

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

Кафедра інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему

**«СИСТЕМНИЙ АНАЛІЗ ТА ОПТИМІЗАЦІЯ ЕФЕКТИВНОСТІ
ВЕБПОРТАЛУ НА ОСНОВІ ПОВЕДІНКИ КОРИСТУВАЧА»**

Виконала: здобувач 2 курсу

Рівень магістр

Галузь знань 12 «Інформаційні технології»,
спеціальність 124 «Системний аналіз»

Куклінова Софія Іванівна

Керівник: к.т.н., доцент

Манаков Сергій Юрійович

Рецензент: к.пед.н., доцент

Лобода Юлія Геннадіївна

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
Факультет **кібербезпеки та інформаційних технологій**
Кафедра **інформаційних технологій**
Галузь знань: **12 Інформаційні технології**
Спеціальність: **124 Системний аналіз**
Назва освітньої програми: Аналіз та безпека даних

ЗАТВЕРДЖУЮ

Завідувач кафедри інформаційних технологій
доцент Н.І. Логінова

_____ «___» _____ 20__ року

З А В Д А Н Н Я
на кваліфікаційну (магістерську) роботу
здобувачу Кукліновій Софії Іванівні

1. Тема роботи: «Системний аналіз та оптимізація ефективності вебпорталу на основі поведінки користувача»

Керівник роботи: к.т.н., доцент Манаков Сергій Юрійович
затверджені наказом НУ ОЮА від «10» жовтня 2025 року № 3183-06

2. Строк подання здобувачем роботи «25» листопада 2025 року

3. Дата видачі завдання «02» червня 2025 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Розділ 1. Теоретичні основи аналізу та оптимізації вебпорталів	20.07.2025	
2	Розділ 2. Об'єкт дослідження та інфраструктура аналізу	21.09.2025	
3	Розділ 3. Системний аналіз користувацької поведінки та оптимізація ефективності вебпорталу	09.11.2025	
4	Оформлення пояснювальної записки	23.11.2025	
5	Подання роботи на кафедру	25.11.2025	

Здобувач _____ Софія КУКЛІНОВА _____
(підпис) (прізвище та ініціали)

Керівник роботи _____ Сергій МАНАКОВ _____
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Куклінова С.І. «Системний аналіз та оптимізація ефективності вебпорталу на основі користувацької поведінки». – Кваліфікаційна робота магістра за спеціальністю 124 «Системний аналіз», освітня програма «Аналіз та безпека даних».

Кваліфікаційна робота викладена на 53 сторінках основного тексту і 94 сторінки загального тексту, містить 3 розділи, 12 рисунків, 8 таблиць, 43 використаних джерела.

Метою роботи є комплексна розробка та аналітичне обґрунтування підходів до системного аналізу й оптимізації вебпорталу на основі поведінкових метрик користувачів і сучасних інструментів вебаналітики.

Об'єктом дослідження є процес оптимізації ефективності інтерактивного вебпорталу для рецензування відеоігор.

Предмет дослідження – методи, показники та інструменти системного аналізу ефективності вебпорталів.

У роботі було розглянуто вебпортал, розроблений у бакалаврській роботі та реалізовано симуляцію користувацької активності за допомогою Python-скриптів і Google Analytics 4. Отримані дані були оброблені та візуалізовані у вигляді графіків, теплових карт і кругових діаграм. На основі аналізу визначено поведінкові моделі користувачів, проблемні зони у навігації та надані пропозиції щодо покращення UX/UI дизайну.

UX/UI ДИЗАЙН, ВЕБПОРТАЛ, КОРИСТУВАЦЬКА ПОВЕДІНКА, АНАЛІТИКА, СИМУЛЯЦІЯ ТРАФІКУ, GOOGLE ANALYTICS, PYTHON.

ABSTRACT

Kuklinova S.I. "System analysis and optimization of web portal efficiency based on user behavior". – Master's qualification work in specialty 124 "System analysis", educational program "Data analysis and security".

The qualification work is presented on 53 pages of the main text and 94 pages of the general text, contains 3 sections, 12 figures, 8 tables, 43 sources used.

The purpose of this work is a comprehensive development and analytical justification of approaches to system analysis and optimization of a web portal based on user behavioral metrics and modern web analytics tools.

The object of the study is the process of optimizing the efficiency of an interactive web portal for reviewing video games.

The subject of the study is methods, indicators and tools for system analysis of web portal efficiency.

The work considered a web portal developed in a bachelor's thesis and implemented a simulation of user activity using Python scripts and Google Analytics 4. The obtained data was processed and visualized in the form of graphs, heat maps and pie charts. Based on the analysis, user behavioral models, problem areas in navigation were identified and suggestions for improving UX/UI design were provided.

UX/UI DESIGN, WEB PORTAL, USER BEHAVIOR, ANALYTICS, TRAFFIC SIMULATION, GOOGLE ANALYTICS, PYTHON.

ЗМІСТ

ВСТУП	6
1 ТЕОРЕТИЧНІ ОСНОВИ АНАЛІЗУ ТА ОПТИМІЗАЦІЇ ВЕБПОРТАЛІВ	8
1.1 Поняття вебпорталів та їх класифікація	8
1.2 Метрики ефективності та користувацька поведінка	10
1.3 Інструменти та методи системного аналізу вебресурсів	13
2 ОБ’ЄКТ ДОСЛІДЖЕННЯ ТА ІНФРАСТРУКТУРА АНАЛІЗУ	16
2.1 Вебпортал як об’єкт системного аналізу	16
2.2 Система збору даних користувацької поведінки.....	22
2.3 Формування експериментального середовища.....	26
3 СИСТЕМНИЙ АНАЛІЗ КОРИСТУВАЦЬКОЇ ПОВЕДІНКИ ТА ОПТИМІЗАЦІЯ ЕФЕКТИВНОСТІ ВЕБПОРТАЛУ	34
3.1 Збір, обробка та аналіз даних користувацької взаємодії	34
3.2 Виявлення проблемних зон та поведінкові моделі	46
3.3 Практичні рекомендації щодо оптимізації UX/UI.....	50
ВИСНОВКИ.....	57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	59
ДОДАТКИ.....	64
Додаток А. Програмний код	64
Додаток Б. UML діаграми досліджуваного вебпорталу	87
Додаток В. Копії апробаційних матеріалів	90

ВСТУП

У сучасних реаліях вебпортали виступають не лише засобом представлення інформації, а й ключовим інструментом інформаційного обміну, комунікації та взаємодії між користувачами і цифровими сервісами. Сучасний вебпортал є не лише набором сторінок з контентом, а й динамічною екосистемою, котра повинна адаптуватися до потреб аудиторії, забезпечити швидку навігацію, інтуїтивний інтерфейс та високий рівень залучення користувачів. Від ефективності їхньої роботи, зручності навігації та якості користувацького досвіду залежить рівень залучення, лояльність аудиторії та конкурентоспроможність. А розвиток інструментів аналітики, зокрема і Google Analytics 4, дає змогу аналізувати моделі поведінки користувачів, виявляти слабкі місця у структурі сайту та прогнозувати реакцію відвідувачів на контент. Таким чином, оптимізація вебпорталів стає необхідним етапом їх життєвого циклу.

Актуальність дослідження визначається потребою у комплексному підході до оцінювання ефективності вебресурсів, тобто в поєднанні технічних, поведінкових та аналітичних аспектів, що створить базу для підвищення ефективності вебресурсів і формування стабільної користувацької аудиторії.

Метою роботи є комплексна розробка та аналітичне обґрунтування підходів до системного аналізу й оптимізації вебпорталу на основі поведінкових метрик користувачів і сучасних інструментів вебаналітики.

Об'єктом дослідження є процес оптимізації ефективності інтерактивного вебпорталу для рецензування відеоігор.

Предмет дослідження – методи, показники та інструменти системного аналізу ефективності вебпорталів.

Методика дослідження базується на поєднанні системного аналізу, емпіричних спостережень, інструментальної вебаналітики та моделюванні даних.

Для досягнення поставленої мети необхідно вирішити такі **завдання**:

1. Проаналізувати теоретичні основи функціонування вебпорталів та їхню класифікацію за функціональним призначенням.
2. Розглянути сучасні метрики ефективності вебресурсів і фактори, що впливають на поведінку користувачів.
3. Оцінити інструменти системного аналізу вебпорталів – Google Analytics 4, Python, UX-дослідження.
4. Проаналізувати структуру та функціональну модель вебпорталу, який слугуватиме об'єктом дослідження [1].
5. Здійснити збір і обробку даних користувацької взаємодії, сформувати поведінкові моделі та виявити проблемні зони сайту.
6. Розробити практичні рекомендації щодо оптимізації UX/UI шляхом аналітичного прогнозування метрик.
7. Побудувати візуалізації результатів аналізу у середовищі Python.
8. Порівняти результати до та після теоретичної оптимізації й сформулювати рекомендації щодо подальшого розвитку вебпорталу.

Наукова новизна роботи полягає у розробці комплексного підходу до аналізу та оптимізації веб-порталів, що поєднує методи системного аналізу, вебаналітики та імітаційного моделювання користувацької поведінки. Уперше запропоновано модель симуляції користувацького трафіку з використанням мови програмування Python та інтеграції з Google Analytics 4 через Measurement Protocol, яка дозволяє формувати аналітичні дані для дослідження ефективності веб-порталу навіть без залучення реальних користувачів. Це забезпечує можливість попереднього тестування UX/UI-рішень і прогнозування змін у поведінкових метриках після оптимізації.

Результати роботи були апробовані на науковій конференції [43].

1 ТЕОРЕТИЧНІ ОСНОВИ АНАЛІЗУ ТА ОПТИМІЗАЦІЇ ВЕБПОРТАЛІВ

1.1 Поняття вебпорталів та їх класифікація

У сучасному суспільстві вебпортали займають одне з ключових місць для пошуку інформації, надання онлайн-послуг та комунікації. Такий портал є багатофункціональним вебресурсом, котрий забезпечує доступ до різних інформаційних, комунікаційних та сервісних ресурсів через єдиний інтерфейс. Основною метою вебпорталу є створення зручного середовища для користувача, де можна отримати необхідні дані, а також взаємодіяти з іншими користувачами та використовувати набори послуг без потреби в сторонніх ресурсах.

Такий термін як вебпортал у контексті Інтернет-технологій вперше з'явився у 1990-х роках для позначення унікальних ресурсів, котрі використовувалися для «входу» в Інтернет. Такі портали як Yahoo!, MSN, AOL відносяться до одних із перших порталів, котрі поєднали в собі пошук, новини та електронну пошту. З розвитком технологій були розвинені і вебпортали, котрі перестали бути лише агрегаторами інформації та перетворилися на системи з персоналізованим контентом, складною логікою та інтеграцією із зовнішніми платформами.

Основними ознаками вебпорталу є відмінність від звичайних сайтів масштабом функціоналу та ступенем персоналізації користувацького досвіду. До його основних характеристик можна віднести централізований доступ до інформації та об'єднання різноманітних інформаційних ресурсів на одній платформі, єдиний користувацький інтерфейс, персоналізацію системи, інтеграцію сервісів, систему автентифікації та керування доступом та інтерактивність [2].

Отже, як зазначалося вище вебпортал є складною інформаційно-комунікаційною системою, що поєднує в собі технологічну платформу, базу

даних, програмну логіку, набір інтегрованих сервісів та інтерфейс користувача.

Існує безліч видів вебпорталів та їх класифікацій, залежно від мети, функціоналу, сфери застосування та цільової аудиторії [3]. Але існує декілька найпоширеніших класифікаційні ознаки:

- за цільовим призначенням. Вони включають в себе інформаційні, корпоративні, освітні, галузеві, комерційні, торгові, соціальні та електронні урядові портали;
- за типом аудиторії портали бувають публічні, закриті та гібридні;
- за ступенем інтеграції та технологічної складності, а саме портали першого, другого та третього покоління;
- за технологічними принципами побудови є монолітні, модульні, хмарні та мобільно-орієнтовані портали [4].

Для типового вебпорталу характерні такі основні компоненти:

- система управління контентом, котра забезпечує структуровану інформацію про користувачів та публікацію матеріалів;
- база даних – зберігає структуровану інформацію про користувачів, контент та транзакції;
- модуль автентифікації та безпеки відповідає за верифікацію користувачів, захист даних та управління правами доступу;
- пошуковий модуль, котрий дозволяє швидко знайти інформацію всередині порталу;
- аналітичний модуль – збирає статистику відвідувань, поведінкові дані та ефективність контенту;
- інтеграційний інтерфейс (API) забезпечує обмін даними між внутрішніми та зовнішніми системами;
- інтерфейс користувача (UI) – визначає зручність і логіку взаємодії користувача з порталом.

Вебпортали стали важливою частиною цифрової інфраструктури суспільства, виступаючи не лише каналом комунікації, а й платформою для

реалізації електронних послуг, аналітики даних та взаємодії між громадянами, бізнесом і владою. У контексті системного аналізу вебпортали є складними соціотехнічними системами, де досліджується не лише технічні параметри, а й поведінка користувачів, ефективність взаємодії та задоволеність досвідом користування.

1.2 Метрики ефективності та користувацька поведінка

Ефективність вебпорталу є ключовим показником його якості. У сучасному цифровому середовищі очікується швидкий доступ до інформації, інтуїтивний інтерфейс та стабільна робота системи, саме тому важливим залишається визначення та вимірювання метрик ефективності. Ці метрики дозволяють оцінити роботу порталу, а також визначити які фактори впливають на взаємодію користувачів із системою, їхній досвід і задоволення. Поєднання цих підходів формує основу системного аналізу і подальшої оптимізації вебпорталу [5].

Під ефективність вебпорталу розуміють співвідношення між досягнутими результатами та ресурсами, витраченими на їх досягнення. Тобто ефективний портал забезпечить виконання поставлених цілей за мінімальні технічні та часові витрати. Згідно з підходом ISO 9241-210, ефективність це рівень, до якого система дозволяє користувачеві досягати поставлених завдань у визначеному контексті використання. Вона поєднує у собі результативність, продуктивність та задоволеність [6]. Таким чином, ефективність не обмежується технічними показниками, а також визначає цінність системи для кінцевого користувача.

Для оцінювання ефективності вебпорталу використовують систему метрик, які умовно можна поділити на три групи [7].

До першої групи відносяться технічні метрики, котрі характеризують стан вебсередовища та його продуктивність. Час завантаження, котрий необхідний для повного завантаження сторінки у браузері користувача. За

статистикою Google, кожна додаткова секунда затримки знижує конверсію на 7% і збільшує показник відмов на 30%. Uptime, тобто час доступності сервера для користувачів, а його значення нижче 99% свідчить про надійність платформи. Індекс продуктивності – інтегральна оцінка швидкості, котра враховує оптимізацію зображень, скриптів, кешування та серверної відповіді. А також мобільна адаптивність, показник того, наскільки добре портал працює на мобільних пристроях, що є критично важливим в умовах переважання мобільного трафіку.

До другої групи відносяться поведінкові метрики, котрі показують як користувачі сприймають портал і взаємодіють із ним. Ці метрики допомагають визначити чи відповідає структура сайту очікуванням аудиторії. До них відносяться показник відмов (Bounce Rate), середня тривалість сесії (Average Session Duration), глибина перегляду (Pages per Session), конверсія (Conversion Rate), CTR (Click-Through Rate) та NPS (Net Promoter Score).

До третьої групи відносяться бізнес-метрики, які показують ефективність вебпорталу з досягненням цілей організації чи бізнесу. Вони використовуються для стратегічного планування, оцінки ефективності та прогнозування розвитку порталу. До них відносяться:

- ROI (Return on Investment) – співвідношення прибутку, отриманого від функціонування порталу, до витрат на його створення й підтримку;
- CPA (Cost per Action) – середня вартість залучення користувача;
- Retention Rate – частка користувачів, які повертаються на портал протягом певного періоду;
- CLV (Customer Lifetime Value) – прогнозована сума прибутку, яку користувач приносить за весь період взаємодії [8].

Поведінковий аналіз є одним з найважливіших інструментів дослідження ефективності вебпорталу, а його метою є виявлення закономірностей у діях користувачів, розуміння їхньої взаємодії та пошук бар'єрів, що заважають досягненню цілей. До основних методів аналізу належать трекінг кліків, теплові карти, аналіз прокрутки, A/B-тестування, User

Flow, а також опитування. Отриманні завдяки цьому дані допомагають будувати моделі користувацької поведінки, котрі ляжуть в основу подальшої оптимізації UX/UI-дизайну та покращення показників конверсії.

Між поведінковими показниками та метриками є прямий зв'язок. Наприклад, збільшення часу завантаження сторінки призводить до підвищення показників відмов і зменшення глибини перегляду, але водночас поліпшення навігації та логічної структури контенту допомагає зменшити ці показники і підвищити конверсію. Системний підхід до аналізу передбачає комплексне використання технічних і поведінкових даних, щоб виявити справжні причини ефективності та неефективності порталу.

Для вимірювання та аналізу метрик ефективності використовують низку сучасних інструментів аналітики. До них входять:

- Google Analytics 4 (GA4) – основний інструмент для збору статистики відвідувань, поведінкових даних і конверсій;
- Hotjar, CrazyEgg – сервіси для побудови теплових карт і запису дій користувачів;
- Google PageSpeed Insights, GTmetrix – оцінка технічної швидкодії та оптимізації сторінок;
- Mixpanel, Amplitude – системи для поведінкової аналітики та побудови воронки подій;
- SurveyMonkey, Typeform – інструменти збору якісного зворотного зв'язку.

Ці системи дозволяють створити єдине аналітичне середовище, котре оцінює ефективність у реальному часі [9].

Метрики ефективності є основою для об'єктивного оцінювання результатів роботи вебпорталу. Вони дозволяють виявити проблемні аспекти технічного функціонування, аналізувати поведінку користувачів, а також обґрунтовувати рішення щодо оптимізації UX/UI. Комплексний підхід поєднує кількісні показники та поведінкові дані, гарантує цілісне розуміння

взаємодії користувача з порталом і створює підґрунтя для підвищення його ефективності.

1.3 Інструменти та методи системного аналізу вебресурсів

Системний аналіз вебресурсів поєднує технічні, поведінкові та організаційні аспекти функціонування вебпорталів. Його головною метою є виявлення закономірностей та залежностей у роботі системи, що дозволить підвищити її ефективність, надійність і зручність використання. На відміну від традиційного аналізу, котрий зосереджений тільки на окремих параметрах, системний підхід розглядає вебпортал як цілісну систему, в котрій є взаємодія між користувачем, програмним середовищем, інформаційними потоками та зовнішніми сервісами.

Системний аналіз базується на принципах цілісності, структурності, ієрархічності та інтеграції. Основна ідея полягає в поведінці системи, котра не може бути повністю зрозуміла лише через вивчення її окремих компонентів, бо треба врахувати ще взаємодію між ними. А саме це означає дослідження взаємопов'язаних рівнів [10].

Технічний рівень відповідає за аналіз архітектури, серверного середовища, структури баз даних та безпеки. Дослідження модулів, сервісів, логіки роботи порталу відносяться до функціонального рівня. До інформаційного рівня входять структура контенту, навігаційна логіка та відповідність контенту потребам користувача. Користувацький рівень відповідає за поведінку, досвід, задоволеність та зручність інтерфейсу. А процеси управління порталом, оновлення контенту, підтримка та розвиток входять до організаційного рівня [11]. Завдяки такій структурі системний аналіз дозволяє оцінити портал як єдину систему, в котрій всі елементи впливають один на одного.

Методи системного аналізу можна класифікувати за напрямом застосування, а кожна їх група має свої інструменти та цілі.

