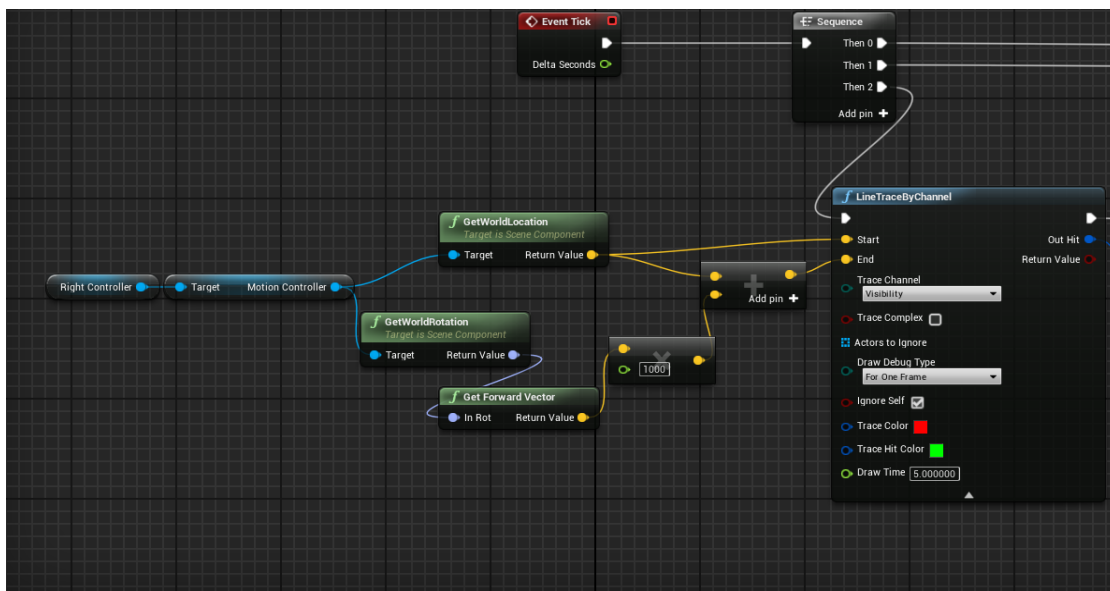


Рисунок 3.20 – Відрисовка лінії захвату цілі

Далі переходимо до вихідних даних BreakHitResult, для HitActor робимо перехід CasttoBP_PickupCube. Це означає, що на рівні ми можемо вибирати тільки сині куби.

Тепер кидаємо гілку і перевіряємо логічне значення для раніше створеного CanDestroy. Якщо воно дорівнює false, то ми нічого не робимо. Якщо true, то викликаємо модуль DestroyActor для знищення обраного об'єкта.

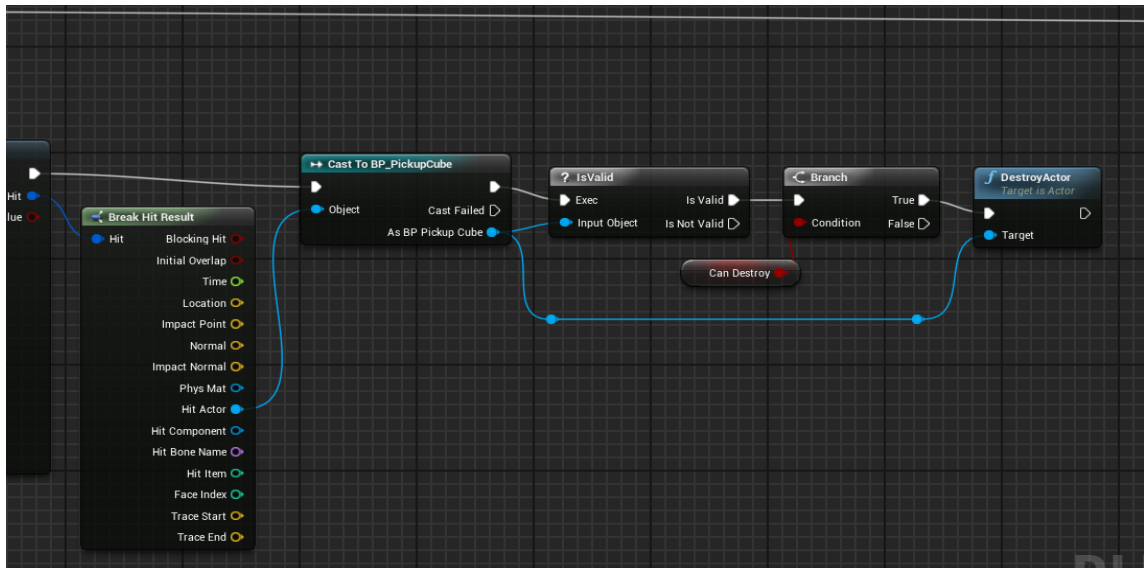


Рисунок 3.21 – Риба знищується

3.4.6 Створення телепорту для руху персонажа

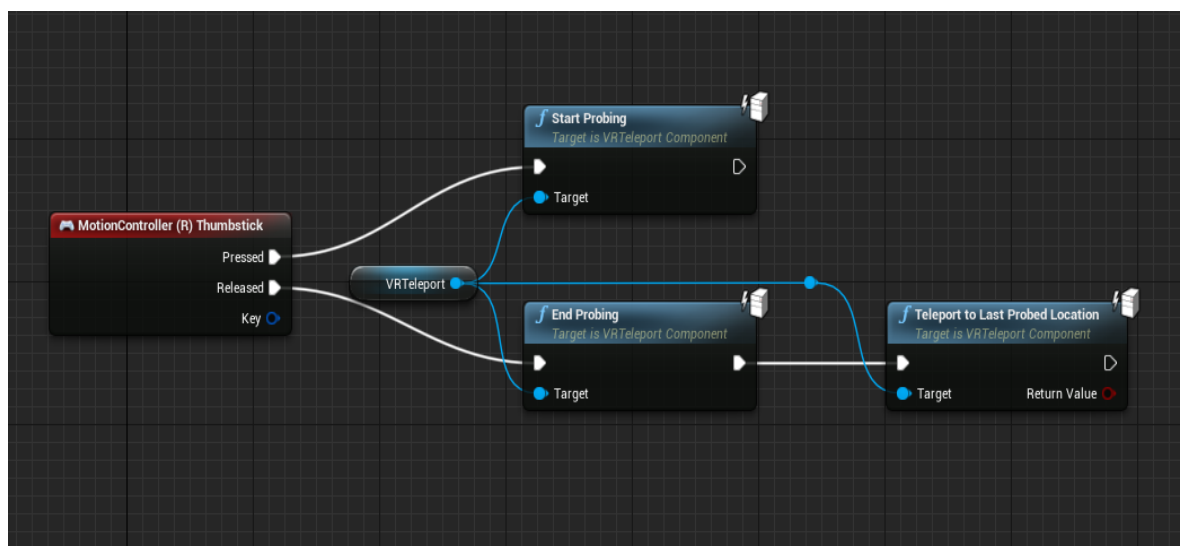


Рисунок 3.22 – Підключення входу контролера до компонента телепорту.

