

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ
КОМП'ЮТЕРНОЇ ГРИ З МЕРЕЖЕВОЮ ВЗАЄМОДІЄЮ
RESEARCH AND SOFTWARE IMPLEMENTATION OF
A COMPUTER GAME WITH NETWORK INTERACTION

Виконав(ла): студент(ка) 2 курсу, групи КНТ-113М
Спеціальності 121 Інженерія програмного
(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

СТРИЖАКОВ В.І.

(ПРИЗВИЩЕ та ініціали)

Керівник ОЛІЙНИК А.О.

(ПРИЗВИЩЕ та ініціали)

Рецензент ІЛЛЯШЕНКО М.Б.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
Кафедра програмних засобів
Ступінь вищої освіти магістр
Спеціальність 121 Інженерія програмного забезпечення
(код і найменування)
Освітня програма (спеціалізація) Інженерія програмного забезпечення
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
“ ” 2024 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

СТРИЖАКОВА Володимира Ілліча

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація комп'ютерної гри з мережевою взаємодією. Research and Software Implementation of a Computer Game with Network Interaction

керівник проєкту (роботи) д.т.н., професор, ОЛІЙНИК Андрій Олександрович,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 07 ” жовтня 2024 року № 398

2. Строк подання студентом проєкту (роботи) 02 грудня 2024 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	ОЛІЙНИК А.О., професор		
Нормоконтроль	КАМІНСЬКА Ж.К., асистент		

7. Дата видачі завдання “9” вересня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка структури програми.	3 тиждень	Розділ 3
5	Розробка програми.	4-7 тижні	Розділи 3,4
6	Тестування та експериментальне дослідження програмного забезпечення.	8 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	9-10 тижні	Додатки
8	Нормоконтроль та рецензування.	11 тиждень	
9	Захист роботи.	12 тиждень	

Студент(ка)

(підпис) Володимир СТРИЖАКОВ
(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Андрій ОЛІЙНИК
(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
88 с., 3 табл., 35 рис., 3 дод., 15 джерел.

**C++, QT, ІГРОВИЙ ПРОЦЕС, КОМП'ЮТЕРНА ГРА, МЕРЕЖЕВА
ВЗАЄМОДІЯ, МОВА ПРОГРАМУВАННЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ.**

Об'єкт дослідження – процес розробки програмного забезпечення для комп'ютерних ігор.

Предмет дослідження – програмні засоби для підтримки комп'ютерних ігор.

Мета роботи – дослідження та розробка програмного забезпечення для комп'ютерної гри з мережевою взаємодією.

Матеріали, методи та технічні засоби: мова програмування C++, фреймворк Qt.

Результати. Розроблено програмне забезпечення для комп'ютерної гри з мережевою взаємодією за допомогою мови програмування C++ та фреймворку Qt.

Практична цінність роботи полягає у розробці програмного забезпечення, що забезпечує мережеву взаємодію для підтримки ігрового процесу.

Висновки. Виконано проєктування програмного забезпечення для комп'ютерної гри з мережевою взаємодією. Розроблено програмне забезпечення для комп'ютерної гри з мережевою взаємодією. Здійснено тестування розробленого програмного забезпечення для комп'ютерної гри з мережевою взаємодією.

Галузь використання – комп'ютерні ігри.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 88 pages, 3 tables, 35 figures, 3 appendixes, 15 sources.

C++, QT, GAME PROCESS, COMPUTER GAME, NETWORK INTERACTION, PROGRAMMING LANGUAGE, SOFTWARE.

The object of research is the process of software development for computer games.

The subject of the research is software for supporting the computer games.

The purpose of this work is research and development of software for a computer game with network interaction.

Materials, methods and technical tools: C++ programming language, Qt development environment.

Results. The software for a computer game with network interaction using the C++ programming language and the Qt development environment have been developed.

The practical value of the work lies in the development of software that provides network interaction to support gameplay.

Conclusions. The software for a computer game with network interaction has been designed. The software for a computer game with network interaction has been developed. The developed software for a computer game with network interaction has been tested.

Scope of use – computer games.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	8
Вступ.....	9
1 Аналіз предметної області.....	11
1.1 Комп'ютерні ігри з мережевою взаємодією	11
1.2 Аналіз програмного забезпечення комп'ютерних ігор з мережевою взаємодією	22
1.3 Висновки за розділом 1	31
2 Матеріали і методи	33
2.1 Вибір мови програмування	33
2.2 Вибір фреймворку для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією	36
2.3 Розробка моделі взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією	39
2.4 Висновки за розділом 2	41
3 Опис програми	42
3.1 Структура програмного забезпечення комп'ютерної гри з мережевою взаємодією	42
3.2 Функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією.....	43
3.3 Особливості розробки програмного забезпечення комп'ютерної гри з мережевою взаємодією.....	47
3.4 Проєктування інтерфейсу програмного забезпечення комп'ютерної гри з мережевою взаємодією.....	48
3.5 Висновки за розділом 3	51
4 Експлуатація, тестування та експериментальне дослідження програми.....	53
4.1 Призначення й умови застосування програми	53
4.2 Характеристики програми комп'ютерної гри з мережевою взаємодією	54
4.3 Інструкція по експлуатації програми.....	54

4.3.1	Звернення до програми	54
4.3.2	Вхідні й вихідні дані	55
4.3.3	Повідомлення.....	55
4.4	Виконання програмного забезпечення комп'ютерної гри з мережевою взаємодією	55
4.5	Тестування програмного забезпечення комп'ютерної гри з мережевою взаємодією	59
4.6	Висновки за розділом 4	59
	Висновки	61
	Перелік джерел посилань	64
	Додаток А Технічне завдання	66
	Додаток Б Фрагмент тексту програми	70
	Додаток В Слайди презентації	80

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

FPS – First Person Shooters;

MVC – Model-View-Controller;

БД – база даних;

КГ – комп'ютерна гра;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер.

ВСТУП

З розвитком технологій і появою нових апаратних засобів (процесори, графічні карти, мережеве обладнання) комп'ютерні ігри постійно оновлюється, використовуючи нові можливості, що дозволяє забезпечити кращу продуктивність та графіку гри, й, відповідно, підвищити рівень задоволеності користувачів [1]-[6].

Останні роки збільшення рівня популярності кіберспорту створює попит на мережеві комп'ютерні ігри з високою якістю графіки, швидкою відповіддю і стабільним підключенням. Відповідно, розробники комп'ютерних ігор працюють над вдосконаленням своїх продуктів, зокрема, програмних засобів, що забезпечують мережеву взаємодію, щоб задовольнити ці вимоги [1]-[6].

У мережевих комп'ютерних іграх дуже важлива боротьба з шахрайством. Розробка ефективних систем захисту від шахрайства потребує постійних оновлень програмного забезпечення комп'ютерних ігор, щоб захистити чесних гравців і забезпечити справедливу гру. Мережеві комп'ютерні ігри часто містять велику кількість персональних даних гравців. Актуальне програмне забезпечення підтримки мережевої взаємодії повинно забезпечувати високий рівень захисту цих даних від кібератак та несанкціонованого доступу. Сучасні мережеві ігри пропонують розширені соціальні функції, такі як голосовий і текстовий чат, спільні ігрові кімнати, інтеграція з соціальними мережами тощо. Ці функції потребують постійного оновлення для забезпечення стабільної роботи та покращення користувацького досвіду [1]-[6].

Мережеві комп'ютерні ігри доступні на різних платформах, що вимагає від розробників адаптації мережевого програмного забезпечення для кожної з платформ, що потребує постійного оновлення та оптимізації [1]-[6].

Проте деяке програмне забезпечення комп'ютерних ігор має складну архітектуру, що може призводити до технічних проблем, зокрема, до

зависання гри, проблем з підключенням до серверів та інших технічних труднощів, що негативно впливають на враження гравців від комп'ютерної гри. Деякі мережеві комп'ютерні ігри часто потребують потужного апаратного забезпечення на стороні клієнта (гравця) для оптимальної роботи, що може бути недоступним для всіх гравців. Це обмежує аудиторію гравців та може створювати нерівні умови для гравців з різними технічними та фінансовими можливостями [1]-[6]. В деяких випадках програмне забезпечення серверів, що підтримують мережеві комп'ютерні ігри, може не справлятися з великим навантаженням під час пікових годин або після виходу великих оновлень. Це може призводити до падінь серверів, тривалих черг на підключення та інших незручностей [1]-[6].

Тому актуальною є розробка програмного забезпечення комп'ютерної гри з мережевою взаємодією. У дипломній кваліфікаційній роботі магістра розв'язується актуальне завдання дослідження та розробки програмного забезпечення для комп'ютерної гри з мережевою взаємодією.

Для досягнення поставленої мети у кваліфікаційній роботі магістра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для комп'ютерної гри з мережевою взаємодією;
- розробити модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією;
- здійснити проєктування програмного забезпечення комп'ютерної гри з мережевою взаємодією;
- створити програмне забезпечення комп'ютерної гри з мережевою взаємодією;
- виконати тестування розробленого програмного забезпечення комп'ютерної гри з мережевою взаємодією.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Комп'ютерні ігри з мережевою взаємодією

Комп'ютерна гра – це електронна гра, яка взаємодіє з користувачем за допомогою мережі, як правило, Інтернет, через комп'ютерні засоби, такі як персональний комп'ютер (ПК), консоль або мобільний пристрій. Гравці керують персонажами або об'єктами у віртуальному світі, виконують завдання, долають перешкоди або змагаються з іншими гравцями чи комп'ютерними супротивниками [1]-[6].

Для доступу до великої кількості ігор в Інтернеті достатньо мати хороше з'єднання. Відеоігри більше не обмежуються лише консолями та ігровими ПК, тепер у них грають і через комп'ютерні мережі. Для ентузіастів відеоігор Інтернет пропонує розширені можливості, будь то командна гра, гра в поодиночку або проти суперника з іншого кінця світу. Інші види ігор, такі як аркади, пасьянси та азартні ігри, також доступні широкій аудиторії. Деякі сайти дозволяють любителям спорту та азартних ігор робити ставки на різні спортивні події. Найпопулярніші комп'ютерні ігри бувають різних жанрів. Однак варто бути обережним, оскільки азартні ігри можуть призвести до залежності, ізоляції та боргів [1]-[6].

Відеоігри стали невід'ємною частиною сучасної культури, і існує багато різновидів, які задовольняють різні вподобання гравців. У цій статті обговорюються різні типи відеоігор та їхня цінність і привабливість.

Проаналізуємо популярні жанри відеоігор, їхні особливості і причини, через які гравці їх полюбують [1]-[6].

Платформенні ігри – це автономні відеоігри, де гравці керують персонажами, які повинні досягати цілей, переходячи з одного рівня на інший. Зазвичай платформенні ігри поєднують 2D-графіку з простими середовищами, хоча все більше з'являється 3D-ігор. Платформенні ігри можуть бути веселими та захопливими, але водночас і дуже складними для програмної реалізації та впровадження [1]-[6].

Ігри симулятори створені для відтворення реалістичних сценаріїв або пов'язаних гіпотез. Вони можуть охоплювати різні форми, такі як авіасимулятори або військові ігри в реальному часі. Ігрові симулятори відрізняються високим рівнем деталізації та можуть бути налаштовані через різноманітні ігрові параметри та механіки. Вони часто орієнтовані на досвідчених гравців, забезпечуючи їм години розваг і викликів [1]-[6].

Шутери від першої особи (First Person Shooters, FPS) є одним з найпопулярніших жанрів в індустрії відеоігор. У таких іграх гравець бачить ігровий світ очима головного героя, використовуючи різні види зброї для знищення ворогів і виконання ігрових завдань. FPS-ігри відомі своєю складністю та чудовою графікою. Вони популярні серед гравців у комп'ютерні ігри усіх вікових категорій та ідеально підходять для тих, хто любить розважальний жанр [1]-[6].

Рольові комп'ютерні ігри (Role-playing games, RPG) – це відеоігри, в яких основний акцент робиться на історіях та персонажах, що розвиваються протягом сюжету. Гравці можуть приймати рішення, що впливають на перебіг подій і взаємодії з іншими персонажами. Зазвичай, RPG розробляються для консолей і ПК та вимагають хорошого сюжету і продуманих персонажів. RPG популярні, тому що дозволяють гравцям зануритися в багатогранний і різноманітний світ, роблячи важливі вибори протягом гри [1]-[6].

Також існують масові багатокористувацькі онлайн ігри, що дозволяють гравцям взаємодіяти в спільному віртуальному світі. Гравці можуть взаємодіяти з іншими користувачами через Інтернет і брати участь у різноманітних заходах, таких як полювання, торгівля, дослідження та пригоди. Масові багатокористувацькі онлайн ігри пропонують насичений і динамічний ігровий процес, у який можна грати будь-де і будь-коли. Ці ігри відомі своїм відкритим світом і функціональністю, що робить їх захопливими та цікавими для гри [1]-[6].

Ігри у вигляді онлайн стратегії, в яких кілька гравців змагаються в масштабних битвах (Multiplayer Online Battle Arena, МОБА). Гравці керують

окремими героями, оснащеними спеціальними навичками, щоб захистити свою базу від ворогів. Вони повинні розробляти стратегії та співпрацювати з командою для перемоги над суперниками. МОБА популярні серед комп'ютерних гравців, які захоплюються військовими іграми та шукають захопливих викликів [1]-[6].

Ігри у вигляді стратегій в реальному часі (Real Time Strategy, RTS) – це жанр ігор, зосереджений на плануванні та управлінні ресурсами в режимі реального часу. Гравці повинні розвивати і розширювати свої бази, проводити стратегічні дослідження, створювати війська та битися з ворогами. RTS пропонують складний і захопливий ігровий сюжет, який може бути дуже цікавим. Хоча вони належать до стратегічних ігор, RTS також включають елементи симуляції. Гравці повинні приймати швидкі та довгострокові стратегічні рішення [1]-[6].

Інтерактивна відеогра або комп'ютерна гра являє собою електронну гру, що передбачає взаємодію з людиною через створення візуального зворотного зв'язку. Гравець використовує обладнання для керування грою та бачить результати своїх дій у віртуальному середовищі. Термін відео у таких іграх спочатку стосувався дисплейних панелей, але з часом став охоплювати будь-які види дисплеїв [1]-[6].

Інтернет технології для гри у відеоігри, включаючи планшети, можуть використовувати настільні комп'ютери, спеціальні пристрої, такі як домашні консолі чи інші автомати, або невеликі портативні пристрої, такі як ноутбуки. З часом популярність певних відеоігор знижувалася, як це було з іграми, що були популярні раніше. Сьогодні відеоігри не лише популярні, але й іноді вважаються мистецтвом [1]-[6].

У відеоіграх ігрова консоль використовується як пристрій введення, який залежить від платформи. Це може бути панель управління з однією кнопкою або комплексний контролер з дванадцятьма кнопками та одним або кількома джойстиками. Ігри для персональних комп'ютерів часто вимагають придбання спеціального ігрового пристрою, хоча сучасні комп'ютерні ігри

зазвичай дозволяють або вимагають використання клавіатури та миші одночасно [1]-[6].

Останнім часом з'явилися нові методи управління, такі як використання камери для зворотного зв'язку з гравцем, сенсорні екрани на мобільних пристроях та технології розпізнавання рухів для ігрових консолей. Разом із цими нововведеннями з'явилися нові концепції та транспортні засоби для цільового використання [1]-[6].

Існують різні способи гри у відеоігри, і багато з них зараз доступні на різних платформах. Ігрові консолі, персональні комп'ютери та мобільні пристрої є основними видами платформ для відеоігор. Портативні консолі були невеликими пристроями для окремих ігор. Згодом їх замінили багатофункціональні портативні консолі. Останнім часом мобільні пристрої, особливо смартфони, а також телевізори стали популярними пристроями для відеоігор, багато з яких адаптовані для використання сенсорних екранів і камер [1]-[6].

Проаналізуємо апаратне забезпечення для підтримки ігрового процесу у комп'ютерних іграх. Ігрові консолі – це спеціалізовані комп'ютери, призначені для відеоігор. На відміну від ПК, консоль використовує власне апаратне забезпечення, яке рідко піддається налаштуванню. Домашні консолі зазвичай підключаються до телевізора і продаються разом з основним пристроєм, хоча можна додати додаткові периферійні пристрої або спеціальні набори ігор. Портативні консолі, окрім автономного живлення, мають усі необхідні периферійні пристрої в одному портативному корпусі. Ігри розробляються з урахуванням можливостей консолі та використовуваних нею цифрових носіїв [1]-[6].

Аркади – це термінальні системи, призначені для використання у громадських місцях. Вони зазвичай складаються з міцного корпусу та інтегрованих периферійних пристроїв. Перші такі автомати були спеціалізованими машинами, які містили лише одну гру, що було їхньою основною функцією. Існують різні типи комп'ютерів, які можна

налаштовувати та оновлювати відповідно до різних вимог комп'ютерних ігор. Хоча вони не призначені спеціально для ігор, їх модульна конструкція дозволяє додавати спеціалізоване обладнання, наприклад відео карти або спеціальні контролери, що робить їх придатними для ігор. Це важливо, щоб ігри працювали належним чином, оскільки вони потребують певної потужності від апаратного забезпечення. Якщо ці вимоги не виконуються, гра може не запускатися або працювати з помилками. Завдяки постійному розвитку комп'ютерних технологій, комп'ютери стають все важливішими для ігрових пристроїв, особливо для графічно насичених ігор. ПК також дозволяють грати в ігри, які більше не підтримуються сучасними платформами. Крім того, ПК підтримують гру в локальній мережі, що дозволяє кільком гравцям змагатися в одній грі [1]-[6].

Ігрові консолі та персональні комп'ютери з часом перейшли до використання контролерів і комбінацій клавіатура/миша як основних периферійних пристроїв, оскільки вони є дуже ефективними для тривалого використання [1]-[6].

Деякі ігри також можуть використовувати спеціальні контролери, але не обмежуються трьома основними системами керування: клавіатурою, мишею та ігровим пристроєм. Наприклад, джойстики та кермові системи з відчуттям зворотного зв'язку використовуються для покращення симуляторів і гоночних ігор [1]-[6].

Крім того, для певних ігор потрібні специфічні контролери. Це характерно для аркадних ігрових автоматів, які використовували спеціалізовані пристрої протягом багатьох років. Наприклад, музичні ігри використовують музичні інструменти, а деякі інші комп'ютерні ігри використовують лазерні або оптичні пістолети [1]-[6], іноді використовуються жести гравця для покращення взаємодії з грою, додатково до традиційних методів керування [1]-[6].

Дизайн ігор також іноді вимагає об'єднання кількох технологій в одному пристрої. Відеоігри передають інформацію через візуальні зображення

та звуки. Зображення зазвичай відображаються на наявних пристроях, таких як телевізори або комп'ютери, можливо, з використанням потокового відео (наприклад, для перегляду на великому екрані). Звуки гри відтворюються через зовнішні динаміки або аудіо системи [1]-[6].

Розвиток технологій віртуальної реальності просунувся вперед з новими консолями, що стало проривом у галузі комп'ютерних ігор. Концепція віртуальної реальності у відеоіграх поглибилася із запуском спеціальних пристроїв [1]-[6].

