

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки,
Факультет комп'ютерних наук і технологій

(повне найменування інституту, назва факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
ДЛЯ ЛІКУВАННЯ ПСИХІЧНИХ РОЗЛАДІВ ЗА ДОПОМОГОЮ ТЕХНОЛОГІЇ
ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ
RESEARCH AND DEVELOPMENT OF SOFTWARE FOR TREATING
MENTAL DISORDERS VIA VR TECHNOLOGY

Виконав: студент(ка) 2 курсу, групи КНТ-120м
Спеціальності 121 Інженерія програмного
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)
Інженерія програмного забезпечення

Малюков М.К.

(прізвище та ініціали)

Керівник Пархоменко А.В.

(прізвище та ініціали)

Рецензент Зеленьова І.Я.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Інститут, факультет ІРЕ, ФКНТ
 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.
С.О. Субботін
 “ ” 2021 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Малюкова Михайла Костянтиновича

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та розробка програмного забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності. Research and Development of Software for Treating Mental Disorders via VR Technology

керівник проєкту (роботи) Пархоменко А. В., к.т.н., доцент,
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “ 5 ” листопада 2021 року № 435

2. Строк подання студентом проєкту (роботи) 16 грудня 2021 року

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

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Дослідження технологій віртуальної реальності для лікування пацієнтів 3. Проєктування програмного забезпечення. 4. Вибір інструментарію розробки. 5. Конструювання програмного забезпечення. 6. Керівництво з використання програмного забезпечення.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-6 Основна частина	Пархоменко А.В., доцент		
Нормоконтроль	Дейнега Л.Ю., ст. викладач		

7. Дата видачі завдання “ 2 ” вересня 2021 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	2-5 тижні	Розділ 1 – 2
3	Розробка архітектури програми	6-8 тиждень	Розділ 2 - 4
4	Розробка програми	9-12тижні	Розділ 5
5	Тестування та експериментальне дослідження програми	12-14 тиждень	Розділ 6
6	Оформлення пояснювальної записки та документів до неї. Нормоконтроль та рецензування	15 тиждень	ПЗ, Додатки
7	Захист роботи	16 тиждень	

Студент(ка)

(підпис) Малюков М.К.
(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис) Пархоменко А.В.
(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
136 с. , 46 рис., 19 табл., 2 дод., 82 джерел.

ПСИХІЧНІ ЗАХВОРЮВАННЯ, ЛІКУВАННЯ, ВІРТУАЛЬНА
РЕАЛЬНІСТЬ, UNREAL ENGINE, ПЛАТФОРМА ДЛЯ ЛІКУВАННЯ.

Об'єкт дослідження – процес лікування психічних захворювань з застосуванням спеціалізованого програмного забезпечення.

Предмет дослідження – технології віртуальної реальності для лікування психічних захворювань.

Мета роботи – дослідження та практична реалізація програмного забезпечення для лікування психічних захворювань за допомогою технології віртуальної реальності.

Матеріали, методи та технічні засоби: структурне та об'єктно-орієнтоване програмування, інтегроване середовище розробки PyCharm, мова програмування Python, рушій Unreal Engine, СКБД PostgreSQL, Docker.

Результати. Розроблено програмне забезпечення, що надає платформу для лікування психічних розладів за допомогою технології віртуальної реальності.

Висновки. Програмне забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності дозволяє взаємодіяти пацієнтам та лікарям, призначати терапію пацієнтам та створювати сценарії для лікування.

Галузь використання – медицина, розділ психічних розладів.

ABSTRACT

Explanatory note to the diploma qualifying work of the magister: 136 pages, 46 figures, 19 tables, 2 appendixes, 82 sources.

MENTAL DISEASES, TREATMENT, VIRTUAL REALITY, UNREAL ENGINE, TREATMENT PLATFORM.

The object of the study is a process of mental disorders treatment using special software.

The subject of research is virtual reality technology for the treatment of mental disorders.

The aim of the work is research and practical implementation of software for the treatment of mental disorders using virtual reality technology.

Materials, methods, and tools: structural and object-oriented programs, PyCharm integrated development environment, Python programming language, Unreal Engine, PostgreSQL, Docker.

Results. Software has been developed which provides platform for the treatment of mental disorders using virtual reality technology.

Conclusions. Software for the treatment of mental disorders with the help of virtual reality technologies allows patients and doctors to interact, prescribe therapy to patients and create scenarios for treatment.

The field of application is medicine, section of mental disorders.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК	9
ВСТУП	10
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
1.1 Обґрунтування необхідності лікування психічних розладів.....	12
1.1.1 Ризики захворювання психічними розладами	18
1.1.2 Охорона здоров'я та діяльність ВООЗ	19
1.2 Аналіз технології віртуальної реальності.....	20
1.3 Аналіз ефективності застосування VR технології.....	24
1.4 Огляд особливостей платформи Facebook Meta	29
1.5 Постановка задач магістерської роботи	34
2 ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ ДЛЯ ЛІКУВАННЯ ПАЦІЄНТІВ	36
2.1 Огляд технологій віртуальної реальності для лікування психічних розладів.....	36
2.1.1 Телетерапія	36
2.1.2 Підвищення електрокортикальної активності	39
2.2 Аналіз існуючого програмного забезпечення для порівняння алгоритмів лікування психічних розладів	42
2.2.1 Постінсультні психічні розлади	42
2.2.2 Фантомні болі.....	44
2.2.3 Фобія тварин.....	48
2.3 Висновок до розділу 2	50
3 ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	51
3.1 Аналіз вимог до системи.....	51
3.2 Вибір методу реалізації проєкту.....	52
3.2.1 Proof of Concept.....	52

	7
3.2.2 Prototype	54
3.2.3 Minimum Viable Product	54
3.3 Вибір патерну проєктування.....	57
3.3.1 MVC.....	57
3.3.2 MVVM.....	58
3.3.3 MVP (Model View Presenter)	59
3.4 Аналіз та вибір способу запуску вебзастосунку	60
3.5 Аналіз та вибір методології для керування процесом розробки.....	63
3.6 Проєктування архітектури системи	65
3.7 Проєктування структури СКБД.....	66
3.8 Висновки до розділу 3	69
4 ВИБІР ІНСТРУМЕНТАРІЮ РОЗРОБКИ.....	71
4.1 Аналіз та вибір СКБД	71
4.2 Обґрунтування вибору мови програмування	73
4.3 Аналіз та вибір середовища розробки	75
4.4 Вибір фреймворку або бібліотеки для вебзастосунку	76
4.5 Вибір програмного забезпечення для розробки VR частини	78
4.5.1 Unity3d.....	78
4.5.2 Unreal Engine	79
4.6 Висновки до розділу 4	82
5 КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	84
5.1 Конструювання вебзастосунку проєкту	85
5.2 Конструювання VR застосунку	88
5.3 Висновки до розділу 5	92
6 КЕРІВНИЦТВО З ВИКОРИСТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ..	94
6.1 Призначення та умови використання системи	94
6.2 Функції системи	94
6.3 Інструкція з запуску системи.....	95

	8
6.4 Початок роботи з програмним забезпеченням.....	98
6.5 Висновки до розділу 6	102
ВИСНОВКИ.....	103
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	104
ДОДАТОК А Текст програми.....	112
ДОДАТОК Б Слайди презентації	129

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API	– Application Programming Interface;
JSON	– JavaScript Object Notatin;
HMD	– Head Mounted Display;
IDE	– Integrated Development Environment;
MVC	– Model View Controller;
ORM	– Object-Relational Mapping;
PoC	– Proof of concept;
PEP	– Python Enhancement Proposals;
UE4	– Unreal Engine 4;
UnrealEd	– Unreal editor.
VR	– Virtual reality;
БД	– база даних;
ВООЗ	– Всесвітня організація охорони здоров'я;
ООП	– об'єктно-орієнтоване програмування;
ОС	– операційна система;
СКБД	– система контролю баз даних.

ВСТУП

Проблема психічних захворювань має великий обсяг на сьогоднішній день. Психічні та поведінкові розлади широко поширені в усьому світі. Всесвітня організація охорони здоров'я (ВООЗ) стверджує, що кожна четверта людина протягом життя може мати розлад психічного здоров'я [1]. У 2016 році близько 16% населення світу постраждало від психічного або поведінкового розладу, що спричинило втрату 162,5 мільйонів років DALY (disability-adjusted life years), тобто років життя з поправкою на інвалідність [2].

Лікарі всього світу прагнуть знайти наві методи лікування психічних розладів. Технологічний процес не стоїть на місці і вже з початку нового сторіччя стрімко розвивається технологія віртуальної реальності. Її впроваджують в усі сфери діяльності людини, вчені досліджують її ефективність у різних напрямках, таких як армія, промисловість, медицина тощо.

Основні функції, які повинне виконувати програмне забезпечення, що розроблюється це надання можливості ефективної взаємодії лікарів та пацієнтів, впровадження новітніх методик лікування за допомогою технології віртуальної реальності, їх оновлення.

Впровадження новітніх технологій у лікування психічних розладів сприяє зростанню уваги на масштаби захворюваності на психічні розлади людей різного віку, великі компанії приймають участь у розвитку новітніх методів лікування та пришвидшують їх реалізацію.

Метою роботи є дослідження та практична реалізація програмного забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Психічні розлади – це стан здоров'я, що супроводжується змінами емоцій, мислення чи поведінки (або їх комбінації). Вони пов'язані зі стресом або проблемами у соціальній, робочій чи сімейній діяльності. На основі статистики, зібраної на території США можна подивитись на розповсюдженість цих захворювань [3]:

- майже кожен п'ятий (19 відсотків) дорослого населення США страждає від тієї чи іншої форми психічного захворювання;
- кожен 24 (4,1 відсотка) має серйозне психічне захворювання;
- кожен 12 (8,5 відсотка) має діагностований розлад, пов'язаний із вживанням психоактивних речовин.

Психічні захворювання піддаються лікуванню. Переважна більшість людей з психічними захворюваннями продовжують функціонувати в повсякденному житті.

Показниками психічного здоров'я є [3]:

- емоції;
- мислення;
- пам'ять;
- темперамент;
- характер.

Чинники, які руйнують психіку людини [3]:

- шкідливі впливи навколишнього середовища;
- фізичні хвороби;
- генетично запрограмовані моделі поведінки;
- різні залежності та звикання;
- шкідливі звички, такі як алкоголь, куріння, наркотики.

Всі ці фактори здатні у тій чи іншій формі впливати на людей, викликаючи тяжкі хвороби [3].

Існує безліч психічних розладів, що виявляються по-різному. У цілому нині їм характерне поєднання аномальних думок, уявлень, емоцій, поведінкових реакцій з оточуючими [4].

До психічних розладів відносять депресію, біполярний афективний розлад, шизофренію та інші психози, деменцію, розумову відсталість та проблеми з розвитком, такі як аутизм.

Існують ефективні стратегії профілактики психічних розладів, таких як шизофренія. Існують дієві методи лікування та догляду, що дозволяють зменшити страждання хворих.

Ключове значення має доступ до медико-санітарних та соціальних структур, що забезпечують лікування та соціальну підтримку.

Тягар психічних розладів продовжує зростати і помітно впливати на системи охорони здоров'я по всьому світу. Воно спричиняє серйозні наслідки для соціальної сфери, прав людини та економіки [4].

1.1 Обґрунтування необхідності лікування психічних розладів

Психічні та поведінкові розлади широко поширені в усьому світі. Всесвітня організація охорони здоров'я (ВООЗ) стверджує, що кожна четверта людина протягом життя може мати розлад психічного здоров'я [1]. У 2016 році близько 16% населення світу постраждало від психічного або поведінкового розладу, що спричинило втрату 162,5 мільйонів років DALY (disability-adjusted life years), тобто років життя з поправкою на інвалідність [2]. Близько двох третин проблем психічного здоров'я спричинено депресією, тривогою, вживанням наркотиків та розладами, пов'язаними з алкоголем [1]. Незважаючи на високу

поширеність психічних, неврологічних розладів і розладів, пов'язаних із вживанням психоактивних речовин, у багатьох країнах продовжує існувати значний розрив у лікуванні. Понад 60% країн витрачають 1% або менше свого бюджету на психічне здоров'я. Працівники психічного здоров'я становлять лише 1% світової робочої сили охорони здоров'я, а 40% країн мають менше одного ліжка в психіатричній лікарні на 10 000 людей [5]. У таблиці 1.1 можна побачити аналіз індикаторів психічних розладів населення України та деяких інших країн [5].

Таблиця 1.1 – Аналіз індикаторів психічних розладів у різних країнах [5]

Індикатор	Україна	Грузія	Вірменія	Киргизька Республіка
1	2	3	4	5
Населення	42,0	3,7	2.9	6,5
Регіон	Східна Європа	Західна Азія	Західна Азія	Середня Азія
Група доходів	Дохід нижче середнього	Дохід вище середнього	Дохід вище середнього	Дохід нижче середнього
ВВП, поточний, млрд, \$	153,2	15,7	12.4	8,5
ВВП на душу населення, поточний, \$	3649	4764	4643	1323
Очікувана тривалість життя при народженні, років	72	74	75	71
Поточні витрати на охорону здоров'я	177	293	408	79
Тягар психічних розладів (DALY на 100 тис. населення)	3416,26	3508,45	3266,01	1273,51
Кількість лікарів-психіатрів на 100 тис. населення	6,93	6,71	3,84	3,51
Дитячі психіатри на 100 тис. населення	0,76	0,30	0,27	н/д

Продовження таблиці 1.1

1	2	3	4	5
Загальна кількість працівників психічного здоров'я на 100 тис.	8,95	9,34	27,39	27,5
Ліжка для психіатричних лікарень на 100 тис.	65,48	36,72	н/д	31,74
Ліжок у психіатричному відділенні загального стаціонару на 100 тис.	1,92	2,53	1,71	1,75

Київський медичний університет у вересні 2019 року запровадив навчальну програму з розширеним mhGAP-IG. Існуючу навчальну програму з психіатрії та наркології для студентів 4 курсу медицини було переглянуто в кілька етапів, щоб реалізувати mhGAP-IG. Співробітникам кафедри неврології, психіатрії та фізичної реабілітації Київського медичного університету презентували mhGAP-IG та обговорили можливі зміни в навчальній програмі. Внесено необхідні зміни, оновлений навчальний план з психіатрії та наркології був затверджений Науковою радою в червні 2019 року та підписаний проректором Київського медичного університету в серпні 2019 року [6].

Були відібрані такі модулі: деменція, епілепсія, розлади, пов'язані з вживанням психоактивних речовин, психічні та поведінкові розлади у дітей та підлітків. Запровадження mhGAP-IG дозволило скоротити дидактичні заняття, зберігаючи при цьому однакову кількість годин, витрачених на інтерактивні практичні заняття, та наголосивши на самостійному вивченні онлайн-матеріалів mhGAP-IG [6].

Впровадження mhGAP-IG було ефективним як на бакалавраті, так і на рівні післядипломної медичної освіти. Впровадження нової програми для лікування психічних розладів нарешті з'явилося на світ. Через те, що кількість хворих на психічні розлади зростає щороку, необхідні нові альтернативні способи лікування та навчання спеціалістів щодо профілактики та лікування інноваційними методами. Добре структурована інформація та наочні матеріали

були визначені як основні переваги процесу навчання та впровадження mhGAP-IG. Програма була оцінена на основі опитування студентів та мешканців, а також відгуків. У таблиці 1.2 наведено програму по навчанню спеціалістів, у якій структурована інформація про університети, що приймають участь, країни, студенти та вибрані найбільш актуальні сфери психічних захворювань [6].

Таблиця 1.2 – Впровадження програми дій щодо проблем психічного здоров'я в освітніх установах [6]

Країна	університет	Рівень освіти для впровадження mhGAP	Вибрані модулі mhGAP	Цільова аудиторія	Кількість годин	Кількість підготовлених студентів
1	2	3	4	5	6	7
Україна	Київський медичний університет, кафедра неврології, психіатрії та фізичної реабілітації	Бакалавр	Деменція Епілепсія Розлади, пов'язані з вживанням психоактивних речовин. Розлади психіки та поведінки дітей та підлітків	Студенти-медики 4 курсу	10	93
		Аспірантура	Деменція Епілепсія Розлади, пов'язані з вживанням психоактивних речовин. Розлади психіки та поведінки дітей та підлітків	Неврологічні ординатори	10	н/д

Продовження таблиці 1.2

1	2	3	4	5	6	7
Вірменія	Єреванський державний медичний університет, кафедра медичної психології	Бакалавр	Психічні та поведінкові розлади дітей та підлітків Депресія Суїцид	Студенти-медики 2 курсу	8	520
		Аспірантура	Психічні та поведінкові розлади дітей та підлітків Депресія Суїцид	Сімейні лікарі	12	51
Грузія	Тбіліський державний медичний університет	Бакалавр (кафедра)	Психічні та поведінкові розлади дітей та підлітків Деменція Суїцид Інші істотні скарги на психічне здоров'я	Студенти-медики 8 семестр	10	41
		Доктор психіатрії	Психічні та поведінкові розлади дітей та підлітків Психоз Депресія	Студенти-медики 9, 10 семестр	8	116
Киргизька Республіка	Киргизько-російський слов'янський університет	Бакалавр	Депресія Суїцид Розлади, пов'язані з вживанням психоактивних речовин. Розлади психіки дітей	Студенти-медики	18	96

У дослідженні, зображеному на рисунку 1.1, наведено перші національні репрезентативні дані про психічне здоров'я дорослих в Україні. Близько трьох чвертей, які потребують психіатричної допомоги, її не отримували. Основними причинами не звернення за медичною допомогою були неусвідомлення необхідності догляду, самолікування, високі витрати на лікування та ліки та низька якість існуючої допомоги [7].

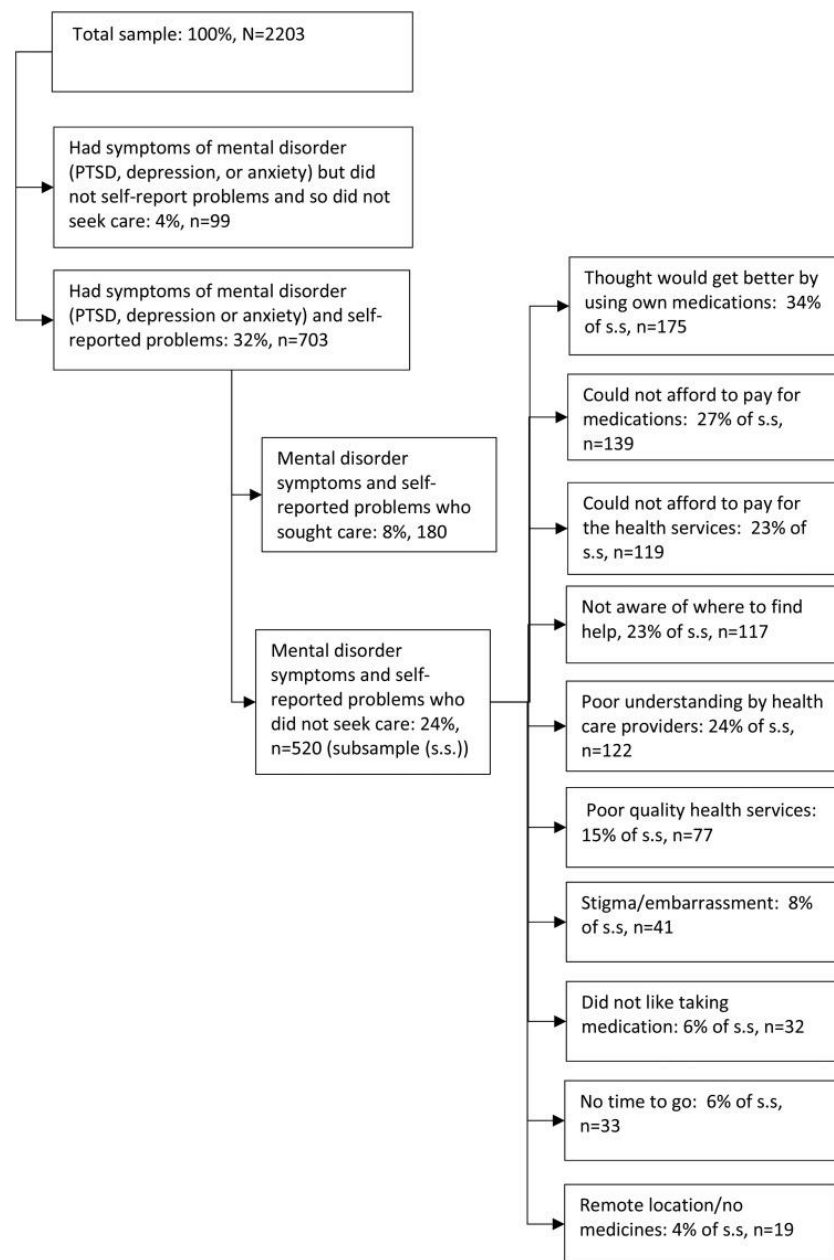


Рисунок 1.1 – Дані про психічне здоров'я дорослих в Україні [7]

Система психічного здоров'я в Україні, як вона функціонує зараз, доволі не реагує на значні потреби населення України в області психічного здоров'я. Ця проблема є не тільки для України але й для багатьох країн світу [8]. Більш широка реформа системи охорони здоров'я, яка нині триває в Україні, є прогресом до системи охорони здоров'я, яка б краще відповідала потребам свого населення. Більше того, прихильність Міністерства охорони здоров'я до реформи психічного здоров'я як ключового компонента цих ширших реформ створює основу для переходу до більшого надання послуг у громаді. Окреслення бачення реформи психічного здоров'я має бути пріоритетним і доповнено чіткою дорожньою картою для її впровадження. Для відстеження прогресу необхідні ключові показники та часові рамки звітності [9].

Ці дослідження відображають той факт, що для надання медичної допомоги хворим на психічні розлади в Україні необхідні масштабні, новітні, комплексні та обґрунтовані системи.

1.1.1 Ризики захворювання психічними розладами

Чинниками психічного здоров'я та психічних розладів є не тільки індивідуальні якості, такі як здатність керувати власними думками, емоціями, поведінкою та спілкуванням з оточуючими, а й соціальні, культурні, політичні та екологічні фактори, такі як національна політика, соціальний захист, рівень життя, умови роботи та підтримка оточуючих. Діаграму розповсюдження психічних захворювань можна побачити на рисунку 1.2 [10].

Стрес, спадковість, харчування, перинатальні інфекції та несприятливий вплив довкілля також можуть викликати психічні розлади.

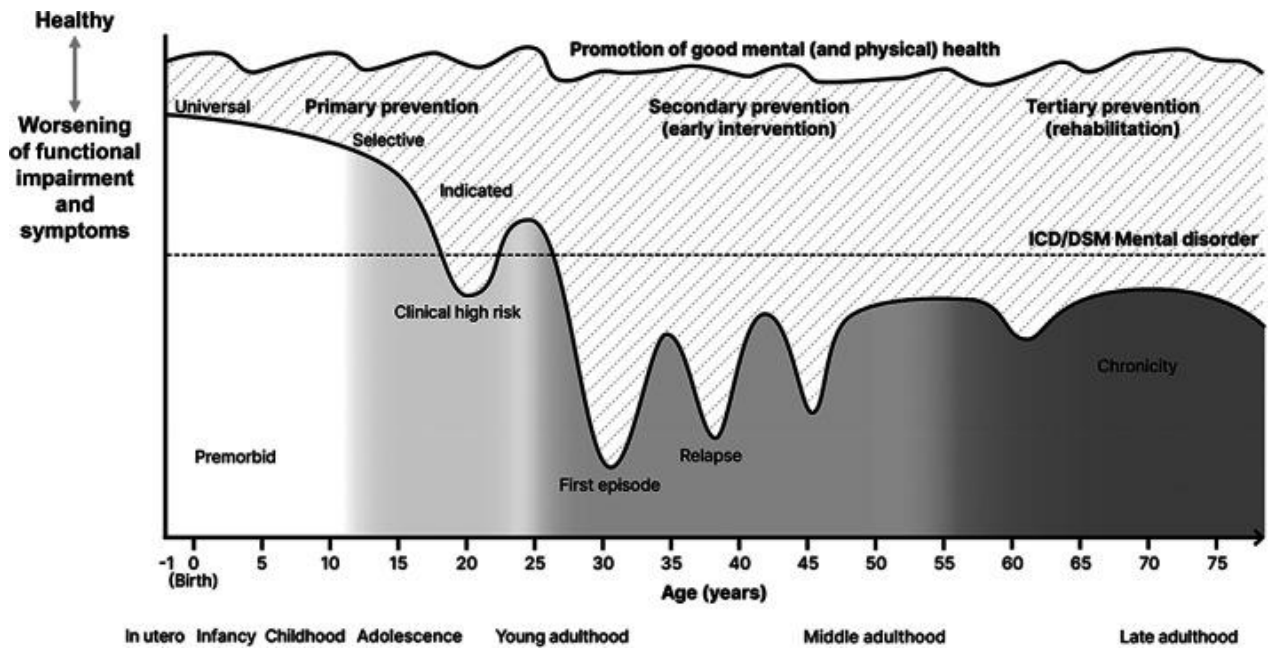


Рисунок 1.2 – Діаграма ризиків захворювання психічними розладами відносно віку людини [10]

1.1.2 Охорона здоров'я та діяльність ВООЗ

Системи охорони здоров'я недостатньо адекватно реагують на проблему психічних розладів. В результаті в усьому світі спостерігається великий розрив між потребами в лікуванні та допомогою. У країнах із низьким та середнім рівнем доходу від 76% до 85% пацієнтів із психічними розладами не отримують жодного лікування [11].

Проблема посилюється ще й тим, що пропонований догляд найчастіше виявляється недостатньо якісним.

Крім структур охорони здоров'я, допомогу та підтримку особам із психічними розладами мають надавати оточуючі. Таким людям часто потрібна допомога в отриманні доступу до програм навчання, адаптованих до їхніх потреб, пошуку роботи та житла, щоб вони могли брати активну участь у громадському житті.

У Плані дій ВООЗ із психічного здоров'я на 2013–2020 рр., схваленому Всесвітньою асамблеєю охорони здоров'я у 2013 році, психічному здоров'ю відводиться важлива роль у забезпеченні кращого здоров'я для всіх. План включає чотири основні цілі [11]:

- підвищення ефективності управління та керівництва у галузі психічного здоров'я;
- надання всеосяжних, комплексних послуг з лікування та соціального захисту пацієнтів із психічними розладами;
- реалізація стратегій підвищення ефективності та профілактики і зміцнення інформаційних систем,
- збору даних та наукових досліджень.

Програма дій ВООЗ з ліквідації прогалин у галузі психічного здоров'я, реалізація якої розпочалася у 2008 році, використовує засновані на фактичних даних технічні посібники, інструменти та навчальні матеріали для підвищення якості послуг, що надаються у країнах, особливо в умовах нестачі ресурсів. В основі її роботи лежить комплекс пріоритизованих умов, а зусилля щодо зміцнення потенціалу спрямовані насамперед на неспеціалізовані медичні установи. Застосовується комплексний підхід, що підвищує пріоритет психічного здоров'я всіх рівнях медичних послуг [12].

Отже, можна зробити висновок, що психічні розлади є розповсюдженим явищем та потребує впровадження новітніх систем профілактики та лікування хворих.

1.2 Аналіз технології віртуальної реальності

Віртуальна реальність (VR) – це створене комп'ютером середовище зі сценами та об'єктами, які здаються реальними, завдяки чому користувач

відчуває, що він занурений у своє оточення. Це середовище сприймається через пристрій, відомий як гарнітура або шолом віртуальної реальності. VR дозволяє нам зануритися у відеоігри, ніби ми є одним із персонажів, навчитися виконувати операції на серці або покращити якість спортивних тренувань, щоб максимізувати продуктивність тощо [13].

Сьогодні технологія віртуальної реальності використовується у таких галузях як медицина, культура, освіта та архітектура – це деякі сфери, які вже скористалися цією технологією, докладніше розглянуто у таблиці 1.3. Від керованих відвідувань VR музеїв до розтину м'язів – віртуальна реальність дозволяє нам перетинати межі, які інакше було б неможливо уявити [14] - [22].

Таблиця 1.3 – Інноваційне використання віртуальної реальності

Сфера використання	Опис використання
1	2
Продуктова промисловість	Віртуальні магазини та ресторани [14]
Ліки	Іспанській національній дослідницькій раді вдалося зменшити наслідки хвороби Паркінсона у кількох пацієнтів, застосувавши лікування, яке використовує VR [15]
ЗМІ	Імерсивна журналістика переносить користувача в місця, де відбувалися події, за допомогою прямої трансляції 360° відео [16]
Освіта	У класах використання VR дозволяє студентам краще зберігати знання та допомагає учням із труднощами у навчанні [17]
Розваги	Користувачі можуть входити в сцену у відеоігри або займатися екстремальними видами спорту, не встаючи з дивана [18]

Продовження таблиці 1.3

1	2
Архітектура	VR допомагає архітекторам краще передбачити простір і представити проєкт своїм клієнтам [19]
Промисловість	Цифрові близнюки – це точні цифрові копії фізичних об'єктів, на яких працівники фабрики можуть практикувати та тестувати їх у віртуальному світі [20]
Культура	Деякі музеї та галереї пропонують віртуальні відвідування або захоплюючі враження, щоб допомогти зрозуміти історію та культуру, пов'язані з кожною роботою [21]
Військові	Міністерство оборони Великобританії використовує віртуальну реальність для навчання в імітованих бойових умовах [22]

Найбільш впізнаваним компонентом технології віртуальної реальності є дисплей на голові у якості окулярів або систем, схожих на шолом. Технологія відображення часто є найбільшою відмінністю між системами віртуальної реальності, у які повністю занурюються і традиційними інтерфейсами користувача. Наприклад, автоматичні віртуальні середовища CAVE [23] активно відображають віртуальний вміст на екранах розміром з кімнату.

З безліччю нових апаратних і програмних опцій, майбутнє пристроїв для відтворення віртуальної реальності розвивається, але поки невідоме. Такі концепції, як HTC Vive Pro Eye [24], Oculus Rift [25] і Playstation VR [26] (рис. 1.3-1.5), лідирують, але є також такі гравці, як Google, Apple, Samsung, Lenovo та інші, які можуть здивувати галузь новим рівнем занурення та зручності використання [23].



