

БАКАЛАВРСЬКА РОБОТА

ДРБ. ІІ - 38.00.00.000 ІІЗ

Група ІІ-21-2

Дмитрук Владислав

2025

Івано-Франківський національний технічний університет нафти і газу
Інститут інформаційних технологій
Кафедра інженерії програмного забезпечення

Дмитрук Владислав Тарасович

(прізвище, ім'я, по батькові)

УДК 004.942
(індекс)

БАКАЛАВРСЬКА РОБОТА

Програмування віртуальної реальності

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121– Інженерія програмного забезпечення

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня Дмитрук В. Т.
(підпис, ініціали та прізвище здобувача)

Науковий керівник Дмитрик Тарас Богданович, асистент
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту
Завідувач кафедри

доц. Бандура В.В.
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ – 2025

Івано-Франківський національний технічний університет нафти і газу

Інститут, факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти бакалавр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедрою

ПЗ

доцент.

В.В. Бандура

“ ____ ” _____ 2025 р.

ЗАВДАННЯ НА ДИПЛОМНУ РОБОТУ БАКАЛАВРА СТУДЕНТОВІ

Дмитрук Владислав Тарасович

(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи) "Програмування віртуальної реальності"

керівник проекту (роботи) асист. Дмитрик Тарас Богданович

затвержені наказом вищого навчального закладу від “ 28 ” квітня 2025 р. № 264/7

2. Строк подання студентом проекту (роботи) 10 червня 2025 р.

3. Вихідні дані до проекту (роботи) Результати і матеріали отримані під час проходження переддипломної практики

4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)

1 Теоретичні основи навчання програмуванню

2. Vг-гра як засіб навчання об'єктно-орієнтованому програмуванню

3. Дизайн та контекст дослідження

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

1. Огляд нашого дослідницького підходу (рис.1.1, ст.15)

2. Мови програмування, що використовуються у VR-проектах (javaScript - найпопулярніша мова: 137 проектів (рис.1.2, ст.18)

3. Покращена версія Imikode з динамічними рівнями та компонентом штучного інтелекту для кращого розуміння повідомлень про помилки, що генеруються компіляторами ООП (рис. 2.3, ст.30)

4. Абстрактна модель взаємозв'язків між змінними в анкеті USE (рис.3.1, ст.40)

5. Процедура дослідження, що показує час, витрачений на виконання кожної дії (рис.3.5, ст.45)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

2. Дата видачі **2025** р. _____

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Визначення та обґрунтування теми роботи	15.02.2025	виконано
2	Огляд існуючих концепцій, рішень та сервісів в даній області	25.02.2025	виконано
3	Побудова моделі або алгоритму власного рішення	15.03.2025	виконано
4	Документування реалізації власного оригінального рішення вибраними засобами	25.04.2025	виконано
5	Оформлення пояснювальної записки бакалаврської роботи	10.06.2025	виконано

Студент _____

(підпис)

Керівник роботи _____

(підпис)

АНОТАЦІЯ

Бакалаврська робота містить 52 сторінки, 16 рисунків, список використаних джерел із 26 найменування,

Метою роботи є аналіз програмування віртуальної реальності як інструменту для навчання об'єктно-орієнтованому програмуванню, з фокусом на розробку та впровадження VR-гри як засобу освіти

Об'єкт дослідження: Процес навчання об'єктно-орієнтованого програмування у вищій школі.

Предмет дослідження: Використання VR-гри Imikode як засобу формування знань і навичок об'єктно-орієнтованого програмування в студентів.

Результати дослідження: розроблено та апробовано методіку використання VR-гри Imikode як інтерактивного засобу навчання об'єктно-орієнтованому програмуванню.

У першому розділі - розділі розглядаються теоретичні основи навчання програмуванню, включаючи виклики вивчення комп'ютерного програмування, застосування віртуальної реальності в освіті та інструменти візуального програмування

Другий розділ - присвячений VR-грі як засобу навчання об'єктно-орієнтованому програмуванню, з фокусом на функціонування гри, модифікацію навчального середовища та методологію оцінки її впливу.

Третій розділ - аналізує дизайн і контекст дослідження, включаючи вибір інструментів, оцінку надійності методіки, проведення дослідження та аналіз результатів.

Висновок: узагальнено ключові результати та окреслено перспективи розвитку програмування віртуальної реальності в освіті.

КЛЮЧОВІ СЛОВА: ВІРТУАЛЬНА РЕАЛЬНІСТЬ, ПРОГРАМУВАННЯ, ОБ'ЄКТНО-ОРИЄНТОВАНЕ ПРОГРАМУВАННЯ, VR-ГРА, UNITY, ОСВІТА, ВІЗУАЛЬНЕ ПРОГРАМУВАННЯ, ПОРІВНЯЛЬНИЙ

ANNOTATION

The bachelor's thesis contains 52 pages, 16 figures, a list of used sources with 26 names,

The purpose of the work is to analyze virtual reality programming as a tool for teaching object-oriented programming, with a focus on the development and implementation of VR games as a means of education

Object of research: The process of teaching object-oriented programming in higher education.

Subject of research: Using the VR game Imikode as a means of forming knowledge and skills of object-oriented programming in students.

Results of the study: a methodology for using the VR game Imikode as an interactive means of teaching object-oriented programming has been developed and tested.

The first section - the section examines the theoretical foundations of programming education, including the challenges of learning computer programming, the application of virtual reality in education and visual programming tools

The second section - is devoted to VR-game as a means of teaching object-oriented programming, with a focus on the functioning of the game, modification of the learning environment and the methodology for assessing its impact.

The third section - analyzes the design and context of the study, including the choice of tools, assessment of the reliability of the methodology, conducting the study and analyzing the results.

Conclusion: the key results are summarized and the prospects for the development of virtual reality programming in education are outlined.

KEYWORDS: VIRTUAL REALITY, PROGRAMMING, OBJECT-ORIENTED PROGRAMMING, VR GAME, UNITY, EDUCATION, VISUAL PROGRAMMING, COMPARATIVE ANALYSIS, LEARNING ENVIRONMENT, METHODOLOGY.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ НАВЧАННЯ ПРОГРАМУВАННЮ	10
1.1. Виклики вивчення комп'ютерного програмування	10
1.2. Віртуальна реальність для освіти з комп'ютерного програмування	13
1.3. Інструменти візуального програмування	17
1.4 Висновок по розділу.....	22
РОЗДІЛ 2. VR-ГРА ЯК ЗАСІБ НАВЧАННЯ ОБ'ЄКТНО-ОРІЄНТОВАНОМУ ПРОГРАМУВАННЮ	23
2.1. Функціонування гри та підхід до навчання	23
2.2. Модифікація та вдосконалення навчального ігрового середовища	26
2.3. Методологія проєктування та оцінювання навчального впливу	31
2.4 Висновок по розділу.....	37
РОЗДІЛ 3. ДИЗАЙН ТА КОНТЕКСТ ДОСЛІДЖЕННЯ	38
3.1. Вибір інструментів та технічних засобів	38
3.2. Оцінка надійності та валідності методики	40
3.3. Методика проведення дослідження	43
3.4. Аналіз отриманих результатів	46
3.5 Висновок по розділу.....	53
ВИСНОВКИ	54
СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА	57
БІБЛІОГРАФІЧНА ДОВІДКА	

					БР.ІІІ – 38.00.00.000 ПЗ			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розроб.</i>		Дмитрук В. Т.			Програмування віртуальної реальності Пояснювальна записка	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевір.</i>		Дмитрик Т.Б.					8	
<i>Реценз.</i>		Піх М.М.				ІФНТУНГ ІІІ-21-2		
<i>Н. Контр.</i>		Піх М.М.						
<i>Затверд.</i>		Бандура В. В.						

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

VR - віртуальної реальності

AI - штучний інтелект

TPL - текстовими -мовами програмування

VPL - візуальні мови програмування

ООП - об'єктно-орієнтованого програмування

VIF - коефіцієнт інфляції дисперсії

CR - аналіз композитної надійності

AVE - середньої вилученої дисперсії ()

					ДРБ.ІІ - 38.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

ВСТУП

Програмування віртуальної реальності (VR) є передовим напрямом сучасної інформатики, що поєднує розробку програмного забезпечення з іммерсивними технологіями для створення інтерактивних цифрових середовищ. VR-технології, реалізовані за допомогою інструментів, таких як Unity, Unreal Engine, і мов програмування, зокрема C# і Python, знаходять широке застосування в освіті, зокрема для навчання комп'ютерного програмування. Використання VR у навчанні дозволяє створювати гейміфіковані середовища, які сприяють кращому засвоєнню складних концепцій, таких як об'єктно-орієнтоване програмування (ООП), через інтерактивний і мотивуючий досвід.

Актуальність теми зумовлена зростанням попиту на інноваційні методи навчання програмуванню в умовах цифрової трансформації освіти. За даними досліджень, традиційні методи викладання програмування часто стикаються з викликами, такими як складність засвоєння абстрактних концепцій і низька мотивація студентів. VR-технології, завдяки своїй здатності візуалізувати код і створювати ігрові сценарії, пропонують ефективне рішення для подолання цих проблем. Водночас розробка VR-додатків для освіти потребує ретельного вибору інструментів, методології проєктування та оцінки їхнього впливу на навчальний процес.

Метою цього дослідження є аналіз програмування віртуальної реальності як інструменту для навчання об'єктно-орієнтованому програмуванню, з фокусом на розробку та впровадження VR-гри як засобу освіти. Робота спрямована на систематизацію знань про виклики навчання програмуванню, можливості VR-технологій, інструменти програмування та методи оцінки їхньої ефективності. Дослідження охоплює як теоретичні аспекти, так і практичні підходи до створення VR-додатків для освіти.

Ця робота має на меті надати комплексне уявлення про програмування віртуальної реальності як інструмент освіти, а також практичні рекомендації для

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

розробників, викладачів і дослідників, які прагнуть створювати інноваційні навчальні рішення в умовах швидкого технологічного прогресу.

Бакалаврська робота присвячена дослідженню програмування віртуальної реальності (VR) як інструменту для створення інноваційних освітніх середовищ, зокрема для навчання об'єктно-орієнтованому програмуванню (ООП). Метою роботи є аналіз можливостей VR-технологій у підвищенні ефективності навчання програмуванню, розробка VR-гри як навчального засобу та оцінка її впливу на студентів. Дослідження зосереджується на теоретичних основах, методології розробки та емпіричних результатах.

У роботі розглянуто теоретичні основи навчання програмуванню, включаючи виклики вивчення комп'ютерного програмування, застосування VR в освіті та інструменти візуального програмування. Проаналізовано VR-гру як засіб навчання ООП, зокрема її функціонування, модифікацію навчального середовища та методологію оцінки впливу. Окремо досліджено дизайн і контекст дослідження, включаючи вибір інструментів, оцінку надійності методики, проведення експериментів і аналіз результатів.

Практична частина дослідження включає розробку прототипу VR-гри на платформі Unity для навчання ООП, наприклад, моделювання класів і об'єктів у віртуальному середовищі. Оцінюються такі аспекти, як залученість студентів, розуміння концепцій програмування та простота використання. Додатково проведено порівняльний аналіз ефективності VR-навчання порівняно з традиційними методами.

Завданнями дослідження: роаналізувати теоретичні підходи до навчання програмування та особливості його засвоєння студентами. Вивчити потенціал технологій віртуальної реальності в освітньому процесі. Розробити або адаптувати VR-гру для навчання основам ООП. Організувати експериментальне навчання із застосуванням VR-ігри. Провести оцінку ефективності використання гри за допомогою анкетування (інструмент USE) та кількісного аналізу даних. Визначити переваги, недоліки та перспективи впровадження VR-технологій у підготовку програмістів.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

Об'єкт дослідження: Процес навчання об'єктно-орієнтованого програмування у вищій школі.

Предмет дослідження: Використання VR-гри Imikode як засобу формування знань і навичок об'єктно-орієнтованого програмування в студентів.

Методи дослідження: контент-аналіз відкритих відповідей, кількісний аналіз, анкетування, проєктування та модифікація ігрового середовищ, аналіз наукової літератури

Наукова новизна: Вперше у вітчизняному освітньому контексті реалізовано експериментальне дослідження впливу VR-гри Imikode на формування базових знань з об'єктно-орієнтованого програмування.

Бакалаврська робота містить 55 сторінок, 1брисунків, три розділи, список використаних джерел із 30 найменуванням.

					ДРБ.ІІ - 38.00.00.000 ІЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. ВИКОРИСТАННЯ ВІРТУАЛЬНОЇ РЕАЛЬНОСТІ ДЛЯ НАВЧАННЯ КОМП'ЮТЕРНОГО ПРОГРАМУВАННЯ

1.1 Виклики вивчення комп'ютерного програмування

У 2019 році в опитуванні, в якому взяли участь 161 учасник з різних континентів (Африка, Азія, Австралазія, Європа, Північна Америка та Південна Америка) Беннедсеном та Касперсеном [9], рівень невдач на вступних –курсах комп'ютерного програмування становив 28%. Однак автор зазначає, що це число є суб'єктивним для різних університетів чи коледжів, де елітні навчальні заклади вважатимуть це число високим, тоді як інші навчальні заклади можуть вважати його –прийнятним. Беккер та ін. [10] стверджують, що хоча фактор, що сприяє вищезгаданим рівням невдач, є багатограним, корінною проблемою, що охоплює цю проблему, є складність навчання програмуванню, що зрештою –призводить до збільшення рівня відсіву. Цей розділ має на меті висвітлити деякі проблеми, з якими стикаються студенти, починаючи вступний курс програмування. За даними Медейроса та ін. [7], проблеми, з якими стикаються студенти, - це формулювання проблеми, вираження рішення, виконання та оцінка рішення, а також поведінка. Однак, щоб краще розташувати дослідження, буде виділено лише головну проблему з кожної категорії та лише проблеми, безпосередньо пов'язані з формуванням знань з програмування. Отже, основними проблемами, з якими стикаються студенти під час вивчення комп'ютерного програмування, є абстрактний характер програмування, труднощі з синтаксисом програмування та поведінка студентів (мотивація та залученість).

Сьогодні комп'ютерне програмування здебільшого викладається мовами програмування високого рівня, де користувачам потрібно мати справу лише зі змінними, масивами та циклами, на відміну від стеків викликів, реєстрів та адрес пам'яті. Хоча це покращує читабельність та зручність використання коду, користувачі можуть вважати концепції програмування занадто теоретичними,

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

що ускладнює для деяких розуміння основ програмування. Ця проблема також висвітлена в дослідженні Дунікана [11], який запропонував аналогії у введенні концепцій програмування, щоб пов'язати концепцію змінних та типів даних з об'єктами та сценаріями, щоб допомогти студентам у навчанні. Дасукі та Квей [12], які опитали 28 студентів нігерійського університету, виявили, що студенти не можуть зрозуміти, як концепції програмування можуть вирішувати реальні проблеми, що призводить до байдужості до програмування. За даними Жираффи та ін. [13], концепції програмування важко зрозуміти через необхідність уявляти абстрактні терміни, які не відповідають реальним об'єктам. Ця проблема також поширена в літературі Гомеса та Мендеса [14] та Елтегані та Бутгерайта [15]. Отже, незважаючи на те, що масиви є найпростішою структурою даних, їх часто називають потенційно складними для розуміння [16], [17], [18]. Зрештою, це заважає студентам застосовувати цю концепцію під час розробки програм [17]. Отже, важливо зіставити концепції програмування з реальними контекстами, щоб допомогти студентам зрозуміти основи та які типи проблем можна вирішити за допомогою програмування.