На сьогодні більшість видавців комп'ютерних ігор випускають свої проекти на декількох платформах. Це обумовлено як технологічними (технологічна сумісність між різними системами), так і економічними причинами (зростання середніх витрат на виробництво). Кожен великий виробник ігрових консолей прагне пропонувати ексклюзивні ігри, щоб просувати свої пристрої. Вони роблять це через власні студії або шляхом укладання контрактів з іншими видавцями. Часто саме ексклюзивні ігри роблять платформу успішною та популярною [1]-[6].

До середини двохтисячних років розповсюдженням відеоігор здебільшого займалися звичайні магазини, що спеціалізуються на електроніці, магазини іграшок або великі роздрібні мережі. З масовим розвитком Інтернету та онлайн магазинів, деякі бренди почали пропонувати цифрові копії відеоігор для прямого завантаження, часто за нижчими цінами, ніж у традиційних роздрібних мережах [1]-[6].

За іншою технологією продажу комп'ютерних ігор кілька вебсайтів дозволяють користувачам завантажувати відеоігри та пропонують доступ до ігор за доступними цінами. Цифрове розповсюдження відеоігор повільно стає стандартом у галузі та демократизується, особливо завдяки загрозам піратства та перевагам для розробників [1]-[6].

Індустрія відеоігор стає однією з провідних у світі. Розробка відеоігор сьогодні є дуже дорогою і може тривати роками, що робить її ризикованою справою, особливо для нових студій. Крім того, студія отримує лише невелику

частку від вартості гри. Хоча деякі розробники продемонстрували, що можна створити гру самостійно, сучасна розробка ігор зазвичай передбачає співпрацю кількох фахівців [1]-[6]. До цього процесу залучені графічні дизайнери, музиканти, аніматори, програмісти та гейм-дизайнери. Компанії з розробки відеоігор використовують внутрішні алгоритми для отримання прибутку від гравців, подібно до відповідної бізнес-моделі, яка спрямована на використання персональних даних користувачів цифрових медіа. Цифрові гіганти збирають і отримують дохід з даних своїх користувачів, що також має схожість з індустрією відеоігор [1]-[6].

У сучасних комп'ютерних іграх часто використовується інформація про поведінку гравців, на основі якої алгоритм пропонує їм придбати додаткові предмети. Інші широко використовувані методи використання особистих даних гравцями алгоритмами включають просування гравців, які вже витратили кошти на гру. Деякі алгоритми протиставляють звичайних гравців з кращими гравцями для збільшення їхніх шансів на успіх і, відповідно, підвищення їх рівня задоволеності комп'ютерною грою. Це означає, що покупка додаткових предметів у комп'ютерній грі може асоціюватися з ідеєю перемоги, що створює позитивний цикл для видавця, де прибуток залежить від покупок товарів [1]-[6].

Інший алгоритм отримання додаткових прибутків від комп'ютерних ігор наближається до типового профілю реального онлайн ігрового сегмента, використовуючи минулі покупки клієнтів для оцінки споживчих звичок гравців. Окрім згаданого вище алгоритму, цей підхід дозволяє індивідуальним гравцям прогресувати в грі залежно від сум, які вони вкладають, спонукаючи тих, хто вже витрачає, до подальших покупок і стимулюючи інших покращувати свої можливості з часом [1]-[6].

Також спостерігається певне зближення між цифровими гігантами і відео ігровою індустрією, де кожен з розробників комп'ютерних ігор відзначається у цій сфері своїми унікальними підходами [1]-[6].

Важливим наслідком випуску ігрових консолей з відповідним програмним забезпеченням було збільшення доходів для видавців. Раніше студії зазвичай заробляли більшість своїх грошей, випускаючи ігри в певний період року. Система транзакцій дозволяє видавцям продовжувати розвивати гру та отримувати стабільний дохід протягом усього життя гри, продаючи віртуальні предмети. Це зменшує ризики і є значним кроком у напрямку фінансової стабільності для гральних компаній, які переглядають свої стратегії прибутку. Крім того, можливість вносити плату за оновлення ігор дозволяє видавцям випускати ігри, які ще перебувають у розробці, і продовжувати їх вдосконалення, отримуючи дохід з моменту початку продажів. Цей підхід відомий як ранній доступ, коли студія робить доступною певну версію гри і використовує зворотний зв'язок від гравців для поліпшення гри [1]-[6].

Зазвичай, коли ігри перетворюються на інструменти, гравці використовують їх для експериментів, що робить ігрове середовище платформою для розвитку. Гравці вносять свій внесок через зворотний зв'язок, що дозволяє видавцям покращувати та розвивати гру. Комп'ютерна гра включає в себе і внески гравців, які видавець використовує для підтримки гри. Комп'ютерна гра залишається інтелектуальною власністю видавця, що може здійснювати зміни без відома авторів, що може викликати юридичні питання. Велика кількість гравців працюють добровольцями, маючи за мету покращення улюбленої гри та сприяння гральній спільноті, що відображає культуру відкритого програмного забезпечення та раннього Інтернету, де співпраця та взаємодопомога були ключовими елементами [1]-[6].

Зростання попиту на хмарні ігри співвідноситься з екологічним впливом відеоігор. Віддалені центри обробки даних та онлайн сервіси потребують додаткового обладнання і енергії через хмару. Якщо половина гравців перейдуть на хмарні ігри, споживання енергії і викиди парникових газів суттєво зростуть [1]-[6]. Одним з ефективних способів обмеження ігор для одного гравця є збереження соціальних зв'язків, що може страждати від надмірного ігрового захоплення. Зменшення контакту з родиною, друзями та

колегами є критерієм для оцінки надмірної залежності. Тому рекомендується встановлювати гнучкі, але чіткі правила, що обмежують час, який гравці витрачають на комп'ютерну гру [1]-[6].

Хоча деякі ігри мають механізми блокування, які припиняють гру через певний час і можуть бути налаштовані, такий підхід не є ідеальним, оскільки раптова перерва може негативно вплинути на гравця, який не встиг досягти своєї мети в грі [1]-[6].

Комп'ютерні відеоігри мають безпосередній вплив на когнітивні та фізичні здібності гравців, зокрема покращують пам'ять, розвивають та обробляють візуальну інформацію. Швейцарське дослідження, яке досліджувало потенційні пошкодження очей, показало, що гравці мають менше напруження зору, ніж ті, хто не грає в ігри. Вони також демонструють кращий периферичний і центральний зір, оскільки можуть швидко фокусуватися. Таким чином, вибір відеоігор може поліпшити зорові здібності. Насправді відеоігри іноді використовуються в терапевтичних цілях або військовими для навчання солдатів [1]-[6].

Деякі комп'ютерні ігри суттєво впливають на когнітивні здібності гравців та дозволяють гравцям мати кращу увагу, менше втомлюватися та залишатися більш зосередженими. Крім того, гравці показують кращі результати візуальних тестів, можуть читати дрібні цифри та виявляти чутливість до змін [1]-[6].

Деякі комп'ютерні ігри, які вимагають фізичної активності, наприклад, з музичними інструментами або з камерами для захоплення руху, сприяють регулярній фізичній активності і часто рекомендуються особам з певними проблемами [1]-[6]. У 2018 році був випущений адаптивний контролер Xbox, спеціально розроблений для людей з обмеженою мобільністю. Цей контролер є периферійним пристроєм, який робить комп'ютерні ігри значно доступнішими [1]-[6].

Комп'ютерні ігри також можуть використовуватися для навчання солдатів з метою покращення часу реакції та швидкості набору. Також

використовуються комп'ютерні ігри на основі віртуальної реальності для лікування різних психічних розладів, таких як агорафобія та акрофобія. Ці комп'ютерні ігри використовують технології, спочатку розроблені для відеоігор [1]-[6]. Відеоігри також використовуються в деяких випадках для додаткового заробітку. Розвиваючі відеоігри, зазвичай частина програм раннього розвитку дітей або як навчальні засоби, останнім часом переросли в навчальні ігри, також відомі як серйозні ігри, які сприяють засвоєнню знань учнями та студентами [1]-[6].

Деякі дослідження показують, що відеоігри сприяють кращій орієнтації в просторі, швидкому переключенню між ідеями та концепціями, а також ефективнішій співпраці. Таким чином, мозок може змінювати свою структуру відповідно до вимог середовища. Відеоігри можуть бути використані для профілактичних або навіть терапевтичних цілей, уповільнюючи прогресування дегенеративних захворювань [1]-[6].

Дослідники використовують комп'ютерні ігри та гарнітури віртуальної реальності, щоб допомогти пацієнтам подолати свої страхи. Пацієнт може працювати з новими, контрольованими ситуаціями і поступово впоратися зі своїми фобіями. Сам пацієнт може вибрати, як він буде справлятися зі страхами, що робить процес менш загрозливим [1]-[6]. Найбільша перевага віртуальної реальності полягає в тому, що користувач стає активним учасником досвіду, а не просто спостерігачем. Важливо, що засоби віртуальної реальності налаштовуються під конкретного користувача, збираючи дані про його погляди та поведінку. Це питання збору персональних даних викликає суперечки щодо захисту конфіденційності в контексті нових технологій. Один із когнітивних ефектів засобів віртуальної реальності полягає в тому, що сприйняття альтернативної реальності може змінювати фізичні та соціальні здібності [1]-[6].

Існує багато видів комп'ютерних ігор, і їх успіх більше залежить від мети гри, ніж від її жанру. Таким чином, жанри відеоігор також мають категорії, і багато ігор можуть поєднувати різні жанри та стилі. Аналізуючи

ігрову механіку, можна зрозуміти, як розробники та видавці класифікують свої комп'ютерні ігри [1]-[6].

Пригодницькі комп'ютерні ігри ставлять гравців у центр подій, які зазвичай включають фізичні випробування. Є багато типів пригодницьких комп'ютерних ігор, які можуть бути гучними [1]-[6].

Ігри на основі платформ отримали свою назву через те, що персонаж гравця взаємодіє з платформами, щоб бігати, стрибати або падати. Цей тип включає безліч різноманітних ігор [1]-[6].

Ігри зі стрільбою дозволяють гравцям використовувати зброю для знищення ворогів або супротивників. Залежно від виду, ці ігри поділяються на кілька категорій. Ігри зі стрільбою від першої особи: у цих іграх гравець бачить дії з перспективи головного героя. Ігри зі стрільбою від третьої особи: у таких іграх гравець бачить персонажа з точки зору, яка зазвичай розташована трохи вище і позаду нього, що забезпечує ширший огляд. В деяких інших іграх зі стрільбою ігровий екран має горизонтальну або вертикальну прокрутку. Ці ігри відомі своєю динамікою і необхідністю швидкої реакції [1]-[6].

Бойові комп'ютерні ігри зосереджуються на поєдинках, переважно рукопашному бою. В цих іграх представлено різних персонажів, кожен з яких має свій унікальний стиль бою. Ігри відповідного жанру часто зосереджені на боях, де гравцям доводиться боротися з численними хвилями ворогів, а не з одним супротивником. Інші ігри акцентують увагу на майстерності та точності, дозволяючи гравцям досягати своїх цілей, обходячи противників і вирішуючи проблеми в грі [1]-[6].

Комп'ютерні ігри на виживання здобули велику популярність. Гравцям часто доводиться боротися з ворогами та збирати ресурси для виживання. Музичні ігри змушують гравців слідувати ритму музики, натискаючи відповідні кнопки на контролері або повторюючи рухи, щоб набрати очки [1]-[6].

Визначено, що комп'ютерна гра являє собою електронну гру, що передбачає взаємодію з людиною через створення візуального зворотного зв'язку. При цьому взаємодія з користувачем часто забезпечується за допомогою мережі, як правило, Інтернет, через комп'ютерні засоби, такі як персональний комп'ютер, консоль або мобільний пристрій; гравець використовує обладнання для керування грою та бачить результати своїх дій у віртуальному середовищі. Проаналізовано популярні жанри відеоігор та їхні особливості. Досліджено засоби, що підтримують ігровий процес у комп'ютерних іграх.

1.2 Аналіз програмного забезпечення комп'ютерних ігор з мережевою взаємодією

Проаналізуємо програмне забезпечення комп'ютерних ігор з мережевою взаємодією [7]-[11].

Програмне забезпечення вебсайту IsThereAnyDeal (рис. 1.1) являє собою ресурс для порівняння відеоігор [7]–[9].

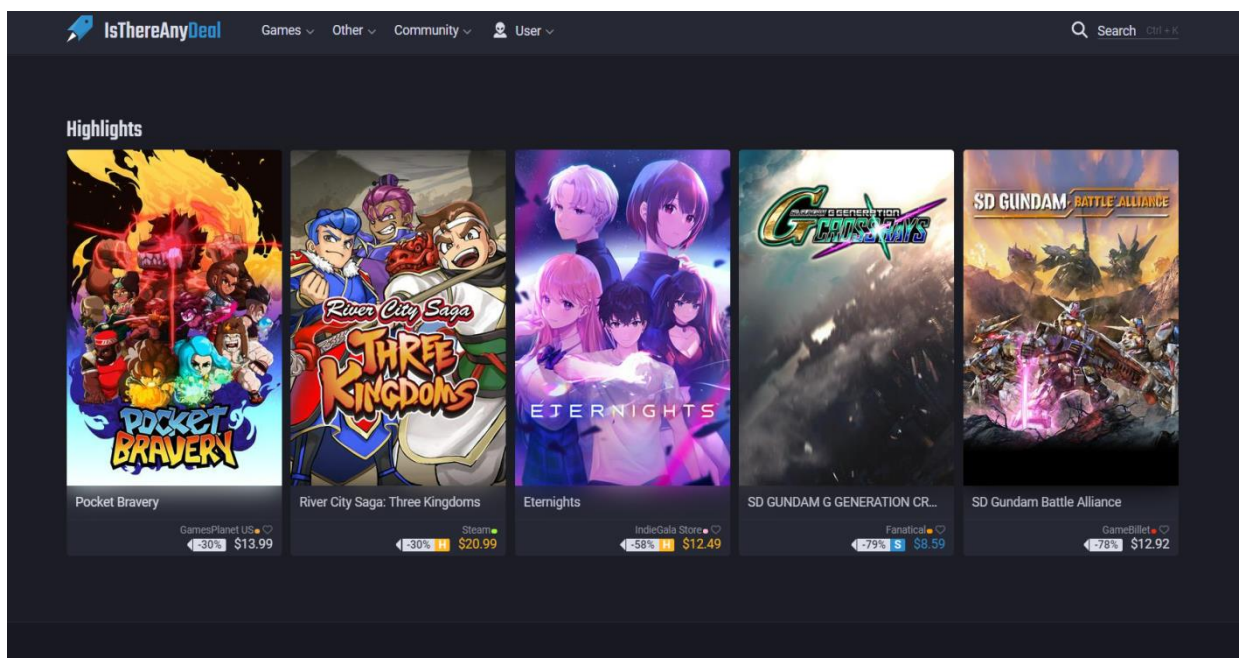


Рисунок 1.1 – Програмне забезпечення IsThereAnyDeal [9]

Програмне забезпечення вебсайту IsThereAnyDeal збирає практично всі наявні пропозиції щодо відеоігор і представляє їх поруч, сортує за розміром знижок. Програмне забезпечення вебсайту IsThereAnyDeal надає можливість користувачу знайти потрібну гру та переглянути всі доступні пропозиції на відповідній сторінці [7]–[9].

Інші корисні можливості програмного забезпечення вебсайту IsThereAnyDeal включають опцію встановлення бажаної ціни та отримання сповіщень, коли ціна комп'ютерної гри падає нижче цього рівня, а також перегляд історії цін, щоб визначити, чи поточна пропозиція комп'ютерної гри дійсно вигідна [7]–[9].

Якщо ціна комп'ютерної гри є основним критерієм користувача, то цей сайт стане для нього незамінним інструментом. Проте IsThereAnyDeal також надає список магазинів, де можна придбати гру навіть без знижок. Це дозволяє використовувати сайт IsThereAnyDeal як ресурс для пошуку місць придбання гри, незалежно від наявності акцій [7]–[9].

IsThereAnyDeal являє собою вебсайт, створений для гравців, які хочуть знайти найкращі пропозиції на відеоігри. Основними характеристиками програмного забезпечення IsThereAnyDeal є такі [7]–[9]:

- засоби цінових оповіщень: користувачі програмного забезпечення IsThereAnyDeal можуть налаштувати сповіщення про зниження цін на конкретні ігри, щоб отримувати повідомлення, коли ціна гри досягає бажаного рівня [7]–[9];

- відстеження історії цін на комп'ютерні ігри: вебсайт IsThereAnyDeal відстежує історію цін на ігри, що дозволяє користувачам бачити, чи є поточна пропозиція найкращою за певний період [7]–[9];

- можливість порівняння магазинів: програмне забезпечення вебсайту IsThereAnyDeal порівнює ціни з багатьох онлайн магазинів комп'ютерних ігор, включаючи Steam, GOG, Humble Bundle, Green Man Gaming та багато інших, щоб забезпечити користувачам найкращі можливі угоди [7]–[9];

– пакети та розпродажі: програмне забезпечення вебсайту IsThereAnyDeal відстежує спеціальні пропозиції, пакети та розпродажі, надаючи інформацію про тимчасові акції та знижки [7]–[9];

– інтеграція з іншими сервісами: програмне забезпечення вебсайту IsThereAnyDeal інтегрується з популярними сервісами, такими як Steam, що дозволяє користувачам імпортувати свої списки бажань комп'ютерних ігор і відстежувати знижки на ігри з їхніх списків [7]–[9];

– спільнота та огляди: користувачі програмного забезпечення вебсайту IsThereAnyDeal можуть залишати відгуки та рейтинги для різних пропозицій, допомагаючи іншим гравцям приймати обґрунтовані рішення про покупки комп'ютерних ігор [7]–[9];

– фільтри та налаштування: програмне забезпечення вебсайту IsThereAnyDeal надає велику кількість фільтрів для налаштування пошуку, включаючи платформи, регіони, типи ігор і багато іншого, що дозволяє знайти саме те, що потрібно [7]–[9];

– зручний інтерфейс: програмне забезпечення вебсайту IsThereAnyDeal має зручний і зрозумілий інтерфейс, який робить процес пошуку та порівняння цін простим і ефективним [7]–[9].

Таким чином, програмне забезпечення вебсайту IsThereAnyDeal корисним інструментом для гравців, які прагнуть заощадити гроші на покупці відеоігор, знаходячи найкращі доступні пропозиції [7]–[9].

Серед недоліків програмного забезпечення вебсайту IsThereAnyDeal можна відзначити обмежену кількість магазинів комп'ютерних ігор, він не включає всі можливі джерела, де можна придбати відеоігри. Деякі менші або регіональні магазини можуть бути відсутніми. Крім того, як правило, переважна кількість порівнюваних комп'ютерних ігор є платною [7]–[9].

Програмне забезпечення CheapShark (рис. 1.2) збирає ціни на відеоігри з кількох онлайн магазинів комп'ютерних ігор, дозволяючи швидко знайти будь-яку гру та порівняти її вартість у різних продавців [7], [10].



Рисунок 1.2 – Програмне забезпечення CheapShark [10]

Програмне забезпечення CheapShark показує найнижчу ціну гри за весь час, а також дату її встановлення. Крім того, ви можете встановити бажану ціну та отримувати сповіщення, коли гра подешевшає до цього рівня. Сторінка з переліком пропозицій на вебсайті CheapShark має цікавий показник сортування під назвою Рейтинг угод. Він враховує різні фактори (наприклад, абсолютну ціну, відсоток знижки, дату випуску), щоб визначити, наскільки вигідною є угода [7], [10].

