

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)

ДОСЛІДЖЕННЯ ТА ЗАСТОСУВАННЯ МОДЕЛЕЙ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ОЦІНКИ ПРОДУКТИВНОСТІ ІНТЕРФЕЙСУ КОРИСТУВАЧА (тема)

Виконав:
здобувач 2 року навчання,
групи ІНФМ-24-1

Кульмінський Я. К.
(прізвище, ініціали)

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

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Науковий керівник доц. Любченко В. А.
(посада, прізвище, ініціали)

Допускається до захисту

Завідувач кафедри інформатики _____
(підпис)

Кобилін О. А.
(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту

Кафедра Інформатики

Рівень вищої освіти другий (магістерський)

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

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Кульмінському Яну Костянтиновичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та застосування моделей штучного інтелекту для оцінки продуктивності інтерфейсу користувача

затверджена наказом університету від 14 листопада 2025 року № 1045Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 6 грудня 2025 р.

3. Вихідні дані до роботи мультимодальні великі мовні моделі GPT-5.1 Thinking і Gemini 2.0 Pro, літературні джерела щодо застосування мультимодальних великих мовних моделей, програмні засоби для реалізації розробленого модифікованого способу автоматизованої оцінки продуктивності інтерфейсу, набір інтерфейсів для оцінки моделями у вигляді зображень і коду.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз можливостей мультимодальних великих мовних моделей у задачах оцінки продуктивності інтерфейсів користувача.

2. Аналіз літературних джерел щодо апробації використання мультимодальних великих мовних моделей у задачах оцінки продуктивності інтерфейсів користувача.

3. Формування покрокового алгоритму способу автоматизованої оцінки продуктивності інтерфейсу.

4. Візуалізація сформованого покрокового алгоритму.

5. Розроблення програмного застосунку, що дасть можливість протестувати моделі у задачі оцінки продуктивності інтерфейсів користувача.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) актуальність проблеми застосування мультимодальних великих мовних моделей для оцінки продуктивності інтерфейсів користувача, об'єкт та мета дослідження, постановка задачі, блок-схеми алгоритму способу автоматизованої оцінки продуктивності інтерфейсу, графіки порівняння оцінок моделей із оцінками експерта, перспективи та апробація роботи.

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

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	29.09.2025	
2	Аналіз завдання, підбір літератури	30.09.25-07.10.25	
3	Аналіз літератури з досліджуваної проблеми	08.10.25-14.10.25	
4	Особливості великих мовних моделей	15.10.25-20.10.25	
5	Дослідження великих мовних моделей	21.10.25-27.10.25	
6	Програмна реалізація	28.10.25-05.11.25	
7	Обґрунтування отриманих результатів	06.11.25-11.11.25	
8	Оформлення пояснювальної записки	12.11.25-14.11.25	
9	Перевірка на нормоконтроль	01.12.25-06.12.25	
10	Перевірка на плагіат	06.12.25-07.12.25	
11	Рецензування	07.12.25-09.12.25	
12	Підготовка презентації та доповіді	10.12.25-11.12.25	
13	Занесення роботи в електронний архів	15.12.25	
14	Попередній захист кваліфікаційної роботи	15.12.25	

Дата видачі завдання 29 вересня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____
(підпис)

доц. Любченко В. А.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 84 с., 6 табл., 14 рис., 1 дод., 41 джерело.

АВТОМАТИЗОВАНА ОЦІНКА, ВЕЛИКА МОВНА МОДЕЛЬ, ІНТЕРФЕЙС КОРИСТУВАЧА, КОМП'ЮТЕРНИЙ ЗІР, МАШИННЕ НАВЧАННЯ, ОЦІНКА UX, ШІ, GEMINI 2.0 PRO, GPT-5.1 THINKING, MLLM.

Об'єктом дослідження є процес автоматичної оцінки продуктивності UI за допомогою моделей ШІ.

Метою дослідження є аналіз можливостей моделей ШІ й розробка способу автоматизованої оцінки продуктивності інтерфейсу, що базується певній системі критеріїв, з подальшою перевіркою його ефективності.

Використано мультимодальні моделі GPT-5.1 Thinking та Gemini 2.0 Pro для аналізу та оцінювання інтерфейсів користувача. Проведено серію експериментів, у межах яких порівнювалися результати автоматизованої оцінки з експертними висновками. Отримані результати візуалізовано у вигляді таблиць і графіків.

Наукова новизна роботи полягає у розробці критеріїв оцінки і підходу до автоматичної оцінки продуктивності UI, що використовує можливості MLLM. Цей підхід має підвищити швидкість, точність і об'єктивність оцінювання.

Взаємозв'язок з іншими роботами полягає у використанні сучасних методів мультимодального аналізу та продовженні досліджень у галузі взаємодії людини й комп'ютера та автоматизованої UX-аналітики.

Рекомендації щодо використання результатів роботи: застосування підходу для створення інструментів автоматичного UX/UI аудиту.

У результаті дослідження розроблено систему для автоматизованої оцінки користувацьких інтерфейсів із використанням моделей штучного інтелекту.

ABSTRACT

Explanatory note to the qualification work: 84 pages, 6 tables, 14 figures, 1 appendix, 41 sources.

AI, AUTOMATED EVALUATION, COMPUTER VISION, GEMINI 2.0 PRO, GPT-5.1 THINKING, LARGE LANGUAGE MODEL, MACHINE LEARNING, MLLM, USER INTERFACE, UX ASSESSMENT.

The object of the research is the process of automatic evaluation of user interface (UI) performance using artificial intelligence models.

The aim of the research is to analyze the capabilities of AI models and to develop a method for automated UI performance assessment based on a defined system of criteria, followed by verification of its effectiveness.

Multimodal models GPT-5.1 Thinking and Gemini 2.0 Pro were used for user interface analysis and evaluation. A series of experiments was conducted in which the results of automated evaluation were compared with expert assessments. The obtained results were visualized in the form of tables and graphs.

The scientific novelty of the work lies in the development of evaluation criteria and an approach to automatic UI performance evaluation that utilizes the capabilities of multimodal large language models (MLLMs). This approach aims to improve the speed, accuracy, and objectivity of assessment.

Interconnection with other works lies in the use of modern multimodal analysis methods and in continuing research in the field of human-computer interaction and automated UX analytics.

Recommendations for using the results: application of the proposed approach for the development of automatic UX/UI audit tools.

As a result of the research, a system for automated evaluation of user interfaces using artificial intelligence models has been developed.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	8
Вступ.....	10
1 Аналіз існуючих методів оцінки продуктивності користувацького інтерфейсу	12
1.1 Сучасні підходи до оцінки інтерфейсів користувача	12
1.1.1 Традиційні методи оцінювання	12
1.1.2 Автоматизовані підходи та методи.....	14
1.1.3 Моделювання користувацької поведінки.....	14
1.1.4 Аналіз візуальних і структурних властивостей	15
1.1.5 Мультимодальні системи оцінки.....	16
1.1.6 Основні обмеження сучасних методів для автоматизації	16
1.2 Сучасні моделі штучного інтелекту і їхні можливості для задач оцінки продуктивності користувацького інтерфейсу.....	17
1.3 Аналіз літературних джерел щодо якості існуючих підходів оцінки інтерфейсів та можливостей моделей штучного інтелекту в їхній оцінці	20
1.4 Постановка задачі дослідження	25
2 Можливості обраних моделей штучного інтелекту для оцінки продуктивності інтерфейсу користувача	27
2.1 Характеристика обраних мультимодальних великих мовних моделей	27
2.1.1 Модель GPT-5.1 Thinking.....	27
2.1.2 Модель Gemini 2.0 Pro.....	28
2.2 Спосіб застосування моделей штучного інтелекту для оцінки продуктивності інтерфейсу.....	30
2.2.1 Оцінка на основі візуальних даних (скріншоти інтерфейсу)	31

2.2.2	Оцінка на основі структурних даних (вихідний код HTML/CSS).....	33
2.2.3	Оцінка на основі мультимодальних даних (скріншот і поведінкові метрики).....	34
2.3	Модифікований спосіб автоматизованої оцінки продуктивності інтерфейсу.....	35
2.3.1	Модифікація системи критеріїв і метрик для автоматизованої оцінки.....	36
2.3.2	Етапи алгоритму автоматизованої оцінки продуктивності інтерфейсу.....	38
2.3.3	Опис функціонування та архітектури системи оцінки.....	40
2.3.4	База даних	43
3	Дослідження моделей штучного інтелекту для оцінки продуктивності інтерфейсу користувача	45
3.1	Вибір інструментальних засобів для реалізації системи.....	45
3.2	Програмна реалізація способу оцінки інтерфейсу	47
3.2.1	Створення і додання тестових даних до бази даних.....	47
3.2.2	Взаємодія із моделями.....	50
3.2.3	Апробація відповідей моделей	53
3.3	Проведення експериментів та дослідження ефективності моделей ...	53
3.3.1	Тестові дані	53
3.3.2	Отримання оцінок.....	58
3.4	Порівняльний аналіз результатів.....	69
3.4.1	Перший інтерфейс	69
3.4.2	Другий інтерфейс	71
3.5	Проведення етапу апробації моделями	73
3.6	Перспективи подальших досліджень	76
	Висновки.....	77
	Перелік джерел посилання	79
	Додаток А Промпт.....	84

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БД – база даних

СУБД – система управління базами даних

ШІ – штучний інтелект

AI – Artificial Intelligence (штучний інтелект)

API – Application Programming Interface (інтерфейс програмування застосунків)

B2B – Business-to-Business (модель комерційних відносин бізнес для бізнесу)

CNN – Convolutional Neural Network (згорткова нейронна мережа)

CMS – Content Management System (система керування вмістом)

CSS – Cascading Style Sheets (каскадні таблиці стилів)

CTA – Call To Action (заклик до дії)

CTR – Click-Through Rate (відношення числа натискань на рекламне оголошення до показів сайту)

DOM – Document Object Model (об'єктна модель документа)

GOMS – Goals, Operators, Methods, Selection rules (цілі, оператори, методи, правила вибору)

HCI – Human-Computer Interaction (людино-комп'ютерна взаємодія)

HTML – HyperText Markup Language (мова розмітки гіпертексту)

HTTP – Hypertext Transfer Protocol (протокол передачі гіпертексту)

JSON – JavaScript Object Notation (нотація об'єктів JavaScript)

IUI – Intellectual User Interface (інтелектуальний інтерфейс користувача)

LLM – Large Language Model (велика мовна модель)

LSTM – Long Short-Term Memory (довгострокова короткочасна пам'ять)

LUIM – Large User Interface Model (велика модель користувацького інтерфейсу)

MLLM – Multimodal Large Language Model (мультимодальна велика мовна модель)

MUI – Mobile User Interface (мобільний інтерфейс користувача)

NLP – Natural Language Processing (обробка природної мови)

png – Portable Network Graphics (портативна мережева графіка)

RNN – Recurrent Neural Network (рекурентна нейронна мережа)

SVG – Scalable Vector Graphics (масштабована векторна графіка)

UI – User Interface (інтерфейс користувача)

UX – User Experience (досвід користувача)

ViT – Vision Transformer (візуальний трансформер)

WCAG – Web Content Accessibility Guidelines (настанови з доступності вебвмісту)

ВСТУП

У сучасному цифровому середовищі користувацький інтерфейс є одним із найважливіших чинників, що визначають ефективність, зручність, легкість використання і загальний досвід взаємодії користувача з програмним продуктом. Продуктивність інтерфейсу напряду впливає на швидкість виконання дій, точність, задоволеність користувачів і, як наслідок, – на успіх застосунку. Тому питання об'єктивної оцінки продуктивності інтерфейсів користувача є дуже актуальним для розробників, аналітиків і дослідників у галузі людсько-комп'ютерної взаємодії.

Традиційні методи оцінки ефективності інтерфейсів базуються на суб'єктивних показниках (опитування користувачів, юзабіліті-тестування) або на статистичних спостереженнях за поведінкою користувачів. Однак такі підходи часто вимагають значних часових і людських ресурсів, а також не завжди забезпечують об'єктивність результатів. Зі збільшенням складності сучасних інтерфейсів виникає потреба в автоматизованих системах, здатних оцінювати продуктивність UI без прямої участі людини.

Технології штучного інтелекту відкривають нові можливості в цьому напрямі. Завдяки розвитку моделей комп'ютерного зору, машинного навчання й великих мовних моделей тепер можливо аналізувати і візуальні характеристики інтерфейсу, й поведінкові аспекти взаємодії користувача. Моделі штучного інтелекту здатні імітувати процеси сприйняття, прогнозувати ефективність розташування елементів, оцінювати когнітивне навантаження користувача та виявляти потенційні проблеми дизайну [1].

Незважаючи на значний прогрес у галузі автоматизованого UI-аналізу, наразі відсутні універсальні підходи до оцінки саме продуктивності інтерфейсу користувача з використанням методів штучного інтелекту. Існуючі дослідження переважно зосереджуються на оцінці юзабіліті або естетики інтерфейсу, тоді як продуктивність як комплексний показник (що поєднує швидкість, точність, навантаження й ефективність виконання задач) залишається недостатньо

формалізованою. Отже, актуальним є завдання розроблення підходу, який дозволить здійснювати автоматизовану оцінку продуктивності інтерфейсу користувача на основі аналізу візуальних, структурних і поведінкових даних із застосуванням сучасних моделей штучного інтелекту. Такий підхід допоможе суттєво скоротити час оцінювання, підвищити об'єктивність результатів і забезпечити можливість масштабованого аналізу інтерфейсів.

Задачею кваліфікаційної роботи є дослідження існуючих підходів до оцінки продуктивності користувацького інтерфейсу та розроблення методу автоматизованої оцінки на основі моделей штучного інтелекту.

Об'єктом цього дослідження є процес автоматичної оцінки продуктивності користувацького інтерфейсу за допомогою моделей штучного інтелекту. Для досягнення поставленої мети передбачається виконання низки взаємопов'язаних завдань: необхідно ретельно проаналізувати існуючі підходи до оцінювання UI, визначити, як саме можливості ШІ можуть бути використані для вимірювання продуктивності інтерфейсів, і створити нову методику автоматизованої оцінки на цій основі. Фінальні етапи включають програмну реалізацію створеної системи та проведення експериментів для підтвердження її точності й загальної ефективності.

Наукова новизна роботи полягає у розробці критеріїв оцінки і підходу до автоматичної оцінки продуктивності UI, що використовує можливості MLLM. Практична цінність дослідження полягає в тому, що розроблений підхід може бути безпосередньо інтегрований у промислові інструменти аналізу взаємодії з користувачем (UX-аналітики) або у системи автоматизованого тестування інтерфейсів. Це забезпечить дизайнерів, розробників і дослідників надійним і швидким засобом для значного підвищення якості та функціональної ефективності їхніх застосунків.