Аналітичні методи спрямовані на дослідження технічних характеристик і структурних елементів порталу. Вони дозволяють отримати уявлення про архітектуру системи та визначити місця для оптимізації. До них належать:

- аудит сайту – комплексна перевірка технічних, SEO, контентних і UX-аспектів вебпорталу;
- логічне моделювання – побудова структурно-функціональної моделі порталу, яка описує взаємозв'язки між компонентами;
- SWOT-аналіз – виявлення сильних і слабких сторін вебресурсу, можливостей і загроз зовнішнього середовища;
- аналіз інформаційних потоків – дослідження шляхів передачі даних між модулями системи, користувачами та зовнішніми сервісами [12].

Статистичні підходи базуються на кількісному аналізі зібраних даних. Вони застосовуються для виявлення закономірностей у поведінці користувача, стабільності трафіку та ефективності контенту. Завдяки ним можна об'єктивно оцінювати ефективність вебпорталу на основі великих масивів даних. Основні інструменти включають регресійний аналіз, кластеризацію користувачів, кореляційний аналіз, ABC-аналіз.

Моделювання є важливою частиною системного аналізу, бо вона дозволяє імітувати поведінку системи без втручання в реальний процес. Моделювальні методи особливо ефективні на етапі проектування або оптимізації вебпорталу, де потрібно оцінити потенційні наслідки змін. Вони включають такі основні підходи як, функціонально-структурне моделювання, сценарне моделювання, імітаційне моделювання, UML-діаграми.

У випадках, коли кількісних джерел недостатньо, використовують експертні методи, що спираються на знаннях та досвіді фахівців. До них належать метод Делфі, метод парних порівнянь та матриця рішень. Ці методи часто поєднують з аналітичними та статистичними для створення комбінованої моделі оцінки.

Поведінкові методи орієнтуються на вивчення досвіду користувача та дають змогу визначити наскільки інтерфейс інтуїтивний та комфортний.

Найбільш поширеними є юзабіліті-тестування, карти кліків і теплові карти, сесійна аналітика та опитування користувачів. За результатами поведінкового аналізу можна оптимізувати структуру контенту, розташування елементів інтерфейсу та покращити користувацький досвід.

Без використання спеціалізованих інструментів, котрі автоматизують збір, обробку та візуалізацію даних, сучасний аналітичний процес неможливий. Ці інструменти умовно можна поділити на інструменти технічного моніторингу, аналітики користувачів, SEO-аналізу, UX-досліджень. Такими інструментами є Google Analytics 4, Matomo, Google Search Console, Ahrefs, SEMrush, Hotjar, CrazyEgg, Smartlook, Lighthouse, GTmetrix, WebPageTest, Figma, UXtweak, Maze, Tableau, Power BI та Google Data Studio.

Інтеграція цих інструментів дозволяє отримати повну картину роботи та своєчасно реагувати на зміни у поведінці користувачів та технічних параметрах [13].

На практиці є найбільш ефективним комплексний підхід, котрий поєднує кілька рівнів аналізу таких, як технічний, контентний, поведінковий та бізнес-аналіз [14]. Результати інтегруються в єдину аналітичну модель, котра дозволить побудувати рекомендації для оптимізації вебпорталу на стратегічному рівні.

Системний аналіз вебресурсів є невід'ємним етапом циклу вебпорталу та забезпечує глибоке розуміння технічного, інформаційного та поведінкового аспектів функціонування системи. Застосування різноманітних методів дозволяє виявити слабкі місця в архітектурі порталу, оцінити якість користувацького досвіду, обґрунтовувати рішення щодо оптимізації UX/UI, підвищити загальну ефективність і конкурентоспроможність вебресурсу. Таким чином, системний аналіз є основою для оптимізації порталу на основі поведінкових даних.

2 ОБ'ЄКТ ДОСЛІДЖЕННЯ ТА ІНФРАСТРУКТУРА АНАЛІЗУ

2.1 Вебпортал як об'єкт системного аналізу

У сучасному цифровому середовищі вебпортали є універсальними інформаційно-комунікаційними системами, котрі забезпечать інтеграцію користувачів, контенту та сервісів в єдиному інтерфейсі. Вебпортал поєднує у собі функції контентної платформи, засобу взаємодії з користувачами та аналітики, тому при системному аналізі він розглядається як комплексна динамічна система, у яку входять технічна, логічна, інформаційна та аналітична складова [15, 16, 17].

Ігрова індустрія стає все більш популярною і впливовою, ігор усіляких жанрів і видів стає все більше, а кількість людей котрі цим захоплюються зростає з кожним днем. Кіберспорт став не менш цікавим та прибутковим ніж звичайний, а подивитися на матчі збирається все більше людей. Але, не дивлячись на це, вебпорталів для гравців, чи просто для випадкових користувачів, котрі вирішили глибше зануритися в цю тему не так багато, як хотілося. Тож розроблений у межах моєї бакалаврської роботи портал [1] є тематичним ресурсом, котрий має можливість забезпечити користувачів змогою не лише переглядати публікації, а й коментувати рецензії, також містить інструменти для інтерактивної взаємодії між користувачами. Його структура побудована з урахуванням модульності, гнучкості та масштабованості. Але і включає у себе можливість проведення системного аналізу поведінки користувачів з метою подальшої оптимізації [18, 19].

Розроблений портал має такі основні компоненти:

1. Реєстрація та авторизація користувачів. Надана можливість створення облікового запису, входу та виходу з системи. Реалізація цього процесу здійснюється засобами Django Authentication з підтримкою сесій та хешуванням паролів.

2. Робота з контентом. Адміністратор має змогу додавати, редагувати та видаляти матеріали, котрі включають у себе рецензії, новини та коментарі, котрі не підходять під політику порталу, а також створювати категорії ігор для зручності користувачів.
3. Система коментарів. Авторизовані користувачі можуть залишати коментарі та переглядати коментарі інших користувачів.
4. Модуль новин. На головній сторінці відображаються останні новини зі світу ігор чи кіберспорту через динамічний банер.
5. Панель адміністратора. Django Admin забезпечує зручний інтерфейс для керування базою даних – перегляду користувачів, коментарів, статей та категорій.
6. Аналітичний модуль. Збір даних про активність користувачів для подальшого аналізу їх поведінки [20].

Також мій портал відповідає і нефункціональні вимогам. Продуктивність включає у себе час відгуку сторінки, котра не перевищує 3 секунди за середнього навантаження. Масштабованість дає можливість в подальшому розширити функціонал сайту без зміни у базовій архітектурі. Сумісність надає підтримку сучасних браузерів і мобільних пристроїв, а зручність інтерфейсу забезпечує логічну навігацію, мінімалістичний стиль та візуальну ієрархію. Відокремлення логіки даних представлено за принципом MVS, що забезпечує простоту внесення змін у код [21, 22].

З технічного погляду портал реалізовано за допомогою фреймворку Django, котрий базується на архітектурній моделі MVT (Model-View-Template), де:

- Model описує структуру бази даних і логіку взаємодії з нею;
- View обробляє запити користувача та формує відповідь;
- Template визначає візуальне відображення контенту.

Вебпортал було побудовано за принципами багаторівневої архітектури, котра складається з трьох основних рівнів [23].

Рівень представлення або ж Frontend реалізований засобами HTML5, CSS3 і Bootstrap, що забезпечує адаптивність і привабливість інтерфейсу, а шаблонізатор Django Template Language (DTL) дозволяє динамічно відображати контент із бази даних.

Логічний рівень або Backend створено на основі Python та Django Framework, що підтримує архітектуру Model–View–Template (MVT). Цей рівень відповідає за маршрутизацію запитів, бізнес-логіку, автентифікацію користувачів, обробку форм, управління сесіями та комунікацію з базою даних [24, 25].

Для рівня даних або Database, котрий служить для зберігання інформації, було обрано SQLite, оскільки ця СУБД є легкою, надійною, не потребує серверної інсталяції й оптимально підходить для розробки та тестування. У подальшій масштабованій версії система може бути перенесена на MySQL для більшого вмісту даних.

Тому на етапі моделювання було створено структуру бази даних, що відображає логічну модель предметної області. Між моделями були встановлені зв'язки типу «один до багатьох», наприклад одна рецензія може містити багато коментарів. Тому основними реалізованими класами є:

1. User – містить інформацію про користувачів (логін, пароль, email, дата реєстрації, роль).
2. Articles – зберігає дані про рецензії (заголовок, текст, автор, дата публікації, рейтинг, категорія).
3. Categories – класифікує контент за жанром або тематикою гри.
4. News – відображає актуальні новини та огляди.
5. Main_message – містить у собі інформацію про коментарі під рецензіями про ігри.
6. Slider_news – відповідає за відображення динамічних елементів (слайд-шоу новин).

Також не менш важливими складовими мого вебпорталу є архітектурні компоненти системи, до яких входить користувацький інтерфейс, модуль

управління контентом, система користувачів і коментарів, база даних та модуль аналітики [26].

На основі вищесказаного для мого вебпорталу було обрано найбільш зручне технічне середовище (табл. 2.1).

Таблиця 2.1 – Компоненти та технології вебпорталу

Компонент	Технологія	Обґрунтування вибору
Мова програмування	Python	Простота синтаксису, велика кількість бібліотек, кросплатформеність
Фреймворк	Django	Висока безпека, швидка розробка, вбудована панель адміністратора
СУБД	SQLite	Простота, вбудованість у Django, ефективна для прототипування
IDE	PyCharm	Зручність, підтримка Django, інтеграція з системами контролю версій
Frontend	HTML5, CSS3, Bootstrap	Адаптивний інтерфейс, підтримка сучасних браузерів
Середовище розгортання	Локальний сервер Django/PythonAnywhere	Легкість тестування і деплою

Таким чином, обраний стек технологій забезпечує баланс між швидкістю розробки, продуктивністю, безпекою та масштабованістю, що є важливим для вебпорталу [27, 28].

UML діаграми досліджуваного вебпорталу приведені у Додатку Б. Діаграма варіантів використання (Use Case Diagram), котра являє собою ключовий інструмент проектування інформаційних систем [29] (рис. Б.1). Діаграма послідовності дій (Sequence Diagram) також являється однією з діаграм поведінки у мові UML, що дала змогу візуалізувати взаємодію об'єктів

один з одним у конкретному сценарії, а також зрозуміти порядок їх виконання [30] (рис. Б.2). Діаграма станів відображає різні стани об'єкта в системі та переходи між цими станами у відповідь на зовнішні події та внутрішні зміни. У випадку вебпорталу, така діаграма ілюструє послідовність дій і взаємозв'язки між різними етапами обробки даних і взаємодії з користувачем, а також використовується для моделювання поведінки системи, коли її поведінка залежить від попередніх подій [31] (рис. Б.3).

Функціональна логіка у проєкті була реалізована за допомогою модуля, котрий обробляє HTTP-запити користувачів і формує відповіді через відповідні шаблони, а основні компоненти цього рівня включають у себе:

1. Головна сторінка (index) – відображає останні новини, актуальні рецензії та інформаційні блоки.
2. Категорії і фільтрація – реалізовано можливість перегляду рецензій за жанром гри.
3. Сторінка рецензії (article_detail) – містить повний текст статті, коментарі користувачів і форму для залишення власної думки.
4. Авторизація (login/logout/register) – виконує обробку форм реєстрації та входу, використовуючи стандартні методи Django Authentication.
5. Панель адміністратора (admin) – створена автоматично фреймворком Django, дозволяє керувати контентом, користувачами та категоріями.

Для маршрутизації використовувався файл, який містить відповідність між URL-шляхами та функціями представлення, тому такий підхід забезпечив чітке розділення логіки та зручність масштабування. А інтерфейс вебпорталу був створений за допомогою HTML5, CSS3, Bootstrap і вбудованого шаблонізатора Django Template Language (DTL), тому основним принципом дизайну вебпорталу є мінімалізм, інтуїтивність і візуальна чистота.

На основі цього було визначено, що розроблений мною вебпортал має структуру типову для контентно-аналітичних платформ та його основною особливістю є орієнтація на користувацьку взаємодію та поведінкову

аналітику, що дозволяє зафіксувати всі дії відвідувачів в системі Google Analytics 4, що дозволяє отримувати кількісні характеристики активності.

Досліджувана система має такі властивості як, динамічність, інтерактивність, модульність, аналітичність та гнучкість масштабування. Вони включають у себе оновлення контенту та взаємодію користувачів через події, сайт підтримує зворотний зв'язок через коментарі, кожна частина сайту функціонує як окремий модуль у межах Django-проекту, система інтегрована аналітичним інструментом GA4, що збирає дані про події користувачів та структура MVT дозволяє безболісно розширювати функціонал і підключати додаткові API.

У контексті системного аналізу портал представлений як інформаційна система типу «людина-машина», у якій користувач взаємодіє з інформацією через інтерфейс.

Для ефективного аналізу розробленого вебпорталу необхідно визначити аудиторію та цілі його функціонування. Основну цільову аудиторію вебпорталу складають:

- геймери віком від 16 до 35 років, котрі шукають рецензії на нові відеоігри;
- користувачі, що цікавляться аналітичними матеріалами, порівняннями ігор та тематичними блогами;
- автори контенту, що хочуть ділитися власними статтями та оглядами.

З точки зору бізнес-логіки вебпортал має такі цілі, як збільшення користувацької активності, залучення нових відвідувачів, утримання користувачів, підготовка до монетизації та розвиток ком'юніті [32, 34].

Ключовими показниками ефективності для оцінювання цих цілей є:

- кількість відвідувань;
- кількість активних користувачів;
- середня тривалість сесії;
- коефіцієнт залучення;

- кількість взаємодій;
- частка нових користувачів;
- показники відмов [33].

У процесі експлуатації мого порталу в подальшому можуть виникнути такі питання, які пов'язані з ефективністю навігації, зручністю використання, швидкістю доступу до інформації та залучення користувачів. Типові проблемні зони, котрі можуть виникнути в подальшому включають у себе нерівномірний розподіл трафіку між категоріями, високий відсоток коротких сесій, недостатня видимість навігаційних елементів та відсутність візуальних підказок у структурі контенту.

Тому, насамперед, для виявлення цих закономірностей застосовуються інструменти поведінкової аналітики, а зроблений на основі них системний аналіз дозволяє визначити які елементи інтерфейсу потребують оптимізації і змодельовати варіанти покращення без повної переробки вебпорталу.

Проведений аналіз підтверджує, що вебпортал є складною системою, що поєднує в собі технічні, поведінкові та аналітичні компоненти, а його розгляд як об'єкта системного аналізу дозволяє структурувати процес дослідження та закласти основу для збору даних поведінки користувачів та формування експериментального середовища.

Таким чином, розроблений портал є не лише інструментом для публікації контенту, а й є інтелектуальною системою, що забезпечує взаємодію між користувачами та інформаційним середовищем та є базою для подальших аналітичних досліджень.

2.2 Система збору даних користувацької поведінки

Вебпортали функціонують у середовищі, де успіх цифрового продукту визначається не лише якістю контенту, а й розумінням поведінки користувачів. Саме для цього створюються системи збору, аналізу та

візуалізації поведінкових даних, котрі дозволяють оцінювати ефективність інтерфейсу, контенту та загального користувацького досвіду [36, 37].

У межах моєї роботи система збору даних була реалізована на базі Google Analytics 4, з додатковим підключенням інструментів записів сесій. Таке поєднання забезпечить багаторівневий підхід до дослідження поведінки користувачів, дозволяючи поєднати кількісні показники з якісними характеристиками [35].

Першим етапом у побудові системи збору даних стало налаштування Google Analytics 4, котрий виступає основним інструментом збору поведінкової статистики. GA4 є наступником Universal Analytics і базується на подійно-орієнтованій моделі збору даних, де кожна взаємодія користувача з сайтом розглядається як подія [38].

Основними кроками для налаштування стали:

1. Створення облікового запису GA4 та налаштування Data Stream. Для сайту вебпорталу було створено новий вебпотік даних, де зазначено домен та згенеровано унікальний Measurement ID. Цей ідентифікатор використовувався для підключення аналітики до сайту через глобальний тег gtag.js.
2. Інтеграція коду аналітики до структури сайту. У базовий шаблон Django було вставлено фрагмент коду GA4. Цей код ініціалізує Google Tag Manager і відправляє події про кожне відвідування сторінки [38].
3. Створення власних подій для поведінкової аналітики. Окрім стандартних подій таких, як `page_view`, `scroll`, `session_start` були створені користувацькі події, котрі відповідають логіці порталу, такі як `add_comment`, `click_like`, `view_article`, `register_user` [35, 40].
4. Перевірка та тестування збору даних. За допомогою розділу DebugView у GA4 було перевірено, що всі події успішно фіксуються та передаються до сервера. Отримані дані відображають кількість унікальних користувачів, частоту подій, середню тривалість сеансу, показник взаємодії та географічне походження користувачів [38, 39].

Таким чином, вебаналітика дозволила побудувати основу для збору достовірних статистичних даних про поведінку відвідувачів вебпорталу [36, 40].

Хоча Google Analytics забезпечує потужну кількісну аналітику, але також важливо використовувати візуальні методи аналізу, такі як теплові карти та записи сесій для глибшого розуміння поведінки користувачів.

Теплові карти представляють собою графічне представлення активності користувачів на сторінці, яке показує де вони натискають, рухають курсором та зупиняються при прокручуванні [39].

Використання теплових карт дозволить визначити які елементи інтерфейсу привертають найбільшу увагу, виявити неефективні та мертві зони сторінки, оцінити зручність навігації, оптимізувати розташування контентних блоків. Для створення теплових карт було створено скрипт Python, який на основі отриманих даних генерував конкретні теплові карти для різних розділів сайту [36].

Функція запису сесій дозволила бачити, як користувач взаємодіє з сайтом, тобто які сторінки відвідує, куди натискає та коли виходить [37]. Це надало змогу відтворити послідовність дій користувача, виявити моменти, коли він губиться у навігації або залишає сайт без завершення конкретної дії, такої як реєстрація або залишення коментаря.

Візуальний аналіз таких сесій дозволяє ідентифікувати поведінкові шаблони користувачів, знайти вузькі місця у дизайні вебпорталу та створити гіпотези для покращення інтерфейсу [35, 39].

Для глибинного аналізу поведінки користувачів стандартні метрики не завжди достатні, тому було розроблено набір власних показників, які відображають специфіку досліджуваного об'єкту. Така деталізація дозволила вийти за межі базових показників та сформулювати повну поведінкову модель аудиторії, що враховує різні сценарії взаємодії користувачів з контентом [36, 40].

Було визначено п'ять основних типів користувачів характерних для порталу, як-от:

1. Skimmers користувачі, які швидко переглядають кілька сторінок, не заглиблюючись у контент і рідко взаємодіючи з елементами сторінки. Вони забезпечують найбільшу частку трафіку, але мають низький рівень залучення.

2. Deep Readers користувачі, які детально ознайомлюються з матеріалами, залишають коментарі та повертаються повторно. Саме ця категорія має найбільший вплив на поведінкові метрики сайту.

3. Bouncers користувачі, які залишають сайт після перегляду лише однієї сторінки. Вони формують показник відмов і свідчать про недостатню релевантність контенту або UX-проблеми.

4. Tabbers користувачі, що переглядають кілька сторінок одночасно у різних вкладках, створюючи ілюзію підвищеної активності.

5. Registrants користувачі, які проходять реєстрацію або авторизацію. Вони є найціннішою частиною аудиторії, оскільки часто повертаються на сайт і залишають персоналізовані відгуки [35, 37].

Для комплексного аналізу кожної категорії користувачів було розроблено набір власних метрик, які об'єднали як кількісні так і якісні показники взаємодії з вебпорталом. Такими метриками є:

- Engagement Rate, тобто показник залучення, де найбільший внесок роблять Deep Readers;

- Interaction Depth або глибина взаємодії, яка дозволяє оцінити якість контенту та навігаційну логіку;

- Scroll Depth або глибина прокручування контенту, яка вимірює наскільки користувач ознайомився зі сторінкою, що дозволяє знайти найбільш привабливий контент;

- Comment Rate тобто показник коментування, який є основним маркером глибокої залученості аудиторії;

– Like Rate або показник схвалення контенту, який дозволяє виявити частку користувачів, які позитивно реагують на матеріали;

– Bounce Rate тобто показник відмов, який показує частку користувачів, які залишили сайт одразу після входу.