Програмне забезпечення CheapShark є зручним інструментом для пошуку найкращих цін на відеоігри. Програмне забезпечення CheapShark має такі особливості [7], [10]:

- простий інтерфейс: програмне забезпечення вебсайту CheapShark має інтуїтивно зрозумілий інтерфейс, який полегшує навігацію та пошук потрібних комп'ютерних ігор [7], [10];

- робота з магазинами комп'ютерних ігор: програмне забезпечення вебсайту CheapShark агрегує ціни з таких популярних магазинів, як Steam, Green Man Gaming, Humble Bundle, Fanatical, та інших. Це дає змогу користувачам отримувати доступ до широкого спектру пропозицій комп'ютерних ігор [7], [10];

– інформаційні сповіщення: користувачі програмного забезпечення вебсайту CheapShark можуть підписатися на сповіщення про зниження цін, що дозволяє не пропустити вигідні угоди на улюблені комп'ютерні ігри [7], [10];

– фільтри та сортування: програмне забезпечення вебсайту CheapShark надає можливість фільтрувати ігри за різними критеріями, такими як відсоток знижки, ціновий діапазон та платформа. Це допомагає знайти найбільш вигідні пропозиції [7], [10];

– засоби обробки відгуків та рейтингів: програмне забезпечення вебсайту CheapShark надає інформацію про відгуки та рецензії користувачів, що допомагає прийняти зважене рішення перед покупкою [7], [10];

– зручність мобільної версії: програмне забезпечення вебсайту CheapShark має адаптовану версію для мобільних пристроїв, що дозволяє зручно використовувати сайт з будь-якого апаратного засобу [7], [10];

– засоби API для зв'язку зі сторонніми програмами: програмне забезпечення вебсайту CheapShark надає доступ до свого API, що дозволяє розробникам інтегрувати інформацію про знижки та ціни у власні додатки та сайти [7], [10].

Завдяки цим можливостям, програмне забезпечення вебсайту CheapShark є цінним ресурсом для гравців, які хочуть економити гроші, купуючи ігри за найвигіднішими цінами [7], [10].

Серед недоліків програмного забезпечення вебсайту CheapShark можна відзначити обмежену кількість магазинів комп'ютерних ігор, він не включає всі можливі джерела, де можна придбати відеоігри. Деякі менші або регіональні магазини можуть бути відсутніми. Крім того, як правило, переважна кількість порівнюваних комп'ютерних ігор є платною [7], [10].

Програмне забезпечення вебсайту Humble Bundle (рис. 1.3) пропонує колекції відеоігор зі знижкою, частина доходів від яких іде на благодійність. Спочатку ці пакети складалися переважно з простих комп'ютерних ігор, але з часом компанія почала включати в них також провідні ігри [7], [11].

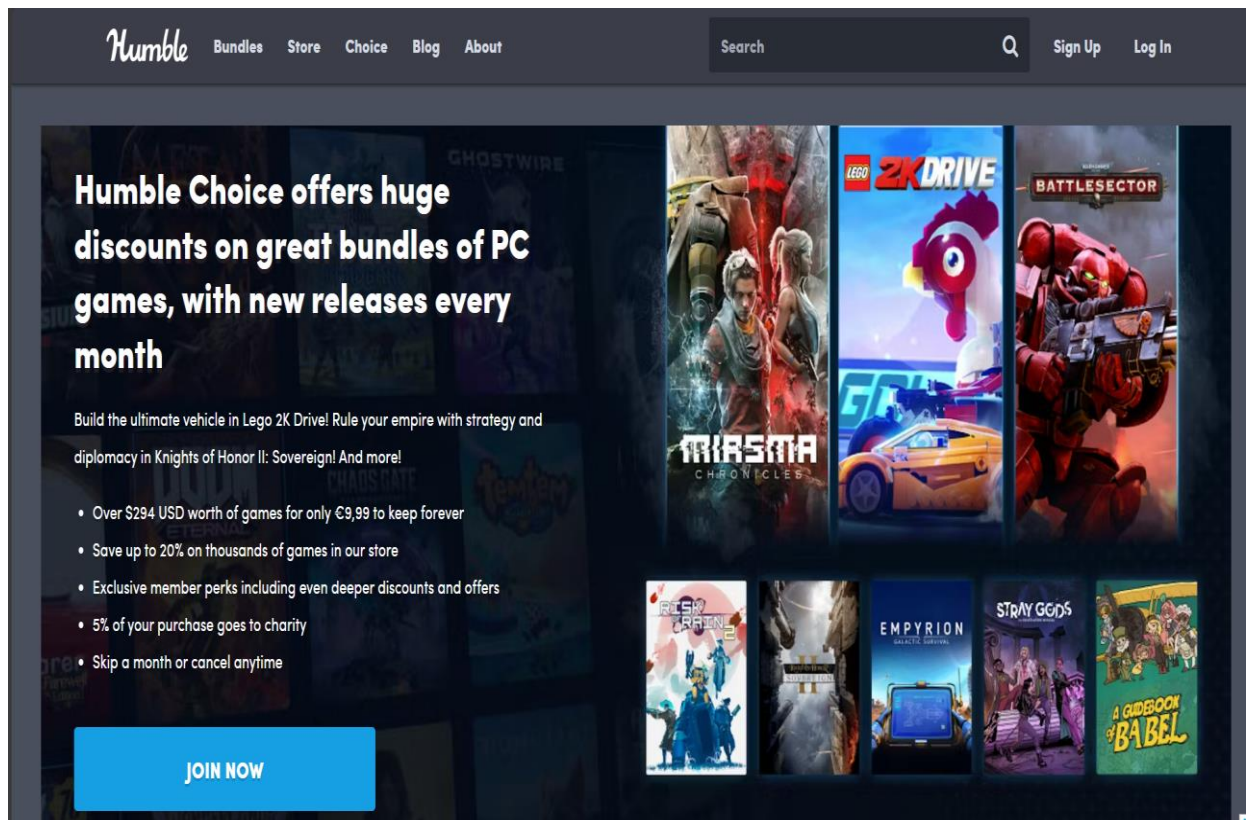


Рисунок 1.3 – Програмне забезпечення Humble Bundle [11]

Програмне забезпечення вебсайту Humble Bundle відомий своїм підходом "плати, скільки хочеш", що дозволяє отримати багато ігор за дуже вигідною ціною. Однак деякі з найбільш бажаних ігор доступні лише за умови сплати певної мінімальної суми [7], [11].

Програмне забезпечення вебсайту Humble Bundle також має підписку Humble Choice, яка надає доступ до вибраних ігор щомісяця. Підписними можуть вибирати певну кількість ігор із запропонованого списку, а також отримують доступ до бібліотеки Humble Trove, яка містить багато комп'ютерних ігор без DRM [7], [11].

Крім того, Humble Bundle надає платформу для розробників, допомагаючи їм привертати увагу до своїх проєктів і отримувати фінансову підтримку. Це робить вебсайт Humble Bundle не тільки місцем для вигідних покупок, але й важливим інструментом для розвитку індустрії комп'ютерних ігор [7], [11]. Значна частина доходів від продажів комп'ютерних ігор

продовжує йти на благодійність, що дозволяє користувачам одночасно підтримувати добрі справи під час покупок комп'ютерних ігор [7], [11].

Програмне забезпечення Humble Bundle має такі особливості [7], [11]:

- платформа для цифрових продуктів: програмне забезпечення вебсайту Humble Bundle пропонує широкий асортимент цифрових продуктів, включаючи комп'ютерні ігри, електронні книги, програмне забезпечення та комікси [7], [11];

- модель "плати, скільки хочеш": користувачі програмного забезпечення вебсайту Humble Bundle можуть визначати свою власну ціну за пакети продуктів. Однак для доступу до певних рівнів пакетів може знадобитися мінімальна сума [7], [11];

- благодійні внески: частина доходів від кожної покупки на вебсайті Humble Bundle йде на благодійність. Користувачі програмного забезпечення вебсайту Humble Bundle можуть вибирати, яка частка їхньої оплати йде на благодійні організації [7], [11];

- засіб Humble Choice: щомісячна підписка, яка надає доступ до вибраних ігор. Користувачі, що підписалися, можуть вибирати певну кількість ігор із запропонованого списку, а також отримують доступ до бібліотеки Humble Trove з багатьма іграми [7], [11];

- засіб Humble Store: Інтернет магазин, де користувачі можуть купувати окремі ігри та інші цифрові продукти зі знижками. Частина доходів від покупок у Humble Store також йде на благодійність [7], [11];

- індивідуальні акції та розпродажі: Humble Bundle регулярно проводить акції та розпродажі, пропонуючи значні знижки на популярні комп'ютерні ігри [7], [11];

- бібліотека Humble Trove: колекція комп'ютерних ігор, доступна для користувачів, що підписалися на Humble Choice, яку можна завантажити та грати без обмежень [7], [11];

– інструменти для розробників комп'ютерних ігор: програмне забезпечення вебсайту Humble Bundle підтримує розробників комп'ютерних ігор, надаючи платформу для просування та продажу їхніх проєктів [7], [11];

– гнучкість у розподілі платежів: користувачі програмного забезпечення вебсайту Humble Bundle можуть налаштувати, яку частку їхньої оплати отримують розробники, благодійні організації та сама платформа Humble Bundle [7], [11];

– висока якість контенту: програмне забезпечення вебсайту Humble Bundle пропонує комп'ютерні ігри, професійні програмні пакети та високоякісні електронні книги [7], [11].

Ці особливості роблять програмне забезпечення вебсайту Humble Bundle привабливою платформою для гравців у комп'ютерні ігри, читачів, розробників і всіх, хто зацікавлений у цифрових продуктах з додатковою користю для суспільства [7], [11].

Крім комп'ютерних ігор, вебсайт Humble Bundle продає електронні книги, комікси, програмне забезпечення для розробників і дизайнерів, а також підписки на різноманітні сервіси. Humble Bundle має активну спільноту користувачів, які обговорюють акції, діляться відгуками та допомагають один одному вирішувати питання. Платформа також надає відмінну підтримку клієнтам, щоб забезпечити задоволеність покупців. Багато ігор, які пропонуються на Humble Bundle, доступні для різних платформ, включаючи Windows, Mac, Linux та іноді навіть мобільні пристрої. Це розширює доступність ігор для більш широкої аудиторії. Humble Bundle має гарну репутацію серед користувачів за високу якість своїх продуктів та надійність у виконанні своїх зобов'язань. Покупці можуть бути впевнені, що вони отримають саме те, що купують [7], [11].

Серед недоліків програмного забезпечення вебсайту Humble Bundle можна відзначити обмежену кількість магазинів комп'ютерних ігор, він не включає всі можливі джерела, де можна придбати відеоігри. Деякі менші або

регіональні магазини можуть бути відсутніми. Крім того, як правило, переважна кількість порівнюваних комп'ютерних ігор є платною [7], [11].

Порівняльну характеристику програмного забезпечення комп'ютерних ігор з мережевою взаємодією наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння програмного забезпечення комп'ютерних ігор з мережевою взаємодією

Критерій порівняння	IsThereAny Deal [9]	CheapShark [10]	Humble Bundle [11]
Засоби цінових оповіщень	+	+	—
Фільтри та налаштування	+—	+	+—
Спільнота та огляди	+—	+	—
Відстеження історії цін на комп'ютерні ігри	+	+—	—
Зручність інтерфейсу	+—	+	+—
Можливість порівняння магазинів комп'ютерних ігор	+	+—	—
Інтеграція з іншими сервісами	+—	+	+—
Засоби API для зв'язку зі сторонніми програмами	—	+	—

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато програмних засобів для підтримки комп'ютерних ігор з мережевою взаємодією. Проте деякі програмні засоби для підтримки комп'ютерних ігор з мережевою взаємодією можуть мати складну архітектуру, що може призводити до технічних проблем, зокрема, до зависання гри, проблем з підключенням до серверів та інших технічних труднощів, що негативно впливають на враження гравців від комп'ютерної гри. Деякі мережеві комп'ютерні ігри часто потребують потужного апаратного

забезпечення на стороні клієнта (гравця) для оптимальної роботи, що може бути недоступним для всіх гравців. Це обмежує аудиторію гравців та може створювати нерівні умови для гравців з різними технічними та фінансовими можливостями [1]-[6]. В деяких випадках програмне забезпечення серверів, що підтримують мережеві комп'ютерні ігри, може не справлятися з великим навантаженням під час пікових годин або після виходу великих оновлень. Це може призводити до падінь серверів, тривалих черг на підключення та інших незручностей. Тому актуальною є розробка програмного забезпечення комп'ютерної гри з мережевою взаємодією.

При розробці програмного забезпечення комп'ютерної гри з мережевою взаємодією необхідно забезпечити такі функціональні вимоги:

- підтримка можливості реєстрації та авторизації користувачів на сервері, який є консольним застосунком;
- підтримка можливості забезпечення мережевої взаємодії гравців з сервером;
- можливість перегляду статистики гравців;
- можливість вибору комп'ютерної гри;
- підтримка можливості перевірки коректності введених гравцем даних;
- підтримка ігрового процесу комп'ютерної гри, обраної користувачем.

1.3 Висновки за розділом 1

Визначено, що комп'ютерна гра являє собою електронну гру, що передбачає взаємодію з людиною через створення візуального зворотного зв'язку. При цьому взаємодія з користувачем часто забезпечується за допомогою мережі, як правило, Інтернет, через комп'ютерні засоби, такі як персональний комп'ютер, консоль або мобільний пристрій; гравець використовує обладнання для керування грою та бачить результати своїх дій у віртуальному середовищі. Проаналізовано популярні жанри відеоігор та їхні

особливості. Досліджено засоби, що підтримують ігровий процес у комп'ютерних іграх.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки комп'ютерних ігор з мережевою взаємодією. Проте деякі програмні засоби для підтримки комп'ютерних ігор з мережевою взаємодією можуть мати складну архітектуру, що може призводити до технічних проблем, зокрема, до зависання гри, проблем з підключенням до серверів та інших технічних труднощів, що негативно впливають на враження гравців від комп'ютерної гри. Деякі мережеві комп'ютерні ігри часто потребують потужного апаратного забезпечення на стороні клієнта (гравця) для оптимальної роботи, що може бути недоступним для всіх гравців. Це обмежує аудиторію гравців та може створювати нерівні умови для гравців з різними технічними та фінансовими можливостями. В деяких випадках програмне забезпечення серверів, що підтримують мережеві комп'ютерні ігри, може не справлятися з великим навантаженням під час пікових годин або після виходу великих оновлень. Це може призводити до падінь серверів, тривалих черг на підключення та інших незручностей. Тому актуальною є розробка програмного забезпечення комп'ютерної гри з мережевою взаємодією.

Сформульовано функціональні вимоги до програмного забезпечення комп'ютерної гри з мережевою взаємодією.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування

При розробці програмного забезпечення комп'ютерної гри з мережевою взаємодією було використано мову програмування C++ [12], [13].

C++ – це мова програмування загального призначення, розроблена як розширення мови C для підтримки структурованого доступу. Вона може бути як мовою системного програмування, так і прикладного. C++ є мовою програмування загального призначення, створеною для системного та прикладного програмування. C++ — це об'єктно-орієнтована мова, яка підтримує багато парадигм програмування, включаючи широкий спектр методів, функцій та стилів програмування [12], [13].

Однією з основних особливостей C++ є здатність підтримувати системне програмування, що робить цю мову придатною для створення операційних систем, драйверів та іншого системного програмного забезпечення. Водночас C++ пропонує широкий спектр бібліотек і функцій, що робить його популярним вибором для розробки настільних додатків, відеоігор та інших складних програм [12], [13].

C++ має велику та активну спільноту розробників, а також багато ресурсів і інструментів для вивчення та використання мови. Деякі ключові особливості C++ включають [12], [13]:

- стандартна бібліотека шаблонів (STL): STL пропонує широкий набір інструментів і алгоритмів для роботи з даними, що робить написання коду зручним і зрозумілим для користувачів [12], [13];

- об'єктно-орієнтоване програмування: C++ підтримує об'єктно-орієнтоване програмування, що дозволяє розробникам створювати класи та об'єкти, а також визначати методи та властивості для цих об'єктів [12], [13];

- графіка: шаблони C++ дозволяють розробникам писати універсальний код, який може працювати з різними типами даних. Це спрощує

створення повторно використовуваного та ефективного коду, який легко тестувати [12], [13];

– обробка винятків: C++ забезпечує потужні можливості для обробки винятків, що полегшує написання коду, який може обробляти помилки та непередбачувані ситуації [12], [13].

Загалом, C++ — це потужна та універсальна мова програмування, яка широко використовується в різноманітних додатках і підходить як для високорівневого програмування, так і для розробки прикладних програм.

C++ — це мова програмування, що підтримує різні парадигми, такі як процедурне, об'єктно-орієнтоване та загальне програмування. Її ефективність і сумісність з мовою C, яка є однією з найпоширеніших мов програмування, роблять C++ популярною у додатках, де продуктивність має ключове значення. C++ — це багатопарадигмальна мова програмування, яка дозволяє використовувати процедурний, об'єктно-орієнтований та загальний підходи до програмування. Стандартна бібліотека C++ включає стандартну бібліотеку шаблонів, яка надає програмістам потужні інструменти, такі як колекції (контейнери) та ітератори. Завдяки високій ефективності та сумісності з мовою C, яка є однією з найпоширеніших мов програмування, C++ здобула широку популярність. Вона використовується у багатьох додатках, де важлива висока продуктивність та ефективність роботи. C++ є невід'ємною частиною розробки програмного забезпечення для критично важливих систем, де продуктивність та надійність є основними вимогами [12], [13].

Як і в C, функції бібліотеки C++ можна використовувати за допомогою директиви `#include`, за якою йде ім'я заголовного файлу. Деякі з цих файлів потребують явного зв'язування. Бібліотека STL є частиною стандартної бібліотеки C++, вона не згадується в стандарті окремо. Це викликає занепокоєння у деяких людей, оскільки функції вводу-виводу в STL відрізняються від тих, що надаються стандартною бібліотекою C [12], [13].

Інкапсуляція дозволяє приховати внутрішню реалізацію класу, зосереджуючись лише на його функціях. У C++ класи можна оголошувати з

використанням зарезервованих ключових слів `public` (члени класу доступні для будь-якого зовнішнього коду), `private` (члени класу доступні лише для інших членів того ж класу або для друзів класу, визначених за допомогою ключового слова `friend`) або `protected` (члени класу мають такі ж обмеження, як і приватні, але вони також доступні у похідних класах) [12], [13].

Члени класу в C++ не обов'язково закривати. Однак оголошення всіх членів класу як `public` може призвести до втрати переваг структурованого програмування. Рекомендується вважати всі дані приватними або принаймні захищеними, а публікувати лише ті методи, які працюють з цими даними. Це допомагає приховати інформацію про реалізацію класу [12], [13].