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ОЦІНКИ ПРОДУКТИВНОСТІ КОРИСТУВАЦЬКОГО ІНТЕРФЕЙСУ

1.1 Сучасні підходи до оцінки інтерфейсів користувача

Користувацький інтерфейс є основним засобом взаємодії між людиною та застосунком. Через нього користувач сприймає функціональні можливості продукту, виконує дії, отримує результати та формує загальне враження від користання. Якість інтерфейсу визначає зручність, ефективність, продуктивність і рівень задоволеності користувача. Оцінювання інтерфейсу полягає у визначенні його ефективності, зручності та відповідності потребам користувача. Його мета – виявлення сильних і слабких сторін дизайну та пошук шляхів удосконалення. Ефективний інтерфейс дає швидке й безпомилкове досягнення цілей без надмірного когнітивного навантаження.

1.1.1 Традиційні методи оцінювання

Класичні підходи до оцінки ефективності користувацьких інтерфейсів сформувалися ще на початку розвитку галузі НСІ, і тривалий час залишалися основним інструментом забезпечення якості дизайну. Їхня особливість лежить в орієнтації на експертну або емпіричну оцінку, засновану на спостереженні за користувачами, аналізі їхньої поведінки та інтерпретації результатів фахівцями у сфері UX-дизайну.

Одним із найвідоміших методів є евристичний аналіз Нільсена (Nielsen's Heuristic Evaluation), що передбачає перевірку інтерфейсу за набором базових принципів, що охоплюють видимість стану системи, зворотний зв'язок, попередження помилок, послідовність і естетичність. Метод дає змогу швидко виявити типові недоліки дизайну, проте потребує залучення досвідчених експертів, а результати здебільшого залежать від їхньої суб'єктивної оцінки [2].

Юзабіліті-тестування (usability testing) базується на спостереженні за користувачами, що виконують визначені сценарії. Під час тестів фіксуються кількість помилок, час виконання завдань і рівень задоволеності, на основі чого рахуються показники продуктивності. Метод дає високу достовірність, але є трудомістким і дорогим, адже потребує спеціального середовища, учасників й аналітиків. Це обмежує можливість масштабування. У практиці UX-дизайну також застосовуються поведінкові аналітичні метрики, отримані з інструментів вебаналітики: CTR (частота переходів), dwell time (тривалість перебування), scroll depth (глибина прокручування) і результати A/B-тестів. Вони дозволяють збирати великі обсяги даних, однак здебільшого відображають поверхневі аспекти поведінки користувача, не пояснюючи її причин. До окремого класу належать когнітивні моделі, такі як GOMS чи Keystroke-Level Model, що прогнозують час виконання завдань і кількість дій. Хоча ці підходи формалізують користувацьку діяльність, вони не враховують індивідуальних і емоційних чинників, що відіграють важливу роль у сучасному UX-дизайні.

Спільною рисою класичних методів є висока залежність від людського фактора, оскільки вони вимагають участі користувачів чи експертів і не підходять для швидкого аналізу великої кількості інтерфейсів. Також ускладнює автоматизацію й порівняння результатів між різними дослідженнями відсутність формалізованих критеріїв.

Попри обмеження, традиційні методи залишаються цінними для глибокого якісного аналізу і перевірки гіпотез, забезпечуючи контекстуальне тлумачення результатів і високу довіру до експертних висновків. Однак зростання масштабів цифрових продуктів і швидких ітерацій розробки зумовлює потребу у створенні автоматизованих систем оцінювання, бо вони поєднують точність людської експертизи з масштабованістю алгоритмічних методів. Саме інтеграція технологій ШІ, машинного навчання та комп'ютерного зору відкриває шлях до формування гібридних систем, здатних поєднувати інтерпретативні можливості людини й аналітичну потужність машинного прогнозування.

1.1.2 Автоматизовані підходи та методи

У дослідженнях проблеми автоматизованої оцінки продуктивності користувацьких інтерфейсів спостерігається тенденція до переходу від ручних експертних процедур до інтелектуальних алгоритмічних систем, що здатні самостійно аналізувати візуальну структуру, поведінкові патерни користувачів і семантику інтерфейсів. Ці системи дозволяють мінімізувати вплив людського фактора, підвищити масштабованість, забезпечити відтворюваність результатів оцінювання.

Загалом автоматизовані підходи до оцінки інтерфейсів можна умовно поділити на два основні напрями:

- моделювання користувацької поведінки за допомогою алгоритмів машинного навчання;
- аналіз візуальних і структурних властивостей інтерфейсу засобами комп'ютерного зору та глибинних нейронних мереж.

1.1.3 Моделювання користувацької поведінки

Цей напрям базується на ідеї, що закономірності поведінки користувачів можна навчити й відтворювати алгоритмічно. Моделі отримують на вхід великі обсяги даних про реальні взаємодії з інтерфейсом – рухи миші, кліки, натискання клавіш, швидкість реакції, зміни фокусу уваги – і створюють предиктивні моделі оцінки ефективності окремих елементів UI.

Алгоритми машинного навчання дозволяють прогнозувати ймовірність помилок, тривалість виконання завдань, рівень когнітивного навантаження та задоволеності користувача. Зокрема, регресійні моделі застосовуються для передбачення часу виконання дій, а рекурентні нейронні мережі (RNN) і LSTM – для моделювання послідовностей дій у складних сценаріях [3].

Перспективним напрямом є симуляція користувача (user simulation), коли нейронна мережа імітує поведінку реальної людини, створюючи синтетичні дані для тестування чи навчання системи. Такі симулятори дають змогу оцінювати UX без залучення фокус-груп, що скорочує час і витрати на дослідження.

1.1.4 Аналіз візуальних і структурних властивостей

Другий напрям автоматизованих методів базується на застосуванні комп'ютерного зору для аналізу візуальних характеристик інтерфейсу. Основна ідея полягає в тому, що якість дизайну можна оцінювати на основі об'єктивних показників зображення – пропорцій, симетрії, контрасту, кольорових поєднань, щільності елементів і зорової ієрархії.

Згорткові нейронні мережі (CNN) є основним інструментом для автоматичної класифікації та оцінки UI. Вони здатні виявляти просторові патерни, розпізнавати структуру інтерфейсу, визначати елементи керування й оцінювати гармонійність їхнього розташування. CNN-моделі ефективно застосовуються для оцінки візуальної складності й естетичної привабливості, що безпосередньо корелює із задоволеністю користувачів.

Моделі на основі візуальних трансформерів (ViT) розширюють ці можливості, аналізуючи не тільки локальні, а й глобальні зв'язки між елементами інтерфейсу. Вони використовуються для визначення зон фокусу уваги, виявлення «візуального шуму» та оцінки логічної послідовності дизайну, формуючи основу систем автоматичного аудиту UX.

Окремий підхід передбачає аналіз HTML-структури інтерфейсу за допомогою моделей обробки природної мови (NLP). Це дає виявляти семантичні зв'язки між компонентами сторінки, оцінювати складність DOM-структури, логіку навігації, доступність, поєднуючи візуальний і структурний аналіз у єдиній системі оцінювання.

1.1.5 Мультиmodalні системи оцінки

Мультиmodalні системи інтегрують кілька типів даних (зображення, текст, код і поведінкові метрики) в єдине аналітичне середовище. Вони використовують великі мовні моделі, здатні одночасно обробляти текстову, візуальну й структурну інформацію. Завдяки цьому можливе комплексне оцінювання інтерфейсу: модель аналізує зовнішній вигляд, «читає» тексти, інтерпретує функціональні зв'язки між елементами і формує висновки, близькі до людського розуміння.

Мультиmodalні моделі, такі як GPT-5.1 Thinking, Claude 3, Gemini 2.0 тощо, поєднують аналіз контенту, тексту й контексту використання, виконуючи класифікацію й аналітичне міркування, пояснення результатів і рекомендації з оптимізації дизайну.

Сучасні автоматизовані методи еволюціонують від вузькоспеціалізованих алгоритмів до універсальних когнітивних систем, що імітують експертну оцінку UI. Такі мультиmodalні підходи забезпечують швидкість, узгодженість критеріїв і масштабованість, формуючи основу інтелектуальних платформ UX-аналітики.

1.1.6 Основні обмеження сучасних методів для автоматизації

Попри значний прогрес, більшість сучасних підходів має низку суттєвих обмежень, які ускладнюють повну автоматизацію процесу оцінки. Насамперед, традиційні методи не забезпечують масштабованості: кожна перевірка потребує участі людини, до того ж, результати мають суб'єктивний характер. Аналітичні моделі на основі статистики обмежені у здатності пояснювати причини низької продуктивності або визначати, які саме елементи дизайну є проблемними.

У випадку застосування нейронних мереж, основною проблемою залишається нестача великих маркованих датасетів, які б містили приклади

інтерфейсів із об'єктивними оцінками юзабіліті. Без таких даних моделі схильні до переобучення або узагальнення лише поверхневих ознак. Крім того, більшість моделей не враховують контекст використання (тип пристрою, цільову аудиторію, сценарії взаємодії), що знижує релевантність результатів.

Ще одним бар'єром для автоматизації є відсутність єдиного формалізованого набору метрик продуктивності UI, придатного для алгоритмічного аналізу. Існуючі показники часто якісні (задоволення користувача, естетичність, когнітивне навантаження) і вимагають інтерпретації людиною.

Також важливо зазначити, що більшість автоматизованих рішень орієнтовані на оцінку ізольованих аспектів (естетики, кольору, контрасту або читабельності) замість комплексного аналізу, який би враховував взаємозв'язки між усіма елементами інтерфейсу.

1.2 Сучасні моделі штучного інтелекту і їхні можливості для задач оцінки продуктивності користувацького інтерфейсу

Стрімкий розвиток технологій штучного інтелекту суттєво змінив підхід до аналізу, проектування та оцінювання користувацьких інтерфейсів. Якщо раніше оцінка інтерфейсу здебільшого базувалася на експертних висновках і користувацьких тестах, то сьогодні з'явилася можливість автоматизувати цей процес завдяки використанню моделей, здатних самостійно аналізувати великі обсяги даних, виявляти закономірності та робити висновки про ефективність дизайну.

Моделі штучного інтелекту – це алгоритмічні системи, що імітують людські когнітивні процеси, зокрема здатність до навчання, розпізнавання закономірностей, прийняття рішень і прогнозування. У контексті оцінки інтерфейсів такі моделі можуть аналізувати як візуальні, так і поведінкові дані: структуру сторінки, кольорову гаму, пропорції елементів, взаємодію

користувачів, швидкість реакції системи тощо. Завдяки машинному навчанню, моделі здатні виявляти приховані залежності між характеристиками дизайну та показниками продуктивності. Так можна робити об'єктивні висновки без участі людини.

Найбільш поширеними напрямками застосування ШІ у сфері оцінки інтерфейсів є комп'ютерний зір, обробка природної мови (NLP), машинне навчання та мультимодальні великі мовні моделі. Комп'ютерний зір використовується для аналізу візуальної структури інтерфейсу, до прикладу, розташування елементів, контрасту, кольорових схем, симетрії, може виявляти порушення принципів юзабіліті.

Нейронні мережі можуть автоматично ідентифікувати занадто дрібні кнопки, неправильні контрасти тексту або надмірну візуальну складність сторінки.

Методи NLP застосовуються для аналізу текстових компонентів інтерфейсу – підписів кнопок, повідомлень, підказок, інструкцій. За допомогою таких моделей можна оцінити зрозумілість, тональність і відповідність мови бренду або цільовій аудиторії. Так можливо автоматично визначати, наскільки тексти у UI є інформативними, дружніми і узгодженими з принципами UX-писемності.

Машинне навчання, зокрема моделі регресії, дерева рішень або нейронні мережі, використовується для побудови моделей прогнозування ефективності. Навчившись на реальних даних про поведінку користувачів (час виконання дій, кількість помилок, кліки, шляхи навігації), система може оцінювати, як зміни в дизайні вплинуть на продуктивність або задоволеність користувача. Такий підхід активно застосовується у платформах аналітики, де поєднуються UI-дизайн із поведінковими метриками.

Окрему увагу сьогодні приділяють мультимодальним великим мовним моделям (MLLM). Вони здатні одночасно працювати з кількома типами даних – текстом, зображеннями, структурованими метриками [4]. Вони дають можливість для оцінки користувацьких інтерфейсів, оскільки модель взмозі

аналізувати візуальний макет або скріншот і враховувати текстовий контент, опис завдань користувача чи аналітичні показники взаємодії. Мультимодальна LLM може «побачити» інтерфейс, розпізнати його елементи, оцінити відповідність принципам дизайну і сформулювати текстовий висновок про зручність або проблемні ділянки.

Завдяки такій універсальності мультимодальні моделі вже застосовуються для автоматизованого UX-аудиту, тестування доступності, перевірки читабельності та логічної послідовності дизайну. Вони виявляють аномалії, некоректні підписи і елементи, що порушують стандарти взаємодії, генерують рекомендації щодо покращення. Наприклад, MLLM може запропонувати змінити формулювання тексту кнопки для кращого розуміння користувачем або виявити, що колір елемента не відповідає вимогам контрастності для користувачів із порушенням зору.

Важливою перевагою мультимодальних систем є можливість контекстного аналізу, тобто врахування взаємозв'язку між різними аспектами інтерфейсу – змістом, візуальним оформленням, метриками продуктивності. Це дає можливість отримати комплексну оцінку, що ближча до людського сприйняття, аніж традиційні алгоритмічні підходи.

Для проведення експериментальної частини дослідження необхідно обрати сучасні мультимодальні великі мовні моделі, здатні здійснювати комплексний аналіз текстових і візуальних даних. Ці моделі поєднують в собі можливості традиційних LLM в розумінні природної мови із потужними засобами обробки зображень. Вони є перспективними інструментами для задач оцінки користувацьких інтерфейсів. Вибір конкретних моделей обумовлюється їхньою відкритістю, точністю, швидкодією та підтримкою різних типів вхідних даних [5, 6].

У межах даного дослідження для аналізу ефективності інтерфейсів було обрано дві мультимодальні моделі – GPT-5.1 Thinking (OpenAI) і Gemini 2.0 Pro (Google DeepMind). Вони належать до найновішого покоління моделей, і мають високу якість інтеграції візуальних і текстових модальностей.

Модель GPT-5.1 Thinking – це мультимодальна модель, здатна обробляти текст і зображення одночасно. Вона забезпечує високий рівень точності при аналізі візуальних структур, графічних інтерфейсів і діаграм, а також дозволяє генерувати аналітичні описи інтерфейсів і пропонувати шляхи їх оптимізації.

Модель Gemini 2.0 Pro вирізняється здатністю до довготривалого контекстного аналізу, обробки великих обсягів даних і підвищеною точністю у візуально-семантичних завданнях. Вона може виконувати детальний контекстуальний опис інтерфейсів, виявляти потенційні проблеми з ергономіки та зручності, а також пропонувати шляхи покращення користувацької взаємодії.