Крім того, необхідність одночасного вивчення синтаксису та семантики може бути надзвичайно складною [19]. За даними Giraffa et al. [13], студентам часто важко перетворити розв'язок задачі, написаний псевдокодом, на синтаксично коректну програму. Отже, синтаксичні помилки та допомога студентам у їх виправленні також повинні розглядатися як частина навчального процесу [7]. Koulouri et al. [19] стверджували, що синтаксичні помилки мають значний вплив на програмістів-початківців, оскільки процес їх виявлення та усунення може займати багато часу. Це пояснює, чому синтаксичні помилки широко обговорюються в галузі педагогіки програмування. Qian та Lehman [20] також детально розглянули це у своєму огляді, стверджуючи, що помилкові уявлення можуть виникати через труднощі як у синтаксичних, так і в концептуальних знаннях. За даними Kohn [21], помилкові уявлення про "концепції програмування також спричиняють синтаксичні помилки. Отже, будуть виділені синтаксичні помилки, що виникають через неправильне

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

розуміння концепцій програмування, таких як змінні, цикли, масиви та оператори if.

По-перше, змінні відповідають міткам, які надаються певним коміткам пам'яті, що зберігають дані. Будучи найважливішою концепцією в комп'ютерному програмуванні, нерозуміння концепції змінних призведе до проблем, оскільки будь-яка інша концепція вимагає застосування змінних. Однак найпоширенішою помилкою щодо змінних є те, що студенти вважають, що змінні можуть зберігати більше одного значення. Це видно з ранніх досліджень, проведених Сліманом та ін. [22], Дукакісом та ін. [23], та нещодавнього дослідження Свідана та ін. [24]. По-друге, студенти вважають, що порядок операторів під час присвоєння значення змінній не є важливим. Це також підтверджується ранніми дослідженнями, проведеними Ма [25], Сіркією та Сорва [26], та нещодавнім дослідженням Кона [27]. У цьому розділі включено старіші дослідження, щоб підкреслити, що ці помилкові уявлення існують протягом усього часу, незалежно від мов програмування. Масиви є розширенням концепції змінних, за винятком можливості зберігати кілька значень одночасно. Це робиться шляхом присвоєння індексних номерів окремим значенням даних, присутнім у масиві. Хоча це необхідно для ефективного пошуку даних, той факт, що індекси починаються з нуля, особливо бентежить тих, хто тільки починає програмувати [28]. Віттол та ін. [29] стверджують, що програми, подібні до Scratch, посилюють цю плутанину, індексуючи перший елемент масиву як одиницю замість нуля.

Відповідно, програмування багатьма сприймається як лякаюче та складне завдання [29]. Це може бути пов'язано з тим, як програмістів зображують у ЗМІ, де їх класифікують як «зануд» та «гіків», що відповідає розуму [30]. Для тих, хто ніколи раніше не програмував, це може здатися причиною для залякування. У тих, хто зрештою записується на вступні курси програмування, може виникнути почуття тривоги, коли людина виявляє, що вона недостатньо «розумна», щоб програмувати. Це додатково підтверджує дослідження, проведене Чангом та Шісоном [31], які опитали 307 учасників, яке показало, що зв'язок між тим, як

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

людина сприймає складність програмування, та рівнем тривоги є прямо пропорційним. Це має вирішальне значення, оскільки сприйняття програмування людиною та її здатність бути хорошим програмістом значно впливають на її навички програмування [2]. З цієї причини мотивація до набуття навичок та знань програмування також може зменшуватися, що ще більше перешкоджає залученню студентів.

1.2 Віртуальна реальність для освіти з комп'ютерного програмування

Використання віртуальних середовищ для навчання підтримується конструктивістською теорією [10, 1, 2] та теорією втіленого пізнання [2,3]. Отже, у цьому розділі висвітлено існуючі методи, що використовують методи віртуальної реальності для розробки досвіду для навчання комп'ютерному - програмуванню. Таніелу та ін. [3] розробили досвід віртуальної реальності під назвою «OOPVR», щоб зменшити абстрактність концепцій ООП за допомогою аналогій. Щоб покращити розуміння студентами об'єктів та класів, OOPVR використовує будинок для зображення класу, де можна побудувати кілька будинків, використовуючи відповідний креслення. Таким чином, це означає, що клас може створювати екземпляри кількох об'єктів. Протягом OOPVR було реалізовано багато аналогій для представлення концепцій програмування, а саме екземплярів, методів та інкапсуляції. Щоб визначити ефективність системи, автори проаналізували результати попередніх та наступних опитувань, в яких 17 студентів опитали щодо їхньої впевненості у візуалізації різних концепцій ООП. Результати показали, що учасники стали значно впевненіше розуміти концепції ООП після цього досвіду.

Аналогічно, [5] запропонував VR-досвід, у якому студенти можуть вивчати основні концепції програмування шляхом візуалізації коду. Студентам надається блок коду, в якому вони зможуть виділяти певні рядки коду або частини рядка коду, щоб спостерігати, як код представлений у віртуальному

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

просторі. Наприклад, якщо студент виділяє код "int x = 10;", йому буде представлено куб, позначений назвою змінної (у цьому випадку назва змінної відповідає "x") та сферою для представлення значення (у цьому випадку значення відповідає 10). Студенти також повинні помістити сферу в куб, щоб представити присвоєння значення змінній. Однак, тестування для підтвердження ефективності VR-досвіду не проводилося.

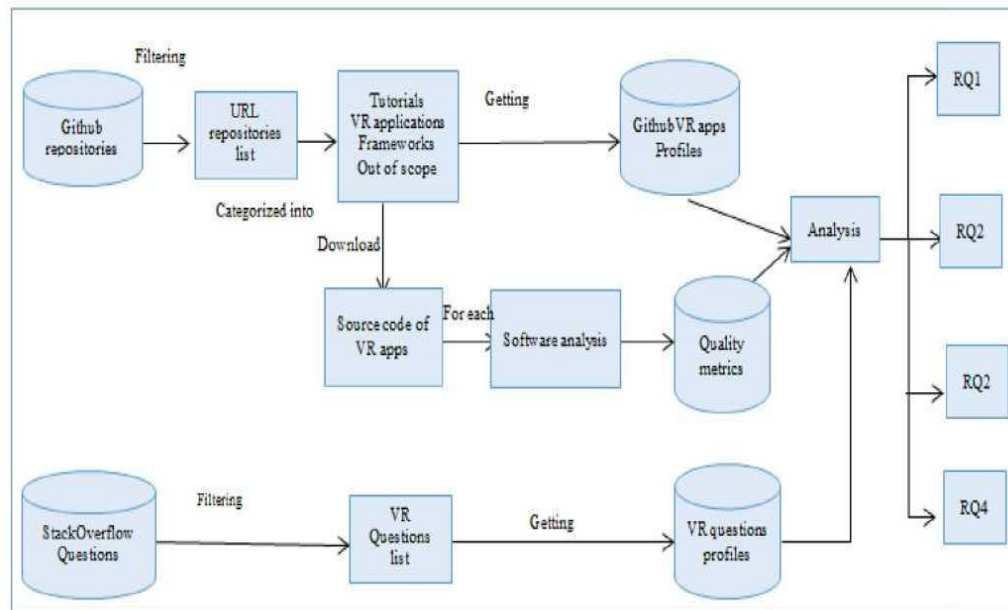


Рисунок 1.1 - Огляд нашого дослідницького підходу.

Вінкур та ін. [36] поєднали елементи віртуальної реальності та гри з блочним програмуванням, щоб розробити «Cubely», який використовується для навчання програмуванню. Cubely містить куби, що представляють програмування, яке можна організувати для побудови програм. Студенти повинні створювати програми з блоків коду, щоб керувати своїм персонажем та направляти його для виконання завдань. Згідно з результатами дослідження, з 19 учасників 18 віддали перевагу Cubely, оскільки він простіший у використанні та цікавіший, ніж типові онлайн-тренінги з кодування, такі як Code.org. Буалі та ін. [7] розробили VR-гру під назвою «Imikode», щоб допомогти студентам ознайомитися з концепціями ООП. Ця система надає студентам можливість

створювати та будувати віртуальні світи за допомогою програмування. Наприклад, студенти можуть використовувати такі команди, як «fox = new Fox()», щоб створити лисицю у своєму віртуальному світі. Незважаючи на це, жодного тестування для підтвердження ефективності системи для студентів не проводилося.

Відповідно, Чен та ін. [3] розробили інструмент віртуальної реальності, який дозволив учням створювати рівні в грі, щоб кинути виклик своїм одноліткам. Перед початком гри учень, який грає роль творця рівнів, пише коди у віртуальному середовищі, щоб розміщувати роботів-персонажів навколо середовища, які виступатимуть у ролі перешкод. Учні, які потім грають у гру, повинні отримувати підказки, розкидані по віртуальному середовищу, долаючи перешкоди, встановлені попереднім учнем. У певному сенсі учні зможуть оцінити ефективність власного коду залежно від того, як реагує система. Автори стверджують, що учні віком від 9 до 13 років, які спробували гру, дали позитивні відгуки щодо залученості та ефективності навчання. Аналогічно, Сегура та ін. [39] розробили гру віртуальної реальності під назвою «VR-OCKS», яка вимагала від учнів використовувати функції коду, представлені блоками в грі, для розв'язання головоломок. Щоб перевірити ефективність системи, було залучено 20 учасників, які грали в VR-OCKS, та ще 20 учасників, які не грали в гру. Кожну групу з 20 учасників додатково розділили вдвічі, щоб виконати завдання, синонімічні головоломкам, представленим у VR-OCKS, у «Kodu» та «Blockly» – двох популярних системах, що використовують візуальне програмування, щоб допомогти учням вивчати програмування. Результати показали, що ті, хто раніше грав у VR-OCKS, змогли пройти на 25% більше рівнів, ніж ті, хто цього не робив.

Стігалл та Шарма [20] розробили VR-гру для навчання концепціям ООП, таким як поліморфізм, успадкування та інкапсуляція. Наприклад, щоб навчити концепції успадкування, студенти повинні побудувати транспортний засіб. Транспортний засіб можна побудувати за допомогою креслення, де дочірні класи, що містять певні функції, розширюватимуть клас Vehicle. Модуль був

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

протестований за допомогою 15 студентів, і було виявлено, що 92% студентів вважають, що система допомогла їм вивчити концепції ООП. Відповідно, Као та ін. [41] розробили гру під назвою «HackVR», щоб також навчити студентів концепціям ООП. HackVR містить вузли, які містять програмні конструкції, такі як об'єкти, умовні оператори, виклики функцій та обробники подій. Студенти повинні створювати програми з набором згаданих вузлів. Наприклад, вузол сутності представляє об'єкти у віртуальному світі (двері), програми, приєднані до вузла, потім запускатимуть віртуальний об'єкт (відчиняючи двері). Автори очікують, що система може допомогти у навчанні концепціям програмування. Однак, поки що не було проведено жодних тестів, щоб підтвердити це.

Джин та ін. [42] розробили "VWorld", інструмент, який дозволяє дітям створювати віртуальні світи та програмувати 3D-об'єкти. Об'єкти у віртуальному середовищі можна програмувати, використовуючи блоки, що представляють інструкції для формування послідовності коду. Тестування проводилося з 3 студентами університету, і було помічено, що інструмент був одночасно цікавим та простим у використанні. Заргам та Камсані [4] розробили інструмент віртуальної реальності, який навчає програмуванню, змушуючи учнів проектувати та програмувати американські гірки у віртуальному середовищі. Студенти також можуть спостерігати за своїм дизайном у віртуальній реальності, щоб виправляти помилки або додавати більше функцій до своїх американських гірок. Щоб перевірити доцільність системи, було залучено 7 учасників віком від 10 до 15 років. Було помічено, що інструмент був цікавим у використанні. Аналогічно, Пармар та ін. [4] розробили "Programming Moves", щоб навчати програмуванню, дозволяючи студентам програмувати аватара для виконання танцювальних рухів. Аватар програмується шляхом перетягування попередньо запрограмованих рухів для формування програмної послідовності. Для тестування системи було залучено 47 учасників, і було виявлено, що студенти вважали інструмент захопливим, простим у використанні та захопливим.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

1.3 Інструменти візуального програмування

Візуальне програмування дозволяє користувачам вільно створювати та маніпулювати програмами за допомогою графічного інтерфейсу, а не лише рядків коду [10]. Це робить програмування менш складним завданням, оскільки воно зосереджене на тому, як вирішити конкретну проблему, а не на синтаксисі певної мови програмування [21]. Це, у свою чергу, дозволяє студентам розвивати свої навички вирішення проблем навіть з обмеженням часу. Порівняно з текстовими -мовами програмування (TPL), візуальні мови програмування (VPL) також можуть сприяти позитивному ставленню до вивчення програмування [52].

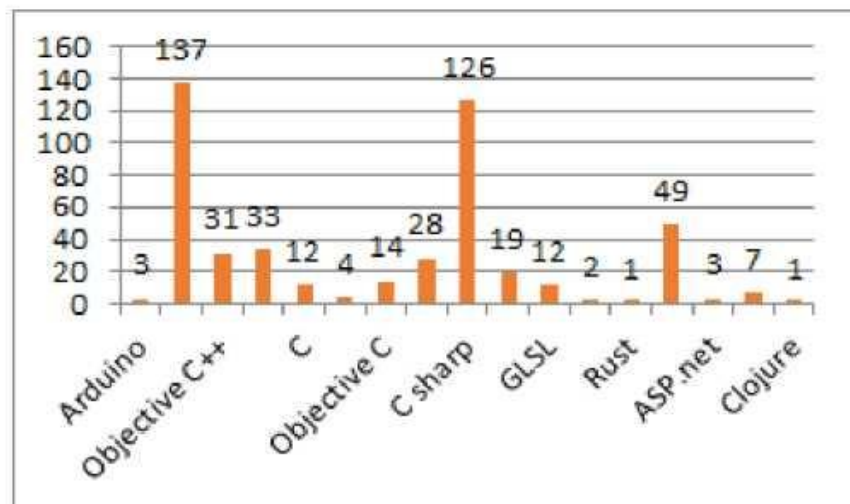


Рисунок 1.2 - Мови програмування, що використовуються у VR-проектах (JavaScript - найпопулярніша мова: 137 проектів)

Блокове програмування застосовує принципи візуального програмування, оскільки воно включає блоки, що представляють певні «блоки коду», що існують у типовій комп'ютерній програмі. Це означає, що користувачі можуть програмувати з цими блоками, надаючи їм візуальний та графічний інтерфейс. Деякі приклади програм для блокового програмування включають Scratch, Alice та Blockly. Розроблений в першу чергу для дітей, Scratch дозволяє користувачам створювати онлайн-проекти за допомогою інтерфейсу на основі

блоків з функцією перетягування [5]. Аналогічно, Alice дозволяє користувачам розробляти та створювати комп'ютерну анімацію з тривимірними моделями [54]. Blockly, розроблений Google, також схожий, за винятком того, що він використовується в програмах, де розробникам надається синтаксис та програмне представлення, щоб зробити процес створення програм більш безперервним [5]. Однак не всі програми візуального програмування базуються на блоках з функцією перетягування. Наприклад, Pure Data [6] містить діаграми потоків та станів, тоді як Kodu використовує піктограми [7].

Окрім того, що візуальні мови програмування роблять програмування менш складним, вони також можуть підвищити мотивацію учнів [5]. За даними Сераджа та ін. [5], під час опитування щодо того, чи надається перевага блокам або коду для програмування, було виявлено, що студенти віддають перевагу блокам, причому дівчата частіше обирають блоки, ніж хлопці. Відповідно, програма Milo (подібна до Scratch, але використовується для занять з машинного навчання) від Рао та ін. [10] показала, що 90% учасників погоджуються, що блочне програмування полегшує розуміння концепцій. Однак інтерфейс цих програм на основі перетягування блоків може створювати проблеми, оскільки зазвичай перетягування блоків займає більше часу, ніж введення тексту [1].