Рисунок 1.3 – Шолом віртуальної реальності HTC VIVE [24]



Рисунок 1.4 – Шолом віртуальної реальності Oculus Rift [25]



Рисунок 1.5 – Шолом віртуальної реальності PlayStation VR [26]

Станом на кінець 2018 року трьома найбільш продаваними гарнітурами віртуальної реальності були Sony PlayStation VR, Oculus Rift від Facebook та HTC Vive. У 2019 році сфера віртуальної реальності розширюється, коли Google, HP, Lenovo та інші бажають захопити частину ринку, що все ще розвивається.

Існує другий клас віртуальної реальності, який насправді є просто оболонкою зі спеціальним об'єктивом, який поєднується зі смартфоном, щоб забезпечити занурення у віртуальний світ. Ці пристрої можна продавати за низькою ціною через низьку вартість матеріалів для виготовлення. Вони забезпечують зменшений досвід занурення у віртуальну реальність, аніж створеного набагато дорожчим обладнанням [27].

Для людей з обмеженими можливостями технологія віртуальної реальності дає змогу відчувати ті емоції та побувати там, де вони не спроможні бути через своє тяжке фізичне становище. Це допомагає психологічно поступово реабілітуватися після важких травм [28].

Таким чином, вже зараз технологія віртуальної реальності є затребованою сферою майже в усіх галузях життя людини. Вона допомагає максимально зануритися у віртуальний світ да дає змогу людям побувати у важкодоступних або нереальних місцях в силу їх місця проживання або фізичного становища. Також допомагає у проектуванні складних систем, навчанні студентів, тестуванні складних фізичних об'єктів тощо.

1.3 Аналіз ефективності застосування VR технології

Технологічний розвиток дає змогу все більше і більше можливостей для впровадження віртуальної терапії у різні сфери психічних захворювань. Доказом цього є виборка оглядів з різних аналізів та систематичних оглядів, опублікованих за останні п'ять років, пов'язаних з використанням віртуальної реальності в

діагностиці та лікуванні розладів психічного здоров'я [29]. У таблиці 1.4 наведено висновки з цих статей, кількість досліджень та типи впроваджених оглядів, які застосовувались під час досліджень та посилання на статті з дослідженнями [30] - [36].

Таблиця 1.4 – Огляди та метааналізи з використанням віртуальної реальності для лікування психічних розладів

Тип огляду	Стаття	Включені дослідження	Висновки (зі статей)
1	2	3	4
Систематичний огляд	[30]	27 систематичних оглядів та аналізів	«Наявні дані підтверджують використання цієї технології для лікування тривожних розладів, знеболення, ожиріння та розладів харчування, а також розладів, пов'язаних зі стресом. Але все одно немає чітких доказів хорошої якості за чи проти використання VR для лікування депресії та шизофренії».
Систематичний огляд (психічне здоров'я)	[31]	285 досліджень	«Середовище віртуальної реальності може притупляти психіатричні симптоми, маніпуляції з віртуальною реальністю можуть допомогти зрозуміти розлади, а простіші психологічні методи лікування можна успішно застосовувати у VR. Найвідоміший висновок полягає в тому, що лікування, засноване на технології віртуальної реальності, може зменшити тривожні розлади, але є багато перспективних шляхів дослідження та лікування».
Систематичний огляд (розлади харчування та ваги)	[32]	3 дослідження	«Спостереження за трьома різними людьми з розладами харчування під наглядом лікарів протягом 1 року показали, що VR для розладів харчування та ваги має вищу ефективність, ніж золотий стандарт у цій галузі, тобто когнітивно-поведінкова терапія».

Продовження таблиці 1.4

1	2	3	4
Систематичний огляд (терапія психічного здоров'я)	[33]	Не повідомляється	«Потрібні додаткові дослідження, перш ніж VR терапію можна вважати стандартом надання допомоги в деяких областях, однак для пацієнтів із посттравматичним стресовим розладом або тривогою, особливо для пацієнтів, які не реагують або не бажають брати участь у традиційній терапії, використання VR терапії може розглядатися як альтернатива. Використання віртуальної реальності для інших станів, таких як хронічний біль, реабілітація та залежності, також є клінічною перспективою».
Систематичний огляд (оцінка психічного здоров'я)	[34]	39 досліджень	«Майже всі досліджені середовища віртуальної реальності були здатні одночасно провокувати та вимірювати психіатричні симптоми. Крім того, у 14 дослідженнях було виявлено значну кореляцію між заходами VR та традиційними діагностичними заходами. ».
Систематичний огляд (психічне здоров'я)	[35]	82 дослідження	«Наш експеримент продемонстрував, що віртуальна реальність ефективна у провокуванні реалістичних реакцій на подразники, яких бояться піддослідні, особливо на тривогу; більше того, було доведено, що природа VR ідеально підходить для лікування болю. Однак, відсутність досліджень, пов'язаних із депресією та стресом, підкреслює прогалини в літературі, які все ще існують».
Систематичні огляди (фобії)	[36]	11 досліджень	«Застосування VR терапії стали ефективною альтернативою, яка може зрівнятися з результатами традиційного лікування фобій з точки зору ефективності. Однак вони також є інструментами, здатними розширити поле психологічного лікування».

Продовження таблиці 1.4

1	2	3	4
Систематичний огляд (клінічна медицина)	[37]	11 досліджень	«Дані 11 відповідних досліджень дають уявлення про три поточні медичні застосування технології VR: відволікання болю, розлади харчування та когнітивна/моторна реабілітація. Загалом, більшість досліджень останнього десятиліття показали, що віртуальна реальність ефективна, проста у використанні, безпечна та сприяє високому задоволенню пацієнтів».
Систематичний огляд і аналіз (процедурний біль)	[38]	20 досліджень	«VR може брати участь у гостро болючих процедурах. Потрібні подальші дослідження для підтвердження результатів, встановлення економічної ефективності та оптимальних клінічних параметрів для використання. Про майбутні випробування слід повідомляти відповідно до встановлених рекомендацій».
Систематичний огляд (клінічна медицина)	[39]	Не повідомляється	«VR показав свою ефективність у зменшенні болю, викликаного опіками, і в управлінні болем в інших ситуаціях. VR терапія стала ефективним вибором для пацієнтів із тривожними розладами.

У кожній терапії мають бути етапи лікування. Розглянувши статтю про відновлення особистості під час посттравматичного синдрому можна виділити три основні етапи лікування з використанням технології віртуальної реальності для постраждалих які показані у таблиці 1.5. На основі цієї таблиці можна виявити патерн лікування захворювань. Ці концепції також можуть бути використані для поліпшення якості життя не тільки для тих, хто звертається за терапевтичним лікуванням, а й для тих, хто в цілому прагне до стійкості, самоактуалізації і більш здорового образу життя.

У яких випадках VR технологія підходить, а коли непридатна для клінічних випадків лікування є важливим завданням, яке необхідно вирішити, перш ніж вона стане досить складною для використання на практиці. Якість впливу і вартість його дизайну, а також технічна грамотність і здатність працювати з обладнанням – це інші аспекти специфічних для віртуальної

реальності проблем, які необхідно вирішити, перш ніж застосовувати такі методи лікування [40].

Таблиця 1.5 – Етапи терапії з використанням технології віртуальної реальності [40]

Глибина досвіду	Вміст експозиції у VR	Засоби, цілі та очікувані результати
0 рівень (експозиція)	Повторне переживання події	Визнання почуття безпорадності, страху і болю, а також гнів, якщо він присутній або виявлений
1-й рівень (рішення)	Віртуальне зображення подразника у іншій обстановці	Пояснення як він чи вона відчував жах, пошук розуміння і, якщо можливо, пробачення вчинку
2-й рівень (різний досвід розповіді)	Віртуальне зображення подразників у іншій та безпечнішій обстановці	Внутрішня боротьба, коли подразники намагаються напасти, або вдається втекти і опинитися у приємному і розслаблюючому стані або вдається врятувати іншу людину з подібної ситуації

Ще більше заглибившись в аспекти небезпеки, які може представляти віртуальна реальність, можна обговорити результати, які може мати повний віртуальний курс реабілітації. Наслідки постійного використання VR [41] можуть привести до появи абсолютно нової галузі досліджень і необхідності в нових керівних принципах.

VR надає можливості вийти за рамки традиційних втручань і дозволяє адаптувати підходи до кожної людини, тим самим, можливо, підвищуючи ефективність і зберігаючи навички. При різній якості доказів аналітична література пропонує позитивні результати лікування психічних захворювань за допомогою віртуальної реальності, в основному розладів, пов'язаних з тривогою [62]. VR є реальністю в поведінковому здоров'ї, можливість використання його

для моделювання зовнішнього і внутрішнього світу може відкрити нові клінічні варіанти в найближчому майбутньому, здатні безпосередньо орієнтуватися на досвід тіла та пов'язані з ним процеси. Ця перспектива надасть радикально нове значення класичній терапії, використовуватимуть передові мультисенсорні технології для зміни фізіологічних процесів для покращення благополуччя [43].

Доступні докази показують, що терапія у складі якої використовується технологія віртуальної реальності може бути застосована для полегшення переживання гострого болю у різних стаціонарних налаштуваннях. Однак його наслідки, ймовірно, зміняться залежно від пацієнта. Цей факт визначає необхідність індивідуалізованого пілотного тестування ефектів терапії віртуальної реальності для кожного конкретного випадку клінічного застосування, а не узагальнення його використання для широкого вказівки щодо сприяння аналгезії.

1.4 Огляд особливостей платформи Facebook Meta

Творець Facebook Марк Цукерберг оголосив про створення Meta всесвіту на заході компанії Connect, орієнтованому на AR/VR, поділившись тим, що нова назва відобразила новий напрям розвитку VR технології– створення метавсесвіту [44].

У серпні 2021 року Facebook організував велику пресу навколо програми VR, призначеної для того, щоб люди могли проводити зустрічі у VR. У вересні у блозі під назвою «Відповідально створюємо метавсесвіт» Facebook оголосив про фонд у розмірі 50 мільйонів доларів, призначений для інвестицій у дослідження, «щоб забезпечити відповідальну розробку цих продуктів». У серпні 2021 року Facebook оголосила про менший фонд розробників у розмірі 10 мільйонів доларів для розробників на своїй платформі Horizon Worlds, а також детально розповіла,

що планує найняти 10 000 співробітників в Європі спеціально для створення своєї платформи метавсесвіту [45].

На початку жовтня 2021 року кілька академічних дослідницьких команд отримують доступ до нового інструменту від Facebook, призначеного для зведення майже універсальних даних у режимі реального часу у найбільшій у світі соціальній мережі.

TechCrunch переглянув новий API академічних досліджень Facebook і поспілкувався з менеджером з продуктів Facebook Кіраном Джагадішем, який очолив проєкт разом із командою Facebook Open Research & Transparency (FORT) [46].

За допомогою інструменту дослідники можуть аналізувати великі фрагменти необробленого тексту, використовуючи такі методи, як аналіз настроїв, який відстежує валентність та емоції, які люди висловлюють у своїй промові на певну тему. Окрім текстових дописів, які містять більшість доступних даних, дослідники також можуть отримати доступ до відповідної інформації, як-от опис груп і сторінок, дати їх створення, а також реакції на публікації.

Команда FORT сподівається тісно співпрацювати з академічними дослідниками для розробки та створення поточних інструментів.

Дослідникам, які завершили процес адаптації команди та погодилися з її політикою конфіденційності, було надано доступ 15 листопада 2021 року. Facebook вимагає від усіх, хто отримує доступ до дослідження, погодитися з обмеженнями конфіденційності, зокрема не ідентифікувати конкретних осіб у даних.

Дослідницький API наразі доступний лише для кількох наукових установ, але команда FORT планує вивчити надання доступу іншим групам, включаючи журналістів. Мета – створити публічну дорожню карту, яка дасть дослідникам і журналістам прозорий погляд на те, над чим працює команда [46].

Метавсесвіт це наступна еволюція соціального зв'язку. Бачення компанії полягає в тому, щоб допомогти оживити метавсесвіт. Віртуальна реальність дозволяє досліджувати нові світи та спільний досвід. Завдяки технології віртуальної реальності, навіть якщо ви перебуваєте далеко один від одного, ви можете разом відкривати нові враження від друзів та родини [47].

Були представлені кімнати Horizon – вперше осмислена віддалена співпраця (рис. 1.6) [48].

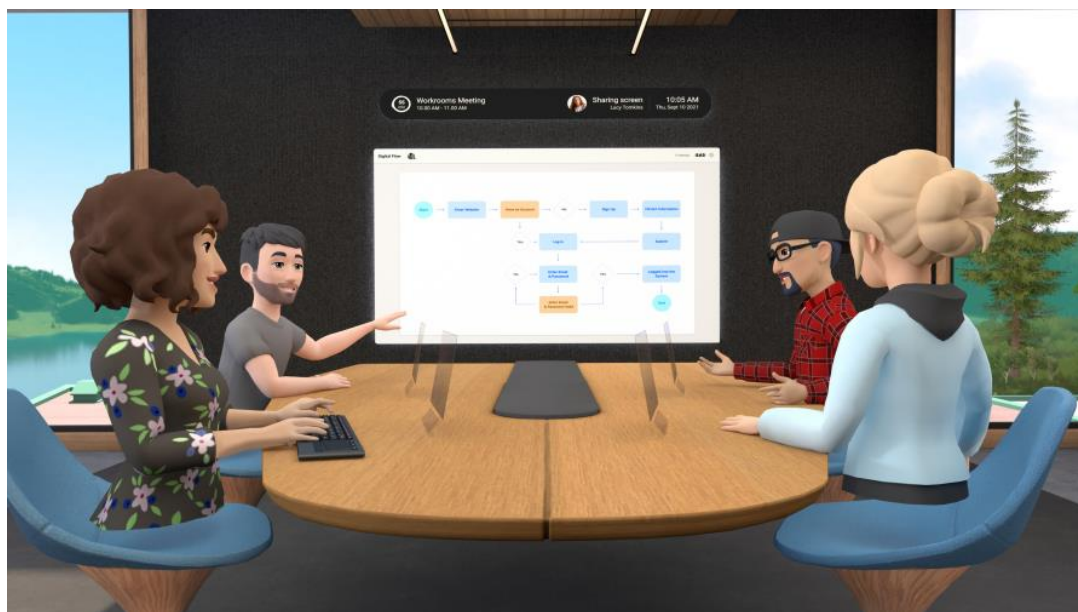


Рисунок 1.6 – Кімната Horizon [48]

Workrooms – це віртуальний простір для зустрічей, де є можливість працювати ефективніше з будь-якого місця. Можна приєднатися до зустрічі у віртуальній реальності як аватар або зателефонувати у віртуальну кімнату зі свого комп'ютера за допомогою відеодзвінка. Існує можливість використовувати віртуальну дошку для мозкового штурму, включити комп'ютер і клавіатуру до віртуальної реальності, щоб працювати разом з іншими, або просто вести виразні бесіди, які більше схожі на те, що ви разом особисто [45].

Робота з віртуальною реальністю не означає, що потрібно використовувати тільки звичайні інструменти. Workrooms – це досвід змішаної реальності, що дає змогу принести свій фізичний стіл і сумісну клавіатуру у віртуальну кімнату, де ви можете побачити, як вони сидять на віртуальному столі для зустрічей. У поєднанні з додатком Oculus Remote Desktop для Mac і Windows можна отримати швидкий доступ одним натиском миші до всього комп'ютера з віртуальною реальністю. Можна робити нотатки під час зустрічей, переносити файли у віртуальну реальність, ділитися своїм екраном з колегами.

Кімната в Workrooms пропонує простір віртуальної дошки для наступного функціоналу [45]:

- створення ескізів у режимі реального часу;
- використання контролеру у якості ручки на письмовому столі перед вами, або біля дошки;
- закріплення зображення зі свого комп'ютера на дошці, з подальшою розміткою для переглядання з іншими користувачами;
- збереження стану дошок у робочих кімнатах, тому користувачі можуть повернутися у будь-який час і продовжити роботу з тієї ж кімнати;
- експорт дошок з віртуальної реальності.

Існує можливість приєднання до віртуальної реальності за допомогою відеодзвінка – користувачі можуть підключитися до кімнати зі свого комп'ютера за допомогою відеодзвінка, або запросити по телефону, просто надіславши посилання на зустріч. Учасники відео з'являться на відеоекрані у віртуальній кімнаті, як у справжньому конференц-залі.

Workrooms – це один з дослідів, який з самого початку був розроблений, щоб використовувати руки, а не контролери, як ваш основний вхід. Це допомагає створити більш природний і виразний соціальний досвід і дозволяє легше перемикатися між фізичними інструментами, такими як клавіатура та контролерами, коли це необхідно [45].

Нотатки зустрічей, спільний доступ до файлів, інтеграція календаря та чат: вебзастосунок Workrooms полегшує співпрацю, незалежно від того, чи перебуваєте користувач у віртуальній реальності чи за комп'ютером. Кожна кімната в Workrooms має місце в Інтернеті, де можна записувати нотатки та дії під час наради, ділитися посиланнями та файлами та спілкуватися зі своєю командою (рис. 1.7) [48].



Рисунок 1.7 – Кімната Horizon з використанням миші та клавіатури [48]

Для використання Workrooms потрібен обліковий запис Workrooms, який є окремим обліковим записом Oculus або Facebook.

Компанія-розробник розраховує на те, що розробники будуть використовувати функціонал Workrooms у своїх власних програмах. Meta продовжує розвивати екосистему віртуальної реальності для роботи.

Віртуальна реальність кардинально змінить спосіб роботи як нова обчислювальна платформа, яка кидає виклик відстані, щоб допомогти людям

краще співпрацювати з будь-якого місця. Horizon Workrooms – це крок до цього бачення.

Meta від компанії Facebook, яка є однією з найбільших компаній у світі, наглядно відображає перспективу створення віртуальних світів у якості віртуальних онлайн-платформ, де люди можуть спілкуватися не тільки трансляціями або аватарами, але й будучи присутніми безпосередньо з співбесідниками. У майбутньому є можливість реалізації приєднання лікувальних платформ до однієї масштабної платформи, такої як Meta, де лікарі зможуть спілкуватися у віртуальній реальності з пацієнтами, що може допомогти у лікуванні.

1.5 Постановка задач магістерської роботи

На основі проведеного аналізу предметної області можна зробити висновок, що психічні розлади є розповсюдженим явищем серед як чоловіків так і жінок різного віку. Технологія віртуальної реальності у якості альтернативного способу лікування або використання під час реабілітаційного періоду є ефективною та перспективною.

В результаті дослідження було прийнято рішення, що буде розроблятися програмне забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності.

Таким чином, розробка програмного забезпечення для лікування психічних розладів є перспективним проєктом створення платформи для лікування психічних розладів.

Метою роботи є дослідження та практична реалізація програмного забезпечення для лікування психічних захворювань за допомогою технології віртуальної реальності.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

- дослідити існуючі технології VR для лікування пацієнтів;
- виконати аналіз вимог до системи;
- розробити архітектуру системи та обрати інструментарій розробки;
- виконати програмну реалізацію VR застосунку;
- розробити керівництво з використання програмного забезпечення.

2 ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ ДЛЯ ЛІКУВАННЯ ПАЦІЄНТІВ

2.1 Огляд технологій віртуальної реальності для лікування психічних розладів

Для лікування психічних розладів за допомогою технології віртуальної реальності використовуються декілька способів реалізації терапій. Основними виступають телетерапія (телепсихологія) та модуляція активності електрокортикалі.

2.1.1 Телетерапія

Телетерапія у телепсихології була створена задовго до появи технології віртуальної реальності. Вона допомагала пацієнтам справлятися з ментальними проблемами такими як побутові негаразди, депресія, апатія, вигорання на робочому місці, фобія тварин тощо. З появою технології віртуальної реальності охорона здоров'я тепер залежить від швидкого вдосконалення комп'ютерних технологій. Наприклад, телепсихологія була успішно впроваджена в систему у справах ветеранів у VR у Сполучених Штатах і віддалено підтримуваних службах боротьби з болем [49]. Розширення ролі технологій для надання психологічних послуг і безперервний розвиток нових технологій відкривають унікальні можливості, міркування та виклики для практики. В опитуванні, проведеному Glueskauf et al. (2018) [49], майже 40% клінічних психологів у США повідомили про використання телепсихології, але в цілому ~60% ніколи не використовували телепсихологію. Вони виявили, що більшість клінічних психологів стурбовані телепсихологією і лікували пацієнтів лише особисто. Понад 50% психологів в опитуванні Glueskauf et al. (2018) були стурбовані тим, що їм потрібно більше

навичок щодо того, як керувати кризовими ситуаціями під час онлайн-терапії, і 79% терапевтів у їхньому дослідженні мали занепокоєння з приводу безпеки, конфіденційності та Закону про переносимість та підзвітність медичного страхування 1996 року (HIPAA) [49].

Під час телетерапії лікарі з пацієнтами зазвичай користуються сучасними месенджерами. Популярність месенджерів, які використовуються зображені на рисунку 2.1. Під час пандемії кількість психічних захворювань зросло. Саме тому зріз попит на психологів та методику телепсихології зокрема. Рівень зростання захворюваності зображений на рисунку 2.2 [49].

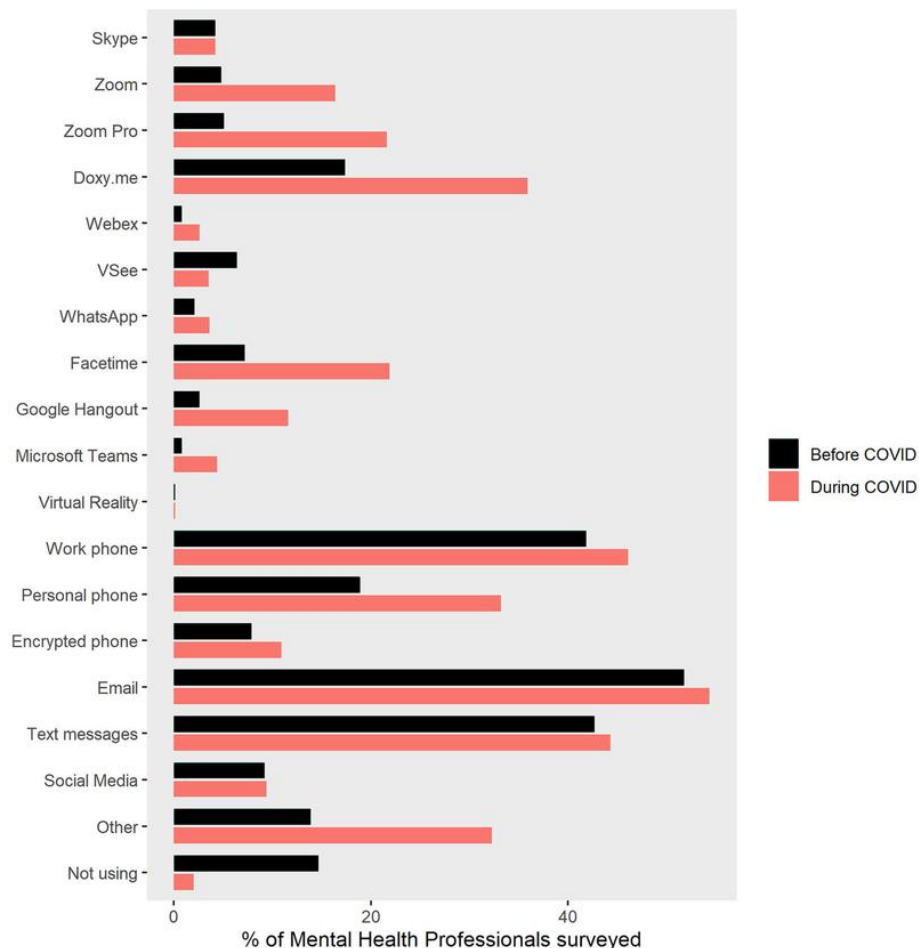


Рисунок 2.1 – Найчастіші види телекомунікацій, які використовуються під час телетерапії [49]

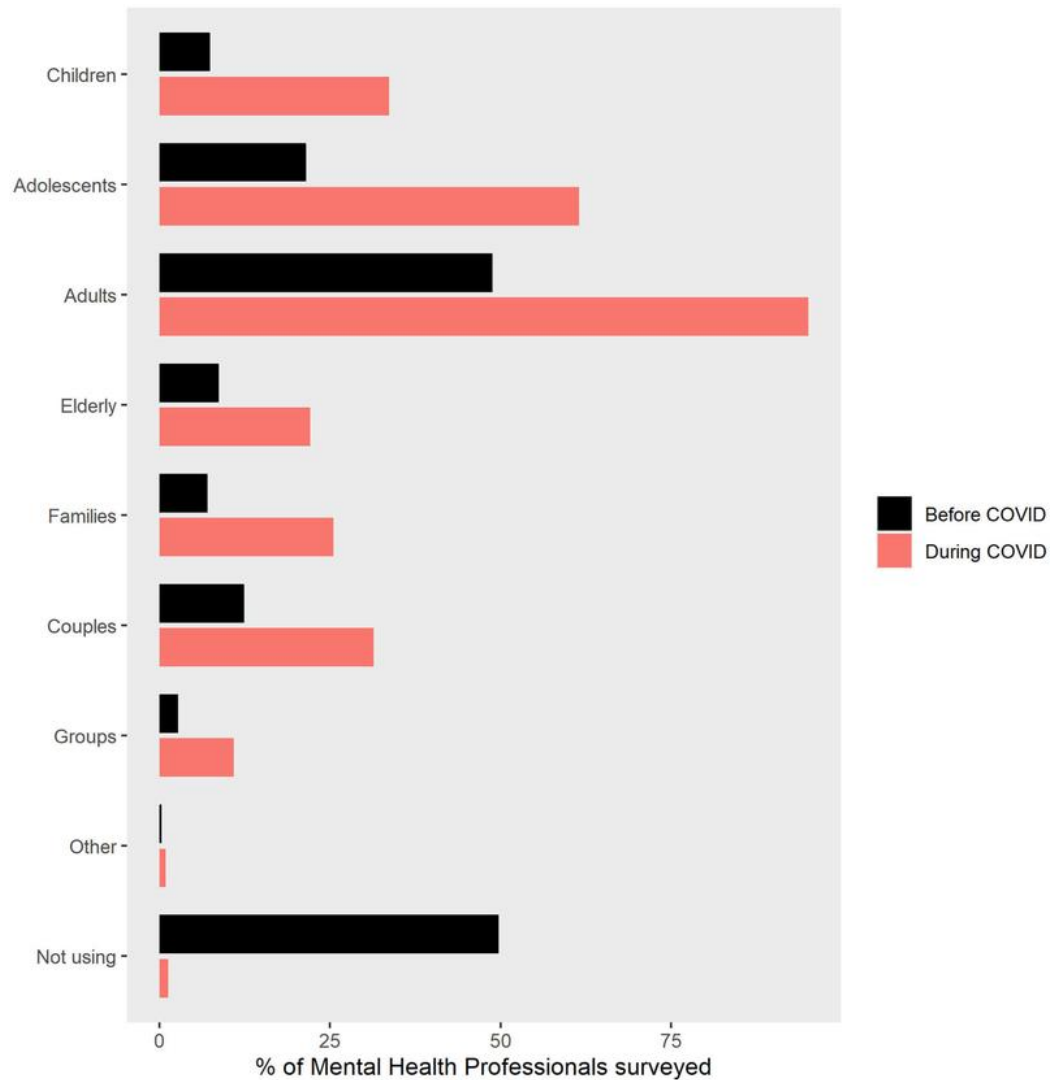


Рисунок 2.2 – Відсоткове збільшення телепсихології під час пандемії в усіх основних категоріях професій [49]

Методика складається з кількох етапів [50]:

- перевірити медичну карту пацієнта;
- переглянути процес телепсихології разом із пацієнтом та отримайте записану згоду або електронною поштою;
- підключити до системи;
- побачити пацієнта через zoom, facetime, skype, google duo або по телефону (потрібен офіційний дозвіл від постачальників телепсихології);

- зателефонувати пацієнту за 20–30 хвилин до прийому та підтвердити протоколи телепсихології;
- підтвердити, що ви розмовляєте з потрібною людиною, перш ніж почати, і особливо перед тим, як роздавати будь-яку інформацію;
- доглядальники можуть бути присутніми, але має бути відомо, хто на сеансах телепсихології;
- застосувати відповідне психологічне втручання;
- створити єдине візуальне враження – гарний зовнішній вигляд;
- дослідити психологічні ідеї та занепокоєння пацієнта, головне питання, яке потрібно прояснити, і мету цього сеансу телепсихології, а також узгодити список проблем;
- застосовувати належні психологічні оцінки, наприклад, щодо тривоги, депресії, соціальних проблем, духовних проблем, самотності тощо;
- підсумувати результати, визначивши ключові результати;
- повідомити пацієнта та батьків/опікунів, що наступний прийом буде надіслано службою пацієнтів на їхній мобільний;
- подальше повідомлення зі стандартизованим, легко читаним листом підказок може бути надіслано пацієнту електронною поштою, щоб повторити психологічні інструкції, обговорені через служби телепсихології;
- підготувати документацію. Рекомендується використовувати мову віртуального відвідування.

2.1.2 Підвищення електрокортикальної активності

Віртуальна реальність надає можливість створювати середовище для лікування, але рухові тренування за допомогою віртуальної реальності не можуть покращити рухові характеристики так сильно, як рухові тренування в реальних

умовах. Перевага занурення у віртуальну реальність полягає в тому, що вона може вносити тимчасові візуальні збурення поверх візуального середовища. Тренування за допомогою технології віртуальної реальності зі збуреннями підвищують електrokортикальну активність та покращують показники рівноваги в порівнянні з тренуваннями за допомогою технології віртуальної реальності без збурень. Короткі зорові збурення покращили показники балансу під час тестування (рис. 2.3) [51].