1.3 Аналіз літературних джерел щодо якості існуючих підходів оцінки інтерфейсів та можливостей моделей штучного інтелекту в їхній оцінці

У статті [7] порівнюється ефективність оцінки зручності користування вебінтерфейсів між великими мовними моделями, зокрема GPT-5.1 Thinking, та людськими експертами, використовуючи метод евристичної оцінки за Нільсеном.

Дослідження виявило, що GPT-5.1 Thinking виявила лише 21% проблем, які знайшли експерти, але при цьому виявила 27 нових питань. Модель краще справлялася з евристичними, пов'язаними з естетикою і мінімалізмом дизайну і відповідністю системи реальному світу, але мала труднощі з виявленням проблем у гнучкості, контролі й ефективності користувача. Крім того, GPT-5.1 Thinking генерував кілька хибнопозитивних результатів через спроби передбачити проблеми.

Це дослідження демонструє потенціал GPT-5.1 Thinking у виявленні певних типів проблем у дизайні інтерфейсів, але також вказує на його обмеження, зокрема у виявленні проблем, що стосуються ефективності користувача. Робота підкреслює важливість обережного використання LLM у

евристичних оцінках та пропонує рекомендації щодо їх застосування в практиці оцінки зручності користування. Дослідження може бути корисним для розуміння сильних і слабких сторін GPT-5.1 Thinking у контексті оцінки зручності користування інтерфейсами.

У статті [8] представлено метод автоматичної оцінки мобільних інтерфейсів користувача (MUI) на основі графічного аналізу скріншотів без використання вихідного коду та участі користувачів. Пропонований підхід поєднує архітектуру DenseNet201 для автоматичного виділення ознак, метод Borderline-SMOTE для балансування датасету й класифікатор K-Nearest Neighbours для класифікації інтерфейсів як «добрі» або «погані». Для оцінки ефективності використано великі публічні датасети, модель досягає середньої точності близько 93%.

Сильними сторонами роботи є висока точність оцінки, можливість аналізувати MUI без залучення користувачів і без доступу до вихідного коду, а також інтеграція сучасних методів машинного навчання для виділення ознак і балансування даних.

Слабкими сторонами можна вважати обмеження застосування моделі лише до мобільних інтерфейсів і потенційну залежність від специфіки використовуваних датасетів. Робота демонструє ефективний підхід до автоматизованої оцінки інтерфейсів, що дозволяє підвищити якість дизайну мобільних застосунків і забезпечити краще задоволення потреб користувачів.

У роботі [9] представлено два підходи до оціночної оцінки зручності інтерфейсів користувача із застосуванням методів машинного навчання. У першій фазі дослідження використовується метод опорних векторів для оцінки статичних характеристик інтерфейсу, таких як розташування віджетів і їхні розміри.

Друга фаза реалізує глибоку Q-мережу (Deep Q-Network) для оцінки динамічних властивостей інтерфейсу, зокрема продуктивності взаємодії користувача та часу виконання завдань. Перша модель дозволяє відокремлювати

та класифікувати інтерфейси з типовими проблемами юзабіліті, а друга моделює взаємодію людини з інтерфейсом і генерує дані для подальшого аналізу.

Сильними сторонами роботи є поєднання аналізу статичних і динамічних характеристик, що дає більш комплексну оцінку зручності, і використання машинного навчання для автоматизації процесу, що знижує витрати часу й ресурсів.

Слабкими сторонами можна вважати обмежену здатність моделей охоплювати всі можливі аспекти поведінки користувачів і залежність від якості вхідних даних. Робота демонструє практичну і наукову цінність застосування комп'ютерних моделей для оцінки UI, даючи можливість швидких і кількісних оцінок складних інтерфейсів.

Дослідження [10] присвячене застосуванню ШІ для автоматизованого тестування юзабіліті. Було розроблено API на базі GPT-5.1 Thinking, яке аналізує скріншоти інтерфейсів і виявляє проблеми у видимості елементів, запобіганні помилок та зворотному зв'язку системи.

Тестування проводилось на чотирьох онлайн-конвертерах одиниць і порівнювалось із оцінками студентів. Результати показали, що ШІ швидко і послідовно визначає базові проблеми, тоді як люди додатково помічають контекстно-залежні нюанси. Стаття підкреслює ефективність гібридного підходу, що поєднує автоматизовану оцінку й людську експертизу, забезпечуючи більш повне тестування юзабіліті.

У статті [11] досліджується роль штучного інтелекту у оцінці UX/UI із використанням ChatGPT-4 Vision для аналізу п'яти корейських онлайн-рітейл платформ. Модель поєднує лінгвістичні та візуальні можливості для синтезу й інтерпретації елементів UX/UI, оцінюючи відповідність стандартам дизайну і виявляючи області для покращення.

Дослідження показало, що платформи різняться за дотриманням евристик Нільсена й актуальних трендів дизайну, а використання ШІ дозволяє отримати комплексну оцінку інтерфейсів. Дослідження демонструє ефективність ШІ-

методів у проведенні глибокого аналізу UX/UI та надає приклад інтеграції великих моделей зору-мови у процес оцінки інтерфейсів.

У роботі [12] представлено UIClip – модель для оцінки якості дизайну інтерфейсів користувача на основі скріншотів і текстових описів. Для навчання моделі було сформовано великий датасет шляхом автоматизованого збору даних, синтетичного доповнення й оцінок людей. UIClip присвоює числову оцінку дизайну й генерує рекомендації щодо його покращення. Порівняння з оцінками 12 дизайнерів показало високу узгодженість результатів моделі з експертними оцінками, перевершуючи базові методи. Основні переваги роботи – великий і різноманітний датасет, мультимодальний підхід (зображення і текст) і можливість оцінювати й пропонувати зміни.

До обмежень можна віднести залежність від людських оцінок при формуванні датасету та невелику кількість експертів для перевірки. Робота демонструє ефективність автоматизованих методів оцінки UI і відкриває шлях до інтеграції таких моделей у системи генерації коду або підказок дизайну.

У статті [13] розглядається ScreenAI, модель зору-мови для аналізу інтерфейсів користувача й задач, орієнтованих на візуальну інформацію. ScreenAI одночасно обробляє зображення і текст, що дозволяє моделі розуміти розташування елементів, логіку взаємодії й контекст.

Модель показує високу точність у класифікації інтерфейсів, пошуку прикладів і формуванні рекомендацій щодо дизайну. Переваги роботи включають мультимодальний аналіз, здатність розпізнавати складні взаємозв'язки між елементами UI і контекстом, а також потенціал для підтримки дизайнерів у прийнятті рішень. Недоліками є висока складність тренування й потреба у великих наборах даних. Дослідження демонструє перспективність інтеграції таких моделей у автоматизовану оцінку і системи рекомендацій UI.

У статті [14] пропонується концепція Large User Interface Models (LUIMs) для генерації інтерфейсів і прогнозування їх зручності у мобільних застосунках за допомогою ШІ. Автори проаналізували 13 сучасних інструментів генерації UI і виявили, що існуючі моделі недостатньо враховують ключові атрибути

юзабіліті: ефективність, легкість навчання, задоволеність і запам'ятовуваність. LUIM об'єднує великі мовні, візуальні й кодові моделі і інтегрує дані про поведінку користувачів і мультимодальні сенсорні входи.

Переваги підходу – системний підхід до інтеграції великих моделей, увага до ключових аспектів UX і потенціал для прискорення фронтенд-розробки й персоналізації інтерфейсів.

Слабкі сторони – складність інтеграції, нестача узгоджених UX-датасетів і складнощі моделювання реалістичної взаємодії користувача. Робота демонструє перспективи автоматизованої генерації UI та прогнозування їх зручності, а також окреслює основні наукові й практичні виклики.

У роботі [15] запропоновано застосування комп'ютерного зору й ШІ для автоматизованого тестування інтерфейсів. Моделі розпізнають наявність елементів UI, їхнє розташування та відповідність макету специфікаціям, що дозволяє швидко знаходити помилки у розташуванні кнопок, полів вводу й інших елементів.

Переваги підходу – прискорення тестування й підвищення точності перевірки відповідності дизайну специфікаціям.

Недоліком може бути обмеженість у розпізнаванні складних інтерактивних сценаріїв і залежність від якості зображень макетів. Дослідження демонструє практичну користь застосування комп'ютерного зору для контролю UI і зменшення потреби у ручному тестуванні.

У статті [16] проведено систематичне картографування досліджень у галузі інтелектуальних інтерфейсів користувача (IUI) та їх оцінки. Проаналізовано 211 робіт за період з 2012 по 2022 роки, зосереджених на покращенні UX і юзабіліті з використанням ШІ.

Визначено ключові характеристики IUI – адаптація, подання й інтелект, а також найпоширеніші дії: адаптація, логічне міркування і подання. Більшість оцінок проводилася через експерименти й опитування, проте рідко враховувалося одночасне оцінювання UX і юзабіліті. Виявлено дефіцит інструментів, методів і метрик, спеціально адаптованих для IUI. Стаття дозволяє

окреслити сучасний стан досліджень у сфері інтелектуальних інтерфейсів, виявити тенденції й прогалини у методах оцінки.

Проведено аналіз літературних джерел, і він показує, що мультимодальні системи здатні одночасно аналізувати текст і зображення, демонструють високу ефективність у виявленні проблем юзабіліті, формуванні рекомендацій і прогнозуванні якості дизайну.

Дослідження вказують на наявність обмежень, пов'язаних із точністю, контекстністю і необхідністю людської участі. Таким чином, підтверджується актуальність використання моделей ШІ для оцінки користувацьких інтерфейсів і можливість вдосконалення способів оцінки інтерфейсів з використанням ШІ.

1.4 Постановка задачі дослідження

Таким чином, автоматизована оцінка продуктивності користувацького інтерфейсу з використанням моделей штучного інтелекту є актуальним завданням для прискорення циклів розробки та підвищення об'єктивності оцінки якості цифрових продуктів.

Прийнято рішення щодо розробки й експериментальної апробації автоматизованого методу оцінки продуктивності UI, який базується на аналізі візуальних даних (скріншотах, макетах, верстці сторінок) за допомогою обраних моделей штучного інтелекту.

Об'єктом дослідження є процес автоматичної оцінки продуктивності користувацького інтерфейсу за допомогою моделей ШІ.

Метою дослідження є аналіз можливостей моделей ШІ й розробка способу автоматизованої оцінки продуктивності інтерфейсу, що базується певній системі критеріїв, з подальшою перевіркою його ефективності.

Для досягнення мети необхідно вирішити такі завдання:

- провести аналіз сучасних підходів і методів, що застосовуються для оцінки продуктивності користувацьких інтерфейсів, і визначити їхні основні обмеження для автоматизації;
- здійснити огляд літературних джерел, що стосуються використання моделей машинного навчання та комп'ютерного зору для аналізу візуальних елементів UI і прогнозування показників юзабіліті;
- сформулювати або модифікувати систему критеріїв і метрик, що буде придатна для автоматизованої оцінки продуктивності інтерфейсу на основі обробки його візуального подання;
- обрати конкретні моделі штучного інтелекту, придатні для реалізації поставленої задачі (наприклад, для виявлення когнітивно складних елементів або прогнозування естетичної якості);
- розробити покроковий алгоритм автоматизованої оцінки продуктивності інтерфейсу з використанням обраних моделей ШІ;
- розробити програмний прототип системи, що реалізує запропонований метод оцінки, забезпечуючи можливість введення тестових даних (зображень/верстки) та отримання кількісної оцінки;
- провести експериментальне дослідження, порівнявши результати роботи розробленого підходу з відомими або експертними оцінками, для підтвердження його точності й ефективності.

2 МОЖЛИВОСТІ ОБРАНИХ МОДЕЛЕЙ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ОЦІНКИ ПРОДУКТИВНОСТІ ІНТЕРФЕЙСУ КОРИСТУВАЧА

2.1 Характеристика обраних мультимодальних великих мовних моделей

2.1.1 Модель GPT-5.1 Thinking

Модель GPT-5.1 Thinking, розроблена компанією OpenAI, є визначним досягненням у сфері великих мультимодальних моделей, інтегруючи обробку тексту, аудіо і зображень на нативному рівні. Її архітектура являє собою єдину нейронну мережу, навчену одночасно обробляти та генерувати вихідні дані у всіх трьох модальностях, що суттєво відрізняє її від попередніх підходів, де мультимодальність часто досягалася шляхом композиції окремих моделей для різних типів даних. Ця уніфікована архітектура забезпечує когерентне і глибоке контекстуальне розуміння взаємозв'язків між візуальними й текстовими елементами, що є критично важливим для комплексного оцінювання користувацьких інтерфейсів.

У контексті задач оцінки продуктивності UI, ключові можливості GPT-5.1 Thinking зосереджені на її здатності до просунутого візуального міркування (Visual Reasoning). Модель демонструє виняткову ефективність у двох основних напрямках:

По-перше, детальний аналіз структури й естетики інтерфейсу. GPT-5.1 Thinking здатна сприймати скріншоти або макети UI, розпізнавати окремі компоненти (кнопки, поля введення, навігаційні панелі) та їхні просторові відношення. Вона може оцінювати візуальні атрибути, такі як контрастність, кольорова гама, вирівнювання і дотримання принципів дизайну (наприклад, юзабіліті-евристик). Наприклад, при поданні зображення інтерфейсу, модель може автоматично ідентифікувати порушення вимог доступності, виявляючи низьку контрастність тексту чи некоректний розмір цільових областей для дотику. Це перетворює модель на автоматизованого UX-аудитора.

По-друге, контекстне розуміння функціональності. Завдяки інтеграції з природною мовою, GPT-5.1 Thinking «бачить» елементи і «розуміє» їхнє функціональне призначення й місце в загальному користувацькому сценарії. Модель може приймати на вхід зображення інтерфейсу разом із текстовим запитом, що описує завдання користувача (наприклад, «Як завершити процес оформлення замовлення?»). На основі цієї інформації GPT-5.1 Thinking здатна проаналізувати логіку навігації, послідовність кроків і виявити потенційні «больові точки» (pain points) або місця, де користувач може зіткнутися з когнітивним навантаженням. Її здатність до нюансованого розуміння дозволяє їй генерувати високоякісні, релевантні й обґрунтовані рекомендації щодо покращення, що за своєю якістю наближаються до висновків експерта.

Технічні переваги GPT-5.1 Thinking також включають значну оптимізацію швидкості й вартості обробки порівняно з її попередниками. Це робить її практичною для застосування у широкомасштабних автоматизованих системах оцінки UI. Хоча вона демонструє високі результати на багатьох задачах, що вимагають мультимодального міркування, її використання у критичних UX-задачах вимагає ретельного валідування, оскільки, як всі MLLM, вона може демонструвати розбіжності з людським сприйняттям у суб'єктивних оцінках, таких як естетична привабливість, емоційний вплив дизайну тощо. Проте, її універсальність і нативна мультимодальність роблять GPT-5.1 Thinking еталонною моделлю для подальших досліджень у галузі автоматизованої оцінки користувацьких інтерфейсів [17–19].