Ігрові інструменти для програмування

Ігрові інструменти в цьому контексті стосуються програм, розроблених для допомоги в навчанні комп'ютерним наукам, зокрема комп'ютерному програмуванню. В огляді, проведеному Ліндбергом та ін. [2] на основі досягнутих результатів програмування ігор, було виявлено, що коли ігрові елементи впроваджуються в контекст навчання комп'ютерному програмуванню, можна досягти таких результатів, як ефективність навчання, мотивація та залученість.

Пападакіс та Калогіаннакіс [3] використовували Classcraft, онлайн-платформу для управління поведінкою та навчанням учнів. Платформа використовує методи гейміфікації, такі як підвищення рівня, отримання здібностей, робота в командах, а також дозволяє вчителям створювати квести та

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

завдання на основі програмування для своїх учнів. Провівши опитування 30 учасників з однієї середньої школи в Греції, було виявлено, що учасники експериментальної групи були залучені та мотивовані протягом усього досвіду. Однак також було виявлено, що між контрольною та експериментальною групою не було суттєвої різниці з точки зору успішності учнів у тестуванні. Крім того, Матрані та ін. [4] використовували гру під назвою «LightBot», щоб перевірити доцільність системи щодо навчання концепціям програмування. У грі учасники повинні керувати роботом для розв'язання головоломок, використовуючи блоки, що представляють команди, ідентичні коду. Результати оцінювання, зібрані від 20 учасників, показали, що гра успішно та повністю впровадила такі концепції програмування, як рекурсія, умовні оператори та функції.

Вонг та Ятім [5] розробили гру під назвою «Одіссея Фенікса», щоб допомогти у вивченні концепцій об'єктно-орієнтованого програмування (ООП). Це досягається шляхом зіставлення ігрових процесів з їхніми відповідними концепціями. Концепція успадкування, наприклад, відображається в елементі розробки гри, де ресурси, необхідні як носовому, так і основному снаряженню, можуть бути спільними, оскільки вони належать до категорії спорядження. Результати, отримані в результаті попереднього та пост-тестування 214 студентів першого курсу університету, були проаналізовані для визначення ефективності гри. Було виявлено, що між результатами попереднього та пост-тестування існували суттєві відмінності, що свідчить про те, що GBL є чудовим інструментом для отримання знань та навчання. Крім того, Ойелере та ін. [66] створили гру «MobileEdu-puzzle», щоб також навчати програмуванню. Гра функціонує, вимагаючи від студентів упорядкування невпорядкованих рядків коду. Було виявлено, що 71% з 51 студента, які взяли участь у дослідженні, сказали, що завдяки цьому досвіду вони змогли ефективно вивчити програмування. Однак, жодного попереднього та подальшого тестування для підтвердження цього факту не було проведено. Учасники також позитивно відповіли на запитання, чи вдалося грі мотивувати їх вивчати програмування.

Ореховацький та Бабич [7] досліджували доцільність гри «CodeCombat»,

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

в якій учасники мали розробити та виконати код на Python для вирішення завдань. Як правило, код пишеться для переміщення персонажа з однієї точки в іншу. Результати, зібрані від 175 учасників, показали, що студенти позитивно ставилися до гри, щоб вивчити концепції програмування . Дюран та ін. [8] розробили гру під назвою PLMan, де студенти повинні створювати контролери за допомогою мови програмування Prolog. Мета полягає в тому, щоб розробити контролер з набором основних правил, таких як «Якщо привид праворуч від вас, рухайтесь ліворуч», щоб ігровий персонаж міг поглинути всі точки, розміщені в лабіринтоподібному середовищі. Гра використовувалася як частина курсу обчислювальної логіки протягом двох років з 2015 по 2016 рік, в середньому 300 студентів кожного семестру. Результати показали, що майже 70% студентів обрали обчислювальну логіку своїм улюбленим предметом.

Даунчароне та Панджабурі [9] розробили гру під назвою «CP m-Game» для навчання студентів програмуванню мовою С. Гра працює таким чином, що студентам пропонується розповідь, у якій їх просять проаналізувати проблему та знайти рішення, розробляючи код з блоків коду. Гра також допомагає студентам визначити, чи є в їхньому коді помилки. Для оцінки ефективності системи було залучено загалом 50 учасників. Учасників попросили пройти попереднє тестування у формі лабораторної оцінки перед випробуванням втручання та повторне тестування після нього. Результати показали, що студенти значно краще виконали повторне тестування, були більш мотивованими, почувалися впевненіше з точки зору синтаксису програмування на С та впевнено проходили тести. Мін та ін. [20] розробили гру під назвою «Engage» для розвитку обчислювального мислення за допомогою гри, заснованої на розповіді. У грі студенти повинні відновити несправну серверну мережу в підводному дослідницькому центрі. Це можна зробити, розробивши код за допомогою мови програмування на основі блоків для розв'язання головоломок. Можливість роботи системи була перевірена за допомогою 14 учасників, які пройшли посттестове опитування, і було виявлено, що гра була дуже захопливою.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

Казімоглу [1] розробив гру під назвою «Запрограмуй свого робота» для підвищення мотивації та впевненості студентів у програмуванні. Гра подібна до «Lightbot», у якій студенти повинні перемістити персонажа з однієї точки в іншу, використовуючи серію команд. Можливість використання системи була перевірена за допомогою 151 студента. Було виявлено, що студенти продемонстрували значне покращення внутрішньої мотивації, знань програмування та впевненості після втручання. Малліаракіс та ін. [72] розробили гру під назвою «СМХ», щоб визначити вплив багатокористувацької онлайн-рольової гри (MMORPG) на навчання та вивчення комп'ютерного програмування. Протягом гри студенти знайомляться з концепціями програмування за допомогою ігрового персонажа. Отримані знання потім можна використовувати для виконання різних завдань, поставлених у грі. Систему було протестовано за допомогою 76 учасників, яких було розподілено до експериментальної групи, а 234 учасники – до контрольної групи. Результати показали, що студенти, які використовували гру, мали вищий середній бал під час проміжних іспитів. Також було виявлено, що учні вважали гру цікавою.

Боннер та Дорнайх [3] розробили гру під назвою «-Чарівниця пів року», щоб збільшити участь жінок у сфері інформатики. Учні повинні виконувати завдання, використовуючи код Python для керування ігровим персонажем. Систему було протестовано на 15 учнях середньої школи. Результати показали, що учасниці частіше обирали галузь, пов'язану з інформатикою, після цього досвіду. Згідно з оглядом Шарми та ін. [4], було виявлено, що серйозні ігри для комп'ютерної освіти можуть покращити сприйняття дівчатами інформатики як життєздатного варіанту кар'єри.

1.4 Висновок по розділу

Вивчення комп'ютерного програмування залишається складним завданням для студентів через абстрактність концепцій, труднощі з синтаксисом і низьку мотивацію, що вимагає нових підходів до викладання, орієнтованих на

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

наочність, практичність і контекстуалізацію знань. Використання віртуальної реальності, візуального програмування та ігрових інструментів демонструє значний потенціал у покращенні мотивації, залученості та розуміння студентів під час вивчення комп'ютерного програмування.

					ДРБ.ІІ - 38.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

РОЗДІЛ 2 . VR-ГРА ЯК ЗАСІБ НАВЧАННЯ ОБ'ЄКТНО-ОРІЄНТОВАНОМУ ПРОГРАМУВАННЮ

2.1 Функціонування гри та підхід до навчання

Це дослідження демонструє використання VR-ігор (Imikode) на заняттях з програмування для навчання та вивчення концепцій об'єктно-орієнтованого програмування (ООП). Як уже було встановлено, відомо, що 3D-ігри покращують продуктивність та результати учнів порівняно з 2D-іграми [3]. Наприклад, якість 3D-середовища у VR-інструменті Imikode дуже полегшує учням визначення їхнього поточного положення у віртуальному світі. Крім того, привабливий характер 3D-гри покращує ідентифікацію та створення об'єктів, не витрачаючи забагато часу на блукання простором. Imikode, 3D-гра, була спеціально розроблена для того, щоб допомогти учням зрозуміти концепції ООП у віртуальному середовищі з ефектом імерсивності. Вибір Imikode серед інших VR-інструментів полягає в його дизайні користувацького інтерфейсу. Imikode дозволяє користувачам грати в ігри з мінімальним наглядом або без нього. Це завдяки інтеграції інтелектуального агента (Боба), який віртуально керує користувачами на кожному кроці гри. Крім того, гра включала багато функцій ООП, адаптованих до потреб студентів. Згідно з опитуванням [4], студенти надавали перевагу простому підходу до вивчення ООП для кращого розуміння, такому як вивчення основ ООП, перш ніж заглиблюватися в класи, конструктори, перевантаження/перевизначення методів та збирання сміття.

Крім того, гра зображує типову обстановку в класі, де учні пишуть на дошці, очищають дошку, коли роблять помилки, та багато іншого. Imikode було реалізовано за допомогою Google Cardboard SDK (комплекти розробки програмного забезпечення), хоча в Imikode можна грати, використовуючи більш просунуті гарнітури, під час розробки враховувалися мінімальні вимоги до пристроїв, що підтримують використання Imikode. Такий підхід надає користувачам з низьким рівнем доходу можливість вивчити концепцію

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

програмування ООП за допомогою Imikode. Таким чином, гра може бути корисною як під час уроку, так і поза ним, і в неї легко грати. Окрім Google Cardboard SDK, інші технології, що використовуються для реалізації Imikode, включають ігровий рушій Unity 3D та Android SDK. Основні концепції ООП, що розглядаються у грі Imikode VR, включають об'єкти, методи (поведінку), методи встановлення та отримання, аргументи методів, перевантаження/перевизначення, збирання сміття при спадкуванні.

Детальний огляд цих концепцій ООП, реалізованих в Imikode, представлено в [6]. Перша версія гри Imikode VR була реалізована, щоб дозволити вчителям ознайомити з основними концепціями ООП, створюючи віртуальний світ зі створенням таких об'єктів, як дерева, будинки та тварини, за допомогою підходу розповіді історій, щоб забезпечити справжнє занурення та залучення у віртуальний простір. Друга версія вирішила недоліки першої версії, надаючи зворотний зв'язок у режимі реального часу, коли допускаються помилки. Крім того, вона мала покращений компонент штучного інтелекту, який дозволяє учням отримувати більш інтерпретаційний та інтелектуальний зворотний зв'язок щодо помилок.

Під час запуску програми Imikode віртуальний персонаж «Боб» знайомить гравця з ігровим середовищем та його метою. Цей персонаж веде гравця протягом гри в розмовній манері за допомогою згенерованого тексту та звуку. Агент, що розмовляє (Боб), навчає гравців, що вони вивчатимуть та практикуватимуть протягом певного періоду часу. Наприклад, агент, що розмовляє, може повідомити гравців про необхідність вивчення класів та об'єктів, після чого гравці додають інструкції у вигляді бульбашок на дошці, щоб показати рівень засвоєння. Боб веде гравця до створення об'єктів у віртуальному середовищі, використовуючи знання про оголошення змінних та оператори присвоєння. Крім того, віртуальний персонаж веде гравця в області виклику методів, використовуючи функції сеттерів та геттерів, включаючи перевизначення методів, перевантаження та компонент штучного інтелекту, який дозволяє учням отримувати інтелектуальний та більш інтерпретативний

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

зворотний зв'язок щодо помилок. Якщо гравець зміг правильно додати синтаксис запису частини коду, віртуальний персонаж відповідає посмішкою, вказуючи напрямок створеного об'єкта. Однак для функції Боб відповідає, вказуючи користувачеві на створене завдання. Однак, якщо гравець не додає правильний код на дошку, Боб хмуриться на такого гравця та повідомляє гравцеві про необхідність повторити завдання. Наприклад, два дерева та два будинки можна побудувати, присвоївши значення змінній (дерево = 2; будинок = 2), подібно до стилю кодування у звичайних мовах програмування.

Однак у середовищі віртуальної реальності це досягається шляхом клацання та перетягування кожної одиниці змінної або значення (дерево = 2; будинок = 2), доступної у списку бульбашок гри Imikode VR, на дошку. Щоб створити екземпляр об'єкта, такого як лисиця, команда bubble, яку гравець вибере та додасть до дошки, - це `fox = new Fox()`. Щоразу, коли гравець правильно вибирає інструкцію, гравець бачить негайне виконання, щоб зрозуміти, як поводить ся ця концепція ООП. Якщо вибрано неправильну інструкцію, Боб повідомляє гравцеві про необхідність повторити спробу. Виклики методів для реалізації поведінки об'єктів змодельовані в концепції ООП Imikode. Наприклад, щоб змусити об'єкт лисиці ходити, гравець повинен вибрати та додати метод під назвою `fox.walk()`. Коли це зроблено правильно, гравець бачить об'єкт лисиці, що ходить у симульованому середовищі. Якщо гравець вибере та додасть метод `fox.die()` у віртуальному просторі, об'єкт лисиці негайно завмирає, а через деякий час залишає простір за допомогою методів збору сміття. Варто зазначити, що виклик методу передуює створенню екземпляра об'єкта, і це необхідно враховувати під час виклику методів.