Обґрунтування вибору мови програмування для розробки програмного забезпечення комп'ютерної гри з мережевою взаємодією наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення комп'ютерної гри з мережевою взаємодією

Критерій порівняння мов програмування	Мова програмування		
	C++	Python	Ruby
Зручність засобів для роботи з мережею	+	+—	+
Багатозадачність	+	+	+—
Ефективність організації системної роботи	+	—	+—
Зручність для створення ПЗ комп'ютерної гри з мережевою взаємодією	+	—	—
Засоби оптимізації мережевої взаємодії	+	+—	+—

Отже, для реалізації програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано мову програмування C++, яка є мовою високого рівня, надає засоби оптимізації мережевої взаємодії, забезпечує низькорівневий доступ до апаратних ресурсів, що особливо корисно для оптимізації графіки, звуку та мережевої взаємодії в іграх.

2.2 Вибір фреймворку для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією

В якості фреймворку для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією було обрано Qt [14], [15].

Qt – це потужний фреймворк для розробки графічних додатків, що підтримує різні програмні і апаратні платформи, такі як Linux, Windows, macOS, Android та вбудовані системи. Він відомий своєю лідерською позицією та високою швидкістю [14], [15].

Qt пропонує два підходи для дизайну інтерфейсу користувача: Qt Widgets і Qt Quick. Qt Widgets інтегруються добре зі зовнішнім виглядом робочого столу, тоді як Qt Quick забезпечує потужність з плавною анімацією і оптимізацією для сенсорних інтерфейсів. Qt є повноцінним фреймворком розробки, який надає інструменти для зручного створення додатків для настільних комп'ютерів, вбудованих систем і мобільних пристроїв [14], [15].

Qt використовується для створення графічних інтерфейсів користувача та програм, які працюють на різних настільних і мобільних платформах. Більшість GUI програм, створених за допомогою Qt, мають крос-платформений інтерфейс, що робить Qt популярним у відомій сцені відеопрограмування. Крім того, можна розробляти програми без графічного інтерфейсу, такі як командні рядки та автономні програми. Qt є потужним середовищем розробки, яке підтримує різні мови програмування через зовнішні розширення, такі як Python за допомогою PySide і PHP за допомогою розширення PHP5. Він також має широку міжнародну підтримку і пропонує

Qt Quick, який використовує мову сценаріїв QML з JavaScript для швидкої розробки, особливо зручної для мобільних пристроїв, з можливістю використання рідного коду для оптимізації продуктивності [14], [15].

Платформа та інструменти Qt доступні як з відкритим кодом, так і за комерційними ліцензіями. Комерційна версія Qt включає додаткові програми та інструменти. Qt спеціально створено для розробки і розгортання додатків на різних настільних і мобільних операційних системах [14], [15].

Інші можливості включають підтримку SQL баз даних, обробку XML та JSON, керування потоками і мережеві функції. Проект Qt є повноцінним інструментарієм для розробки фреймворків і програм на основі Qt [14], [15].

Qt Creator надає засоби для управління всіма етапами життєвого циклу розробки програм, від їх створення до розгортання на цільовій платформі. Цей інструмент автоматизує багато завдань, таких як створення проектів через майстри, які крок за кроком ведуть розробника через процес створення проекту, створення необхідних файлів та налаштування параметрів згідно з вибором користувача. Крім того, Qt Creator прискорює написання коду за допомогою семантичної підказки, аналізу синтаксису, автодоповнення, функцій перетворення та інших корисних функцій [14], [15].

Qt Designer є інструментом для створення графічних інтерфейсів користувача (GUI) для програм, розроблених у Qt Creator. Програміст за допомогою цього засобу може візуально створювати та налаштовувати віджети та діалогові вікна, а потім тестувати їх за допомогою різних сценаріїв і умовних підказок. Qt Designer також доступний як окремий інструмент. Обидва варіанти Qt Designer мають схожі можливості, але відрізняються застосуванням підтримуваних файлів та перемиканням між різними режимами розгортання і платформами [14], [15].

Обґрунтування вибору фреймворку для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору фреймворку для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією

Критерій порівняння фреймворків	Фреймворки		
	Qt	GTK	Flutter
Мова програмування	C++ і QML	C і інші мови через зв'язки	Dart і JavaScript
Швидкодія	Висока швидкодія завдяки оптимізаціям і компіляції	Висока швидкодія, але менша ніж у Qt	Висока швидкодія, оптимізована для мобільних платформ
Інтеграція з мережевими технологіями	Повна підтримка мережних бібліотек та протоколів	Підтримка мережних бібліотек	Нативна підтримка HTTP та мережних запитів
Засоби розробки	Qt Creator і Qt Designer для розробки і UI	Glade для розробки і GTK Inspector для відлагодження	Власний інструмент розробки та відлагодження
Відкритість та спільнота	Ліцензія LGPL або комерційна ліцензія	Ліцензія LGPL або GPL, активна спільнота	Ліцензія BSD, активна спільнота

Таким чином, для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано фреймворк Qt, оскільки він підтримує

широкий спектр платформ, має потужні засоби для розробки інтерфейсу користувача і мережевої інтеграції, високу швидкодію та добре розвинені інструменти розробки.

2.3 Розробка моделі взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією

З метою можливості розробки програмного забезпечення комп'ютерної гри з мережевою взаємодією створимо відповідну модель у вигляді діаграми прецедентів.

При розробленні моделі взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією враховуємо те, що користувачі програми мають можливості реєстрації та авторизації користувачів на сервері, мережевої взаємодії гравців з сервером, перегляду статистики гравців, вибору комп'ютерної гри, перевірки коректності введених гравцем даних, підтримки ігрового процесу комп'ютерної гри, обраної користувачем.

Основний успішний сценарій користувача при роботі з програмним забезпеченням комп'ютерної гри з мережевою взаємодією можна описати таким чином:

- користувач програмного забезпечення комп'ютерної гри з мережевою взаємодією авторизується;
- після авторизації користувача сервер програмного забезпечення комп'ютерної гри з мережевою взаємодією у відповідь повертає основну сторінку вебсайту з комп'ютерними іграми;
- користувач бачить відповідну сторінку вебсайту з комп'ютерними іграми та шукає потрібну гру;
- користувач за необхідності читає опис відповідної комп'ютерної гри;
- користувач натискає відповідну кнопку для запуску комп'ютерної гри;

- користувач грає в обрану комп'ютерну гру;
- після завершення гри користувачу відображається статистика;
- користувач завершує роботу з програмним забезпеченням комп'ютерної гри з мережевою взаємодією.

Враховуючи сценарії роботи користувачів при роботі з комп'ютерною грою з мережевою взаємодією, модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією подамо у вигляді діаграми прецедентів, зображеної на рис. 2.1.

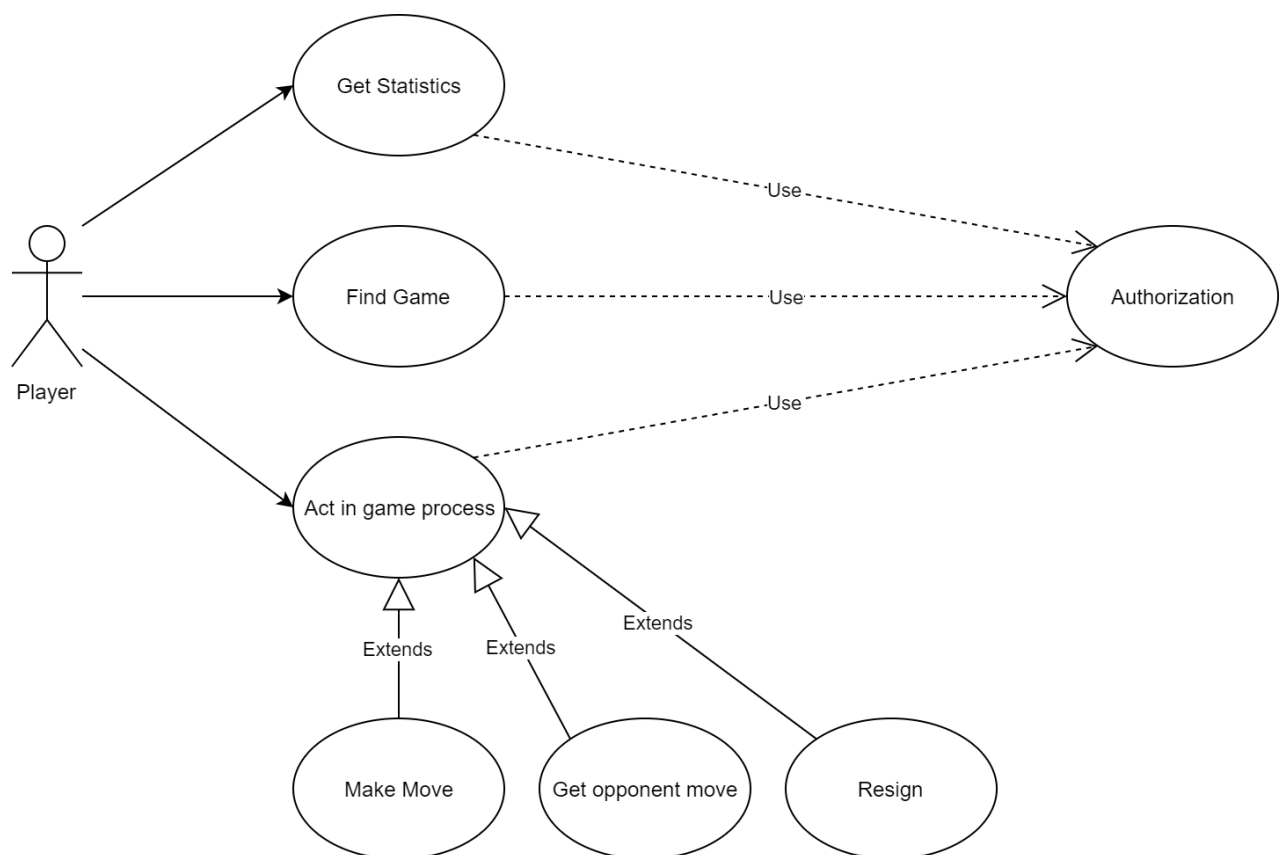


Рисунок 2.1 – Модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією

Таким чином, створено модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення комп'ютерної гри з мережевою взаємодією.

2.4 Висновки за розділом 2

Для реалізації програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано мову програмування C++, яка є мовою високого рівня, надає засоби оптимізації мережевої взаємодії, забезпечує низькорівневий доступ до апаратних ресурсів, що особливо корисно для оптимізації графіки, звуку та мережевої взаємодії в іграх.

Для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано фреймворк Qt, оскільки він підтримує широкий спектр платформ, має потужні засоби для розробки інтерфейсу користувача і мережевої інтеграції, високу швидкодію та добре розвинені інструменти розробки.

Створено модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення комп'ютерної гри з мережевою взаємодією.

3 ОПИС ПРОГРАМИ

3.1 Структура програмного забезпечення комп'ютерної гри з мережевою взаємодією

Структуру програмного забезпечення комп'ютерної гри з мережевою взаємодією наведено на рис. 3.1.

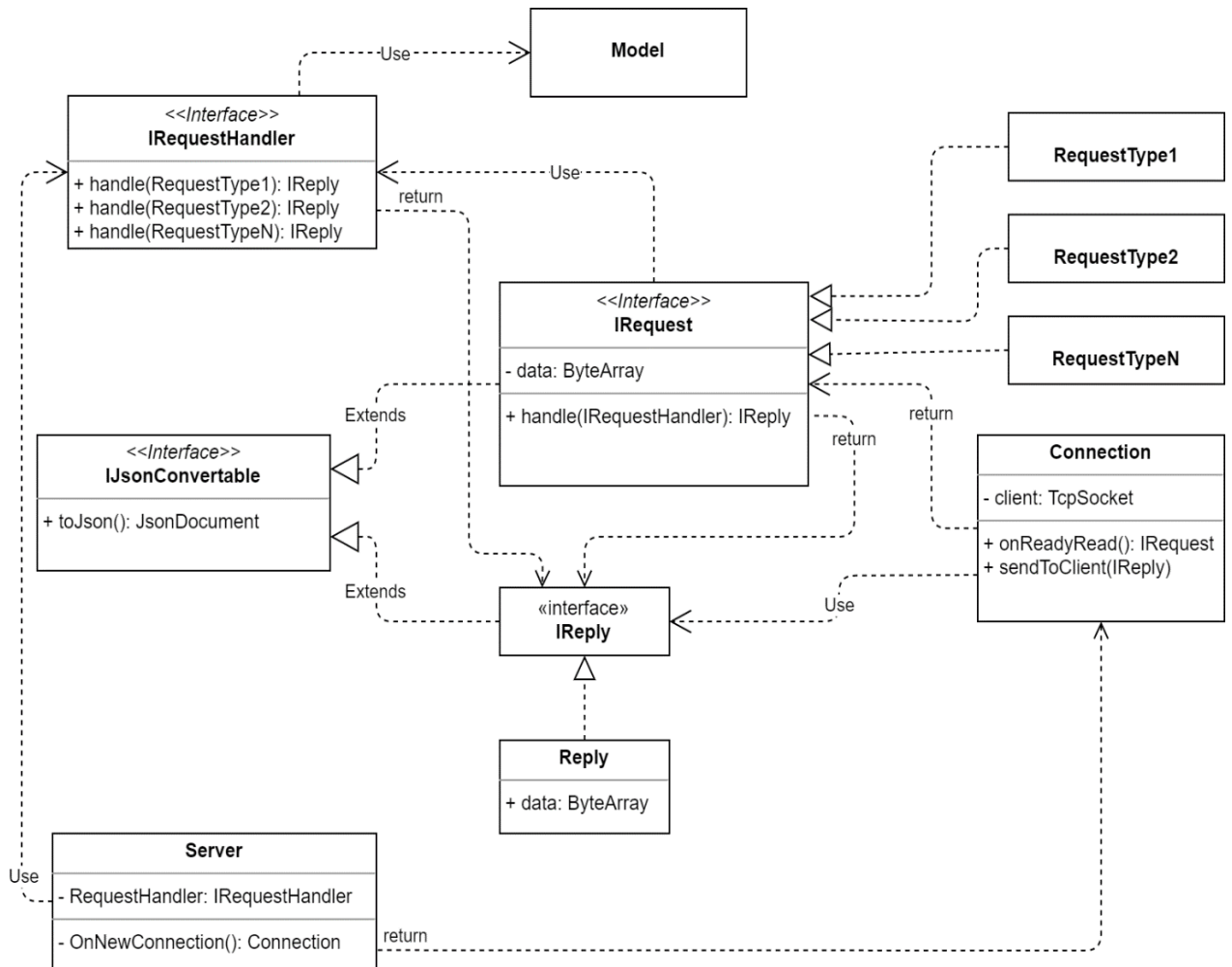


Рисунок 3.1 – Структура програмного забезпечення комп'ютерної гри з мережевою взаємодією

Як видно з рис. 3.1, основними класами програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано класи, що реалізують роботу серверу, ігрових моделей, зв'язку, а також запитів клієнтів та відповідей серверу.

Абстрактну модель програмного забезпечення комп'ютерної гри з мережевою взаємодією на основі діаграми компонентів наведено на рис. 3.2.

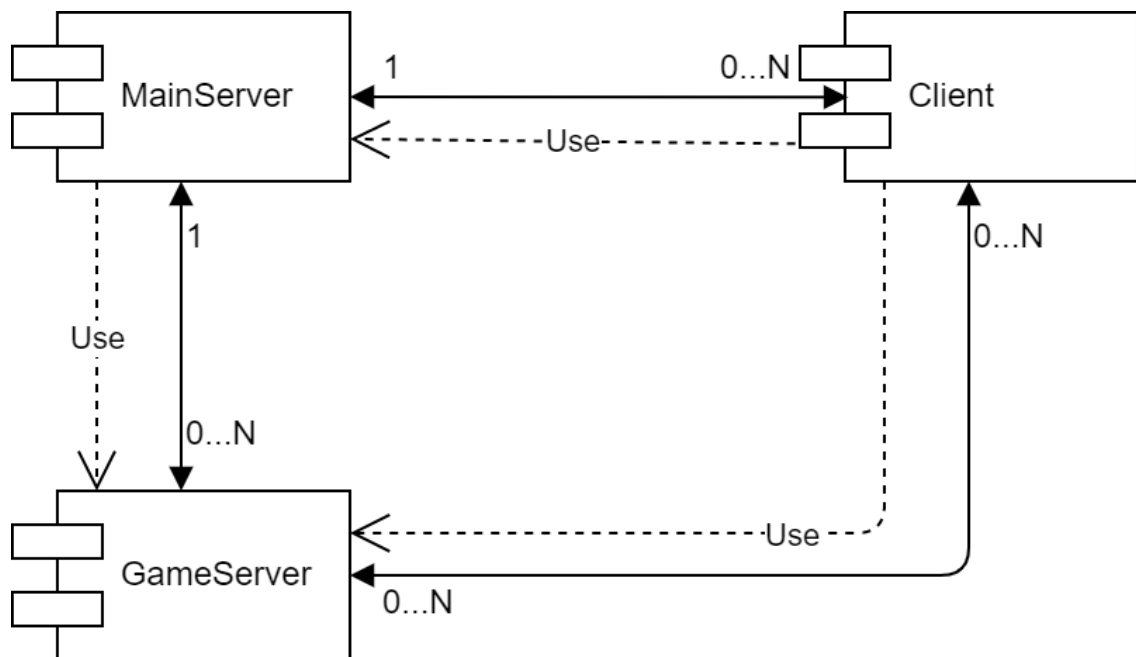


Рисунок 3.2 – Діаграма компонентів програмного забезпечення комп'ютерної гри з мережевою взаємодією

Як видно з рис. 3.2, розроблене програмне забезпечення комп'ютерної гри з мережевою взаємодією передбачає роботу з одним головним сервером, декількома ігровими серверами та декількома клієнтами (користувачами, гравцями). При цьому кожний клієнт програмного забезпечення комп'ютерної гри з мережевою взаємодією встановлює з'єднання з головним сервером та ігровим сервером.

3.2 Функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією

Функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією подамо за допомогою схеми у вигляді діаграми діяльності, зображеної на рис. 3.3.