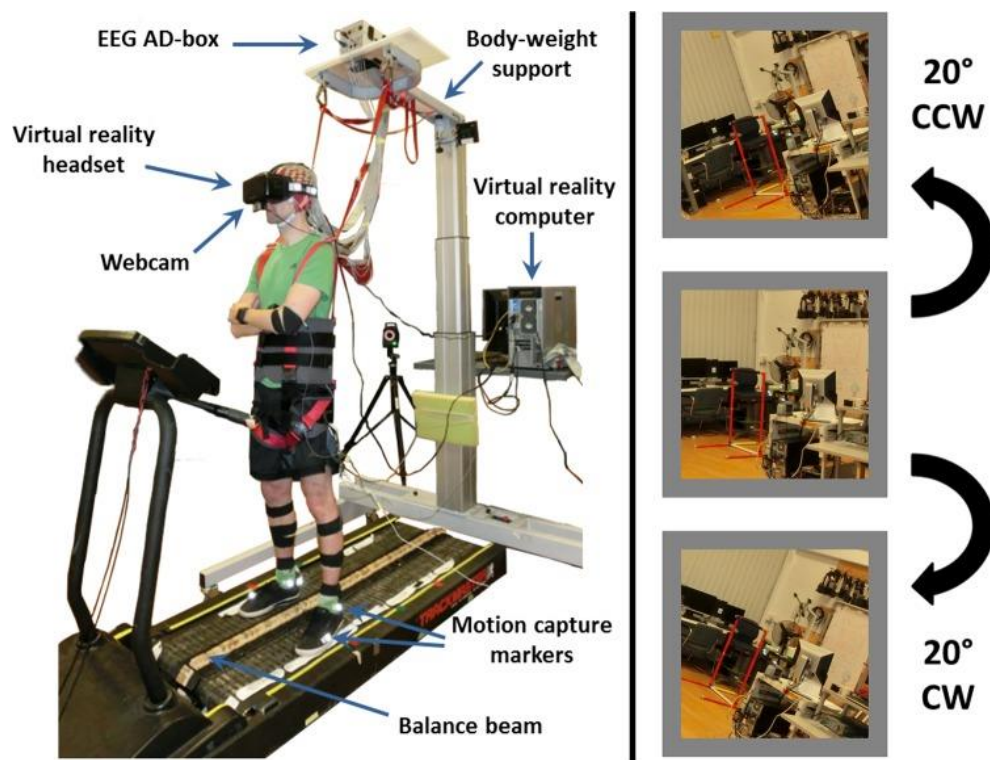


Рисунок 2.3 – Апаратне забезпечення для модуляції електrokортикальної активності [51]

Поліпшення моторної продуктивності під час тренування віртуальної реальності без збурень було набагато нижчим, ніж покращення під час тренування віртуальної реальності з збуреннями. Спостерігається зниження короткочасного рухового навчання, яке відбувається під час використання

віртуальної реальності в порівнянні з невикористанням технології віртуальної реальності. Одним із факторів, що сприяє, може бути поле зору.

Локалізація джерела ЕЕГ (електроенцефалограма) показана синім кольором для тренування віртуальної реальності з збуреннями, червоним для тренування віртуальної реальності без тренування збурень і помаранчевим для незмінного вигляду (рис. 2.4) [51].

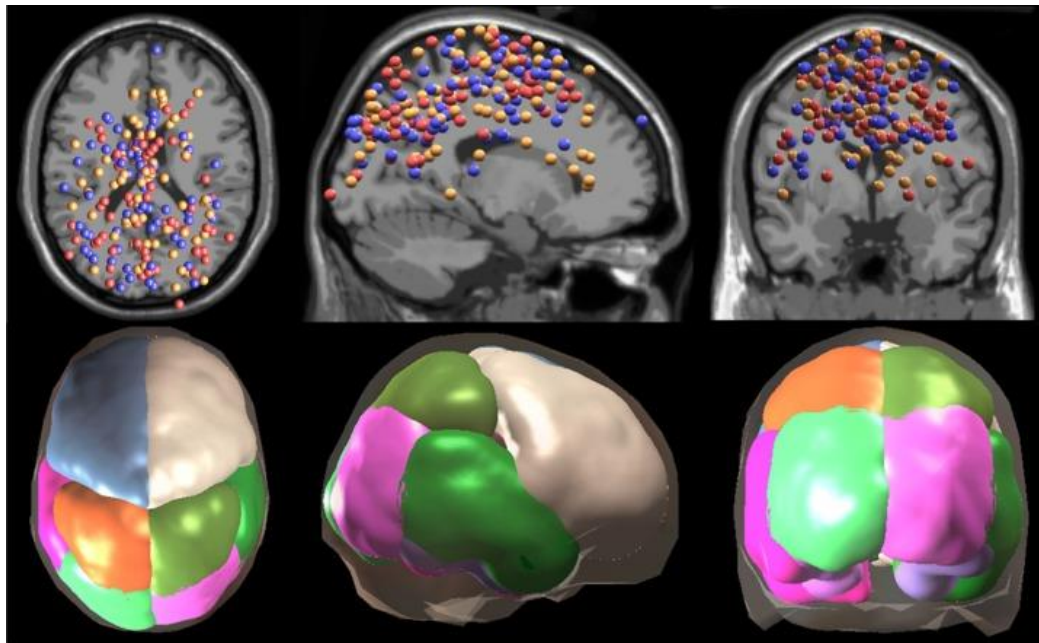


Рисунок 2.4 – Збудження частин головного мозгу під час терапії [51]

Була використана новітня технологія збурення візуального обертання, щоб подолати негативний вплив віртуальної реальності під час завдання балансу при ходьбі. Ці збурення викликали чіткі електрокортикальні реакції в тім'яній, потиличній та поясній областях через конфлікти між сенсорним введенням під час рівноваги. Різниця спектрів потужності між групами тренінгів віртуальної реальності показала збільшення тета-потужності та зниження альфа-потужності під час тренування з збуреннями порівняно з віртуальним тренуванням без збурень у кількох областях кори, що вказує на те, що збурення викликали зміну

активності кори. Виявлені нами покращення при ходьбі на променях не можуть бути пояснені як ефект адаптації, що свідчить про короточасні поліпшення рухового навчання в контролі рівноваги [51].

2.2 Аналіз існуючого програмного забезпечення для порівняння алгоритмів лікування психічних розладів

2.2.1 Постінсультні психічні розлади

Віртуальна реальність та інтерактивні відеоігри з'явилися як новітні підходи до лікування в реабілітації після інсульту, зокрема за допомогою комерційних ігрових консолей, які швидко впроваджуються в клінічних умовах. Це оновлення Кокранівського огляду, опублікованого спочатку в 2011 році, а потім знову в 2015 році [52].

Було зроблено 72 випробування, в яких взяли участь 2470 учасників. Цей огляд включає 35 нових досліджень на додаток до досліджень, включених у попередню версію цього огляду. Розміри вибірки в дослідженні, як правило, були невеликими, а втручання відрізнялися як з точки зору цілей лікування, так і використовуваних пристроїв віртуальної реальності. Ризик упередженості, присутній у багатьох дослідженнях, був неясним через погану звітність.

Знайдені докази того, що використання віртуальної реальності та інтерактивних відеоігор не було більш корисним, ніж звичайні підходи до терапії для покращення функції верхніх кінцівок за рахунок збудження коротикальних волокон мозку. Віртуальна реальність може бути корисною для покращення функції верхніх кінцівок і повсякденної діяльності, якщо використовується як доповнення до звичайного догляду (для збільшення загального часу терапії). Не було достатньо доказів, щоб зробити висновки про вплив віртуальної реальності

та інтерактивних відеоігор на швидкість ходи, рівновагу, участь або якість життя. Цей огляд показав, що час від початку інсульту, тяжкість ураження та тип пристрою (комерційного чи спеціального) не мали сильного впливу на результат. Існувала тенденція, що припускала, що більша доза (більше 15 годин загального втручання) була кращою, як і спеціальні програми віртуальної реальності. Під час випробувань виявлено, що віртуальна реальність є безпечним втручанням, яке ефективно для покращення функції рук і повсякденної діяльності після інсульту. Більше поліпшення спостерігалось при вищій дозі, але зв'язок не був статистично значущим. Досягнення виявляється клінічно значущим завдяки аналізам, які показують розумні розміри ефекту (тобто помірний вплив на функцію верхніх кінцівок (стандартизована середня різниця) і незначний або помірний вплив на функцію, (докази середньої якості). Оскільки втручання у віртуальну реальність можуть сильно відрізнятися (від недорогих комерційних ігрових консолей до дорогих індивідуальних програм), незрозуміло, які характеристики втручання є найважливішими. Аналіз випробувань дають чітких вказівок щодо того, які програми віртуальної реальності перевершують інші, оскільки втручання у віртуальну реальність можуть сильно відрізнятися (від недорогих комерційних ігрових консолей до дорогих індивідуальних програм), незрозуміло, які характеристики втручання є найважливішими [52].

Відсутність побічних ефектів, включаючи заколисування, нудоту, головний біль або біль, свідчить про те, що ці фактори не повинні викликати великого занепокоєння клініцистів, однак це може відрізнятися залежно від характеристик людини, апаратного та програмного забезпечення віртуальної реальності та завдання. Клініцисти, які наразі мають доступ до програм віртуальної реальності, повинні бути впевнені, що їх використання в рамках комплексної програми реабілітації виглядає розумним, враховуючи цілі, здібності та уподобання пацієнта [52].

2.2.2 Фантомні болі

Надмірний біль під час медичних процедур є світовою медичною проблемою. Наприклад, запобігання інфікуванню шляхом частого очищення рани має вирішальне значення для виживання після важкої опікової травми. Нажаль, більшість дітей з великими важкими опіками повідомляють про сильний або нестерпний біль під час цих важливих сеансів очищення/протирання опікової рани в резервуарі реанімації, незважаючи на лікування потужними знеболюючими засобами незадовго до лікування рани. Надмірний біль під час медичних процедур збільшує у пацієнтів ризик розвитку довгострокових медичних проблем, таких як постійний або хронічний біль або посттравматичний стресовий розлад (ПТСР). На додаток до травматичних спогадів про первинний нещасний випадок або подію з опіковою травмою (наприклад, пожежа), спогади про сеанси очищення травматичних ран можуть сформувати «гарячі точки» патологічної пам'яті, які потребують майбутньої психотерапії для ПТСР. Зростає актуальність розробки ефективних нефармакологічних анальгетиків [53].

Нинішній короткий огляд досліджує потенціал віртуальної реальності для покращення знеболювання для ручної терапії, щоб зробити фізичну терапію більш веселою та допомогти мотивувати пацієнтів співпрацювати зі своїм терапевтом. Ручна VR терапія має потенціал для широкого кола пацієнтів, які потребують терапії рук, включаючи пацієнтів з гострим болем і потенційно хронічним болем [53].

Відволікання уваги є одним із найпростіших психологічних методів лікування гострого болю. Однак найбільш широко використовуване звичайне відволікання, прослуховування музики під час догляду за раною, є кроком у правильному напрямку, але має обмежену знеболювальну ефективність. Додаткова віртуальна реальність із зануренням стає надзвичайно ефективним високотехнологічним відволіканням, яке виводить відволікання на абсолютно

новий рівень і може призвести до значного клінічно значущого зменшення гострого болю. Під час віртуальної реальності пацієнти дивляться в окуляри віртуальної реальності та взаємодіють з віртуальними об'єктами у створеному комп'ютером світі віртуальної реальності під час болючих медичних процедур. Суть віртуальної реальності занурення полягає в ілюзії користувача «перебувати там» у тривимірному комп'ютерному світі, ніби віртуальний світ – це місце, яке вони відвідують, ілюзія, відома як «присутність» або «відчуття присутності» у віртуальній реальності. Зменшення болю на комп'ютерній томографії зображається як зменшення мозкової активності за допомогою телетерапії. Для наглядності, порівняння терапій зображено на рисунку 2.5 [53].

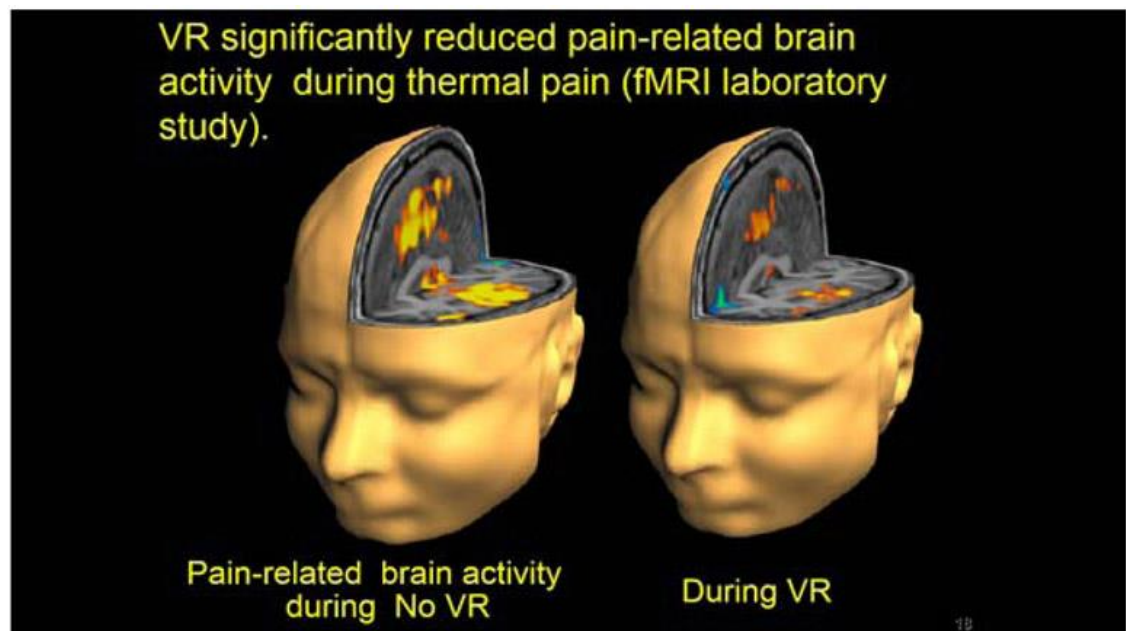


Рисунок 2.5 – Вплив VR технологій на мозок під час терапії [53]

Солдати з бойовими опіками повідомили про значне зниження інтенсивності болю під час VR та значно більше задоволення під час VR. Результати дослідження зображені на рисунку 2.6 [53].

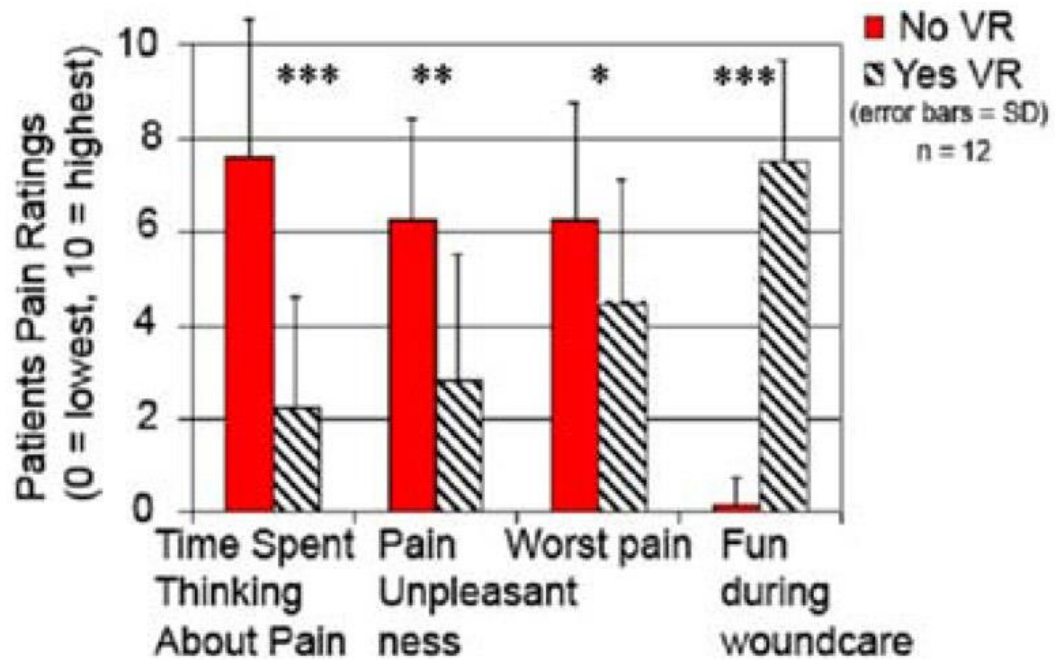


Рисунок 2.6 – Різниця відчуття болю під час звичайної терапії та терапії з використанням технології віртуальної реальності [53]

Технологія віртуальної реальності виявляється ефективною як додатковий нефармакологічний анальгетик для зменшення гострого процедурного болю під час догляду за ранами та фізичної та трудової терапії, включаючи терапію рук. Мільярди доларів інвестуються в масове виробництво продуктів віртуальної реальності та доповненої реальності для продажу широкій публіці (наприклад, в першу чергу для розваг). Технології віртуальної реальності та доповненої реальності, що готові до експлуатації, можуть використовуватися фізичними та трудовими терапевтами, щоб допомогти покращити медичні та психологічні результати пацієнтів, які проходять терапію кінцівок, підвищити рівень задоволеності пацієнтів і таким чином зменшити витрати на охорону здоров'я. У майбутньому VR технології можуть підвищити рівень дотримання пацієнтами виконання домашніх вправ з ручної терапії, зробивши вправи терапії кінцівок менш болючими та веселішими під час домашнього використання. Крім того, VR технології можна використовувати щоб допомогти відстежувати коли і наскільки

добре пацієнти виконують вправи з ручної терапії, а також для кількісної оцінки того, чи збільшує прихильність до лікування довгострокову функціональність.

Використання технології відстеження рук оптичної камери, встановленої на шоломі віртуальної реальності (наприклад, XTAL [55], рис. 2.7), дозволяє пацієнтам, які проводять терапію кінцівок, використовувати свої реальні жести рук для керування жестами кіберруки, що дозволяє пацієнтам взаємодіяти з віртуальними об'єктами у віртуальному світі.



Рисунок 2.7 – Використання технології відстеження рук оптичної камери, встановленої на шоломі віртуальної реальності [53]

В роботі [54] представлено технологію, що дозволяє пацієнтам, які проводять терапію кінцівок, використовувати свої реальні жести рук для керування жестами кіберруки, що дозволяє пацієнтам взаємодіяти з віртуальними об'єктами у віртуальному світі. Тактильно з допомогою звукових хвиль можна додатково підвищити ілюзію знаходження у віртуальному світі.

2.2.3 Фобія тварин

Фобія тварин (тобто комах, змій, птахів або інших тварин) є однією з найпоширеніших форм специфічних фобій, особливо у жінок. Показники цього захворювання протягом життя коливаються від 5 відсотків до понад 12 відсотків, залежно від дослідження, з найвищими показниками поширеності серед молодих людей [56].

Експозиційна терапія під назвою «in vivo» (iVET) є лікуванням вибору для конкретних фобій, включаючи фобію тварин, оскільки було показано, що вона перевершує всі інші форми психосоціального лікування без впливу. Проте інші форми лікування опромінення, такі як терапія опроміненням у віртуальній реальності (VR терапія), були визнані такими ж ефективними, як iVET, особливо щодо заходів після лікування за допомогою телетерапії.

Віртуальна реальність і доповнена реальність – це дві технології, які можна використовувати як альтернативу звичайної терапії. У першому випадку пацієнта просять взаємодіяти з створеним комп'ютером тривимірним середовищем або об'єктом. В останньому віртуальний об'єкт, створений комп'ютером, накладається на реальність. Таким чином, хоча обидві технології мають здатність покращувати або збагачувати досвід знаходження у віртуальному світі, використовуючи штучні, створені комп'ютером елементи, вони відрізняються за своїм призначенням. Зокрема, якщо віртуальна реальність спрямована на транспортування користувача в інше місце, замінюючи існуюче фізичне середовище віртуальним, створюючи таким чином ефект занурення, у доповненій реальності віртуальними елементами є ті, які транспортуються в місце розташування користувача та додаються до світ користувача, що, можливо, призводить до іншої якості досвіду роботи з технологіями віртуальної реальності [57].

Було виявлено, що використання цих альтернативних форм впливу (тобто віртуальної реальності та доповненої реальності) має певні переваги порівняно з традиційними терапіями. Наприклад, деякі клініцисти та пацієнти неохоче використовують iVET, оскільки вважають це жорстоким. Насправді є дані, які свідчать про те, що віртуальна реальність має набагато нижчий рівень відмов, ніж лікування «in vivo». Крім того, VR терапії і терапія доповненою реальністю (AR терапії) пропонують екологічні методи лікування, коли доступність інших способів лікування обмежена [58].

В даний час існує велика кількість доказів ефективності VR терапії та AR терапії у лікуванні широкого кола психічних розладів. Наприклад, зараз їх використання підтримується при посттравматичних стресових розладах та тривожних розладах, включаючи панічний розлад, соціальний тривожний розлад і специфічні фобії.

Таблиця 2.1 показує середні бали учасників для всіх результатів дослідження в різних умовах, а також результати тесту базового дисбалансу. Критерії Колмогорова–Смірнова та Левене виявили, що припущення параметричних даних були порушені (тобто нормальність і однорідність дисперсії) [59].

Таблиця 2.1 – Опис медіанних значень базових балів і медіанних значень балів після лікування в результатах дослідження [59]

Етап лікування	Віртуальна реальність	Доповнена реальність	In vivo
	Mdn	Mdn	Mdn
Дистанційне попереднє лікування	0,64	0,63	0,63
Дистанційне післялікування	-0,77	-0,67	-0,67
Попереднє лікування тривоги	0,37	0,54	0,90
Тривога після лікування	-0,49	-0,19	-0,92

Таким чином, базовий дисбаланс за трьома умовами був досліджений за допомогою тесту Краскала-Уолліса показав, що базові оцінки були задовільними для усіх умов та результатів.

2.3 Висновок до розділу 2

В цілому, технологія віртуальної реальності у якості допоміжного апарату або альтернативи звичайної терапії та профілактики психічних захворювань показала себе достатньо ефективною.

В ході проведених досліджень були проаналізовані типи лікування за допомогою віртуальної реальності, а саме стимуляція кортикальних волокон мозку та телетерапія, необхідні критерії, яких слід дотримуватись під час створення віртуального середовища.

Тому в якості технології для створення програмного забезпечення для лікування психічних розладів було обрано технологію віртуальної реальності, таке рішення було обумовлено рядом причин:

- перспектива технології;
- доведена ефективність;
- можливість використання технології у якості допоміжної до основної терапії.

Алгоритми потребують подальшого розвитку, оскільки для вирішення поставлених задач за допомогою технології віртуальної реальності необхідно удосконалювати взаємодію лікаря та пацієнта, спростити доступ пацієнта до терапії та пришвидшити впровадження нових методик лікування, а також консультації кваліфікованих лікарів та спеціалістів.

3 ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Аналіз вимог до системи

Програмне забезпечення призначене для лікування психічних розладів за допомогою технології віртуальної реальності.

Програмне забезпечення повинне відповідати наступним функціональним вимогам:

- надати засоби для підключення до VR застосунку для лікування;
- надати засоби для взаємодії лікарів та пацієнтів через вебзастосунок;
- надати засоби для оновлення способів лікування пацієнтів;
- отримати дані про пацієнтів та процес їх лікування.

Програмне забезпечення повинне відповідати наступним нефункціональним вимогам:

- працювати у вебзастосунку;
- надавати доступ для авторизованих користувачів;
- надавати зручний та зрозумілий інтерфейс.

Аналіз взаємодії користувачів з програмним забезпеченням, що розроблюється наведено на рисунку 3.1 у вигляді UML-діаграми прецендентів використання.

На UML-діаграмі зображено, що користувачу необхідно авторизуватися у системі перш ніж взаємодіяти з програмним забезпеченням, що розроблюється.

Початкові дані – це інформація, яку пацієнт надає лікарю для отримання рекомендацій щодо лікування психічних розладів за допомогою технології віртуальної реальності. Користувач повинен бути авторизований у системи для того, щоб мати можливість використовувати функціонал системи.

У якості вихідних даних користувач пацієнт отримуватиме можливість взаємодіяти зі сценаріями терапії у віртуальній реальності. Лікар отримуватиме зворотний зв'язок від пацієнта.

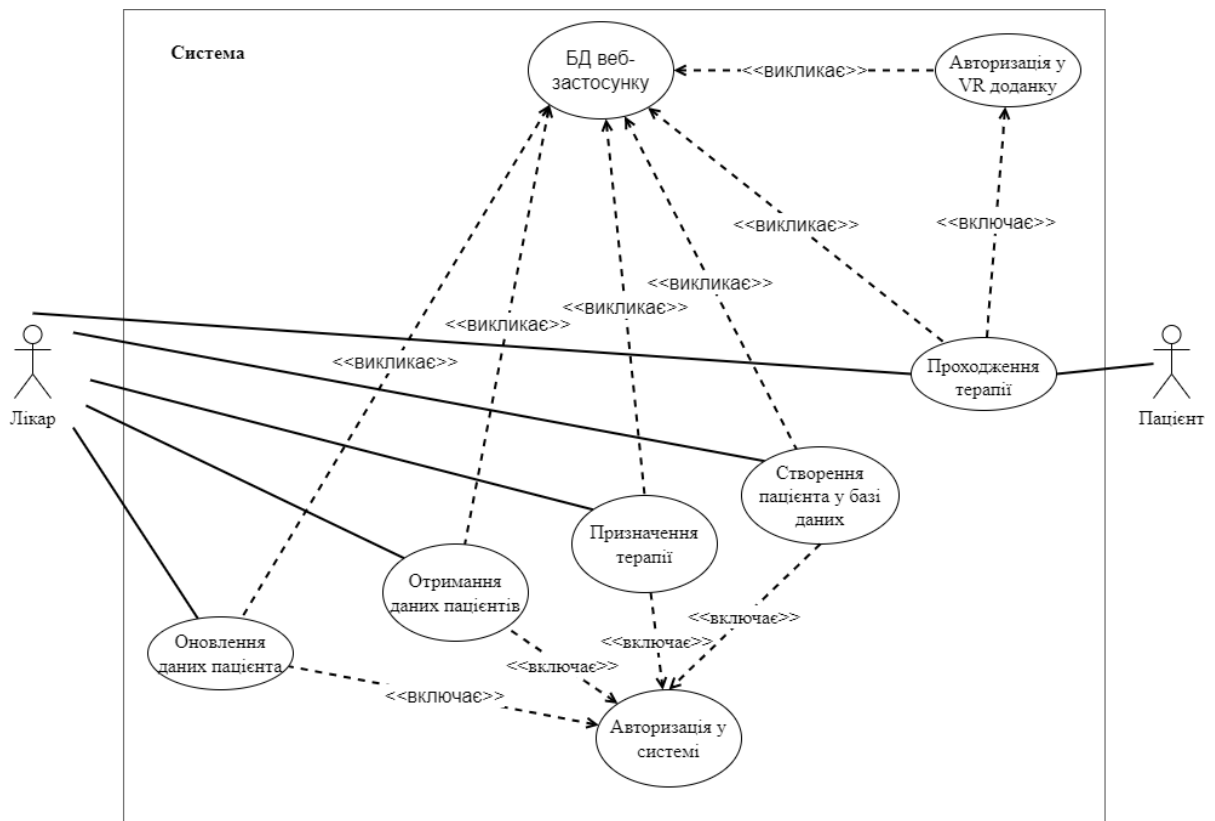


Рисунок 3.1 – Діаграма прецедентів використання

3.2 Вибір методу реалізації проєкту

3.2.1 Proof of Concept

Proof of Concept (PoC) це найкращий підхід для перевірки початкової ідеї та її життєздатності для практичної реалізації. Це загальний термін для надання логічного доказу того, що його насправді можна розробити. Настав час глибше вникнути в тему та зрозуміти, як зменшити ризики, використовуючи доказ концепції [60].

Метою PoC є:

- підтвердити концепцію, яку передбачається розробити;
- дослідити різні рішення, які могли б по суті втілити ідею в життя.

Доказ концепції це демонстрація практичної доцільності ідеї, щоб довести, що метод, ідея чи технологія працюють.

Перевіряються лише критичні функції, і результат не може бути використаний у подальшій розробці. Оскільки мета PoC полягає в тому, щоб переконатися, що концепція працює за короткий час, незначні аспекти можна опустити. У таких випадках, отримавши зелене світло на етапі PoC, команда може почати писати проєкт з нуля.

Підтвердження концепції є необов'язковим етапом проєкту, і в більшості випадків він не є обов'язковим. Але нові можливості пов'язані з:

- певними ризиками, у тому числі загрозами безпеці;
- високою складністю проєктів;
- дефіцитом кваліфікованих кадрів.

Це означає, що перед початком проєкту потрібно переконатися, що рішення працює. А витрати та ризики пропорційні бізнес-результату.

Підтвердження концепції в розробці додатків не є типовим доказом концепції для інших продуктів тому що вони служать різним цілям. Доказ концепції зазвичай вимагається, щоб показати, що ідея має потенціал і користується попитом. Необхідно підтвердити концепцію розробки програмного забезпечення:

- для підтвердження технічних проблем;
- знайти конкретну відповідь на функціональне запитання;
- протестувати нові функції.

PoC в процесі розробки програми є підтвердженням того, що деякі технічні проблеми можна вирішити і яким чином. Наприклад, як продукт можна інтегрувати в існуючу систему підприємства з технічної сторони [60].

3.2.2 Prototype

Модель прототипування – це метод розробки системи, за допомогою якого прототип будується, тестується, а потім переробляється за потреби до тих пір, поки не буде досягнутий прийнятний результат, на основі якого може бути розроблена повна система або продукт. Ця модель найкраще працює в сценаріях, коли не всі вимоги проєкту детально відомі заздалегідь. Це ітераційний процес методом проб і помилок, який відбувається між розробниками та користувачами [61].

Використання моделі-прототипу може принести багато переваг, зокрема:

- клієнти починають висловлювати свою думку щодо продукту, підвищуючи задоволеність клієнтів;
- відсутні функції та помилки легко виявляються;
- прототипи можна повторно використовувати в майбутніх, більш складних проєктах;
- він підкреслює спілкування в команді та гнучкі методи дизайну;
- користувачі краще розуміють, як працює продукт;
- швидший зворотній зв'язок із клієнтами дає краще уявлення про потреби клієнтів.

3.2.3 Minimum Viable Product

Мінімально життєздатний продукт або MVP – це техніка розробки, за допомогою якої на ринок вводиться новий продукт з основними характеристиками, але достатніми, щоб привернути увагу споживачів. Кінцевий продукт виходить на ринок лише після отримання достатнього відгуку від початкових користувачів продукту [62].

Мінімально життєздатний продукт (MVP) – це найбільш скорочена версія продукту, яку ще можна випустити. Демо-версії продуктів, краудфандингові проєкти та цільові сторінки – це звичайні приклади MVP [62].

У деяких випадках MVP може бути частиною інших існуючих інструментів для перевірки його життєздатності, перш ніж він буде розроблений як власний інструмент або програмне забезпечення.

Основні характеристики:

- достатня цінність, щоб люди готові було використати його або придбати спочатку;
- демонстрація достатньої користі у майбутньому, щоб зберегти ранніх користувачів;
- забезпечення циклу зворотного зв'язку, щоб скеровувати майбутній розвиток.