Фігура На рис. 2.1 показано скріншот гравця, який створює дерева, будинки та екземпляри об'єктів лисиць, як пояснено вище. Ліва частина рис. На рисунку 2.1 показано, як гравець створює ці об'єкти, а на правому екрані показано результат виконання кожного рядка інструкції.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

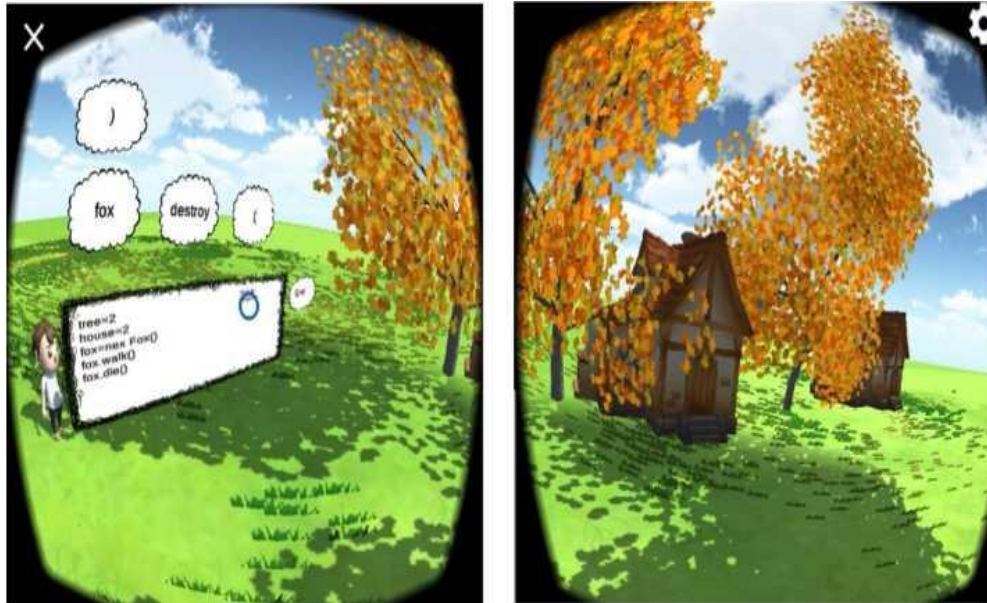


Рисунок 2.1 - Знімки екрана гравця, який створює об'єкти у середовищі Imikode VR

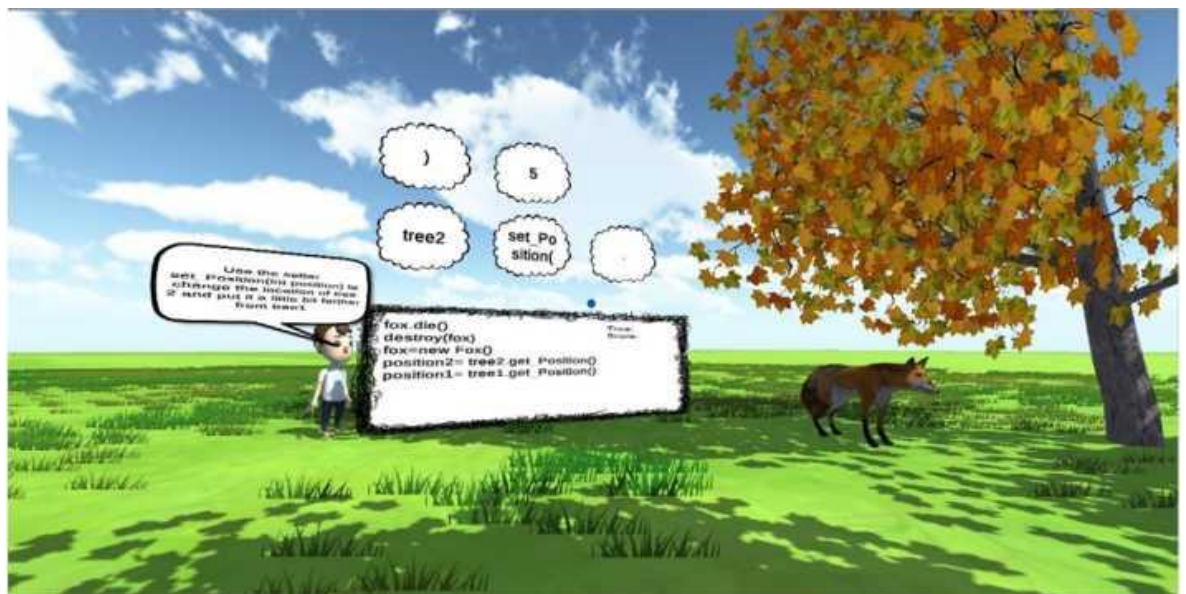


Рисунок 2.2 - Середовище Imikode для віртуальних об'єктів, що створюються за допомогою концепцій ООП

Більш просунуті концепції ООП та їх застосування показані на рис. 2.2. Наприклад, як використовувати методи getter та setter для створення змінних та присвоєння їм значень за допомогою функції з аргументами, зображено на рис. 2.2.

2.2 Модифікація та вдосконалення навчального ігрового середовища

У контексті навчання програмуванню, особливо на початкових етапах, ключову роль відіграє не лише навчальний матеріал, а й середовище, в якому він засвоюється. Одним із факторів, що ускладнює процес навчання для новачків, є неінтуїтивність повідомлень про помилки, які генеруються компіляторами мов програмування. Як зазначають [8], багато учнів не розуміють значення цих повідомлень, що знижує їхню мотивацію до подальшого вивчення.

Щоб подолати цю проблему, було запропоновано низку підходів, зокрема, візуалізацію помилок, адаптивне навчання та інтерактивні середовища. Одним із перспективних рішень у цьому напрямі є розробка ігрових VR-середовищ з використанням технологій штучного інтелекту, таких як **Imikode VR** - віртуальна навчальна гра, орієнтована на засвоєння об'єктно-орієнтованого програмування (ООП).

Покращена версія гри Imikode VR являє собою інтерактивне навчальне середовище з елементами штучного інтелекту, спрямоване на полегшення засвоєння основ об'єктно-орієнтованого програмування в умовах віртуальної реальності. Однією з ключових переваг цієї версії є наявність функціоналу, який дозволяє динамічно змінювати складність завдань залежно від рівня знань користувача. На відміну від традиційних середовищ програмування, де повідомлення про помилки мають стандартний і часто складний для новачків формат, Imikode VR пропонує користувачеві спрощені пояснення, адаптовані до рівня його підготовки. Це реалізується за рахунок вбудованого AI-агента, який аналізує код, розпізнає типові помилки та надає зрозумілі пояснення як у текстовому, так і у голосовому форматі.

Завдяки можливості миттєвого зворотного зв'язку гравець отримує повідомлення про результат виконання програми одразу після запуску, що значно підвищує ефективність навчального процесу. У випадку, якщо студент допускає помилку при створенні об'єктів або виклику методів, система автоматично генерує діалогове повідомлення, яке пояснює суть помилки і

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

пропонує можливі шляхи її виправлення. Такий підхід сприяє формуванню аналітичного мислення, а також навичок самостійного виправлення коду, що є надзвичайно важливим у процесі навчання програмуванню.

Навчальні матеріали у грі структуровані за принципом модульності, що дозволяє охоплювати окремі теми в межах окремих рівнів. Наприклад, перші рівні присвячені створенню класів, оголошенню атрибутів і методів, а наступні рівні поступово вводять поняття конструкторів, наслідування, поліморфізму та інкапсуляції. Кожен рівень передбачає виконання певного завдання у VR-середовищі, яке включає написання коду, перевірку результатів і виправлення можливих помилок за допомогою вбудованих підказок. Така послідовна структура забезпечує гнучкість навчального процесу та дозволяє учням поступово нарощувати складність засвоюваного матеріалу.

Imikode VR використовує сучасні засоби візуалізації, зокрема графічний рушій Unity, у поєднанні з технологіями віртуальної реальності, такими як Oculus SDK, що дає змогу створити повністю занурююче середовище. Окрім цього, гра має модуль аналізу синтаксису та семантики, який сприймає введений студентом код, компілює його у хмарному середовищі та надсилає результати назад до клієнта. Ці результати обробляються AI-компонентом, що базується на сучасних методах обробки природної мови. Таким чином, повідомлення про помилки трансформуються з сухого технічного тексту у зрозумілі поради, що сприяє глибшому розумінню та усвідомленню структури програмного коду.

У порівнянні з класичними інтегрованими середовищами розробки, такими як Eclipse чи NetBeans, Imikode VR значно ефективніше справляється з дидактичними завданнями, оскільки поєднує в собі ігрові елементи, візуалізацію та гейміфікований процес навчання. Учні не лише сприймають навчальний матеріал, але й активно взаємодіють з ним у просторі, що сприяє кращому закріпленню знань. Це підтверджується результатами експериментального впровадження гри у навчальний процес, яке продемонструвало зростання успішності та підвищення мотивації серед учасників. Студенти показали вищий рівень розуміння концепцій ООП, легше опрацьовували складні теми та з

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

меншою кількістю помилок справлялися із завданнями на написання коду.

Варто також зазначити, що Imikode VR має потенціал бути впровадженим у дистанційне навчання. Завдяки підтримці технологій WebXR та хмарних обчислень, ця система може бути використана без необхідності встановлення складного програмного забезпечення, що робить її доступною для широкого кола користувачів. Система може автоматично зберігати прогрес користувача, надавати статистику викладачу, а також дозволяти коригування сценаріїв навчання відповідно до потреб групи або індивідуального користувача.

Мотивація до покращення гри

Базова версія Imikode VR забезпечувала взаємодію користувача з простим VR-середовищем, у якому учень міг створювати об'єкти, задавати їм властивості та викликати методи. Проте, наявні обмеження — зокрема, статичність рівнів, відсутність динамічної адаптації складності та брак зрозумілих пояснень помилок - зумовили необхідність суттєвого оновлення гри.

Основні вдосконалення

1. Інтеграція компонента штучного інтелекту

Ключовою інновацією у вдосконаленій версії Imikode VR стала поява агента на основі AI, що має такі функції:

- **Негайний зворотний зв'язок:** при виникненні помилки (наприклад, некоректний виклик методу), агент пояснює, у чому саме полягає проблема, використовуючи просту і доступну мову.
- **Адаптивне навчання:** на основі поведінки гравця (час виконання завдань, кількість помилок) система підлаштовує складність наступних рівнів.
- **Візуалізація помилок:** помилки підсвічуються у VR-просторі, що дозволяє учневі візуально асоціювати синтаксичні та семантичні конструкції коду з їхнім результатом.

2. Динамічна генерація рівнів

На відміну від статичних сценаріїв у попередній версії, оновлена версія підтримує:

- Генерацію завдань на основі історії користувача.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

- Можливість створення нових сценаріїв викладачем.
- Наявність тематичних "світів" для кожної концепції ООП

(інкапсуляція, наслідування, поліморфізм тощо).

3. Покращена підтримка ООП

У новій версії гри акцент зроблено на активне застосування основних принципів об'єктно-орієнтованого програмування:

- **Конструктори:** учень вчиться оголошувати та використовувати конструктори в класах.
- **Створення екземплярів об'єктів:** система перевіряє правильність створення об'єктів, їхніх властивостей та методів.
- **Миттєве відображення результатів:** після створення об'єкта користувач одразу бачить ефект у VR-середовищі (наприклад, об'єкт з'являється у віртуальному класі з відповідними параметрами).

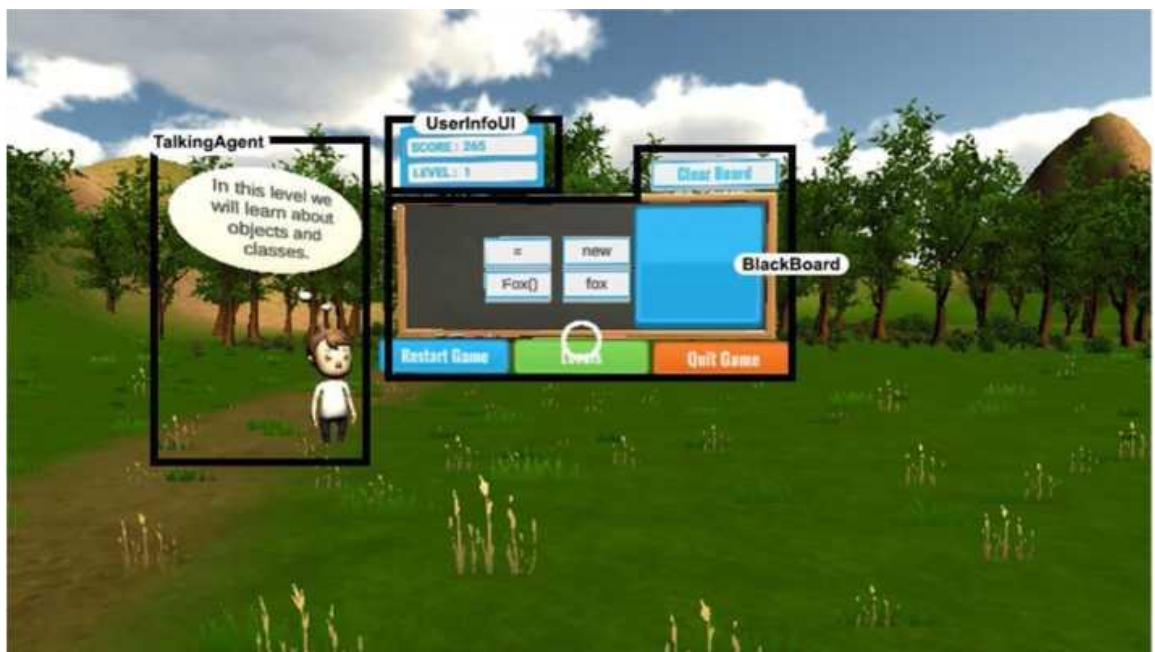


Рисунок 2.3 - Покращена версія Imikode з динамічними рівнями та компонентом штучного інтелекту для кращого розуміння повідомлень про помилки, що генеруються компіляторами ООП

Окрім компонента штучного інтелекту в покращеному Imikode, можна динамічно додавати кілька рівнів гри, щоб надати учням більше можливостей

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33

досліджувати середовище віртуальної реальності для захопливого навчання. У цій покращеній VR-грі Imikode користувачі навчаються більше завдяки більшій кількості вбудованих функцій та концепцій ООП, таких як конструктори, миттєве відображення результатів та миттєве представлення помилок. Якщо гравець допускає помилку під час створення екземплярів об'єктів або під час виклику методів, таким гравцям надається негайний зворотний зв'язок через компонент штучного інтелекту гри. На рисунку 2.3 показано агента, який розмовляє та повідомляє гравцям, що вони вивчатимуть. На рисунку 2.2 представлено цілі та очікуваний результат гри Imikode VR.

На рисунку 2.3 зображено внутрішній інтерфейс гри, де штучний агент у вигляді персонажа надає рекомендації та повідомляє користувача про хід гри. Гравець взаємодіє з ігровим середовищем через програмні блоки, які представляють класи та методи. У разі виникнення помилки (наприклад, неправильне викликане ім'я методу), агент миттєво повідомляє про це голосовим або текстовим повідомленням.

Очікуваний вплив на навчальний процес

Покращена версія гри Imikode VR сприяє:

- **Зменшенню когнітивного навантаження:** завдяки поясненням, візуалізації та голосовим підказкам.
- **Покращенню розуміння повідомлень про помилки:** агент діє як “репетитор”, що навчає користувача розуміти суть проблеми.
- **Збільшенню мотивації:** через гейміфікацію, поступове зростання складності та систему нагород.

Imikode VR у своїй покращеній версії є прикладом поєднання передових педагогічних підходів, гейміфікації та новітніх технологій штучного інтелекту. Такий підхід може суттєво трансформувати процес вивчення ООП, особливо для новачків, які стикаються зі складністю традиційних текстових IDE. Успішна реалізація такого середовища може слугувати моделлю для подібних систем у галузі комп'ютерної освіти.

					ДРБ.ІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

2.3 Методологія проєктування та оцінювання навчального впливу

Методологія дослідження базується на змішаному підході, який поєднує елементи експериментального, порівняльного та аналітичного методів з використанням сучасних цифрових технологій, зокрема віртуальної реальності (VR), штучного інтелекту (AI) та автоматизованих систем навчання програмуванню. Основною метою методології є розробка, впровадження та оцінка ефективності покращеної версії навчальної VR-гри **Imikode**, яка орієнтована на засвоєння принципів об'єктно-орієнтованого програмування (ООП) студентами початкового рівня підготовки.

Першим етапом дослідження стало вивчення проблеми нерозуміння повідомлень про помилки, які виникають під час компіляції коду. Було проаналізовано наукові праці, що стосуються когнітивних бар'єрів у вивченні програмування [10], та зроблено висновок, що більшість студентів-початківців не мають достатніх знань для інтерпретації діагностичних повідомлень компілятора. Це обґрунтовує необхідність створення інструменту, який трансформує повідомлення в адаптовані, зрозумілі пояснення із візуалізацією у VR-середовищі.

Наступним етапом було проєктування архітектури гри Imikode VR із використанням таких технологій: Unity як рушій для 3D-візуалізації, C# як основна мова програмування логіки, Oculus SDK для реалізації VR-інтерфейсу, а також модуль AI, побудований на основі бібліотек машинного навчання для класифікації та спрощення діагностичних повідомлень. Гру було структуровано на навчальні рівні відповідно до ієрархії понять ООП – від основ до складних концепцій, таких як поліморфізм та наслідування.