2.1.2 Модель Gemini 2.0 Pro

Модель Gemini 2.0 Pro, розроблена Google DeepMind, є ключовим гравцем серед мультимодальних великих мовних моделей (MLLM), особливо вирізняючись своєю революційною довжиною контекстного вікна. Завдяки архітектурі, яка ефективно використовує принцип Мішани Експертів

(Mixture-of-Experts, MoE), ця модель здатна обробляти до одного мільйона токенів (з піковим потенціалом до десяти мільйонів токенів), що є безпрецедентним показником для комерційних моделей. Ця характеристика кардинально змінює підхід до оцінки складних користувацьких інтерфейсів, дозволяючи інтегрувати значно більші обсяги даних для аналізу [20].

Універсальність Gemini 2.0 Pro полягає в її здатності нативно обробляти та взаємопов'язувати текст, зображення, аудіо й навіть відео (як послідовності кадрів), що робить її ідеальною для комплексного, мультимодального аналізу UX/UI. Її застосування у сфері оцінки інтерфейсів концентрується на потужних можливостях.

Gemini 2.0 Pro демонструє глибокий аналіз повних користувацьких шляхів (End-to-End User Flow Analysis). Завдяки надвеликому контекстному вікну, Gemini 2.0 Pro може приймати на вхід повну послідовність скріншотів, що відображають увесь шлях користувача від початку до завершення певної задачі (наприклад, реєстрація, вхід, налаштування профілю, багатоетапне оформлення замовлення). Модель здатна виявляти проблеми з логічною послідовністю, узгодженістю елементів між екранами й ефективністю навігації в цілому, що раніше було доступно лише в рамках трудомістких ручних аудитів. Це дозволяє автоматично оцінювати, як один екран впливає на досвід взаємодії з наступним.

Gemini 2.0 Pro має інтеграцію візуально-семантичної інформації з поведінковими даними. Модель демонструє підвищену точність у візуально-семантичних завданнях, здатна розпізнавати, що зображено на екрані (візуальний аспект) й розуміти, чому цей елемент має таке розташування й як він функціонально пов'язаний з іншими компонентами (семантичний аспект). В умовах експериментального дослідження це дозволяє Gemini 2.0 Pro ефективно співвідносити візуальні характеристики UI (отримані зі скріншотів) з кількісними метриками продуктивності (доприкладу, часом виконання завдання, коефіцієнтом завершення, кількістю кліків), які можуть бути надані в текстовому або структурованому форматі в межах того ж контекстного вікна.

Використання архітектури MoE дає Gemini 2.0 Pro високу якість міркування і ефективну обробку даних. Її здатність до високоточного відкликання інформації (Needle In A Haystack тести) на екстремально довгих контекстних послідовностях гарантує, що важливі, але малопомітні деталі інтерфейсу (наприклад, дрібний шрифт, прихований елемент) не будуть втрачені при оцінці загального макету. Gemini 2.0 Pro є надзвичайно потужним інструментом для виявлення тонких, прихованих проблем юзабіліті, що можуть суттєво впливати на кінцеву продуктивність користувача [21–24].

2.2 Спосіб застосування моделей штучного інтелекту для оцінки продуктивності інтерфейсу

Застосування мультимодальних великих мовних моделей для оцінки продуктивності користувацьких інтерфейсів базується на принципі «LLM як суддя» (LLM as a Judge), де модель замість людини-експерта здійснює аналіз, міркування та формулює оціночний висновок [25].

Процес оцінки продуктивності за допомогою MLLM складається із трьох етапів:

Етап 1. Формування вхідних мультимодальних даних: підготовка скріншотів, вихідного коду (HTML/CSS) або поведінкових метрик.

Етап 2. Контекстуалізація завдання (Prompt Engineering): створення деталізованого та структурованого промпту, який включає роль моделі, контекст інтерфейсу, метрики оцінки (наприклад, евристики юзабіліті).

Етап 3. Генерація оцінки і обґрунтування: отримання кількісної оцінки й якісного обґрунтування від моделі.

На рисунку 2.1 зображено схему процесу оцінки продуктивності за допомогою MLLM.

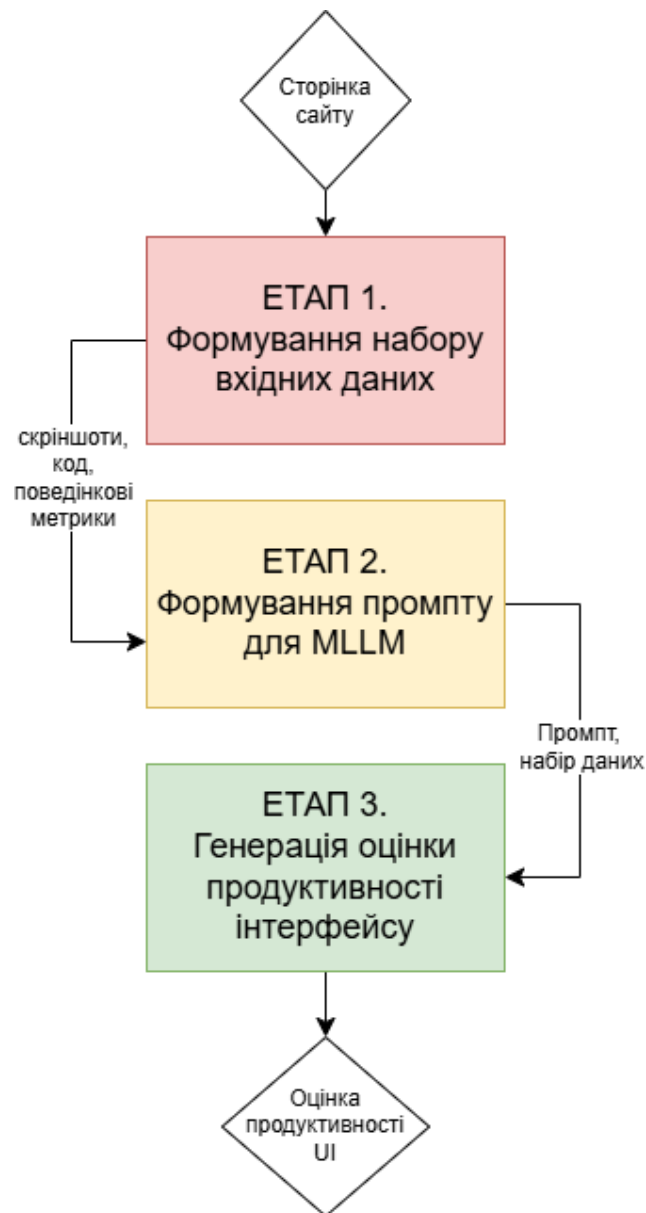


Рисунок 2.1 – Процес оцінки продуктивності за допомогою MLLM

2.2.1 Оцінка на основі візуальних даних (скріншоти інтерфейсу)

Оцінка продуктивності інтерфейсу, поданого у вигляді скріншота сторінки сайту (зображення), використовує можливості MLLM до візуального розпізнавання, аналізу макета та семантичного розуміння елементів. Цей метод дозволяє моделі виступати в ролі візуального UX-аудитора. Для отримання результатів від моделі, використовується детальний промпт.

Мета промпта – змусити модель послідовно виконати ролі UX-експерта, сфокусувавши її увагу на ключових критеріях юзабіліті.

Текст промпту може звучати так, як зображено на рисунку 2.2.

«Ти – провідний UX/UI експерт із глибоким знанням евристик юзабіліті Якоба Нільсена і стандартів доступності WCAG 2.1.
 Твоє завдання – провести комплексну оцінку наданого скріншота користувацького інтерфейсу з точки зору його потенційної продуктивності й зручності.
 Контекст: [Детальний опис контексту, наприклад: «Це сторінка оплати в мобільному застосунку електронної комерції. Оціни її для користувача, який поспішає завершити транзакцію.»]
 Критерії оцінки (оцінка від 1 до 5, де 5 – найкраща):
 - видимість стану системи (Visibility of System Status): наскільки чітко відображається поточний стан (прогрес, помилки);
 - узгодженість і стандарти (Consistency and Standards): чи відповідають елементи загальноприйнятим конвенціям і внутрішній узгодженості;
 - естетика та мінімалізм дизайну (Aesthetic and Minimalist Design): чи відсутня надлишкова інформація або візуальний шум;
 - запобігання помилкам (Error Prevention): наскільки дизайн допомагає користувачеві уникнути помилок;
 - розпізнавання замість згадування (Recognition rather than Recall): чи є елементи, дії й опції легко розпізнаваними.
 Завдання:
 - проаналізуй візуальну структуру скріншота (кольори, контраст, розміри елементів, ієрархія);
 - надай кількісну оцінку за кожним із п'яти критеріїв;
 - сформулюй детальний обґрунтований висновок (критику) по кожному критерію, вказуючи, які елементи потребують покращення.
 Формат відповіді:
 Критерій 1: [Оцінка]/5. Обґрунтування: [Текст]
 Критерій 2: [Оцінка]/5. Обґрунтування: [Текст]
 Загальний висновок: [Текст, що узагальнює основні проблеми та рекомендації.]»

Рисунок 2.2 – Текст промпту для оцінки на основі скріншотів

2.2.2 Оцінка на основі структурних даних (вихідний код HTML/CSS)

Аналіз вихідного коду (HTML і CSS) дозволяє моделям оцінити інтерфейс з точки зору його технічної якості, доступності на рівні структури й потенційної продуктивності завантаження. Це особливо актуально для Gemini 2.0 Pro, здатного обробляти великі кодові блоки. Для отримання відповіді від моделі, також використовується промпт [26–28].

Мета промпта – перевірити технічну відповідність коду стандартам UX/доступності, а не тільки його функціональність.

Приклад тексту промпту зображено на рисунку 2.3.

```
«Ти експерт із фронтенд-розробки та вебдоступності
(WCAG 2.1 A, AA).
Тобі надано фрагмент HTML і CSS коду користувачького
інтерфейсу.
Твоє завдання – оцінити його структуру, семантику,
доступність та потенційний вплив на продуктивність.
Вхідні дані (код):
[Повний фрагмент HTML та відповідний CSS]
Критерії оцінки (оцінка від 1 до 5, де 5 – найкраща):
- семантика HTML (Semantic HTML): наскільки коректно
використовуються теги (<button>, <h1>, <nav>, <aria-
label> тощо);
- доступність (Accessibility): чи є критичні
порушення WCAG (відсутність alt-текстів, некоректне
фокусування, проблеми з читанням з екрана);
- продуктивність коду (Code Performance): наскільки
структура CSS і HTML сприяє швидкому рендерингу
(наприклад, уникнення надмірної глибини DOM,
ефективність CSS-селекторів).
Завдання:
- проаналізуй наданий код і вкажи на конкретні рядки,
де знайдено порушення;
- надай кількісну оцінку за кожним із трьох
критеріїв;
- сформулюй практичні рекомендації щодо виправлення
знайдених проблем.
Формат відповіді:
Критерій 1: [Оцінка]/5. Обґрунтування: [Текст із
посиланнями на номери рядків коду]
Критерій 2: [Оцінка]/5. Обґрунтування: [Текст із
прикладями, як це виправити]
Загальний висновок: [Узагальнення технічної якості та
головних пріоритетів виправлення]».
```

Рисунок 2.3 – Текст промпту для оцінки на основі вихідного коду

2.2.3 Оцінка на основі мультимодальних даних (скріншот і поведінкові метрики)

Це найбільш комплексний метод, і він використовує унікальну перевагу MLLM – здатність одночасно обробляти візуальний вхід (скріншот) та структурований текстовий вхід (поведінкові метрики). Це дозволяє моделі встановлювати причинно-наслідкові зв'язки між дизайном (візуальний аспект) і фактичною продуктивністю користувача (метрики) [29, 30].

Мета промпта – провести перехресний аналіз, пояснюючи причину отриманих метрик через візуальні проблеми. Приклад тексту промпту зображено на рисунку 2.4.

Ти висококваліфікований UX-аналітик, який проводить А/В тестування.
 Вам надано:
 - візуальний макет: скріншот інтерфейсу;
 - поведінкові метрики: структуровані дані з реального користувацького тестування.
 Вхідні метрики:
 - коефіцієнт конверсії/завершення завдання: [Наприклад: 55%];
 - середній час виконання завдання: [Наприклад: 45 секунд];
 - коефіцієнт помилок (натискання невірних елементів): [Наприклад: 15% користувачів];
 Завдання:
 - проаналізуй скріншот та інтерпретуйте надані метрики;
 - встанови кореляцію, поясни, які саме візуальні або текстові елементи на скріншоті (дизайн) ймовірно спричинили отримані метрики (наприклад, чому низький коефіцієнт конверсії, або чому високий час виконання);
 - надай кількісну оцінку (загальна оцінка продуктивності від 1 до 5);
 - сформулюй гіпотезу покращення, запропонуй конкретну зміну дизайну для підвищення найбільш критичної метрики.
 Формат відповіді:
 Загальна оцінка продуктивності: [Оцінка]/5.
 Аналіз кореляції:
 [Назва Метрики]: [Значення]. Ймовірна візуальна причина: [Пояснення, з посиланням на конкретні елементи на скріншоті (кольори, розмір, розташування)].
 Гіпотеза покращення: [Опис рекомендованої зміни дизайну й прогнозований вплив на метрику]

Рисунок 2.4 – Текст промпту для оцінки на основі отриманих метрик і скріншоту

2.3 Модифікований спосіб автоматизованої оцінки продуктивності інтерфейсу

Для підтвердження доцільності використання моделей штучного інтелекту у задачах оцінювання ефективності користувацьких інтерфейсів у дослідженні будуть залучені експерти-практики в галузі UX/UI. Вони здійзнять незалежну оцінку результатів, отриманих від моделей, що дасть змогу порівняти якість суджень MLLM із якістю експертного підходу. На основі такого порівняння можна буде зробити висновок щодо реального потенціалу мультимодальних великих мовних моделей у даній сфері.

Для підвищення якості відповідей моделей запропоновано вдосконалити існуючий підхід до оцінювання інтерфейсів, який використовують LLM та MLLM. Модифікований спосіб автоматизованої оцінки продуктивності UI спирається на поєднання когнітивного підходу експертного аналізу з можливостями мультимодальних мовних моделей, що здатні обробляти як текстову, так і візуальну інформацію.

Головна ідея підходу полягає у моделюванні процесу мислення UX-експерта: враховуються числові оцінки критеріїв, глибоке семантичне розуміння інтерфейсу, його логічної структури, функціональності, відповідності сценаріям використання та зручності для користувача. Таким чином, система оцінювання виходить за межі традиційних методів, які здебільшого ґрунтуються на поведінкових показниках (наприклад, час виконання завдання, кількість кліків, частота помилок) і часто ігнорують когнітивні, інформаційні та естетичні аспекти користувацької взаємодії.