Для делегування подій з відповіддю ви можете встановити параметр відповіді `ShouldPerformDefaultImplementation` на `false`, якщо ви хочете обійти поведінку компонента за замовчуванням. Прикладом може бути виконання телепортного ефекту, відмінного від типового:

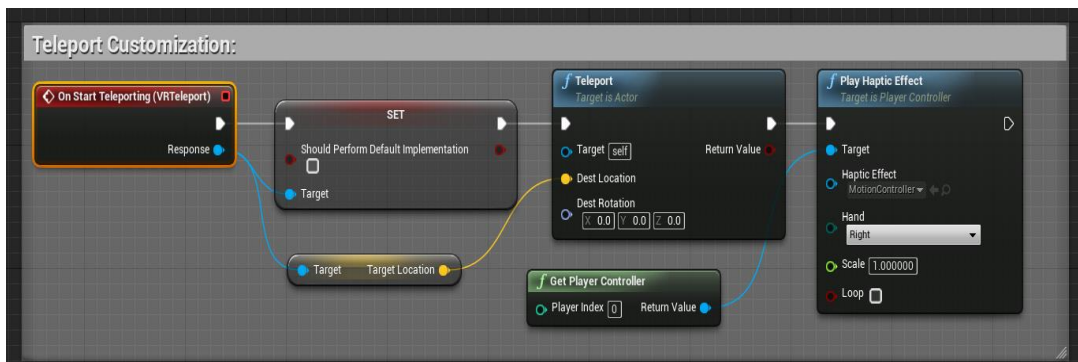


Рисунок 3.23 – Обхід поведінки телепорту за замовчуванням та реалізація `custombehavior`

Телепортація починається, коли Player запускає події `TeleportLeft` або `TeleportRight` натисканням на мініатюру контролера руху. Ці події реалізовані у програмі `MotionControllerPawn`. На наступному скріншоті відображається подія `TeleportLeft`. При натисканні він активує телепортерну дугу лівого контролера та вимикає телепортерну дугу правого контролера, оскільки лише один може бути активним. Гравець вказує на потрібне місце, і коли вони відпускають мініатюру, викликається подія `ExecuteTeleportaMotionControllerPawn` для виконання ефекту камери та телепортування гравця:

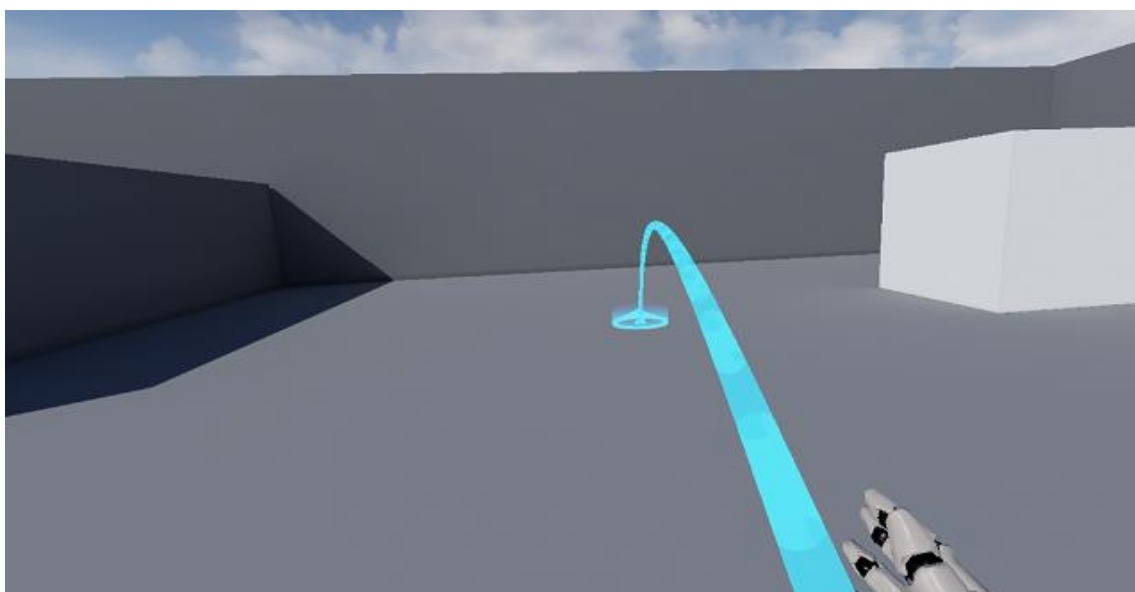


Рисунок 3.24 – вигляд телепорту гри

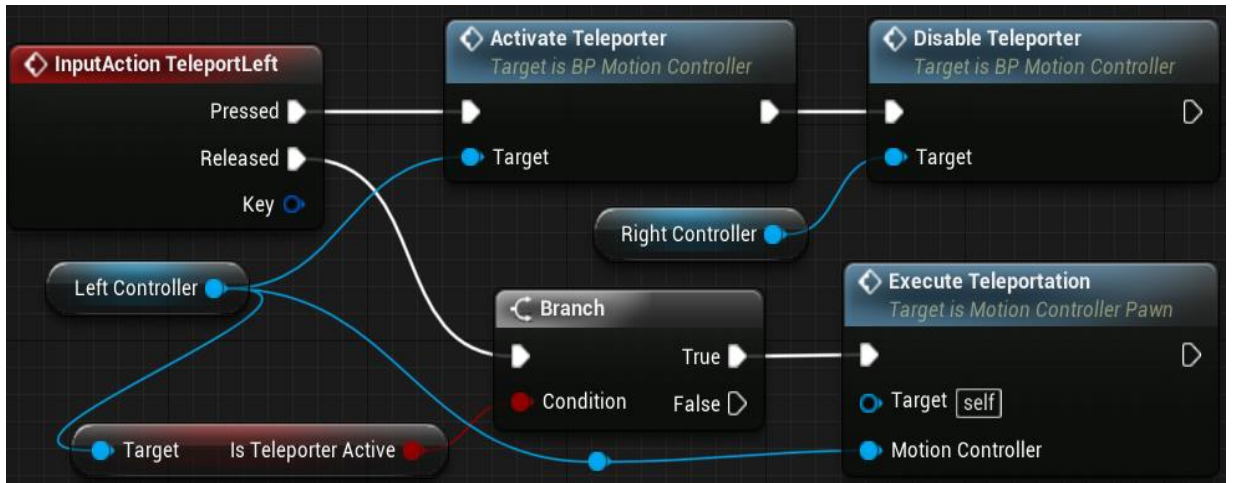


Рисунок 3.25 – Телепорт гравця на Blueprint

3.5 Реалізація інтерфейсу HUD

HUD (head-up чи heads-updisplay, «хад») – є частиною графічного інтерфейсу гравця, яка дозволяє отримати різноманітну ігрову інформацію не викликаючи додаткові меню. Свою назву отримала через схожість з технологією, яка була розроблена для сучасних військових літаків та дозволяє пілотові бачити важливу інформацію прямо на лобовому склі.

Основні типи подібної інформації:

- текстові індикатори очок життя (здоров'я), запасу амуніції, досвіду, часу, навички, назви локації, завдання, субтитри і т.п;
- міні-карта;
- спрайтові і текстурні наложенія: покажчик прицілу, піктограми зброї і предметів, радар, індикатори у вигляді смуг, шкал, графіків, також бризки крові або інших рідин (на обличчі грає), подряпини і кульові отвори на шоломі, відблиски від джерел світла і т.п;
- Тривимірні об'єкти, які є частиною гравця: моделі зброї з тримають їх руками, в деяких іграх також ніг, скафандра, шолома, керованої машини, літака, човна і т.п;

Часто елементи HUD розташовуються по периметру екрану і (або) напівпрозорі, щоб не заважати основному процесу гри.



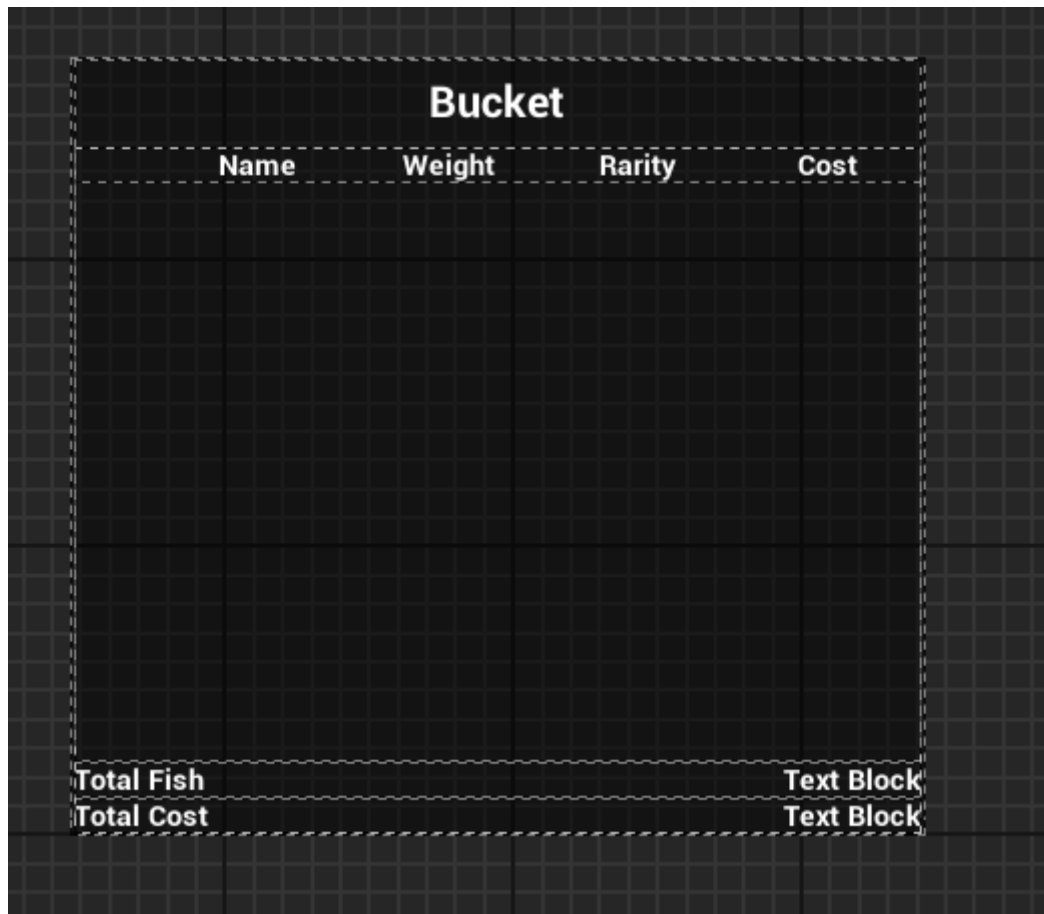
Рисунок 3.26 – HUD гри

На рисунку 3.26 бачимо що гравець може побачити при будь якому положенні камери, а саме невелика свідка з управлінням у грі у верхньому лівому куті екрана. Кількість пійманої риби та кількість балів підраховується у верхньому правому куті екрана



Рисунок 3.27 – Піймав рибу

Коли гравець піймає рибу він побачить детальну інформацію про неї – назву, вагу, рідкість та ціну. Гра буде давати вибір або випустити, або покласти до кошика з рибою на рисунку 3.27.



3.28 – Дизайн кошика

Коли гравець піймає риби він побачить слот з рибою, рисунок 3.29



3.29 – Слот риби

На рисунку 3.29 слот риби у кошику буде показувати усю інформацію про пійманих риб.

4 ЕТАПИ ПРОЕКТУВАННЯ ГРИ ТА СТВОРЕННЯ ІГРОВИХ РІВНІВ

Проектування гри ділиться на вісім етапів [16]:

1. Підготовка письмової інструкції, т. Е. Специфічного набору вимог, обмежень, передбачуваного результату гри. До процесу безпосередньої розробки гри групі розробників необхідно чітко визначити мету - вид інформації і засоби її уявлення. Розробка специфікації до проекту гри дозволяє значно скоротити процес розробки гри і передбачити можливі перешкоди. Крім того, це дозволяє замовнику вносити необхідні корективи на початковому етапі створення гри.