Інноваційною особливістю гри є впровадження адаптивної системи навчання: гра фіксує типові помилки, які робить гравець, і відповідно до цього змінює рівень складності завдань. Це дозволяє формувати індивідуальну траєкторію навчання кожного користувача. Додатково, за допомогою AI-компоненту, помилки користувача пояснюються у зручній формі: штучний агент

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

у VR-середовищі озвучує причину помилки, надає приклад правильного використання та пропонує повторити спробу. Це стимулює гравця не лише знаходити і виправляти помилки, але й глибше розуміти причини їх виникнення.

Для оцінки ефективності впровадженої системи було проведено експериментальне навчання за участі двох груп студентів: контрольної (яка використовувала традиційне середовище програмування) та експериментальної (яка проходила навчання через Imikode VR). Експеримент тривав протягом чотирьох тижнів. Перед початком та після завершення курсу обом групам було запропоновано пройти тестування на знання основ ООП, а також анкету для оцінки рівня мотивації, зацікавленості та впевненості у написанні коду.

Для збору та обробки даних було застосовано такі методи:

- **Кількісний аналіз** результатів тестування (оцінка середнього балу, частоти помилок, темпів виконання завдань).
- **Якісний аналіз** анкетних відповідей (метод відкритого кодування).
- **Порівняння** за ключовими критеріями між групами з використанням t-тесту Ст'юдента для визначення статистичної значущості.

Зібрані дані показали, що студенти експериментальної групи демонстрували вищі результати в усіх аспектах: у них зменшилась кількість синтаксичних помилок, покращилось розуміння принципів ООП, а також підвищилась загальна мотивація до навчання. Крім того, було виявлено, що надання зворотного зв'язку у реальному часі сприяє швидшому закріпленню нових знань і розвитку навичок самоконтролю.

Таким чином, обрана методологія підтверджує доцільність впровадження VR-інструментів у процес навчання програмуванню. Поєднання ігрових механік, візуалізації, голосового супроводу та адаптивного навчання створює ефективне навчальне середовище, здатне не лише компенсувати недоліки традиційних підходів, але й значно покращити результативність навчання у галузі комп'ютерних наук.

На рисунку 2.4 зображено загальну модель/фреймворк запропонованої методології, поділену на модулі, спрямовані на вирішення проблем

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

абстрактності програмування та помилкових уявлень. Запропонована методологія використовує елементи віртуальної реальності, такі як здатність представляти концепції програмування за допомогою конкретних об'єктів, з якими можна взаємодіяти.

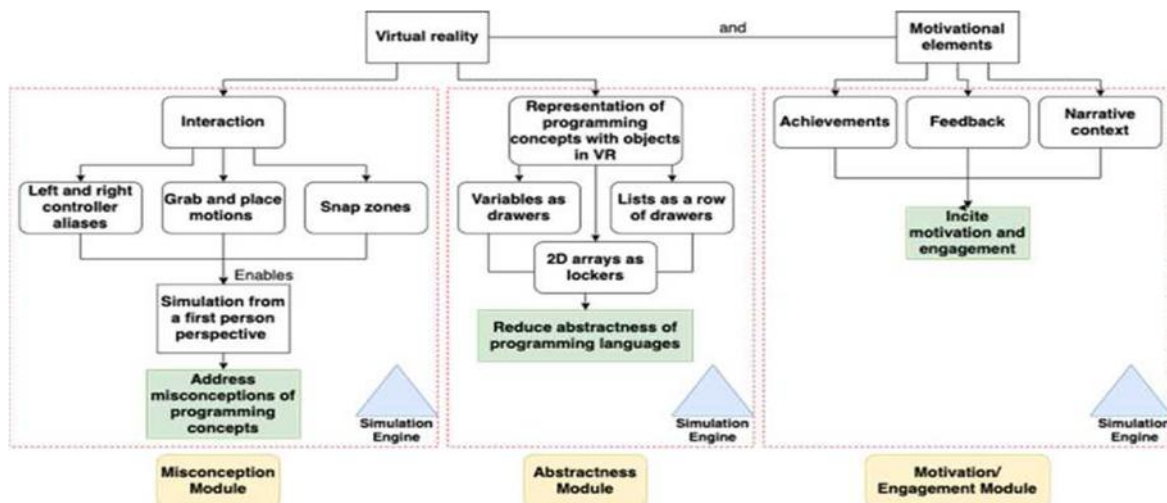


Рисунок 2.4 - Структура запропонованої методології.

Взаємодія в запропонованій системі стає можливою завдяки представленню лівої та правої рук як псевдонімів контролера. За допомогою цих псевдонімів можливі природні рухи руки, такі як захоплення та розміщення об'єктів, а також візуальні підказки для взаємодії, такі як зони прив'язки. Це, у свою чергу, дозволяє розробити симуляцію з перспективою від першої особи, яка дозволяє студентам виявляти поширені помилки програмування, формуючи «модуль помилкових уявлень». Більш конкретно, дозволяючи студентам взаємодіяти з об'єктами, це дозволяє студентам виявляти та усувати помилки програмування в середовищі віртуальної реальності. Це пояснюється детальніше в наступних розділах нижче.

Модуль абстрактності став можливим завдяки представленню змінних як шухляд, списків як рядків шухляд та двовимірних масивів як шафок у віртуальному середовищі. Цей модуль має на меті зменшити абстрактність концепцій програмування. Наразі модуль абстрактності в основному стосується

структур даних через обмеження в часі та відповідно до обсягу цього дослідження. Нарешті, мотиваційні елементи, такі як наративний контекст, забезпечення миттєвого зворотного зв'язку та наявність системи досягнень, складають модуль мотивації для стимулювання мотивації у студентів протягом усього періоду навчання. Важливо також зазначити, що система, розроблена на основі цього модуля, не призначена для заміни фактичних лекцій, а лише для доповнення викладання.

Інструменти моделювання дозволяють користувачеві програмувати віртуальне представлення робота без необхідності фізичного контакту з реальним. Крім того, вони дають змогу користувачеві навчати робота офлайн або дистанційно, без необхідності перебувати на місці або зупиняти поточне завдання, яке виконує робот. VR – це перспективний інструмент, який досліджувався в багатьох роботах для дистанційного керування роботом, використовуючи його змодельоване представлення [10], [11], [12]. Можливість безпосередньої взаємодії з реалістичною симуляцією робота також використовувалася для спрощення програмування робота, роблячи його максимально інтуїтивно зрозумілим завдяки використанню зручних інтерфейсів та візуалізації меню для зберігання необхідних конфігурацій [13]. Користувач може, наприклад, взаємодіяти з віртуальним представленням робота, перетягуючи потрібне посилання в потрібне положення. Кнопки, меню та текстова інформація можуть відображатися в потрібному положенні, щоб забезпечити інтуїтивне програмування для недосвідченого користувача.

Доповнена реальність (AR) – це ще одна технологія, яка викликає інтерес у багатьох робототехнічних застосуваннях [14], [15]. Це пов'язано з технологічним прогресом останніх років, який призвів до застосування програмування в робототехніці [16], [17], [7]. У роботі Онга та ін. представлена система програмування на основі доповненої реальності (AR) для промислового застосування в [18]. За допомогою ручного покажчика користувач може визначити точки маршруту та шлях, яким має рухатися реальний робот. Цей

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

підхід включає використання даних датчиків для включення алгоритмів планування шляху та виявлення зіткнень. На жаль, ці методи не дозволяють досягти високої точності у визначенні траєкторій робота. Фактично, це залежить від точності моделювання, що може призвести до проблем у завданнях, які мають такі вимоги.

Поєднання цих методів може забезпечити ефективний, гнучкий, інтуїтивно зрозумілий та точний спосіб програмування робота. Офлайн-метод можна використовувати для швидкого та інтуїтивного визначення програми робота. Реальне обладнання потім можна використовувати лише для невеликих налаштувань, які необхідно виконувати онлайн. В існуючій літературі бракує робіт з цього питання. Лише кілька робіт досліджували, як інтегрувати інструменти віртуальної реальності з існуючими фреймворками з відкритим кодом.

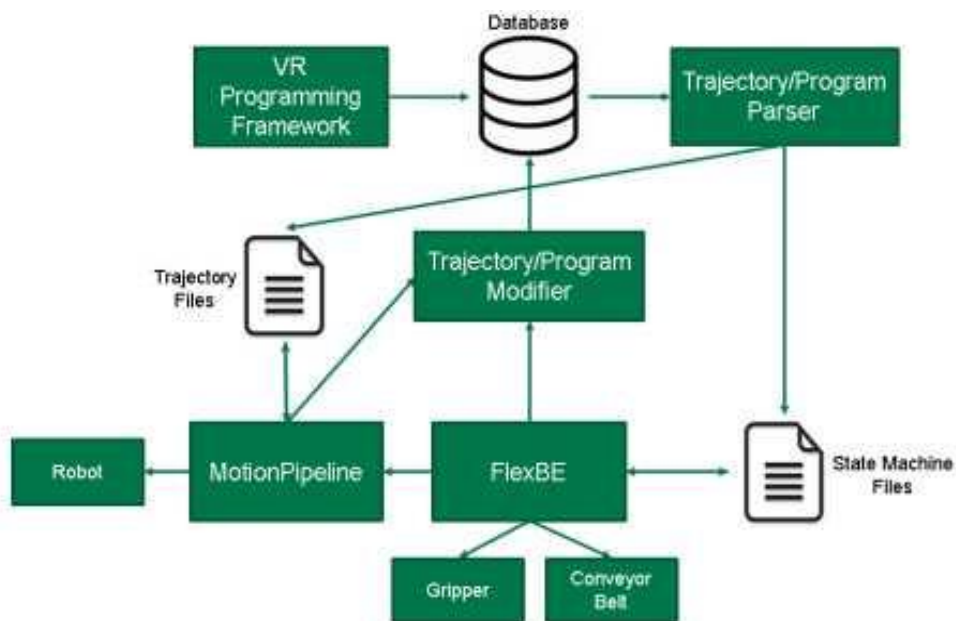


Рисунок 2.5 - Огляд архітектури системи.

Ми також виявили брак досліджень у програмуванні складніших завдань, які включають різні траєкторії руху робота та роботу додаткового обладнання. Включення визначення інших операцій, таких як керування захоплювачем або переміщення конвеєрної стрічки, дозволило користувачеві визначити всю роботизовану задачу, що включає різне обладнання та синхронізацію між

здіяними компонентами. Робота Шилінгера та ін. представляє FlexBE, зручний високорівневий механізм поведінки, який дозволяє визначити бажаний робочий процес завдання, поєднуючи прості структурні блоки [23]. Вони представляють собою основні кроки, такі як виконання певної траєкторії, активація захоплювача або очікування сигналу від датчика.

У цій роботі ми пропонуємо VR-фреймворк для інтуїтивного та швидкого визначення програми робота. Це включає визначення необхідних траєкторій, а також визначення всього завдання, яке також включає операції, що включають інші апаратні компоненти. Запропонована архітектура системи має на меті заповнити прогалину між онлайн- та офлайн-програмуванням. Таким чином, користувач може легко та швидко визначити потрібну програму у VR. Якщо потрібні коригування траєкторій робота, це можна зробити на реальному роботі, використовуючи наданий графічний інтерфейс, який можна поєднати з навчальним підвісним пультом. Використання спільної бази даних між двома компонентами дозволяє змінювати та оновлювати програму з обох сторін, залежно від потреб користувача.

2.4 Висновок до розділу

Гра Imikode VR демонструє ефективне застосування імерсивного навчання для засвоєння концепцій ООП, поєднуючи інтерактивне середовище, візуальний зворотний зв'язок і підтримку віртуального агента, що сприяє кращому розумінню програмування. Покращена версія значно підвищує ефективність засвоєння ООП, зменшуючи когнітивне навантаження та підвищуючи мотивацію студентів до вивчення програмування. Запропонований VR-фреймворк забезпечує інтуїтивне, гнучке та ефективне програмування роботів шляхом поєднання онлайн- та офлайн-методів, що дозволяє користувачеві визначати складні завдання, включаючи траєкторії руху та взаємодію з додатковим обладнанням, у віртуальному середовищі.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. ДИЗАЙН ЕКСПЕРИМЕНТУ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

Було оцінено зручність використання VR-гри Imikode для сприяння вивченню вступних курсів програмування серед учасників з нігерійського університету (RQ1) та те, як інструмент USE сприяє забезпеченню задоволеності користувачів (RQ2). Дизайн дослідження включає змішаний метод, який був використаний для відповіді на дослідницькі питання, подібний до методів, що використовувалися Hariyanto et al. (2020) та Sunday et al. (2022). У наступних підрозділах обговорювалися інструмент та матеріал; потім учасники дослідження, процедура дослідження та, нарешті, те, як проводився аналіз даних.

3.1 Вибір інструментів та технічних засобів

У цьому дослідженні для збору інформації про досвід студентів гри у віртуальній реальності та її вплив на покращення процесу навчання вступному програмуванню на основі об'єктів було використано анкету USE, що містить 30 пунктів. Анкета USE, що застосовувалася в цьому дослідженні, являє собою семибальну шкалу Лікерта, що складається з чотирьох (4) змінних: корисність, простота використання, легкість навчання та задоволення, як представлено в таблиці 3 . У цій анкеті також є варіант «НЗ» (не застосовується) для пунктів, які студенти вважають недоречними. Структура анкети наведена в Додатку І. Окрім 30 запитань, учасники мають варіант з відкритими відповідями, щоб надати коментарі щодо свого досвіду гри у віртуальній реальності. Їхні коментарі в цій частині були використані для опитування зібраних кількісних даних.

ЛУНД (2001) спочатку запропонував тривимірну (3) конструкцію анкети USE, -що включає корисність, задоволення та зручність використання. Пізніше дослідження виявило суттєвий вплив шкали зручності використання на шкалу корисності, і було виявлено, що ці дві шкали також впливають на шкалу задоволення. У цьому підрозділі використані конструкції були визначені наступним чином: Корисність. Цей конструкт показує, наскільки корисною є гра

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Imikode VR. Крім того, він означає –ступінь довіри користувачів до гри Imikode VR у покращенні їхнього навчання.

Простота використання Простота використання стосується того, наскільки легко студентам грати в гру Imikode VR. Іншими словами, вона визначає, наскільки легко грати в гру.

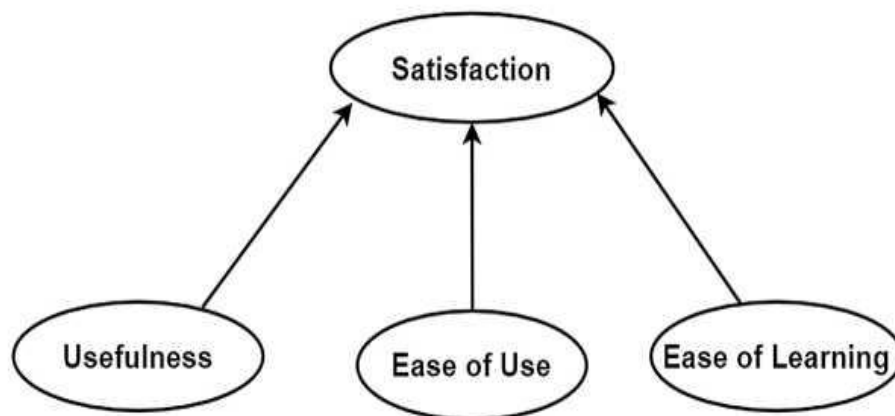


Рисунок 3.1 - Абстрактна модель взаємозв'язків між змінними в анкеті USE (Hariyanto et al., 2020)

Легкість навчання. Легкість навчання пов'язана з рівнем зусиль, які докладаються для навчання. Для того, щоб студенти отримали максимальну користь від технології в цьому навчальному процесі, обсяг необхідних зусиль має бути мінімальним.