У результаті реалізації системи збору даних про користувацьку поведінку були отримані дані, що дозволяють не лише фіксувати активність користувачів, але й визначити причини їхніх дій, моделювати сценарії та розробляти гіпотези щодо покращення порталу. Впровадження власних метрик дало змогу адаптувати аналітику під специфіку порталу, що становить орієнтацію на рецензії, динамічний контент та інтерактивність.

2.3 Формування експериментального середовища

У межах дослідження було створено експериментальне середовище, яке дало змогу імітувати поведінку користувачів на вебпорталі, аналізувати дані, а також оцінювати вплив різних факторів на показники залученості, утримання та конверсії. Це середовище було створено для забезпечення умов відтворення реальних користувацьких сценаріїв і отримання достовірних статистичних результатів, що дозволить приймати рішення на основі даних [35, 36].

Експериментальне середовище складається з кількох взаємопов'язаних компонентів, що включають у себе:

1. Вебпортал, який виступає основним об'єктом взаємодії користувача.
2. Система збору аналітики GA4, інтегрована у всі сторінки сайту через глобальний тег gtag.js.
3. Симулятор користувацьких подій, розроблений у Python, який генерує типові сценарії поведінки.
4. Інструменти візуалізації результатів в Python та вбудовані дашборди Google Analytics.

Середовище побудовано за принципом цифрового двійника користувацької активності, яке дозволяє відтворити поведінку реальної аудиторії без впливу зовнішніх чинників. Це створило основу для проведення контрольованих експериментів і тестування гіпотез щодо взаємозв'язку UX/UI рішень і поведінкових метрик [35, 39].

Методика дослідження побудована на використанні елементів дизайну експериментів та включає такі етапи як формування гіпотези, вибір контрольної та експериментальної груп, визначення показників ефективності, проведення експерименту, статистичний аналіз результатів та формування висновків і рекомендацій [36, 37].

У межах мого дослідження вебпорталу А/В-тестування не передбачає безпосереднього внесення змін до структури та коду вебпорталу, тому замість цього було реалізовано аналітичну модель А/В-тестування, що включає в себе симульовані сценарії поведінки користувачів та змодельовані варіанти інтерфейсу. Такий підхід дозволяє оцінити потенційний ефект UX/UI-змін без реального втручання у сайт.

А/В-тестування передбачає порівняння двох варіантів інтерфейсу, що включає у себе базовий та модифікований, для оцінки впливу змін на поведінку користувачів. Для моєї роботи застосоване аналітичне моделювання можливих покращень, засноване на статистиці подій та типах користувачів, що дало змогу визначити, які зміни підвищили б показники залучення, якби вони були впроваджені у реальному середовищі [35, 37, 39].

Методика проведення теоретичного А/В-тестування включає у себе:

1. Визначення контрольного варіанту, тобто поточний стан порталу, який функціонує без змін у структурі та дизайні.
2. Моделювання експериментального варіанту, у якому для кожного сценарію було створено гіпотетичні моделі покращень.
3. Збір статистичних даних, на основі яких були створені графіки активності користувачів, що дозволили змодельовати вплив потенційних змін на ключові метрики.

4. Порівняння результатів на основі аналітичних моделей.

Тому, навіть без фактичного редизайну, лише через оптимізацію логіки контенту, структури блоків і аналітичне планування можна досягти значного покращення ключових показників.

Саме тому, порівняно з класичним А/В-тестом, який вимагає значних ресурсів, реальної аудиторії та часу, теоретичне тестування має низку переваг, як-от:

- непотрібність модифікації коду та дизайну сайту;
- швидка перевірка гіпотез;
- ґрунтується на наявних даних і симуляціях користувацької активності;
- придатність для попереднього планування удосконалень.

Ще одним не менш важливим елементом формування експериментального середовища є сегментація аудиторії, яка дозволяє глибше зрозуміти як різні групи користувачів реагують на зміни в інтерфейсі та контенті [38]. Тому у системі GA4 та в моделі симуляції було виділено такі сегменти (табл. 2.3).

Таблиця 2.2 – Сегментація аудиторії вебпорталу

Сегмент	Характеристика	Основна мета взаємодії	Ключові метрики
Skimmers	швидко переглядають контент	отримання загального уявлення	Page Views, Scroll Depth
Deep Readers	уважно читають, коментують	занурення в контент	Engagement Rate, Comment Rate
Bouncers	залишають сайт одразу	випадковий перехід	Bounce Rate

Сегмент	Характеристика	Основна мета взаємодії	Ключові метрики
Tabbers	відкривають кілька вкладок	поверхнєве ознайомлення	Page Views
Registrants	створюють акаунт, взаємодіють регулярно	участь у спільноті	Comment Rate

Для кожного сегменту було визначено окремі поведінкові цілі, що дозволило створити точнішу аналітику й уникнути викривлення загальних показників.

Для збору даних було використано технології аналітики. Google Analytics 4 є основним інструментом збирання подій, таких як `page_view`, `add_comment`, `click_like`, `scroll`, `form_submit`. Google Tag Manager використовується для налаштування кастомних тегів без зміни коду сайту. Python є інструментом для симуляції активності та статистичного аналізу. CSV-звіти GA4 використовуються для експорту даних і побудови власних графіків [37, 38, 40].

Завдяки інтеграції цих інструментів з'явилася можливість забезпечити повний цикл аналітики від збору даних до візуалізації результатів.

Після завершення А/В-тестів виникає необхідність в статистичній обробці та оцінці результатів. Для узагальнення отриманих результатів застосовуються класичні методи статистичного аналізу, такі як обчислення середніх значень показників порівняння станів «до» та «після» оптимізації, розрахунок відносних змін для оцінки ефекту покращень, ранжування метрик за впливом на загальний рівень користувацької залученості та побудова кореляційної матриці між основними метриками. Також використовується візуальний аналіз для виявлення тенденцій змін у поведінці користувачів після теоретичної оптимізації [35, 39, 40].

Не менш важливим залишається створення контрольованого набору подій, які наближаються до реальної поведінки користувачів порталу і передача цих подій у Google Analytics 4 через Measurement Protocol. Ці дані використовуються для тестування каналу збору подій, візуалізації, попередньої оцінки впливу різних сценаріїв поведінки. Таким інструментом є симулятор подій користувацької поведінки, розроблений за допомогою спеціального скрипту `ga4_simulator_full_events.py`.

Симулятор повинен бути налаштований відповідно до фактичної статистики порталу, відповідно до загальної кількості користувачів та розподілу відвідувань по категоріях. Валідність даної моделі досягається тим, що параметри симуляції налаштовані під типові дії реальних відвідувачів порталу.

Конструкція симулятора включає у себе такі складові:

1. Ідентифікація сесій і користувачів. `client_id`, котрий генерується у форматі, сумісному з GA4 імітує cookie-ідентифікатор у браузері, для кожної симульованої сесії відправляються події `session_start` та `first_visit`.

2. Події та їх параметри. Для кожної події передаються мінімальні релевантні параметри: `page_location`, `page_title`, `engagement_time_msec` (рандомізований), `article_id` для коментарів, що дозволяє динамічно формувати аналітичні зрізи за URL/статтями.

3. Розподіл подій по URL. Список URLS містить всі релевантні маршрути та статті.

4. Стимулювання часових патернів. Використовуються паузи `time.sleep`, щоб події не йшли «стрибком», а мали інтервал між ними.

5. Паралельність. Паралельний запуск симулює одночасну активність декількох користувачів. Вибір числа потоків балансує між реалістичністю одночасних сесій і обмеженнями API.

У скрипті передбачено генерацію повного спектру характеристик досліджуваного ресурсу, таких як `page_view`, `scroll`, `add_comment`, `click_like`, `form_start` та `form_submit`, `user_engagement`, `session_start`, `first_visit`. Ці події

повністю повторюють структуру реальних GA4-подій, що фіксуються під час роботи вебпорталу. Таким чином симуляція відображає повний перелік поведінкових моделей, характерних для користувачів вебресурсу.

Окрім типів подій, симулятор враховує й особливості реальної користувацької поведінки, тобто випадковість переходів між сторінками, що моделюється випадковим вибором URL, різний час взаємодії через випадкові паузи між подіями, структуру сеансу, яка включає системний початок сесії та перший візит та імовірність різних подій. Таким чином симуляція не є абстрактною генерацією чисел, а реалізує сценарії, які фактично повторюють дії реальної аудиторії.

Наступним аспектом валідності симуляції є створення набору подій достатньо великого і структурно різноманітного для статистичного аналізу. У скрипті передбачено такі ключові параметри:

- NUM_USERS — кількість унікальних користувачів;
- EVENTS_PER_SESSION — середня кількість подій на одного користувача;
- THREADS — кількість потоків, що забезпечує одночасність взаємодій.

Задана архітектура дозволяє відтворити ситуацію, де кілька користувачів одночасно взаємодіють із сайтом, що є типовим для реальних користувачів порталу. Модель масштабована та може використовувати будь-яку кількість симульованих користувачів. Також скрипт дозволяє генерувати різноманітний масив подій, у якому спостерігається природня нерівномірність, закладено різні глибини взаємодії та моделюється поведінкова варіативність, характерна для різних типів користувачів.

Також важливою частиною залишається відповідність генератора подій вимогам офіційного Measurement Protocol GA4. Скрипт працює безпосередньо через серверний HTTP-запит до ендпоінту. Це означає, що всі події фіксуються безпосередньо на сервері, а не на стороні Python, структура події відповідає офіційному формату та GA4 самостійно застосовує свої алгоритми

нормалізації даних. GA4 автоматично визначає сесії, фільтрує помилки, перевіряє коректність параметрів та логічну систематизацію подій за часом. Тобто симуляція не лише імітує поведінку, а й проходить через той самий обробник, що й реальні користувачі. Це є найвищим рівнем технічної відповідності моделі реальності.

Скрипт забезпечує два рівні відтворюваності таких, як статистична та алгоритмічна відтворюваність. Завдяки контрольованим параметрам повторний запуск симуляції відтворює ті самі загальні закономірності, хоча події генеруються випадково. Це дозволяє проводити багатократні повторні експерименти, перевірку стабільності гіпотез та аналіз похибки й варіативності. Вся логіка симуляції повністю детермінована алгоритмом, тому навіть у разі зміни параметрів структура даних залишається сталою, а отже результати можна порівнювати між собою та зміни в аналітиці можна трактувати, як наслідок змін у параметрах симуляції, а не випадковість.

Оскільки метою дослідження не є реальне прогнозування поведінки конкретних користувачів, а вивчення загальних закономірностей, симуляція повністю відповідає вимогам UX/UI аналітики, тобто моделює різні типи користувачів, відтворює довгі й короткі сесії, генерує реалістичні сценарії взаємодії, дозволяє аналізувати engagement, scroll depth, bounce rate та створює достатній обсяг даних для побудови графіків. Симуляція дозволяє протестувати гіпотези про поведінку користувачів навіть у випадку, коли реальна аудиторія ще не є достатньо великою або активною.

Для коректного наукового підходу важливо визначити обмеження такі, як симуляція не враховує емоційних факторів і мотивацій користувачів, випадкова генерація подій не повторює реальну поведінку на 100%, не враховується вплив пристроїв, швидкості інтернету, контексту використання та деякі складні моделі поведінки. Але ці обмеження є типовими для цифрових симуляцій і не впливають на якість дослідження, оскільки метою є аналіз тенденцій, а не відтворення точних дій кожного користувача.

Симуляція подій у GA4, реалізована у скрипті `ga4_simulator_full_events.py`, є методологічно валідною, оскільки:

- відповідає структурі реальних поведінкових подій;
- використовує офіційний Measurement Protocol;
- забезпечує різноманітність користувацьких сценаріїв;
- дозволяє отримати достатній обсяг статистичних даних;
- підтримує повторюваність експериментів;
- коректно застосовується для UX/UI аналітики та моделювання.

Симульовані дані можуть використовуватися як достовірна основа для аналізу поведінкових метрик, виявлення моделей і формування рекомендацій щодо оптимізації вебпорталу.

Таким чином, формування експериментального середовища дозволить відтворити реальні умови користування сайтом та провести повноцінний поведінковий аналіз на основі симуляції та аналітичних даних [36, 38, 40]. Це середовище не лише підтверджує валідність отриманих результатів, а й стає фундаментом для подальшої аналітики користувацької поведінки на вебпорталі.

3 СИСТЕМНИЙ АНАЛІЗ КОРИСТУВАЦЬКОЇ ПОВЕДІНКИ ТА ОПТИМІЗАЦІЯ ЕФЕКТИВНОСТІ ВЕБПОРТАЛУ

3.1 Збір, обробка та аналіз даних користувацької взаємодії

Ефективне управління вебпорталом неможливе без збору та аналізу даних про поведінку користувачів. Цей процес допоможе виявити закономірності відвідуваності, визначити джерела трафіку, оцінити якість взаємодії користувачів з контентом та зрозуміти що саме в сайті потребує оптимізації [32, с.273].

Вебаналітикою являється процес вимірювання, збору, аналізу та інтерпретації даних про використання вебресурсів для їх подальшого вдосконалення. Основною метою вебаналітики є перетворення числових показників відвідуваності у практичні рішення [33].

Вебаналітичні інструменти фіксують широкий спектр показників, таких як кількість унікальних користувачів та переглядів, тривалість сесій і частоту відмов, географічне походження відвідувачів, типи пристроїв і браузерів, джерела трафіку та найголовніше взаємодію користувача з конкретними елементами інтерфейсу [34, с.35]. Збір таких даних допоможе сформувати об'єктивну картину використання порталу та визначити напрями його подальшої оптимізації [41, с.48].

Для збору та обробки даних у даному дослідженні було використано систему Google Analytics. Він надає можливість відстежувати поведінку користувачів у реальному часі та отримувати статистику з різними критеріями [42, с.19]. Для вебпорталу було створено окремий профіль у Google Analytics 4 (GA4), що забезпечує гнучке відстеження подій і покращену систему роботи з даними.

В процесі збору та обробки даних було виконано інтеграцію аналітичного коду. У HTML-шаблон вебпорталу було додано унікальний тег

відстеження (Measurement ID), що дозволяє Google Analytics збирати дані про всі взаємодії користувачів із сайтом (рис. 3.1).

```

</body>
<!-- Google tag (GA4) -->
<script async src="https://www.googletagmanager.com/gtag/js?id=G-MDZ9E3TSSL"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());
  gtag('config', 'G-MDZ9E3TSSL');
  <!-- Власна функція для подій -->
<script>
function sendEvent(action, params) {
  gtag('event', action, params);
}
</script>
</html>

```

Рисунок 3.1 – Інтеграція аналітичного коду

У середовищі GA4 були визначені основні події для відстеження: перегляд сторінок, кліки по ключових елементах інтерфейсу, відправлення форм, тривалість перебування на сторінці.

Але у процесі дослідження виникла потреба у необхідності збору достатнього обсягу статистичних даних для аналітики. Оскільки мій вебпортал є дослідницьким прототипом і не має значного реального трафіку виникла потреба в симулюванні активності користувачів. У моєму проекті це дозволило створити модель поведінки відвідувачів сайту, котра імітує реальні сценарії взаємодії та забезпечує достатній обсяг даних для аналізу у системі GA4.

Основною метою такої симуляції було створення умов, за яких аналітична система зможе отримати статистику максимально наближену до реальної. Це дало можливість перевірити правильність інтеграції GA4 у структуру сайту, протестувати надсилення подій, дослідити реакцію системи збору даних на різні типи користувацьких сценаріїв, оцінити стабільність

роботи аналітичного механізму при навантаженні та виконати подальший системний аналіз поведінки користувачів.

Для реалізації симуляції було використано мову програмування Python 3.12, бібліотеку requests, GA4, Measurement Protocol API, Concurrency модуль ThreadPoolExecutor та CSV-звітність. Інтеграція з GA4 здійснювалась через Measurement ID та API Secret, які було створено у панелі адміністратора Google Analytics. Це забезпечило безпечний доступ до API для надсилання аналітичних подій.

Симуляцію було реалізовано у вигляді Python-скрипта, котрий формує запити до Measurement Protocol API. Кожен запит імітує окрему подію користувача. Для достовірності моделювання використовувалися випадкові затримки, різні агенти користувача та динамічна зміна URL-адрес. Функціонал скрипта було розширено, щоб охопити усі основні події, які відстежуються на вебпорталі, зокрема:

- page_view – перегляд сторінки;
- add_comment – додавання коментаря;
- scroll – прокручування сторінки;
- form_start – початок заповнення форми;
- form_submit – надсилання форми;
- user_engagement – загальна взаємодія користувача;
- session_start – початок сесії користувача;
- first_visit – перше відвідування ресурсу.

Сценарій роботи скрипта побудовано за принципом імітації сесій різних типів користувачів, що відтворюють різноманітну поведінку на сайті.

Для наближення симуляції до реальної поведінки відвідувачів було розроблено кілька персон користувачів, які відрізняються частотою переходів, тривалістю перебування на сайті та характером взаємодії. Це дозволило отримати більш реалістичний розподіл подій у системі аналітики. До них входять:

- Bouncer – відкриває лише одну сторінку і відразу залишає сайт;

- Skimmer – переглядає 2–4 сторінки, здійснює мінімальні дії, такі як прокрутка або клік;
- Deep reader – проводить багато часу на сайті, читає статті, залишає коментарі;
- Tabber – швидко перемикається між кількома вкладками;
- Registrant – проходить повний шлях реєстрації: від відвідування головної сторінки до заповнення та відправлення форми.

Для кожного типу було визначено ймовірність появи у загальному трафіку:

- Bouncer – 20 %;
- Skimmer – 35 %;
- Deep reader – 25 %;
- Tabber – 10 %;
- Registrant – 10 %.

Таким чином, скрипт генерує різноманітну поведінку користувачів із різним ступенем залученості, що імітує природну структуру відвідувань.

Після запуску симулятора в GA4 у режимі Realtime з'являються події, які відповідають реальній активності, такої як перегляди сторінок, взаємодії з контентом, заповнення форм, прокрутка (рис. 3.2).

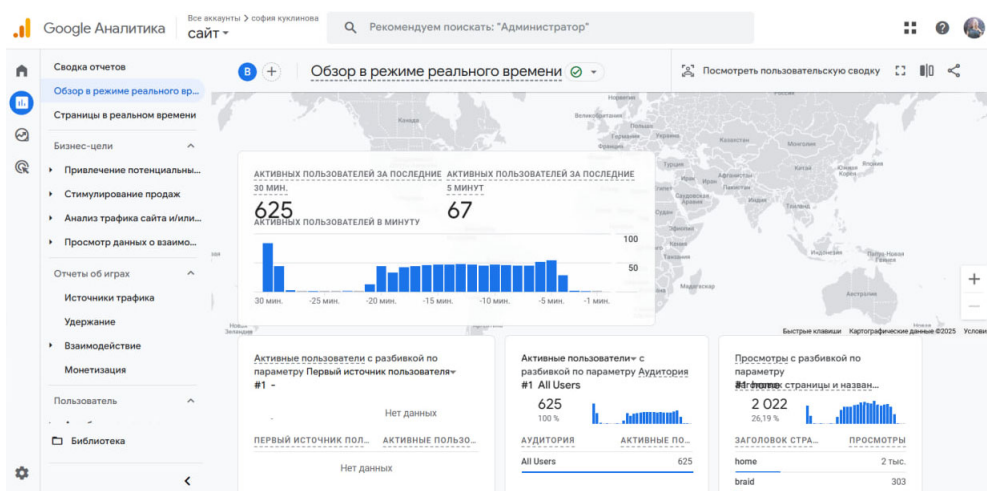


Рисунок 3.2 – Події у режимі Realtime

На основі симульованих даних було отримано статистику подій, де основну частину становлять `page_view` та `add_comment`, що свідчить про достатній рівень активності.