2. Детальний опис системних компонентів проблеми, т. Е. Вичерпний опис системної термінології з даної проблеми. Зазвичай це досягається за допомогою невеликих карток, використовуваних для визначення окремої ідеї, поняття «комплексної реальності» - складної, яка взаємодіє або динамічної системи, абстрактної або конкретною. Картки готуються за допомогою брейнстормінга («мозковий атаки»), перегляду літератури, опитування експертів. Потім ці картки можуть бути зведені в таблицю основних понять, концептуальну карту, тривимірну модель, перфокарту для ЕОМ і т. П.

3. Вибір компонентів, складових проблему, для використання в грі. Мета ігрового моделювання - забезпечити основу для організованого спілкування з комплексної проблеми. Для цього необхідно виокремити з кола проблем або з системи ті складові, які вимагають подальшого обговорення. При цьому керуються розробленою раніше специфікацією. Зайва деталізація проблеми небажана.

4. Планування включення системних компонентів в гру. На цьому етапі необхідно спланувати, як включити системні компоненти в елементи гри. Це можна здійснити за допомогою матриці ігрових елементів, яка показує, яким чином даний системний компонент може бути включений в елементи гри: сценарій, імпульс, послідовність циклу, етапи гри, правила, ролі, модель, послідовність рішень і зв'язку, систему обліку, покажчики, символіку, обладнання. Для системного відтворення будь-якої проблеми в грі необхідно включити компоненти проблеми в один або кілька елементів гри. Цей процес «картування» проблеми в матрицю ігрових елементів дає кілька результатів:

- фіксує прийняті рішення;

- дозволяє врахувати всі компоненти проблеми;
- вже на початковому етапі розробки гри дозволяє спланувати, яким чином проблема буде представлена в грі;
- забезпечує чернетка проекту гри.

5. Підсумовування змісту кожного елемента гри. Для побудови гри необхідно визначити два параметри кожного елемента: основний зміст і ігрові механізми для подання цього змісту в грі. Для опису змісту кожного елемента гри необхідно підсумувати всі нотатки з осередків кожного стовпця матриці. Наприклад, в стовпці «ролі» все рядки матриці, системно описують проблему, покажуть, кого з гравців необхідно включити. Таким чином, сума по цьому стовпцю представить повний список ролей, які необхідно використовувати.

6. Вибір ігрових механізмів з особистого репертуару технік розробників і осіб, що використовують гру.

7. Конструювання гри і її апробація. Гра може вважатися готовою тільки після того, як буде проведена не менше 10 разів з різними групами (так зване «правило десяти») і останні три гри не зажадають істотних змін. (В навчальних цілях проводиться одноразове випробування гри.)

8. Оцінка гри. Основою для оцінки гри служать специфікації і результати гри (час освоєння, загальний час, труднощі, ступінь включення учасників в гру) [16].

4.1 Можливості розробки гри в середовищі Unreal Engine 4

Віртуальне середовище являє собою тривимірну модель уявного навчального корпусу, зайнятого приміщеннями. Велике значення при організації взаємодії 3d-об'єктів в просторі має виявлення зіткнень (Collision detection). Collision detection – це спосіб, за допомогою якого аналізується 3D-простір сцени на предмет зіткнень між об'єктами. Привласнюючи об'єкту компонент Collider, ми фактично розміщуємо навколо нього невидиму сітку - так званий колайдер, який імітує форму об'єкта і інформує про наявність зіткнення з іншим об'єктом. Наприклад, в грі-симуляторі боулінгу кулі будуть мати просту сферичну форму колайдера (Sphercollider), в той час як у об'єктів-кеглів колайдер буде мати форму циліндра/капсули або, для більшої реалістичності зіткнень, буде використовувати меш (mesh), який є не чим іншим, як описом геометрії 3d-

моделі. Інформація про зіткнення колайдерів надходить в фізичний движок, який повідомляє про зіткнення об'єктів і їх подальшу реакцію на це зіткнення, засновану на напрямку і силі удару, швидкості та інших факторах. Використання колайдерів, які повторюють форму меша моделі, з одного боку, дає більш точне визначення зіткнень, але в той же час призводить до збільшення витрат на їх обчислення [15].

Для того щоб змусити взаємодіяти між собою наявні в сцені тривимірні моделі, треба створити скрипт на мові C++ або використати візуальний скрипт Blueprints. Далі потрібні будуть скрипти, що описують управління персонажем в тривимірному робочому середовищі. Перше, що забезпечує інтерактивність, це можливість огляду 3D простору за допомогою датчиків руху голову. Підключаємо за допомогою даного скрипта. Цей скрипт прив'язується до об'єкту «камера» і обертає його в сторону, в яку зміщується голова, відповідно до чутливості. Якщо значення кута, прийняте камерою, виходить за рамки, зазначені в максимумі і мінімумі, то воно буде повернуто до допустимого значення. Обертання здійснюється стандартною функцією `transform.rotate`, застосованої до об'єкту «камера».

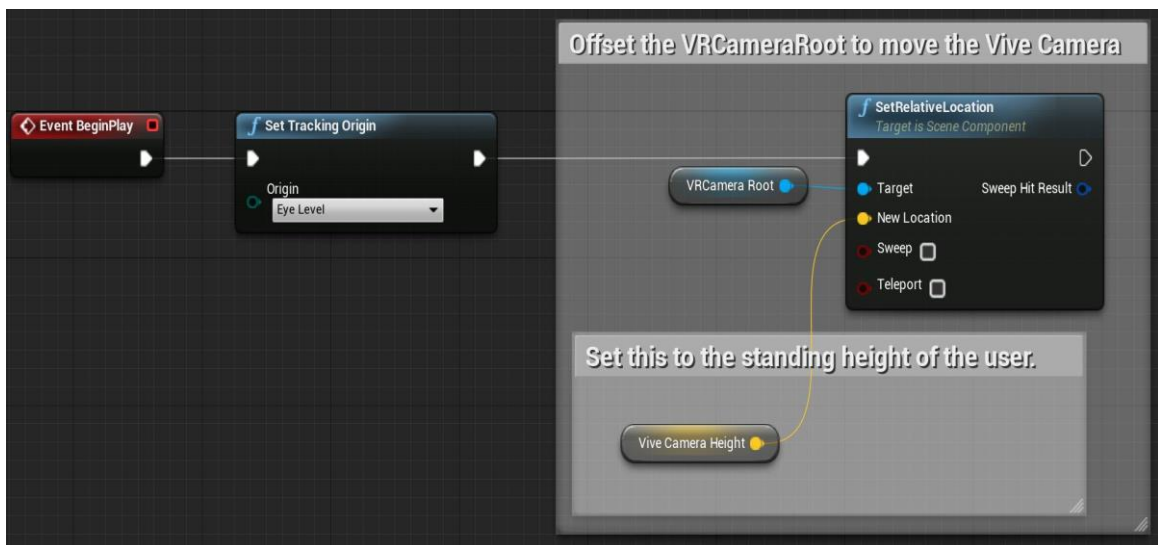


Рисунок 4.1 – Створення камери для шолому гравця

Таким чином досягається збільшення швидкодії, що разом з використанням запечених текстур і інших спрощень дозволяє досягти гарних результатів оптимізації

4.2 Крива складності

Крива складності – це функція залежності навичок гравця від часу проведеного у грі. Якщо використовувати математичну формулу то її можна представити у вигляді $s = f(t)$, де S – навички, t – час [14].