Задоволення. Цей конструкт підкреслює, наскільки студенти задоволені грою Imikode VR. Зазвичай, після гри або використання продукту, людина отримує певний рівень задоволення.

Залежна змінна. Це змінна, яка залежить від даних інших змінних. У цьому дослідженні конструкт задоволення є залежною змінною, оскільки він залежить від даних інших конструктів, щоб визначити рівень задоволення, який студенти отримують від гри Imikode VR.

Незалежна змінна. Ця змінна не залежить від жодної іншої змінної щодо результату гри. Отже, конструкції Корисність, Легкість використання та Легкість навчання слугують незалежною змінною.

Рисунок 3.1 представляє модель взаємозв'язків між змінними в анкеті USE.

3.2 Оцінка надійності та валідності методики

Для дослідження факторної структури нашої шкали та її структурної валідності було проведено як дослідницький факторний аналіз (EFA) (див. таблицю 3.1), так і підтверджувальний факторний аналіз (див. рис. 3.2 та 3.3). Крім того, відповідно [16], для подальшої перевірки валідності шкали було проведено аналіз композитної надійності (CR) та середньої вилученої дисперсії (AVE).

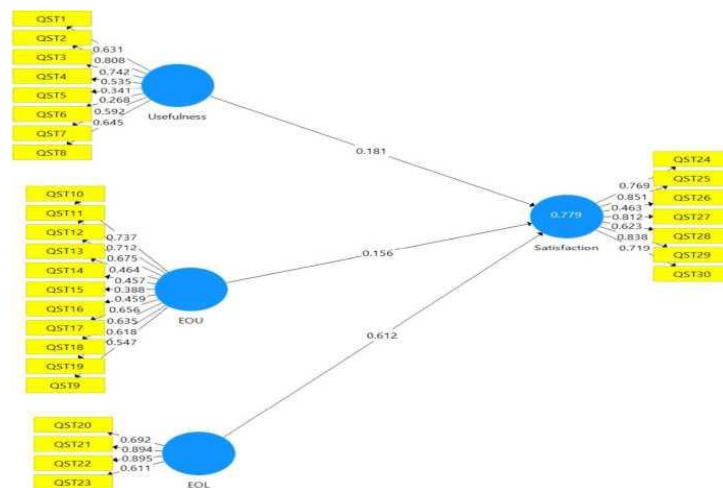


Рисунок 3.2 -Початкове факторне завантаження окремих питань та змінних

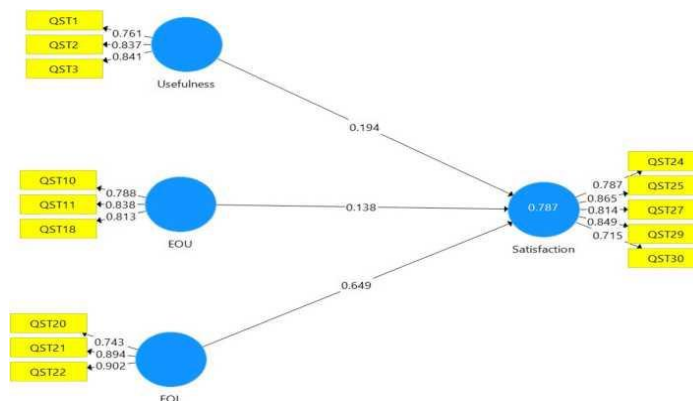


Рисунок 3.3 - Остаточне факторне навантаження окремих питань та змінних

На рис. 3.2 питання в анкеті USE були використані для визначення того, чи завантажаться вони відповідними факторами. На рисунку зображено початкове факторне навантаження окремих питань і змінних, і було виявлено, що деякі фактори були меншими за порогове значення 0,7. Ці фактори були виключені, оскільки було досягнуто порогового значення для кожного завантаження пункту, яке становить 0, , що є тестом надійності.

На рис. 3.3 деякі фактори були виключені через їх порогові значення. Метрики якості моделі, що включають альфа Кронбаха, розмір ефекту, f , композитну надійність та вилучену середню дисперсію, були обчислені та представлені в таблиці 3.3 .

Стіл на рис. 3.3 показано критерії якості для латентних змінних. З таблиці видно, що значення альфа Кронбаха досягли порогу 0,7, що забезпечує прийнятний рівень надійності дослідницького інструменту. Крім того, надійність ρ_A та композитного показника також відповідає порогу 0,7, що означає валідну шкалу для інструменту, тоді як середня отримана дисперсія перевищує контрольне значення 0,5, що додатково підтверджує шкалу. Розмір ефекту було обчислено f , щоб визначити рівень втручання змінної над іншою змінною (Cohen, 1988). За словами Коена, розмір ефекту f є низьким, якщо $f < 0,2$, і високим, якщо $f > 0,5$. Згідно з таблицею, існує слабкий вплив між корисністю, зручністю використання та задоволеністю порогом ($f < 0,2$), тоді як існує великий вплив між легкістю навчання та задоволеністю порогом ($f > 0,50$). Наслідком цього є те, що легкість, з якою студенти виявили, що гра віртуальної реальності Imikode у вивченні об'єктно-орієнтованого про

грамування принесла їм більше задоволення порівняно з іншими конструкціями. Стовпчаста діаграма, що зображує значення кореляції дослідницького інструменту на рис. 3.4 представлено нижче. На рис. 3.4 представлено всі значення кореляції для кожного пункту (Q1-Q30), і було виявлено, що значення перевищують r -значення (0,25) на рівні значущості 0,05. Таким чином, кожне корельоване значення вважається валідним, таким, що задовольняє критерії. Отже, цей інструмент корисний для оцінки навчальних ігор

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

віртуальної реальності.

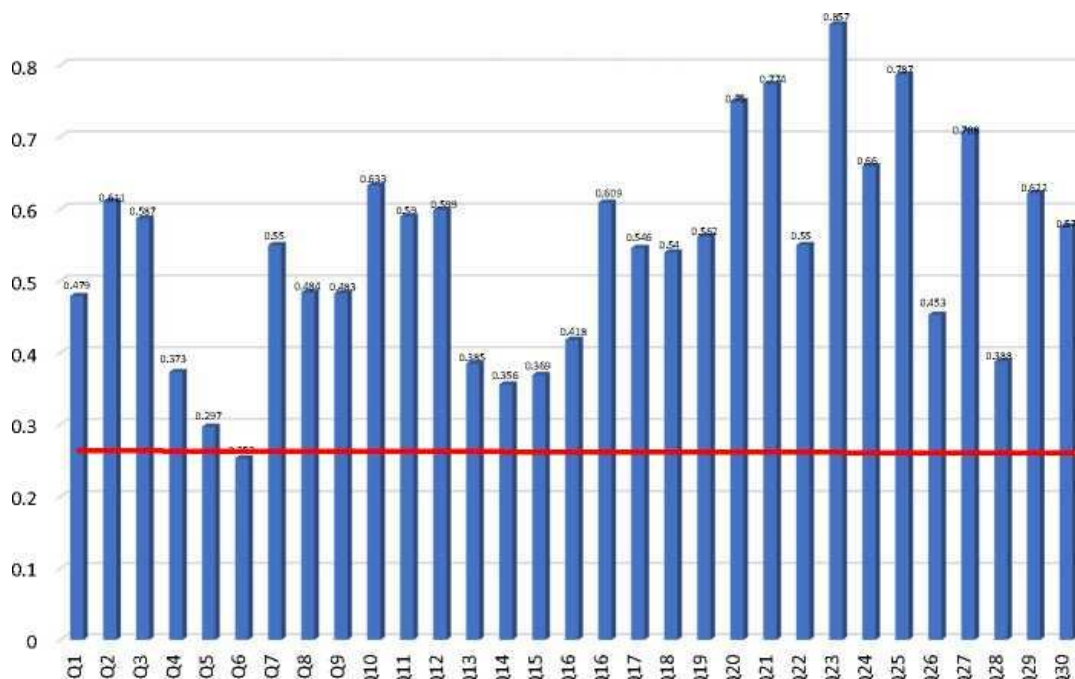


Рисунок 3.4 - Стовпчаста діаграма, що показує значення кореляції інструменту USE

Учасниками цього дослідження були студенти бакалаврату з інформатики, які вивчали вступні курси програмування на першому курсі навчання в Університеті Усману Данфодійо, кафедра інформатики, Сокото, Нігерія. Студенти мали фундаментальні знання з інформатики, отримані у середній школі. Учасники, що склалися з 103 студентів бакалаврату (80 чоловіків та 23 жінки) віком від 16 до 20 років, оцінили гру віртуальної реальності за допомогою анкети USE (див. Додаток I). Учасники охоче погодилися взяти участь у дослідженні, щоб випробувати можливості технології віртуальної реальності в освіті з програмування.

3.3 Методика проведення дослідження

VR-додаток було встановлено на мобільний пристрій, а потім підключено до картонної гарнітури Google, яку було представлено всім учасникам цього дослідження для вивчення основних концепцій ООП. Спочатку студентам було

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

представлено 30-хвилинну презентацію контенту з використанням традиційного методу лекції про технологію віртуальної реальності та основні концепції ООП, такі як об'єкти, методи (поведінка), методи встановлення та отримання, аргументи методів та збір сміття для знищення невикористовуваних об'єктів. Також обговорювалася роль VR у навчанні, зокрема, як використовувати VR-технології для вивчення концепції ООП. Крім того, студентам було показано, як вставити мобільний пристрій у гарнітуру Google та запустити встановлену VR-гру. Під час лекції було проведено презентацію про те, як використовувати VR-гарнітуру для ходьби та навігації в різних сценах у змодельованому середовищі. Нарешті, студентам було наголошено на необхідності надавати чесний зворотний зв'язок під час заповнення анкети. Під час цього методу участі та обговорення вчителі та студенти ставили запитання та обговорювали концепцію VR для викладання та навчання. Серед запитань студентів були такі: «Що я побачу в змодельованому світі?», «Чи потрібно мені постійно тримати гарнітуру віртуальної реальності під час навігації в змодельованому середовищі?», «Чи відрізняється змодельоване середовище від реального?» і, нарешті, «Що станеться, якщо зі мною щось трапиться у віртуальному середовищі?» Лектор відповів на кожне запитання студентів так, що вони були задоволені. Під час цієї сесії запитань та обговорення, яка тривала близько 45 хвилин, викладач відповів на перше запитання, повідомивши студентів, що те, що вони побачать у змодельованому світі, відображає реальний світ.

На друге запитання лектор сказав студентам, що не обов'язково постійно тримати гарнітуру, оскільки картонна гарнітура Google розроблена таким чином, щоб вона відповідала положенню, в якому користувачеві зручно її розмістити під час навчання. На третє запитання лектор відповів, що змодельоване середовище є відображенням реального середовища, і що вони не однакові як у теорії, так і на практиці. Нарешті, на останнє запитання лектор повідомив студентам, що оскільки віртуальне середовище створює ілюзію реального середовища, все, що відбувається з людиною в цьому середовищі, не впливає на таку людину в реальному світі. Навчальний процес тривав один (1) тиждень, і кожен студент

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

грав у гру Imikode VR протягом 30 хвилин. Після завершення навчального процесу кожен студент одразу ж заповнював анкету USE, яка являє собою паперовий інструмент за 7-бальною шкалою Лікерта. Однак заповнення анкети кожним студентом зайняло приблизно 10-15 хвилин. Слід зазначити, що дані були зібрані методом вибірки для зручності, оскільки студенти, про яких йде мова, охоче погодилися взяти участь у цьому дослідженні. Крім того, студенти були в межах досяжності дослідників, а це означає, що дослідники не відчували труднощів із залученням їх до участі в дослідженні.

Студентам було доручено уважно прочитати запитання та вибрати лише один варіант із семи для кожного запитання, який їм підходить. Учасники відповіли на всі 30 запитань, включаючи запитання відкритої форми анкети, і їхні відповіді були зібрані негайно. Однак варто зазначити, що кожна відповідь була анонімною та чесною, виходячи з фактичного використання VR-гри для покращення їхньої академічної успішності з програмування. Процедура дослідження проілюстрована на рис. 3.3 та різні сцени навчального процесу на рис. 3.4 н. е.

На рис. О 3.4-й годині а) учні продемонстрували використання VR-гри в класі ; у цьому випадку вони одягли картонну гарнітуру Google, тримаючи контролер. Оскільки це був перший випадок використання такого пристрою, учні були зайняті спогляданням змодельованого світу та прекрасного оточення. Крім того, учні також дізналися про різні функції VR-гри та як у неї грати.

Дві різні групи студентів грали у VR-гру в комп'ютерній лабораторії, як зображено на рис. 3.2 б; цього разу студенти були зайняті створенням об'єктів та викликом методів для покращення своїх навичок програмування. Однак на цьому етапі.

У процесі навчання студенти вже опанували навігацію у віртуальному середовищі та різні функції VR-ігор. Використовували гарнітуру Google, щоб грати у VR-гру безпосередньо перед іншими учнями в класі. Таким чином, вони показали всім учням у класі, як одягати гарнітуру Google Cardboard, як орієнтуватися у віртуальному світі та, нарешті, як рухатися в симульованому

середовищі.

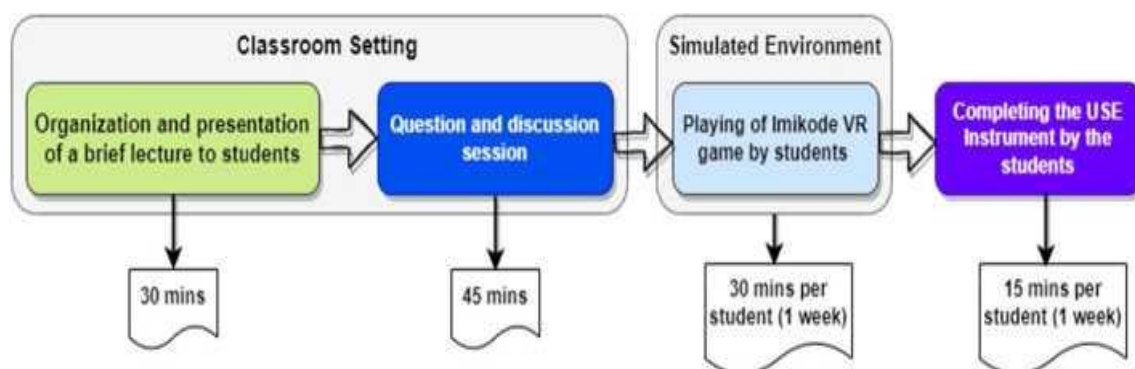


Рисунок 3.5 - Процедура дослідження, що показує час, витрачений на виконання кожної дії

Для аналізу кількісних даних використовувалися програми IBM SPSS statistics версії 21 [10] та Smart PLS для операційних систем Windows . Отримані дані були закодовані відповідно до категорій конструктів у програмах SPSS та Smart PLS. Щоб забезпечити високу якість даних, троє співавторів перевірили їх на своїх різних комп'ютерах, щоб забезпечити цілісність та узгодженість даних на різних платформах. Були використані методи описової статистики, дисперсійного аналізу (ANOVA) та множинного лінійного регресійного аналізу, включаючи дослідження f&t-тесту даних за допомогою цих інструментів. Відповіді студентів були проаналізовані за допомогою форми зворотного зв'язку інструменту USE на основі якісного аналізу контенту, а процедура, описана [10], була виконана щодо структурування контенту та тематичного аналізу. У цьому випадку відкриті форми зворотного зв'язку були прочитані кілька разів, щоб зрозуміти та класифікувати відповіді кожного студента на основі певних тем та відповідно до дослідницьких питань. Крім того, було виявлено дублювання тем серед форм зворотного зв'язку, які були класифіковані відповідно за такими поняттями, як корисність, простота використання, легкість навчання та задоволення. Деякі відповіді, які не зовсім відповідали цим поняттям, були виявлені, оскільки вони більше схожі на рекомендації та/або недоліки; отже, такі відповіді були профільовані та класифіковані за темою рекомендацій та

недоліків інструменту віртуальної реальності. Аналіз показав, що зі 103 студентів, які відповіли на відкриту форму зворотного зв'язку, 10 студентів або забули, або навмисно вирішили не відповідати на відкриті запитання в анкеті USE. Решта 93 студентів надали різні коментарі, починаючи від деяких позитивних коментарів (88) і закінчуючи рекомендацією (1), тоді як інші вказали на деякі недоліки технології (4). Нарешті, ці теми були ретельно відібрані, а потім сегментовані на окремі блоки.