Залучення MLLM дозволяє значно розширити спектр аналізованих характеристик інтерфейсу – від оцінки візуального подання та сприйняття інформаційної ієрархії до аналізу консистентності елементів, доступності та відповідності стандартам юзабіліті. У перспективі це може забезпечити часткову або повну автоматизацію експертної оцінки, зменшити витрати ресурсів та прискорити процес тестування інтерфейсів на ранніх етапах розробки [31].

2.3.1 Модифікація системи критеріїв і метрик для автоматизованої оцінки

Традиційна оцінка юзабіліті ґрунтується на евристичних Якоба Нільсена, які мають якісний характер і передбачають участь людини-експерта. Для автоматизованої оцінки інтерфейсу за допомогою мультимодальних моделей ці принципи трансформуються у кількісно вимірювані критерії, адаптовані для аналізу зображень, коду та текстових елементів інтерфейсу.

Запропонована модифікована система критеріїв поєднує класичні евристики Нільсена (набір універсальних принципів оцінки юзабіліті, що описують, наскільки інтерфейс є зрозумілим, передбачуваним і дружнім до користувача) з аспектами, які можуть бути ефективно розпізнані штучним інтелектом за допомогою комп'ютерного зору, розпізнавання тексту й семантичного аналізу [32].

У таблиці 2.1 наведено запропоновані критерії.

Таблиця 2.1 – Критерії оцінки інтерфейсів

Класична евристика	Адаптований критерій MLLM	Опис оцінювання	Оцінка
1	2	3	4
Видимість стану системи	Наочність зворотного зв'язку й інформування користувача	Наявність повідомлень, статусів, прогресу, підтверджень	1–5
Узгодженість і стандарти	Візуальна консистентність стилю	Наявність повідомлень, статусів, прогресу, підтверджень	1–5
Запобігання помилкам	UX-ризиків й ймовірність виникнення помилок	Чіткість форм, СТА, валідація, попередження	1–5

Продовження таблиці 2.1

1	2	3	4
Естетика та мінімалізм	Візуальна чистота та когнітивне навантаження	Баланс контенту, шум, використання простору	1–5
Розпізнавання замість згадування	Зрозумілість та інтуїтивність елементів керування	СТА впізнавані, зрозумілі іконки, логічність архітектури	1–5
Доступність (WCAG)	Технічна і візуальна доступність	Контрастність, розміри, шрифти, ALT, клавіатурна навігація	1–5
Новий критерій	Сучасність інтерфейсу	Відповідність сучасним UI-гайдам, візуальні тренди, responsive-дизайн	1–5
Новий критерій	Відповідність аудиторії й сценарію	Чи інтерфейс відповідає бізнес-меті, tone-of-voice, юзерам	1–5

Важливим аспектом є агрегована метрика продуктивності інтерфейсу – UI Productivity Index (I_{UI}), що формується шляхом зваженого усереднення кількісних оцінок, отриманих від MLLM за кожним критерієм.

Ця метрика розраховується за формулою

$$I_{UI} = \frac{\sum_{i=1}^n C_i}{n}, \quad (2.1)$$

де C_i – кількісна оцінка i -го критерію;

n – кількість критеріїв.

Така формула дозволяє стандартизувати результат, що забезпечує можливість порівняння інтерфейсів між собою.

2.3.2 Етапи алгоритму автоматизованої оцінки продуктивності інтерфейсу

На основі обраних моделей (GPT-5.1 Thinking, Gemini 2.0 Pro) та модифікованої системи метрик запропоновано алгоритм автоматизованої оцінки продуктивності інтерфейсу, який реалізує принцип інтегрованої мультимодальної обробки даних.

Алгоритм базується звичайному алгоритмі процесу оцінки продуктивності за допомогою MLLM, але розширяє його і включає 5 етапів:

Етап 1. Підготовка вхідних даних.

Користувач збирає усю необхідну інформацію про інтерфейс – скріншоти, фрагменти HTML/CSS-коду, метрики поведінки користувачів. Дані нормалізуються, очищуються та перетворюються у формат, придатний для подальшої обробки моделлю.

Етап 2. Формування промпта.

Система автоматично формує деталізований запит до моделі з урахуванням типу даних. Промпт включає інструкцію щодо ролі моделі («Ти – UX-дослідник»), і текст, подібний наведеним раніше прикладам промптів), опис критеріїв оцінювання, очікуваний формат відповіді й приклад структури вихідних даних (JSON).

Етап 3. Обробка даних моделями.

Моделі аналізують об'єднані дані, застосовуючи мультимодальні можливості (візуальне розпізнавання, семантичний аналіз, текстове тлумачення).

На цьому етапі GPT-5.1 Thinking або Gemini 2.0 Pro формують структурований звіт у вигляді таблиці або JSON-об'єкта з оцінками та обґрунтуваннями.

Етап 4. Апробація результатів.

На етапі апробації здійснюється взаємна оцінка результатів мультимодальних моделей (GPT-5.1 Thinking, Gemini 2.0 Pro) з метою підвищення об'єктивності й узгодженості висновків.

Кожна модель виконує метааналіз отримує відповіді інших моделей і виконує аналіз – виявляє логічні чи методологічні недоліки, уточнює аргументацію та коригує підсумкові оцінки. Процес реалізується за принципом крос-верифікації, коли результати однієї моделі стають вхідними даними для іншої.

Етап 5. Агрегація та візуалізація.

Розраховується підсумковий UI Productivity Index (I_{UI}), зважуючи оцінки за формулою 2.1. Результати відображаються у вигляді таблиць, діаграм або теплових карт, що демонструють сильні й слабкі сторони інтерфейсу.

На рисунку 2.5 зображено схему модифікованого алгоритму автоматизованої оцінки продуктивності інтерфейсу.

Розроблений прототип дає змогу реалізувати повний цикл оцінки – від аналізу зображення чи верстки до отримання кількісної оцінки продуктивності, що може бути використана для порівняння різних версій інтерфейсу або оптимізації UI-рішень у процесі розробки.

Запропонована методика забезпечує автоматизацію процесу оцінки й створює основу для формування нового покоління інструментів UX-аналітики, здатних поєднувати семантичне розуміння інтерфейсу з об'єктивною кількісною оцінкою.

Вона забезпечує інтеграцію традиційних кількісних метрик і якісних висновків, згенерованих моделями штучного інтелекту, що прискорює процес оцінки продуктивності інтерфейсів і підвищує точність і глибину оцінювання, коли експерт-людина може пропустити деякі проблемні місця.

Окрім того, вона має покращити існуючий простий спосіб отримання оцінки від моделей штучного інтелекту, бо впроваджує нові етапи – оцінку декількома моделями та критичний аналіз однією моделлю відповідей інших моделей [33, 34].

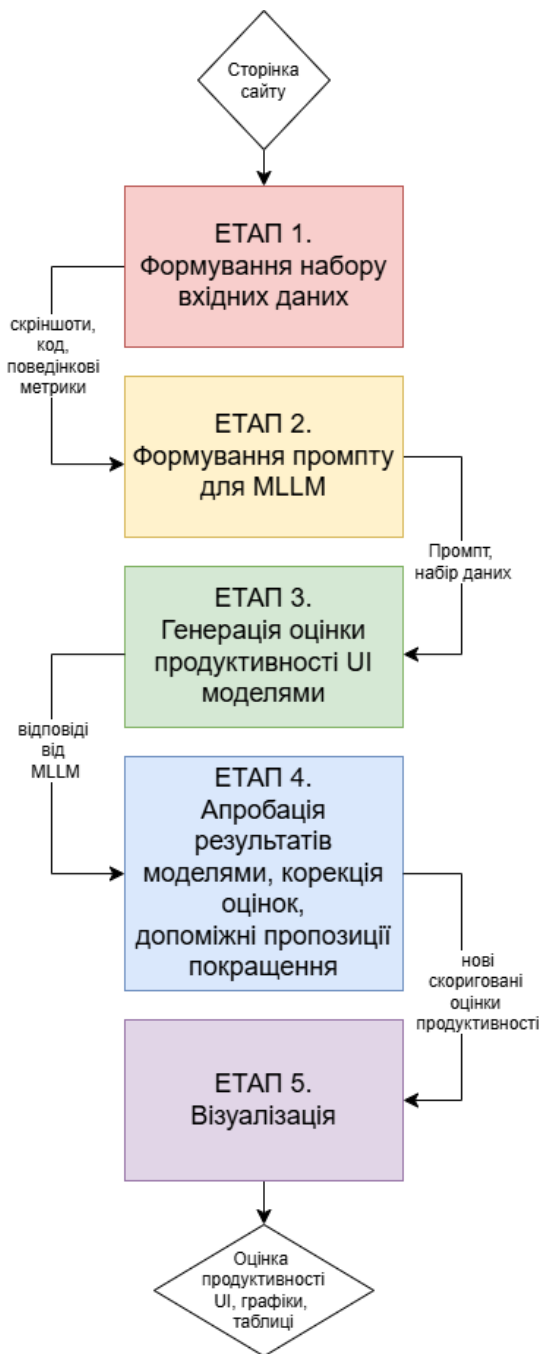


Рисунок 2.5 – Модифікований алгоритм автоматизованої оцінки продуктивності інтерфейсу

2.3.3 Опис функціонування та архітектури системи оцінки

Архітектура запропонованої системи передбачає послідовну взаємодію кількох функціональних компонентів, що забезпечують повний цикл обробки

даних – від введення вхідних матеріалів до формування інтегральної оцінки якості інтерфейсу.

На початковому етапі здійснюється завантаження тестових даних, які можуть бути представлені у вигляді зображень інтерфейсу (скріншотів), HTML чи CSS коду, даних про дії користувачів або текстових описів структури сторінки.

Завдяки цьому система підтримує як візуальні, так і структурні формати введення, що дозволяє моделювати різні сценарії оцінки користувацьких інтерфейсів.

Попередня обробка даних включає масштабування, кодування зображень у форматі Base64 для передачі до моделей, а також форматування HTML-файлів у текстовий вигляд.

Після отримання вхідних даних система автоматично формує контекстуалізований запит (промпт), який визначає роль штучного інтелекту, критерії оцінювання, формат відповіді й тип даних, що аналізується.

Сформований запит передається до вибраної мультимодальної великої мовної моделі (GPT-5.1 Thinking, або Gemini 2.0 Pro) через відповідний API-інтерфейс. Отримана відповідь містить кількісні оцінки за кожним критерієм, які надаються у форматі JSON (JavaScript Object Notation) – універсальної структури для зберігання даних у вигляді пари «ключ-значення».

Така форма подання дозволяє автоматизовано витягувати значення, необхідні для подальшої обробки, зокрема числові оцінки і текстові обґрунтування результатів.

Після отримання відповіді виконується обробка, агрегування та зберігання даних. Для кожного критерію обчислюються індивідуальні оцінки, після чого формується інтегральний показник продуктивності інтерфейсу I_{UI} згідно з формулою 2.1.

Отримані значення зберігаються у структурованому форматі та використовуються для порівняння між різними моделями штучного інтелекту.

Результати зберігаються в базі даних для зручності подальшого використання.

На завершальному етапі результати візуалізуються у вигляді діаграм, таблиць або текстових звітів, що відображають сильні й слабкі сторони проаналізованого інтерфейсу. Такий підхід забезпечує окрім кількісної, іще й якісну інтерпретацію результатів.

На рисунку 2.6 зображено послідовний потік даних з одного блоку застосунку в наступний.

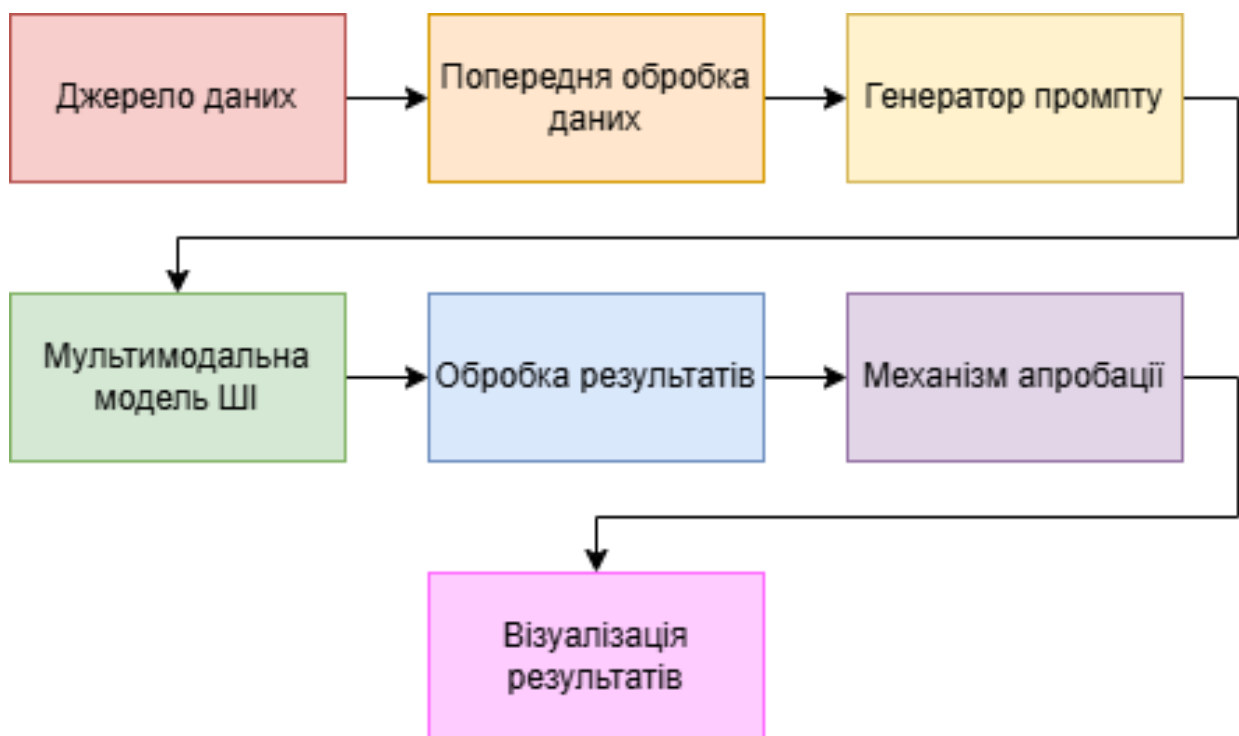


Рисунок 2.6 – Взаємодія блоків застосунку

У подальшому передбачено розширення системи шляхом реалізації механізму колаборативної оцінки, коли відповіді однієї моделі можуть бути передані іншій для формування критики та вдосконалення висновків.

Архітектура застосунку є гнучкою й розширюваною, що дозволяє інтегрувати нові моделі, типи даних або додаткові критерії оцінки без зміни основної логіки роботи системи. Створене середовище – це ефективний інструмент для апробації методики автоматизованої оцінки продуктивності

інтерфейсу користувача на основі сучасних мультимодальних моделей штучного інтелекту [35–37].