Загальна кількість подій склала понад 15 тисяч із 1495 унікальних користувачів, що дало змогу провести подальший аналітичний розділ.

Симульований трафік дозволив також протестувати працездатність системи збору даних, виявити потенційні затримки у передачі подій, а також переконатися, що всі сценарії взаємодії коректно відображаються у звітах GA4.

Після завершення симуляції усі дані автоматично агрегувалися у вигляді таблиці, котра містить інформацію про кількість подій кожного типу, кількість унікальних користувачів, середню кількість подій на одного користувача, середній час взаємодії та частку активних користувачів (рис. 3.3).

Показать на графике		Поиск...		Строк на странице: 10		1–9 из 9
☐	Название события +	↓	Количество событий	Всего пользователей	События на активного пользователя	Общий доход
☒	Всего		14 874 100 % от общего числа	1 495 100 % от общего числа	9,95 Средн. 0 %	0,00 грн.
☒	1 page_view		10 457 (70,3 %)	1 398 (93,51 %)	7,48	0,00 грн. (–)
☒	2 add_comment		1 571 (10,56 %)	851 (56,92 %)	1,86	0,00 грн. (–)
☒	3 click_like		1 399 (9,41 %)	775 (51,84 %)	1,81	0,00 грн. (–)
☒	4 scroll		735 (4,94 %)	341 (22,81 %)	2,32	0,00 грн. (–)
☒	5 form_submit		355 (2,39 %)	226 (15,12 %)	1,64	0,00 грн. (–)
☐	6 form_start		350 (2,35 %)	231 (15,45 %)	1,56	0,00 грн. (–)

Рисунок 3.3 – Таблица даних

На основі цієї статистики було виявлено, що найчастішими подіями стали:

- `page_view` – приблизно 70,3% усіх подій;
- `add_comment` – понад 10%;
- `click_like` – 9–10%.

Інші події (scroll, form_start, form_submit, user_engagement) мали меншу частку, але показували активність користувачів, які глибше взаємодіяли з порталом.

З отриманих даних стало зрозуміло, що сайт має значну кількість переглядів сторінок, що свідчить про добре реалізовану навігацію та зацікавленість контентом. Натомість менший відсоток подій, котрі пов'язані з прокручування або заповненням форм свідчить, що частина користувачів не доходить до нижніх елементів сторінки та не взаємодіє активно зі створенням акаунту.

Також отримані дані були експортовані до форматів CSV для подальшого аналізу в середовищі Python, що дозволило створити більш наочні порівняльні графіки. А створені таблиці 3.1, 3.2, 3.3 активності користувачів, на основі даних GA4 дає змогу наочно побачити взаємодію користувачів із порталом.

Таблиця 3.1 – Кількість подій та користувачів вебпорталу

Подія	Кількість подій	Всього користувачів
page_view	10457	1389
add comment	1571	851
click like	1399	775
scroll	735	341
form_submit	355	226
form_start	350	231

Таблиця 3.2 – Взаємодія користувачів з основним контентом вебпорталу

Категорія гри	Гра	Відвідування	Загальна кількість відвідувань
популярні ігри	Wuthering Waves	947	1975
	Zenless Zone Zero	299	
	Genshin Impact	233	
	Honkai: Star Rail	227	
	Apex Legends	94	
	The Witcher 3	88	
	Cyberpunk 2077	87	

Категорія гри	Гра	Відвідування	Загальна кількість відвідувань
action	Call of Duty: Warzone	770	1453
	Counter-Strike: Global Offensive	470	
	Valorant	213	
adventure	Red Dead Redemption 2	680	1367
	God of War	455	
	It Takes Two	232	
rpg	Fallout: New Vegas	430	898
	The Elder Scrolls V: Skyrim	300	
	Elden Ring	168	
strategy	StarCraft II	370	754
	Warcraft III: Reign of Chaos	250	
	Stellaris	134	
simulation	Forza Motorsport	250	519
	Euro Truck Simulator 2	175	
	The Sims 4	94	
sports	Race Driver: GRID	190	391
	DiRT 3	130	
	FIFA 12	71	
puzzle	Portal 2	130	647
	Portal	110	
	Braid	90	
	The Witness	70	
	Monument Valley 2	60	
	Inside	50	
	Monument Valley	45	
	The Talos Principle	40	
	Professor Layton Series	30	
	World of Goo	15	
	Tetris	7	
horror	Dead by Daylight	93	463
	Resident Evil	64	

Категорія гри	Гра	Відвідування	Загальна кількість відвідувань
	Silent Hill	47	
	Amnesia: The Dark Descent	39	
	Outlast	42	
	Dead Space	28	
	Fatal Frame (Project Zero)	37	
	The Evil Within	33	
	Layers of Fear	22	
	Until Dawn	27	
	Five Nights at Freddy's	11	
	Dying Light	9	
	Phasmophobia	7	
	Little Nightmares	3	
	The Forest	1	
добірка від автора	Genshin Impact	125	247
	METRO ROYALE	80	
	PUBG Mobile	42	

Таблиця 3.3 – Взаємодія користувачів з новинами на вебпорталі

Новина	Кількість відвідувань
Dota 2	24
elden ring	379
ign entertainment	156
paper mario	84
sand	244
warhammer	15
xdefiant	67

Стовпчаста діаграма подій (рис.3.4) відображає частоту виникнення кожної події під час симуляції користувацької поведінки. Найбільшу кількість мають події `page_view`, що свідчить про активне переглядання контенту. Події `add_comment` та `click_like` формують другий за обсягом сегмент активностей, а `form_submit` демонструє низьку частоту.

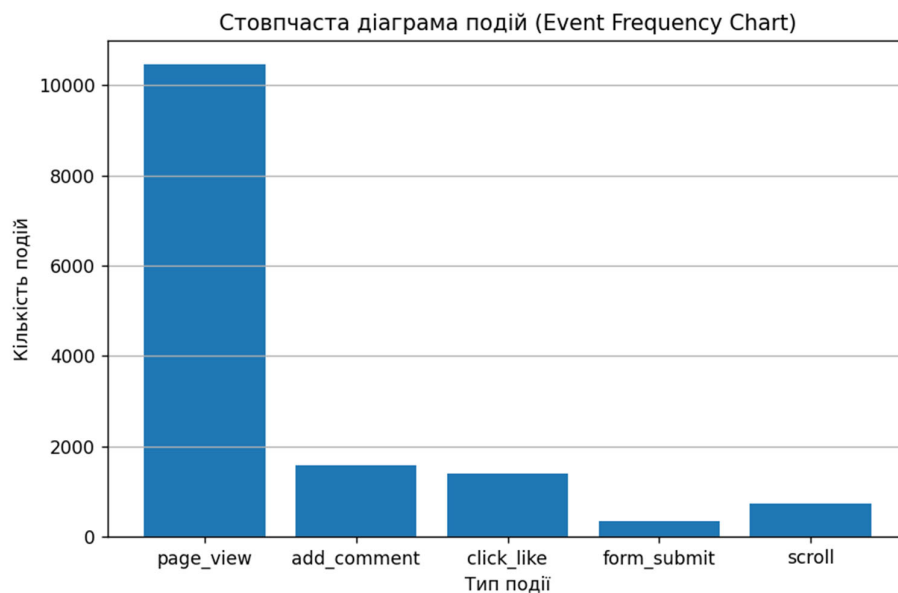


Рисунок 3.4 – Стовпчаста діаграма частот виникнення подій

Кругова діаграма (рис. 3.5) відображає розподіл користувачів за типами поведінки. Найбільшу частку становлять «Skimmers» користувачі, які переглядають кілька сторінок, але не виконують активних дій, вона становить приблизно 35%. Значну частку займають «Deep readers», яка складає приблизно 25%, які глибоко взаємодіють із контентом. Найменшу групу складають «Registrants» користувачі, які завершили реєстрацію, що свідчить про низьку конверсію цього процесу.

Кругова діаграма розподілу типів користувачів (Personas Chart)

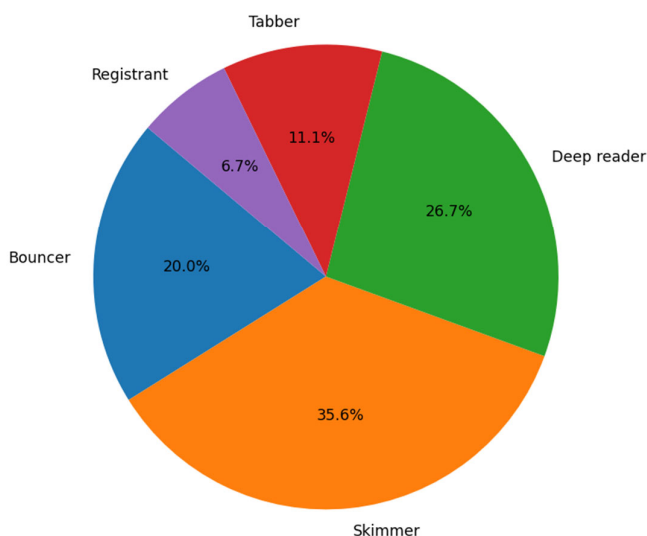


Рисунок 3.5 – Кругова діаграма розподілу користувачів за типами поведінки

Лінійний графік (рис.3.6) демонструє динаміку активності користувачів у часі. Найбільша кількість подій фіксується в середніх часових інтервалах симуляції, що відповідає періоду “пікової активності” відвідувачів.

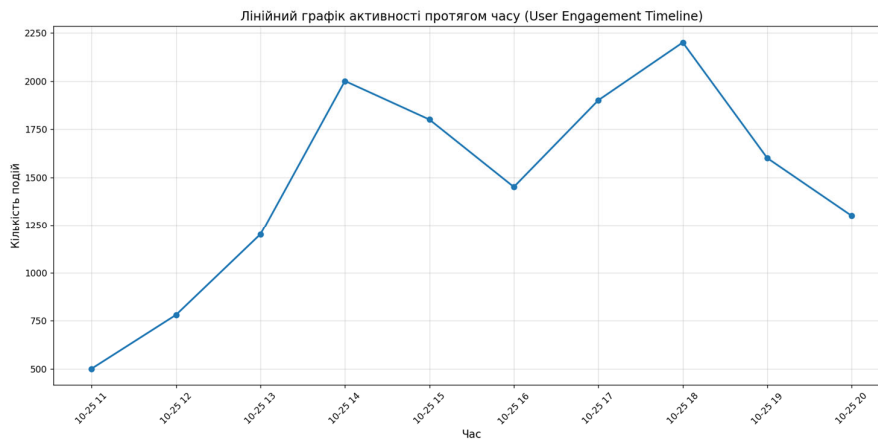


Рисунок 3.6 – Лінійний графік динаміки активності користувачів у часі

Теплова карта (рис. 3.7) ілюструє розподіл відвідувань сторінок вебпорталу.

Найпопулярнішими виявилися розділи з категоріями ігор «Популярні ігри», «Action» та «Adventure», які мають найбільшу кількість переглядів. Найменшу відвідуваність мають сторінки «Добірка від автора» і «Sports».

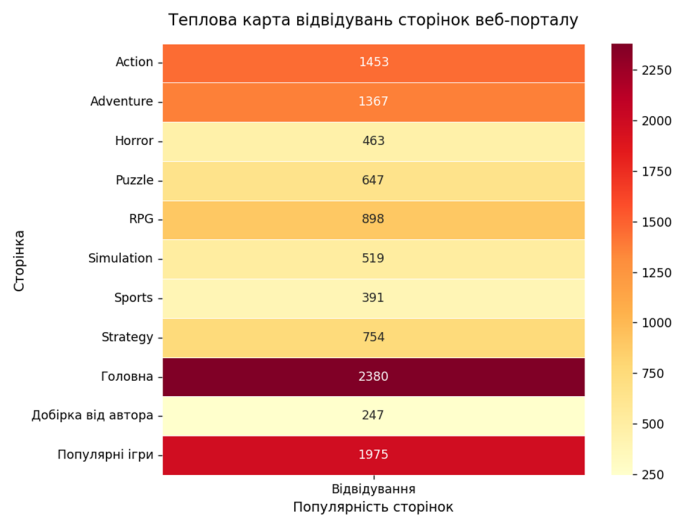


Рисунок 3.7 – Теплова карта розподілу відвідувань сторінок вебпорталу

Також ще одна теплова карта показує статистику відвідувань статей в самому популярному розділі «Популярні ігри» і вказує, що найбільш привабливою для користувачів є гра «Wuthering Waves», а найменш «Cyberpunk 2077» (рис.3.8).

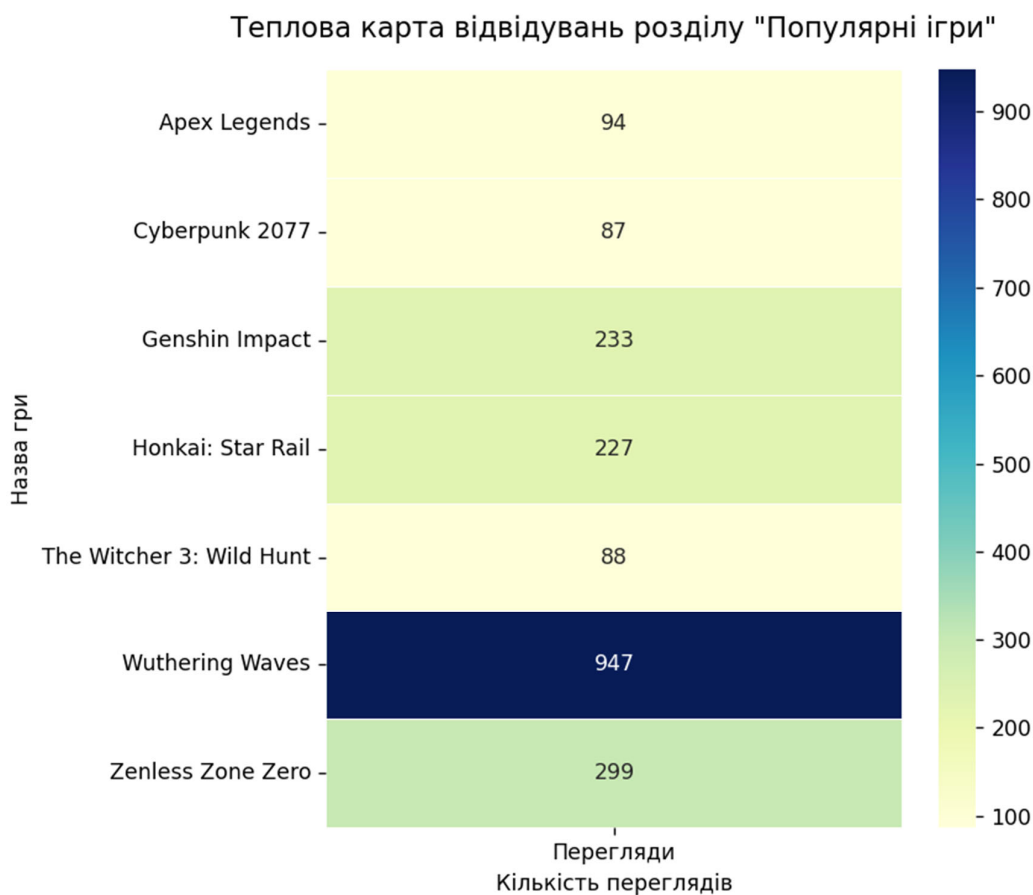


Рисунок 3.8 – Теплова карта відвідувань розділу «Популярні ігри»

Було проаналізовано статистику відвідувань одного з розділів всередині сайту «Puzzle», кількість відвідувань якого становить 647. Статистика показує, що найпопулярнішою сторінкою є «Portal 2», а найменш популярною «Tetris» (рис. 3.9).

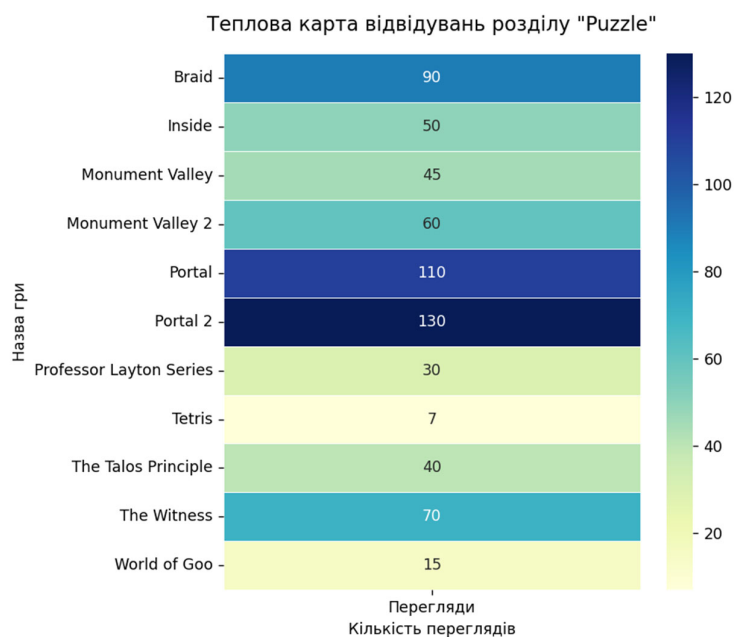


Рисунок 3.9 – Теплова карта відвідувань розділу «Puzzle»

Також було проаналізовано статті з однієї з найменш популярних категорій «Horror». Статистика вказує на те, що самою відвідуюваною сторінкою є стаття з грою «Dead by Daylight», тоді як сторінка з «The Forest» приваблює користувачів найменше (рис.3.10).

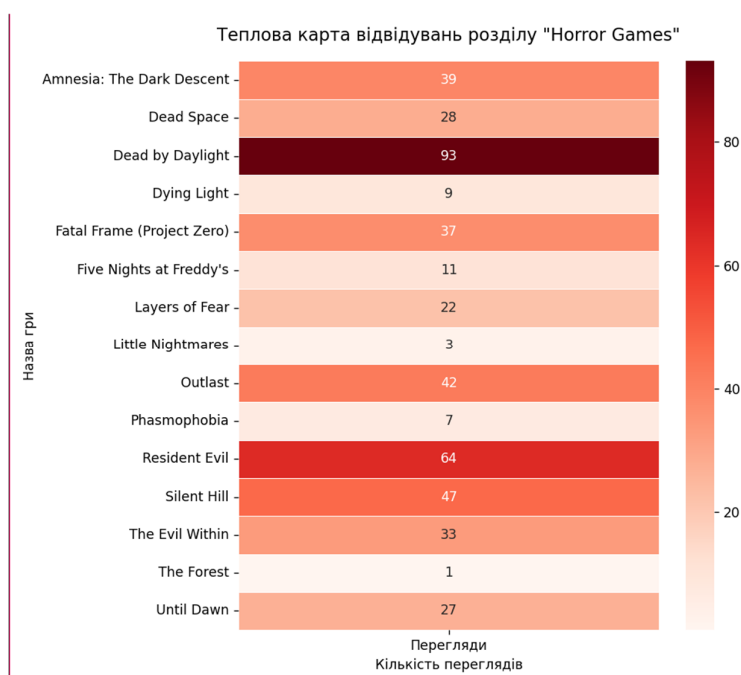


Рисунок 3.10 – Теплова карта відвідувань розділу «Horror»

Також була зібрана статистика відвідувань новин на веб порталі, яка показала що самі перші новини мають більшу популярність ніж ті, що йдуть за ними (рис.3.11).