3.4 Аналіз отриманих результатів

Перед аналізом відповідей студентів на анкету за допомогою описової статистики було проведено вимірювання зручності використання освітньої гри віртуальної реальності Imikode . Цей показник вимірювання відображає ступінь прийняття або неприйняття гри Imikode VR користувачами, тому в цьому дослідженні середнє значення кожної змінної було використано для опису результатів вимірювання зручності використання , як запропонував Nielsen (1994). Також було використано середній бал кожної змінної (див. Таблицю 1) шляхом визначення середнього значення бали та групування їх відповідно до представлення кожної змінної за семибальною шкалою Лікерта. Однак напрямок відповідей за цією шкалою показує рівень прийняття або відхилення, як зазначили [19]/ У цьому випадку було помічено, що середнє значення кожної змінної знаходилося в межах прийняттого діапазону, оскільки жодне значення не було нижче 5,0, отже, кожне значення наближається до піку твердого прийняття. Це означає, що з підвищенням кожного середнього балу після досягнення порогу прийняття 5,0 рівень прийняття ще більше зростає (див. Таблицю 3.1). Крім того, подібно до [22] семибальна шкала Лайкерта була перетворена на дві категорії «біноміальних даних» – прийняття та відхилення відповідно до відповідей учасників «згода» чи «незгода», де оцінка 5 (скоріше згоден), 6 (згоден) та 7 (повністю згоден) була згрупована в категорії прийняття. Аналогічно було зроблено для категорії відхилення, де оцінки 1 (категорично не

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

згоден), 2 (не згоден) та 3 (скоріше не згоден) були об'єднані та класифіковані в категорії відхилення. Однак оцінка нейтральності була виключена з аналізу, оскільки вона означає невизначеність або нейтральність. Причина, через яку оцінка Лайкерта була перетворена та узгоджена до балів від 0 до 100, полягала в тому, щоб дозволити глибший аналіз даних з біноміальної точки зору, де 0-49 означає нижче середнього, а 50-100 – прийнятне значення (Debevc & Bele, 2008). З таблиці можна спостерігати тенденції таких конструктів, як корисність, простота використання, легкість навчання та задоволення. Усі конструкції перевищили значення балу 50, таким чином задовольняючи критерії прийняття.

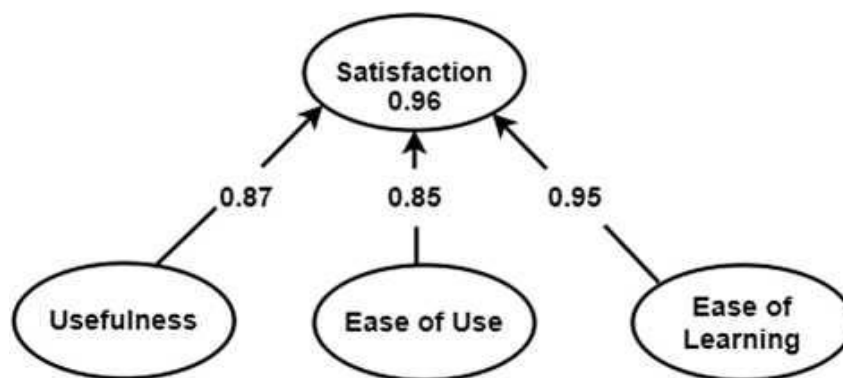


Рисунок 3.6 - Побудована модель інструменту USE, що показує ступінь зручності використання

Крім того, якщо побіжно розглянути середній бал за чотирма змінними, то він становить 90,94, що перевищує 50. На основі цих результатів було зроблено висновок, що студенти вважають навчальну гру Imikode VR дуже корисною для вивчення об'єктно-орієнтованого програмування. Аналогічно, з оцінки від 0 до 100 в анкеті USE у таблиці 1 було виявлено, що 90,94 відсотка користувачів були задоволені її використанням як засобу підвищення своїх навичок програмування. Однак, наслідком цього є те, що, наприклад, коли в класі 100 студентів, понад 90 відсотків студентів, які грають в навчальну гру Imikode VR, будуть задоволені нею як чудовим засобом навчання програмуванню. Ми представляємо модельну конструкцію на рис. 10 що показує ступінь зручності використання гри Imikode VR.

Крім того, що стосується розподілу балів, як показано в таблиці 8 було помічено, що бали мали негативну асиметрію для всіх змінних, що означає, що лівий хвіст розподілу довший відносно правого хвоста, і тому не симетричний. Крім того, жоден зі студентів не відповів «Н/Д». Однак це можна пояснити тим, що студентів правильно спрямовували щодо різних аспектів анкети USE, включаючи навчання за допомогою гри Imikode VR.

Відкритий зворотний зв'язок студента

Зворотній зв'язок від користувачів, як запропоновано [26], передбачає надання інформації про рівень продуктивності або розуміння користувачем певного інструменту, що здатне спрямовувати майбутній розвиток. У цьому дослідженні, окрім кількісних відповідей, студенти надавали свої відповіді у формі відкритого питання у формі кількісного зворотного зв'язку. Ці питання дозволяють студентам надавати коментарі або пропозиції щодо того, що вони відчують на основі свого досвіду гри у віртуальній реальності. Зі 103 студентів, які взяли участь у цьому дослідженні, 10 студентів не надали відгуків про гру у віртуальній реальності, а решта 93 студенти надали як позитивні відгуки, так і деякі важливі рекомендації щодо покращення системи навчання у віртуальній реальності, включаючи деякі недоліки технології. Зворотній зв'язок класифіковано за змінними.

З наведених вище коментарів зрозуміло, що студентам сподобалася гра у VR-гру, і вона значною мірою допомогла їм розширити свої знання з об'єктно-орієнтованого програмування, незважаючи на недоліки технології. Цікаво відзначити, що не всі відповіді студентів були позитивними, як видно з таблиці 9, де лише один студент надав рекомендацію, і дуже мало студентів вказали на деякі недоліки технології.

Передумови для проведення цього аналізу були отримані з класичної моделі припущень для тестованих змінних [24]. Відповідно, проведення цього тесту необхідне для того, щоб зробити результат гідним довіри та дати іншим дослідникам більше впевненості. Ці тести включають тест на нормальність, тест на мультиколінеарність та тест на гетероскедастичність. Отже, необхідно

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

переконатися, що використані змінні мають нормальний розподіл та позбавлені мультиколінеарності та гетероскедастичності відповідно. У наступних розділах було показано, як досягається класична модель припущень для тестованих змінних.

У множинній лінійній регресії перше припущення серед класичних припущень вимагає проведення перевірки нормальності заданих змінних, щоб перевірити, чи розподілені змінні нормально. Цього можна досягти, уважно вивчивши гістограму та графіки даних. У цьому дослідженні, поверховий погляд на рис. На рисунку 3.7 показано, що графік має приблизно дзвоноподібну форму та симетричний відносно середнього значення. Більшість значень даних обертаються навколо середнього значення, що свідчить про наявність нормального розподілу. На рис. 3.8, графіки даних показують незначне відхилення від лінії апроксимації методом найменших квадратів, що вказує на звичайний нормальний розподіл.

Друге припущення в моделі множинної регресії полягає в тому, що в даних немає мультиколінеарності. Дані вважаються вільними від мультиколінеарності тоді і тільки тоді, коли кожна незалежна змінна має толерантність (T) вище 0,1, а також має коефіцієнт інфляції дисперсії (VIF) нижче 10 (Kim, 2019). Однак у цьому дослідженні це видно з таблиці 10 що VIF для корисності (1,918), простоти використання (2,489) та простоти навчання (2,481) був нижчим за 10. Аналогічно, T для корисності (0,521), простоти використання (0,402) та простоти навчання (0,403) був більшим за 0,1. Отже, ми робимо висновок, що ці змінні позбавлені мультиколінеарності, що означає, що незалежні змінні є справді незалежними. Стіл На рисунку 3.7 представлено короткий виклад моделі, який показує мінливість задоволеності студентів –використанням VR-гри як інструменту для навчання програмуванню, що пояснюється корисністю, простотою використання та легкістю вивчення змінних. У короткий виклад моделі показано, як залежна змінна використовується для вимірювання близькості даних до підібраної лінії регресії.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

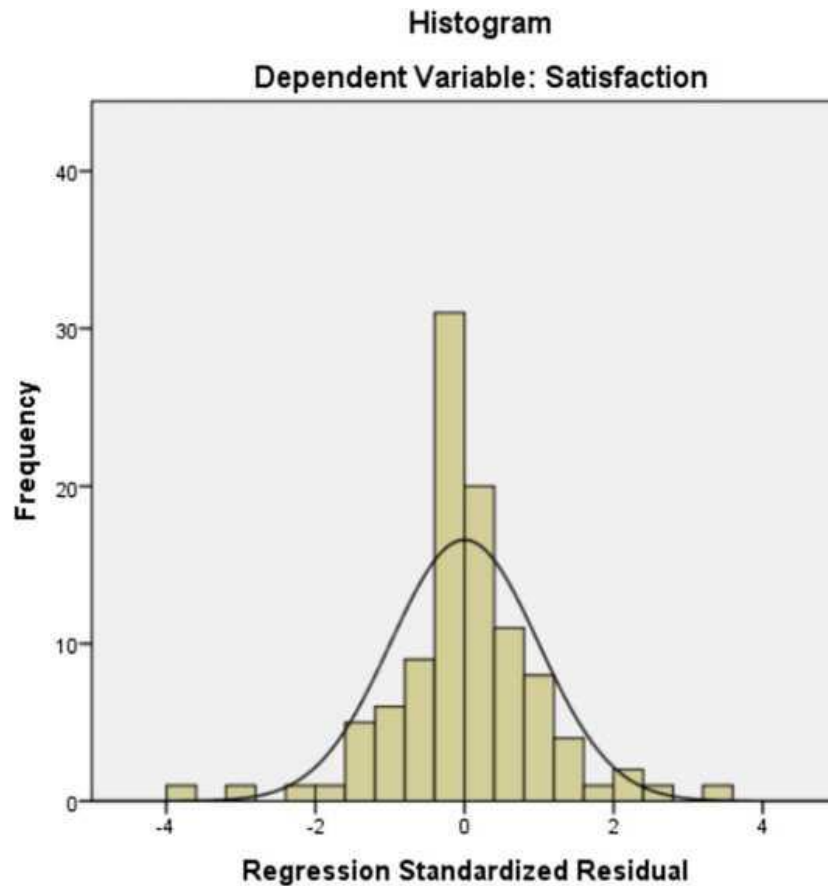


Рисунок 3.7 - Гістограма для тесту на нормальність

R² показує, що 70,1% мінливості задоволеності студентів пояснюється змінними простотою використання, легкістю навчання та корисністю.

Нарешті, було проведено тест на гетероскедастичність, щоб дослідити, чи задовольняють дані гомоскедастичність, яка є останнім припущенням у множинній лінійній регресії. Однак у цьому випадку припускалося, що в даних немає гетероскедастичності. Найкращий спосіб перевірити це – за допомогою діаграми розсіювання, де стандартизований залишок регресується до стандартизованого прогнозованого значення. Слід зазначити, що коли гомоскедастичність присутня або коли припущення не може бути відхилено, оскільки в даних не буде закономірності. На рис. 3.9 видно, що дані щільно упаковані з одного кінця, а потім починаються розширюючись у міру просування до іншого кінця, що представляє типове традиційне відображення гомоскедастичності в регресійній моделі.

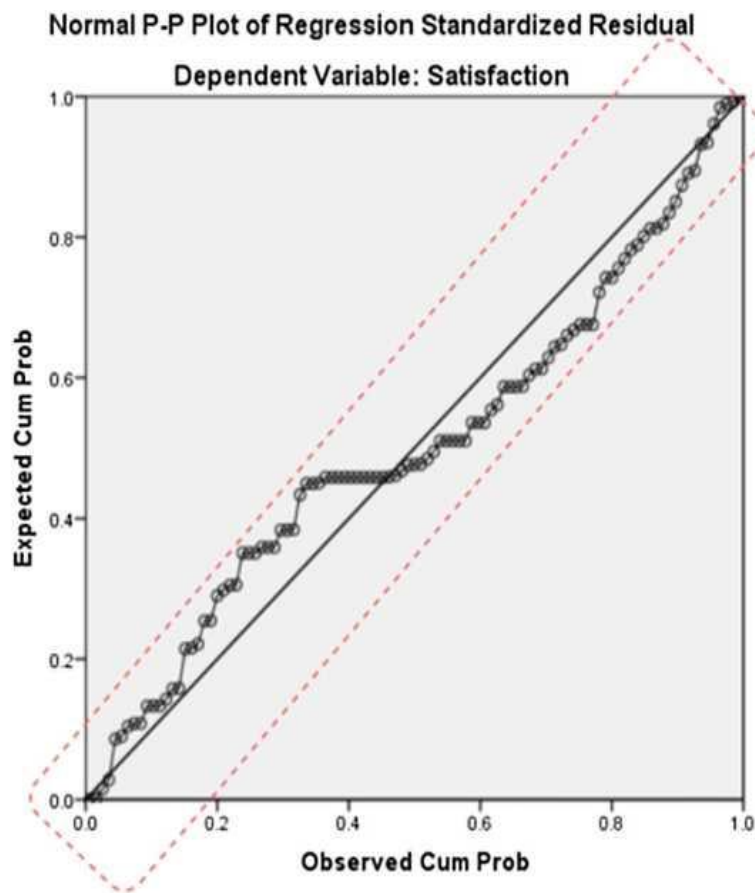


Рисунок 3.8 - Графік даних для тесту на нормальність

Було проведено множинну лінійну регресію, щоб показати лінійний зв'язок між задоволеністю студентів легкістю використання, легкістю навчання та корисністю навчальних VR-ігор. Харіянто та ін. (2020) стверджували, що метою використання множинного лінійного регресійного аналізу є розуміння зв'язку між двома або більше незалежними змінними та однією залежною змінною. Отже, в цьому дослідженні було досліджено f-тестовий розподіл, щоб зрозуміти змінні легкості використання, легкості навчання та корисності. Аналогічно, було проведено частковий t-тестовий розподіл, щоб встановити внесок кожної з незалежних змінних у задоволеність студентів використанням VR-інструменту для навчання.