Для порівняння методів реалізації проєкту основні характеристики зібрані у таблиці 3.1.

Таблиця 3.1 – Порівняння методів реалізації проєкту

Критерій	POC	Prototype	MVP
1	2	3	4
Мета створення	Перевірка здійсненності ідеї	Перевірка реалізації та ефективності кількох функцій	Створити життєздатний продукт, доклавши мінімум зусиль
Функції	Можливо одна або декілька функцій	Декілька функцій, які не увійшли до MVP	Основні функції для того, щоб залишатися життєздатним
Аудиторія	Всередині команди	Потенційні інвестори	Групи клієнтів

Продовження таблиці 3.1

1	2	3	4
Подальше використання	Реалізація функції може бути використана у подальшій розробці	Дизайн може бути використаний у подальшій розробці	Перша версія продукту
Цінність	Заділ на майбутнє	Потенціал для потенційних інвесторів	Готовий продукт
Коли розробляється	Коли невідомо, чи можна реалізувати ідею чи окрему функцію	Коли економічне обґрунтування не доведено, ризики невідомі	Коли є фінансування та ризики мінімальні
Необхідні ресурси	Необхідна технічна експертиза для реалізації ідеї	Технічні ресурси майже не потрібні, розробки може не бути	Потрібна технічна експертиза та ресурси для створення продукту
Період розробки	Найкоротший	Середній	Середній

Таким чином, оптимальним рішенням є вибір у якості метода реалізації проєкту саме PoC, оскільки розробка орієнтована на втілення ідеї, висока швидкість розробки та відсутність фінансових ресурсів для реалізації готового продукту та необхідних вузьких кваліфікацій.

3.3 Вибір патерну проєктування

3.3.1 MVC

Шаблон MVC використовується для сучасних вебзастосунків, оскільки він дозволяє програмі бути масштабованою, підтримуватися та легко розширюватися [63].

Шаблон MVC допомагає розбити код на окремі компоненти. Таким чином, набагато легше керувати та вносити зміни до будь-якої сторони, не заважаючи одна одній. Складається з трьох основних частин, а саме з моделі, представлення та контролера [63].

Завдання моделі – просто керувати даними. Незалежно від того, чи надходять дані з бази даних, API чи об'єкта JSON, модель відповідає за керування ними.

Завдання представлення – вирішувати, що користувач побачить на своєму екрані і як.

Сутність контролера полягає в отриманні, зміні та наданні даних користувачеві. По суті, контролер є сполучною ланкою між представленням і моделлю.

За допомогою функцій геттера і сеттера контролер витягує дані з моделі та ініціалізує представлення. Якщо є якісь оновлення з представлень, він змінює дані за допомогою функції налаштування.

Переваги шаблону MVC [63]:

- найпривабливішою концепцією шаблону MVC є розділення проблем;
- сучасні вебзастосунки дуже складні, і внесення змін іноді може бути проблемою;
- керування інтерфейсом у менших окремих компонентах дозволяє програмі бути масштабованою, підтримуваною та легко розширюваною.

3.3.2 MVVM

Model-View-ViewModel (MVVM) – це структурний шаблон проектування, який розділяє об'єкти на три окремі групи [64]:

- моделі зберігають дані програми. Зазвичай це структури або прості класи;
- представлення відображають візуальні елементи та елементи керування на екрані;
- моделі представлень перетворюють інформацію моделі у значення, які можна відобразити на поданні. Зазвичай це класи, тому їх можна передавати як посилання [64].

MVVM добре працює, якщо вашій програмі потрібно багато перетворень від моделі до перегляду. Однак не кожен об'єкт точно вписується в категорії моделі, перегляду чи моделі перегляду. Замість цього ви повинні використовувати MVVM в поєднанні з іншими шаблонами проектування.

Крім того, MVVM може бути не корисним у випадку коли вперше створюєте свою програму. Оскільки вимоги застосунка змінюються, доведеться вибирати різні шаблони дизайну на основі вимог, що змінюються. Можна вводити MVVM пізніше в житті програми, коли вам це дійсно потрібно.

Основні критерії патерну MVVM [64]:

- контролер визначає, які представлення має бути відображено в даний момент;
- події можуть вплинути тільки на контролер. Контролер може вплинути на модель і визначити інше представлення;
- можливо кілька представлень лише одного контролера.

3.3.3 MVP (Model View Presenter)

Архітектура MVP (Model – View – Presenter) є одним із найпопулярніших архітектурних шаблонів і діє при організації проєкту [65].

Модель має сутності та послуги. Сутності використовується для вмісту даних, які співставляються з таблицею в базі даних для використання інших причин. Представлення має обов'язки, які взаємодіють з користувачами.

У ньому немає ніякої логіки. Він спрямовуватиме введені користувачем дані та команди (події) до доповідача. Представлення зазвичай має посилання на свого пред'явника. Пред'явник є проміжною частиною між представленням і поделлю. Він буде грати ту ж роль, що й контролер у шаблоні MVC.

Вся бізнес-логіка, логіка даних, перевірка введених даних, отримання подій від представлень, перетворення даних з представлення в модель або з моделі в представленнях реалізована в пред'явнику.

У порівнянні з представленням і Контролером у шаблоні MVC, представлення та пред'явник що присутні у шаблоні MVP, повністю відокремлені один від одного, і вони спілкуються за допомогою інтерфейсу. Пред'явник взаємодіє з представленнями через інтерфейс представлення, це означає, що пред'явник може виконувати всі завдання презентації та навігації без будь-якої залежності від технології інтерфейсу користувача [65]:

- кінцевий користувач взаємодіє лише з представленням;
- один перегляд зіставляється лише з одним доповідачем;
- перегляд посилянь пред'явник, але він не має посилання на модель. І пред'явник також знає про представлення, яке з ним пов'язане;
- шаблон забезпечує двосторонній зв'язок між представленням і пред'явником.

Було обрано розробляти програмне забезпечення за підходом MVC оскільки він дозволяє програмі бути масштабованою, підтримуватися та легко розширюватися.

3.4 Аналіз та вибір способу запуску вебзастосунку

Для роботи програмного забезпечення необхідна реалізація технологій автентифікації та функціонал для роботи лікарів з даними пацієнтів.

Програмне забезпечення повинне мати можливість надавати безперебійну роботу. Існує різні способи запуску програмного забезпечення на сервері, найбільш розповсюдженою є технологія контейнеризації.

Звичайний спосіб запуску потребує попереднього встановлення усіх необхідних бібліотек та технологій певних версій. Потім необхідно налаштувати кожен із технологій, та вирішити усі проблеми з контролем версій.

Контейнеризація включає поєднання програмного забезпечення разом із усіма пов'язаними з ним залежностями та конфігурацією, бібліотеками, необхідними для швидкого ефективного запуску в усіх необхідних обчислювальних середовищах [66].

Контейнери програм містять компоненти виконання, такі як файли, віртуальні середовища та бібліотеки, що необхідні для запуску потрібного програмного забезпечення. Контейнери додатків споживають набагато менше ресурсів, аніж віртуальні машини, оскільки контейнери не мають повноцінної ОС. Повний набір інформації для виконання в контейнері – це так названі «images». Двигун контейнера розгортає ці «images» на хості [66].

Контейнери працюють на базі існуючих ОС, запускається власна система Linux під час роботи з ОС Windows. Це спрощує та підвищує швидкість запуску програмного забезпечення та надає можливість розроблювати програмне

забезпечення у однаковому середовищі неважливо у якій ОС розгорнуті контейнери. Саме тому для програмного забезпечення що розроблюється буде здійснена система контейнеризації.

На даний момент існує декілька найпопулярніших систем контейнеризації, наприклад, Portainer та Docker та Kubernetes [67].

В таблиці 3.2 зображений опис найпопулярніших систем контейнеризації.

Таблиця 3.2 – Список найпопулярніших систем контейнеризації

Назва системи	Опис
Docker	Docker усуває повторювані повсякденні завдання конфігурації та використовується протягом усього життєвого циклу розробки для швидкої, легкої та портативної розробки додатків – настільних і хмарних. Комплексна наскрізна платформа Docker включає інтерфейси інтерфейсу користувача, інтерфейси командної команди, API та безпеку, які розроблені для спільної роботи протягом усього життєвого циклу доставки програми [67]
Portainer	Portainer – це легкий інтерфейс користувача для керування контейнерами з відкритим вихідним кодом, який дозволяє без зусиль обробляти ваші хости Docker або кластери Swarm. Він підтримує платформи Linux, Windows і OSX. Він містить один контейнер, який можна виконати на будь-якому механізмі Docker [68]
Kubernetes	Портативна, розширювана платформа з відкритим кодом для управління навантаженнями та послугами з контейнерами, що полегшує як декларативну конфігурацію, так і автоматизацію. Має велику, швидко зростаючу екосистему. Послуги, підтримка та інструменти Kubernetes широко доступні. Він дозволяє балансувати навантаження між окремими кластерами або віртуальними машинами. Kubernetes базується на системі контейнерів Docker [69]

Станом на 2021 рік Docker [68] є однією із найпопулярніших систем контейнеризації. Саме тому вона має величезну спільноту розробників які діляться досвідом з новими користувачами. Проаналізувавши таблицю 3.2 можна зробити висновок, що система Docker є ефективною та зручною, але через те, що у системі, що розроблюється більше ніж один компонент, які потребують контейнеризації, можна обрати Docker-compose, оскільки ця система дозволяє

без проблем налаштувати конфігурацію для усіх необхідних контейнерів та зв'язати їх між собою. Усі інші системи мають схожий функціонал, але вони менш розповсюджені, деякі функції платні та мають меншу підтримку спільноти розробників. На рисунку 3.2 можна побачити систему контейнеризації мікросервісів за допомогою Docker. На рисунку 3.3 зображена архітектура системи з використанням Docker технології. Створюється докер-файл, який збирає «image», який вже збирає контейнер на основі конфігураційних файлів. Далі ці «image» можна відправити на віддалений сервер на dockerhub та завантажувати з будь-якого місця вже з готовою конфігурацією.

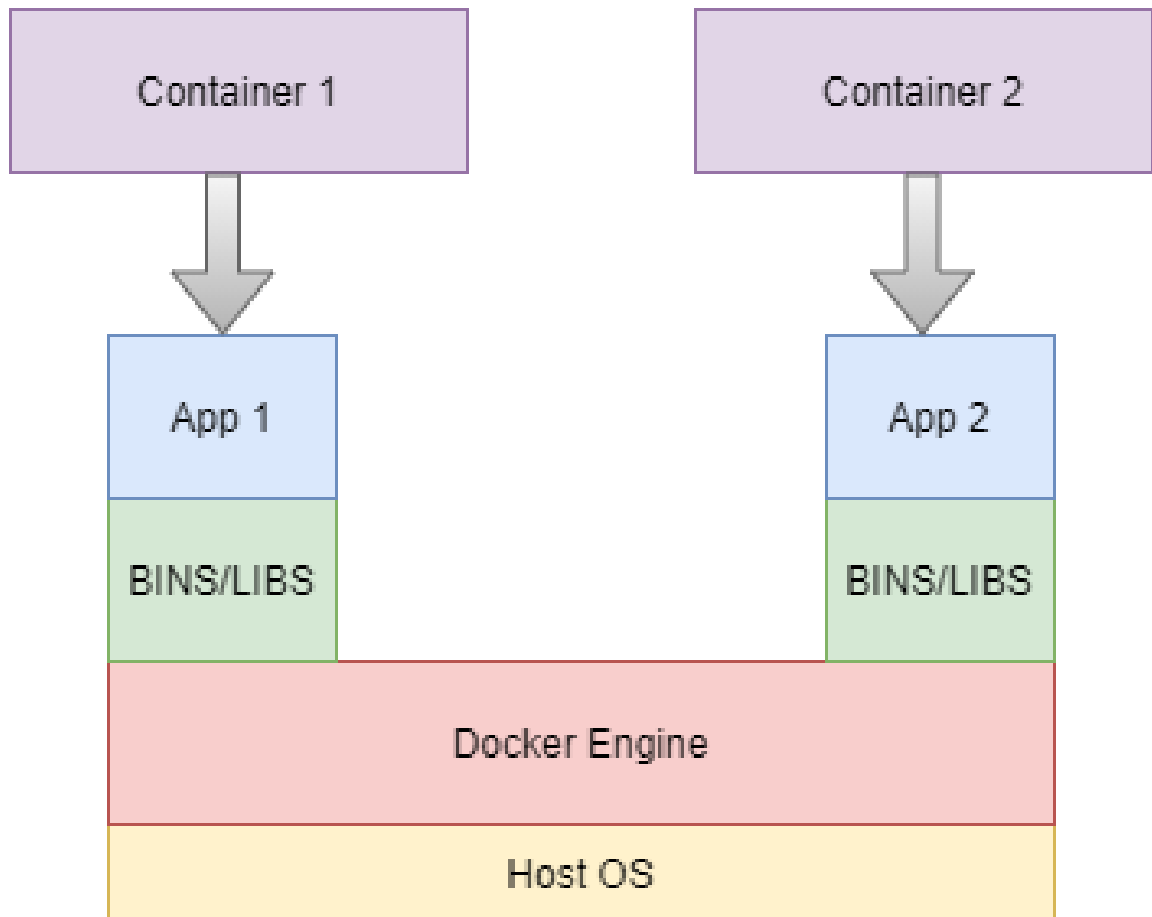


Рисунок 3.2 – Система контейнеризації Docker

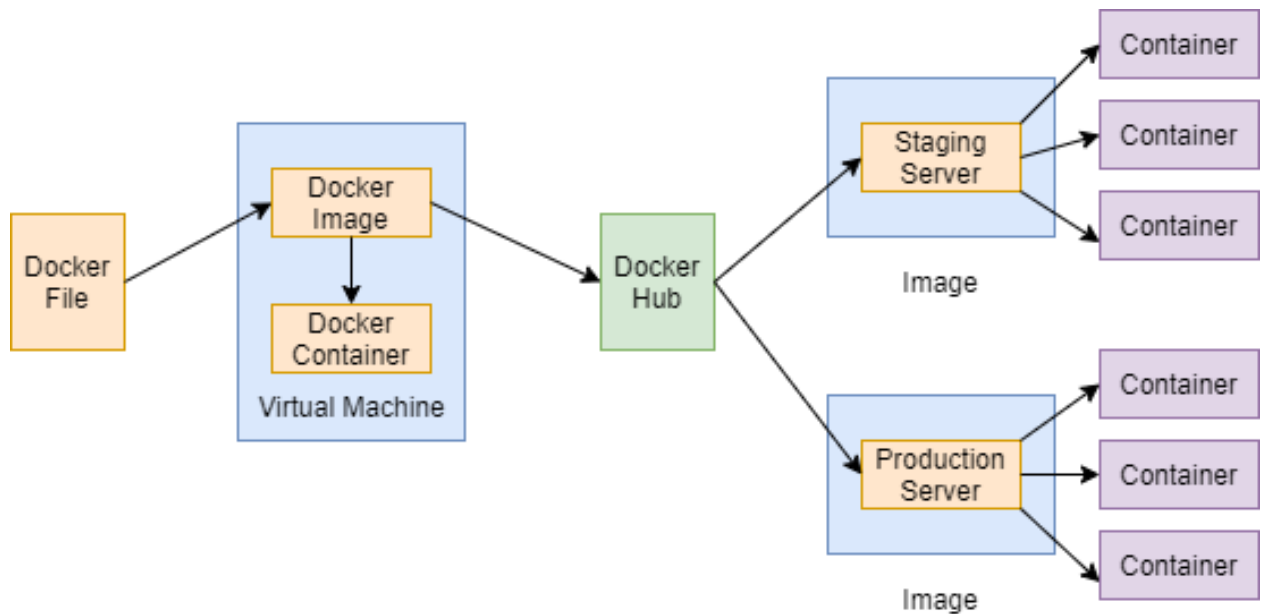


Рисунок 3.3 – Архітектура системи з використанням Docker

Таким чином, Docker є оптимальним рішенням для розробки програмного забезпечення. Також програмне забезпечення, що розроблюється треба завантажити на віддалений репозиторій, для запобігання проблем та для легкого завантаження програмного забезпечення на сервер.

3.5 Аналіз та вибір методології для керування процесом розробки

Для оптимального розподілу часу, ресурсів та покращення якості вихідного продукту необхідно дотримуватися певної методології розробки. В таблиці 3.3 можна розглянути найбільш популярні методології розробки для керування проєктами станом на 2021 рік [70]. На даний момент найбільш популярними методологіями розробки є Agile, Waterfall та Scrum, який є оновленою версією Agile.

Таблиця 3.3 – Методології розробки програмного забезпечення [70]

Методологія	Опис
Agile	Це методологія, яка зосереджена на адаптивному плануванні, еволюційному розвитку та постійному вдосконаленні за рахунок гнучкого реагування на зміни. Основна мета цієї методології – забезпечити ранній випуск програмного забезпечення з меншою ймовірністю помилок або помилок. У цьому методі розробка поділяється на різні фази, і вони відомі як спринти. Кожен спринт триватиме кілька тижнів, і члени команди розробників проєкту отримають список результатів для цього конкретного спринту.
Waterfall	Модель водоспаду є найбільш традиційною і широко використовуваною методологією розробки. Це класичний цикл розробки додатків, а підхід Waterfall – це послідовна модель, яка розділяє будівництво на заздалегідь визначені фази. Кожна фаза повинна бути завершена до початку наступного кроку без перекриття задач між етапами. Такий підхід до розробки не дозволяє нам повернутися до попередніх етапів, якщо потрібні якісь зміни.
Scrum	Scrum вносить ясність, структуру та послідовність у процес команди. Хоча кожна команда розвивається незалежно, можна використовувати Scrum, щоб переконатися, що всі знаходяться на одній хвилі. Програмування на основі мікросервісів, DevOps та всі аспекти DevOps вимагають від розробників програмного забезпечення бути гнучкими та вимагають від них вивчення кількох нових технологій програмування та прийняти нові підходи. Scrum виводить розробку програмного забезпечення на новий рівень і дозволяє гнучко вносити зміни до програмного продукту, що розроблюється безпосередньо під час розробки [71]

Проаналізувавши таблицю 3.3, можна зробити висновок, що для розроблюваного продукту найкраще підходить методологія Scrum.

3.6 Проєктування архітектури системи

Програмне забезпечення потребує вебзастосунок, базу даних для зберігання даних користувачів та сценаріїв лікування та VR застосунку для надання доступу пацієнтам до терапії за допомогою технології віртуальної реальності. Архітектура програмного забезпечення, що розроблюється, зображена на рисунку 3.4.

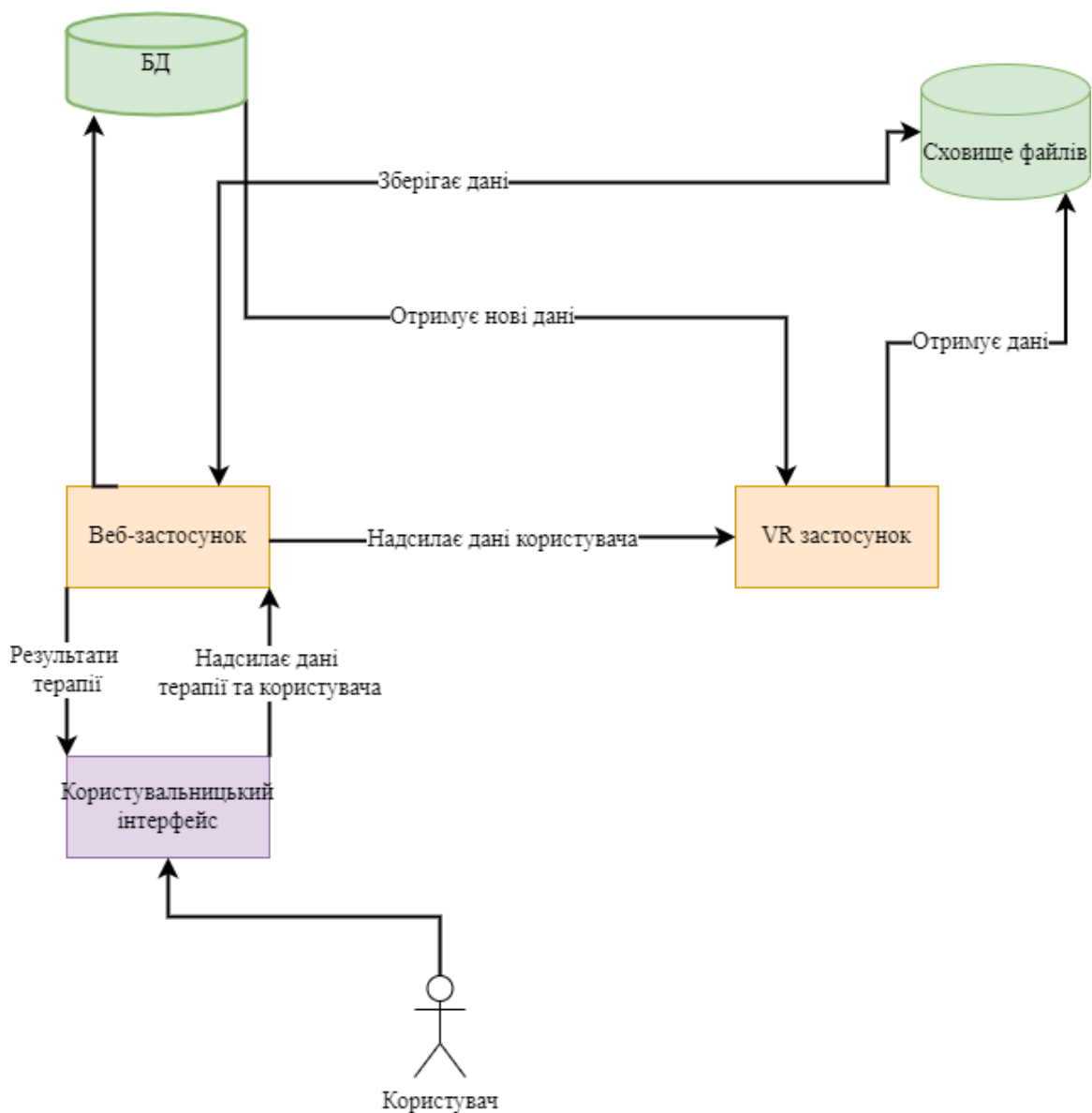


Рисунок 3.4 – Архітектура програмної системи

Вебзастосунок повинен взаємодіяти з базою даних та локальним сховищем файлів, якими буде користуватись VR застосунок. Користувач взаємодії з користувальницьким інтерфейсом, який надає можливість описувати результати терапії, створювати сценарії для лікування, створювати пацієнтів, працювати з даними.

VR застосунок повинен отримувати дані з локального сховища файлів та отримувати нові дані про користувачів та сценарії лікування з віддаленої бази даних.

У результаті спроектована архітектура програмного забезпечення, що дозволяє реалізувати взаємодію між вебзастосунком та VR застосунком програмного забезпечення, що розроблюється.

3.7 Проектування структури СКБД

Для програмного забезпечення що розроблюється необхідно зберігати дані користувачів. На рисунку 3.5 зображена концептуальна модель бази даних, яка складається з чотирьох сутностей: пацієнт, лікар, захворювання та сценарій лікування. Сутності це множина понять реальних або абстрактних об'єктів реального світу, процесів чи явищ. Атрибутами називають інформаційне відображення властивостей об'єктів.

На рисунку 3.5 блакитним позначена назва сутності, помаранчевим унікальний ідентифікатор (id) кожної сутності.

Опис атрибутів і сутностей даного програмного продукту зображено в таблиці 3.4.

Атрибут id відповідає за персональний унікальний ключ для кожної сутності, що автоматично генерується СКБД.

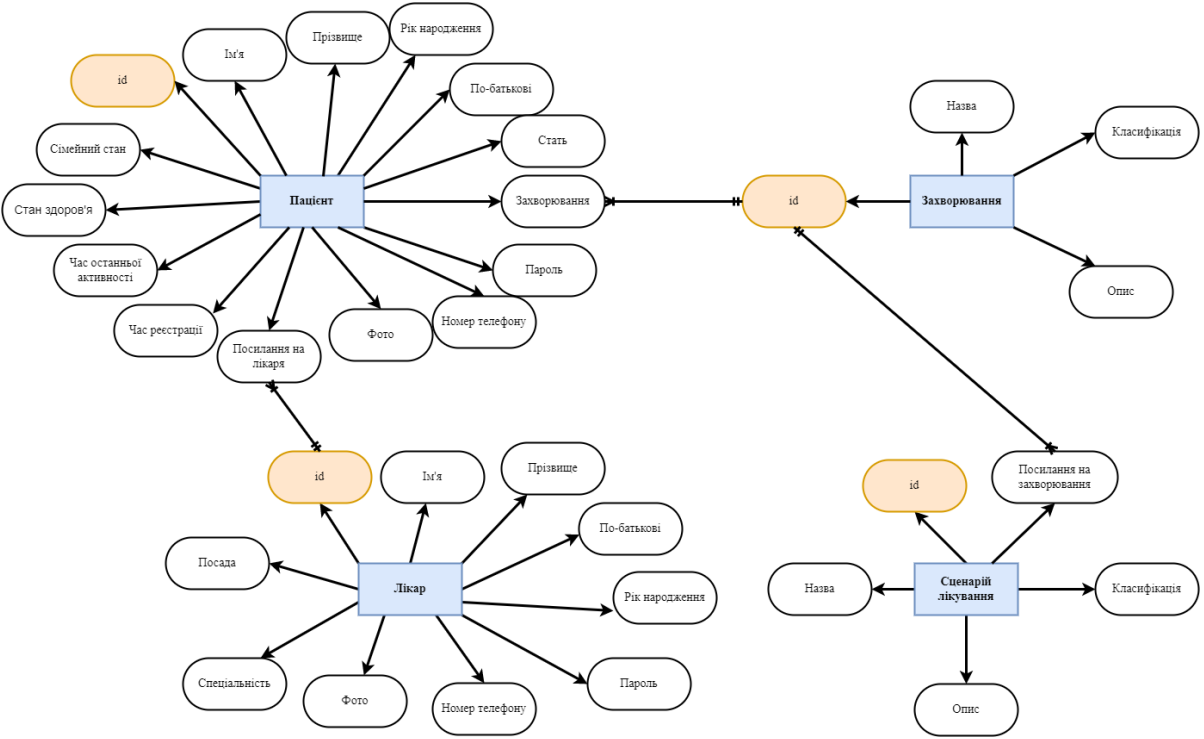


Рисунок 3.5 – Концептуальна модель БД

Таблиця 3.4 – Опис сутностей і атрибутів

Атрибут		Опис сутності
1	2	3
Пациент	<u>_id</u>	Містить інформацію про пацієнта. Вона складається з персональних даних пацієнта, його лікаря, стан здоров'я та час реєстрації. Атрибут персональний ключ пацієнта є унікальним. Містить атрибут стан здоров'я, який корегує лікар під час терапії.
	Ім'я	
	Прізвище	
	По-батькові	
	Захворювання	
	Пароль	
	Номер телефону	
	Фото	
	Посилання на лікаря	
	Час реєстрації	
	Час останньої активності	
	Сімейний стан	
	Стан здоров'я	
	Стать	

Продовження таблиці 3.4

1	2	3
Лікар	_id	Містить інформацію про лікаря. Складається з персональних даних лікаря, його посади та спеціальності.
	Ім'я	
	Прізвище	
	По-батькові	
	Рік народження	
	Пароль	
	Номер телефону	
	Фото	
	Спеціальність	
	Посада	
Захворювання	_id	Містить інформацію про захворювання, які можна лікувати та сценарії для лікування
	Назва	
	Класифікація	
	Опис	
	Посилання на сценарій лікування	
Сценарій лікування	_id	Містить унікальний ідентифікатор для сценарію, назву та опис того, що відбувається під час терапії
	Назва	
	Класифікація	
	Опис	

Наведені сутності зв'язані між собою зв'язками відношень один до багатьох та багато до багатьох. Схема бази даних наведена на рисунку 3.6.

Спроектована база даних надає можливість правильно зберігати дані користувачів, швидко та зручно витягувати дані, за рахунок зв'язків сутностей.

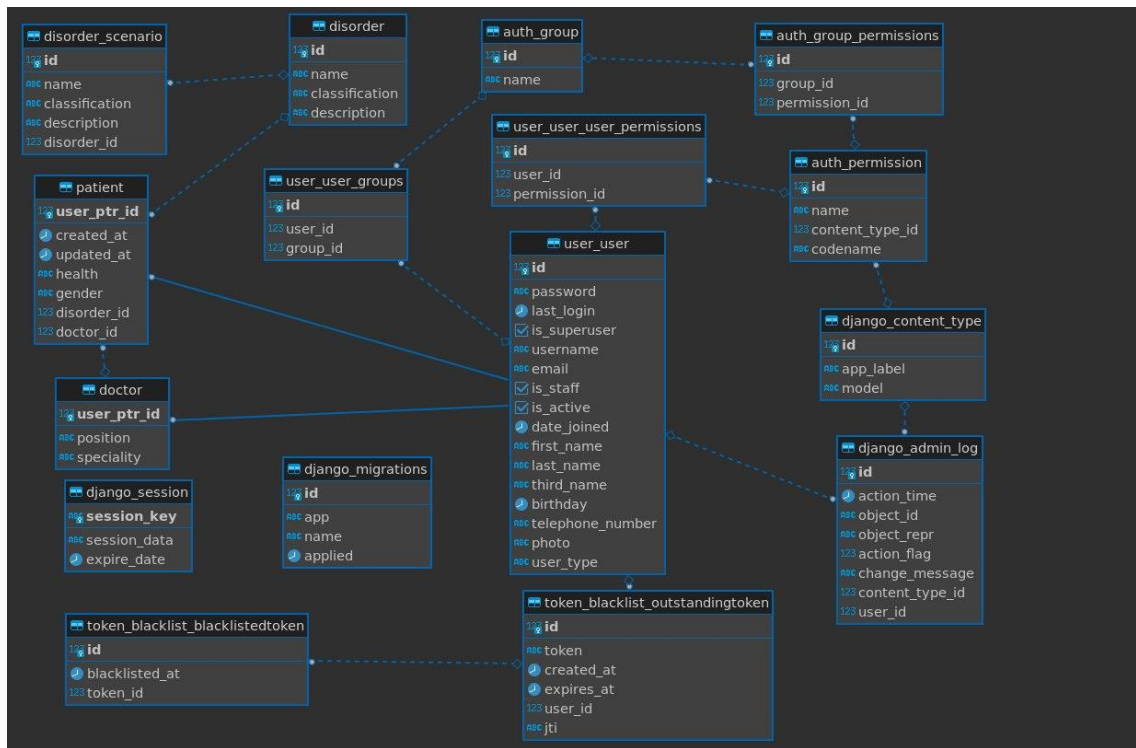


Рисунок 3.6 – Схема бази даних

Була спроектована структура бази даних, яка дозволить ефективно та зручно маніпулювати даними піцієнтів, лікарів, захворювань та сценаріями для лікування.