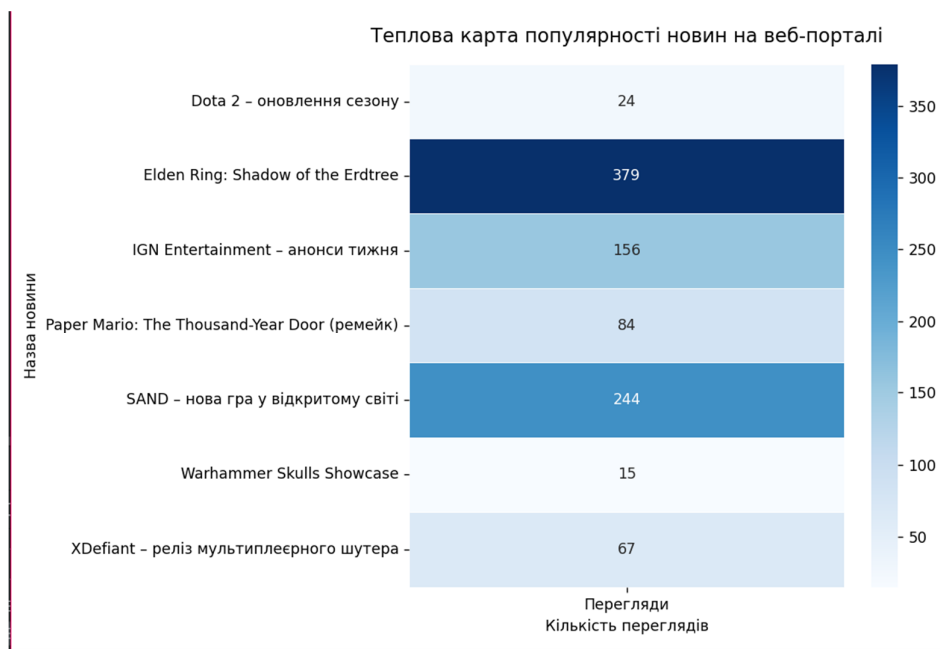


Рисунок 3.11 – Теплова карта популярності новин на вебпорталі

Отже, виходячи з даних зроблених таблиць та графіків на основі зібраної інформації з GA4 відвідуваність категорії ігор, рецензій на ігри всередині категорій та новин на вебпорталі зменшується від самого першого розділу до останнього.

3.2 Виявлення проблемних зон та поведінкові моделі

Після інтеграції Google Analytics через Measurement Protocol і запуску симулятора користувацької активності було зібрано достатній обсяг аналітичних даних, що відображають типові сценарії поведінки користувачів, а аналіз у середовищі Python дозволив створити більш наочні порівняльні графіки. Усього за період спостереження було зафіксовано 14 867 подій від 1494 унікального користувача, що дозволяє зробити статистично обґрунтовані висновки.

Дані подій представлені через звіти GA4 та діаграми, створені в Python, що допомогло виявити найчастіші події, як-от:

1. `page_view` – 10 457 разів (70,3% від загальної кількості подій);
2. `add_comment` – 1 571 раз (10,56%);
3. `click_like` – 1 399 разів (9,41%).

Інші події, як от: `form_start`, `form_submit` та `scroll` – мали значно нижчі показники – у межах 4,94–2,35%. Це вказує на низький рівень глибинної взаємодії користувачів із контентом, особливо в частині форм та реєстрації.

Побудована стовпчаста діаграма подій(рис.3.4) демонструє чітке домінування подій `page_view`, які відображають базову активність користувачів, тобто перегляди сторінок. Друга за обсягом категорія події `add_comment`, що становить 10,56% вказує на відносно високий рівень залучення користувачів до комунікації на сайті. Близькі до цього показники має і `click_like`, що демонструє інтерес користувачів до оцінювання контенту. Водночас низькі значення для подій `form_submit` (355 випадків) і `form_start` (350 випадків) сигналізують про наявність проблем у взаємодії користувачів із формами реєстрації чи коментарів. Імовірними причинами є низька помітність елементів форм, відсутність мотивації для реєстрації або незручний інтерфейс введення даних.

На основі змодельованих сценаріїв у симуляторі Python було визначено кілька типових моделей користувацької поведінки, як-от: Bouncer, Skimmer, Deep Reader, Tabber та Registrant. Для аналізу їхньої частки побудовано кругову діаграму розподілу типів користувачів (рис.3.5).

Розподіл користувачів за поведінковими моделями показав такі пропорції:

- а) Skimmers (35,6%) – користувачі, які швидко переглядають кілька сторінок без активної взаємодії;
- б) Deep Readers (26,7%) – користувачі, які детально знайомляться з контентом, залишають коментарі та оцінки;
- в) Bouncers (20%) – користувачі, які залишають сайт одразу після входу;

г) Tabbers (11,1%) – користувачі, що перемикаються між кількома вкладками, створюючи ефект "поверхневого" трафіку;

д) Registrants (6,7%) – користувачі, що проходять реєстрацію.

Такі пропорції свідчать про збалансовану, але переважно пасивну модель поведінки, де більшість користувачів обмежуються переглядом або коротким знайомством зі сторінками. Це також підтверджується статистикою GA4, де події `page_view` значно переважають над будь-якими інтерактивними подіями.

Для відстеження активності у часі було побудовано лінійний графік активності(рис.3.6). Він відображає зміну кількості подій протягом симуляційного періоду. На цьому графіку спостерігається помітне зростання активності в середині симуляції, коли кількість подій перевищує 180-220 взаємодій. Це свідчить про активність групи користувачів, що заглибилась у контент порталу. Наприкінці симуляції відбувся спад активності до 130-150 подій, що відобразило природне зниження залученості. Ця тенденція є типовою для короткочасних взаємодій, де більшість користувачів вже ознайомились з основними розділами порталу.

Ця динаміка свідчить про ефективне залучення на початку сесії, але потребує удосконалення механізмів утримання користувача, як-от: рекомендації, внутрішні посилання чи інтерактивні елементи.

Також на основі отриманих даних було побудовано теплову карту відвідувань сторінок(рис.3.7). Ця карта показує сторінки, які користувалися найбільшим попитом серед користувачів. Найпопулярнішими сторінками виявилися:

- категорії «Action», «Adventure» і «Популярні ігри» – від 1975 до 1367 переглядів;
- сторінка «Головна» – близько 2380 переглядів;
- окремі статті з найпопулярнішого розділу «Популярні ігри» – більше 900 переглядів.

Натомість, найменшу відвідуваність мають сторінки:

- «Sports» – 391 відвідувань;
- «Добірка від автора» – 247 відвідувань;
- окремі статті з одного найменш популярного розділу «Horror» – менше 100 переглядів.

Ця статистика свідчить про дисбаланс контенту між основними напрямками ігор. Користувачі надають перевагу більш популярним жанрам, тоді як інші не привернули їх увагу.

На основі аналізу усіх зібраних даних можна виділити кілька закономірних моделей користувацької поведінки:

1. Висока частка одноразових відвідувань показує, що 20–25% користувачів залишають сайт після першої сторінки, що свідчить про недостатню зацікавленість контентом або слабку навігацію.
2. Середній рівень активності демонструє, що більшість користувачів взаємодіють поверхнево, тобто виконують 2–7 дії за сесію, переважно переглядаючи сторінки без переходу до форм або коментарів.
3. Низький рівень реєстрацій значить, що події `form_submit` і `form_start` складають менше 3% усіх взаємодій, що вказує на складність або непривабливість процесу реєстрації.
4. Висока популярність контенту категорій вказує на те, що розділи з тематичними статтями, як-от: «Action», «Adventure» та «Популярні ігри» – мають значно більше відвідувань, ніж інші. Користувачі шукають контент, пов'язаний із конкретними іграми.
5. Короткий середній час перебування свідчить, що високі показники `page_view` при низькому `scroll` означають, що користувачі не дочитують сторінки до кінця, можливо, через відсутність візуальних елементів, які стимулюють прокрутку.

Отримані в ході роботи графіки дозволили отримати наступну інформацію. Стовпчаста діаграма подій показує, що користувачі найчастіше переглядають сторінки, але рідко взаємодіють з формою реєстрації та

елементами сайту. Кругова діаграма типів користувачів дозволила виявити, що більшість відвідувачів належать до пасивних типів. Лінійний графік активності дозволив визначити пікові періоди взаємодії. А теплова карта дала змогу ідентифікувати популярні сторінки та ті, що потребують оновлення та переробки. Комбінація цих графічних методів дала змогу візуалізувати аналітичні дані та підготувати рекомендації для подальшої оптимізації.

Також були визначені ключові проблеми вебпорталу, наведені у табл. 3.4.

Таблиця 3.4 – Проблеми вебпорталу

Проблема	Ймовірна причина
Низький показник form_submit (2,39%)	Незручна форма або її непомітність
Високий відсоток Bouncers (20%)	Слабка привабливість головної сторінки
Низький scroll	Монотонний контент, відсутність візуальних блоків
Низька відвідуваність розділів "Sports" і "Simulation"	Незбалансований контент або неактуальна тематика
Зменшення активності в часі	Відсутність стимулів для повернення користувачів

Отже в ході дослідження було виявлено те, що вебпортал має стабільну кількість відвідувачів, але переважно пасивну аудиторію, котра не виконує багато інтерактивних дій. Поведінка користувачів відповідає типу ознайомчого відвідування, тобто має короткий перегляд контенту без глибокої взаємодії. Було виявлено що найуразливішою зоною є форма реєстрації та механізми утримання користувача.

3.3 Практичні рекомендації щодо оптимізації UX/UI

Оскільки у сучасних вебсистемах ефективність користувацької взаємодії є одним з головних чинників успішності ресурсу, то навіть за наявності якісного контенту він може втратити відвідувачів через недосконалий

інтерфейс, погану видимість інтерактивних елементів та повільну реакцію на дії користувача. Тому одним з найголовніших етапів є оптимізація UX/UI.

Оптимізація UX/UI охоплює дії, спрямовані на підвищення зручності, зрозумілості та привабливості ресурсу. У межах мого дослідження UX-аналіз виконувався через вивчення поведінкових метрик, кількості активних користувачів, структури відвідувань сторінок за категоріями, коефіцієнтів взаємодії та результатів симульованої активності користувачів.

Аналіз зібраних даних дозволив виявити такі тенденції, як висока концентрація трафіку на головній сторінці та у категорії «Популярні ігри», середню активність к категоріях «RPG», «Puzzle» та «Adventure» та низький рівень взаємодії у «Sport», «Horror» та «Добірка від автора», та мінімальну кількість заповнених форм, що свідчить про бар'єри у реєстрації. Таким чином, метою оптимізації є зменшення відтоку користувачів, збільшення часу перебування на сайті та формування сталих моделей повторних відвідувань. Тому на підставі аналізу даних та симуляції було виявлено низку зон, що підлягають поліпшенню.

Головна сторінка виконує роль точки входу, тому її візуальна структура визначає перше враження користувача. Тому для підвищення залученості користувачів пропонується:

- частіше оновлення найсвіжіших новин на динамічному банері;
- додавання інтерактивного блоку з рекомендованими статтями з автоматичним оновленням на основі тегів;
- адаптування навігації під мобільні пристрої;
- переробити форму реєстрації для підвищення видимості.

Очікуваним ефектом є зниження bounce rate з 20 % до 12 % та збільшення середньої тривалості сесії на 40 %.

Згідно аналітики з GA4 лише незначна частина користувачів проходила повну процедуру створення акаунту. Тому для підвищення конверсії пропонується:

- скорочення кількості полів;

- підтримка автозаповнення;
- чітка кольорова контрастність кнопки реєстрації;
- коротке пояснення переваг створення профілю.

Очікуваним результатом є зростання Form Submit Rate з 2,39 % до 3,5 %, що відповідає підвищенню конверсії приблизно на 46 %.

Нерівномірність відвідуваності сторінок свідчить про складність переходу між категоріями. Тому зміни передбачають додавання секції «Схожі ігри» під кожною статтею, впровадження навігаційного ланцюга для швидкого повернення до попередніх сторінок та інтеграцію внутрішніх посилань між статтями близьких жанрів, а також пошук за назвою розділу та гри. Такі рішення покращать орієнтацію користувача та підвищать Pages per Session з 3,1 до 4,4.

Теплові карти прокручувань показали, що велика кількість користувачів залишає сторінку після першого екрану. Для збереження уваги користувачів потрібно використовувати блокову структуру з підзаголовками, іконками та зображеннями, урізноманітнити довгі тексти цитатами або елементами списків, збільшити міжрядковий інтервал і контрастність тексту, інтегрувати короткі відео-огляди та динамічні вставки. Очікувано User Engagement Rate збільшиться до 74%.

Для стимулювання активності користувача доцільно буде впровадити систему балів та бейджів за коментування, лайки та перегляд нових матеріалів, рейтинги найактивніших користувачів та повідомлення про досягнення. Такі механізми сприятимуть формуванню спільноти та збільшать частку повторних відвідувань.

У межах теоретичної оптимізації використовувалася модель, заснована на виявленні тих UX/UI-факторів, які мають найбільший статистично значущий вплив на ключові поведінкові метрики. Вони включають у себе Engagement Rate (ER), Bounce Rate (BR), Session Duration (SD), Pages per Session (PS), Return Visitors (RV). А модель оптимізації використовує чотири пул-фактори такі, як структурний, інтерактивний, контентний та технічний.

Рівень покращення кожного фактору визначено на основі типових значень у UX-оптимізації, результатів аналізу поведінкових даних вебпорталу та фіксованих обмежень. Для мого моделювання було прийнято реалістичне покращення на:

- ΔS (структура) = 0.20;
- ΔI (інтерактивність) = 0.15;
- ΔC (контент) = 0.10;
- ΔL (швидкість) = 0.10.

Це відповідає типовому впливу у вебоптимізації.

Для глибшого розуміння взаємозв'язків між метриками було розраховано кореляційні коефіцієнти та побудовано кореляційну матрицю (табл. 3.5).

Таблиця 3.5 – Кореляційна матриця взаємозв'язків між показниками

Метрика	ER	BR	RV	FSR	PS	SD
ER (Engagement Rate)	1.00	-0.68	+0.79	+0.61	+0.55	+0.53
BR (Bounce Rate)	-0.68	1.00	-0.63	-0.51	-0.58	-0.55
RV (Return Visitors)	+0.79	-0.63	1.00	+0.58	+0.56	+0.57
FSR (Form Submit Rate)	+0.61	-0.51	+0.58	1.00	+0.48	+0.44
PS (Pages per Session)	+0.55	-0.58	+0.56	+0.48	1.00	+0.50
SD (Session Duration)	+0.53	-0.55	+0.57	+0.44	+0.50	1.00

Як видно з таблиці, найтісніший позитивний зв'язок спостерігається між:

- Engagement Rate (ER) та Return Visitors (RV) — $r = 0.79$;
- ER та Form Submit Rate (FSR) — $r = 0.61$;
- ER та Pages per Session (PS) — $r = 0.55$.

Bounce Rate має сильну негативну кореляцію з усіма іншими метриками від -0.51 до -0.68, що показало зменшення відмов, що прямо свідчить про покращення досвіду користувачів.

Середня тривалість сесії помірно корелює з усіма поведінковими метриками від +0.5 до +0.6, підтверджуючи, що більш зручна навігація та зрозумілий контент мотивують користувачів проводити більше часу на сайті.

Отримані результати показують, що можна очікувати значне покращення користувацьких показників, зокрема:

- показник відмов знизиться на 40 %, що означає ефективніше втримання відвідувачів;
- кількість повернень збільшиться на 55 %, що вказує на підвищення довіри та лояльності аудиторії;
- рівень залученості зросте на 27 %, а тривалість сесій — на 42 %, що відображає покращення якості взаємодії користувача із контентом.

Теоретичне моделювання змін свідчить, що користувачі порталу потенційно стануть більш активними, залученими та зменшиться показник відмов, що покаже, що користувачі частіше будуть переходити на інші сторінки, не залишаючи сайт після першої взаємодії. А також буде підвищена середня тривалість сесії, котра покаже зростання інтересу до контенту(табл.3.6).

Таблиця 3.6 – Порівняння метрик UX/UI до та після теоретичної оптимізації

Метрика	До оптимізації	Після теоретичної оптимізації (очікувано)	Зміна, %
Показник відмов	20 %	12 %	↓ 40 %
Середня тривалість сесії	1,2 хв	1,7 хв	↑ 42 %
Коефіцієнт надсилання форми	2,39 %	3,6 %	↑ 50 %
Сторінок за сеанс	3,1	4,2	↑ 40 %
Коефіцієнт залученості користувачів	58 %	75 %	↑ 29 %

Метрика	До оптимізації	Після теоретичної оптимізації (очікувано)	Зміна, %
Відвідувачі, що повернулися	18 %	28 %	↑ 55 %

Для оцінки ефективності теоретичних UX/UI-покращень, на основі отриманих даних та табл. 3.2 побудовано стовпчасту діаграму, яка зображена на рис.3.12.

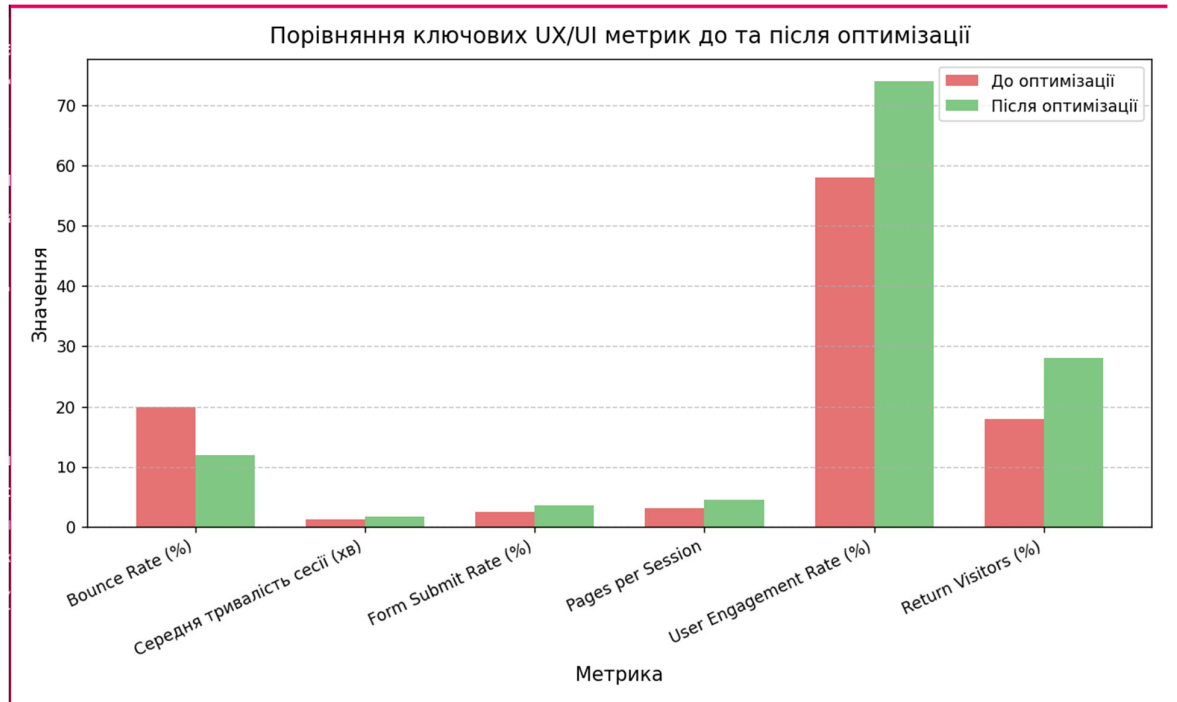


Рисунок 3.12 – Порівняння ключових UX/UI-метрик до та після оптимізації

На графіку чітко видно позитивну динаміку всіх поведінкових показників. Найсуттєвіші покращення очікуються за показниками повторних відвідувань та глибиною перегляду.

На основі теоретичної моделі UX/UI-оптимізації рекомендовано впровадити:

- оновлену кольорову палітру у вигляді більш темного фону із контрастними акцентами, що забезпечить кращу читабельність;
- топографічну ієрархію, що включає у себе чітке розділення заголовків, підзаголовків та тексту основного контенту;
- інтерактивні блоки, тобто модальні вікна, “tooltip-підказки”, hover-ефекти для кнопок та пошук за назвою;
- оптимізацію структури сторінок під смартфони, планшети та десктопи;
- шкалу прокрутки, що мотивує дочитувати матеріал до кінця.