2.3.4 База даних

База даних розробленої системи забезпечує зберігання, структуровану організацію та подальший аналіз результатів оцінки інтерфейсів користувача за допомогою мультимодальних моделей штучного інтелекту. Її структура побудована відповідно до принципів реляційного моделювання даних і складається з трьох основних таблиць: *interfaces*, *models* та *evaluations*.

Таблиця *interfaces* призначена для зберігання інформації про інтерфейси або тестові приклади, що підлягають аналізу. Вона містить такі поля:

- «*id*», що є первинним ключем, який забезпечує унікальність кожного запису;
- «*name*», що є назвою інтерфейсу чи прикладу, який оцінюється;
- «*type*», що є типом інтерфейсу (наприклад, зображення, HTML-сторінка чи інший формат даних);
- «*path*», що є шляхом до файлу або посилання на його розташування в системі.

Таблиця *models* містить відомості про використані моделі штучного інтелекту. Вона включає такі поля:

- «*id*», що є унікальним ідентифікатором моделі;
- «*name*», що є назвою моделі (наприклад, GPT-5.1 Thinking, Gemini 2.0 тощо);
- «*api_endpoint*», що є адресою API або назвою інтеграції, через яку здійснюється взаємодія з відповідною моделлю.

Таблиця *evaluations* зберігає результати оцінювання, які отримані внаслідок роботи моделі з певним інтерфейсом. Її структура включає:

- «*id*», що є первинним ключем запису оцінки;

- «interface_id», що є зовнішнім ключем, що вказує на інтерфейс, який було оцінено;
- «model_id», що є зовнішнім ключем, що визначає модель, яка виконала оцінку;
- «score», що є числовим значенням оцінки якості або ефективності інтерфейсу;
- «response_text», що є текстовою відповіддю моделі, яка містить пояснення або опис оцінки.

На рисунку 2.7 зображено схему таблиць бази даних.

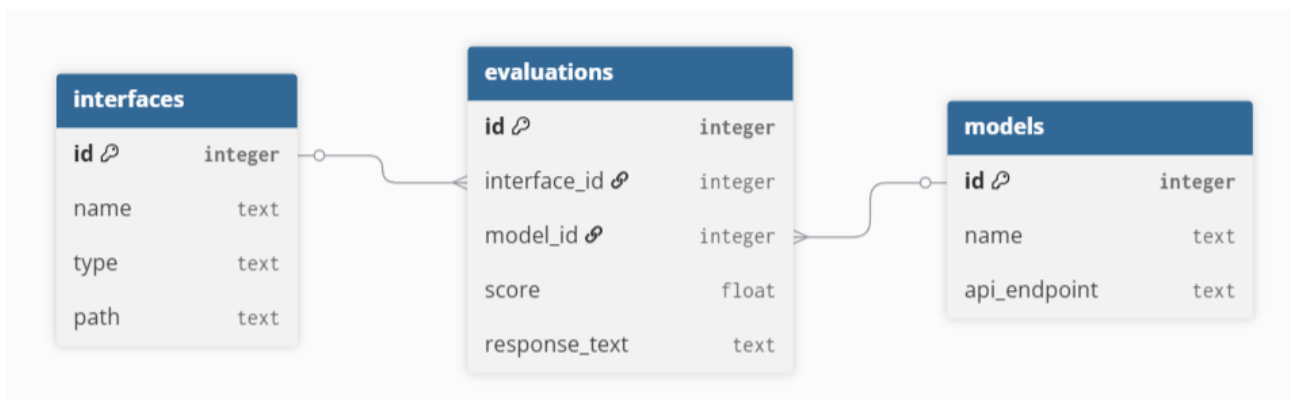


Рисунок 2.7 – Схема таблиць бази даних

Розроблена структура бази даних дає логічну цілісність і простоту масштабування системи. Зв'язки між таблицями реалізовані через зовнішні ключі.

3 ДОСЛІДЖЕННЯ МОДЕЛЕЙ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ОЦІНКИ ПРОДУКТИВНОСТІ ІНТЕРФЕЙСУ КОРИСТУВАЧА

3.1 Вибір інструментальних засобів для реалізації системи

Для реалізації системи оцінки продуктивності користувацького інтерфейсу з використанням мультимодальних моделей штучного інтелекту було обрано сучасні інструментальні засоби. Їх вибір зумовлений технічними вимогами системи і необхідністю створення середовища, придатного для експериментальних досліджень та подальшої розробки прототипу.

Основною платформою для реалізації програмної частини стало Google Colab – інтерактивне хмарне середовище, що підтримує виконання Python-коду в браузері. Його використання дозволяє здійснювати розробку, тестування і налагодження коду без потреби у локальній установці спеціалізованого програмного забезпечення.

Перевагою Colab є доступ до обчислювальних ресурсів Google Cloud, зокрема до графічних процесорів, що може бути корисним при обробці великих обсягів візуальних даних або при зверненні до моделей, які потребують значних обчислювальних потужностей. Крім того, Colab забезпечує легку інтеграцію з Google Drive, що спрощує зберігання проміжних результатів і організацію спільної роботи.

Мовою реалізації обрано Python, бо вона є домінуючим інструментом у сфері штучного інтелекту, машинного навчання і обробки даних. Його багата екосистема бібліотек надає широкі можливості для реалізації систем, що поєднують обчислення, роботу з базами даних, API-взаємодію та візуалізацію результатів.

У межах роботи використовуються такі бібліотеки Python, як requests (для виконання HTTP-запитів до API моделей), pandas (для структурування й аналізу даних), json (для роботи з результатами у форматі JSON), matplotlib та plotly (для побудови графіків і візуалізацій).

Крім того, інтеграція Python із СУБД SQLite забезпечує можливість зберігання результатів оцінювання у локальному форматі без потреби в зовнішньому сервері баз даних.

Для збереження результатів роботи системи використовується база даних SQLite, що є легкою, вбудованою системою управління базами даних, орієнтованою на використання у вбудованих або експериментальних застосунках. У контексті дослідження вона слугує для зберігання таких даних:

- отримані від моделей текстові відповіді;
- числові або категоріальні оцінки продуктивності інтерфейсу;
- метадані (ідентифікатори тестових даних, дата і час запиту, назва моделі, використаний промпт тощо).

Використання SQLite є виправданим через його простоту, автономність і повну інтеграцію з Python.

Зв'язок із мовними моделями здійснюється за допомогою платформи OpenRouter, бо вона забезпечує уніфікований доступ до кількох сучасних моделей штучного інтелекту через спільний API-інтерфейс. Використання OpenRouter дозволяє надсилати запити у форматі JSON та отримувати структуровані відповіді від різних моделей. Такий підхід спрощує процес інтеграції, уникає потреби в окремих підключеннях до кожного розробника моделей, дає стандартизований механізм взаємодії для побудови системи порівняльного аналізу результатів оцінки.

Формат обміну даними JSON використовується як основний, оскільки він універсальний, легкий для обробки, добре підтримується у Python. Саме в цьому форматі відбувається передача запитів, результатів і проміжних параметрів оцінки між компонентами системи.

Для подальшого аналізу й інтерпретації результатів у системі реалізовано візуалізаційний компонент, який забезпечує графічне подання даних. Бібліотеки matplotlib і plotly дозволяють створювати графіки розподілу оцінок, порівняння між моделями, часові залежності та інші типи діаграм, що полегшують виявлення закономірностей у поведінці моделей під час оцінювання інтерфейсів.

Візуалізація також слугує важливим інструментом перевірки якості результатів і полегшує процес інтерпретації.

Сукупність обраних засобів створює повноцінне експериментальне середовище, у межах якого можливо реалізувати всі етапи методики: від введення тестових даних (зображень або фрагментів коду інтерфейсу) до формування узагальнених кількісних і якісних показників продуктивності. Така архітектура дозволяє масштабувати систему в майбутньому – зокрема, інтегрувати додаткові моделі або типи даних, розширювати базу метрик і додавати автоматизовані сценарії обробки.

Сформований набір технологічних засобів створює основу для подальшої реалізації системи автоматизованої оцінки продуктивності інтерфейсу користувача на базі моделей штучного інтелекту.

3.2 Програмна реалізація способу оцінки інтерфейсу

3.2.1 Створення і додання тестових даних до бази даних

Для зберігання інформації про інтерфейси, моделі й результати їх оцінювання була створена база даних SQLite. Робота з базою реалізується за допомогою стандартного модуля `sqlite3`.

У лістингу 3.1 наведено код для створення БД.

Лістинг 3.1 Реалізація створення БД:

```
conn = sqlite3.connect('db.sqlite')  
cursor = conn.cursor()
```

Після підключення до файлу бази даних створюються три таблиці:

- «`interfaces`», яка зберігає дані про інтерфейси, що підлягають аналізу;
- «`models`», яка містить дані про використані мовні моделі;

– «evaluations», що призначена для фіксації результатів оцінювання кожного інтерфейсу кожною моделлю.

Структура таблиць задається за допомогою SQL-запитів, наведених в лістингу 3.2.

Лістинг 3.2 Код SQL-запитів для створення таблиць:

```
cursor.execute("""
CREATE TABLE IF NOT EXISTS interfaces (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    name TEXT NOT NULL,
    type TEXT,
    path TEXT)
""")

cursor.execute("""
CREATE TABLE IF NOT EXISTS models (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    name TEXT NOT NULL,
    api_endpoint TEXT)
""")

cursor.execute("""
CREATE TABLE IF NOT EXISTS evaluations (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    interface_id INTEGER NOT NULL,
    model_id INTEGER NOT NULL,
    score REAL,
    response_text TEXT,
    FOREIGN KEY (interface_id) REFERENCES interfaces (id),
    FOREIGN KEY (model_id) REFERENCES models (id))""")
```

У таблиці `interfaces` кожен запис містить:

- «`name`». Це назва інтерфейсу (наприклад, «HTML Layout Sample»);
- «`type`». Це тип інтерфейсу (`html`, `image`, `code` тощо);
- «`path`». Це шлях до файлу, який містить код або зображення інтерфейсу.

У таблиці `models` фіксується назва моделі та її `endpoint` – адреса для звернення через API.

Таблиця `evaluations` пов'язана з двома попередніми через зовнішні ключі `interface_id` і `model_id`, що забезпечує цілісність даних.

Після створення структури бази даних до неї додаються тестові записи – приклади інтерфейсу й моделі, які використовуються для початкового тестування роботи системи. Це дозволяє перевірити коректність усіх етапів взаємодії між компонентами. У лістингу 3.3 наведено дані для таблиць.

Лістинг 3.3 Дані для таблиць моделей і даних:

```
models_data = [
    ("GPT-5.1 Thinking", "https://openrouter.ai/models/openai/gpt-5.1-
thinking"),
    ("Gemini 2.0 Pro", "https://openrouter.ai/models/google/gemini-2.0-pro")]

interfaces_data = [
    # HTML приклади
    ("Landing Page Layout", "html", "samples/landing_page.html"),
    ("Dashboard UI", "html", "samples/dashboard_layout.html"),

    # Зображення інтерфейсів
    ("Mobile App Screen", "image", "samples/mobile_ui.png"),
    ("E-commerce Product Page", "image", "samples/product_ui.jpg"),
    # Текстові або структуровані файли
    ("UX Metrics JSON", "metrics", "samples/ux_metrics.json"),
    ("Form Validation Code", "code", "samples/form_validation.js")]
```

Тут «models_data» – це список кортежів із назвами та API-ендпоінтами моделей, які можуть бути використані для оцінки, «interfaces_data» – описує інтерфейси, для яких буде проводитися аналіз.

У лістингу 3.4 наведено вставлення даних у таблиці через метод `executemany()`, який дозволяє додати кілька записів за один виклик.

Лістинг 3.4 Вставка даних у таблиці:

```
cursor.executemany("""  
    INSERT INTO models (name, api_endpoint)  
    VALUES (?, ?)  
""", models_data)
```

```
cursor.executemany("""  
    INSERT INTO interfaces (name, type, path)  
    VALUES (?, ?, ?)  
""", interfaces_data)
```

```
conn.commit()
```

```
conn.close()
```

3.2.2 Взаємодія із моделями

Основним етапом функціонування системи є взаємодія з мовними моделями для здійснення автоматизованої оцінки користувацьких інтерфейсів. Цей процес реалізовано за допомогою асинхронних запитів до OpenRouter API, що забезпечує роботу з різними мовними моделями, зокрема із обраними в дослідженні та іншими підтримуваними сервісом.

Для виконання HTTP-запитів використовується бібліотека `openai` (клас `AsyncOpenAI`), яка підтримує асинхронну обробку, що дозволяє надсилати

кілька запитів одночасно. Такий підхід значно підвищує продуктивність системи у випадках, коли потрібно оцінити велику кількість інтерфейсів.

Ініціалізація клієнта для роботи з OpenRouter показана в лістингу 3.5. Змінна `OPENROUTER_API_KEY` містить ключ доступу до API, необхідний для автентифікації запитів. Об'єкт `client` використовується для надсилання повідомлень до моделей і отримання результатів їх обробки.

Лістинг 3.5 Ініціалізація клієнта для роботи з OpenRouter:

```
from openai import AsyncOpenAI
OPENROUTER_API_KEY = "code_placeholder"
client = AsyncOpenAI(
    base_url="https://openrouter.ai/api/v1",
    api_key=OPENROUTER_API_KEY,
)
```

Перш ніж відправляти запит до моделі, необхідно отримати вміст інтерфейсу, який буде аналізуватися. Для цього реалізовано допоміжну функцію `load_interface_content()`, що зчитує текстові або графічні файли залежно від типу інтерфейсу, зазначеного в базі даних. Ця функція забезпечує універсальність: незалежно від того, чи є інтерфейс HTML-кодом, скріншотом або набором метрик, система може передати відповідні дані моделі у зручному форматі. Функція наведена в лістингу 3.6.

Лістинг 3.6 Функція `load_interface_content()`:

```
def load_interface_content(interface_type: str, path: str):
    if not os.path.exists(path):
        return f"[Файл {path} не знайдено]"

    if interface_type.lower() in ["html", "text", "code", "metrics"]:
        with open(path, "r", encoding="utf-8") as f:
```

```

return f.read()

if interface_type.lower() in ["image", "png", "jpg", "jpeg"]:
    with open(path, "rb") as f:
        encoded = base64.b64encode(f.read()).decode("utf-8")
    return f"[Зображення в base64]\n{encoded[:500]}..."

```

Основна логіка взаємодії з моделлю реалізована у функції `evaluate_interface()`. Вона виконує кілька послідовних дій:

- зчитує дані про модель та інтерфейс із бази даних;
- формує текстовий запит (prompt) для аналізу;
- надсилає цей запит до моделі через API;
- отримує результат і зберігає його у базу даних.

Після надсилання запиту модель повертає текстову відповідь, яка містить як аналітичний опис, так і числову оцінку інтерфейсу.

Запит здійснюється асинхронно за допомогою методу `client.chat.completions.create()`.