3.8 Висновки до розділу 3

У даному розділі були визначені основні вимоги до програмного забезпечення, що розроблюється.

Було обрано розробляти програмне забезпечення за підходом Proof of Concept через високу складність проєкту та витрати на проєкт, що розроблюється.

Було прийняте рішення розгортати проєкт у контейнерах Docker. Також програмне забезпечення, що розроблюється треба завантажити на віддалений

репозиторій, для запобігання проблем та для легкого завантаження програмного забезпечення на сервер.

У якості вхідних даних пацієнт надає лікарю інформацію для отримання рекомендацій щодо лікування психічних розладів за допомогою технології віртуальної реальності. У якості вихідних даних користувач пацієнт отримуватиме можливість взаємодіяти зі сценаріями терапії у віртуальній реальності. Лікар отримуватиме зворотний зв'язок від пацієнта.

Була спроектована структура бази даних, яка дозволить ефективно та зручно маніпулювати даними піцієнтів, лікарів, захворювань та сценаріями для лікування.

4 ВИБІР ІНСТРУМЕНТАРІЮ РОЗРОБКИ

4.1 Аналіз та вибір СКБД

Для програмного забезпечення, що розроблюється необхідно зберігати дані клієнтів та лікарів. Для цього необхідно використовувати спеціалізовану базу даних.

База даних – це організована сукупність структурованої інформації або даних, які зазвичай зберігаються в електронному вигляді в комп’ютерній системі. База даних зазвичай контролюється системою керування базами даних (СКБД). Разом дані та СКБД разом із додатками, які з ними пов’язані, називають системою баз даних, часто скорочено до просто бази даних [72]. Бази даних бувають реляційні (SQL) та нереляційні (NoSQL). Основні відмінності реляційних та нереляційних баз даних [73]:

- бази даних SQL є реляційними, бази даних NoSQL нереляційними;
- бази даних SQL використовують структуровану мову запитів і мають попередньо визначену схему. Бази даних NoSQL мають динамічні схеми для неструктурованих даних;
- бази даних SQL мають вертикальне масштабування, а бази даних NoSQL горизонтальне;
- бази даних SQL засновані на таблицях, тоді як бази даних NoSQL є сховищами документів, ключів і значень, графіків або широких стовпців;
- бази даних SQL краще для багаторядкових транзакцій, тоді як NoSQL краще для неструктурованих даних, таких як документи або JSON.

Найпопулярніші СКБД станом на початок 2021 року є MySQL, PostgreSQL, MSSQL, MongoDB та SQLite. Статистика використання відображена у рисунку 4.1. Ці кількісні характеристики означають обсяг дискусій щодо системи, кількість позицій на роботу з цими СКБД та інтерес у цілому [74].

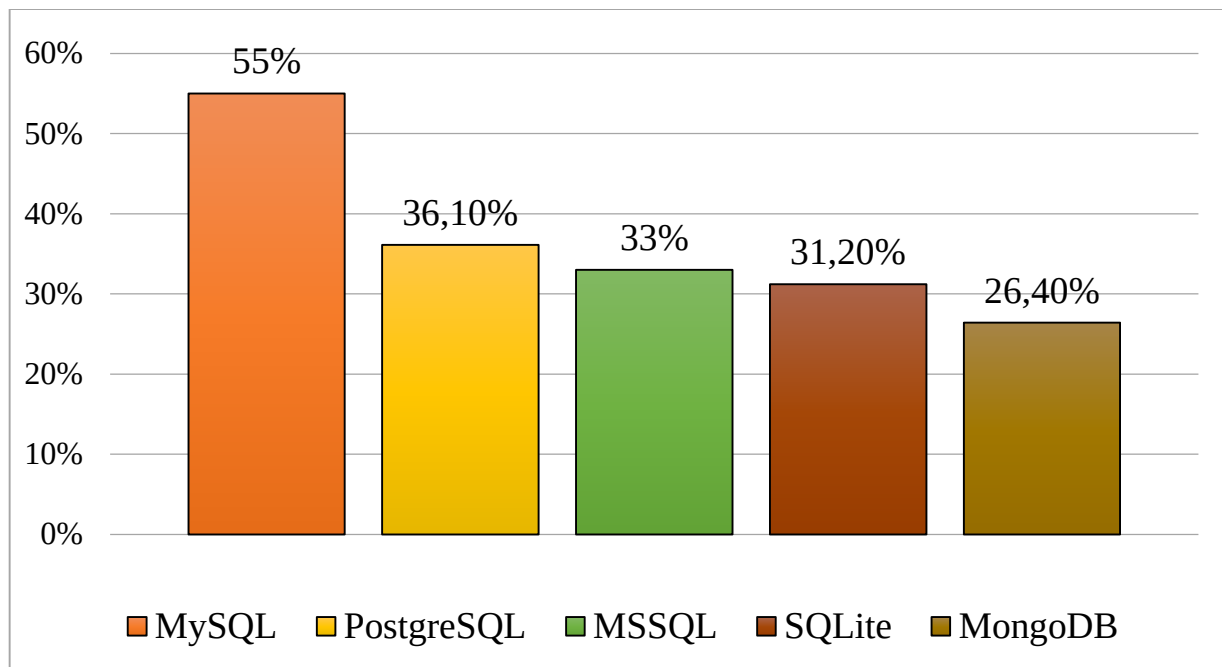


Рисунок 4.1– Статистика популярності СКБД [74]

Порівняння особливостей та відмінностей СКБД відображене у таблиці 4.1, де якість підтримки суспільством, підтримки виробником, документації оцінюється величиною від 1 до 10.

Таблиця 4.1 – Порівняння СКБД

Назва СКБД	MySQL	PostgreSQL	MSSQL	MongoDB	SQLite
Тип БД	реляційна	реляційна	реляційна	нереляційна	реляційна
Рік релізу	1995	1997	1989	2009	2000
Підтримка спільнотою	9	10	8	10	6
Підтримка розробників	8	9	9	8	9
Якість документації	9	10	7	10	7
Споживання ресурсів	7	9	8	6	7
Кошторис ПЗ	безкоштовне	безкоштовне	платне	безкоштовне	платне

СКБД програмного забезпечення, що розроблюється повинно зберігати та оброблювати дані, витримувати високе навантаження, тобто вміти оброблювати

велику кількість запитів за одиницю часу та мати можливість зберігати дані із гнучкою структурою.

Згідно з даними таблиці 4.1 найкращим рішення буде використання PostgreSQL, оскільки вона є реляційною БД, гарно написану документацію, має гарну підтримку спільнотою та виробника.

4.2 Обґрунтування вибору мови програмування

Існує велика кількість мов програмування для вирішення різних задач, та багато рейтингів, які визначають їх популярність, наприклад найпопулярніший сайт спільноти розробників Stackoverflow [75], який склав рейтинг мов програмування серед користувачів. На рисунку 4.2 відображена статистика популярності мов програмування станом на вересень 2021 року [75].

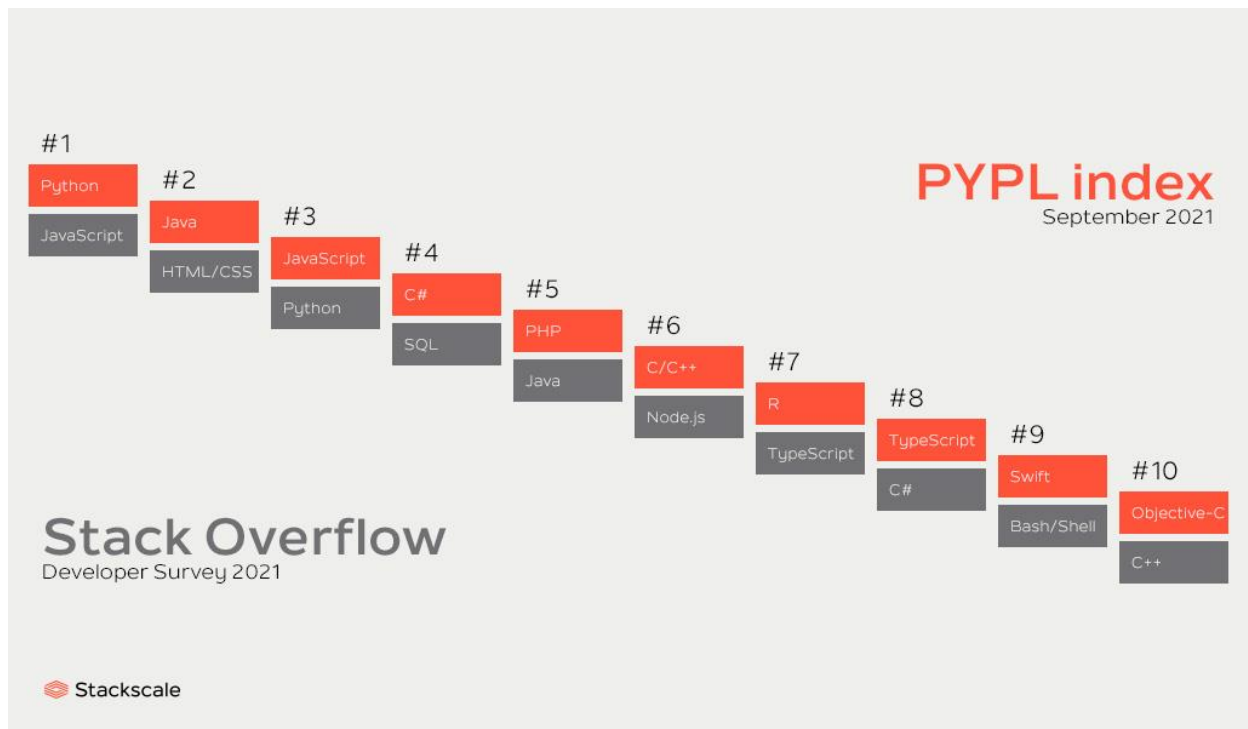


Рисунок 4.2 – Статистика популярності використання мов програмування [75]

На рисунку видно, що Python та JavaScript [76], є найпопулярнішими мовою програмування, на другому місці Java. У таблиці 4.2 наведено порівняння найпопулярніших мов програмування за шкалою від 1 до 10, де 10 – найкраща оцінка, а 1 – найгірша.

Таблиця 4.2 – Порівняння мов програмування

Мова	Python	JavaScript	Java
Рік релізу	1991	1995	1991
Підтримка спільнотою	10	9	8
Підтримка розробників	9	9	6
Якість документації	8	9	8
ООП	модифікований	мінімальний	повноцінний
Підтримка операційними системами	Windows, Linux, MacOS	Windows, Linux, MacOS	Windows, Linux, MacOS
Стандарти синтаксису коду	PEP8	ECMA-262	-

Для вебзастосунку програмного забезпечення, що розроблюється мова програмування повинна відповідати таким вимогам:

- мати можливість ефективно втілювати нові функції;
- підтримуватись різними операційними системами;
- мати якісну документацію, підтримку спільноти та розробників для пришвидшення розробки.

Тому Python є оптимальною мовою для розроблюваного програмного забезпечення. Він має велику базу бібліотек для розробки вебзастосунків. Дана мова програмування має бібліотеки для взаємодії із усіма сучасними базами даних та надає можливість взаємодіяти із API сторонніх вебзастосунків або мікросервісів, має добре прописаний синтаксис коду та є найпопулярнішою мовою програмування станом на вересень 2021 року.

4.3 Аналіз та вибір середовища розробки

Для програмування на мові Python існує багато середовищ розробки.. Кожна з них наділена функціоналом який надає можливість швидко та якісно розроблювати програмне забезпечення. Найпопулярніші з них наведені у таблиці 4.3.

Таблиця 4.3 – Порівняльний аналіз популярних середовищ розробки

Назва	PyCharm	Visual Studio Code	Vim
1	2	3	4
Додатки	Існує багато додатків, але частина з них не безкоштовна.	Існує велика кількість плагінів, усі безкоштовні.	Додатки відсутні через те що робота здійснюється через консольний інтерфейс.
Швидкість налаштування	Необхідно встановити, обрати тему та базові додатки.	Необхідно встановити, обрати тему та базові додатки.	Необхідно встановити, обрати тему та базові додатки.
Інтуїтивність інтерфейсу	Відмінна	Відмінна	Погана через наявність лише консольного інтерфейсу
Перевірка синтаксису	Наявна без додаткових модифікацій.	Наявна після встановлення додатку	Немає
Модифікація інтерфейсу	Наявна без додатків, є вбудовані теми оформлення є можливість їх налаштування.	Наявна без додатків, є вбудовані теми оформлення є можливість їх налаштування.	Немає
Авто-підстановка коду	Наявна без додаткових модифікацій	Тільки після встановлення додатку	Немає
Швидкість запуску	Від 10 до 20 секунд, треба чекати доки завантажаться усі додатки	Від 5 до 10 секунд.	Майже миттєво

Проаналізувавши таблицю 4.3, можна зробити висновок, що PyCharm є оптимальним рішенням для програмного забезпечення, що розроблюється, оскільки він має увесь необхідний функціонал відразу після встановлення а також гнучкість під усі поставлені задачі.

4.4 Вибір фреймворку або бібліотеки для вебзастосунку

Python це високорівнева мова програмування з динамічною строгою типізацією, великою кількістю сумісних фреймворків, яка орієнтована на підвищення продуктивності розробників, читання та якості коду.

Для розробки вебзастосунку програмного забезпечення, що розроблюється для підвищення ефективності та швидкості розробки можна використати один з найбільш популярних python фреймворків (рис. 4.3). Порівняння фреймворків наведено у таблиці 4.4.

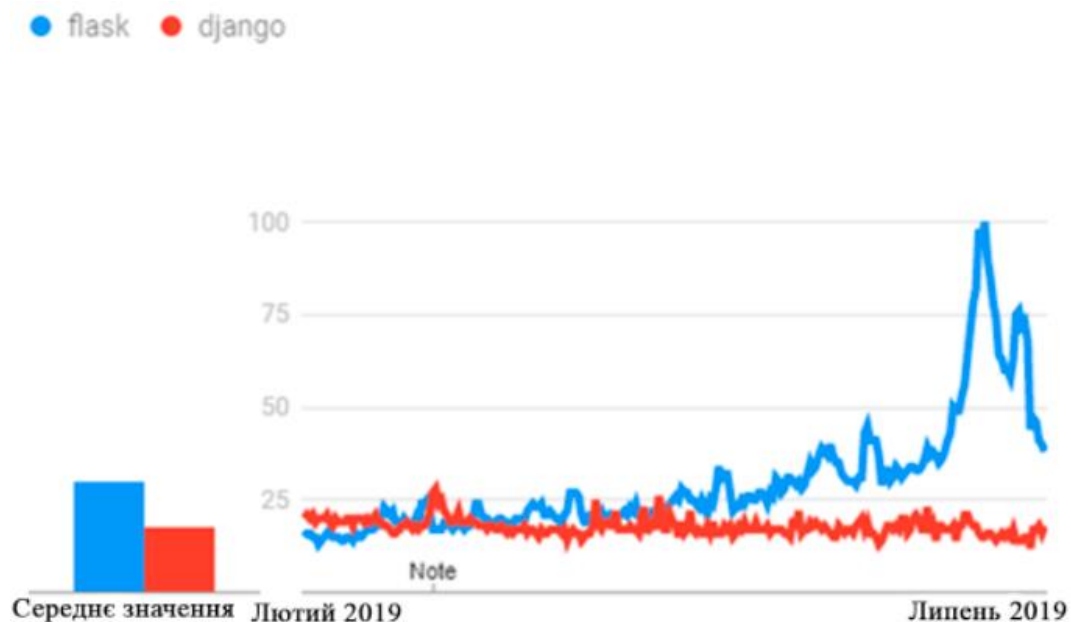


Рисунок 4.3 – Порівняння популярності Django та Flask [77]

Таблиця 4.4 – Порівняння рішень для розробки вебзастосунків [78], [79]

Назва	Django	Flask
Версія	3.2.7	2.0.2
Автентифікація та авторизація	Django має вбудовану функцію аутентифікації та авторизації у своїй структурі, надає функції для керування обліковими записами та підтримки сеансів.	Flask використовує сесии на основі файлів cookie для процесу автентифікації та авторизації. Однак, щоб отримати цю функцію, вам потрібно покладатися на зовнішні бібліотеки та плагіни.
Тестування	Django має вбудовану підтримку для тестування. Для модульного тестування використовується фреймворк Python unittest.	Flask також підтримує вбудоване тестування. Він також використовує ту саму структуру модульного тестування для свого процесу тестування.
Адміністрування	Інструмент адміністратора Django – це вбудований інструмент для завантаження, за допомогою якого розробники можуть створювати вебзастосунки без будь-якого зовнішнього введення.	Функції адміністратора не настільки помітні, як у Django.
Робота з базами даних	Вбудована система ORM дозволяє розробникам використовувати будь-яку базу даних і виконувати звичайні завдання БД без необхідності писати довгі запити.	З Flask розробники повинні працювати з різними базами даних, використовуючи системи ORM для Python і SQLAlchemy як інструментарій SQL. Запити SQL мають бути написані для загальних завдань.
Шаблонізатори	Він поставляється з вбудованим механізмом шаблонів, який заощаджує багато часу на розробку.	Механізм шаблонів Flask Jinja2 заснований на шаблонах Django.
Самостійність	Django це готовий фреймворк з яким можна працювати відразу після встановлення	Flask потребує додаткові модифікації для розширення функціоналу

Було обрано саме Django фреймворк через його обширні можливості без додаткових модифікацій, зручну систему адміністрування, вбудований інтерфейс адміністрування та вбудовану систему Django ORM.

4.5 Вибір програмного забезпечення для розробки VR частини

4.5.1 Unity3d

Unity3D це крос-платформний 3D-рушій і зручне середовище розробки. Досить простий для нових користувачів і досить потужний для експертів. Unity розроблений для створення 3D-ігор та програм для мобільних пристроїв, комп'ютерів, Інтернету та консолей тощо [80].

Розробка віртуального 3D-середовища за допомогою рушія це стратегія об'єднання різних мультимедійних даних в одну платформу. Характеристика рушія, з попередньо встановленими інтерактивними та навігаційними інструментами та інтерфейсом взаємодії з технологією віртуальної реальності дозволяє користувачам досліджувати віртуальні об'єкти та взаємодіяти з ними.

Після проведеного аналізу функціональності рушія Unity3d можна виділити переваги та недоліки, які відображені у таблиці 4.5 [80].

Таблиця 4.5 – Переваги та недоліки використання Unity 3d

Переваги	Недоліки
Наявність безкоштовної версії для розробки	Складність роботи з багатокomпонентними схемами
Наявність потужного та інтуїтивно зрозумілого редактора	Складність роботи з ООП
Підтримка крос-платформної розробки	Потреба у великих обчислювальних потужностях під час розробки
Модульність системи компонентів	
Візуалізація елементів редактора у режимі реального часу	
Підтримка технології віртуальної реальності	

На рисунку 4.4 зображений графічний інтерфейс Unity3d.

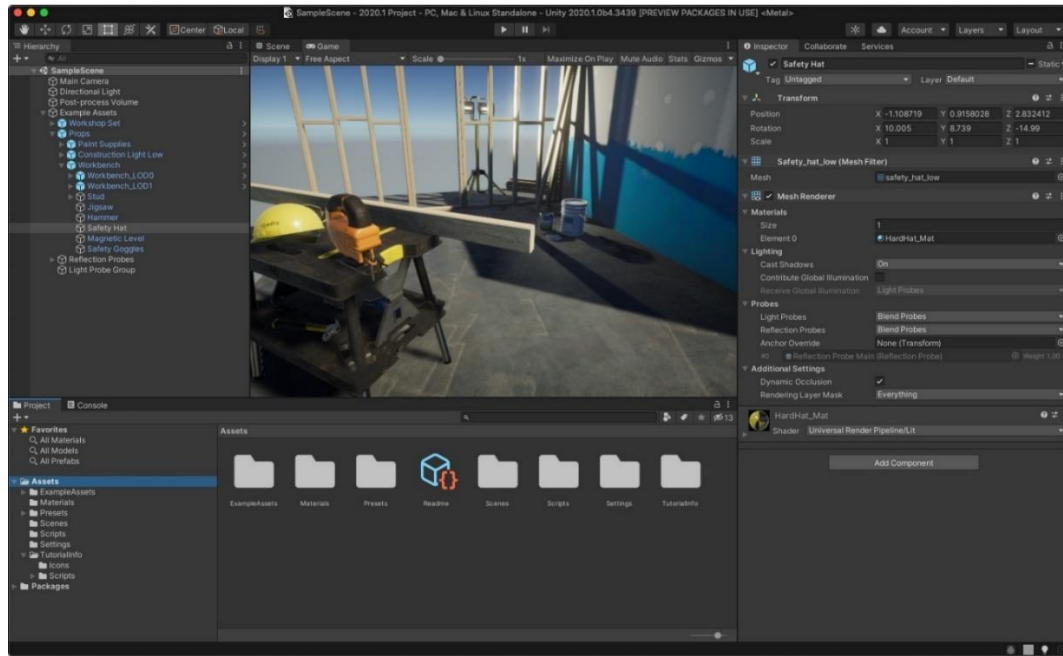


Рисунок 4.4 – Графічний інтерфейс Unity3d

Використовуючи рушій, можна створити простий та ефективний спосіб отримати досвід роботи з віртуальною реальністю, таку як маніпулювання об'єктами, взаємодія неігрових персонажів з користувачем або середовищем.

4.5.2 Unreal Engine

Unreal Engine – це сучасний ігровий рушій, написаний на мові програмування C++ розробила компанія Epic Games. Unreal Engine надає можливість створювати та редагувати 3D анімації, створювати спецефекти в кінофільмах та іграх та розробляти навчальні програми. Код програми на цьому рушії працює на усіх сучасних платформах і операційних системах. Таким чином, однією з особливостей Unreal Engine є його варіативність та універсальність.

До складу технології Unreal входить: графічний рушій, рушій управління фізикою, система штучного інтелекту, управління мережевою і файлової системами, потужний і багатофункціональний вбудований редактор UnrealEd.

UnrealEd (UEd, Unreal Editor) – інструменти редагування рівнів та ресурсів у режимі реального часу, основані на принципі моделювання твердих тіл.

У Unreal Engine можна виділити декілька особливостей, а саме:

- користувач має можливість здійснювати маніпуляції з властивостями 3D-об'єктів, змінювати їх за допомогою команд;
- усі зміни що вносяться користувачем, наприклад, розташування об'єктів, розподіл ефектів, оновлення текстур об'єктів видно через інтерфейс редактора, що дозволяє користувачам перевіряти потенційні помилки;
- редактор Unreal Engine також дозволяє маніпулювати об'єктами, що створені за допомогою графічних пакетів;
- наявність вбудованої системи візуального скриптингу Blueprints.

Переваги та недоліки рушія наведені у таблиці 4.6.

Система Visual Scripting Blueprint в Unreal Engine – це система сценаріїв ігрового процесу, заснована на концепції використання інтерфейсу на основі вузлів для створення елементів ігрового процесу з Unreal Editor. Як і в багатьох поширених мовах сценаріїв, він використовується для визначення об'єктно-орієнтованих класів або об'єктів у механізмі.

Ця система надзвичайно гнучка та потужна, оскільки надає можливість дизайнерам використовувати практично весь спектр концепцій та інструментів, які зазвичай доступні лише програмістам. Крім того, спеціальна розмітка Blueprint, доступна в реалізації Unreal Engine на C++, дозволяє програмістам створювати базові системи, які можуть бути розширені дизайнерами. [81].

Таблиця 4.6 – Переваги та недоліки використання Unreal Engine 4:

Переваги	Недоліки
1	2
Наявність безкоштовної версії для розробки	Складна документація
Наявність потужного та інтуїтивно зрозумілого редактора	Складність Blueprints
Візуалізація елементів редактора у режимі реального часу	Потреба у великих обчислювальних потужностях під час розробки
Blueprints зручні для реалізації базової логіки	
Якість графіки	
Крос-платформність	
Підтримка технології віртуальної реальності	

На рисунку 4.5 зображений графічний інтерфейс Unreal Engine 4.

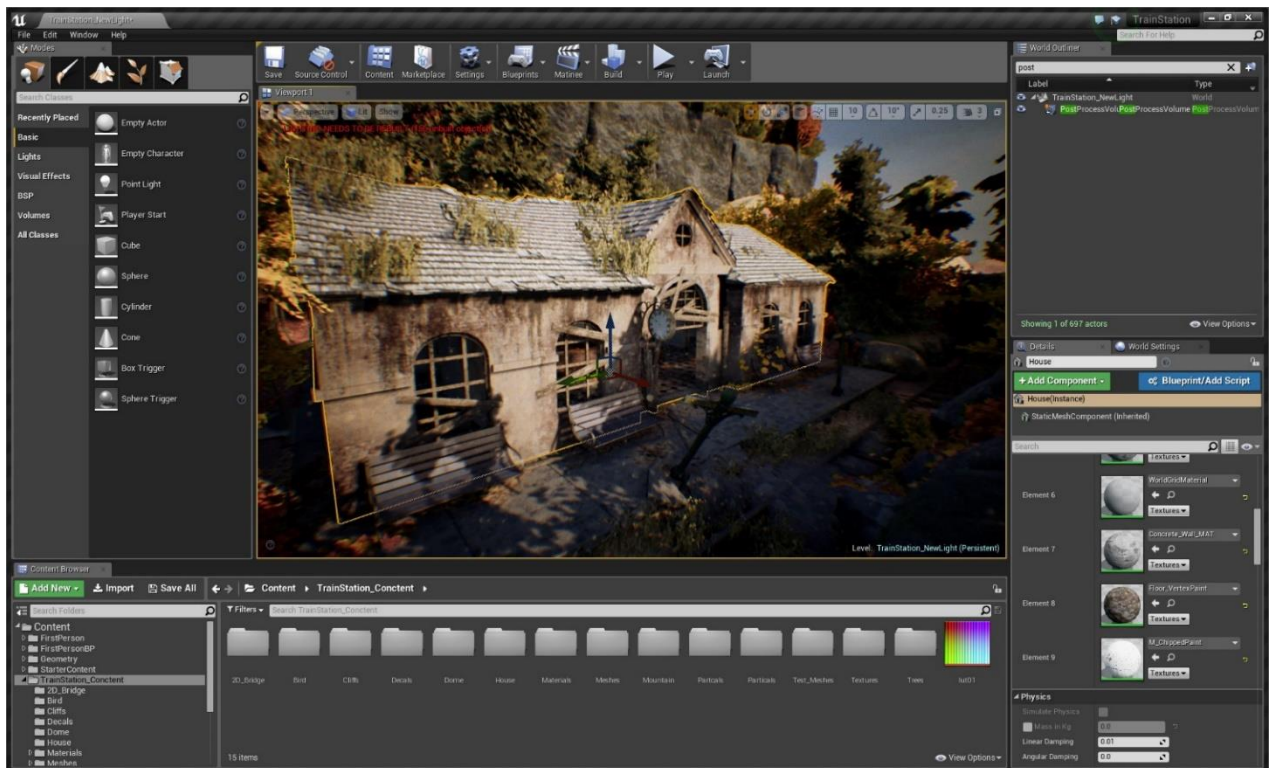


Рисунок 4.5 – Графічний інтерфейс Unreal Engine 4

Таким чином можна порівняти обидва рушія за необхідними критеріями та виставити оцінку від 1 до 10 (де 1 – найгірший результат, 10 – найкращий результат) (таблиця 4.7).

Таблиця 4.7– Порівняння Unity3d та Unreal Engine 4

Критерій	Unity3d	Unreal Engine 4
Документація	9	7
Базовий функціонал	7	9
Адаптивність	9	9
Модифікації та додатки	9	10
Швидкість роботи	7	7
Масштабованість	8	8
Крос-платформність	8	10
Підтримка 3D	9	10
Кошторис використання	8	10
Наявність системи візуального скриптингу	Ні	Так

Таким чином було обрано у якості інструментарію розробки саме Unreal Engine через його функціональність, кросс-платформенність, підтримку ООП під час розробки та якість графіки та наявність системи візуального скриптингу.

4.6 Висновки до розділу 4

У цьому розділі був розглянутий аналіз технологій віртуальної реальності та вибір необхідного інструментарію для розробки системи.

У якості рушію для розробки VR частини проєкту обрано у якості інструментарію розробки саме Unreal Engine через його функціональність, кросс-платформенність, підтримку ООП під час розробки та якість графіки та наявність системи візуального скриптингу.

Було обрано саме Django фреймворк через його обширні можливості без додаткових модифікацій, зручну систему адміністрації та вбудовану систему Django ORM.

Був обраний PyCharm у якості IDE, оскільки він має увесь необхідний функціонал відразу після встановлення а також гнучкість під усі поставлені задачі.

Проаналізовані найбільш популярні СКБД та обрана саме PostgreSQL, оскільки вона є реляційною БД, гарно написану документацію, має гарну підтримку спільнотою та виробника.

Python обраний у якості мови програмування для вебзастосунку через велику базу бібліотек для розробки вебзастосунків та підтримки спільнотою.

5 КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Програмне забезпечення, що розроблюється складається з двох основних частин – з вебзастосунку та VR застосунку. Оскільки програмне забезпечення складається з декількох сервісів, прийнято рішення створити відповідно дві директорії для вебзастосунку та в окремій директорії розробку VR застосунку проєкту як зображено на рисунку 5.1.