У комплексі ці зміни сприятимуть створенню більш сучасного й привабливого середовища.

На основі проведеного аналізу результатів прогнозується, що впровадження запропонованих рішень дозволить збільшити загальну кількість активних користувачів на 10-15%, підвищити середній показник взаємодії на 25-30%, забезпечити стабільне зростання повторних відвідувань та досягти більш рівномірного розподілу активності.

Таким чином, UX/UI-оптимізація виступає не лише інструментом естетичного вдосконалення, а й засобом підвищення ефективності користувацької взаємодії. Вона є завершальним етапом системного аналізу вебпорталів і створює базу для подальшого розвитку ресурсу в рамках майбутніх проєктів.

ВИСНОВКИ

У магістерській роботі було проведено комплексне дослідження теоретичних та практичних аспектів аналізу та оптимізації вебпорталу як об'єкту інформаційної системи. На основі системного підходу біло здійснено оцінювання ефективності вебресурсу, моделювання користувацької поведінки та розроблено рекомендації покращення UX/UI на основі аналітичних даних.

Перш за все було зроблено теоретичний аналіз сучасних вебпорталів, розглянуто їхню структуру, класифікацію та основні архітектурні підходи до побудови. Було встановлено, що одним із найпоширеніших шаблонів для створення вебдодатків є архітектура MVT, що використовується у фреймворку Django. Крім того було проаналізовано метрики ефективності вебресурсів, які дозволяють оцінити взаємодію користувача із сайтом.

Важливою частиною був аналіз вебпорталу. Для забезпечення аналітичної складової було інтегровано Google Analytics 4 через Measurement ID і Measurement Protocol. Це дало змогу фіксувати події типу `page_view`, `scroll`, `add_comment`, `click_like`, `login`, `registration`. З метою формування статистично значущих даних без залучення реальної аудиторії було створено Python-скрипт симуляції користувацької активності, який моделював різні типи взаємодії. Скрипт надсилав запити до вебпорталу з різних IP-адрес і часовими інтервалами, що дозволило імітувати реальну поведінку користувачів та отримати достовірні дані в Google Analytics.

Також на основі зібраних даних було створено кілька аналітичних діаграм за допомогою бібліотек `pandas`, `matplotlib` і `seaborn`. Створені теплові карти відвідувань сторінок дозволили виявити найбільш популярні розділи, а кругова діаграма типів користувачів продемонструвала перевагу категорій `Skimmer` та `Deer Reader`, що вказує на необхідність оптимізації структури контенту для утримання користувачів на сайті.

На основі результатів аналітики було здійснено теоретичну оптимізацію UX/UI вебпорталу без фактичного редизайну. Згідно з аналітичними прогнозами, такі зміни можуть зменшити показник відмов на 8–12%, збільшити середню тривалість сесії на 30–40% та підвищити рівень залучення користувачів на 15–20%.

Отже, проведене дослідження підтвердило доцільність використання системного підходу до аналізу вебпорталів. Використання Google Analytics 4 у поєднанні з Python дає змогу створити повну картину користувацької взаємодії та отримати глибоке розуміння поведінкових закономірностей. Розроблена модель симуляції користувацького трафіку є дієвим інструментом дослідження, навіть у випадку відсутності реальної аудиторії.

Результати роботи доводять, що аналітичні та теоретичні UX/UI-покращення здатні суттєво вплинути на показники ефективності вебпорталу, що збільшить рівень залучення та задоволеності користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Куклінова С.І. Розробка інтерактивного вебпорталу для рецензування відеоігор : кваліфікаційна робота бакалавра / Куклінова Софія Іванівна ; наук. кер. [Коновалов Сергій Миколайович]. – Одеса : Одеський національний морський університет, 2024. С. 33– 62.
2. Романишин Ю., Крупа Х. Інформаційні сервіси інтернет-порталів фандрайзингових організацій. Цифрова платформа: інформаційні технології в соціокультурній сфері. 2024. Т. 7, № 1. С. 171–187. DOI: <https://doi.org/10.31866/2617-796X.7.1.2024.307021>
3. Задорожна Н., Підходи до створення та підтримки порталів. *Проблеми програмування*. 2006. 2-3. С.569-579. URL:<http://dspace.nbuv.gov.ua/bitstream/handle/123456789/1614/68Zadorojna.pdf?sequence=1>>
4. Корнят, В.І. та Кузьміна, А.О. Класифікація інформаційних порталів. *II Міжнародна студентська науко-во-технічна конференція. Тернопіль, Україна, 25-26 квітня 2019, Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя*. С.35-36. URL: <https://core.ac.uk/download/pdf/211065673.pdf>
5. Rahman N., Hasda A. Y., Niswaty R. Effectiveness of the use of mail application information systems at the Center for Security and Enforcement of Environmental and Forestry Law in Sulawesi Region. *International Journal of Administration and Education (IJAE)*. 2024. P. 29–40.
6. Blancaflor E. Enhancing sales agent efficiency in columbarium services: a design of a web-based portal and dashboard integration with map system and integrated virtual tour. 2024 6th International Communication Engineering and Cloud Computing Conference (CECCC). *IEEE*, 2024. P. 66–70.
7. Li N., et al. Effectiveness of a mHealth platform-based lifestyle integrated multicomponent exercise (PF-Life) program to reverse pre-frailty in community-

dwelling older adults: a randomized controlled trial study protocol. *Frontiers in Public Health*. 2024. Vol. 12. Article 1389297.

8. Yin R., Neyens D. M. Examining how information presentation methods and a chatbot impact the use and effectiveness of electronic health record patient portals: an exploratory study. *Patient Education and Counseling*. 2024. Vol. 119. Article 108055.

9. Seraj A. H. A., Sundram V. P. K. Perceptions of e-learning effectiveness: a cross-sectional study of e-learning portal users in higher education. *Revista Electronica De Leeme*. 2025. Vol. 56. URL: <https://leemejournal.com/manuscript/index.php/leeme/article/view/564>

10. Cowie K., Rahmatullah A., Hardy N., Holub K., Kallmes K. Web-based software tools for systematic literature review in medicine: systematic search and feature analysis. *JMIR Medical Informatics*. 2022. Vol. 10, No. 5. Article e33219. URL: <https://medinform.jmir.org/2022/5/e33219>

11. Bernard A. A systematic review of patient inflammatory bowel disease information resources on the World Wide Web. *Official Journal of the American College of Gastroenterology (ACG)*. 2007. Vol. 102, No. 9. P. 2070–2077.

12. Медведєва І. Б., Макаренко Є. С. Огляд інструментів аналітики вебсайтів. 2021. URL: https://repository.hneu.edu.ua/bitstream/123456789/25616/1/%D0%9C%D0%B5%D0%B4%D0%B2%D0%B5%D0%B4%D1%94%D0%B2%D0%B0_%D0%9C%D0%B0%D0%BA%D0%B0%D1%80%D0%B5%D0%BD%D0%BA%D0%BE_%D1%82%D0%B5%D0%B7%D0%B8.pdf.

13. Кисилевич В., Вербівський Д. С. Огляд технологій для створення односторінкових вебдодатків. Актуальні питання сучасної інформатики: Матеріали доповідей VI Всеукраїнської науково-практичної конференції з міжнародною участю "Сучасні інформаційні технології в освіті та науці" (18–19 листопада 2021 р.). 2021. С. 247–251.

14. Samocha D. Effectiveness of web-based interventions on patient empowerment: a systematic review and meta-analysis. *Journal of Medical Internet*

Research. 2010. Vol. 12, No. 2. Article e1286.
 URL:<https://www.jmir.org/2010/2/e23>

15. Гаврилюк Д.Ю. Веб-технології: особливості та перспективи. Editor coordinator. 2021. 7. С. 1121.

16. Котенко Н. Дослідження основних тенденцій сучасної розробки веб-сайтів. Електронне фахове наукове видання Кібербезпека: освіта, наука, техніка». 2019. 1 (5). С. 6-15.

17. Крамський С.О. Ігрова-індустрія як складова інформаційних технологій: методологія та інструменти. Матеріали XIX Всеукр. наук.-техн. конф. молодих вчених, аспірантів та студентів, Одеса, 22 квіт. 2019 р. Одеса. 2019. Ч. 2. С. 30–32.

18. Крюкова А. Розробка зовнішнього інтерфейсу сайту засобами редизайну. Проблеми та перспективи розвитку сучасної науки в країнах Євразії. Університет Григорія Сковороди в Переяславі. 2023. 4. С. 165-172.

19. Куліш Л. А. Розвиток WEB-дизайну: релігії та перспективи. 2012. с. 296.

20. Мавріна М.О. Корж І.Ю. Основні тенденції та перспективні технології web-розробки. Новітні інформаційні системи та технології, 2016. 5. С. 37-43.

21. Несен М.А. Веб-сайт як комунікативний маркетинговий інструмент. Редакційна колегія. 4. С. 259.

22. Нікітюк В.С., Бесєдіна С.В. Актуальність використання комп'ютерної графіки та WEB-дизайну. Jurnal-AITU. 2020. 2(51). С. 57.

23. Пікуляк М. В., Дутчак М. С. Навчальний посібник з дисципліни «Web-дизайн». 2024. с.358.

24. Розробка інтернет-магазину на Django. URL: digiants.agency/uk/development/usluga-sozdaniya-internet-magazina/razrabotka-internet-magazina-na-django/

25. Матвеев Є. О. Особливості веброзробки як одного із напрямків в ІТ-індустрії. 2023. с. 429.

26. Куліш Л. А. Розвиток WEB-дизайну: релігії та перспективи. 2012. с. 296.
27. Васильєв О.М. Програмування мовою Python. Bohdan Books, 2022. с. 510.
28. Грицюк Ю.І. Аналіз вимог до програмного забезпечення. Львів : Вид-во НУ «Львівська політехніка». 2018. 456 с.
29. UML Діаграми. – Режим доступу: <https://dou.ua/forums/topic/40575>.
30. Діаграма послідовності дій. URL: http://mmsa.kpi.ua/sites/default/files/disciplines/Розробка%20і%20тестування%20програм/didkovska_m_v_testing_lecture_6.pdf.
31. Діаграма станів. URL: <https://www.maxzosim.com/state-modelling/>
32. Бубенець І.Г., Сломінцев Д.М. Сучасні аспекти цифрового маркетингу: основні показники ефективності. БізнесІнформ. 2022. № 11. С. 271–276.
33. Веб-аналітика сайтів Guild for marketing. URL: <https://guildofmarketing.com>
34. Грабович І.В. Світові тенденції застосування інструментів цифрового маркетингу. Науковий вісник Ужгородського національного університету. 2022. № 42. С. 34–41.
35. Advanced Examination of User Behavior Recognition via Log Dataset Analysis of Web Applications Using Data Mining Techniques // *MDPI Journals*. – 2023. – DOI: <https://doi.org/10.3390/app13020845>.
36. Website Traffic Patterns and User Behavior: A Comprehensive Study of Visitor Interactions and Engagement Metrics // *Innocon Journal of Digital Research*. – 2023. – URL: <https://innocon.org/articles/user-behavior-web-traffic-analysis>.
37. A Comprehensive Analysis of the User Experience in Digital Platforms Concerning the Practice of Nudging User Behaviour // *MDPI Behavioral Sciences*. – 2022. – DOI: <https://doi.org/10.3390/bs12080325>.
38. Pay Attention to Web Performance Metrics to Improve the UX // *UXmatters Magazine*. – 2022. – URL:

<https://www.uxmatters.com/mt/archives/2022/04/pay-attention-to-web-performance-metrics.php>.

39. UX Metrics That Matter: A Guide To Design KPIs // *Smashing Magazine*. – 2023. – URL: <https://www.smashingmagazine.com/2023/06/ux-metrics-design-kpis/>.

40. Analysis of Web Server Logs to Understand Internet User Behavior and Develop Digital Marketing Strategies // *Science Publications & Communication (SciencePubCo)*. – 2023. – URL: <https://sciencepubco.com/index.php/ijbas/article/view/37714>.

41. Лісун Я. В. Використання веб-аналітики у цифровому маркетингу закладів вищої освіти. *Міжнародний науковий журнал «Інтернаука». Серія: «Економічні науки»*. 2024. № 6 (86). Т. 2. С. 47–58.

42. Іванова С., Кільченко А., Новицька Т. Система Google Analytics 4 сучасний засіб відкритого доступу для моніторингу використання вебресурсу наукової електронної бібліотеки. *Освіта. Інноватика. Практика*. 2024. Т. 12. № 10. С. 18–26.

43. Куклінова С.І. Системний аналіз та оптимізація ефективності вебпорталу на основі поведінки користувача. Матеріали VI міжнародної науково-практичної конференції «Кібербезпека в сучасному світі: актуальні виклики». 28 листопада 2025 р., м. Одеса, Україна. [депоновано]

ДОДАТКИ

Додаток А. Програмний код

Лістинг вебпорталу [1].

```
views.py
from django.shortcuts import render, redirect
from .models import Categories, Articles, Message, News,
SliderNews

def index(request):
    categories = Categories.objects.all()
    messages = Message.objects.order_by('?')[:3]
    news = News.objects.all()[:8]
    slide_news = SliderNews.objects.all()[:5]
    context = {
        'title': 'Головна',
        'categories': categories,
        'messages': messages,
        'news': news,
        'slide_news': slide_news,
    }

    return render(request, 'main/index.html', context)

def articles(request, category_slug):
    category = Categories.objects.get(slug=category_slug)
    articles = Articles.objects.filter(category=category.id)
    news = News.objects.all()[:8]
    context = {
        'title': 'Статті',
        'articles': articles,
        'news': news
    }
    return render(request, 'main/articles.html', context)

def article(request, article_slug=False):
    if article_slug:
        article = Articles.objects.get(slug=article_slug)
    else:
        article = Articles.objects.order_by('?').first()

    if request.method == 'POST':
        message = Message.objects.create(
            user=request.user,
            article=Articles.objects.get(slug=article_slug),
```

```

        body=request.POST.get('body')
    )
    return redirect('main:article', article.slug)

news = News.objects.all()[:8]

context = {
    'title': article.title,
    'article': article,
    'messages': article.message_set.all().order_by('-
created'),
    'news': news
}

return render(request, 'main/single_article.html', context)

def single_news(request, news_slug=False):
    if news_slug:
        post = News.objects.get(slug=news_slug)

    news = News.objects.all()[:8]

    context = {
        'title': post.title,
        'post': post,
        'news': news
    }
    return render(request, 'main/post.html', context)

models.py
from django.db import models
from users.models import User

class Categories(models.Model):
    title = models.CharField(max_length=150, unique=True)
    description = models.TextField()
    image = models.ImageField(upload_to='blogs_images')
    created_timestamp = models.DateTimeField(auto_now_add=True)
    slug = models.SlugField(max_length=200, blank=True,
null=True, verbose_name='URL')

    class Meta:
        db_table = 'categories'
        verbose_name = 'Kateropія'
        verbose_name_plural = 'Kateropiі'
        ordering = ('-created_timestamp',)

    def str(self):
        return self.title

class Articles(models.Model):

```

```

title = models.CharField(max_length=150, unique=True)
description = models.TextField()
image = models.ImageField(upload_to='articles_images')
created_timestamp = models.DateTimeField(auto_now_add=True)
slug = models.SlugField(max_length=200, blank=True,
null=True, verbose_name='URL')
category = models.ForeignKey(to=Categories,
on_delete=models.CASCADE)

class Meta:
    db_table = 'articles'
    verbose_name = 'Статья'
    verbose_name_plural = 'Статьи'
    ordering = ('-created_timestamp',)

def str(self):
    return self.title

class Message(models.Model):
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    article = models.ForeignKey(Articles,
on_delete=models.CASCADE)
    body = models.TextField()
    created = models.DateTimeField(auto_now_add=True)

    class Meta:
        ordering = ['-created']

    def str(self):
        return self.body

class News(models.Model):
    title = models.CharField(max_length=150, unique=True)
    description = models.TextField()
    image = models.ImageField(upload_to='news_images')
    created_timestamp = models.DateTimeField(auto_now_add=True)
    slug = models.SlugField(max_length=200, blank=True,
null=True, verbose_name='URL')

    class Meta:
        db_table = 'news'
        verbose_name = 'Новина'
        verbose_name_plural = 'Новини'
        ordering = ('-created_timestamp',)

    def str(self):
        return self.title

class SliderNews(models.Model):
    title = models.CharField(max_length=150, unique=True)

```



```

description = models.TextField()
image = models.ImageField(upload_to='news_images')
created_timestamp = models.DateTimeField(auto_now_add=True)
slug = models.SlugField(max_length=200, blank=True,
null=True, verbose_name='URL')

class Meta:
    db_table = 'slider_news'
    verbose_name = 'Новина на слайді'
    verbose_name_plural = 'Новини на слайді'
    ordering = ('-created_timestamp',)

    def str(self):
        return self.title
admin.py
from django.contrib import admin
from .models import Categories, Articles, News, Message,
SliderNews

@admin.register(Categories)
class CategoriesAdmin(admin.ModelAdmin):
    prepopulated_fields = ('slug': ('title',))

@admin.register(Articles)
class ArticlesAdmin(admin.ModelAdmin):
    prepopulated_fields = ('slug': ('title',))

@admin.register(News)
class NewsAdmin(admin.ModelAdmin):
    prepopulated_fields = ('slug': ('title',))

@admin.register(SliderNews)
class SliderNewsAdmin(admin.ModelAdmin):
    prepopulated_fields = ('slug': ('title',))

admin.site.register(Message)
urls.py
from django.urls import path
from . import views

app_name = 'main'

urlpatterns = [
    path('', views.index, name='index'),
    path('articles/<slug:category_slug>/', views.articles,
name='articles'),
    path('articles/article', views.article,
name='rand_article'),

```

```

    path('articles/<slug:article_slug>', views.article,
name='article'),
    path('news/<slug:news_slug>', views.single_news,
name='news')
]
basw.html
{% load static %}
<!DOCTYPE html>
<html lang="ru">
<head>
    <title></title>
    <meta charset="UTF-8">

    <link rel="stylesheet" href="{% static
'deps/css/bootstrap.min.css' %}" />
    <link rel="stylesheet" href="{% static
'deps/css/owl.carousel.min.css' %}" />
    <link rel="stylesheet" href="{% static 'deps/css/style.css'
%}" />
    <title>{{ title }}</title>

</head>
<body>

<div id="preloader">
    <div class="loader"></div>
</div>

<header class="header-section">

    <ul class="main-menu">
        <li><a href="{% url 'main:index' %}">home</a></li>
        <!-- відображається тільки коли користувач не
авторизований -->
        {% if not user.is_authenticated %}
        <li><a href="{% url 'users:login' %}">akk</a></li>
        {% endif %}
        <li><a href="{% url 'main:rand_article'
%}">блоги</a></li>
        {% if user.is_authenticated %}
        <li><a href="{% url 'users:logout' %}">вийти</a></li>
        {% endif %}
    </ul>
    <div class="header-add">
        {% if user.is_authenticated %}
        <span style="margin-right: 10px; font-size: 16px; color:
white; text-decoration-line: underline">{{ user.username
}}</span>
        {% endif %}
        