Отриманий результат обробляється, і з нього виділяється числове значення оцінки. Дані записуються у таблицю `evaluations`. У такий спосіб кожен результат зберігається у базі даних із зазначенням, яка саме модель оцінювала який інтерфейс. Це дозволяє надалі здійснювати порівняння між моделями й аналізувати достовірність автоматичної оцінки.

Щоб усі інтерфейси могли оцінюватися різними моделями одночасно, реалізовано головну асинхронну функцію `main()`, яка формує список завдань і запускає їх паралельно.

Такий підхід дозволяє масштабувати процес оцінювання – у разі додавання нових інтерфейсів чи моделей система автоматично обробляє всі можливі комбінації.

3.2.3 Апробація відповідей моделей

Етап апробації реалізує механізм взаємної перевірки результатів, отриманих від мультимодальних моделей, з метою підвищення достовірності, узгодженості та аналітичної глибини оцінювання інтерфейсів.

На цьому етапі система здійснює крос-верифікацію – кожна модель аналізує відповіді інших моделей, формуючи метаоцінку, яка уточнює, доповнює або виправляє попередні висновки.

Основною є функція `get_model_feedback()`, яка здійснює вибірку результатів оцінювання з бази даних. Отримані результати передаються до функції `generate_meta_prompt()`, яка формує спеціалізований запит для іншої моделі.

У цьому запиті модель отримує текстову інструкцію на кшталт: «Ти – експерт із UX-дизайну. Оціни наведений звіт, виявивши можливі логічні неточності, непослідовність аргументації й надмірну суб'єктивність у висновках».

Після цього функція `cross_validate_responses()` відправляє сформований метапромпт до другої моделі, яка виконує аналіз результатів попередньої.

У процесі крос-верифікації модель проводить порівняння оцінок за ключовими параметрами (ефективність навігації, естетика, швидкість виконання завдань, когнітивне навантаження), а також виявляє протиріччя між висновками різних систем.

3.3 Проведення експериментів та дослідження ефективності моделей

3.3.1 Тестові дані

У межах експериментальної частини дослідження було обрано кодову базу «Landing Page Starter Template» з репозиторію на GitHub [38] як один із тестових

об'єктів користувацького інтерфейсу. Цей вибір пояснюється кількома важливими критеріями:

- шаблон має відкриту ліцензію (MIT Licence – Massachusetts Institute of Technology License), що дозволяє його використання у наукових цілях;
- він містить реальну HTML-версію інтерфейсу (файл index.html), а також супровідні стилі Tailwind CSS і графічний матеріал (hero.png), що забезпечує багатомодальність (текстова, структурна й візуальна інформація);
- шаблон є універсальним для аналізу UI, він включає навігаційну панель, основний контент, заклик до дій та ресурсну інформацію – тобто охоплює типові компоненти сучасного вебінтерфейсу.

Приклад, як виглядає обраний для дослідження інтерфейс, зображено на рисунках 3.1 – 3.4, а саме – його верхній рекламно-інформаційний блок сторінки, розділ із вмістом сторінки, розділ із цінами і нижня інформаційна секція сторінки.

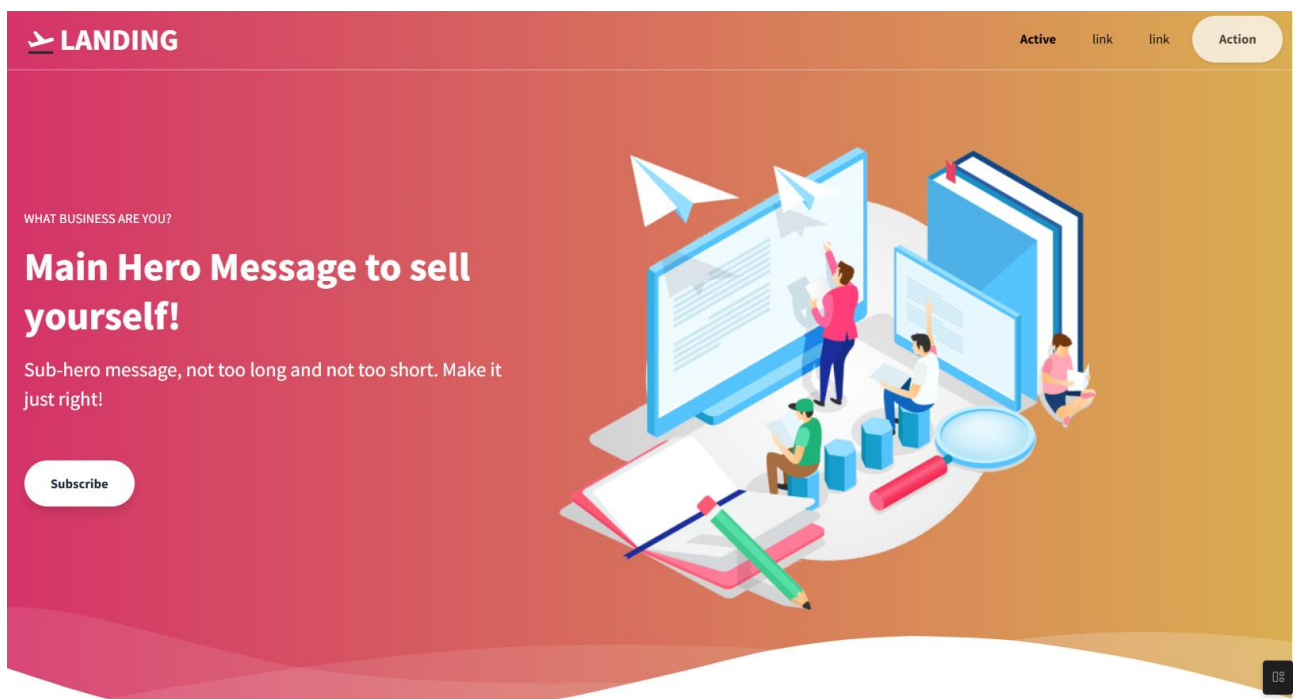


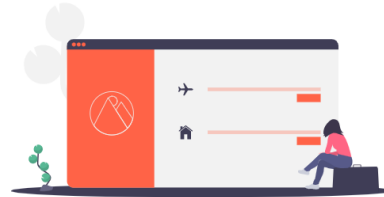
Рисунок 3.1 – Верхній рекламно-інформаційний блок сторінки

Title

Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam at ipsum eu nunc commodo posuere et sit amet ligula.

Images from: [undraw.co](#)



Lorem ipsum dolor sit amet

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam at ipsum eu nunc commodo posuere et sit amet ligula.

Images from: [undraw.co](#)



Рисунок 3.2 – Розділ із вмістом сторінки

Pricing

Free	Basic	Pro
Thing	Thing	Thing
Thing	Thing	Thing
Thing	Thing	Thing
£0 for one user	£x.99 / per user	£x.99 / per user
Sign Up	Sign Up	Sign Up



Рисунок 3.3 – Розділ із цінами

Код даного інтерфейсу надає можливість досліджувати такі аспекти продуктивності UI: розташування елементів, навігаційну послідовність, читабельність, зорову ієрархію, відповідність стандартам доступності, реактивність на різних типах пристроїв (за умови адаптації).

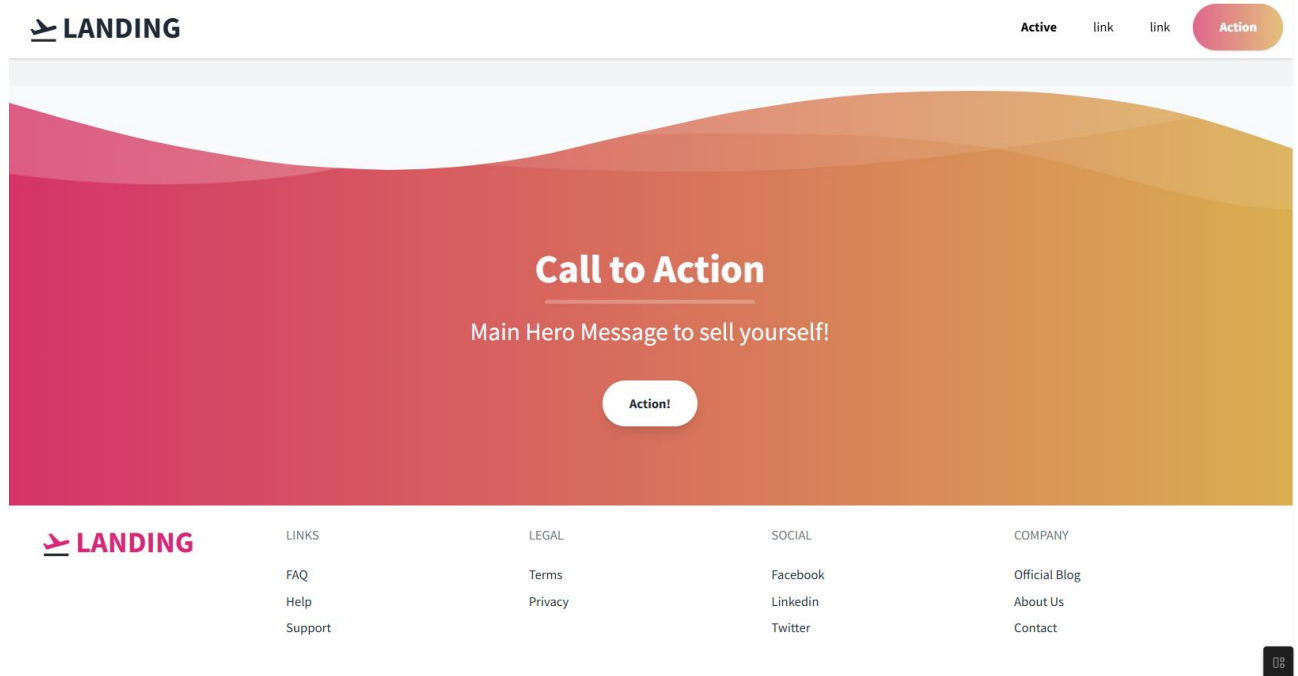


Рисунок 3.4 – Нижня інформаційна секція сторінки

Переваги дизайну та коду даного тестового інтерфейсу, які виявили експерти-люди:

Сильні сторони дизайну:

- чітка зорова ієрархія: основні елементи, такі як навігація, заголовки, СТА-кнопки, виділяються контрастними кольорами і розташовані у логічному порядку;
- адаптивність: сторінка коректно відображатися на різних пристроях та екранах різних розмірів (використовується Tailwind CSS);
- сучасна естетика: мінімалістичний стиль із достатнім простором між елементами робить інтерфейс легким для сприйняття й приємним для користувача;
- зрозуміла навігація, верхнє меню містить логічно згруповані розділи, що полегшує доступ до основного контенту й підвищує ефективність взаємодії;
- використання закликів до дії. Чітко позначені кнопки СТА спонукають користувача виконати потрібні дії без зайвого навантаження.

Сильні сторони коду:

- читабельність і структурованість: HTML-код розбитий на логічні блоки (header, main, section, footer) із коментарями й зрозумілими класами Tailwind CSS;

- модульність стилів: використання класів Tailwind забезпечує відокремлення стилів від контенту й дозволяє швидко модифікувати зовнішній вигляд без редагування HTML;

- лаконічність: код не містить зайвих тегів або дублювання стилів, що полегшує його обробку автоматизованими системами та скорочує час парсингу;

- відповідність сучасним стандартам, використання семантичних тегів і чистого HTML5 підвищує сумісність із браузерами й інструментами аналізу;

Наступний сайт [39] було обрано для перевірки здатності мультимодальних моделей оцінювати інтерфейси з низькою якістю дизайну. Його структура і візуальна організація надають можливість протестувати алгоритми на нестандартних та складних для аналізу UI.

Особливості дизайну:

- низька візуальна ієрархія: елементи сторінки часто змішані без чіткого пріоритету, що ускладнює користувачу швидку орієнтацію;

- застарілий стиль. Використання великої кількості тексту, неузгоджених кольорових рішень і відсутність сучасних візуальних патернів робить інтерфейс менш привабливим;

- складна навігація, меню й посилання розташовані нелогічно, деякі важливі функції заховані у підменю або в менш помітних областях;

- нестандартне компонування. Контент розташований без урахування просторової й логічної ієрархії, що створює когнітивне навантаження при використанні.

Особливості коду:

- неструктурований HTML, код сторінки містить багато застарілих тегів і відсутність чіткої семантичної структури, що ускладнює парсинг і автоматизований аналіз;

- складність стилізації, використання застарілих CSS-фреймворків і inline-стилів робить код менш читабельним та важким для обробки;
- обмежена адаптивність. Сайт погано підлаштовується під різні розміри екрану, що створює додаткові виклики для моделей, які аналізують візуальне подання інтерфейсу;
- наявність великої кількості зайвих елементів: банери, спливаючі вікна і надлишковий текст утруднюють визначення ключових зон уваги для оцінки UI.

Вибір цього сайту дозволяє протестувати, наскільки мультимодальні моделі здатні виявляти слабкі сторони інтерфейсу, формулювати обґрунтовані зауваження та давати об'єктивні оцінки навіть у складних і неідеальних умовах.

Вигляд інтерфейсу зображено на рисунку 3.5.



Рисунок 3.5 – Вигляд сторінки

3.3.2 Отримання оцінок

Після формування набору тестових даних і опису моделей, було проведено процес автоматизованого отримання оцінок від мультимодальних LLM.

Для кожної вибраної мультимодальної моделі було використано уніфікований prompt-шаблон з метою забезпечення відтворюваності та можливості порівняння результатів. Оцінка проводилася окремо для кожного тестового прикладу. Для кожного інтерфейсу передавалися такі дані:

- скріншот головної сторінки у форматі .png;
- фрагмент HTML-коду, що відповідає головному блоку сторінки;
- промпт, що містить завдання для моделі.

Промпт зазначено в додатку А. Кожна модель повинна виконати завдання:

- надати якісно-описову експертну оцінку UI/UX;
- сформулювати числові оцінки за п'ятибальною шкалою за кожним критерієм.

Після надання моделям промпту і даних щодо першого інтерфейсу, отримано оцінки інтерфейсу, наведені в таблиці 3.1.