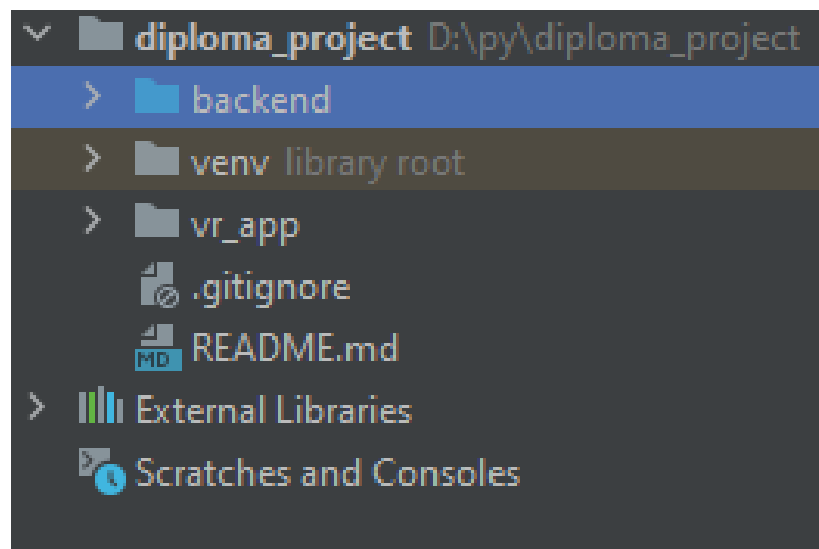


Рисунок 5.1 – Структура файлової системи проєкту

Файлова структура проєкту включає в собі файл для конфігурацій проєкту «.env» та приклад, що зберігається в репозиторії – «.env.local», а файл конфігурації для запуску «.env.develop». Файл «.gitignore» створений для ігнорування непотрібних файлів та файлів з приватною інформацією. Файл README.md містить у собі інформацію про необхідні компоненти, інструкцію для запуску та налаштування.

5.1 Конструювання вебзастосунку проекту

Вебзастосунок являє собою програмне забезпечення, що надає API для взаємодії лікарів з пацієнтами. Структура файлів зображена на рисунку 5.2.

Оскільки вебзастосунок розроблюється на мові програмування Python – кожна директорія повинна включати файл ініціалізації «__init__.py». Цей файл визначає директорію у якості пакетного модуля.

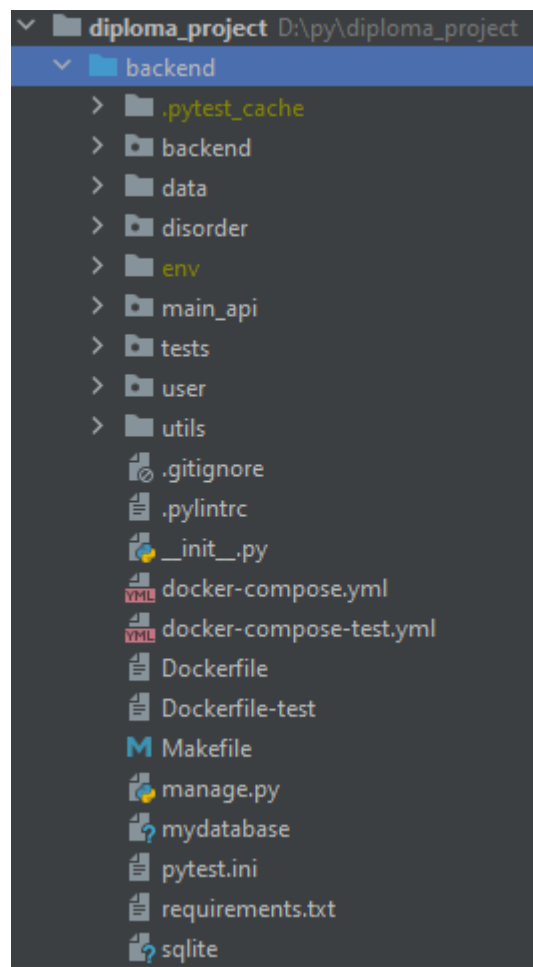


Рисунок 5.2 – Структура файлової системи вебзастосунку

Архітектура вебзастосунку наведена на рисунку 5.3. Файли що використовуються описані у таблиці 5.1.

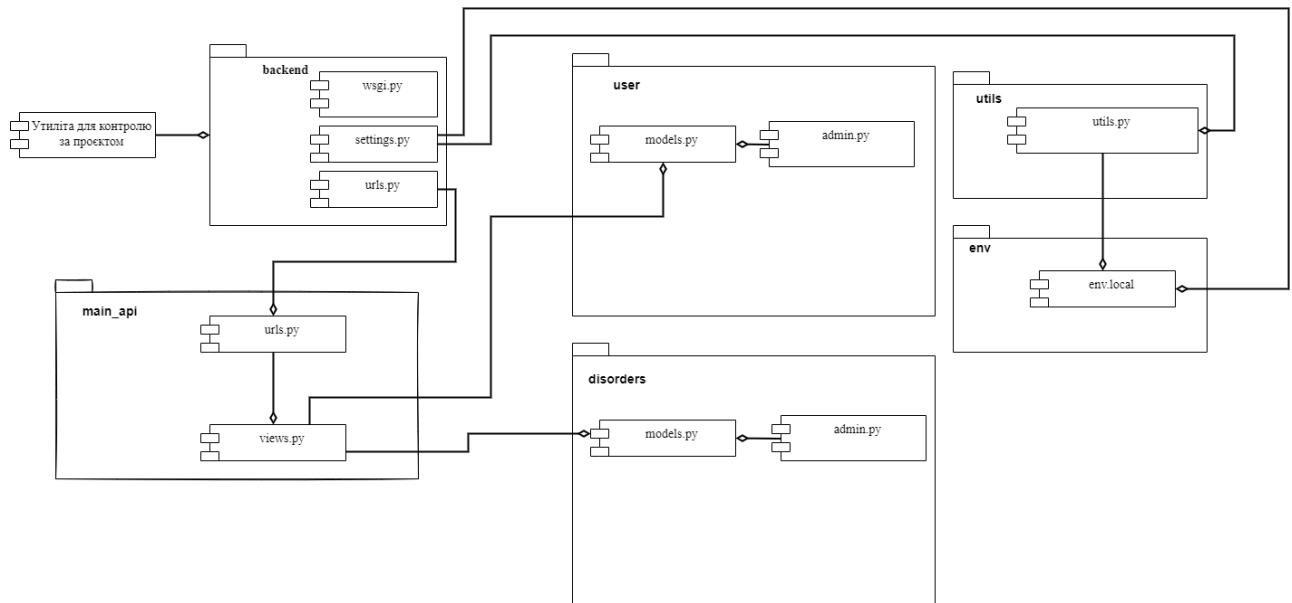


Рисунок 5.3 – Розширена архітектура вебзастосунку

Таблиця 5.1 – Опис файлів та директорій

Назва	Опис
1	2
backend/settings.py	Містить основні налаштування проєкту
backend/urls.py	Містить конфігурацію URL-адрес
backend/asgi.py	Містить дані для ініціалізації застосунку
backend/wsgi.py	Містить дані для ініціалізації застосунку
__init__.py	Пустий файл що вказує на те, що директорія є модулем
apps.py	Файл конфігурації модулю
models.py	Містить моделі та відповідає за створення таблиць у базі даних завдяки вбудованій технології Django ORM
serializers.py	Містить серіалізатори даних
./migrations/	Директорія з міграціями для бази даних
./views/	Директорія з представленнями
constants.py	Файл констант
permissions.py	Містить конфігурацію дозволів
choices.py	Містить кортежі даних для випадального меню
admin.py	Файл для відображення об'єктів у адміністративній панелі
.gitignore	Містить патерни файлів та папок що необхідно приховати від системи контролю версій Git
.pylintrc	Файл конфігурації лінтеру Python

Продовження таблиці 5.1

1	2
docker-compose.yml	Файл налаштувань Docker контейнеру у якому розгортається вебзастосунок
docker-compose-test.yml	Файл налаштувань Docker контейнеру у якому розгортаються тести вебзастосунка
Dockerfile	Файл конфігурації образу Docker
Dockerfile-test	Файл конфігурації образу Docker для тестів
Makefile	Містить команди для керування проєктом з терміналу
manage.py	Файл конфігурації вебзастосунку
pytest.ini	Файл конфігурації запуску тестування вебзастосунку
requirements.txt	Містить назви необхідних бібліотек що треба завантажити для роботи вебзастосунка
initadmin.py	Містить команду для ініціалізації адміністратора під час першого запуску вебзастосунку
./env/	Містить константи з секретною інформацією вебзастосунка

Усі представлення знаходяться в одній директорії `./views/` та описують необхідний функціонал вебзастосунку.

Файли моделей `models.py` в усіх модулях описують класи та залежності таблиць у базі даних через Django ORM.

Файли адміністрування `admin.py` містять конфігурацію для відображення даних у адміністративній панелі.

Файл `Makefile` містить спрощені та інтуїтивно зрозумілі команди для створення образів Docker або їх завантаження, запуску, зупинення та видалення контейнерів, а також команду для запуску контейнеру з тестами.

Файл `.pylintrc` містить конфігурацію модулю, що перевіряє якість коду проєкту та не дає пройти тестам якщо код не відповідає стандартам PEP.

Директорія `env/` відповідає за налаштування різних віртуальних середовищ, наприклад, для тестування та локального запуску.

Діаграма діяльності створення пацієнта зображена на рисунку 5.4.

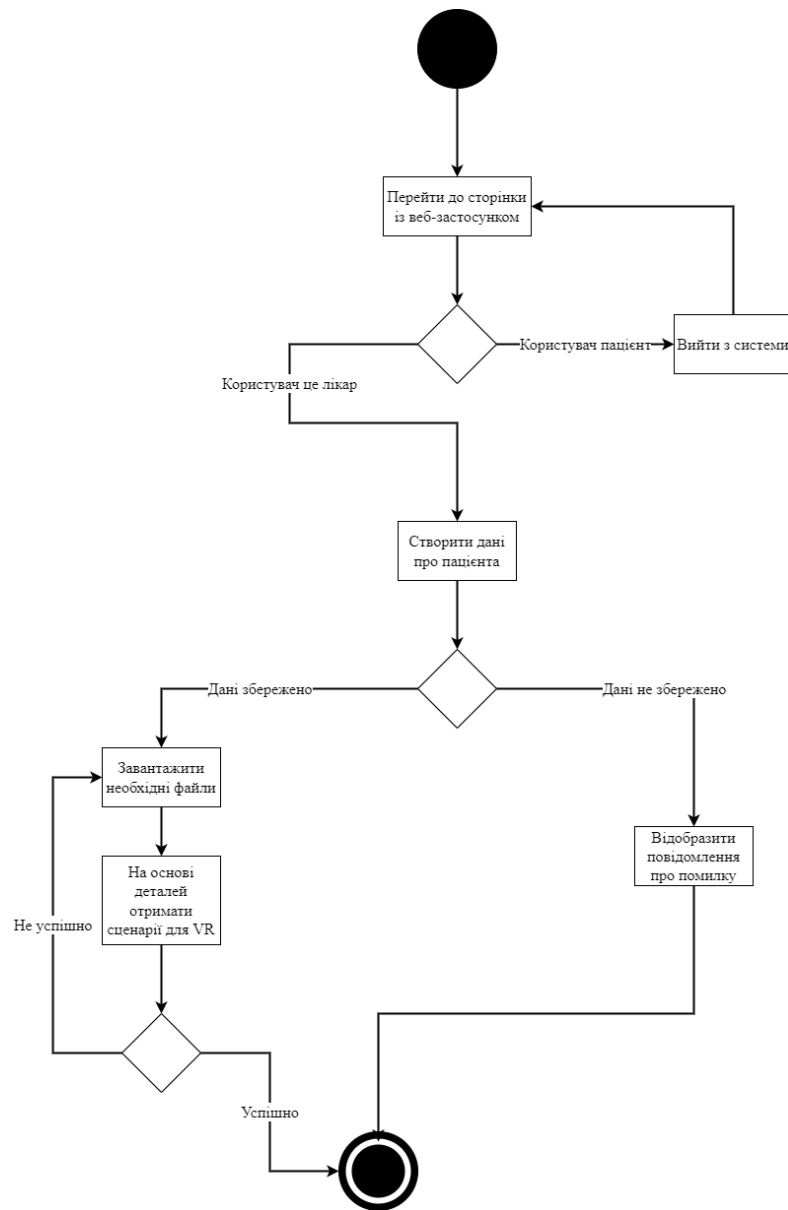


Рисунок 5.4 – Діаграма діяльності створення пацієнта

5.2 Конструювання VR застосунку

Для розробки буде використана система візуального скриптингу Unreal Engine Visual Scripting Blueprints [81].

Для розробки застосунку з використанням технології віртуальної реальності необхідний компонент камери та один компонент контролера

руху (рис. 5.5). За замовчуванням камера вже налаштована на підтримку HMD (Head Mounted Display). Файлова система наведена на рисунку 5.6 та опис файлів та директорій VR застосунку наведені у таблиці 5.2.

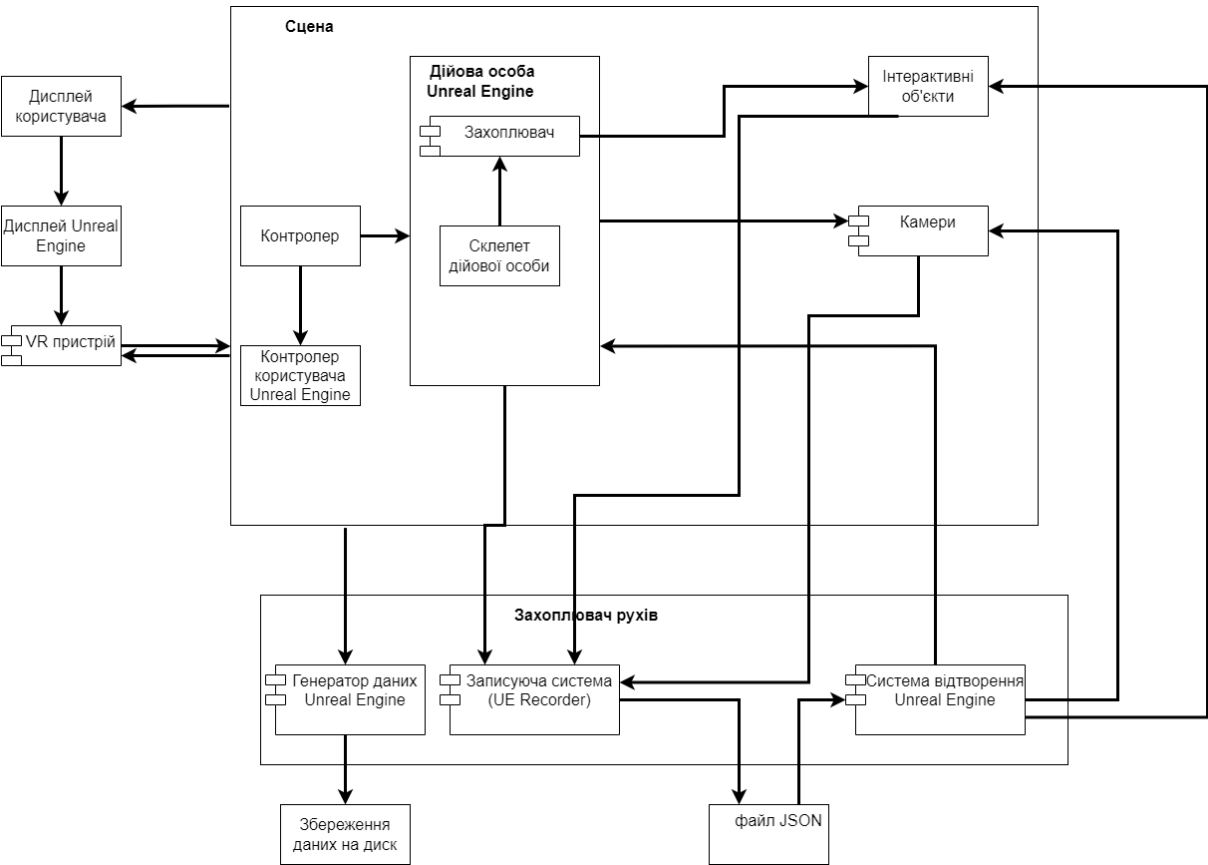


Рисунок 5.5 – Архітектура VR застосунку

Таблиця 5.2 – Опис файлів та директорій

Назва	Опис
1	2
./Engine	Містить основні файли проєкту
Setup.bat	Містить дані ініціалізації проєкту
.editorconfig	Містить конфігурацію середовища розробки
.gitignore	Містить файли що ігноруються системою контролю версій
GenerateProjectFiles	Містить налаштування для генерації вихідних файлів
LICENCE.md	Містить дані про використання ліцензії на розробку
README.md	Містить інформацію про проєкт

Після того, як UE4 було клоновано на вашу робочу станцію, треба налаштувати залежності збірки та створити файли проєкту, які дозволять відкрити проєкт UE4 (рис. 5.7). Щоб налаштувати залежності, треба зконфігурувати файл Setup.bat, розташований у корені репозиторію Unreal Engine. Це завантажить близько 100 ГБ залежностей, тому необхідно переконатись, що є достатньо вільного місця на диску. Після завершення налаштування треба запустити GenerateProjectFiles.bat. У системі візуального скриптингу Blueprints реалізовано завантаження ід сценаріїв лікування (рис. 5.8), які передчасно завантажені до локального сховища VR застосунку та автентифікація за даними, що зберігаються у базі даних вебзастосунка. Автентифікація реалізована за допомогою віртуальної клавіатури, де необхідно ввести логін та пароль після реєстрації лікарем у вебзастосунку у вікні з клавіатурою(рис. 5.9).

Усі моделі об'єктів які використовувались у VR застосунку були отримани на офіційній платформі Unreal Engine з розділу «безкоштовні» [82].

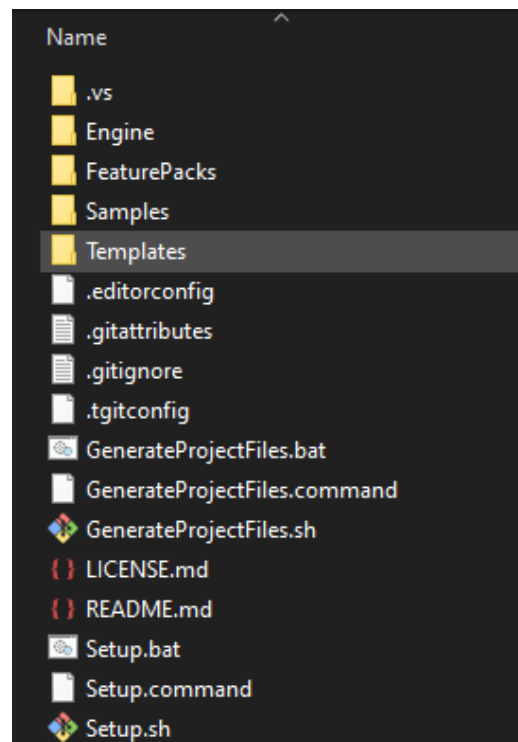


Рисунок 5.6 – Файлова структура VR застосунку

Рисунок 5.8 – Релізація завантаження сцен до проєкту з подальшим їх запуском

На рисунку 5.9 був описаний алгоритм створення віртуальної клавіатури для авторизації користувача.

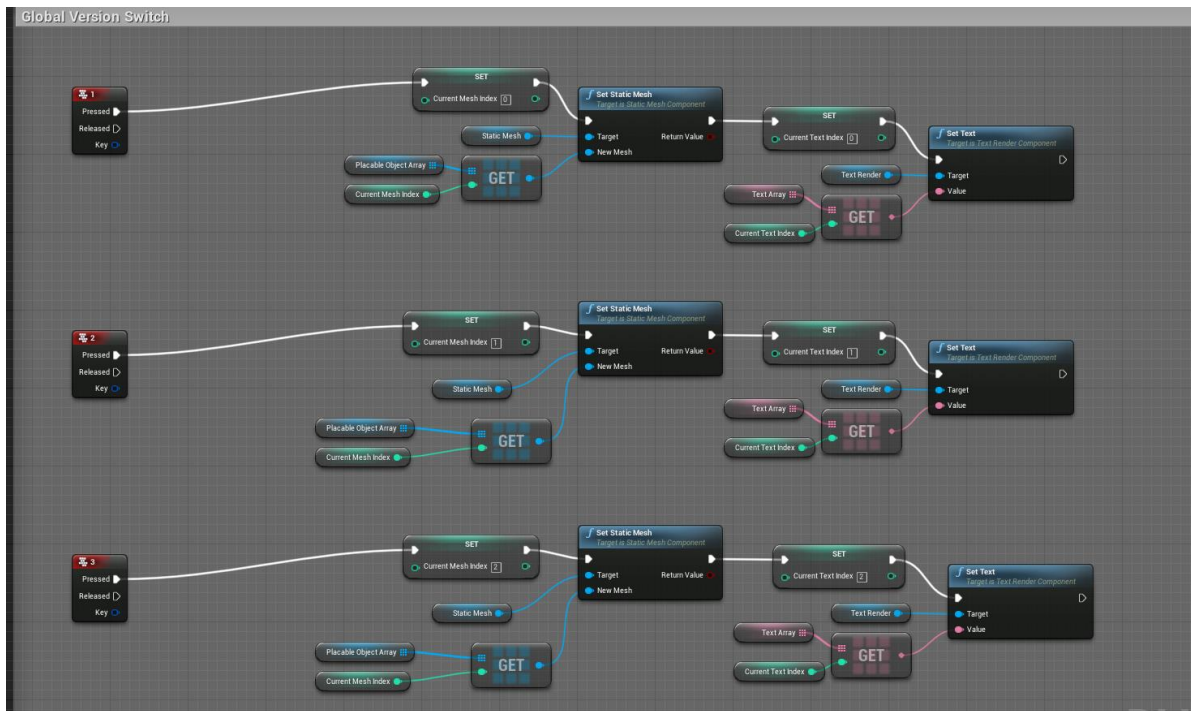


Рисунок 5.9 – Алгоритм роботи віртуальної клавіатури

Для оновлення списку сценаріїв для лікування лікарям необхідно розробити план лікування, отримати дозвіл на проведення терапії та віддати на розробку програмістам, що будуть підтримувати програмне забезпечення. Після реалізації нових сценаріїв лікування треба оновити додаток на усіх комп'ютерах, що використовуються для лікування.

5.3 Висновки до розділу 5

В даному розділі були описані етапи конструювання системи.

Була розроблена файлова структура, яка містить конфігураційні файли, файли середовищ та директорії з сервісами.

Був розроблений та налаштований вебзастосунок програмного забезпечення, що розроблюється, відповідає за роботу вебзастосунку та зберігає дані про користувачів у базі даних. Був розроблений VR застосунок для застосування технології віртуальної реальності для лікування психічних розладів.

Був розроблений функціонал, який дозволить користувачеві підключитись до віртуальної реальності та почати терапію згідно з рекомендаціями лікаря.

Була розроблена клієнтську частину, що являє собою зручну систему, та інтуїтивно зрозумілий інтерфейс, який дозволить ефективно та швидко працювати лікарям з пацієнтами.

6 КЕРІВНИЦТВО З ВИКОРИСТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

6.1 Призначення та умови використання системи

Для підключення до вебзастосунку на сервері знадобиться безперебійне підключення до мережі Інтернет, а також операційна система Windows 7, FreeBSD 3, Linux 2.2 та новіші.

Для користування вебзастосунком необхідне:

- підключення до мережі Інтернет;
- наявність комп'ютерної периферії (клавіатура, комп'ютерна миша);
- браузер Mozilla Firefox 4.0 , Google Chrome 9, Opera 12.0, Apple Safari або новіші.

Для підключення до віртуальної реальності необхідний шолом віртуальної реальності, наприклад HTC Vive, Oculus Rift, Samsung GearVR з доступом у мережу Інтернет.

6.2 Функції системи

Розроблена система повинна виконувати наступні функції:

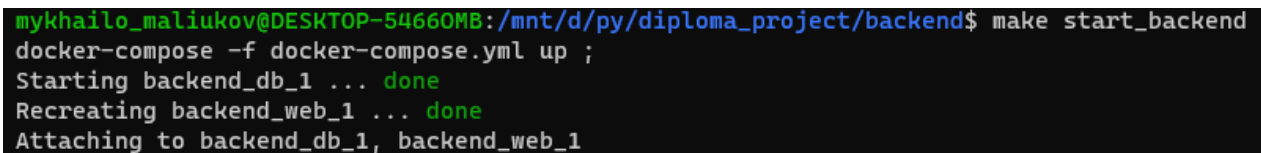
- надавати доступ тільки для авторизованих користувачів;
- надавати доступ до персональних даних пацієнтів тільки лікарям та адміністраторам;
- надавати програмне забезпечення для підключення до віртуальної реальності.

6.3 Інструкція з запуску системи

Проект розгортається у віртуальному середовищі контейнерів Docker та транслює вебзастосунок на порті 8000 та базу даних на порі 5432. Перед запуском необхідно встановити docker, docker-compose, git. Усі налаштування знаходяться у файлах Docker та docker-compose.yml. Під час запуску контейнерів, а саме контейнеру з вебзастосунком з назвою «backend_web» та контейнеру з базою даних PostgreSQL під назвою «backend_db» завантажуються усі необхідні модулі, що вказані у файлі requirements.txt, образ системи (у нашому випадку це ОС Ubuntu) та у разі успішної збірки та завантаження підіймається віртуальна операційна система для роботи. Після цього генеруються (якщо потрібно) міграції та застосовуються до бази даних та створюється адміністратор, якщо адміністраторів не існує (файл initadmin.py). У разі успішного запуску на екрані можна побачити відповідні написи (рис. 6.1).

Діаграма діяльності з запуску вебзастосунку наведена на рисунку 6.2. Створити файли віртуального середовища по прикладу у директорії /env/ з назвами «env.local» та «.env.test». Зконфігурувати їх та запустити команду створення образу, якщо помилок не виявлено.

Для запуску VR застосунку необхідно підключити VR пристрій до комп'ютера та запустити VR застосунок з подальшою авторизацією.



```
mykhailo_maliukov@DESKTOP-54660MB:/mnt/d/py/diploma_project/backend$ make start_backend
docker-compose -f docker-compose.yml up ;
Starting backend_db_1 ... done
Recreating backend_web_1 ... done
Attaching to backend_db_1, backend_web_1
```

Рисунок 6.1 – Демонстрація успішного запуску вебзастосунку

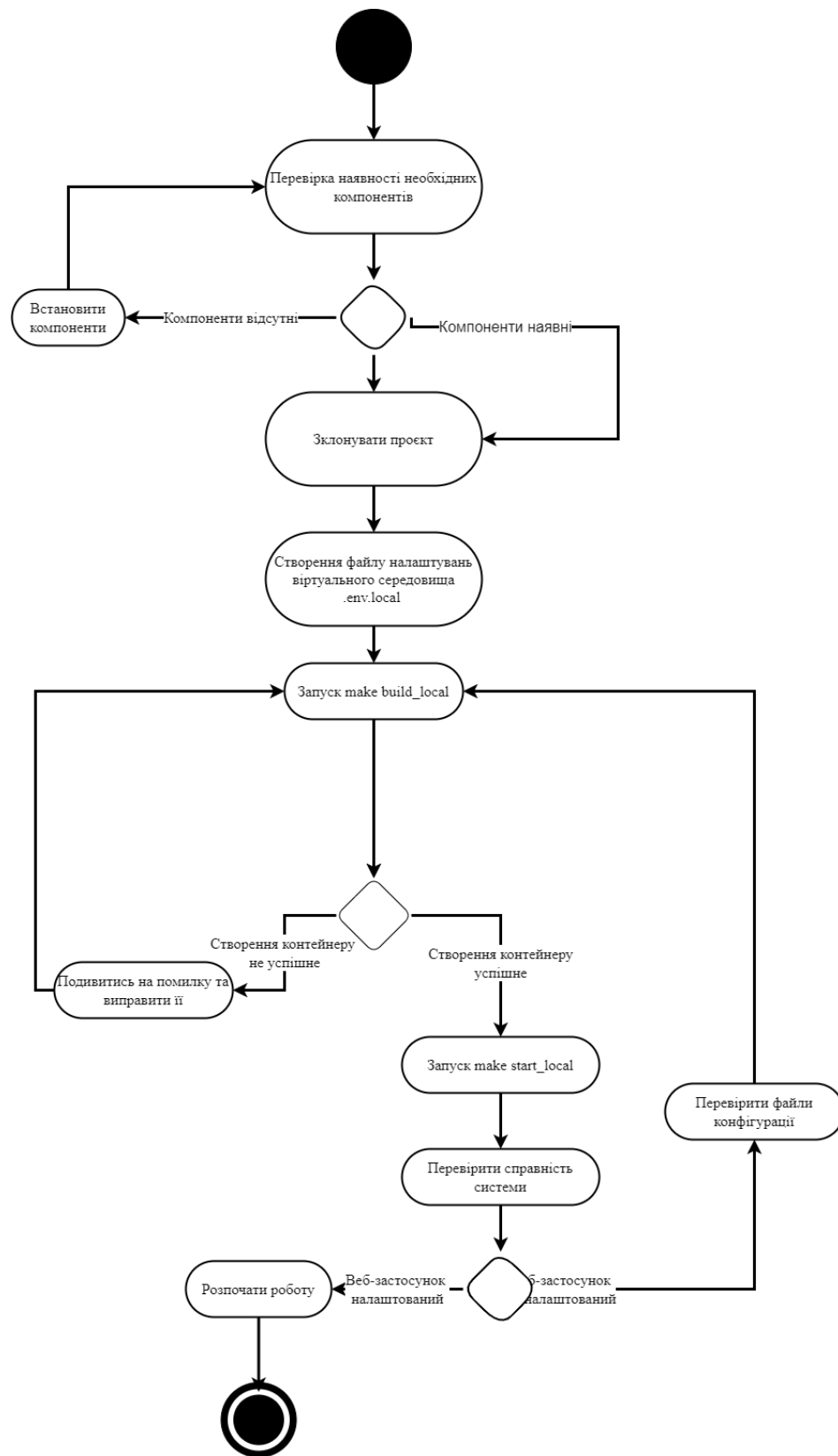


Рисунок 6.2 – Діаграма запуску вебзастосунку

На рисунку 6.3 наведена діаграма діяльності з запуску VR застосунку.