Крива складності дає наочне уявлення, наскільки швидко гравець навчається та перевіряє щоб гравцю не наскучила сама гра. Добре збалансована крива складності вводить гравця у потоковий стан.

Потоковий стан – це стан у якому гравець максимально зосереджується на самій грі та отримує задоволення від процесу гри. Коли гра дуже складна то гравець відчуває це та йому вона починає набридати. Коли гра дуже легка то гравець не відчуває певного виклику який повинен бути присутній майже в усіх іграх. Приклад кращого потокового стану показаний на рисунку 4.2.

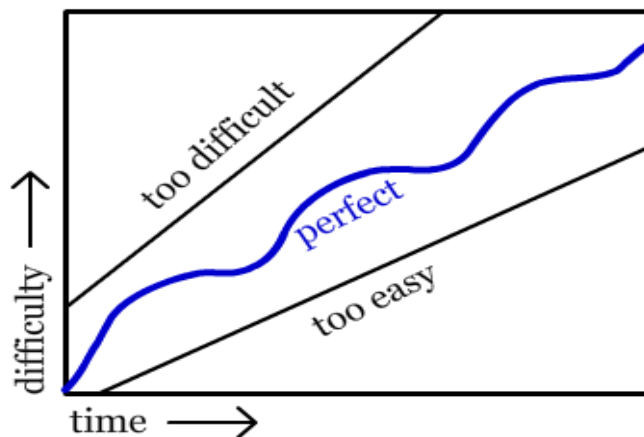


Рисунок 4.2 – Потоківий стан

Отже, можемо зробити висновок що крива складності формується за допомогою наступних даних, а саме:

- Користувачі – усі гравці які завантажили ігровий додаток;
- Рівні у грі – гра має складатися з певних рівнів за допомогою яких можна відстежити прогрес гравця;
- Ігрове життя – ігрова одиниця за допомогою якої можна відстежити де саме гравець витрачає ігрове життя щоб виявити складні моменти гри та проаналізувати їх;
- Ігрове оточення – ігровий простір на якому відбуваються дії;

– Ігровий жанр – характерні особливості які притаманні грі формують ігровий жанр.

4.3 Перший рівень

Перший рівень буде самим легким з усіх рівні. На ньому не будуть присутні ускладнюючі ігровий процес елементи та механіки гри. Це надасть змогу людині, яка вперше грає або використовує VR шолом та контролери пристосуватися до віртуальної середи.

Персонаж буде знаходитись на платформі по середини кімнати, а навколо нього буде вода, де він зможе ловити рибу. Людина зможе вибрати любий напрямок ловлі.

Поведінка риби буде досить проста, спочатку рівня генерується різна кількість та видів риби. Далі вони будуть рухатись по заданій площині від точки А до Б, при проходженні заданого напрямку, надається нова координата.

Гра закінчується після того, як вийде час на таймері. Результат гри буде внесено у таблицю.

4.4 Другий рівень

Другий рівень буде важчим для гравця. Візуально рівень буде таким самим, але тепер на ньому з'являться нові механіки. Навколо платформи будуть розставленні «Кущі очерета» на яких може застрягти вудочка, але 1 вільною рукою гравець зможе взяти та зняти крючок. Таким чином ускладнюємо гру, кожна помилка буде витрачати час людини, що призводить до втрати загальної суми балів в кінці гри.

Поведінка риби буде така ж сама для комфортної гри. Гра завершується після того як вийде час на таймері. Результат гри буде занесенню у таблицю.

4.5 Третій рівень

Третій рівень матиме усі механіки та елементи гри які ускладнюють гру але гру ускладнить наступне:

1) Тепер у риби буде різна характеристика яка впливатиме на її вартість у «Балах» в таблиці. Цих методом ми можемо створити ілюзію реального улову в нашій грі-симуляторі

2) Більш пророблений ШІ для риби. Тепер риба, яка за розміром вийшла велика, почне плавати як надалі від гравця, в неї буде маленький шанс підплисти до платформи.

4.6 Четвертий рівень

Цей рівень створений для відтворення реальних умов. Тепер персонаж буде знаходитись на березі річки. Початкове положення буде на дерев'яному мосту а по різних бокам кущі.

С особливостей рівня буде відро у яке потрібно складувати рибу. Тепер гравцю необхідно знімати з крючка рибу. Також риба може впасти у воду якщо її впустити(віджати кнопку на контролері)

5 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ

Використання ідеї застосування віртуальної реальності в навчанні на симуляторах має широке застосування як серед навчальних закладів або спеціальних установ, які працюють за принципами звичайних методологій, так і для тих, хто використовує системи віртуальної реальності вдома. Цей підхід був розроблений не тільки для того, щоб дітям в дома було цікаво грати в віртуальній реальності, а й для того, щоб дорослі використовувати цей застосунок для поліпшення своїх навичок.

5.1 Проведення аналізу даних

Будемо класифікувати ідею за такими характеристиками.

- економічна;
- технічна;
- естетична
- інтерфейс, зручність;
- ергономічна;
- забезпечення безпечного середовища;
- швидкість досягнення результату;
- надійність

Таблиця 5.1 Сильні і слабкі сторони ідеї проекту

№	Характеристика ідеї	Слабка сторона	Сильна сторона
1	Економічна	Немає коштів на маркетинг і немає клієнтів для пошуку і формування ринку збуту	Дешева реалізація, гнучкість розробки і рішення безпосередніх завдань освіти
2	Технічна	Необхідні потужні комп'ютери та дороге додаткове обладнання.	Використання великого набору функціоналу і можливість гнучкого налаштування.