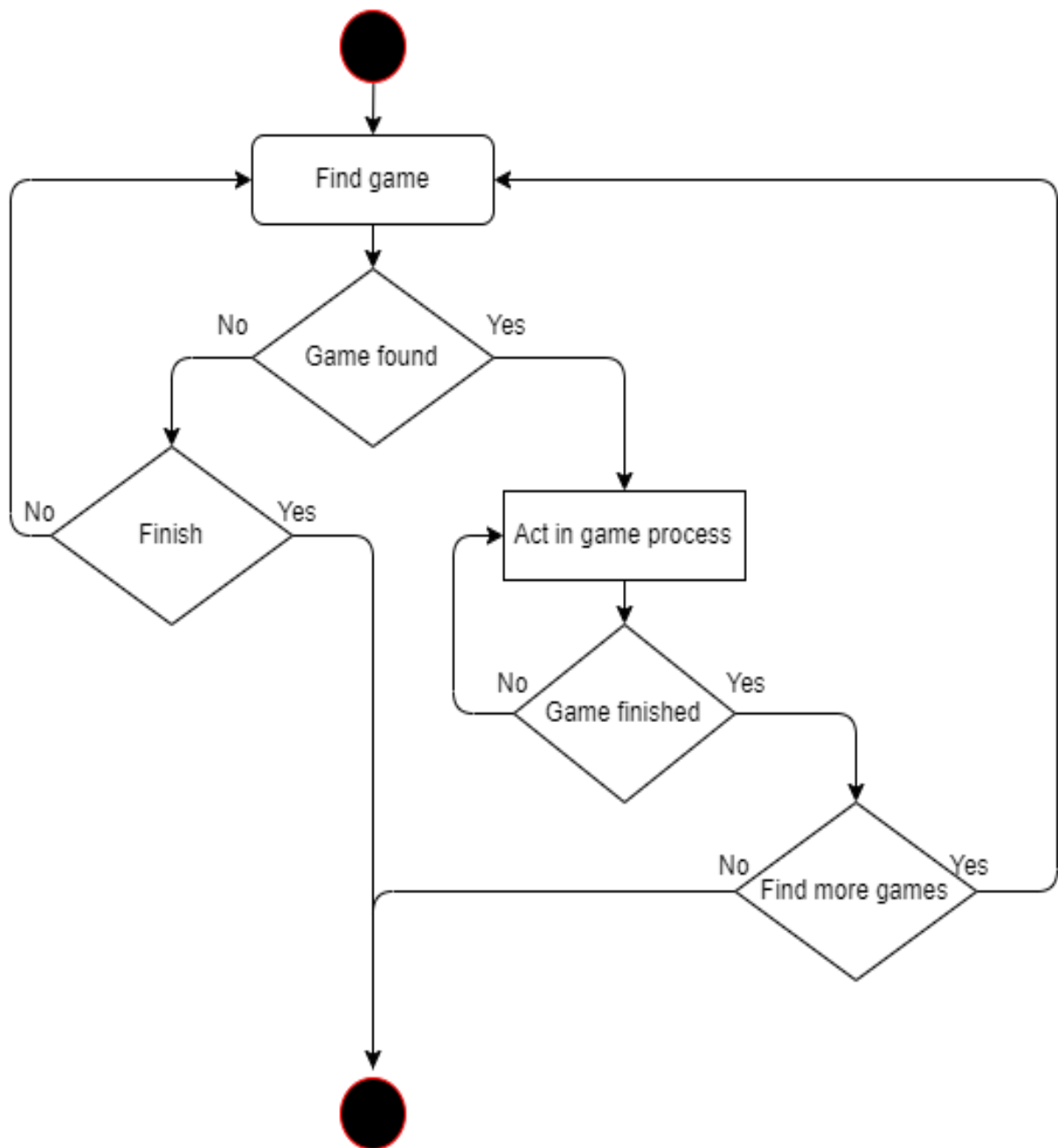


Рисунок 3.3 – Схема функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією у вигляді діаграми діяльності

Схема функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією у вигляді діаграми діяльності показує процес діяльності клієнта-гравця, ціль якого – грати у комп'ютерну гру.

Діаграма станів програмного забезпечення комп'ютерної гри з мережевою взаємодією наведена на рис. 3.4.

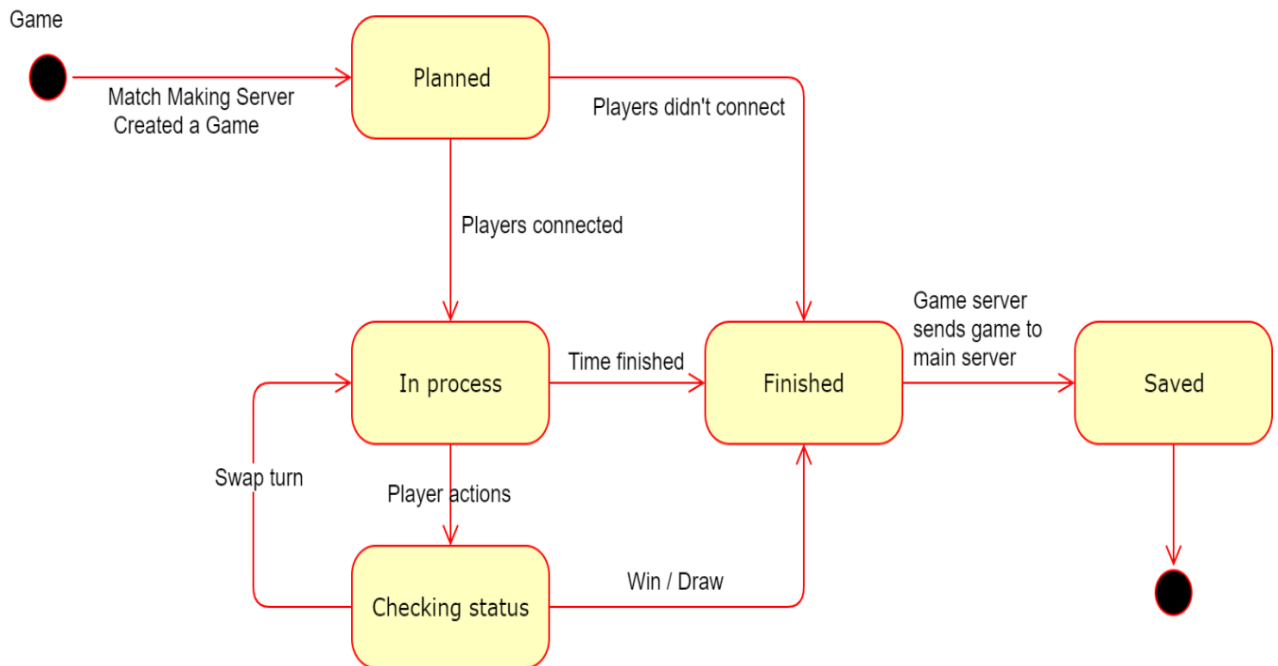


Рисунок 3.4 – Діаграма станів програмного забезпечення комп'ютерної гри з мережевою взаємодією

Дана діаграма програмного забезпечення комп'ютерної гри з мережевою взаємодією показує усі можливі стани комп'ютерної гри. Комп'ютерна гра може бути запланованою (комп'ютерна гра створена, але не всі користувачі підключилися), в процесі, в процесі перевірки статусу, закінченою та збереженою.

Діаграма кооперацій програмного забезпечення комп'ютерної гри з мережевою взаємодією наведена на рис. 3.5. Діаграма кооперацій програмного забезпечення комп'ютерної гри з мережевою взаємодією демонструє кооперацію компонентів ПЗ та їх складових при різних запитах та діях гравця у комп'ютерну гру.

Діаграма послідовності основного процесу програмного забезпечення комп'ютерної гри з мережевою взаємодією наведена на рис. 3.6. Діаграма послідовностей програмного забезпечення комп'ютерної гри з мережевою взаємодією демонструє лише послідовні, а не асинхронні операції чи дії, або узагальнює їх до такого рівня.

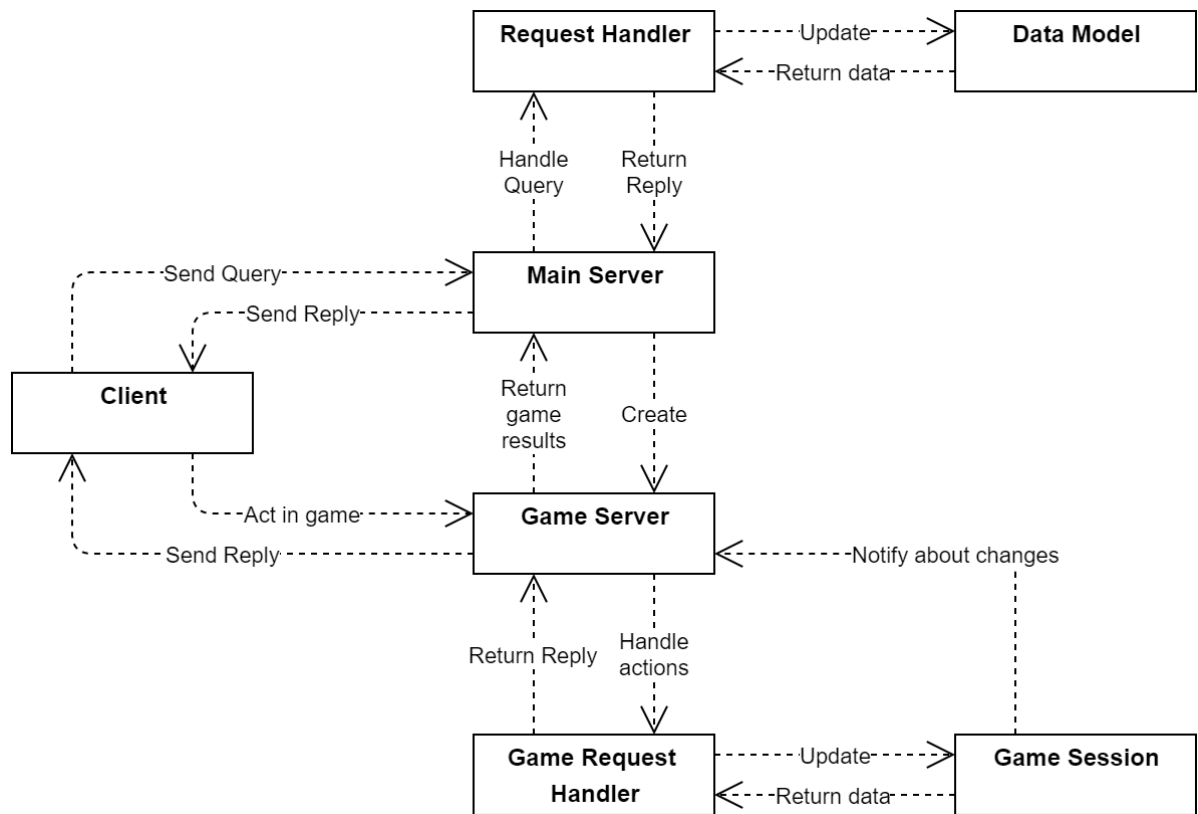


Рисунок 3.5 – Діаграма кооперацій програмного забезпечення комп'ютерної гри з мережевою взаємодією

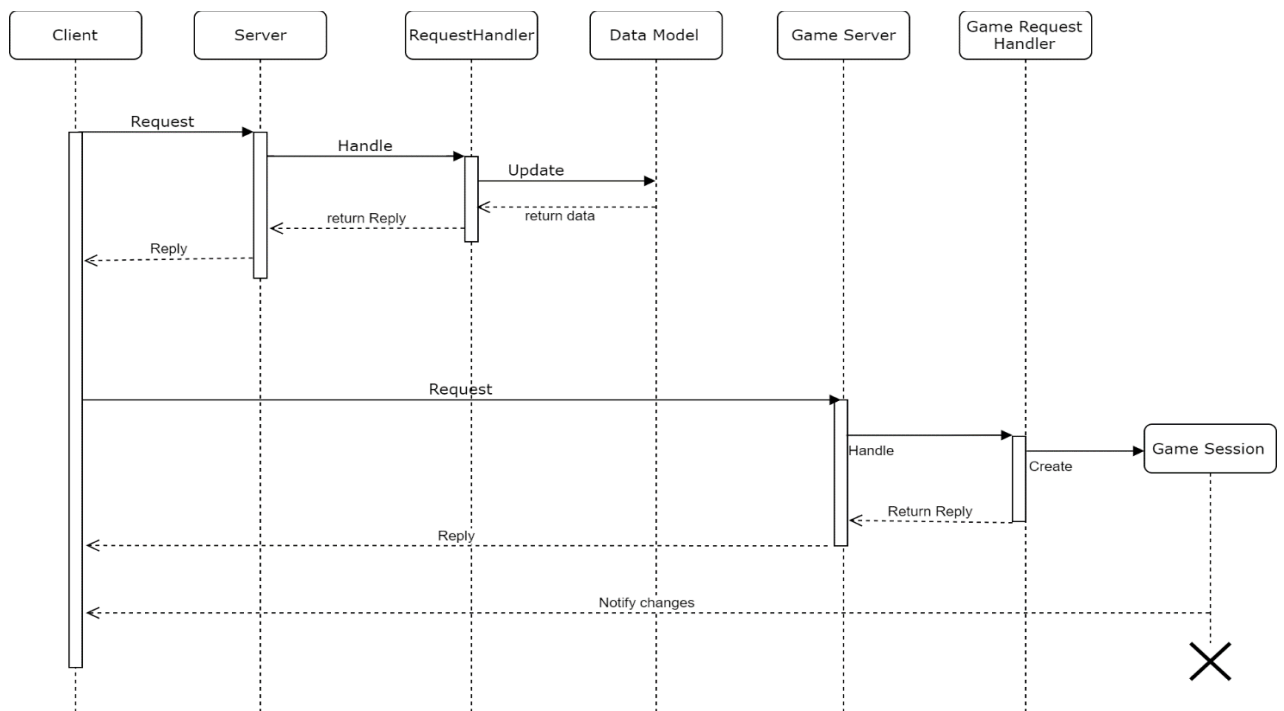


Рисунок 3.6 – Діаграма послідовності основного процесу програмного забезпечення комп'ютерної гри з мережевою взаємодією

3.3 Особливості розробки програмного забезпечення комп'ютерної гри з мережевою взаємодією

Для ігрового та головного серверів програмного забезпечення комп'ютерної гри з мережевою взаємодією основною архітектурною проблемою є обробка TCP-з'єднань, отримання запитів та формування відповідей.

Для клієнта програмного забезпечення комп'ютерної гри з мережевою взаємодією основною архітектурною проблемою є відображення ігрових даних (моделей) в інтерфейсі користувача.

Для успішної обробки з'єднань, отримання запитів та формування відповідей у програмному забезпеченні комп'ютерної гри з мережевою взаємодією використано архітектурне рішення Model – Controller. Зазвичай у цьому підході присутній і View, але сервери повертають запити не у вигляді вебсторінок, а у вигляді даних, які клієнт відображає самостійно.

Для успішного відображення можна застосувати архітектурне рішення Model – View – Presenter. Воно дозволяє відокремити відображення від моделі даних, що забезпечує незалежність модулів один від одного.

Для розробки було обрано фреймворк Qt, який пропонує велику бібліотеку з більш ніж 1200 класів для різноманітних рішень. Ось як цей фреймворк полегшить розробку системи:

- містить модуль network, який включає класи з кросплатформеним кодом для обробки мережевої взаємодії;
- підтримує систему подій та сигналів-слотів (патерн проектування Observer);
- забезпечує систему відображення модель – уявлення (Model – View);
- має модуль з віджетами для створення графічного інтерфейсу користувача.

Таким чином, архітектура клієнтської частини спрощується завдяки вбудованій системі модель-уявлення (Model – View) у фреймворку Qt.

Мережева взаємодія стає простішою, оскільки Qt забезпечує кросплатформену роботу, усуваючи необхідність писати різний код для кожної платформи. Система сигналів у Qt полегшує розробку, оскільки об'єкти можуть бути підписані на події, що їх цікавлять, і обробляти їх відповідним чином. Qt має зручний набір класів для роботи з потоками та паралельними обчисленнями, що може прискорити роботу серверних компонентів.

3.4 Проєктування інтерфейсу програмного забезпечення комп'ютерної гри з мережевою взаємодією

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією виконано з урахуванням вимог технічного завдання. При розробленні програмного забезпечення комп'ютерної гри з мережевою взаємодією враховані функціональні вимоги до програми:

- підтримка можливості реєстрації та авторизації користувачів на сервері, який є консольним застосунком;
- підтримка можливості забезпечення мережевої взаємодії гравців з сервером;
- можливість перегляду статистики гравців;
- можливість вибору комп'ютерної гри;
- підтримка можливості перевірки коректності введених гравцем даних;
- підтримка ігрового процесу комп'ютерної гри, обраної користувачем.

Інтерфейс головної форми програмного забезпечення комп'ютерної гри з мережевою взаємодією наведено на рис. 3.7.



Онлайн ігри — Нові ігри

НОВІ ІГРИ ★★★★★

НОВІ

КРАЩІ

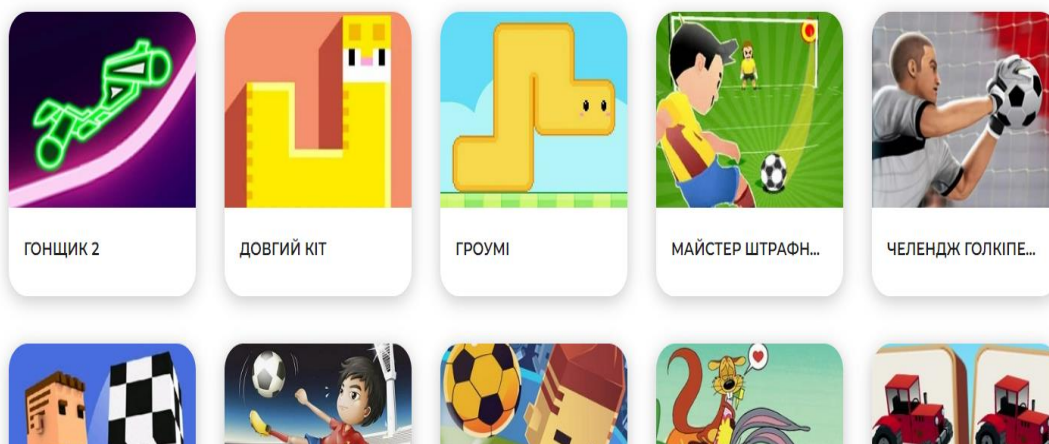


Рисунок 3.7 – Інтерфейс головної форми програмного забезпечення комп'ютерної гри з мережевою взаємодією

Форму авторизації користувача наведено на рис. 3.8.

АВТОРИЗАЦІЯ

РЕЄСТРАЦІЯ

АВТОРИЗАЦІЯ

Ваш E-mail

Введіть пароль

ПОКАЗАТИ

☐ Запам'ятайте мене
 [Забули пароль?](#)

УВІЙТИ

Рисунок 3.8 – Форма авторизації користувача

Форму реєстрації наведено на рис. 3.9.

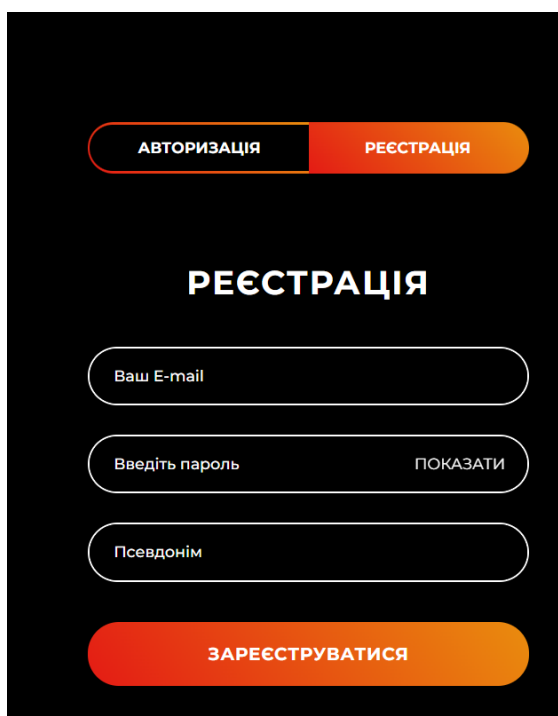
The image shows a registration form on a dark background. At the top, there are two tabs: 'АВТОРИЗАЦІЯ' (Authorization) and 'РЕЄСТРАЦІЯ' (Registration), with the latter being active. Below the tabs is the title 'РЕЄСТРАЦІЯ'. The form consists of three input fields: 'Ваш E-mail' (Your E-mail), 'Введіть пароль' (Enter password) with a 'ПОКАЗАТИ' (Show) toggle, and 'Псевдонім' (Nickname). At the bottom is a large orange button labeled 'ЗАРЕЄСТРУВАТИСЯ' (Register).