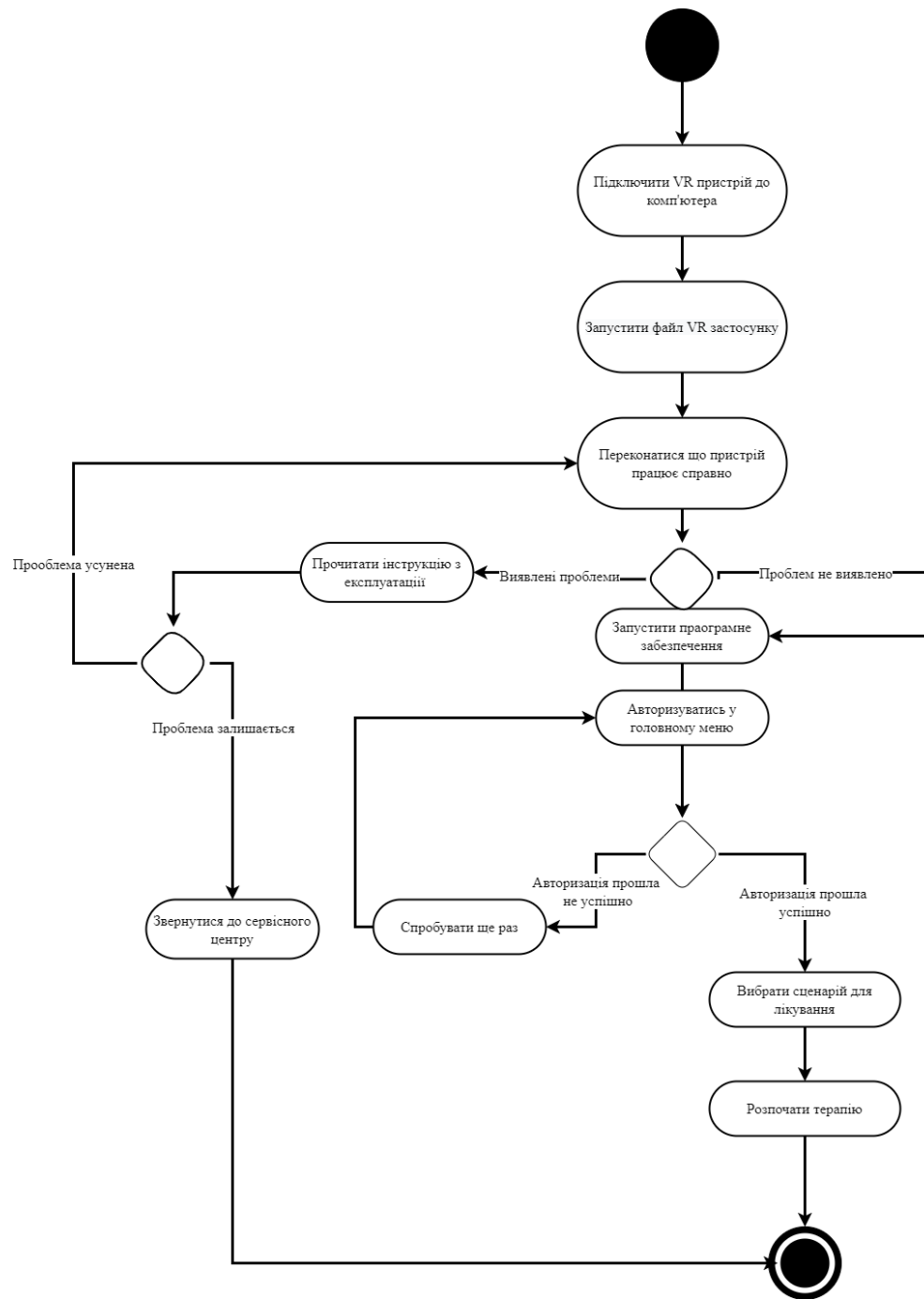


Рисунок 6.3 – Діаграма запуску VR застосунку

У разі проблем з VR пристроєм необхідно виявити причину, якщо це неможливо обов'язково треба звернутись до сервісного центру для уникання непередбачуваних травм.

6.4 Початок роботи з програмним забезпеченням

Першим користувачем вебзастосунку є адміністратор. Логін та пароль вказані у файлі .env.local. Далі адміністратор з адміністративної панелі (рис. 6.4) створює лікарів з їх персональними даними (рис. 6.5) та вносить у базу даних інформацію про захворювання та сценарії їх лікування. Сценарії лікування одобряються кваліфікаційними лікарями та імплементуються за допомогою команди програмістів та завантажуються у VR застосунок. Кожен сценарій має свій опис та id, по якому користувач у VR застосунку зможе вибрати необхідний.

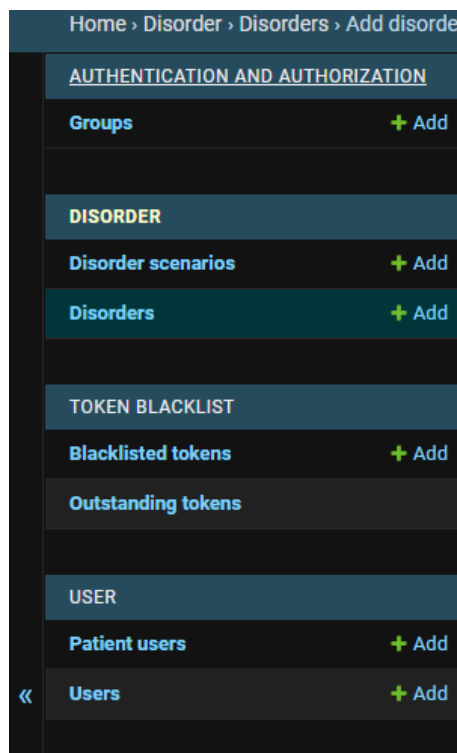


Рисунок 6.4 – Адміністративна панель

Лікарі заходять під своїми логіном та заносять у базу даних інформацію про пацієнтів, створюють сценарії лікування та описують захворювання (рис. 6.6-6.9)

The screenshot shows a web form titled "Add doctor user". It contains several input fields: "First name:" with a placeholder "username", "Last name:" with a placeholder "lastname", "Third name:" with a placeholder "thirdname", and "Telephone number:" with a placeholder "phone_number". Below these is a "Birthday:" section with "Date:" and "Time:" pickers, a "Today" button with a calendar icon, and a "Now" button with a clock icon. A note below the time picker states "Note: You are 2 hours ahead of server time." Further down are "Position:" and "Speciality:" fields. At the bottom right, there are three buttons: "Save and add another", "Save and continue editing", and "SAVE".

Рисунок 6.5 – Процес створення докторів

The screenshot shows a web form titled "Add patient user". It contains several input fields: "First name:" with a placeholder "username", "Last name:" with a placeholder "lastname", "Third name:" with a placeholder "thirdname", and "Telephone number:" with a placeholder "phone_number". Below these is a "Birthday:" section with "Date:" and "Time:" pickers, a "Today" button with a calendar icon, and a "Now" button with a clock icon. A note below the time picker states "Note: You are 2 hours ahead of server time." Further down are "Doctor:" (a dropdown menu), "Disorder:" (a dropdown menu with a plus icon), "Health:" (a text input field), "Gender:" (a dropdown menu with "OTHER" selected), and "Password:" (a text input field). At the bottom right, there are three buttons: "Save and add another", "Save and continue editing", and "SAVE".

Рисунок 6.6 – Процес створення пацієнтів

Add disorder scenario

Name:

Classification:

Description:

Disorder:

Рисунок 6.7 – Процес створення опису сценаріїв лікування

Add disorder

Name:

Classification:

Description:

Рисунок 6.8 – Процес створення опису захворювань

✓ The patient user "Сковорода" was added successfully.

Select patient user to change

Action: 0 of 2 selected

☐	PATIENT USER
☒	Сковорода
☐	Шевченко

2 patient users

Рисунок 6.9 – Вікно перегляду пацієнтів

Для транслявання зображення на персональний комп'ютер у апаратному забезпеченні пацієнта можна увімкнути функцію «Headset Casting» у налаштуваннях (рис. 6.10).

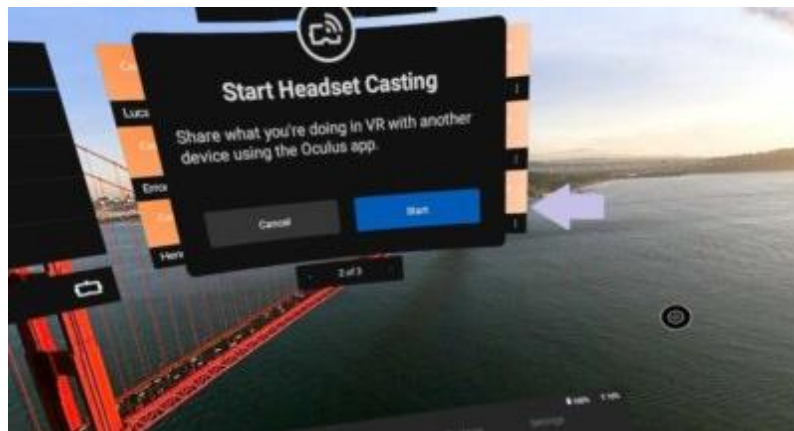


Рисунок 6.10 – Налаштування трансляції зображення

Після завантаження на пристрій програмного забезпечення та авторизації через віртуальну клавіатуру (рис. 6.11)



Рисунок 6.11 – Вікно клавіатури для введення даних для авторизації

Після авторизації пацієнт обирає необхідний сценарій для лікування, який він обрав після консультації з кваліфікованим лікарем та починає терапію за сценарієм, який розробляють кваліфіковані спеціалісти. Приклад сценарію, що не схвалений професійними лікарями та не має медичної ефективності через відсутність сертифікації можна побачити на рисунку 6.12.

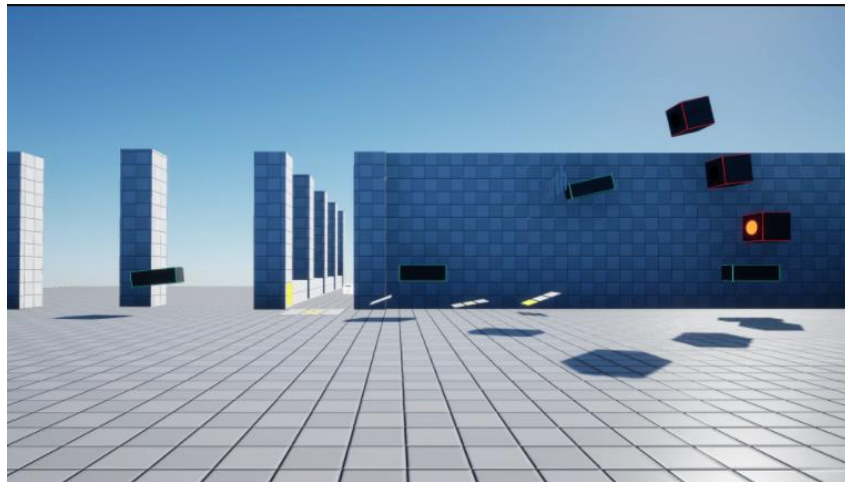


Рисунок 6.12 – Приклад виконання сценарію терапії пацієнтом

Після завершення терапії пацієнту необхідно вийти з програми та звернутися до лікаря за допомогою будь-якого існуючого месенджера та дізнатися про подальші дії, отримати консультацію.

6.5 Висновки до розділу 6

У розділі розглянуто призначення та умови використання програмного забезпечення та процес запуску. Також описане налаштування та наведено детальну інструкцію для користувачів з використання.

У результаті створений спеціальний інтерфейс для керування процесом лікування пацієнтів у вигляді вебзастосунку та інтерфейс для взаємодії користувачів з віртуальним середовищем у віртуальній реальності у вигляді VR застосунку.

ВИСНОВКИ

В результаті проведених досліджень було виявлено, що психічні розлади є розповсюдженим явищем серед як чоловіків так і жінок різного віку. Технологія віртуальної реальності у якості альтернативного способу лікування або використання під час реабілітаційного періоду є ефективною та перспективною.

На основі проведеного аналізу був розроблений вебзастосунок, який відповідає за взаємодію лікарів з даними пацієнтів, сценаріями лікування та описами захворювань, а також VR застосунок для застосування технології віртуальної реальності для лікування психічних розладів.

Наукова новизна роботи полягає в тому, що отримала подальший розвиток технологія віртуальної реальності для лікування психічних розладів у вигляді платформи для лікування, що об'єднує вебзастосунок та VR застосунок, кожен з яких виконує певні функції.

Практична цінність роботи полягає в тому, що розроблене програмне забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності дозволяє взаємодіяти пацієнтам та лікарям, призначати терапію пацієнтам та створювати сценарії для лікування.

В подальшому розроблене програмне забезпечення можна розширити для терапії захворювань інших розділів медицини.

Результати роботи було представлено на Щорічній науково-практичній конференції НУЗП Тиждень науки – 2021, 19-23 квітня 2021 р.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- 1 Saraceno, B. The WHO world health report 2001 on mental health [Text] / B. Saraceno // *Epidemiologia e Psichiatria Sociale*. – 2002. – Vol. 11(2). – P. 83–87.
- 2 Rehm, J. Global burden of disease and the impact of mental and addictive disorders [Text] / J. Rehm, K. D. Shield // *Current Psychiatry Reports*. – 2019. – Vol. 21(2). – P. 1-10.
- 3 What is mental illness? [Electronic source]. – Access mode: <https://www.psychiatry.org/patients-families/what-is-mental-illness>.
- 4 Global, regional, and national incidence, prevalence, and years lived with disability for 354 diseases and injuries for 195 countries and territories, 1990–2017: a systematic analysis for the Global Burden of Disease Study 2017 [Text] / [S. L. James, D. Abate, S. M. Abay et al.] // *The Lancet*. – 2018. – Vol. 392(10159). – P. 1789–1858.
- 5 The Implementation of the WHO mental health gap intervention guide (mhGAP-IG) in Ukraine, Armenia, Georgia and Kyrgyz Republic [Text] / [I. Pinchuk, Y. Yanchik, O. Kopchak et al.] // *International Journal of Environmental Research and Public Health*. – 2021. – Vol. 18(9). – P. 43-91.
- 6 Reforming undergraduate psychiatry training in Ukraine [Text] / [O. Kopchak, I. Pinchuk, B. Ivnev et al.] // *Journal of Medical Education and Curricular Development*. – 2020. – Vol. 7. – P. 1-23.
- 7 Mental health care utilisation among internally displaced persons in Ukraine: results from a nation-wide survey [Text] / [B. Roberts, N. Makhashvili, J. Javakhishvili et al.] // *Epidemiology and Psychiatric Sciences*. – 2017. – Vol. 28(1). – P. 100–111.
- 8 Mobile based mhGAP-IG depression screening in Kenya [Text] / [W. Musyimi, V. N. Mutiso, Z. R. Haji et al.] // *Community Mental Health Journal*. – 2016. – Vol. 54(1). – P. 84–91.

9 Transitioning to community-based mental health service delivery: opportunities for Ukraine [Text] / [E. Quirke, O. Suvalo, O. Sukhovii et al.] // Journal of Market Access & Health Policy. – 2020. – Vol. 8(1). – P. 1-11.

10 Preventive psychiatry: a blueprint for improving the mental health of young people [Text] / [P. Fusar-Poli, C. U. Correll, C. Arango et al.] // World Psychiatry. – 2021. – Vol. 20(2). – P. 200–221.

11 Lifetime and 12-month treatment for mental disorders and suicidal thoughts and behaviors among first year college students [Text] / [R. Bruffaerts, P. Mortier, R. P. Auerbach et al.] // International Journal of Methods in Psychiatric Research. – 2019. – Vol. 28(2). – P. 17-64.

12 WHO world mental health surveys International college student project: prevalence and distribution of mental disorders. [Text] / [R. P. Auerbach, P. Mortier, R. Bruffaerts et al.] // Journal of Abnormal Psychology. – 2018. – Vol. 127(7). – P. 623–638.

13 Virtual reality: another world within sight [Electronic source] – Access mode: <https://www.iberdrola.com/innovation/virtual-reality>.

14 VR в ресторанной индустрии [Электрон. ресурс]. – Режим доступа: www.buro247.kz/lifestyle/technology/ar-vr-tekhnologii-v-restorannoy-industrii.html.

15 Virtual reality: [Electronic source] // iberdrola.com – Access mode: https://www.iberdrola.com/documents/20125/40414/Infographic_innovative_uses_virtual_reality.pdf/66a81753-51bd-0e72-2a6d-f5aaea023e03?t=1627369981610.

16 CNN VR [Electronic source]. – Access mode: <https://edition.cnn.com/vr>.

17 Engage: virtual communication made real [Electronic source]. – Access mode: <https://immersivevreducation.com/>.

18 Виртуальная реальность в Steam [Электрон. ресурс]. – Режим доступа: <https://store.steampowered.com/vr/>.

19 Architectural virtual reality [Electronic source]. – Access mode: <https://enscape3d.com/features/architectural-virtual-reality/>.

- 20 VR/AR and 3d modeling [Electronic source]. – Access mode: <https://www.nixsolutions.com/cutting-edge/vrar-development-3d-modeling/>.
- 21 10 amazing virtual museum tours [Electronic source]. – Access mode: <https://www.virtualiteach.com/post/2017/08/20/10-amazing-virtual-museum-tours>.
- 22 Virtual reality in the military [Electronic source]. – Access mode: <https://www.vrs.org.uk/virtual-reality-military/>.
- 23 Defy reality® – yhe 3d commerce blog [Electronic source]. – Access mode: <https://www.marxentlabs.com/defy-reality-the-3d-commerce-blog/>.
- 24 Vive business [Electronic source]. – Access mode: <https://business.vive.com/ua/product/vive-pro-eye/>.
- 25 Quest 2 [Electronic source]. – Access mode: <https://www.oculus.com/quest-2/>.
- 26 PlayStation VR [Electronic source]. – Access mode: <https://inlnk.ru/G6K7O>.
- 27 What is virtual reality? [Electronic source]. – Access mode: <https://www.marxentlabs.com/what-is-virtual-reality/>.
- 28 Treating psychological and phsyical disorders with VR [Text] / [L. F. Hodges, P. Anderson, G. C. Burdea et al.] // IEEE Computer Graphics and Applications. – 2001. – Vol. 21(6). – P. 25–33.
- 29 Riva, G. Neuroscience of virtual reality: from virtual exposure to embodied medicine [Text] / G. Riva, B. K. Wiederhold, F. Mantovani // Cyberpsychology, Behavior, and Social Networking. – 2019. – Vol. 22(1). – P. 82–96.
- 30 Transforming experience: the potential of augmented reality and virtual reality for enhancing personal and clinical change [Text] / [G. Riva, R. M. Baños, C. Botella et al.] // Frontiers in Psychiatry. – 2016. – Vol. 7. – P. 1-11.
- 31 Freeman, D. 7 Virtual reality (VR) for the treatment of mental health disorders [Text] / D. Freeman // Journal of Neurology, Neurosurgery & Psychiatry. – 2020. – Vol. 91(8). – P. 32-33.

32 Riva, G. Letter to the editor: virtual reality in the treatment of eating and weight disorders [Text] / G. Riva // *Psychological Medicine*. – 2017. – Vol. 47(14). – P. 2567–2568.

33 Review of virtual reality treatment in psychiatry: evidence versus current diffusion and use [Text] / [M. C. Mishkind, A. M. Norr, A. C. Katz et al.] // *Current Psychiatry Reports*. – 2017. – Vol. 19(11). – P. 1-10.

34 Van, Bennekom M. J. Virtual reality objectifies the diagnosis of psychiatric disorders: a literature review [Text] / M. J. van Bennekom, P. P. de Koning, D. Denys // *Frontiers in Psychiatry*. – 2017. – Vol. 8. – P. 1-25.

35 Head-mounted virtual reality and mental health: critical review of current research [Text] / [S. W. Jerdan, M. Grindle, H. C. van Woerden et al.] // *JMIR Serious Games*. – 2018. – Vol. 6(3). – P. 14.

36 Recent progress in virtual reality exposure therapy for phobias: a systematic review [Text] / [C. Botella, J. Fernández-Álvarez, V. Guillén et al.] // *Current Psychiatry Reports*. – 2017. – Vol. 19(7). – P. 1-13.

37 Systematic review of randomized controlled trials on the effectiveness of virtual reality training for laparoscopic surgery [Text] / [K. Gurusamy, R. Aggarwal, L. Palanivelu et al.] // *British Journal of Surgery*. – 2008. – Vol. 95(9). – P. 1088–1097.

38 Clinical efficacy of virtual reality for acute procedural pain management: A systematic review and meta-analysis [Text] / [E. Chan, S. Foster, R. Sambell et al.] // *PLOS ONE*. – 2018. – Vol. 13(7). – P. – 1-10.

39 Application of virtual reality technology in disaster medicine [Text] / [Y. Duan, J. Zhang, M. Xie et al.] // *Current Medical Science*. – 2019. – Vol. 39(5). – P. 690–693.

40 Georgieva, I. Reconstructing personal stories in virtual reality as a mechanism to recover the self [Text] / I. Georgieva, G. V. Georgiev // *International Journal of Environmental Research and Public Health*. – 2019. – Vol. 17(1). – P. 26.

41 Virtual reality [Electronic source]. – Access mode: <https://www.iberdrola.com/top-stories/technology/what-is-augmented-reality>.

42 Evidence on virtual reality–based therapies for psychiatric disorders: meta-review of meta-analyses [Text] / [L. Dellazizzo et al.] // Journal of Medical Internet Research. – 2020. – Vol. 22(8). – P. – 1-10.

43 Embodied medicine: mens sana in corpore virtuale sano [Text] / [G. Riva et al.] // Frontiers in Human Neuroscience. – 2017. – Vol. 11. – P. 1-10.

44 Facebook’s new academic research API opens in early access [Electronic source]. – Access mode: <https://techcrunch.com/2021/11/15/facebooks-researcher-api-meta-academic-research>.

45 Facebook changes name to Meta as it refocuses on virtual reality [Electronic source]. – Access mode: <https://www.reuters.com/technology/facebooks-zuckerberg-kicks-off-its-virtual-reality-event-with-metaverse-vision-2021-10-28/>.

46 Facebook data for independent research [Electronic source]. – Access mode: <https://research.fb.com/data/>.

47 Connection is evolving and so are we. [Electronic source]. – Access mode: <https://about.facebook.com/meta/>.

48 Introducing horizon workrooms: remote collaboration reimagined [Electronic source]. – Access mode: <https://about.fb.com/news/2021/08/introducing-horizon-workrooms-remote-collaboration-reimagined/>.

49 Therapists make the switch to telepsychology to safely continue treating their patients during the COVID-19 pandemic. Virtual reality telepsychology may be next [Text] / [M. Sampaio, M. V. Navarro Haro, B. De Sousa et al.] // Frontiers in Virtual Reality. – 2021. – Vol. 1. – P. 1-10.

50 Toward establishing telepsychology guideline. Turning the challenges of COVID-19 into opportunity [Text] / [M. M. J. Alqahtani, H. A. Alkhamees, A. M. Alkhalaf et al.] // Ethics, Medicine and Public Health. – 2021. – Vol. 16. – P. 100-612.

51 Peterson, S. M. Transient visual perturbations boost short-term balance learning in virtual reality by modulating electrocortical activity [Text] / S. M. Peterson, E. Rios, D. P. Ferris // *Journal of Neurophysiology*. – 2018. – Vol. 120(4). – P. 1998–2010.

52 Laver, K. Virtual reality for stroke rehabilitation [Text] / K. Laver // *Virtual Reality in Health and Rehabilitation*. – 2020. – P. 19–28.

53 Virtual reality hand therapy: A new tool for nonopioid analgesia for acute procedural pain, hand rehabilitation, and VR embodiment therapy for phantom limb pain [Text] / [H. G. Hoffman, D. A. Boe, E. Rombokas et al.] // *Journal of Hand Therapy*. – 2020. – Vol. 33(2). – P. 254–262.

54 New hand gesture technology could wave goodbye to passwords [Electronic source]. – Access mode: <https://www.forbes.com/sites/daveywinder/2019/09/09/exclusive-new-hand-gesture-technology-could-wave-goodbye-to-passwords/?sh=9f81b4d5286d>.

55 XTAL [Electronic source]. – Access mode: <https://vrgineers.com/xtal/>.

56 Epidemiology of specific phobia subtypes: Findings from the Dresden Mental Health Study [Text] / [E. S. Becker, M. Rinck, V. Turke et al.] // *European Psychiatry*. – 2007. – Vol. 22(2). – P. 69–74.

57 Augmented reality: a brand new challenge for the assessment and treatment of psychological disorders [Text] / [I. A. Chicchi Giglioli, F. Pallavicini, E. Pedrolì et al.] // *Computational and Mathematical Methods in Medicine*. – 2015. – Vol. 2015. – P. 1–12.

58 The use of virtual reality technology in the treatment of anxiety and other psychiatric disorders [Text] / [J.L. Maples-Keller, B. E. Bunnell, S. J. Kim et al.] // *Harvard Review of Psychiatry*. – 2017. – Vol. 25(3). – P. 103–113.

59 Virtual reality exposure therapy in anxiety disorders: a quantitative meta-analysis [Text] / [D. Opreș, S. Pinteș, A. García-Palacios et al.] // *Depression and Anxiety*. – 2011. – Vol. 29(2). – P. 85–93.

- 60 Proof of concept in app development: poc definition & development stages [Electronic source]. – Access mode: <https://fulcrum.rocks/blog/proof-of-concept/>.
- 61 Prototyping model [Electronic source]. – Access mode: <https://searchcio.techtarget.com/definition/Prototyping-Model>.
- 62 Minimum viable product (MVP) [Electronic source]. – Access mode: <https://www.techopedia.com/definition/27809/minimum-viable-product-mvp>.
- 63 The Model View Controller pattern – MVC architecture and frameworks explained [Electronic source]. – Access mode: <https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained>.
- 64 Arcos-Medina, G. Comparative study of performance and productivity of MVC and MVVM design patterns [Text] / G. Arcos-Medina, J. Menéndez, J. Vallejo // KnE Engineering. – 2018. – Vol. 1(2). – P. 241.
- 65 MVP architectural pattern [Electronic source]. – Access mode: <https://ducmanhphan.github.io/2019-08-05-MVP-architectural-pattern/>.
- 66 What is conatinerization [Electronic source]. – Access mode: <https://searchitoperations.techtarget.com/definition/application-containerization-app-containerization/>.
- 67 Docker [Electronic source]. – Access mode: <https://www.docker.com/>.
- 68 Top 10 best container software in 2021 [Electronic source]. – Access mode: <https://www.softwaretestinghelp.com/container-software>.
- 69 Kubernetes [Electronic source]. – Access mode: <https://kubernetes.io/>.
- 70 Top 5 software development methodologies in 2021 [Electronic source]. – Access mode: <https://www.bugreporting.co/blog/top-5-software-development-methodologies-in-2021>.
- 71 Software development methodologies: the top 10 explained [Electronic source]. – Access mode: <https://www.designveloper.com/blog/software-development-methodologies>.

72 What is a database? [Electronic source]. – Access mode: <https://www.oracle.com/database/what-is-database/>.

73 SQL vs NoSQL: 5 critical differences [Electronic source]. – Access mode: <https://www.xplenty.com/blog/the-sql-vs-nosql-difference>.

74 Top 10 databases to use in 2021 [Electronic source]. – Access mode: <https://towardsdatascience.com/top-10-databases-to-use-in-2021-d7e6a85402ba>.

75 Most popular programming languages in 2021 [Electronic source]. – Access mode: <https://www.stackscale.com/blog/popular-programming-languages-2021>.

76 Java [Electronic source]. – Access mode: <https://www.java.com/en/>.

77 Flask vs Django [Electronic source]. – Access mode: <https://coderlessons.com/tutorials/kompiuternoe-programmirovaniye/uchebnik-python/62-nastoi-protiv-dzhango>.

78 Flask vs Django in 2021: Which Framework to Choose? [Electronic source]. – Access mode: <https://hackr.io/blog/flask-vs-django>.

79 Django vs Flask in 2021: Battle of Backend Web Development [Electronic source]. – Access mode: <https://citrusbug.com/blog/django-vs-flask>.

80 Introduction to Unity3D [Electronic source]. – Access mode: <https://code.tutsplus.com/tutorials/introduction-to-unity3d--mobile-10752>.

81 Blueprint visual scripting [Electronic source]. – Access mode: <https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/Blueprints>.

82 Marketplace [Electronic source]. – Access mode: <https://www.unrealengine.com/marketplace/en-US/assets?count=20&keywords=vr&sortBy=relevancy&sortDir=DESC&start=0>.