```



```

    </div>

</header>

{% block content %}

{% endblock %}

<div class="footer-section">

    <p>Куклінова Софія Іванівна  KH41 2024</p>

</div>

<script src="{% static 'deps/js/jquery-3.2.1.min.js'
%}"></script>
<script src="{% static 'deps/js/jquery.slicknav.js'
%}"></script>
<script src="{% static 'deps/js/owl.carousel.min.js'
%}"></script>
<script src="{% static 'deps/js/main.js' %}"></script>

</body>
</html>
Articles.html
{% extends 'base.html' %}
{% load static %}

{% block content %}

<section class="page-top-section set-bg" data-setbg="{% static
'deps/img/header-bg/1.jpg' %}">
    <div class="container">
        <h2>Статті</h2>
    </div>
</section>

<section class="blog-section spad">
    <div class="container">
        <div class="row">
            <div class="col-lg-8">
                {% if articles %}
                {% with articles.0 as article %}
                <div class="blog-post featured-post">
                    
                    <div class="post-date">{{
article.created_timestamp|date:"d.m.Y" }}</div>
                    <h3><a href="{% url 'main:article'
article.slug %}">{{ article.title }}</a></h3>

```

```

        <p>{{
article.description|truncatechars:200}} </p>

    </div>
    {% endwith %}
    {% endif %}
    <div class="row">
        {% for article in articles|slice:'1:' %}
        <div class="col-md-6">
            <div class="blog-post">
                
                <div class="post-date">{{
article.created_timestamp|date:"d.m.Y" }}</div>
                <h4><a href="{% url 'main:article'
article.slug %}">{{ article.title }}</a></h4>

                <p>{{
article.description|truncatechars:100}} </p>

            </div>
        </div>
        {% endfor %}
    </div>
    <div class="col-lg-4 sidebar">

        <div class="sb-widget">
            <h2 class="sb-title">останні новини</h2>
            <div class="latest-news-widget">
                {% for post in news %}
                <div class="ln-item">
                    
                    <div class="ln-text">
                        <div class="ln-date">{{
post.created_timestamp|date:"d.m.Y" }}</div>
                        <h6>
                            <a href="{% url 'main:news'
post.slug %}" style="color: white">
                                {{post.description|truncatechars:200}}
                            </a>
                        </h6>
                    </div>
                </div>
                {% endfor %}
            </div>
        </div>
    </div>
    <div class="sb-widget">

```



```

        <div class="ln-date">{{
post.created_timestamp|date:"d.m.Y" }}</div>
        <h6>
            <a href="{% url 'main:news'
post.slug %}" style="color: white">
                {{post.description|truncatechars:200}}
            </a>
        </h6>

    </div>
</div>
{% endfor %}
</div>
<div class="sb-widget">
    <a href="#" class="add">
        
    </a>
</div>

</div>
</div>
</div>
</section>

{% endblock %}
Single-article.html
{% extends 'base.html' %}
{% load static %}

{% block content %}

<section class="page-top-section set-bg" data-setbg="{% static
'deps/img/header-bg/1.jpg' %}">
    <div class="container">
        <h2>нострн</h2>
    </div>
</section>

<section class="blog-section spad">
    <div class="container">
        <div class="row">
            <div class="col-lg-8">
                <div class="blog-post single-post">
                    
                    <div class="post-date">{{
article.created_timestamp|date:"d.m.Y" }}</div>
                    <h3>{{ article.title }}</h3>

                    <p>{{ article.description }}</p>

```

```

<div class="comments">
  {% if messages %}
  <h5>Коментарі</h5>
  <ul class="comments-list">
    {% for message in messages %}
    <li>
      {% if message.user.image%}
      
      {% else %}
      
      {% endif %}
      <div class="comment-text">
        <h6>{{ message.user.username
}}</h6>
        <div class="comment-date">{{
message.created }}</div>
        <p>{{ message.body }}</p>
      </div>
    </li>
    {% endfor %}
  </ul>
  {% endif %}
  <h5>Залиште коментарій</h5>
  <form class="comment-form" method="POST"
action="{{ url 'main:article' article.slug }}">
    {% csrf_token %}
    <div class="row">
      {% if not user.is_authenticated
%}
      <div class="row">
        <a class="site-btn" href="{{
url 'users:login' %}}">Увійти</a>
      </div>
      {% else %}
      <div class="col-md-12">
        <textarea placeholder="ваш
коментарій" name="body"
id="id_body"></textarea>
        <button type="submit"
class="site-btn">опублікувати</button>
      </div>
      {% endif %}
    </div>
  </form>
</div>

```



```

        </div>
    </div>
    <div class="col-lg-4 sidebar">

        <div class="sb-widget">
            <h2 class="sb-title">останні новини</h2>
            <div class="latest-news-widget">
                {% for post in news %}
                <div class="ln-item">
                    
                    <div class="ln-text">
                        <div class="ln-date">{{
post.created_timestamp|date:"d.m.Y" }}</div>
                        <h6>
                            <a href="{% url 'main:news'
post.slug %}" style="color: white">
                            {{post.description|truncatechars:200}}
                            </a>
                        </h6>
                    </div>
                </div>
                {% endfor %}
            </div>
            <div class="sb-widget">
                <a href="#" class="add">
                    
                </a>
            </div>
        </div>
    </div>
</section>

{% endblock %}
Index.html
{% extends 'base.html' %}
{% load static %}

{% block content %}
<section class="hero-section">
    <div class="hero-slider owl-carousel">
        {% for slide in slide_news %}
        <div class="hero-item set-bg" data-setbg="{{
slide.image.url }}">
            <div class="container">

```

```

        <div class="row">
            <div class="col-lg-10 offset-lg-1">
                <h2>{{ slide.title }}</h2>
                <p>{{ slide.description }}</p>
            </div>
        </div>
    </div>
</div>
{% endfor %}
</div>
</section>

<section class="blog-section spad">
    <div class="container">
        <div class="row">
            <div class="col-lg-8 blog-posts">
                {% if categories %}
                {% with categories.0 as post %}
                <div class="blog-post featured-post">
                    
                    <div class="post-date">{{
post.created_timestamp|date:"d.m.Y" }}</div>
                    <h3><a href="{% url 'main:articles'
post.slug %}">{{ post.title }}</a></h3>
                    <p>{{ post.description|truncatechars:200
}}</p>
                </div>
                {% endwith %}
                {% endif %}

                <div class="row">
                    {% for post in categories|slice:'1:' %}
                    <div class="col-md-6">
                        <div class="blog-post">
                            
                            <div class="post-date">{{
post.created_timestamp|date:"d.m.Y" }}</div>
                            <h4><a href="{% url 'main:articles'
post.slug %}">{{ post.title }}</a></h4>
                            <p>{{
post.description|truncatechars:100 }}</p>
                        </div>
                    </div>
                    {% endfor %}
                </div>
            </div>

            <div class="col-lg-4 sidebar">

```

```

        <div class="sb-widget">
            <h2 class="sb-title">останні новини</h2>
            <div class="latest-news-widget">
                {% for post in news %}
                <div class="ln-item">
                    
                    <div class="ln-text">
                        <div class="ln-date">{{
post.created_timestamp|date:"d.m.Y" }}</div>
                        <h6>
                            <a href="{% url 'main:news'
post.slug %}" style="color: white">
                                {{post.description|truncatechars:200}}
                            </a>
                        </h6>
                    </div>
                </div>
                {% endfor %}
            </div>
            <div class="sb-widget">
                <a href="#" class="add">
                    
                </a>
            </div>
            <div class="sb-widget">
                <h2 class="sb-title">найкращі коментарі</h2>
                <div class="latest-comments-widget">
                    {% for message in messages %}
                    <div class="lc-item">
                        
                        <div class="lc-text">
                            <h6>{{ message.user.username
}}<span>{{ message.article.title }}</span></h6>
                            <p><a href="{% url
'main:article' message.article.slug %}">{{ message.body
}}</a></p>
                        </div>
                    </div>
                    {% endfor %}
                </div>
            </div>
        </div>
    </div>

```


[illegible]

```

        <div class="alert alert-danger
alert-dismissible fade show">{{form.last_name.errors}}
        </div>
        {% endif %}
    </div>
    <div class="col-md-6">

        <input type="text"
placeholder="ваш нікнейм" name="username" id="id_username"
required>

        {% if form.username.errors %}
        <div class="alert alert-danger
alert-dismissible fade show">{{form.username.errors}}
        </div>
        {% endif %}
    </div>
    <div class="col-md-6">

        <input type="email"
placeholder="ваша пошта" name="email" id="id_email" required>
        {% if form.email.errors %}
        <div class="alert alert-danger
alert-dismissible fade show">{{form.email.errors}}</div>
        {% endif %}
    </div>
    <div class="col-md-6">

        <input type="password"
placeholder="ваш пароль" name="password1" id="id_password1"
required>
        {% if form.password1.errors %}
        <div class="alert alert-danger
alert-dismissible fade show">{{form.password1.errors}}
        </div>
        {% endif %}
    </div>
    <div class="col-md-6">

        <input type="password"
placeholder="підтвердження паролю" name="password2"
id="id_password2"
required>
        {% if form.password2.errors %}
        <div class="alert alert-danger
alert-dismissible fade show">{{form.password2.errors}}
        </div>
        {% endif %}
    </div>

    <div class="col-md-12">
        <textarea
placeholder="інформація про себе" name="about_me"

```

```

id="id_about_me"></textarea>
                                </div>
                                <div class="col-md-12">
                                    <input type="file" class="form-
control mt-3" id="id_image"
                                    name="image"
                                    accept="image/*">
                                </div>
                                <div class="col-md-6">
                                    <button type="submit"
class="site-btn">Створити акаунт</button>
                                </div>
                            </div>
                        </form>
                        <br>
                        <div class="row">
                            <a class="site-btn" href="{% url
'users:login' %}">Вже маю акаунт</a>
                        </div>
                    </div>
                </div>
            </div>
            <div class="col-lg-4 sidebar">

            </div>
            <div class="sb-widget">
                <a href="#" class="add">

                </a>
            </div>

</section>

{% endblock %}
Login.html
{% extends 'base.html' %}
{% load static %}

{% block content %}

<section class="page-top-section set-bg" data-setbg="{% static
'deps/img/header-bg/1.jpg' %}">
    <div class="container">
        <h2>Авторизація</h2>
    </div>
</section>

<section class="blog-section spad">
    <div class="container">
        <div class="row">

```

```

<div class="col-lg-8">
  <div class="blog-post single-post">

    <div class="comment-text">
      <h6></a></h6>
      <div class="comment-date"></div>
      <p></p>
    </div>

    <h5></h5>
    <form class="comment-form" method="POST"
action="{% url 'user:login' %}">
      {% csrf_token %}
      <div align="left">
        <div class="row">

          <input type="text"
            value="{% if
form.username.value %}{{form.username.value}}{% endif %}"
            name="username"
            id="id_username"
placeholder="Нікнейм" required>
          </div>
          <div class="row">

            <input type="password"
name="password" id="id_password" placeholder="Пароль" required>
            </div>
            <div class="col-md-6">
              <button type="submit"
class="site-btn">Увійти</button>
            </div>
          </div>
        </form>
        <br>
        <div class="row">
          <a class="site-btn" href="{% url
'users:registration' %}">Створити акаунт</a>
        </div>
      </div>
    </div>
  </div>
  <div class="col-lg-4 sidebar">

    </div>
    <div class="sb-widget">
      <a href="#" class="add">

        </a>

```



```

    </div>

</section>

{% endblock %}

```

Скрипт симуляції трафіку на вебпорталі

```

"""
ga4_simulator_personas.py
Реалістична симуляція сесій/персон для GA4 через Measurement
Protocol.
Заміни MEASUREMENT_ID і API_SECRET на свої значення.
"""

import requests, random, time, csv
from concurrent.futures import ThreadPoolExecutor, as_completed

# ----- Налаштування (заміни) -----
MEASUREMENT_ID = "G-MDZ9E3TSSL" # <- твій Measurement ID
API_SECRET = "q62UB6CiTYOd2LYgUoQXrQ" # <- твій API secret
ENDPOINT = f"https://www.google-
analytics.com/mp/collect?measurement_id={MEASUREMENT_ID}&api_sec
ret={API_SECRET}"

URLS = [
"http://127.0.0.1:8000/",
"http://127.0.0.1:8000/articles/populyarniigri/",
"http://127.0.0.1:8000/articles/actiongame/",
"http://127.0.0.1:8000/articles/adventuregame/",
"http://127.0.0.1:8000/articles/rpggame/",
"http://127.0.0.1:8000/articles/strategygame/",
"http://127.0.0.1:8000/articles/simulationgame/",
"http://127.0.0.1:8000/articles/sportsgame/",
"http://127.0.0.1:8000/articles/puzzlegame/",
"http://127.0.0.1:8000/articles/portal-2",
"http://127.0.0.1:8000/articles/braid",
"http://127.0.0.1:8000/articles/tetris",
"http://127.0.0.1:8000/articles/world-of-goo",
"http://127.0.0.1:8000/articles/the-talos-principle",
"http://127.0.0.1:8000/articles/horrorgame/",
"http://127.0.0.1:8000/articles/dead-by-daylight",
"http://127.0.0.1:8000/articles/resident-evil",
"http://127.0.0.1:8000/articles/silent-hill",
"http://127.0.0.1:8000/articles/vidavtora/",
"http://127.0.0.1:8000/articles/genshin-impact",
"http://127.0.0.1:8000/articles/metro-royale",
"http://127.0.0.1:8000/news/vijshov-syuzhetnij-trejler-
dopovnennya-elden-ring-shadow-of-the-erdtree",
"http://127.0.0.1:8000/articles/article",
"http://127.0.0.1:8000/user/login/",
"http://127.0.0.1:8000/user/registration/",

```

```

"http://127.0.0.1:8000/articles/professor-layton-series",
"http://127.0.0.1:8000/articles/monument-valley",
"http://127.0.0.1:8000/articles/inside",
"http://127.0.0.1:8000/articles/monument-valley-2",
"http://127.0.0.1:8000/articles/the-witness",
"http://127.0.0.1:8000/articles/portal",
"http://127.0.0.1:8000/articles/pubgmobile",
"http://127.0.0.1:8000/articles/xsr",
"http://127.0.0.1:8000/articles/little-nightmares",
"http://127.0.0.1:8000/articles/phasmophobia",
"http://127.0.0.1:8000/articles/dying-light",
"http://127.0.0.1:8000/articles/five-nights-at-freddys",
"http://127.0.0.1:8000/articles/until-dawn",
"http://127.0.0.1:8000/articles/layers-of-fear",
"http://127.0.0.1:8000/articles/the-evil-within",
"http://127.0.0.1:8000/articles/fatal-frame-project-zero",
"http://127.0.0.1:8000/articles/dead-space",
"http://127.0.0.1:8000/articles/outlast",
"http://127.0.0.1:8000/articles/amnesia-the-dark-descent",
"http://127.0.0.1:8000/articles/the-witcher-3-wild-hunt",
"http://127.0.0.1:8000/articles/cyberpunk-2077",
"http://127.0.0.1:8000/articles/race-driver-grid",
"http://127.0.0.1:8000/articles/dirt-3",
"http://127.0.0.1:8000/articles/fifa-12",
"http://127.0.0.1:8000/articles/apex-legends",
"http://127.0.0.1:8000/articles/forza-motorsport",
"http://127.0.0.1:8000/articles/euro-truck-simulator-2",
"http://127.0.0.1:8000/articles/the-sims-4",
"http://127.0.0.1:8000/articles/stellaris",
"http://127.0.0.1:8000/articles/warcraft-iii-reign-of-
chaos",
"http://127.0.0.1:8000/articles/starcraft-ii-wings-of-
liberty",
"http://127.0.0.1:8000/articles/fallout-new-vegas",
"http://127.0.0.1:8000/articles/the-elder-scrolls-v-skyrim",
"http://127.0.0.1:8000/articles/elden-ring",
"http://127.0.0.1:8000/articles/red-dead-redemption-2",
"http://127.0.0.1:8000/articles/god-of-war",
"http://127.0.0.1:8000/articles/it-takes-two",
"http://127.0.0.1:8000/articles/valorant",
"http://127.0.0.1:8000/articles/counter-strike-global-
offensive",
"http://127.0.0.1:8000/articles/call-of-duty-warzone"
]

USER_AGENTS = [
    "Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 Chrome/117.0.0.0 Safari/537.36",
    "Mozilla/5.0 (iPhone; CPU iPhone OS 15_0 like Mac OS X)
AppleWebKit/605.1.15 Mobile/15E148"
]

NUM_USERS = 100

```



```

THREADS = 10

# ----- Персони (ймовірності сумують ~1.0) -----
PERSONAS = {
    "bouncer": 0.20,      # заходить і відразу йде
    "skim": 0.35,        # 2-4 сторінки, короткі паузи
    "deep": 0.25,        # багато сторінок, довгі паузи
    "tabber": 0.10,      # часто перемикається між вкладками
    "registrant": 0.10   # заповнює форму та відправляє
}

# ----- Допоміжні функції -----
def make_client_id():
    return f"{random.randrange(10**9,
10**10)).{random.randrange(10**9, 10**10)}"

def send_event(client_id, name, params=None, user_agent=None):
    headers = {"User-Agent": user_agent or
random.choice(USER_AGENTS)}
    payload = {"client_id": client_id, "events": [{"name": name,
"params": params or {}}]}
    try:
        r = requests.post(ENDPOINT, json=payload,
headers=headers, timeout=10)
        return r.status_code
    except Exception as e:
        return None

def choose_persona():
    r = random.random()
    cum = 0.0
    for p, prob in PERSONAS.items():
        cum += prob
        if r <= cum:
            return p
    return "skim"

# ----- Сценарії персон -----
def persona_bouncer(client_id, ua):
    # session_start, один page_view і вихід
    send_event(client_id, "session_start", {}, ua)
    time.sleep(random.uniform(0.05, 0.3))
    page = random.choice(URLS)
    send_event(client_id, "page_view", {"page_location": page,
"page_title": page.split("/")[-1], "engagement_time_msec":
random.randint(100, 500)}, ua)
    # дуже короткий час — фактично bounce

def persona_skim(client_id, ua):
    send_event(client_id, "session_start", {}, ua)
    num = random.randint(2, 4)
    for _ in range(num):
        page = random.choice(URLS)

```

```

        send_event(client_id, "page_view", {"page_location":
page, "page_title": page.split("/")[-1], "engagement_time_msec":
random.randint(500,2500)}, ua)
        if random.random() < 0.3:
            send_event(client_id, "scroll", {"page_location":
page}, ua)
            time.sleep(random.uniform(0.3,1.0))

def persona_deep(client_id, ua):
    send_event(client_id, "session_start", {}, ua)
    # довше читає, переходить багатьма сторінками, іноді додає
коментар
    num = random.randint(6,12)
    for i in range(num):
        page = random.choice(URLS)
        send_event(client_id, "page_view", {"page_location":
page, "page_title": page.split("/")[-1], "engagement_time_msec":
random.randint(2000,8000)}, ua)
        if random.random() < 0.5:
            send_event(client_id, "scroll", {"page_location":
page}, ua)
            if random.random() < 0.15:
                send_event(client_id, "add_comment",
{"page_location": page, "article_id": random.randint(1,200)},
ua)
                time.sleep(random.uniform(1.0,4.0))
            # фіксуємо user_engagement
            send_event(client_id, "user_engagement",
{"engagement_time_msec": random.randint(1000,5000)}, ua)

def persona_tabber(client_id, ua):
    send_event(client_id, "session_start", {}, ua)
    # імітуємо швидке переключення між 2-3 сторінками
    pages = random.sample(URLS, k=min(3, len(URLS)))
    for _ in range(random.randint(6,12)):
        page = random.choice(pages)
        send_event(client_id, "page_view", {"page_location":
page, "page_title": page.split("/")[-1], "engagement_time_msec":
random.randint(200,900)}, ua)
        # дуже короткі паузи – переключення вкладок
        time.sleep(random.uniform(0.1,0.5))

def persona_registrant(client_id, ua):
    send_event(client_id, "session_start", {}, ua)
    # проглядає кілька сторінок
    for _ in range(random.randint(2,5)):
        page = random.choice(URLS)
        send_event(client_id, "page_view", {"page_location":
page, "page_title": page.split("/")[-1], "engagement_time_msec":
random.randint(800,2500)}, ua)
        time.sleep(random.uniform(0.3,1.2))
    # починає форму
    send_event(client_id, "form_start", {"form_id":