Рисунок 3.9 – Форма реєстрації користувача

Форму з прикладом підтримки ігрового процесу комп'ютерної гри, обраної користувачем, наведено на рис. 3.10.

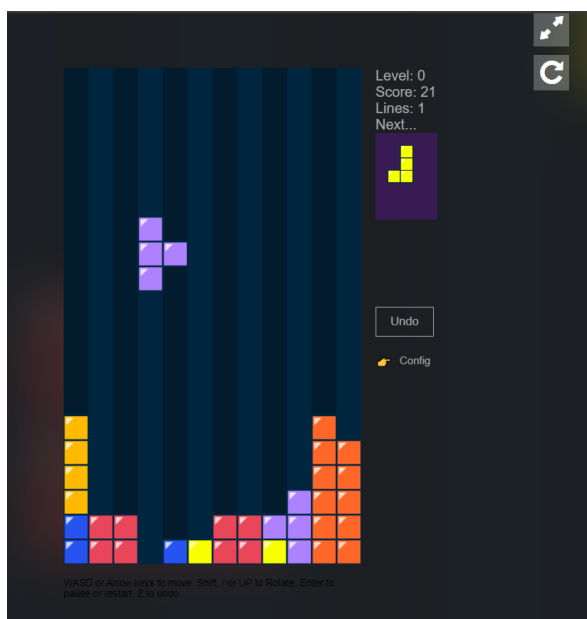


Рисунок 3.10 – Форма з прикладом підтримки ігрового процесу комп'ютерної гри, обраної користувачем

Сторінка перегляду ігрової статистики наведена на рис. 3.11.

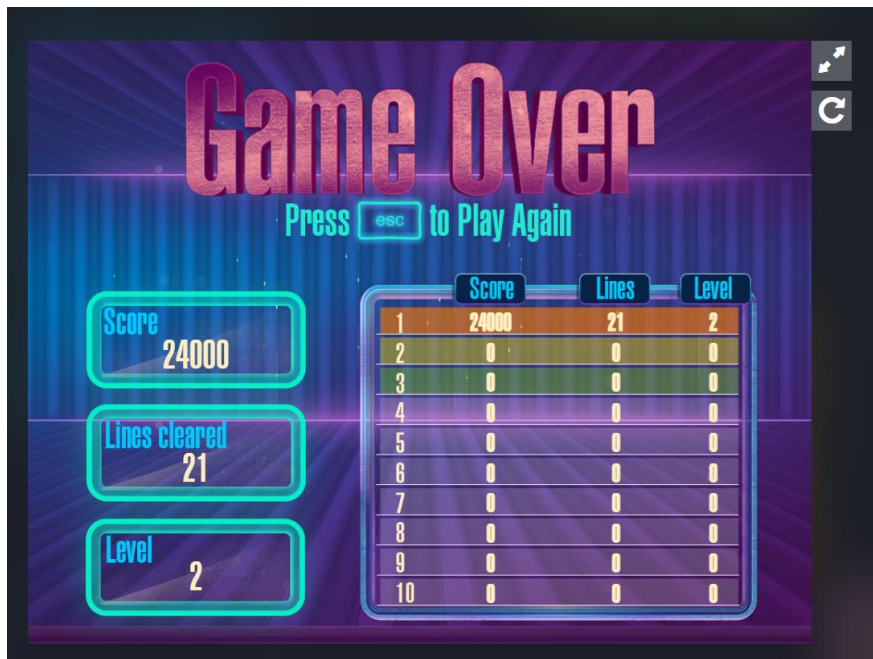


Рисунок 3.11 – Сторінка перегляду ігрової статистики

3.5 Висновки за розділом 3

Розроблено програмне забезпечення комп'ютерної гри з мережевою взаємодією.

Запропоновано структуру програмного забезпечення комп'ютерної гри з мережевою взаємодією. Основними програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано класи, що реалізують роботу серверу, ігрових моделей, зв'язку, а також запитів клієнтів та відповідей серверу.

Визначено, що розроблене програмне забезпечення комп'ютерної гри з мережевою взаємодією передбачає роботу з одним головним сервером, декількома ігровими серверами та декількома клієнтами (користувачами, гравцями). При цьому кожний клієнт програмного забезпечення комп'ютерної гри з мережевою взаємодією встановлює з'єднання з головним сервером та ігровим сервером.

Описано функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією.

Описано особливості реалізації програмного забезпечення комп'ютерної гри з мережевою взаємодією. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для підтримки процесу комп'ютерної гри з мережевою взаємодією. Програма написана на мові C++, яка є мовою високого рівня, надає засоби оптимізації мережевої взаємодії, забезпечує низькорівневий доступ до апаратних ресурсів, що особливо корисно для оптимізації графіки, звуку та мережевої взаємодії в іграх. В якості фреймворку для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано Qt, оскільки він підтримує широкий спектр платформ, має потужні засоби для розробки інтерфейсу користувача і мережевої інтеграції, високу швидкодію та добре розвинені інструменти розробки.

Програмне забезпечення комп'ютерної гри з мережевою взаємодією забезпечує виконання таких функцій:

- підтримка можливості реєстрації та авторизації користувачів на сервері, який є консольним застосунком;
- підтримка можливості забезпечення мережевої взаємодії гравців з сервером;
- можливість перегляду статистики гравців;
- можливість вибору комп'ютерної гри;
- підтримка можливості перевірки коректності введених гравцем даних;
- підтримка ігрового процесу комп'ютерної гри, обраної користувачем.

Для експлуатації програмного забезпечення комп'ютерної гри з мережевою взаємодією необхідно таке апаратне та програмне забезпечення:

- оперативна пам'ять об'ємом 2000 Мб і більше;
- не менше 1 Гб вільного простору на запам'ятовуючому пристрою апаратного засобу;
- процесор із тактовою частотою не менше 2.0 ГГц;

- наявність вебсерверу зі встановленою операційною системою;
- веббраузер на стороні клієнта та доступ до мережі Інтернет.

Функціональні характеристики розробленого програмного забезпечення комп'ютерної гри з мережевою взаємодією наведено у технічному завданні (додаток А). Фрагмент тексту програмного забезпечення комп'ютерної гри з мережевою взаємодією наведено у додатку Б.

4.2 Характеристики програми комп'ютерної гри з мережевою взаємодією

Програмне забезпечення комп'ютерної гри з мережевою взаємодією складається з декількох незалежних модулів, кожен з яких має свою окрему функцію. У розроблюваному застосунку підтримки комп'ютерної гри з мережевою взаємодією клієнтська частина містить засоби для зв'язку з користувачами. На клієнтській частині зосереджений графічний інтерфейс користувача програми комп'ютерної гри з мережевою взаємодією. На серверній частині програмного забезпечення комп'ютерної гри з мережевою взаємодією відбуваються основні обчислювальні процеси. Також сервер виступає посередником між клієнтом і базою даних, координуючи запити клієнтів програмного забезпечення комп'ютерної гри з мережевою взаємодією. Клієнти та сервер програмного забезпечення комп'ютерної гри з мережевою взаємодією взаємодіють через Інтернет.

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску програмного забезпечення комп'ютерної гри з мережевою взаємодією необхідно запустити виконуваний файл.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення комп'ютерної гри з мережевою взаємодією є інформація про користувачів, дії гравців під час гри у комп'ютерні ігри та ін.

Вихідними даними програмного забезпечення комп'ютерної гри з мережевою взаємодією є створені у базі даних записи про гравців у комп'ютерні ігри, відповіді на запити користувачів, інформація про дії суперників, результати гри та ін.

4.3.3 Повідомлення

Користувач програмного забезпечення комп'ютерної гри з мережевою взаємодією може отримати такі повідомлення: повідомлення про некоректно введені дані у процесі авторизації чи реєстрації, повідомлення про початок гри, повідомлення про завершення гри та ін.

4.4 Виконання програмного забезпечення комп'ютерної гри з мережевою взаємодією

Програмне забезпечення надає можливість користувачу грати в різноманітні комп'ютерні ігри з мережевою взаємодією (основний функціонал комп'ютерних ігор забезпечується шляхом зв'язку з сервером, що дає можливість користувачу не встановлювати на власний комп'ютер комп'ютерні ігри).

Після запуску програмного забезпечення комп'ютерної гри з мережевою взаємодією користувачу відображається головна сторінка (рис. 4.1). Тут користувачу відображається головне меню, де він має можливість обрати типи комп'ютерних ігор, знайти комп'ютерну гру за

назвою, подивитися свої улюблені ігри. Також на головній сторінці надається можливість реєстрації та авторизації.

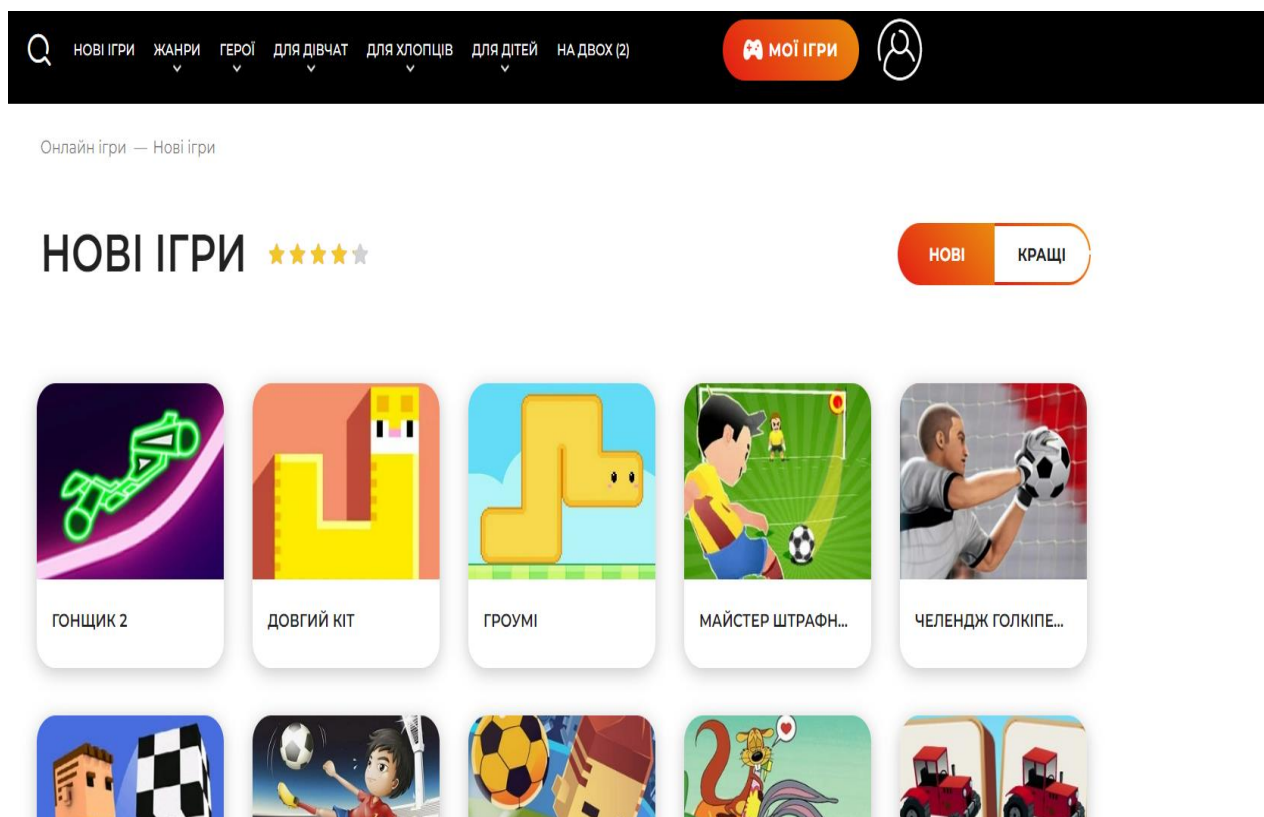
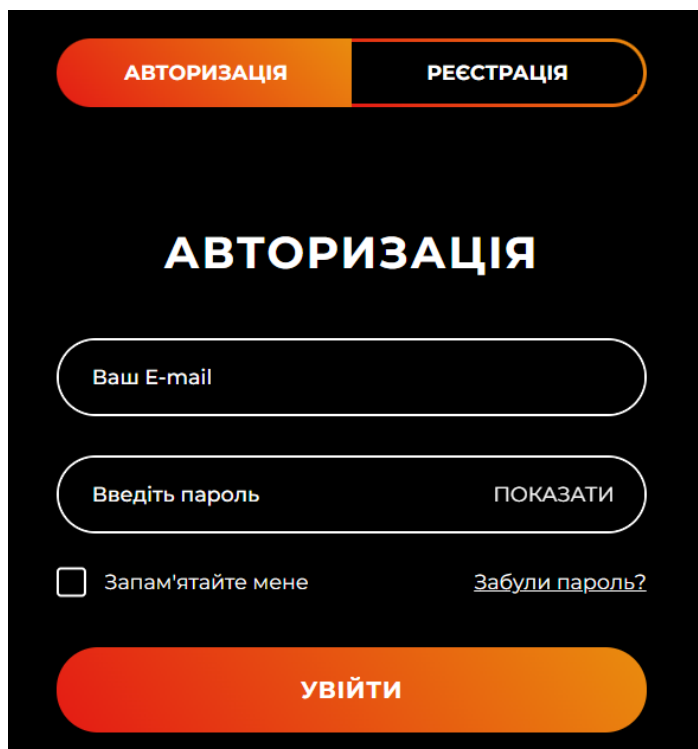


Рисунок 4.1 – Інтерфейс головної форми програмного забезпечення комп'ютерної гри з мережевою взаємодією

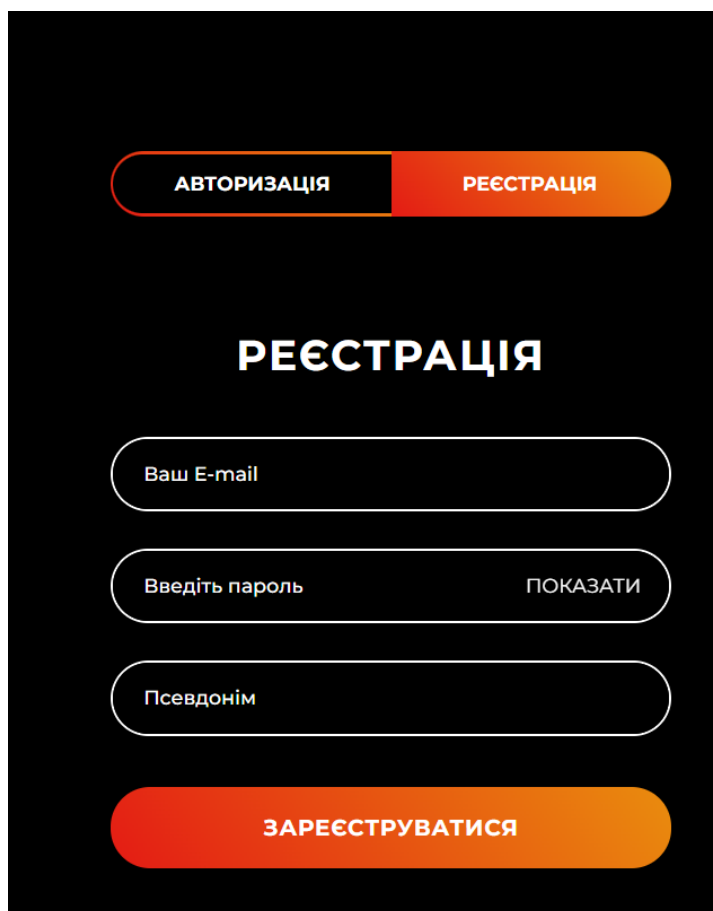
Для отримання можливості використання повного функціоналу розробленого програмного забезпечення, зокрема, збереження улюблених ігор у відповідному списку та перегляду ігрової статистики, користувачу необхідно авторизуватися (рис. 4.2).

Незареєстровані користувачі повинні пройти процедуру реєстрації. Форму реєстрації наведено на рис. 4.3.



The image shows a user authentication form on a dark background. At the top, there are two tabs: 'АВТОРИЗАЦІЯ' (highlighted in orange) and 'РЕЄСТРАЦІЯ'. Below the tabs is the title 'АВТОРИЗАЦІЯ' in large white letters. The form contains three input fields: 'Ваш E-mail', 'Введіть пароль' (with a 'ПОКАЗАТИ' toggle), and a checkbox for 'Запам'ятайте мене' with a link 'Забули пароль?'. At the bottom is a large orange button labeled 'УВІЙТИ'.

Рисунок 4.2 – Авторизація користувача



The image shows a user registration form on a dark background. At the top, there are two tabs: 'АВТОРИЗАЦІЯ' and 'РЕЄСТРАЦІЯ' (highlighted in orange). Below the tabs is the title 'РЕЄСТРАЦІЯ' in large white letters. The form contains three input fields: 'Ваш E-mail', 'Введіть пароль' (with a 'ПОКАЗАТИ' toggle), and 'Псевдонім'. At the bottom is a large orange button labeled 'ЗАРЕЄСТРУВАТИСЯ'.

Рисунок 4.3 – Реєстрація користувача

Після вибору користувачем комп'ютерної гри розпочинається ігровий процес (запускається обрана гра). При цьому забезпечується підтримка мережевої взаємодії гравців з сервером, що дає можливість користувачу не встановлювати на власний комп'ютер комп'ютерні ігри.

Приклад підтримки ігрового процесу комп'ютерної гри, обраної користувачем, наведено на рис. 4.4.