ДОДАТОК А
Текст програми

A.1 backend/backend/settings.py

```

import os
from datetime import timedelta
from pathlib import Path

from utils.env_utils import load_env

BASE_DIR = Path(__file__).resolve().parent.parent

load_env()
ENV = os.getenv('environment_type')

SECRET_KEY = os.getenv('DJANGO_SECRET_KEY')
ALLOWED_HOSTS = os.getenv('DJANGO_ALLOWED_HOSTS')
SERVER_HOST = os.getenv('SERVER_HOST')
DJANGO_ADMIN_PAGE = os.getenv('DJANGO_ADMIN_PAGE')

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'rest_framework',
    'rest_framework_simplejwt.token_blacklist',
    'drf_yasg2',
    'storages',

    'user.apps.UserConfig',
    'rest_api.apps.RestApiConfig',
    'disorder.apps.DisorderConfig',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

```

```

REST_FRAMEWORK = {
    'DEFAULT_FILTER_BACKENDS':
    ['django_filters.rest_framework.DjangoFilterBackend'],
    'DEFAULT_AUTHENTICATION_CLASSES': (

'rest_framework_simplejwt.authentication.JWTAuthentication',
    ),
    'DEFAULT_PERMISSION_CLASSES': [
        'rest_framework.permissions.IsAuthenticated',
    ],
    'DEFAULT_PAGINATION_CLASS':
'rest_framework.pagination.PageNumberPagination',
    'PAGE_SIZE': 50,
}

ROOT_URLCONF = 'backend.urls'

TEMPLATES = [
    {
        'BACKEND':
'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',

'django.contrib.messages.context_processors.messages',
            ],
        },
    ],
]

WSGI_APPLICATION = 'backend.wsgi.application'

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': os.getenv('POSTGRES_DB'),
        'USER': os.getenv('POSTGRES_USER'),
        'PASSWORD': os.getenv('POSTGRES_PASSWORD'),
        'HOST': os.getenv('DATABASE_HOST'),
        'PORT': os.getenv('DATABASE_PORT'),
    }
}

```

```

AUTH_USER_MODEL = 'user.User'

SIMPLE_JWT = {
    'ACCESS_TOKEN_LIFETIME': timedelta(days=31),
    'REFRESH_TOKEN_LIFETIME': timedelta(days=365),
    'ROTATE_REFRESH_TOKENS': True,
}

SWAGGER_SETTINGS = {
    'SECURITY_DEFINITIONS': {
        'api_key': {
            'type': 'apiKey',
            'in': 'header',
            'name': 'Authorization'
        }
    }
}

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
        'django.contrib.auth.password_validation.UserAttributeSimilarityVal
        idator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

LOGGING = {
    'version': 1,
    'disable_existing_loggers': False,
    'formatters': {
        'main_formatter': {
            'format': '%(asctime)s %(name)s
            [% (filename)s: %(lineno)d] %(levelname)-8s %(message)s'
        },
    },
}

```

```

    'handlers': {
        'main_handler': {
            'class': 'logging.StreamHandler',
            'formatter': 'main_formatter'
        },
    },
    'loggers': {
        'main_logger': {
            'level': 'INFO',
            'handlers': ['main_handler']
        },
        'error_logger': {
            'level': 'ERROR',
            'handlers': ['main_handler']
        }
    }
}

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

STATIC_URL = '/staticfiles/'
STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')

```

A.2 backend/backend/urls.py

```

from django.contrib import admin
from django.urls import path
from django.urls import include
from drf_yasg2 import openapi
from drf_yasg2.views import get_schema_view
from rest_framework import permissions

SchemaView = get_schema_view(
    openapi.Info(
        title="diploma API",
        default_version='v1',
    ),

```

```

        public=True,
        permission_classes=(permissions.AllowAny,),
    )

urlpatterns = [
    path('admin/', admin.site.urls),
    path('swagger/', SchemaView.with_ui('swagger',
cache_timeout=0), name='schema-swagger-ui'),
    path('api/v1/', include('main_api.urls'))
]

```

A.3 backend/disorder/models.py

```

from django.db import models

class Disorder(models.Model):
    name = models.CharField('name', max_length=50, blank=False,
null=False, default='name')
    classification = models.CharField('classification',
max_length=80, blank=False, null=False, default='classification')
    description = models.TextField('description', max_length=50,
blank=False, null=False, default='name')

    class Meta:
        db_table = 'disorder'

class DisorderScenario(models.Model):
    name = models.CharField('name', max_length=50, blank=False,
null=False, default='name')
    classification = models.CharField('classification',
max_length=80, blank=False, null=False, default='classification')
    description = models.TextField('description', max_length=50,
blank=False, null=False, default='name')
    disorder = models.ForeignKey(Disorder, blank=True, null=True,
on_delete=models.CASCADE)

    class Meta:
        db_table = 'disorder_scenario'

```

A.4 backend/main_api/constants.py

```

ERROR_RESPONSE_EXAMPLE = '{"message': 'error', 'payload':
[{'field': '...', " \
                                "'data': [{'description': '...', 'code':
'...'}]}]}"
SUCCESS_RESPONSE_EXAMPLE = '{"message': 'ok', 'payload': null} or
{'message': '...', 'payload': [{'field': ['...']}]}"

```

A.5 backend/main_api/permissions.py

```
# pylint: disable=R0901, E1131
from rest_framework.permissions import AllowAny, IsAdminUser,
IsAuthenticated
from rest_framework.viewsets import ModelViewSet

class PermissionsMixin(ModelViewSet):

    permission_classes_by_action = {
        "default": [AllowAny | IsAdminUser],
        "destroy": [IsAdminUser],
        "perform_create": [IsAdminUser]
    }

    def get_permissions(self):
        try:
            # return permission_classes depending on `action`
            return [
                permission()
                for permission in
self.permission_classes_by_action[self.action]
            ]
        except KeyError:
            # action is not set return default permission_classes
            return [
                permission()
                for permission in
self.permission_classes_by_action["default"]
            ]

class PermissionsPostsMixin(PermissionsMixin):

    permission_classes_by_action = {
        "default": [AllowAny | IsAdminUser],
        "partial_update": [IsAuthenticated],
        "destroy": [IsAuthenticated],
        "perform_create": [IsAuthenticated]
    }

    }

class PermissionsUsersMixin(PermissionsMixin):

    permission_classes_by_action = {
        "default": [AllowAny | IsAdminUser],
        "partial_update": [IsAuthenticated],
```

```

        "destroy": [IsAdminUser],
        "perform_create": [IsAdminUser]

    }

```

```

class IsSuperUser(IsAdminUser):
    def has_permission(self, request, view):
        return bool(request.user and request.user.is_superuser)

```

A.6 backend/main_api/urls.py

```

from django.urls import path
from rest_framework_simplejwt import views as jwt_views

from main_api.views import SignUpAdminView, SignUpView,
PatientDetailView

urlpatterns = [
    path('signup/', SignUpView.as_view(), name='signup'),
    path('signup/admin/', SignUpAdminView.as_view(),
name='signup_admin'),
    path('login/', jwt_views.TokenObtainPairView.as_view(),
name='token_obtain_pair'),
    path('login/refresh/', jwt_views.TokenRefreshView.as_view(),
name='token_refresh'),
    path('users/me/', PatientDetailView.as_view({'get':
'retrieve'}), name='users_info'),
    path('disorders_scenario/',
DisorderScenarioView.as_view({'get': 'retrieve'}),
name='scenario_info'),
]

```

A.7 backend/main_api/urls.py

```

from django.contrib import admin

from disorder.models import DisorderScenario, Disorder
from user.models import User, PatientUser, DoctorUser

@admin.register(DoctorUser)
class UserAdmin(admin.ModelAdmin):
    fields = ('first_name', 'last_name', 'third name',
              'telephone_number', 'birthday', 'position',
              'speciality'
              )

```

```
@admin.register(PatientUser)
class PatientAdmin(admin.ModelAdmin):
    fields = ('first_name', 'last_name', 'third_name',
              'telephone_number', 'birthday', 'doctor',
              'disorder', 'health', 'gender', 'password'
              )
    readonly_fields = ('created_at', 'updated_at')

admin.site.register(Disorder)
admin.site.register(DisorderScenario)
```

A.8 backend/user/choices.py

```
PATIENT = 'PATIENT'
DOCTOR = 'DOCTOR'
ADMIN = 'ADMIN'

MALE = 'MALE'
FEMALE = 'FEMALE'
OTHER = 'OTHER'

USER_ROLES = ((PATIENT, PATIENT),
               (DOCTOR, DOCTOR),
               (ADMIN, ADMIN),)

GENDER = ((MALE, MALE),
           (FEMALE, FEMALE),
           (OTHER, OTHER),)
```

A.9 backend/user/models.py

```
from django.contrib.auth.models import AbstractUser
from django.db import models
from django.utils import timezone

from disorder.models import Disorder
from user.choices import ADMIN, PATIENT, GENDER, OTHER, USER_ROLES

class User(AbstractUser):
    first_name = models.CharField('first name', max_length=50,
                                   blank=False, null=False, default='username')
    last_name = models.CharField('last name', max_length=50,
                                  blank=False, null=False, default='lastname')
    third_name = models.CharField('third name', max_length=50,
                                   blank=False, null=False, default='thirdname')
    birthday = models.DateTimeField(auto_now_add=False, blank=True,
                                     null=True)
```

```

telephone_number = models.CharField(max_length=13, blank=False,
null=False, default='phone_number')
photo = models.ImageField(default="", null=True, blank=True,
upload_to='src/images')

```

```

user_type = models.CharField(max_length=11,
                             choices=USER_ROLES,
                             default=PATIENT)

```

```

class DoctorUser(User):
    position = models.CharField('position', max_length=50,
blank=False, null=False, default='position')
    speciality = models.CharField('speciality', max_length=50,
blank=False, null=False, default='speciality')

```

```

class Meta:
    db_table = 'doctor'

def __init__(self, *args, **kwargs):
    super().__init__(*args, **kwargs)
    self.__original_name = self.username

def save(self, *args, **kwargs):
    self.username = self.last_name
    super().save(*args, **kwargs)

```

```

class PatientUser(User):
    created_at = models.DateTimeField(auto_now_add=True,
blank=True, null=True)
    updated_at = models.DateTimeField(auto_now=True, blank=True,
null=True)
    doctor = models.ForeignKey(DoctorUser, blank=True, null=True,
on_delete=models.CASCADE)
    disorder = models.ForeignKey(Disorder, blank=True, null=True,
on_delete=models.CASCADE)
    health = models.CharField(max_length=150, blank=False,
null=False)
    gender = models.CharField(max_length=6, choices=GENDER,
default=OTHER)

```

```

class Meta:
    db_table = 'patient'

def __init__(self, *args, **kwargs):
    super().__init__(*args, **kwargs)
    self.__original_name = self.username

```

```
def save(self, *args, **kwargs):
    self.username = self.last_name
    super().save(*args, **kwargs)
```

A.10 backend/user/choices.py

```
import os
from pathlib import Path

from dotenv import load_dotenv

def load_env():
    project_root = Path(__file__).parent.parent
    if os.getenv('environment_type') == 'local':
        env_file_path = "env/.env.local"
    elif os.getenv('environment_type') == 'test':
        env_file_path = "env/.env.test"
    else:
        env_file_path = "env/.env.example"

    dotenv_path = os.path.join(project_root, env_file_path)

    if os.path.exists(dotenv_path):
        load_dotenv(dotenv_path)
    else:
        raise FileNotFoundError(f'Env file not found:
{dotenv_path}')
```

A.11 backend/docker-compose.yml

```
version: "3.9"

services:
  db:
    image: postgres:13
    env_file:
      - env/.env.local
    volumes:
      - ./data/db:/var/lib/postgresql/data
    ports:
      - 5432:5432

  web:
    build: .
    environment:
      - environment_type=local
      - PYTHONUNBUFFERED=1
    command: >
```

```

    bash -c "pip install -r requirements.txt
    && python manage.py makemigrations
    && python manage.py migrate --run-syncdb
    && python manage.py initadmin
    && python manage.py runserver 0.0.0.0:8000"
volumes:
  - ./app
ports:
  - "8000:8000"
depends_on:
  - db

```

A.12 backend/Dockerfile

```

FROM python:3
WORKDIR /app
COPY requirements.txt /app/
RUN pip install -r requirements.txt
COPY . /app/

```

A.13 backend/Makefile

```

current_dir = $(shell pwd)

start_backend:
    docker-compose -f docker-compose.yml up ;

stop_backend:
    docker-compose -f docker-compose.yml stop;

build_backend:
    docker-compose -f docker-compose.yml build;

down_backend:
    docker-compose -f docker-compose.yml down;

run_test:
    docker-compose -f docker-compose-test.yml up --build --abort-
on-container-exit;

```

A.13 backend/manage.py

```

#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys

```

```

def main():
    """Run administrative tasks."""
    os.environ.setdefault('DJANGO_SETTINGS_MODULE',
'backend.settings')
    try:
        from django.core.management import
execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed
and "
            "available on your PYTHONPATH environment variable? Did
you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)

if __name__ == '__main__':
    main()

```

A.14 backend/pytest.ini

```

[pytest]
DJANGO_SETTINGS_MODULE = backend.settings
python_files = tests.py test_*.py *_tests.py
addopts = --create-db -p no:warnings

```

A.15 backend/requirements.txt

```

django==3.2.6
django-rest-framework==3.12.4
drf-yasg2==1.19.4
django-filter==21.1
python-dotenv==0.19.1
psycopg2-binary==2.9.1
django-rest-framework-simplejwt==4.6.0
Pillow==8.4.0
boto3==1.20.3
django-storages==1.12.3

pytest==6.2.3
pytest-django==4.2.0
pylint==2.7.4

```

A.16 backend/user/management/initadmin.py

```
import os

from django.contrib.auth import get_user_model
from django.core.management.base import BaseCommand
from django.db import transaction
UserModel = get_user_model()

class Command(BaseCommand):
    @transaction.atomic
    def handle(self, *args, **options):
        django_username = os.getenv('DJANGO_ADMIN_USER')
        django_password = os.getenv('DJANGO_ADMIN_PASSWORD')
        django_email = f'{django_username}@admin.com'

        if
UserModel.objects.filter(username=django_username).count() == 0:
            admin = UserModel.objects.create_superuser(
                email=django_email, username=django_username,
password=django_password, is_active=True,
            )
            admin.set_password(django_password)
            admin.is_admin = True
            admin.save()
```

A.17 vr_app/config/DefaultEngine.ini

```
[/Script/HardwareTargeting.HardwareTargetingSettings]
; quality settings
TargetedHardwareClass=Desktop
AppliedTargetedHardwareClass=Desktop
DefaultGraphicsPerformance=Maximum
AppliedDefaultGraphicsPerformance=Maximum

[/Script/WindowsTargetPlatform.WindowsTargetSettings]
;Ndisplay Requirements
DefaultGraphicsRHI=DefaultGraphicsRHI_DX11
+TargetedRHIs=PCD3D_SM5
-TargetedRHIs=PCD3D_SM4
-TargetedRHIs=GLSL_430
-TargetedRHIs=GLSL_150

[/Script/Engine.RendererSettings]
; exposure control and tone mapping settings
r.Tonemapper.Sharpen=2
r.DefaultFeature.AutoExposure.ExtendDefaultLuminanceRange=True
r.DefaultFeature.Bloom=False
r.DefaultFeature.AutoExposure=False
```

```

r.DefaultFeature.MotionBlur=False
r.SceneRenderTargetResizeMethod=2

;dynamic gi settings
r.LightPropagationVolume=0

; quality settings
r.ReflectionCaptureResolution=2048
r.AllowStaticLighting=True
r.HighResScreenshotDelay=8
r.DefaultBackBufferPixelFormat=4
r.AllowGlobalClipPlane=False
r.GBufferFormat=3

; shader settings
r.PostProcessing.PropagateAlpha=1
r.SupportSkyAtmosphereAffectsHeightFog=True
r.DefaultFeature.LensFlare=False
r.ClearCoatNormal=False
r.NormalMapsForStaticLighting=False

[/Script/EngineSettings.GameMapsSettings]
;defines what level to be loaded by default in the editor and at
playback time
GameDefaultMap=/Game/VProdProject/Maps/Ndisplay.Ndisplay
EditorStartupMap=/Game/VProdProject/Maps/Main.Main

[/Script/UnrealEd.UnrealEdEngine]
; Remove Engine Template maps
-
TemplateMapInfos=(ThumbnailTexture=Texture2D'/Engine/Maps/Templates
/Thumbnails/VR-Basic.VR-Basic',Map="/Engine/Maps/Templates/VR-
Basic")

;Allows for Hardware Accelerated Video Decoding
[/Script/WmfMediaFactory.WmfMediaSettings]
AllowNonStandardCodecs=True
HardwareAcceleratedVideoDecoding=True
LowLatency=False
NativeAudioOut=False

;Adds Virtual Scouting Widget for VR Scouting
[/Script/VPUtilitiesEditor.VPUtilitiesEditorSettings]
VirtualScoutingUI=/VirtualProductionUtilities/Editor/VirtualScoutin
gWidget.VirtualScoutingWidget_C

; Enables WebControl API
WebControl.EnableServerOnStartup=1

```

```

;Adds Remote Session for Vcam
[RemoteSession]
+Channels=(Name=FRemoteSessionFrameBufferChannel,Mode=Write)
+Channels=(Name=FRemoteSessionInputChannel,Mode=Read)
+Channels=(Name=FRemoteSessionXRTrackingChannel,Mode=Read)

;Setup for Media Profiles
[/Script/MediaFrameworkUtilities.MediaProfileSettings]
bApplyInCommandlet=False
MediaSourceProxy=/Game/VprodProject/MediaIO/MediaSource-01.MediaSource-01
MediaSourceProxy=/Game/VprodProject/MediaIO/MediaSource-02.MediaSource-02
MediaOutputProxy=/Game/VprodProject/MediaIO/MediaOutput-01.MediaOutput-01
StartupMediaProfile=None

;Setup for Multiuser
[/Script/Concert.ConcertClientConfig]
bIsHeadless=False
bInstallEditorToolbarButton=True
bAutoConnect=False
DefaultServerURL=
DefaultSessionName=
DefaultSessionToRestore=
DefaultSaveSessionAs=
ClientSettings=(DisplayName="",AvatarColor=(R=1.000000,G=1.000000,B=1.000000,A=1.000000),DesktopAvatarActorClass=/ConcertSyncClient/DesktopPresence.DesktopPresence_C,VRAvatarActorClass=/ConcertSyncClient/VRPresence.VRPresence_C,DiscoveryTimeoutSeconds=5,SessionTickFrequencySeconds=1,LatencyCompensationMs=0.000000,Tags=)
EndpointSettings=(bEnableLogging=False,PurgeProcessedMessageDelaySeconds=30,RemoteEndpointTimeoutSeconds=60)

;Setup for Ndisplay Cluster
[/Script/DisplayClusterEditor.DisplayClusterEditorSettings]
bEnabled=True

[/Script/WindowsTargetPlatform.WindowsTargetSettings]
;Ndisplay Requirements
DefaultGraphicsRHI=DefaultGraphicsRHI_DX11
+TargetedRHIs=PCD3D_SM5
-TargetedRHIs=PCD3D_SM4
-TargetedRHIs=GLSL_430
-TargetedRHIs=GLSL_150

[/Script/Engine.Engine]
+ActiveGameNameRedirects=(OldGameName="TP_ME_VprodBP",NewGameName="/Script/vr_app")

```

```
+ActiveGameNameRedirects=(OldGameName="/Script/TP_ME_VprodBP",NewGameName="/Script/vr_app")
```

A.18 vr_app/config/DefaultVirtualProductionUtilites.ini

```
[/Script/VPUutilitiesEditor.VPUutilitiesEditorSettings]
VirtualScoutingUI=/VirtualProductionUtilities/Editor/VirtualScoutingWidget.VirtualScoutingWidget_C
FlightSpeed=0.500000
GripNavSpeed=0.250000
bUseMetric=False
bUseTransformGizmo=False
bUseGripInertiaDamping=True
InertiaDamping=0.950000
bIsHelperSystemEnabled=True
ScoutingSubsystemEditorUtilityActorClassPath=/VirtualProductionUtilities/VirtualProductionHelpers.VirtualProductionHelpers_C
```

A.19 vr_app/config/WindowsEngine.ini

```
[/Script/Engine.Engine]
GameEngine=/Script/DisplayCluster.DisplayClusterGameEngine
UnrealEdEngine=/Script/DisplayClusterEditor.DisplayClusterEditorEngine
GameViewportClientClassName=/Script/DisplayCluster.DisplayClusterViewportClient
```

A.20 vr_app/config/DefaultEditorSettings.ini

```
[/Script/VREditor.VRModeSettings]
bEnableAutoVREditMode=False
bAutokeySequences=True
InteractorHand=Right
bShowWorldMovementGrid=False
bShowWorldMovementPostProcess=False
bShowWorldScaleProgressBar=True
UIBrightness=1.500000
GizmoScale=0.800000
DoubleClickTime=0.250000
TriggerPressedThreshold_Vive=0.330000
TriggerPressedThreshold_Rift=0.500000
InteractorClass=/VirtualProductionUtilities/VR/VirtualScoutingInteractor.VirtualScoutingInteractor_C
TeleporterClass=BlueprintGeneratedClass'/VirtualProductionUtilities/VR/VirtualScoutingTeleporter.VirtualScoutingTeleporter_C'
bScaleWorldWithDynamicPivot=True
bAllowSimultaneousWorldScalingAndRotation=True
```

ДОДАТОК Б
Слайди презентації

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»

Дипломна кваліфікаційна робота
на тему:

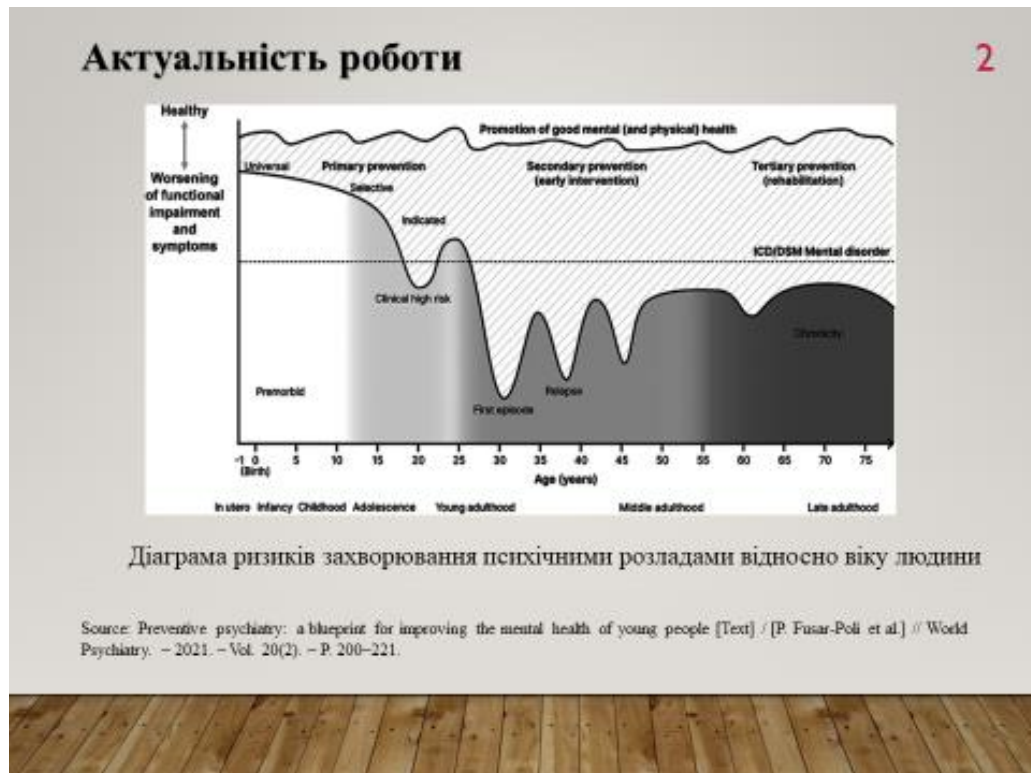
Дослідження та розробка програмного забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності
Research and Development of Software for Treating Mental Disorders via VR Technology

Виконав
ст. гр. КНТ-120м
Керівник
к.т.н. доц.

М. К. МАЛЮКОВ

А. В. Пархоменко

Рисунок Б.1 – Титульный слайд



Діаграма ризиків захворювання психічними розладами відносно віку людини

Source: Preventive psychiatry: a blueprint for improving the mental health of young people [Text] / [P. Fusar-Poli et al.] // World Psychiatry. - 2021. - Vol. 20(2). - P. 200-221.

Актуальність роботи		3
Сфери використання VR технологій	Опис використання	
Продуктова промисловість	Віртуальні магазини та ресторани	
Ліки	Іспанської національної дослідницької ради вдалося зменшити наслідки захворює Паркінсона у кількох пацієнтів, застосувавши лікування, яке використовує VR	
ЗМІ	Імерсивна журналістика переносить користувача в місця, де відбувалися події, за допомогою прямої трансляції 360° відео	
Освіта	У класах використання VR дозволяє студентам краще зберігати знання та допомагає учням із труднощами у навчанні	
Розваги	Користувачі можуть взяти в сцену у відеогри або займатися екстремальними видами спорту, не встаючи з дивана	
Архітектура	RV допомагає архітекторам краще передбачити простір і представити проєкт своїм клієнтам	
Промисловість	Цифрові близнюки — це точні цифрові копії фізичних об'єктів, на яких працівники фабрик можуть практикувати та тестувати їх у віртуальному світі	
Культура	Деякі музеї та галереї пропонують віртуальні відвідування або захоплюючі враження, щоб допомогти зрозуміти історію та культуру пов'язані з кожною роботою	
Військові	Мейстерство оборони Великобританії використовує віртуальну реальність для навчання в імітованих бойових умовах	

Рисунок Б.3 – Актуальність роботи

Мета та завдання роботи	4
Мета роботи – дослідження та практична реалізація програмного забезпечення для лікування психічних захворювань за допомогою технології віртуальної реальності. Завдання роботи: • дослідити існуючі технології VR для лікування пацієнтів; • виконати аналіз вимог до системи; • розробити архітектуру системи та обрати інструментарій розробки; • виконати програмну реалізацію VR застосунку; • розробити керівництво з використання програмного забезпечення.	

Рисунок Б.4 – Мета та завдання роботи



Рисунок Б.7 – Проектування програмного забезпечення

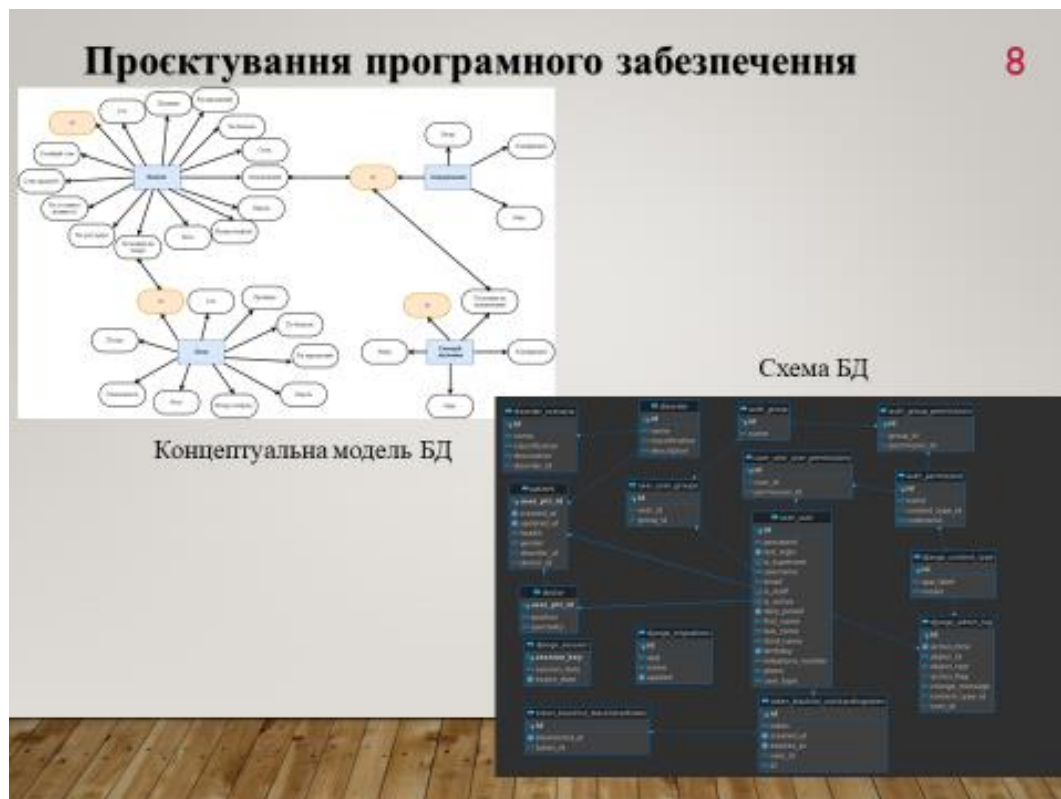


Рисунок Б.8 – Проектування програмного забезпечення



Рисунок Б.9 – Конструювання програмного забезпечення



Рисунок Б.10 – Конструювання програмного забезпечення



Рисунок Б.11 – Конструювання програмного забезпечення

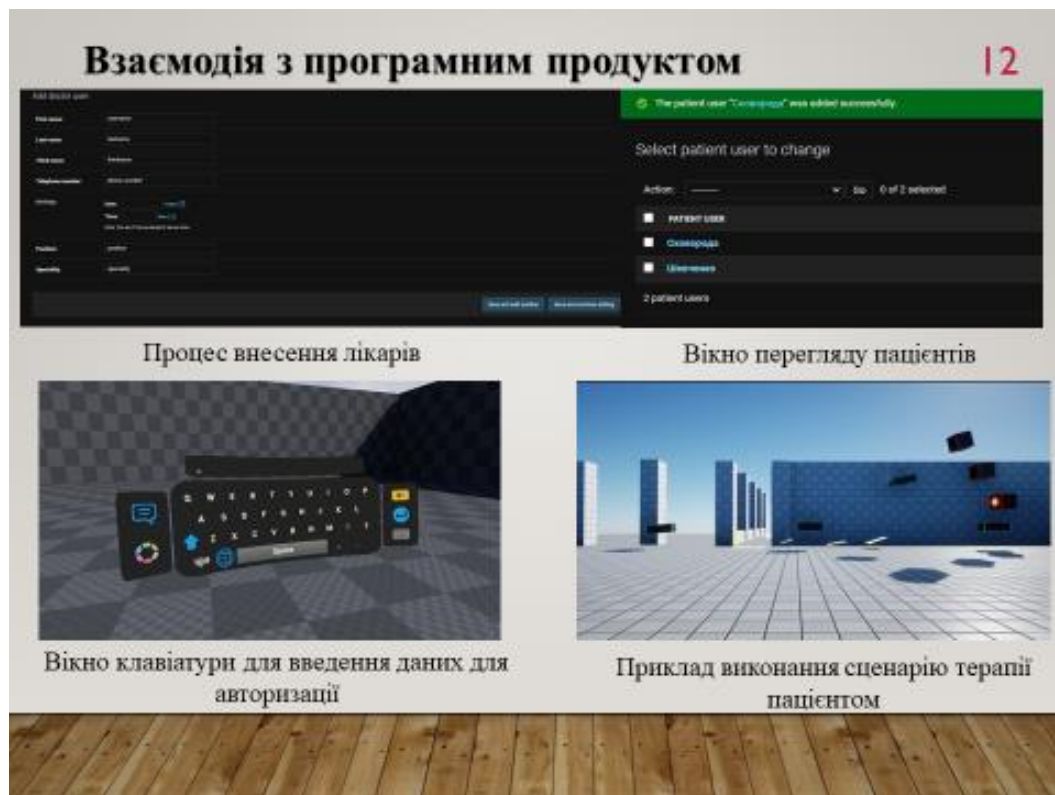


Рисунок Б.12 – Взаємодія з програмним продуктом

Висновки

13

В результаті виконання роботи було розроблено та налаштовано вебзастосунок, який відповідає за збереження даних про користувачів, захворювання та сценарії лікування, а також VR застосунок для використання технології віртуальної реальності для лікування психічних розладів.

Наукова новизна роботи полягає в тому, що отримала подальший розвиток технологія віртуальної реальності для лікування психічних розладів у вигляді платформи для лікування, що об'єднує вебзастосунок та VR застосунок, кожен з яких виконує певні функції.

Практична цінність роботи полягає в тому, що розроблене програмне забезпечення для лікування психічних розладів за допомогою технології віртуальної реальності дозволяє взаємодіяти пацієнтам та лікарям, призначати терапію пацієнтам та створювати сценарії для лікування.

В подальшому розроблене програмне забезпечення можна розширити для терапії захворювань інших розділів медицини.

Результати роботи було представлено на Щорічній науково-практичній конференції НУЗП Тиждень науки – 2021, 19-23 квітня 2021 р.

Рисунок Б.13 – Висновки

14

Дякую за увагу!

Рисунок Б.14 – Фінальний слайд