Змн.	Арк.	№ докум.	Підпис	Дата

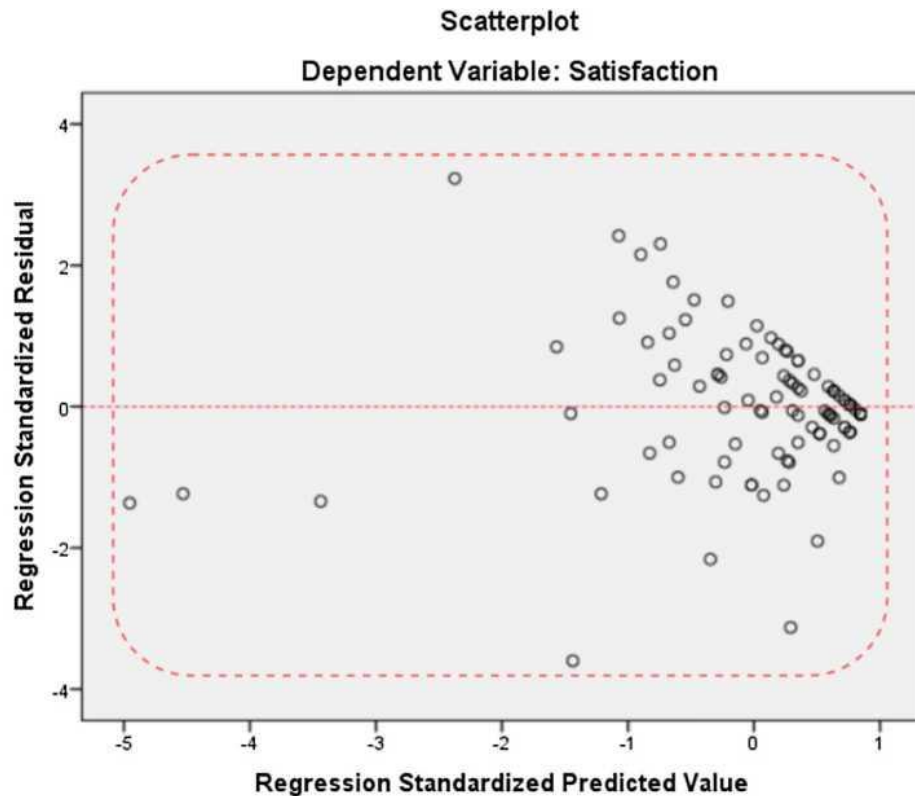


Рисунок 3.9 - Діаграма розсіювання гомоскедастичності

Попри значні досягнення у використанні VR-ігор для покращення навичок програмування у студентів, а також успішну валідацію дослідницького інструменту, який може бути застосований іншими науковцями, все ще зберігаються певні обмеження та виклики, що вимагають подальшого аналізу й розвитку. Хоча кількість учасників у дослідженні виявилася статистично достатньою для отримання первинних висновків, важливо продовжити подібні експерименти із залученням ширшого кола студентів з інших університетів Нігерії. Це дозволить перевірити узагальненість результатів і точніше оцінити ефективність VR-підходу до навчання програмуванню в різних академічних і соціально-економічних контекстах.

Одна з ключових проблем, з якою зіштовхнулися дослідники, полягає в низькому рівні обізнаності щодо потенціалу VR-технологій у галузі освіти. З огляду на це, було запропоновано організувати інформаційні кампанії та тренінги у студентських кампусах з метою популяризації VR як сучасного інструменту цифрового навчання. Водночас, незважаючи на високу оцінку користувачами освітньої ефективності VR-гри, існує необхідність у вирішенні

низки технічних та логістичних труднощів, пов'язаних із впровадженням таких технологій у навчальний процес.

Зокрема, серйозною перешкодою стала обмежена доступність VR-обладнання. У дослідженні, що проводилося в Університеті Усману Данфодійо, студентам було доступно лише дві гарнітури VR, що значно ускладнило організацію навчального процесу. Проте, замість скорочення експерименту, команда дослідників організувала поетапне проходження гри всіма 103 учасниками протягом одного тижня, що свідчить про високу організаційну ефективність, але водночас підкреслює проблему нестачі технічних ресурсів.

Варто зазначити, що фокус дослідження був спрямований на педагогічну доцільність та зручність використання VR-ігор у навчанні ООП, однак самі технологічні аспекти VR-гарнітур, включно з варіантами мобільного VR та особливостями апаратного забезпечення, не розглядалися як об'єкти детального аналізу. Тим не менш, було встановлено, що обмеження в доступності гарнітур, а також відсутність належної технічної підтримки користувачів, можуть мати негативний вплив на ефективність навчання та подальше масштабування подібних ініціатив.

Отримані результати демонструють високий рівень задоволеності VR-грую серед учасників експерименту та підтверджують її позитивний вплив на формування навичок об'єктно-орієнтованого програмування. На основі цього було запропоновано кілька напрямів для подальших досліджень. Насамперед, варто розширити функціональність VR-гри шляхом включення більш складних концепцій ООП, таких як поліморфізм, успадкування, інтерфейси, аплети, рекурсія та структури даних. Крім того, перспективним є дослідження інтеграції елементів штучного інтелекту у VR-середовище, зокрема аналіз ролі AI-агентів у персоналізації навчання, адаптації складності та автоматичному поясненні помилок під час програмування.

Важливо також розглянути міждисциплінарні перспективи застосування VR у вищій освіті. Наприклад, використання VR-ігор може бути адаптоване для вивчення інших складних дисциплін – фізики, хімії, анатомії – де візуалізація

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

просторових структур і процесів має вирішальне значення. Такий підхід сприятиме формуванню глибших когнітивних моделей та розвитку критичного мислення.

З точки зору стійкого розвитку VR-освіти, майбутні дослідження мають зосередитися також на економічній доцільності та моделях фінансування впровадження таких технологій, зокрема на партнерстві з комерційними виробниками VR-рішень, залученні донорських організацій або створенні відкритих платформ з відкритим кодом.

Таким чином, хоча перші результати дослідження свідчать про високу ефективність VR-ігрового підходу до навчання програмуванню, подальший розвиток цієї галузі потребує вирішення як педагогічних, так і технологічних викликів, розширення аудиторії користувачів, удосконалення програмного забезпечення, посилення технічної бази та вивчення міждисциплінарних і соціоекономічних аспектів застосування віртуальної реальності в освіті.

3.5 Висновок по розділу

Дослідження показало, що гра у віртуальній реальності Imikode VR є ефективним інструментом для покращення навчання об'єктно-орієнтованому програмуванню, зокрема завдяки її простоті навчання, яка мала найбільший вплив на задоволення користувачів. Використана анкета USE підтвердила надійність і валідність оцінки взаємозв'язків між корисністю, зручністю, легкістю навчання та задоволенням.

Методика дослідження продемонструвала, що інтеграція VR-гри Imikode у навчальний процес є ефективною, забезпечуючи високу залученість студентів, які позитивно оцінили її корисність, простоту використання та навчання, що підтверджено як кількісним, так і якісним аналізом зібраних даних.

Аналіз результатів показав високий рівень задоволеності студентів використанням VR-гри Imikode для вивчення ООП, що свідчить про ефективність такого підходу

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

ВИСНОВКИ

Дослідження, проведене в рамках цієї роботи, підкреслює значний потенціал програмування віртуальної реальності (VR) як інструменту для підвищення ефективності навчання комп'ютерному програмуванню, зокрема об'єктно-орієнтованому програмуванню (ООП). Аналіз теоретичних основ показав, що виклики вивчення програмування, такі як складність засвоєння абстрактних концепцій і низька мотивація студентів, можуть бути ефективно подолані через використання VR-технологій. Застосування віртуальної реальності в освіті, зокрема через гейміфіковані середовища, сприяє кращому розумінню ООП, тоді як інструменти візуального програмування, такі як Scratch або Blockly, створюють основу для інтуїтивного освоєння програмування.

Другий розділ, присвячений VR-грі як засобу навчання, продемонстрував, що інтерактивне ігрове середовище, розроблене з використанням платформ, таких як Unity або Unreal Engine, забезпечує мотивацію та практичний досвід для студентів. Функціонування гри, засноване на моделюванні концепцій ООП, таких як класи, об'єкти та спадкування, дозволяє студентам експериментувати з кодом у реальному часі. Модифікація та вдосконалення навчального ігрового середовища, зокрема через адаптацію інтерфейсів і додавання нових сценаріїв, підвищують гнучкість і доступність навчання. Методологія проєктування та оцінювання навчального впливу підтвердила, що VR-гра сприяє покращенню розуміння програмування та підвищенню залученості студентів.

Третій розділ, що аналізує дизайн і контекст дослідження, показав, що ретельний вибір інструментів, таких як Oculus Quest або Unity, і технічних засобів є критично важливим для створення ефективних VR-додатків. Оцінка надійності та валідності методики підтвердила достовірність результатів, тоді як методика проведення дослідження, що включає експерименти та опитування, дозволила зібрати об'єктивні дані. Аналіз отриманих результатів виявив

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

статистично значуще покращення знань і навичок студентів у групах, які використовували VR-гру, порівняно з традиційними методами навчання.

Узагальнюючи, програмування віртуальної реальності є перспективним інструментом для трансформації освіти з комп'ютерного програмування, що сприяє мотивації, інтерактивності та глибшому розумінню ООП. Результати дослідження можуть бути використані викладачами, розробниками освітнього програмного забезпечення та дослідниками для створення інноваційних навчальних рішень. Перспективи подальших досліджень включають інтеграцію штучного інтелекту для персоналізації VR-навчання, розробку кросплатформних VR-додатків і вдосконалення методик оцінки їхнього впливу на навчальний процес. Ці напрямки сприятимуть розвитку більш доступних і ефективних освітніх технологій у майбутньому.

					ДРБ.ІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Adobe. Virtual Reality: Principles and Applications. *adobe.com*, 2023. —
Режим доступу: <https://www.adobe.com/ua/products/substance3d/learn/virtual-reality>
2. Anthes, C., García-Hernández, R. J., Wiedemann, M., & Kranzlmüller, D. State of the Art of Virtual Reality Technology. *IEEE Aerospace Conference Proceedings*, 2016, 1–19. - Режим доступу: <https://doi.org/10.1109/AERO.2016.7500674>
3. Blascovich, J., & Bailenson, J. *Infinite Reality: Avatars, Eternal Life, New Worlds, and the Dawn of the Virtual Revolution*. - New York, NY: William Morrow, 2011. - 320 с.
4. Craig, A. B., Sherman, W. R., & Will, J. D. *Developing Virtual Reality Applications: Foundations of Effective Design*. - Burlington, MA: Morgan Kaufmann, 2009. - 448 с. - Режим доступу: <https://www.elsevier.com/books/developing-virtual-reality-applications/craig/978-0-12-374943-7>
5. Designtalk. Storyboard VR: Rapid Prototyping for Virtual Reality. *designtalk.club*, 2016. - Режим доступу: <https://designtalk.club/storyboard-vr>
6. DOU. Virtual Reality in Education: Tools and Trends. *dou.ua*, 2023. -
Режим доступу: <https://dou.ua/lenta/articles/vr-in-education>
7. Epic Games. Unreal Engine 5: VR Development Guide. *unrealengine.com*, 2025. - Режим доступу: <https://www.unrealengine.com/en-US/developer-resources/vr>
8. Greengard, S. *Virtual Reality*. - Cambridge, MA: MIT Press, 2019. - 264 с. - Режим доступу: <https://mitpress.mit.edu/books/virtual-reality>
9. Heim, M. *Virtual Realism*. - Oxford, UK: Oxford University Press, 2000. - 264 с. - Режим доступу: <https://doi.org/10.1093/acprof:oso/9780195138740.001.0001>
10. Imena. History of Virtual Reality: From 19th Century to Present. *imena.ua*, 2019. - Режим доступу: <https://www.imena.ua/blog/vr-history>

					ДРБ.ІІ - 38.00.00.000 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

11. IT Enterprise. Virtual Reality: Best Practices in Industry. *it.ua*, 2024. - Режим доступу: <https://www.it.ua/knowledge-base/virtual-reality-best-practices>
12. Jerald, J. *The VR Book: Human-Centered Design for Virtual Reality*. - New York, NY: ACM Books, 2015. - 618 с. - Режим доступу: <https://doi.org/10.1145/2792790>
13. Kyivstar Business Hub. VR, AR, and IR: Applications in Business. *hub.kyivstar.ua*, 2025. - Режим доступу: <https://hub.kyivstar.ua/vr-ar-business>
14. Lanier, J. *Dawn of the New Everything: Encounters with Reality and Virtual Reality*. - New York, NY: Henry Holt and Co., 2017. - 368 с. - Режим доступу: <https://us.macmillan.com/books/9781627794091>
15. Lemon School. Perspectives of Virtual and Augmented Reality Technologies. *lemon.school*, 2024. - Режим доступу: <https://lemon.school/blog/vr-ar-perspectives>
16. MOYO. How to Use VR Headsets: A Guide for Beginners. *moyo.ua*, 2022. - Режим доступу: <https://www.moyo.ua/news/vr-headsets-guide>
17. Naurok. Immersive Technologies in Education: VR and AR. *naurok.com.ua*, 2024. - Режим доступу: <https://naurok.com.ua/vr-ar-education>
18. Osvitoria. Virtual and Augmented Reality in Education. *osvitoria.media*, 2019. - Режим доступу: <https://osvitoria.media/vr-ar-learning>
19. Parisi, T. *Programming 3D Applications with HTML5 and WebGL: 3D Animation and Visualization for Web Pages*. - Sebastopol, CA: O'Reilly Media, 2014. - 404 с. - Режим доступу: <https://www.oreilly.com/library/view/programming-3d-applications/9781449363918>
20. Yasinska, O. A., Slyusar, M. T. (2019). Virtual and augmented reality in the library. URL:<http://conference.nbuu.gov.ua/report/view/id/836>.
21. Woodford, C. (2021). Virtual reality. URL: <https://www.explainthatstuff.com/virtualreality.html>.
22. VIAR (2022). Virtual Reality In Education – How Are Schools Using VR? URL:<https://www.viar360.com/education-schools-using-virtual-reality/>.

					ДРБ.ІІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

23. FinancesOnline (2022). 74 Virtual Reality Statistics You Must Know in 2021/2022: Adoption, Usage &Market Share. URL: <https://financesonline.com/virtual-reality-statistics/>.

24. Galadzhii, A. (2020) The Future is Now: Best 10 Educational VR Apps. URL:<https://litslink.com/blog/the-future-is-now-best-10-educational-vr-application>

25. Radiosvoboda. Opportunities of Virtual Reality Technologies. *radiosvoboda.org*, 2017. - Режим доступу: <https://www.radiosvoboda.org/a/vr-opportunities/28901123.html>

26. Shen. Virtual Reality in Facility Management. *shen.ua*, 2020. - Режим доступу: <https://shen.ua/vr-facility-management>

27. TeachHub. Virtual Reality: Principles and Educational Benefits. *teach-hub.com*, 2020. - Режим доступу: <https://teach-hub.com/vr-education>

28. Unity Technologies. Unity VR Development: Getting Started. *docs.unity3d.com*, 2024. - Режим доступу: <https://docs.unity3d.com/Manual/VR.html>

29. UzhNU. Virtual Reality in Medicine: Current and Future Applications. *uzhnu.edu.ua*, 2023. - Режим доступу: <https://www.uzhnu.edu.ua/vr-medicine>

30. Wikipedia. Віртуальна реальність. *uk.wikipedia.org*, 2024. - Режим доступу: https://uk.wikipedia.org/wiki/Віртуальна_реальність

31. Zyphron, A., & Koval, A. VR in Education: Developing Interactive Learning Environments. *Journal of Educational Technology Development*, 2023, 19(4), 33–47. - Режим доступу: <https://doi.org/10.1007/s11423-023-10245-8>

					ДРБ.ІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: " Програмування віртуальної реальності"

Обсяг пояснювальної записки: 59 аркушів

Дата закінчення дипломної роботи 10 червня 2025р.

Підпис студента _____

					ДРБ.ІІ - 38.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63