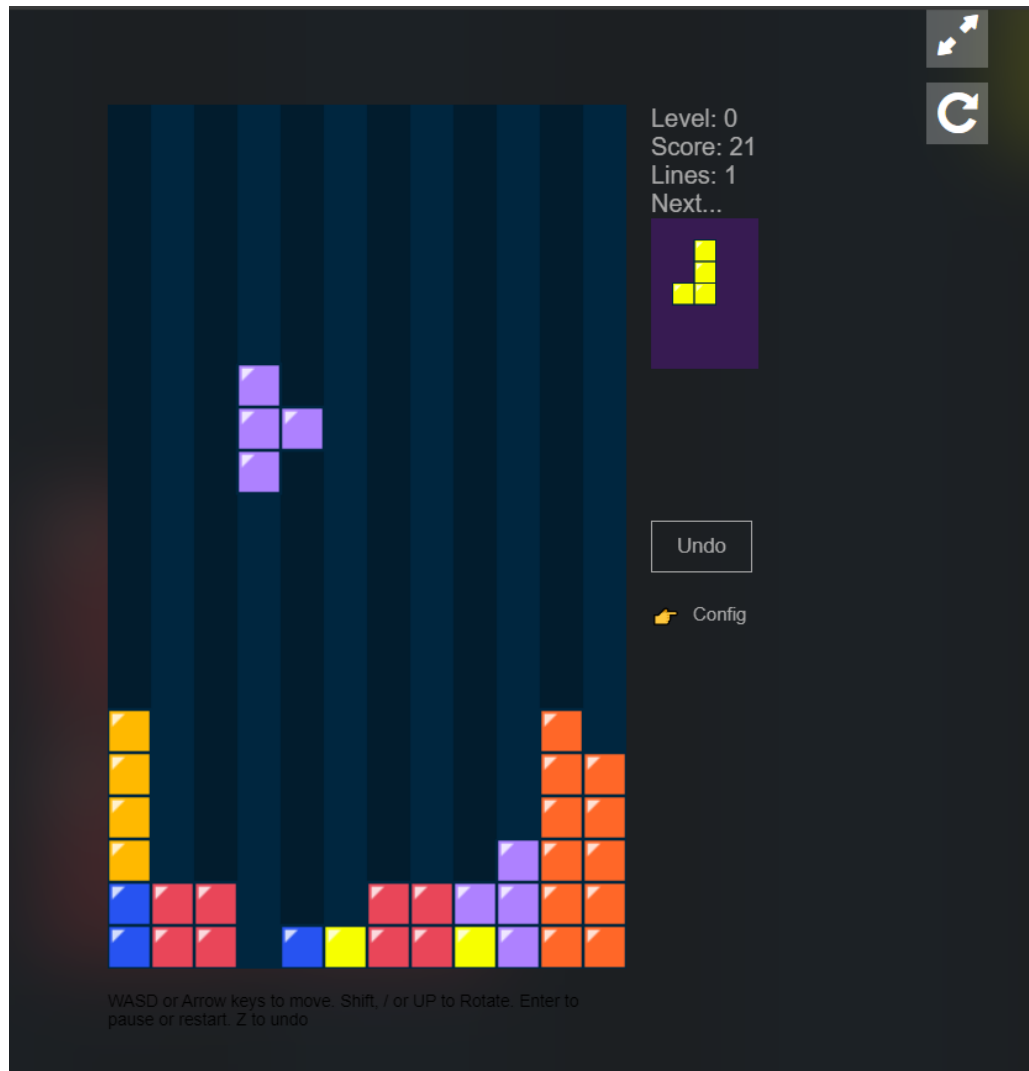


Рисунок 4.4 – Підтримка ігрового процесу комп'ютерної гри, обраної користувачем

Після завершення гри користувач має можливість перегляду ігрової статистики (рис. 4.5).



Рисунок 4.5 – Перегляд ігрової статистики

4.5 Тестування програмного забезпечення комп'ютерної гри з мережевою взаємодією

Виконано тестування програмного забезпечення комп'ютерної гри з мережевою взаємодією.

Виявлено, що програма для підтримки ігрового процесу функціонує правильно та злагоджено. Вона реалізує всі функціональні вимоги та успішно виконує свою основну задачу підтримки процесу комп'ютерної гри.

Розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з підтримкою функціонування комп'ютерних ігор.

4.6 Висновки за розділом 4

Описано програмне забезпечення комп'ютерної гри з мережевою взаємодією. Виконано тестування розробленого програмного забезпечення комп'ютерної гри з мережевою взаємодією. Результати тестування

програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з підтримкою функціонування комп'ютерних ігор.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано та досліджено процес розробки програмного забезпечення для комп'ютерних ігор.

Визначено, що комп'ютерна гра являє собою електронну гру, що передбачає взаємодію з людиною через створення візуального зворотного зв'язку. При цьому взаємодія з користувачем часто забезпечується за допомогою мережі, як правило, Інтернет, через комп'ютерні засоби, такі як персональний комп'ютер, консоль або мобільний пристрій; гравець використовує обладнання для керування грою та бачить результати своїх дій у віртуальному середовищі. Проаналізовано популярні жанри відеоігор та їхні особливості. Досліджено засоби, що підтримують ігровий процес у комп'ютерних іграх.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки комп'ютерних ігор з мережевою взаємодією. Проте деякі програмні засоби для підтримки комп'ютерних ігор з мережевою взаємодією можуть мати складну архітектуру, що може призводити до технічних проблем, зокрема, до зависання гри, проблем з підключенням до серверів та інших технічних труднощів, що негативно впливають на враження гравців від комп'ютерної гри. Деякі мережеві комп'ютерні ігри часто потребують потужного апаратного забезпечення на стороні клієнта (гравця) для оптимальної роботи, що може бути недоступним для всіх гравців. Це обмежує аудиторію гравців та може створювати нерівні умови для гравців з різними технічними та фінансовими можливостями. В деяких випадках програмне забезпечення серверів, що підтримують мережеві комп'ютерні ігри, може не справлятися з великим навантаженням під час пікових годин або після виходу великих оновлень. Це може призводити до падінь серверів, тривалих черг на підключення та інших

незручностей. Тому актуальною є розробка програмного забезпечення комп'ютерної гри з мережевою взаємодією.

Сформульовано функціональні вимоги до програмного забезпечення комп'ютерної гри з мережевою взаємодією.

Для реалізації програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано мову програмування C++, яка є мовою високого рівня, надає засоби оптимізації мережевої взаємодії, забезпечує низькорівневий доступ до апаратних ресурсів, що особливо корисно для оптимізації графіки, звуку та мережевої взаємодії в іграх.

Для створення програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано фреймворк Qt, оскільки він підтримує широкий спектр платформ, має потужні засоби для розробки інтерфейсу користувача і мережевої інтеграції, високу швидкодію та добре розвинені інструменти розробки.

Новизна роботи полягає у тому, що створено модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення комп'ютерної гри з мережевою взаємодією.

Розроблено програмне забезпечення комп'ютерної гри з мережевою взаємодією.

Запропоновано структуру програмного забезпечення комп'ютерної гри з мережевою взаємодією. Основними програмного забезпечення комп'ютерної гри з мережевою взаємодією обрано класи, що реалізують роботу серверу, ігрових моделей, зв'язку, а також запитів клієнтів та відповідей серверу.

Визначено, що розроблене програмне забезпечення комп'ютерної гри з мережевою взаємодією передбачає роботу з одним головним сервером, декількома ігровими серверами та декількома клієнтами (користувачами, гравцями). При цьому кожний клієнт програмного забезпечення комп'ютерної

гри з мережевою взаємодією встановлює з'єднання з головним сервером та ігровим сервером.

Описано функціонування програмного забезпечення комп'ютерної гри з мережевою взаємодією.

Описано особливості реалізації програмного забезпечення комп'ютерної гри з мережевою взаємодією. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією.

Виконано тестування розробленого програмного забезпечення комп'ютерної гри з мережевою взаємодією. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з підтримкою функціонування комп'ютерних ігор.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Les différents genres de jeu vidéo expliqués [Electronic resource]. – Access mode: <https://www.oldgames-fr.com/genres-jeu-video/>.
2. Les différents genres de jeu vidéo [Electronic resource]. – Access mode: <https://gamingcampus.fr/boite-a-outils/les-differents-genre-de-jeu-video.html>.
3. Comprendre les genres de jeux vidéo : Un guide pour choisir le meilleur jeu pour vous [Electronic resource]. – Access mode: <https://gameover.fr/jeux-video/>.
4. Les différents types de jeux vidéo en 2024 [Electronic resource]. – Access mode: <https://plarium.com/fr/blog/video-game-genres/>.
5. Guide des genres de jeux vidéo : 10 types de jeux vidéo populaires [Electronic resource]. – Access mode: <https://lightups.io/guide-video-game-genres>.
6. Quels sont les différents types de jeux vidéo [Electronic resource]. – Access mode: <https://stuffgaming.fr/quels-sont-les-differents-types-de-jeux-video/>.
7. The Top 11 Game Discount Sites to Buy Video Games Cheaply [Electronic resource]. – Access mode: <https://www.makeuseof.com/tag/top-5-sites-video-game-deals-bargains/>.
8. The 8 Best Gaming News Sites and Game Review Sites [Electronic resource]. – Access mode: <https://www.makeuseof.com/tag/10-websites-latest-game-reviews-gaming-news/>.
9. IsThereAnyDeal [Electronic resource]. – Access mode: <https://isthereanydeal.com/>.
10. CheapShark [Electronic resource]. – Access mode: <https://www.cheapshark.com/>.
11. Humble Bundle [Electronic resource]. – Access mode: <https://www.humblebundle.com/>.
12. C++ Introduction [Electronic resource]. – Access mode: https://www.w3schools.com/cpp/cpp_intro.asp.
13. Learn C++ Programming [Electronic resource]. – Access mode: <https://www.programiz.com/cpp-programming>.

14. Introduction to Qt [Electronic resource]. – Access mode: <https://doc.qt.io/qt-6/qt-intro.html>.

15. Qt The Future of Digital Experiences [Electronic resource]. – Access mode: <https://www.qt.io/>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення може використовуватися для підтримки процесу функціонування комп'ютерних ігор.

А.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Дослідження та програмна реалізація комп'ютерної гри з мережевою взаємодією», затверджене наказом Національного університету «Запорізька політехніка» № 398 від 07 жовтня 2024 р.

А.2 Призначення розробки

Програмний продукт призначений для підтримки процесу управління комп'ютерними іграми.

А.3 Основні вимоги до програми, що розробляється

А.3.1 Вимоги до функціональних характеристик

Програмне забезпечення комп'ютерної гри з мережевою взаємодією забезпечує виконання таких функцій:

- підтримка можливості реєстрації та авторизації користувачів на сервері, який є консольним застосунком;
- підтримка можливості забезпечення мережевої взаємодії гравців з сервером;
- можливість перегляду статистики гравців;
- можливість вибору комп'ютерної гри;
- підтримка можливості перевірки коректності введених гравцем даних;

– підтримка ігрового процесу комп'ютерної гри, обраної користувачем.

А.3.2 Вимоги до інтерфейсу програми

Інтерфейс програмного забезпечення комп'ютерної гри з мережевою взаємодією повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

А.3.3 Вимоги до надійності

Програмне забезпечення комп'ютерної гри з мережевою взаємодією повинно забезпечити надійне функціонування.

А.3.4 Умови експлуатації

Для експлуатації програмного забезпечення комп'ютерної гри з мережевою взаємодією необхідна наявність персонального комп'ютера.

А.3.5 Вимоги до складу та параметрів технічний засобів

Для експлуатації програмного забезпечення комп'ютерної гри з мережевою взаємодією необхідно таке апаратне та програмне забезпечення:

- оперативна пам'ять об'ємом 2000 Мб і більше;
- не менше 1 Гб вільного простору на запам'ятовуючому пристрою апаратного засобу;
- процесор із тактовою частотою не менше 2.0 ГГц;

- наявність вебсерверу зі встановленою операційною системою;
- веббраузер на стороні клієнта та доступ до мережі Інтернет.

A.3.6 Вимоги до маркування і пакування

Програма для підтримки комп'ютерної гри з мережевою взаємодією може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення комп'ютерної гри з мережевою взаємодією».

A.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення комп'ютерної гри з мережевою взаємодією є інформація про користувачів, дії гравців під час гри у комп'ютерні ігри та ін.

Вихідними даними програмного забезпечення комп'ютерної гри з мережевою взаємодією є створені у базі даних записи про гравців у комп'ютерні ігри, відповіді на запити користувачів, інформація про дії суперників, результати гри та ін.

ДОДАТОК Б
Фрагмент тексту програми

```

TcTSrvr::TcTSrvr(quint16 rpt, QObject *prnt) : QObject(prnt)
{
    msrvr = new Qtrpsrv(this);
    vrqwstHndlr = new RqwstHndlr(this);

    knct(msrvr, &Qtrpsrv::ncnct,
          this,    &TcTSrvr::onNcnct);

    if (!msrvr->listen(Qhstdrs::Any, rpt))
    {
        std::cerr << "Не запущено сервер, проблемы з портом " << rpt << ".\n";
        exit(0);
    }
}

void TcTSrvr::onNcnct()
{
    Ktskt * klnt = msrvr->nextPhnrtKntnf();
    Kntnf * ncnct = new Kntnf(klnt, this);

    knct(ncnct, &Kntnf::rqwstReady,
          vrqwstHndlr,    &RqwstHndlr::onRqwstReady);
}

class Rqwst
{
public:
    Rqwst(Kntnf* sender);
    virtual ~Rqwst() = 0;
    virtual Krthyf hrft(RqwstHndlr* hndlr) = 0;

    enum class Type
    {
        Pomlk = 0,
        Tstr = 1,
        FndGr = 2,
        Pidklch = 3,
        PidKrstv = 4,
        GtStat = 5,
        Nevid = 1000
    };
};

```

```

private:
    Kntnf* vsndr;
};

class RqwstHndlr : public QObject
{
    Q_OBJECT
public:
    explicit RqwstHndlr(QObject *prnt = nullptr);

    Krthyf hrft(const TstrRqwst &rqwst);
    Krthyf hrft(const NevidRqwst &rqwst);
    Krthyf hrft(const PomlkRqwst &rqwst);

    Krthyf hrft(const PidklchRqwst &rqwst);
    Krthyf hrft(const PidKrstvRqwst &rqwst);
    Krthyf hrft(const FndGrRqwst &rqwst);
    Krthyf hrft(const GtStatRqwst &rqwst);

void RqwstHndlr::onRqwstReady(std::shrd_ptr<Rqwst> rqwst)
{
    Kntnf * kntnf = static_cast<Kntnf*>(sender());

    Krthyf krthyf = rqwst->hrft(this);

    kntnf->onKrthyfReady(krthyf);
}

class Kntnf : public QObject
{
public:
    explicit Kntnf(Ktskt *klnt, QObject *prnt = nullptr);
    ~Kntnf();

public slots:
    void onReadyRead();
    void onKrthyfReady(const Krthyf &krthyf);

signals:
    void rqwstReady(std::shrd_ptr<Rqwst> rqwst);

```

```

private:
    Ktskt *vklnt;
};

Kntnf::Kntnf(Ktskt *klnt, QObject *prnt) : QObject(prnt)
{
    vklnt = klnt;
    knct(vklnt, &Ktskt::readyRead, this, &Kntnf::onReadyRead);

    knct(vklnt, &Ktskt::diskncted,
          this,      &Kntnf::deleteLater);
}

Kntnf::~~Kntnf()
{
    delete vklnt;
    vklnt = nullptr;
}

void Kntnf::onReadyRead()
{
    QByteArray dany = vklnt->readAll();
    std::shrd_ptr<Rqwst> rqwst = std::make_shrd<NevidRqwst>(dany, this);
    emit rqwstReady(rqwst);
}

bool DanybaseKntor::createPostgreSqlKntnf() const
{
    QSqlDanybase kntnf = QSqlDanybase::addDanybase("QPSQL", ttt::kntnfName);

    kntnf.setHostName("localhost");
    kntnf.setDanybaseName(ttt::danybaseName);
    kntnf.setUserName("UsrNm");
    kntnf.setPswrd("pswrd");
    kntnf.setRpt(5432);
    bool opened = kntnf.open();

    if (kntnf.lastError().isValid())
    {
        qDebug() << kntnf.lastError();
    }
    declareTables();
    return opened;
}

```

```

}

bool DanybaseKnctor::declareTables() const
{
    bool vsiTablCrtd = true;

    vsiTablCrtd = vsiTablCrtd && crtTblUsrs();

    return vsiTablCrtd;
}

bool DanybaseKnctor::crtTblUsrs() const
{
    QString strZapt = QString("CREATE TABLE IF NOT EXISTS %1 ("
                                " %2 SERIAL, "
                                " %3 VARCHAR(32) UNIQUE, "
                                " %4 VARCHAR(32), "
                                " %5 VARCHAR(64) "
                                ");")
        .arg(ttt::Krstvchs::tblnm)
        .arg(ttt::Krstvchs::id)
        .arg(ttt::Krstvchs::usrLgn)
        .arg(ttt::Krstvchs::userPswrd)
        .arg(ttt::Krstvchs::avtTkn);

    QSqlDanybase kntnf = QSqlDanybase::danybase(ttt::kntnfName);
    QSqlZapt zapt(kntnf);

    if (!zapt.exec(strZapt))
    {
        qDebug() << zapt.lastError();
        return false;
    }

    return true;
}

Krthyf RqwstHndlr::hrft(const PidklchRqwst &rqwst)
{
    QJsonObject krthyfObject;
    krthyfObject.insert("type", static_cast<int>(Krthyf::Type::Pidklch));

    vmdl->setTable(ttt::Krstvchs::tblnm);

```

```

vmdl->setFilter(QString("%1='%2'")
                .arg(ttt::Krstvchs::usrLgn)
                .arg(rqwst.usrLgn()));
vmdl->select();

int klkRyadk = vmdl->klkRyadk();
if (klkRyadk == 0)
{
    krthyfObject.insert("stds", "fail");
    krthyfObject.insert("povidoml", "Нема користувача.");
}
else if (klkRyadk > 1)
{
    krthyfObject.insert("stds", "fail");
    krthyfObject.insert("povidoml", "Внутрішня помилка.");
}
else
{
    QSqlRkrd user = vmdl->rkrd(0);

    if (rqwst.userPswrd() ==
user.value(ttt::Krstvchs::userPswrd).toString())
    {
        krthyfObject.insert("stds", "success");
        krthyfObject.insert("avtTkn",
                            user.value(ttt::Krstvchs::avtTkn).toString());
        krthyfObject.insert("usrLgn",
                            rqwst.usrLgn());
    }
    else
    {
        krthyfObject.insert("stds", "fail");
        krthyfObject.insert("povidoml", "Невірний пароль.");
    }
}

return
Krthyf(QJsonDocument(krthyfObject).toJson(QJsonDocument::JsonFormat::Compact));
}

Krthyf RqwstHndlr::hrft(const PidKrstvRqwst &rqwst)
{
    QJsonObject krthyfObject;

```

```

    krthyfObject.insert("type", static_cast<int>(Krthyf::Type::PidKrstv));

    vmdl->setTable(ttt::Krstvchs::tblnm);

    QSqlRkrd rkrd = vmdl->rkrd();

    rkrd.remove(rkrd.indexOf(ttt::Krstvchs::id));
    rkrd.setValue(ttt::Krstvchs::usrLgn,
                  rqwst.usrLgn());
    rkrd.setValue(ttt::Krstvchs::userPswrd,
                  rqwst.userPswrd());
    rkrd.setValue(ttt::Krstvchs::avtTkn,
                  QUuid::createUuid().toString());

    if (vmdl->insertRkrd(-1, rkrd))
    {
        krthyfObject.insert("stds", "success");
        vmdl->submitAll();
    }
    else
    {
        krthyfObject.insert("stds", "fail");
        krthyfObject.insert("povidoml", vmdl->lastError().text());
    }

    return Krthyf(QJsonDocument(krthyfObject)
                  .toJson(QJsonDocument::JsonFormat::Compact));
}

Krthyf RqwstHndlr::hrft(const FndGrRqwst &rqwst)
{
    QJsonObject krthyfObject;
    krthyfObject.insert("type", static_cast<int>(Krthyf::Type::FndGr));

    vmdl->setTable(ttt::Krstvchs::tblnm);
    vmdl->setFilter(QString("%1='%2'"
                           " and "
                           "%3='%4'")
                  .arg(ttt::Krstvchs::usrLgn)
                  .arg(rqwst.usrLgn())
                  .arg(ttt::Krstvchs::avtTkn)
                  .arg(rqwst.avtTkn().toString()));

    vmdl->select();