3	Естетична	На фоні інших аналогів поступається в візуальних рішеннях	Має близьку до реальності візуалізацію
4	Ергономічна	Немає	Має простий та зрозумілий інтерфейс для будь якого користувача, не потребує налаштування
5	Безпечне оточення	Довге використання може спричинити морську хворобу	Не потребує багато місця ігрової зони, гравець може телепортуватися стоячи на одному місці
6	Швидкість досягнення результату	Немає	Так як гра відбувається на час, гравцю необхідно швидше пристосовуватися до нових механік.
7	Надійність	Збої у роботі можуть вплинути на справність і коректність роботи системи	Як тільки програма встановлена вірогідність на збій дуже мала

Таблиця 5.2 Аналіз фактору можливостей

№	Фактор	Зміст можливостей	Можлива реакція
1	Поява нових технологій за допомогою UE4	Розширення спектру можливостей для аналізу користувача і додавання методів забезпечення симуляції	Впровадження нових технологій, оптимізація і стабілізація роботи програми
2	Розширення можливостей віртуальної реальності	Додавання зручності і сегментів, де можна застосовувати нові ідеї	Розширення сегмента застосування
3	Збільшення кількості бізнес процесів	Новим гравцям потрібні такі додатки	Створення та надання послуг

		для моніторингу активності користувача	
4	Поява нових технологій віртуальної реальності для симуляторів	Можливість впровадити ці технології в існуюче розширення або розробка нових рішень	Вивчення цих технологій і їх інтеграція з існуючими рішеннями

5.2 Аналіз результатів

Є багато способів визначити формуючі гру елементи. Сама вдала схема зображена на рисунку 5.1, яку назвав Джессі Шелл «елементарна тетрада»[14].

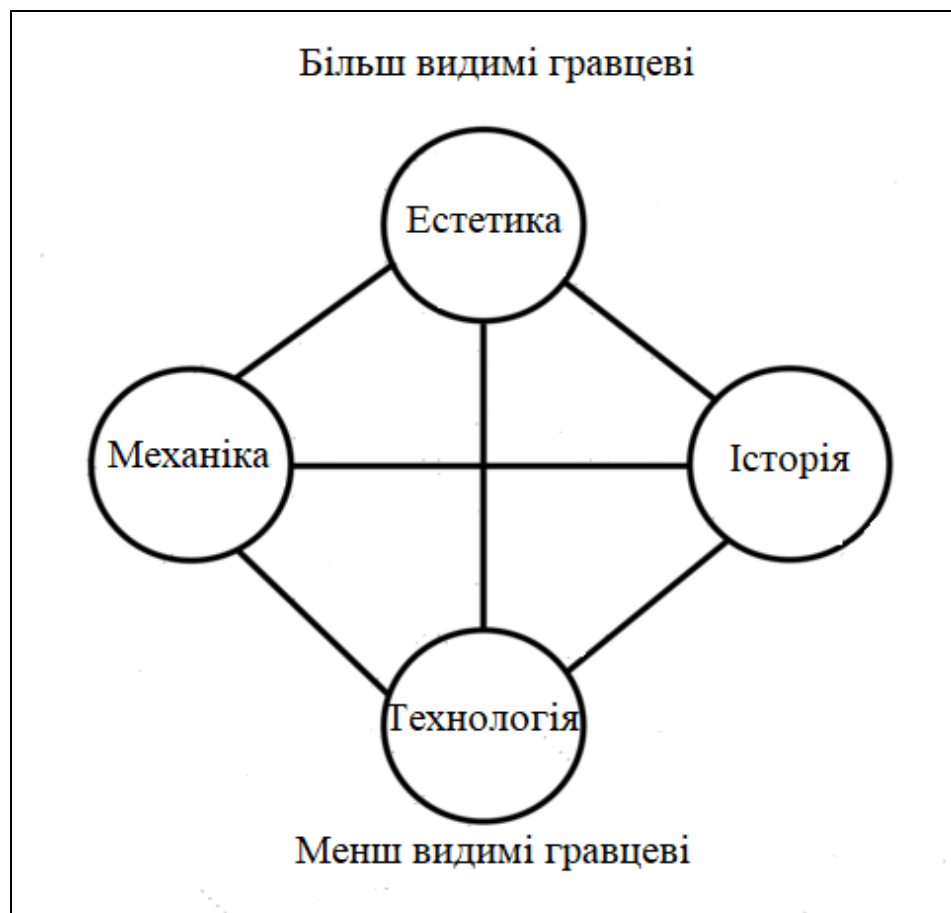


Рисунок 5.1 – Елементарна тетрада

Розглянемо усі чотири складові и то як вони взаємодіють між собою [14]:

1) **Механіка.** Це правила и процеси гри. Механіка описують цілі гри, методи, якими може або не може користуватися гравець в процесі досягнення цієї цілі, та т, що відбувається, коли він вибирає відповідний для прогресу метод. Якщо порівняти гри з більш лінійним досвідом (книги, фільми і т.д.), можна побачити, що лінійні типи розваг можуть включати в себе естетику, історію і технологію, але не механіку, і тому можна сказати, що механіка - це те, що робить гру грою. Коли ви визначаєтесь з основою вашої гри - набором механік, потрібно визначитися з технологією, яка забезпечує їх функціонування, і естетикою, яка акцентує на них увагу гравця. А також з історією, яка піднесе механіки в зрозумілій для гравця формі.

2) **Історія:** Це послідовність подій, які відбуваються у вашій грі. Оповідання може бути як лінійним, так і містить безліч побічних ліній і поворотів сюжету. Коли у вас є історія, яку ви хочете розповісти через гру, необхідно визначитися з механіками, які будуть посилювати історію і забезпечувати розвиток сюжету. Як будь-якому оповідачеві, вам потрібно буде вибрати естетику, яка закріпить ідеї вашої історії. Для того, щоб розповідь було органічно по відношенню до інших елементів вашої гри, потрібно визначитися з найбільш придатною технологією.

3) **Естетика:** Це те, як ваша гра виглядає, звучить, яка вона на запах і на смак, і як вона відчувається. Естетика - надзвичайно важливий аспект геймдизайн, тому що вона має саме пряме відношення до досвіду, який отримує гравець. Якщо у вас вже є візуальне і звукове супровід, в яке ви збираєтесь занурити гравця, потрібно буде вибрати технологію, яка дозволить як реалізувати задуману естетику, так і розвинути її на ваш розсуд. Потрібно буде вибрати і механіки, які дозволять гравцеві відчути себе в тому світі, який визначений вашої естетикою. Історія з послідовністю подій дозволять вашій естетиці проявлятися в потрібній мірі і в потрібному місці, виробляючи найбільший вплив на гравця.