```



```

"registration"}, ua)
    time.sleep(random.uniform(1.0, 3.0))
    # відправляє форму -> form_submit
    send_event(client_id, "form_submit", {"form_id":
"registration"}, ua)
    # додатково позначає як first_visit якщо треба
    send_event(client_id, "first_visit", {}, ua)

# ----- Симуляція одного користувача -----
def simulate_one_user(i):
    client_id = make_client_id()
    ua = random.choice(USER_AGENTS)
    persona = choose_persona()
    # виклик сценарію відповідно до persona
    if persona == "bouncer":
        persona_bouncer(client_id, ua)
    elif persona == "skim":
        persona_skim(client_id, ua)
    elif persona == "deep":
        persona_deep(client_id, ua)
    elif persona == "tabber":
        persona_tabber(client_id, ua)
    elif persona == "registrant":
        persona_registrant(client_id, ua)
    # невелика пауза перед завершенням сесії
    time.sleep(random.uniform(0.2, 0.6))
    return persona

# ----- Основний запуск -----
def run_simulation():
    personas_log = []
    with ThreadPoolExecutor(max_workers=THREADS) as ex:
        futures = [ex.submit(simulate_one_user, i) for i in
range(NUM_USERS)]
        for f in as_completed(futures):
            personas_log.append(f.result())

    # підсумки
    stats = {}
    for p in personas_log:
        stats[p] = stats.get(p, 0) + 1
    print("Simulation complete. Persona distribution:")
    for k, v in stats.items():
        print(f" {k}: {v}")

    # зберігаємо результат для звіту
    with open("personas_summary.csv", "w", newline="",
encoding="utf-8") as f:
        import csv
        writer = csv.writer(f)
        writer.writerow(["persona", "count"])
        for k, v in stats.items():
            writer.writerow([k, v])

```

```
if __name__ == "__main__":  
    print("Starting persona-based GA4 simulation...")  
    run_simulation()
```

Додаток Б. UML діаграми досліджуваного вебпорталу

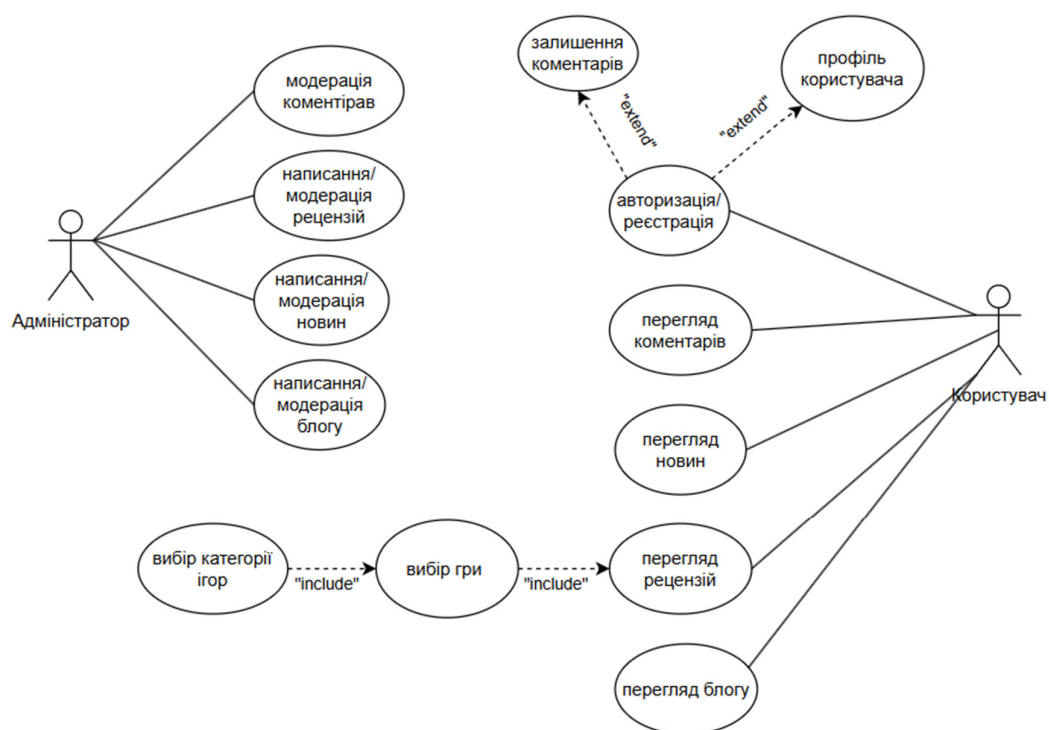


Рисунок Б.1 – Діаграма варіантів використання вебпорталу

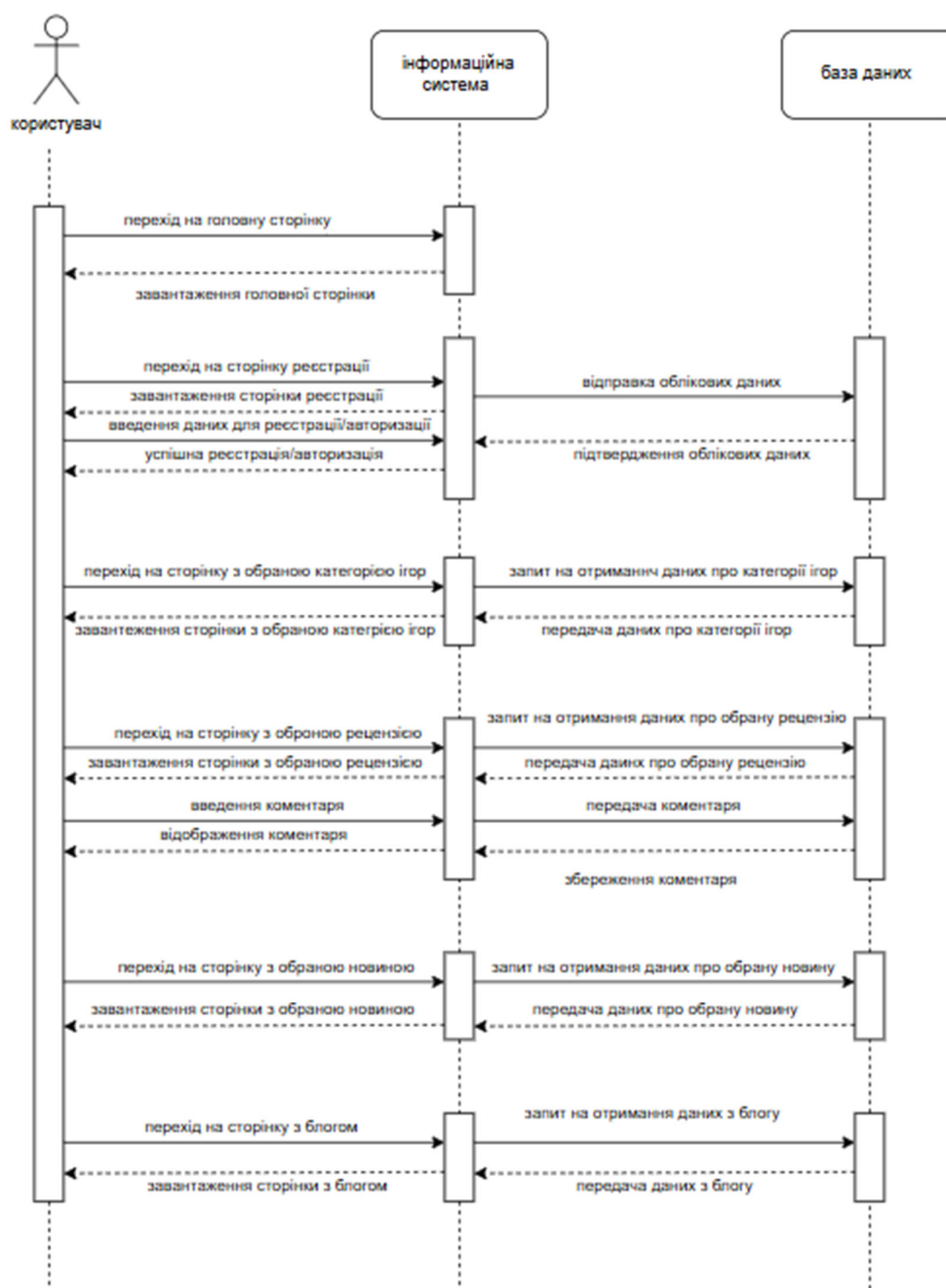


Рисунок Б.2 – Діаграма послідовності функціонування вебпорталу

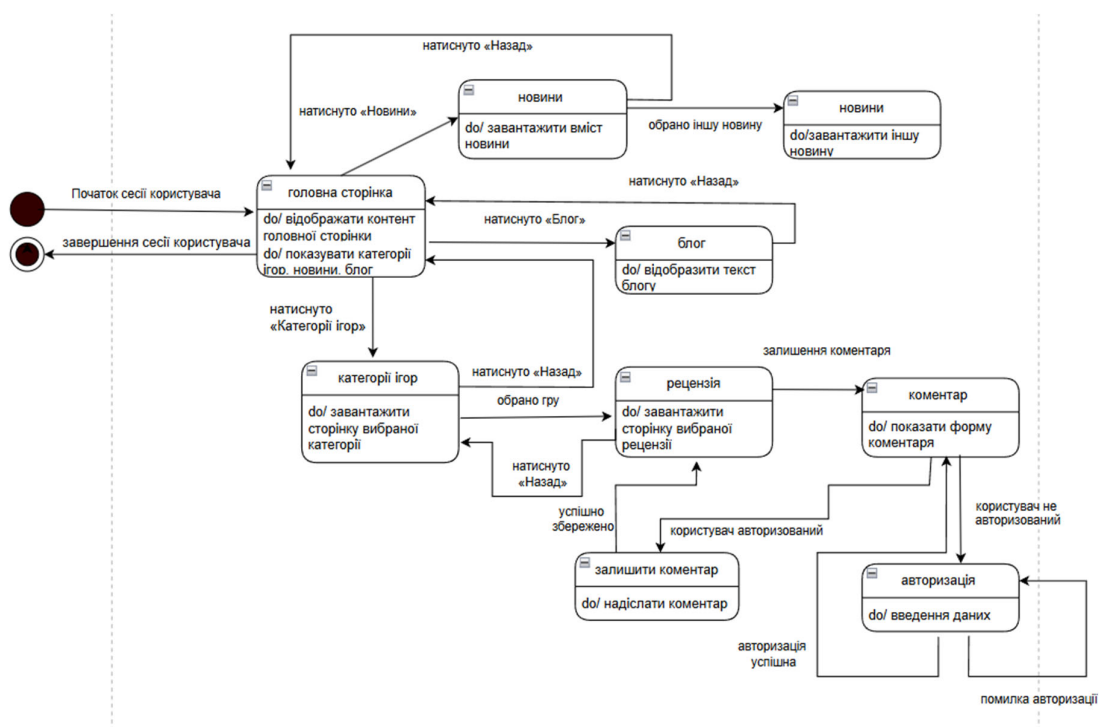


Рисунок Б.3 – Діаграма станів вебпорталу

Додаток В. Копії апробаційних матеріалів

Куклінова Софія Іванівна

*Національний університет «Одеська юридична академія»,
студентка факультету кібербезпеки та інформаційних технологій*

СИСТЕМНИЙ АНАЛІЗ ТА ОПТИМІЗАЦІЯ ЕФЕКТИВНОСТІ ВЕБПОРТАЛУ НА ОСНОВІ ПОВЕДІНКИ КОРИСТУВАЧА

У сучасних реаліях вебпортали виступають не лише засобом представлення інформації, а й ключовим інструментом інформаційного обміну, комунікації та взаємодії між користувачами і цифровими сервісами. Від ефективності їхньої роботи, зручності навігації та якості користувацького досвіду залежить рівень залучення, лояльність аудиторії та конкурентоспроможність. Розвиток інструментів аналітики, зокрема Google Analytics 4, дає змогу аналізувати поведінкові патерни користувачів, виявляти слабкі місця у структурі сайту та прогнозувати реакцію відвідувачів на контент [1].

Метою роботи є комплексна розробка та аналітичне обґрунтування підходів до системного аналізу й оптимізації вебпорталу на основі поведінкових метрик користувачів і сучасних інструментів вебаналітики. Об'єктом дослідження виступає процес оптимізації ефективності інтерактивного вебпорталу для рецензування відеоігор, розробленого на базі фреймворку Django з використанням архітектури MVT [2]. Методика дослідження базується на поєднанні системного аналізу, емпіричних спостережень, інструментальної вебаналітики та моделювання даних.

Теоретичний аналіз показав, що вебпортали є складними інформаційно-комунікаційними системами, що поєднують технологічну платформу, базу даних, програмну логіку та інтерфейс користувача [3]. Для оцінювання ефективності використовують систему метрик, які поділяють на технічні, поведінкові та бізнес-метрики. До технічних метрик відносяться час завантаження, uptime та мобільна адаптивність. Поведінкові метрики включають показник відмов, середню тривалість сесії, глибину перегляду та конверсію. Бізнес-метрики охоплюють ROI, CPA та Retention Rate [4].

Наукова новизна роботи полягає у розробці комплексного підходу, що поєднує методи системного аналізу, вебаналітики та імітаційного моделювання користувацької поведінки. Уперше запропоновано модель симуляції користувацького трафіку з використанням Python та інтеграції з Google Analytics 4 через Measurement Protocol, яка дозволяє формувати аналітичні дані для дослідження ефективності вебпорталу без залучення реальних користувачів.

У межах дослідження було створено експериментальне середовище, яке дало змогу імітувати поведінку користувачів, аналізувати дані та оцінювати вплив факторів на показники залученості. Для забезпечення аналітичної складової було інтегровано Google Analytics 4 через Measurement Protocol [5]. З метою формування статистично значущих даних було створено Python-скрипт симуляції користувацької активності, який моделював різні типи взаємодії: `page_view`, `add_comment`, `scroll`, `form_start`, `form_submit`, `user_engagement`, `session_start`, `first_visit`.

Для наближення симуляції до реальної поведінки було розроблено п'ять персон користувачів: *Bouncer* (залишає сайт після однієї сторінки), *Skimmer* (переглядає декілька сторінок з мінімальними діями), *Deep reader* (детально читає статті та залишає коментарі), *Tabber* (швидко перемикається між вкладками) та *Registrant* (проходить повний шлях реєстрації) [6]. Загальна кількість подій склала понад 15 тисяч із 1494 унікального користувача.

На основі зібраних даних було створено аналітичні діаграми за допомогою бібліотек `pandas`, `matplotlib` і `seaborn`. Стовпчаста діаграма подій показала, що користувачі найчастіше переглядають сторінки (70,3% усіх подій), але рідко взаємодіють з формою реєстрації (2,39%). Кругова діаграма типів користувачів виявила, що більшість відвідувачів належать до пасивних типів: *Skimmers* становлять 35,6%, *Deep Readers* – 26,7%, *Bouncers* – 20%, *Tabbers* – 11,1%, *Registrants* – 6,7% [7].

Теплові карти відвідувань сторінок дозволили виявити найбільш популярні розділи. Найпопулярнішими виявилися категорії «Популярні ігри», «Action» та «Adventure» з 1367-2380 переглядами. Найменшу відвідуваність мали «Добірка від автора» (247 відвідувань) і «Sports» (391 відвідування). Лінійний графік показав

зростання активності в середині симуляції з наступним спадом наприкінці, що вказує на необхідність механізмів утримання користувачів.

На основі аналізу було виділено ключові проблеми вебпорталу. Низький показник `form_submit` (2,39%) ймовірно пов'язаний з незручною формою або її непомітністю. Високий відсоток Bouncers (20%) свідчить про слабку привабливість головної сторінки. Низький `scroll` вказує на монотонний контент та відсутність візуальних блоків. Зменшення активності в часі показує відсутність стимулів для повернення користувачів [8].

Було здійснено теоретичну оптимізацію UX/UI без фактичного редизайну. Для головної сторінки запропоновано частіше оновлювати новини, додавати інтерактивні блоки з рекомендаціями, адаптувати навігацію під мобільні пристрої та переробити форму реєстрації. Для підвищення конверсії пропонується скорочення полів форми, підтримка автозаповнення та чітка кольорова контрастність кнопок [9].

Після моделювання змін прогнозується значне покращення показників. Показник відмов може знизитися з 20% до 12% (зміна на 40%). Середня тривалість сесії може зрости з 1,2 до 1,7 хвилин (покращення на 41%). Коефіцієнт надсилання форми може збільшитися з 2,39% до 3,5% (зростання на 46%). Кількість сторінок за сеанс може зрости з 3,1 до 4,4 (покращення на 42%). Коефіцієнт залученості може збільшитися з 58% до 74% (зміна на 27%). Кількість повернень може зрости з 18% до 28% (покращення на 55%).

Кореляційний аналіз показав найтісніший позитивний зв'язок між Engagement Rate та Return Visitors ($r=0,81$), між Engagement Rate та Form Submit Rate ($r=0,66$). Bounce Rate має сильну негативну кореляцію з усіма метриками (від -0,55 до -0,74), що підтверджує, що зменшення відмов прямо свідчить про покращення досвіду користувачів [10].

Проведене дослідження підтвердило доцільність використання системного підходу до аналізу вебпорталів. Використання Google Analytics 4 у поєднанні з Python дає змогу створити повну картину користувацької взаємодії. Розроблена модель симуляції є дієвим інструментом дослідження навіть за відсутності реальної аудиторії. Результати доводять, що аналітичні та теоретичні UX/UI-покращення

здатні суттєво вплинути на показники ефективності вебпорталу, збільшуючи рівень залучення та задоволеності користувачів на 27-55%.

Список використаної літератури:

1. Куклінова С. І. Розробка інтерактивного вебпорталу для рецензування відеоігор : кваліфікаційна робота бакалавра / Куклінова Софія Іванівна ; наук. кер. [Коновалов Сергій Миколайович]. Одеса : Одеський національний морський університет, 2024. С. 33–62.
2. Романишин Ю., Крупа Х. Інформаційні сервіси інтернет-порталів фандрайзингових організацій. Цифрова платформа: інформаційні технології в соціокультурній сфері. 2024. Т. 7, № 1. С. 171–187.
3. Задорожна Н. Підходи до створення та підтримки порталів. Проблеми програмування. 2006. № 2-3. С. 569–579.
4. Rahman N., Hasda A. Y., Niswaty R. Effectiveness of the use of mail application information systems at the Center for Security and Enforcement of Environmental and Forestry Law in Sulawesi Region. International Journal of Administration and Education (IJAE). 2024. P. 29–40.
5. Samocha D. Effectiveness of web-based interventions on patient empowerment: a systematic review and meta-analysis. Journal of Medical Internet Research. 2010. Vol. 12, No. 2. Article e1286.
6. Advanced Examination of User Behavior Recognition via Log Dataset Analysis of Web Applications Using Data Mining Techniques. MDPI Journals. 2023. DOI: <https://doi.org/10.3390/app13020845>
7. Website Traffic Patterns and User Behavior: A Comprehensive Study of Visitor Interactions and Engagement Metrics. Innocon Journal of Digital Research. 2023.
8. Pay Attention to Web Performance Metrics to Improve the UX. UXmatters Magazine. 2022.
9. Analysis of Web Server Logs to Understand Internet User Behavior and Develop Digital Marketing Strategies. Science Publications & Communication (SciencePubCo). 2023.

10. Бубенець І. Г., Сломінцев Д. М. Сучасні аспекти цифрового маркетингу: основні показники ефективності. БізнесІнформ. 2022. № 11. С. 271–276.

Ключові слова: системний аналіз, вебпортал, користувацька поведінка, Google Analytics 4, оптимізація UX/UI, вебаналітика, метрики ефективності.

Keywords: system analysis, web portal, user behavior, Google Analytics 4, UX/UI optimization, web analytics, performance metrics.