```

```

int klkRyadk = vmdl->klkRyadk();
if (klkRyadk == 0)
{
    krthyfObject.insert("stds", "fail");
    krthyfObject.insert("povidoml", "Auth failure");
}
else if (klkRyadk > 1)
{
    krthyfObject.insert("stds", "fail");
    krthyfObject.insert("povidoml",
                        "Внутрішня помилка.");
}
else
{
    krthyfObject.insert("stds", "success");
    krthyfObject.insert("povidoml", "Пошук опонента");
    addUserToFndGr(rqwst.sender());
}

return Krthyf();
return Krthyf(QJsonDocument(krthyfObject)
               .toJson(QJsonDocument::JsonFormat::Compact));
}

Krthyf RqwstHndlr::hrft(const GtStatRqwst &/*rqwst*/)
{
    return Krthyf();
}

void RqwstHndlr::onRqwstReady(std::shrd_ptr<Rqwst> rqwst)
{
    Kntnf * kntnf = static_cast<Kntnf*>(sender());

    Krthyf krthyf = rqwst->hrft(this);

    kntnf->onKrthyfReady(krthyf);
}

void RqwstHndlr::addUserToFndGr(Kntnf *kntnf)
{
    Kntnf * opnt = findOpnt();
    if (opnt != nullptr)

```

```

{
    vmtchMkng.removeOne(opnt);

    QJsonObject krthyfObject;
    krthyfObject.insert("type", static_cast<int>(Krthyf::Type::GameFound));
    krthyfObject.insert("stds", "success");
    krthyfObject.insert("gmSrvip", "localhost");
    krthyfObject.insert("gameServerRpt", 3232);
    krthyfObject.insert("gmnum", QUuid::createUuid().toString());

    Krthyf krthyf(QJsonDocument(krthyfObject)
                  .toJson(QJsonDocument::JsonFormat::Compact));
    opnt->onKrthyfReady(krthyf);
    kntnf->onKrthyfReady(krthyf);
}
else
{
    vmtchMkng.push_back(kntnf);
}
}

Kntnf *RqwstHndlr::findOpnt() const
{
    if (vmtchMkng.empty())
        return nullptr;
    return vmtchMkng.first();
}

GameWidget::GameWidget(quint16 rpt, const QUuid & gmnum, QWidget *prnt) :
    QWidget(prnt),
    ui(new Ui::GameWidget),
    vgmnum(gmnum)
{
    ui->setupUi(this);
    vsckt = new Ktskt(this);

    knct(vsckt, &Ktskt::readyRead,
          this, &GameWidget::onReadyRead);

    vsckt->knctToHost("localhost", rpt);

    for (int i = 0; i < 15; ++i)

```

```

{
    for (int j = 0; j < 15; ++j)
    {
        QPushKnpk * knpk = new QPushKnpk("");
        knpk->setSizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
        knpk->setMinimumSize(QSize(30, 30));
        knpk->setMaximumSize(knpk->minimumSize());
        knct (knpk, &QPushKnpk::clicked,
               this,    &GameWidget::onFieldClicked);
        ui->gridLayout->addWidget(knpk, i, j);
    }
}

knctToGame();
}

```

ДОДАТОК В
Слайди презентації

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»
Кафедра програмних засобів

Дослідження та програмна реалізація комп'ютерної гри з мережевою взаємодією

Виконав
ст. гр. КНТ-113м

Володимир СТРИЖАКОВ

Керівник
д.т.н., професор

Андрій ОЛІЙНИК

Рисунок В.1 – Слайд 1

Об'єктом дослідження є процес розробки програмного забезпечення для комп'ютерних ігор.

Предмет дослідження – програмні засоби для підтримки комп'ютерних ігор.

Мета роботи – дослідження та розробка програмного забезпечення для комп'ютерної гри з мережевою взаємодією.

Рисунок В.2 – Слайд 2

Задачі роботи:

- виконати аналіз предметної області та програмних засобів для комп'ютерної гри з мережевою взаємодією;
- розробити модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією;
- здійснити проєктування програмного забезпечення комп'ютерної гри з мережевою взаємодією;
- створити програмне забезпечення комп'ютерної гри з мережевою взаємодією;
- виконати тестування розробленого програмного забезпечення комп'ютерної гри з мережевою взаємодією.

3

Рисунок В.3 – Слайд 3

Аналіз програмного забезпечення комп'ютерних ігор з мережевою взаємодією

Критерій порівняння	IsThereAny Deal	CheapShark	Humble Bundle
Засоби цінових оповіщень	+	+	–
Фільтри та налаштування	+–	+	+–
Спільнота та огляди	+–	+	–
Відстеження історії цін на комп'ютерні ігри	+	+–	–
Зручність інтерфейсу	+–	+	+–
Можливість порівняння магазинів комп'ютерних ігор	+	+–	–
Інтеграція з іншими сервісами	+–	+	+–
Засоби API для зв'язку зі сторонніми програмами	–	+	–

4

Рисунок В.4 – Слайд 4

Вибір мови програмування

Критерій порівняння мов програмування	Мова програмування		
	C++	Python	Ruby
Зручність засобів для роботи з мережею	+	+-	+
Багатозадачність	+	+	+-
Ефективність організації системної роботи	+	-	+-
Зручність для створення ПЗ комп'ютерної гри з мережевою взаємодією	+	-	-
Засоби оптимізації мережевої взаємодії	+	+-	+-

5

Рисунок В.5 – Слайд 5

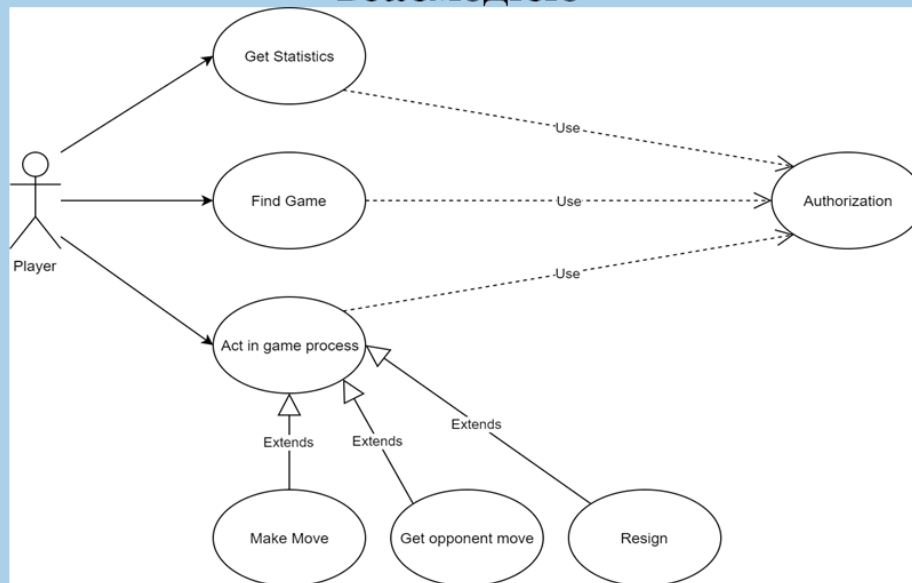
Обґрунтування вибору фреймворку

Критерій порівняння фреймворків	Фреймворки		
	Qt	GTK	Flutter
Мова програмування	C++ і QML	C і інші мови через зв'язки	Dart і JavaScript
Швидкодія	Висока швидкодія завдяки оптимізаціям і компіляції	Висока швидкодія, але менша ніж у Qt	Висока швидкодія, оптимізована для мобільних платформ
Інтеграція з мережевими технологіями	Повна підтримка мережних бібліотек та протоколів	Підтримка мережних бібліотек	Нативна підтримка HTTP та мережних запитів
Засоби розробки	Qt Creator і Qt Designer для розробки і UI	Glade для розробки і GTK Inspector для відлагодження	Власний інструмент розробки та відлагодження
Відкритість та спільнота	Ліцензія LGPL або комерційна ліцензія	Ліцензія LGPL або GPL, активна спільнота	Ліцензія BSD, активна спільнота

6

Рисунок В.6 – Слайд 6

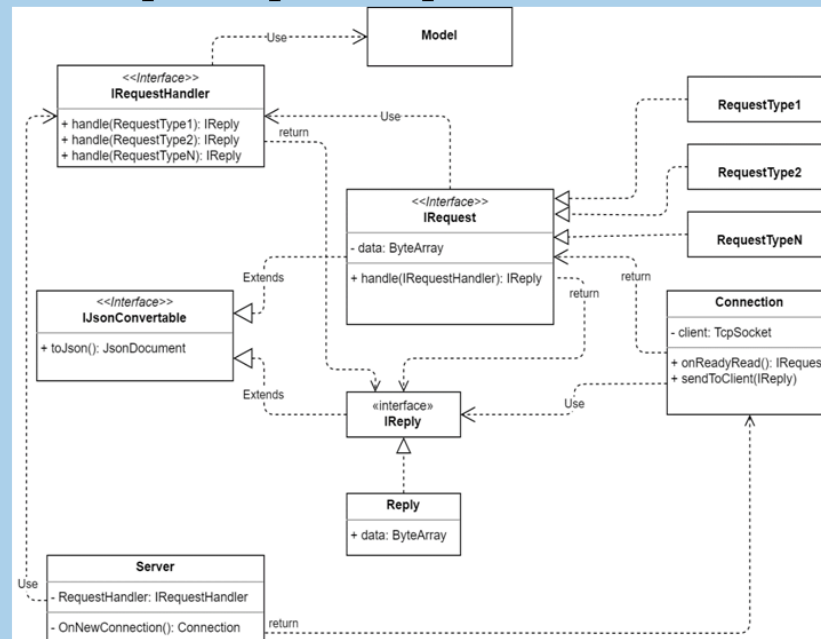
Модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією



7

Рисунок В.7 – Слайд 7

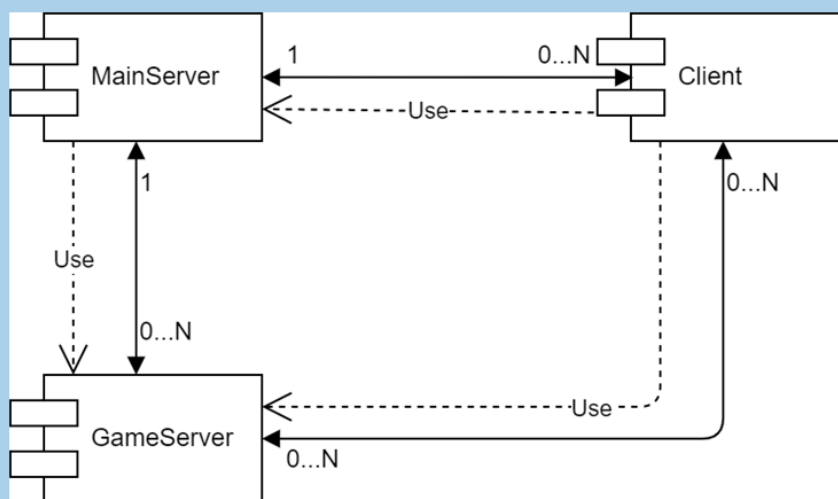
Структура програмного забезпечення комп'ютерної гри з мережевою взаємодією



8

Рисунок В.8 – Слайд 8

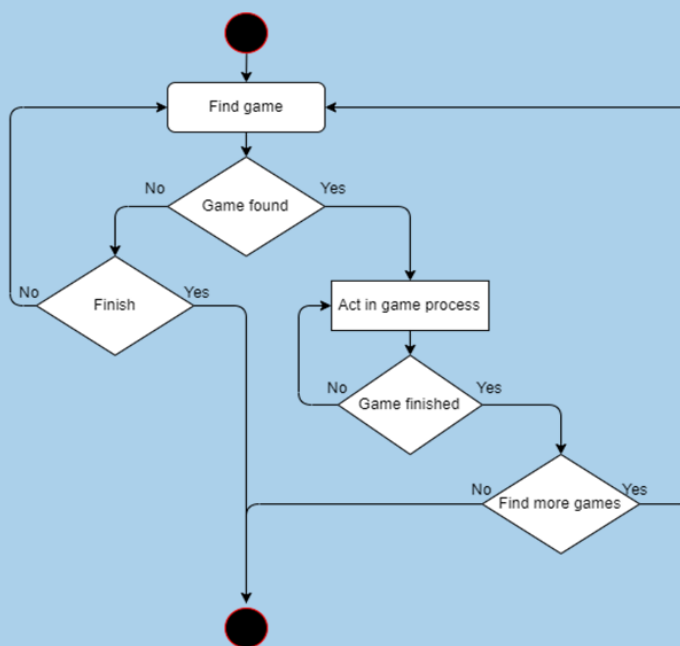
Діаграма компонентів



9

Рисунок В.9 – Слайд 9

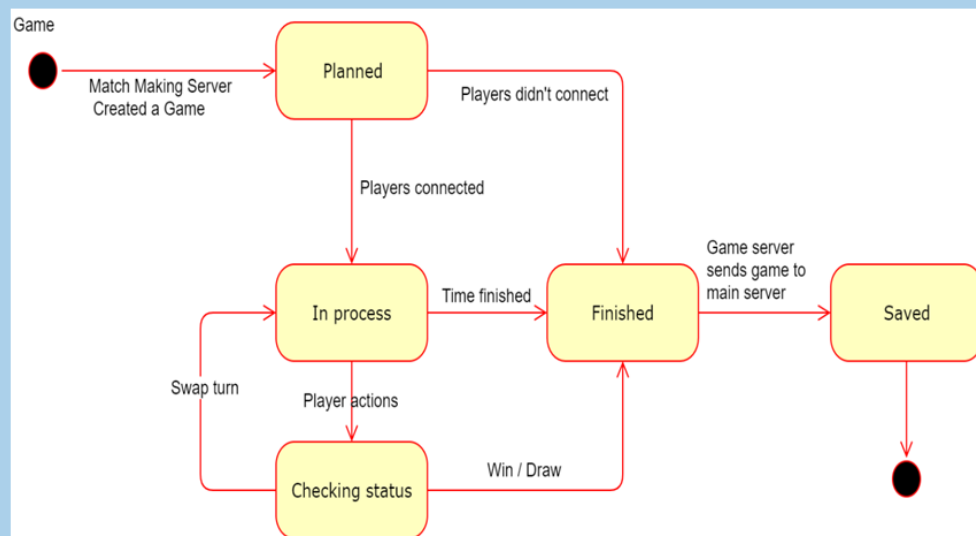
Схема функціонування ПЗ у вигляді діаграми діяльності



10

Рисунок В.10 – Слайд 10

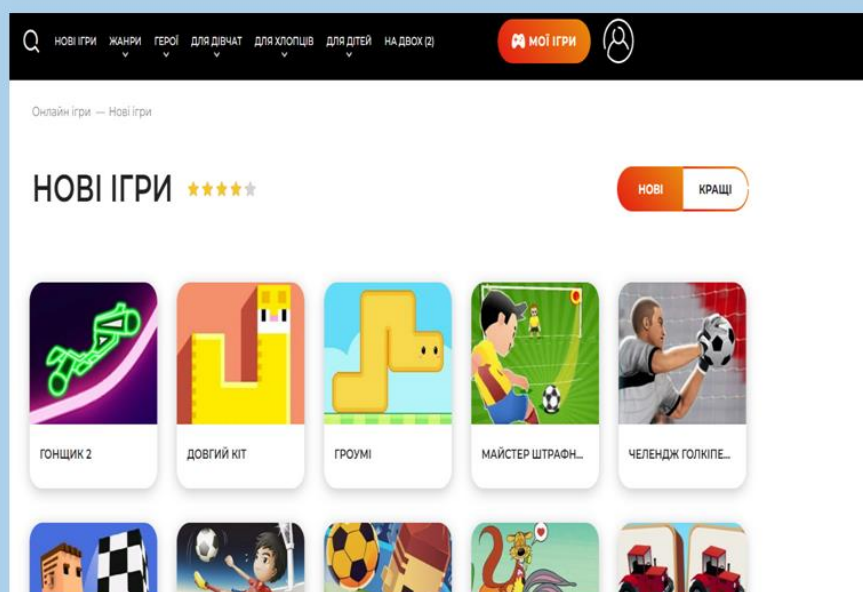
Діаграма станів програмного забезпечення комп'ютерної гри з мережевою взаємодією



11

Рисунок В.11 – Слайд 11

Робота з програмою. Головна сторінка



12

Рисунок В.12 – Слайд 12

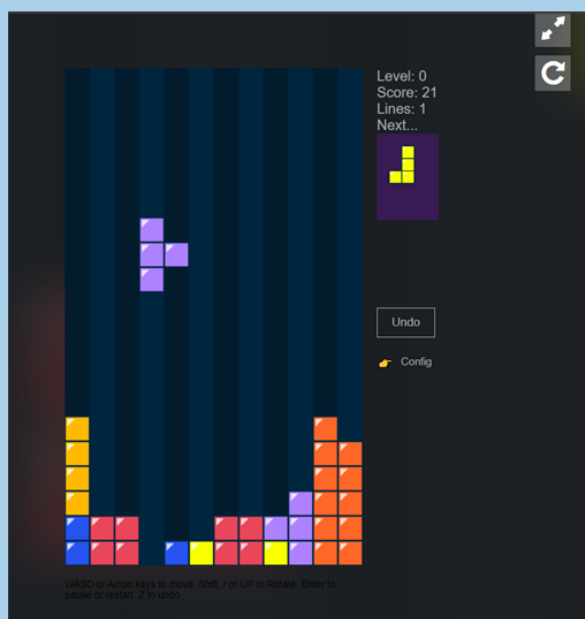
Робота з програмою. Форма реєстрації та авторизації користувача

The image shows two mobile app screens for user registration and login. The left screen is titled 'РЕЄСТРАЦІЯ' (Registration) and features a dark background with orange and white text. It has a toggle for 'РЕЄСТРАЦІЯ' and 'АВТОРИЗАЦІЯ' (Login). Below the toggle are three input fields: 'Ваш E-mail', 'Введіть пароль' (with a 'ПОКАЗАТИ' toggle), and 'Псевдонім'. At the bottom is a large orange button labeled 'ЗАРЕЄСТРУВАТИСЯ'. The right screen is titled 'АВТОРИЗАЦІЯ' (Login) and has a similar layout. It features a toggle for 'АВТОРИЗАЦІЯ' and 'РЕЄСТРАЦІЯ'. Below the toggle are two input fields: 'Ваш E-mail' and 'Введіть пароль' (with a 'ПОКАЗАТИ' toggle). There is a checkbox for 'Запам'ятайте мене' (Remember me) and a link 'Забули пароль?' (Forgot password?). At the bottom is a large orange button labeled 'УВІЙТИ' (Login).

13

Рисунок В.13 – Слайд 13

Підтримка ігрового процесу комп'ютерної гри, обраної користувачем



14

Рисунок В.14 – Слайд 14

Перегляд ігрової статистики



15

Рисунок В.15 – Слайд 15

Висновки

В ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано та досліджено процес розробки програмного забезпечення для комп'ютерних ігор.

Новизна роботи полягає в тому, що створено модель взаємодії користувача з програмним забезпеченням комп'ютерної гри з мережевою взаємодією, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення комп'ютерної гри з мережевою взаємодією.

Практичне значення роботи полягає у тому, що розроблено програмне забезпечення комп'ютерної гри з мережевою взаємодією.

Результати тестування програмного забезпечення показали, що розроблена програма може використовуватися за призначенням.

16

Рисунок В.16 – Слайд 16