4) **Технологія:** Це відноситься не стільки до сфери "високих технологій", скільки до матеріалів і взаємодій, які забезпечують функціонування вашої гри. Технологія, яку ви обираєте для своєї гри, дозволяє програмувати її на виконання бажаних дій і ігнорування всіх інших. Це свого роду посередник, через який зображується естетика, реалізується механіка і розповідається історія [14]

Кожен елемент необхідний. Яку б гру ви не створювали, потрібно враховувати всі чотири елементи. Всі елементи однаково важливі, і всі вони

однаково сильно впливають один на одного. Кожен з них в однаковій мірі впливає на досвід, який користувач отримує у вашій грі, а значить, кожен заслуговує однакової уваги.

Порівняємо усі чотири елемента з нашою грою та іграми аналогами.

Таблиця 5.3 Порівняння ігор

	Fisherman	CrazyFishing	FancyFishingVR	AmigoFishing
Механіка	Основною цілю гри є ловля риби на час та зароблені як найбільше очок за одну ігрову сесію	Цілі тут немає. Гравець пересувається по локаціям та ловить що хоче. А трофеї складає у домі.	У грі подали нестандартний вид на рибалку, суміщено рибалку та тир. Гравець знаходиться під водою в якому має інвентар в якому може вибрати зброю. Ігровий процес складається з попаданням зброєю у рибу яка плаває навколо гравця. У грі присутні різні режими	Гравець знаходиться глибоко під водою у субмарині. Треба ловити рибу за допомогою пушок які зтягують рибу до підводної човен. Фактично, ця гра поєднує у собі рибалку та shooter. За пійману рибу отримує гроші для покращення зброї. Після ловлі певної кількості риби гравець отримує рівень і відкриваються нові пушки у магазині. Гра є нескінченною та після виходу зберігається ігровий процес.
Історія	Відсутня	Відсутня	Відсутня	Відсутня

Естети ка	Упор на реалістичну графіку.	Проста та приємна мультяшна графіка.	Графіка у грі більш реалістична, ніж у першого аналогу, але досі примітивна	Графіка у грі більш реалістична, ніж у першого аналогу, але досі примітивна
Технол огія	Зроблена для HTC Vive VR Headset	Зроблена для. HTC Vive VR Headset	Зроблена для HTC Vive VR Headset та Vavle index	Зроблена для HTC Vive VR Headset, Vavle index та Oculus rift

Було проведено тестування шести людей на ігровому додатку Fisherman. Де трое почали проходити гру с першого рівня, а інші люди почали одразу з 4 рівня. Гравці проходять рівень тільки один раз. У таблиці 5.4 наведено середні значення балів які получили гравці на рівнях.

Таблиця 5.4 Середні значення отриманих результатів

Рівень	Кількість балів	Кількість балів
1	385	–
2	355	–
3	350	–
4	325	270

З отриманих результатів можна зробити висновок що гравці з першого по третій рівень здобувають необхідні навички для того щоб пройти четвертий рівень. Адже на четвертому рівні знаходяться усі ускладнюючі механіки. Також можна відмітити, що с кожною новою механікою гравці отримують все менш балів за проходження рівня, що пояснюється підвищенням його складності.

ВИСНОВКИ

У магістерській атестаційній роботі проведено дослідження методів та технологій розробки навчальних програм із застосуванням віртуальної реальності. Згідно мети дослідження розглянуті програмні засоби та середовища, технічне обладнання та їх особливості для віртуальної реальності, створено проект для реалізації навчального процесу, проведено дослідження впливу віртуальної реальності на навчання. Навчання у віртуальній реальності може проводитися, зокрема, у формі гри. Робота в навчальних VR-програмах покращує традиційні показники образної короткочасної пам'яті, спостережливості, стійкості, концентрації уваги, здатності до узагальнення і класифікації.

За результатами дослідження отримані наступні висновки:

1) віртуальна реальність – створений технічними засобами світ, який передається людині через його відчуття: зір, слух, дотик і інші. Віртуальна реальність імітує як вплив, так і реакції на вплив. Для створення переконливого комплексу відчуттів реальності комп'ютерний синтез властивостей і реакцій віртуальної реальності проводиться в реальному часі;

2) віртуальні технології мають безліч сфер застосування, серед них: архітектура, дизайн, автомобілебудування, геофізика, тренування рятувальних служб, пілотів, операторів важкої техніки; освіта, мистецтво, сфера продажів, мистецтво, туризм та багато інших;

3) навчальні програми, створені з VR, перш за все стимулюють мислення людини. VR сприяє прогресивному формуванню як процесуальних, так і операційних характеристик мислення, а також розвиває форми розумової активності;

4) гру формують такі складові, як «ігрова механіка», «історія», «естетика» та «технологія», які забезпечують зацікавленість у навчанні, саме навчання та заохочування;

5) складність навчальних програм та сам процес навчання забезпечується наявністю рівнів складності, кожному з яких відповідає складність ігрових механік, елементів (атрибутів) гри, штучного інтелекту. Результати досліджень апробовані при розробці гри-симулятора Fisherman, створеної на платформі

UnrealEngine 4, мовою C++ та із застосуванням blueprints. Проведено аналіз гри з метою виявлення недоліків та переваг за параметрами: економічними, технічним, естетичними, ергономічним, безпеки, швидкості досягнення результату, надійності. Проведено аналіз фактору можливостей додатку, аналіз результатів ігор- аналогів та розробленого додатку за чотирма складовими: естетика, механіка, історія та технологія.

Результати досліджень відображені у тезах доповіді Міжнародної студентської конференції «Радіоелектроніка та молодь у XXI ст.». Отже, завдання магістерської атестаційної роботи виконано в повному обсязі.

Щодо впровадженні розробленого навчального ігрового додатку, то наразі відбувається завантаження програмного додатку на торгівельний майданчик Viveport, на якому зібрано велику кількість ігор, розроблених під платформу HTC vive з використанням віртуальної реальності.

Під час завантаження програмного додатку не було труднощів, а оцінка та перевірка ігрового матеріалу триває. Після надання дозволу для розміщення гри, усі бажаючі зможуть використовувати гру Fisherman.

Навчальні віртуальні програми не можуть повністю замінити викладання в навчальних закладах, їх доцільно широко використовувати при вивченні найбільш складних тем різних предметів, а також для тренінгу професійних навичок в різних видах діяльності.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Вільна енциклопедія Вікіпедія [Електронний ресурс] / Мережева енциклопедія Wikipedia. 2000. - Forefront TMG 2010: Режим доступу: <http://ru.wikipedia.org/> . - Загл. с екрана.
- 2 Орлов О.К «Створення алгоритму поведінки риб на основі реальних умов рибалки» 24-й Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті». Зб. матеріалів форуму. Т. 6. – Харків: ХНУРЕ. 2020
3. Steven M. LaValle. Virtual Reality — England: Cambridge University Press, 2017. — 380 с
4. Віртуальна реальність в освітньому процесі [Електронний ресурс] – Режим доступу https://revolution.allbest.ru/pedagogics/01032601_0.html
5. VR-технології як метод і засіб навчання [Електронний ресурс] – Режим доступу https://revolution.allbest.ru/pedagogics/00906669_0.html
6. Юлія Трач, vr-технології як метод і засіб навчання [Електронний ресурс]: стаття, / Юлія Трач – Київ: Київського національного університету культури і мистецтв, 2017 – 14 с. – Режим доступу - <https://docplayer.net/72742933-Vr-tehnologiyi-yak-metod-i-zasib-navchannya.html>
7. Пономарьова С.В., Калита Н.І. Задача моделювання мотивації членів інтелектуальної команди // Міжнародна науково-технічна конференція молодих вчених, аспірантів та студентів «Комп'ютерні науки, інформаційні технології та системи управління» (CSYSC-2018): Зб. матеріалів конференції. – Івано-Франківськ: ПНУ ім. В. Стефаника, 2018. – С. 121-123.
8. Математична модель адаптації інтелектуальної команди / Н.І. Калита, С.В. Пономарьова // АСУ та прилади автоматики. – 2019. – №0 (00). – С.00-00.
9. Аналіз даних и процесів: учбових посібник для вузів / А. А. Барсегян, М. С. Куприянов, И. И. Холод [и др.]. -СПб.: БХВ-Петербург, 2009. -512 с.
- 10 Орлов О.К «Создавання алгоритму випадкової loot генерації основанийому на реальній рибалці» в 22-му Міжнародному молодіжному форумі «Радіоелектроніка і молодь у ХХІІ столітті»
11. Steven M. LaValle. Virtual Reality — England: Cambridge University Press, 2017. — 380 с

12. RealityTechnologies. TheUltimateGuidetoVirtualRealityTechnology [Електронний ресурс]. — Режим доступу: <http://www.realitytechnologies.com/virtual-reality>. — Дата публікації: 25.01.2017.
13. Мартин Фаулер. UML. Основи. Третье издание. Коротке керівництво за стандартною мовою об'єктного моделювання. Пер. з англ. — СПб: Символ Плюс, 2004. — 192 с
14. Jesse Schell The Art of Game Design: A Book of Lenses / Jesse Schell - Carnegie Mellon University, 2008. - 520 с N.Nosov, "Virtual Psychology" - М.: "Аграф", 2000 - 207s.
15. UnrealEngine.com :Сайтоффіційнійдокументаціїдо Unreal Engine 4. Режим доступу <https://docs.unrealengine.com/en-us/>
16. um.co.ua – Учбові Матеріали для студентів і школярів України, Етапи проектування ігри, Режим доступу - <http://um.co.ua/4/4-6/4-66272.html>
- 17 В.Є. Климнюк, Віртуальнареальність в освітньому процесі, [Електронний ресурс]наукова стаття / В.Є. Климнюк, 2018, Режим доступу https://www.researchgate.net/publication/325873912_Virtualna_realnist_v_osvitnomu_procesi
18. Мартин, Роберт К. Чистый код: создание, анализ и рефакторинг. Библиотека программиста [Текст]/ Роберт К. Мартин. — Изд-во: Вильямс, 2015. — 464 с.
19. Scott Rogers Level up! the guide to great video game design, 2010. - 483 с A John Wiley & Sons, Ltd., Publication.
20. Frank Popper. From Technological to Virtual Art, електронна книга [URL:https://mitpress.mit.edu/books/technological-virtual-art](https://mitpress.mit.edu/books/technological-virtual-art)
21. Pyster, A., D. Olwell, N. Hutchison, S. Enck, J. Anthony, D. Henry, and A. Squires (eds). Guide to the Systems Engineering Body of Knowledge (SEBoK) version 1.0. — The Trustees of the Stevens Institute of Technology, 2012.
22. Kossiakoff A., Sweet W. N., Seymour S. J., Biemer S. M. Systems Engineering Principles and Practice. — 2-е изд. — Hoboken, New Jersey: A John Wiley & Sons, 2011. — 599 с. — ISBN 978-0-470-40548-2
23. Сулейманов Д.Ш., Шакірова Д.М., Гільмуллін Р.А. Віртуальний музей бібліотека «Наукові школи РТ» як освітня інтернету серед // Міжнародний електронний журнал "Освітні технології та суспільство (Educational Technology &

Society). - 2013. - т.16. - №3. - С. 655883. - ISSN 1436-4522. URL: <http://ifets.ieee.org/russian/periodical/journal.html>

24. Подкосова Я.Г., Варламов О.О., Остроух А.В., Краснянський М.Н. Аналіз перспектив використання технологій віртуальної реальності в дистанційному навчанні // Питання сучасної науки і практики. - 2011. - №2 (33). - С. 104 - 111.

25. Thakral S. Virtual Reality and M-Learning /S. Thakral, P. Manhas, C. Kumar // International Journal of Electronic Engineering Research. - 2010. - Vol. 2. - No. 5. - P. 659–661.

26 Future Reality: Virtual, Augmented & Mixed Reality (VR, AR & MR) Primer (2016) // Bank of America Merrill Lynch. URL: https://www.bofaml.com/content/dam/boamlimages/documents/articles/ID16_1099/virtual_reality_primer_short.pdf.

27. Augmented Reality and Virtual Reality Market by Offering (Hardware & Software), Device Type (HMD, HUD, Handheld Device, Gesture Tracking), Application (Enterprise, Consumer, Commercial, Healthcare, Automotive), and Geography – Global Forecast to 2023 (2018) URL: <https://www.marketsandmarkets.com/MarketReports/augmented-reality-virtual-reality-market-1185.html>.

28 ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання / Нац. стандарт України. – Вид. офіц. – [Чинний від 2017-07-01]. – Київ: ДП «УкрНДНЦ», 2016. – 26 с.