Таблиця 3.1 – Оцінки першого інтерфейсу

Критерій	Модель	Коментар моделі	Оцінка	
			Модель	Експерт
1	2	3	4	5
Видимість стану системи	GPT-5.1 Thinking	Є чіткі заголовки, кнопки з діями та зрозуміла ієрархія тексту. Але не вистачає станів UI: (:hover, :active, aria, loading-індикаторів), валідацій і повідомлень про статус дії.	4	4
	Gemini 2.0 Pro	Зворотний зв'язок реалізовано через стандартні transition-ефекти фреймворка Tailwind. Це не забезпечує розширеного інформування, але підтверджує зміну стану при наведенні. Активний елемент навігації добре виділений.	4	

Продовження таблиці 3.1

1	2	3	4	5
Узгодженість і стандарти	GPT-5.1 Thinking	Система використовує узгоджену кольорову палітру, шрифтові стилі та однотипні блоки; дотримано композиційної гармонії. Tailwind забезпечує чітку стилістичну узгодженість, структура блоків логічна, всі стилі в одному форматі, використано каскад і однакові spacing-патерни.	5	5
	Gemini 2.0 Pro	Висока візуальна консистентність у кольорах, типографіці й використанні сітки. Усі ключові елементи (СТА-кнопки, заголовки секцій, розділювачі) використовують єдиний градієнт, один шрифт та ідентичні класи тіней/відступів. Це гарантує стилістичну єдність.	5	
Запобігання помилкам	GPT-5.1 Thinking	Інтерфейс простий та мінімальний, ризик помилок низький, однак кнопка «Action» у хедері виглядає контекстно невизначеною й може спричинити неправильний вибір. Код правильний, мінімальне використання кастомного CSS знижує ризик конфліктів. Проте немає поведінкових обмежень для кнопок, форм, немає інпутів із валідацією – відповідно, немає захисту від user-error.	4	4
	Gemini 2.0 Pro	Низький ризик виникнення помилок завдяки мінімалістичному дизайну й простій структурі. Відсутність чіткої візуальної ієрархії для неактивних посилань у хедері може спричинити когнітивне «заблукання». Юзабіліті-пастки відсутні. Потенційний ризик: приховане мобільне меню (Hamburger) може бути неочевидним для користувачів з низькою цифровою грамотністю, хоча це стандартний шаблон.	4,5	

Продовження таблиці 3.1

1	2	3	4	5
Естетика та мінімалізм	GPT-5.1 Thinking	Дизайн чистий, просторий, без інформаційного перевантаження; ілюстрації збалансовані та не відволікають від основних СТА. Код чистий, секції не перевантажені, візуальний ритм збережено, використані сучасні UI-практики.	5	4
	Gemini 2.0 Pro	Висока візуальна чистота завдяки щедрому використанню білого простору (bg-white), мінімалістичному дизайну. Гradient у Hero-секції додає «ваги», але загальне когнітивне навантаження залишається низьким.	4,5	
Розпізнавання замість згадування	GPT-5.1 Thinking	Структура лендингу стандартна та передбачувана, однак кнопки використовують універсальні назви без value proposition. Кнопки і СТА легко впізнавані, однак мікротекст кнопок типу «Sign Up» може виглядати універсально й не розкриває цінність дії.	4	4,5
	Gemini 2.0 Pro	Елементи керування (кнопки, посилання) стандартні та чітко розпізнаються. Інтуїтивність підтримується завдяки яскравому градієнту, тіням і заокругленим кутам, особливо в секції ціноутворення, де ключові СТА-кнопки виділені. Загальна інтуїтивність керування є високою, оскільки використовуються стандартні UI-патерни.	5	

Продовження таблиці 3.1

1	2	3	4	5
Доступність (WCAG)	GPT-5.1 Thinking	<p>Відсутні видимі фокус-стейти, контрастність градієнтів потенційно недостатня для WCAG рівня AA, а текст на фоні може бути проблемним для малозорих.</p> <p>Немає aria-атрибутів, контраст частин тексту не гарантовано перевірено, нема skip-links, focus outlines від Tailwind не кастомізовані, а деякі контейнери не мають semantic landmarking (main, nav, footer застосовані, але немає section role, aria-label, sr-only текстів).</p>	2	2,5
	Gemini 2.0 Pro	<p>Потенційні проблеми з доступністю WCAG AA: недостатній контраст тексту на градієнтних фонах і необхідність перевірки фокусу та семантичної структури (heading hierarchy) для скрінрідерів. Текст у Hero-секції білий на градієнті. Хоча він великий, контрастність кольору («#d53369» до «#daae51») з білим текстом може не пройти перевірку WCAG AA/AAA. Також відсутній `alt` текст для ілюстрацій.</p>	3	
Сучасність інтерфейсу	GPT-5.1 Thinking	<p>Модерновий flat- та semi-illustrative стиль відповідає сучасним UI-трендам; компоненти схожі на Material/Marketing landing best practices. Tailwind, utility-first, SVG-ілюстрації, адаптивна верстка – все відповідає сучасним стандартам 2023-2024.</p>	5	5
	Gemini 2.0 Pro	<p>Дизайн відповідає сучасним трендам (градієнти, мінімалізм, плоскі ілюстрації в стилі «undraw», мікро анімації). Дотримано основних UI-гайдів для Landing Page.</p>	5	

Продовження таблиці 3.1

1	2	3	4	5
Відповідність аудиторії та сценарію	GPT-5.1 Thinking	Верстка підходить для маркетингового лендингу, але не вистачає conversion-trigger-елементів: bullet benefits, trust elements, testimonials, дія після signup. Message-focus і диференціація ціннісних пропозицій можуть бути більш персоналізованими.	4	4
	Gemini 2.0 Pro	Інтерфейс ідеально відповідає сценарію Landing Page. Чіткі СТА і виділення тарифного плану «Basic» спрямовують цільову аудиторію до конверсії, виконуючи свою основну функцію.	5	

Коментар із пропозиціями покращення від GPT-5.1 Thinking: «Покращити контраст та фокусні стани згідно WCAG AA, уточнити формулювання СТА для підвищення намірів користувачів, додати системні повідомлення та hover/active стани, а також розширити контентну частину для створення унікальної ціннісної пропозиції».

Коментар із пропозиціями покращення від Gemini 2.0 Pro: «Рекомендації включають: білий текст на градієнті для забезпечення відповідності WCAG (можливо, використати темніший колір тексту або накласти темний оверлей на градієнт), уніфікацію стилю СТА-кнопок (наприклад, між Него-секцією й секцією Ціноутворення) для підвищення візуальної консистентності, додавання візуального ховер-ефекту (зворотного зв'язку) для посилань у хедері й футері. Варто додати alt-атрибути до всіх зображень і SVG-іконок для кращої доступності, переконатися, що у мобільному меню (коли воно розгорнуте) є чіткий засіб закриття та правильна послідовність фокусу (Tab Order)».

Аналогічно, моделям надається промпт і дані щодо другого інтерфейсу, і отримано його оцінки.

Оцінки наведені в таблиці 3.2.

Таблиця 3.2 – Оцінки другого інтерфейсу

Критерій	Модель	Коментар моделі	Оцінка	
			Модель	Експерт
1	2	3	4	5
Видимість стану системи	GPT-5.1 Thinking	<p>Відсутні підказки та системні повідомлення, пошук не має явного фідбеку після взаємодії; неясно, що очікується від користувача.</p> <p>У коді немає системних повідомлень, плейсхолдер у полі пошуку відсутній, відсутні ARIA-атрибути й механізми повідомлення про помилки або стани запиту.</p>	2	1,5
	Gemini 2.0 Pro	<p>Зворотний зв'язок та інформування користувача про системні стани (наприклад, завантаження, помилки, активний елемент) візуально не представлені або є мінімальними.</p> <p>Наочність стану елементів керування низька.</p> <p>Зворотний зв'язок (Feedback) відсутній у коді; кольори посилань задані застарілими глобальними атрибутами (link/vlink).</p> <p>Це не гарантує візуальної чіткості стану. Немає індикаторів успіху чи помилки.</p>	2	

Продовження таблиці 3.2

1	2	3	4	5
Узгодженість і стандарти	GPT-5.1 Thinking	Колірна гама надмірно насичена та хаотична, шрифти й іконографія різні за стилем, відсутня сітка й єдність візуальних патернів. Використовуються застарілі HTML-теги (, <center>), inline-стили і фонове зображення без адаптивності; немає консистентної UI-сітки, UI побудований на <table> замість сучасного layout-підходу.	1	1
	Gemini 2.0 Pro	Візуальна консистентність порушена: відсутня єдина типографічна ієрархія, шрифти та кольори тексту змінюються без очевидної логіки. Відступи і вирівнювання (сітка) елементів є хаотичними. Стиль хаотичний через використання інлайн-стилізації тегами і глобальними HTML-атрибутами. Відсутність CSS-файлу і використання таблиць для розмітки призводить до неконтрольованих та неконсистентних відступів і вирівнювання.	1	
Запобігання помилкам	GPT-5.1 Thinking	Нечітка навігація та неінтуїтивний пошук можуть спричинити помилки. Відсутність label для поля вводу, невизначений очікуваний формат введення і можливість некоректного пошуку збільшує ризик помилок; відсутня валідація та пояснення помилкових результатів.	2	2
	Gemini 2.0 Pro	Хоча юзабіліті-пастки відсутні, нерівномірне розташування елементів (наприклад, рядок пошуку і його кнопка) може призвести до помилок взаємодії або вимагати додаткового часу на візуальне сканування. Основні функції чіткі, але несемантичні зображення-іконки поруч із посиланнями знижують передбачуваність кліку, ускладнюють сканування інформації.	2	

Продовження таблиці 3.2

1	2	3	4	5
Естетика та мінімалізм	GPT-5.1 Thinking	Фон створює візуальний шум і знижує читабельність, елементи розміщені без чіткої ієрархії, підвищуючи когнітивне навантаження. Код підтверджує використання фону з зображенням та відсутність структурної ієрархії, що негативно впливає на візуальну чистоту, легкість сканування контенту.	1	1
	Gemini 2.0 Pro	Когнітивне навантаження екстремально високе через агресивний, текстурований фон, який створює сильний візуальний шум і конкурує з основним контентом. Це значно погіршує візуальну чистоту. Фон, заданий атрибутом background, конкурує з контентом. Використання для відступів ускладнює читання та призводить до візуального безладу.	1	
Розпізнавання замість згадування	GPT-5.1 Thinking	Пошукове поле й кнопки не мають явних підписів, іконки погано асоціюються з функціями, що ускладнює розпізнаваність. Елементи керування не марковані семантично (<label>, <button>), навігаційні посилання представлено як текст з іконкою, а значення елементів візуально, але не функціонально очевидні.	2	2
	Gemini 2.0 Pro	Інтуїтивність пошукового поля висока. Однак, розпізнаваність посилань, супроводжуваних іконками, є низькою, оскільки іконки візуально не вказують на функцію або категорію, а лише на наявність посилання. Хоча форма пошуку є стандартною, іконки-посилання мають низьку розпізнаваність (Affordance) та не відповідають очікуванням (Affinity) щодо медичних продуктів.	2,5	

Продовження таблиці 3.2

1	2	3	4	5
Доступність (WCAG)	GPT-5.1 Thinking	Низький контраст тексту і фону, ймовірна відсутність alt-текстів, немає клавіатурної навігації. В коді відсутні alt-описи для частини зображень, немає семантичної розмітки (header, nav, main, footer), контраст порушує WCAG, немає tabindex і доступних підказок для screen readers.	1	1
	Gemini 2.0 Pro	Критично низька доступність WCAG. Недостатній контраст кольору тексту (особливо жовтого та світло-зеленого) на складному, багатобарвному, текстурованому градієнтному фоні. Текст посилань link="#FFFF00" на текстурованому фоні не забезпечує необхідного контрасту. Відсутність семантичних тегів і alt-атрибутів для всіх зображень є порушенням.	1	
Сучасність інтерфейсу	GPT-5.1 Thinking	Інтерфейс застарілий, не відповідає сучасним UI-гайдам (Material, Human Interface, Fluent), відсутня адаптивність і responsive підхід. Сторінка побудована на HTML 4.0 Transitional і <table>-layout, що суперечить сучасним UI/UX стандартам і не відповідає mobile-first та responsive-design практикам.	1	1
	Gemini 2.0 Pro	Дизайн абсолютно не відповідає сучасним UI-гайдайдям, використовує HTML 4.0 та табличну розмітку, що повністю суперечить сучасним UI/UX-практикам, які вимагають використання CSS для стилізації і семантичної верстки. Стилїстика (текстурований фон, тіні, низька якість графіки) є сильно застарілою (legacy design) і не відповідає очікуванням кінцевих користувачів у 2020-х роках.	1	

Продовження таблиці 3.2

1	2	3	4	5
Відповідність аудиторії та сценарію	GPT-5.1 Thinking	Список продуктів потенційно релевантний цільовій аудиторії медичних та технічних фахівців, але спосіб подання знижує ефективність пошуку, застарілий і не підтримує швидке прийняття рішень, фільтрацію, сортування чи навігаційну структуру.	3	2
	Gemini 2.0 Pro	Інтерфейс виконує функцію каталогу, але його непрофесійний вигляд (застарілість та іконка «Skull») може негативно вплинути на довіру B2B-аудиторії, оскільки він не відповідає очікуванням медичної індустрії.	2	

Коментар із пропозиціями покращення від GPT-5.1 Thinking: «Оптимізувати контраст та кольорову палітру, застосувати сучасну сітку і візуальну ієрархію, покращити навігацію й affordances, додати aria-атрибути, семантичну розмітку й адаптивність, модернізувати пошуковий блок і впровадити відповіді системи (плейсхолдери, повідомлення та стейти дій). Перейти на сучасний HTML5 та semantic markup, впровадити адаптивний responsive дизайн, замінити <table>-layout на flex або grid, забезпечити контрастність і доступні елементи керування, додати формальні <label>, aria-атрибути і плейсхолдери, змінити колірну схему для читабельності, впровадити CMS або компонентний UI фреймворк для підтримки масштабованості».

Коментар із пропозиціями покращення від Gemini 2.0 Pro: «Рекомендації включають негайне видалення текстурованого фону і заміну його на чистий білий або нейтральний колір для підвищення контрасту (WCAG). Модернізація типографіки й уніфікація шрифтів для створення чіткої ієрархії. Реорганізація списку продуктів у структуровану сітку з чіткими іконками та достатнім відступом (whitespace) для зниження когнітивного навантаження. Необхідна повна реструктуризація інтерфейсу: видалення всіх стилів із HTML-коду

(bgcolor, font, center) та їхнє перенесення у зовнішній CSS-файл, заміна текстурованого фону на чистий, нейтральний колір для підвищення читабельності та контрасту (WCAG), забезпечення семантичної розмітки (використання тегів h1, h2, ul, p замість font/center) і додавання описових alt-атрибутів для всіх зображень».

3.4 Порівняльний аналіз результатів

Після отримання оцінок, отримано результати для обох інтерфейсів для кожного критерію. Середні оцінки за всіма критеріями обох інтерфейсів наведено в таблиці 3.3.

Таблиця 3.3 – Середні оцінки за всіма критеріями для обох інтерфейсів

Інтерфейс	GPT-5.1 Thinking	Gemini 2.0 Pro	Експерт
Інтерфейс 1	4,13	4,5	4,19
Інтерфейс 2	1,63	1.56	1,44

3.4.1 Перший інтерфейс

Як видно з таблиці 3.3, обидві моделі демонструють високу узгодженість з оцінками експерта, хоча Gemini 2.0 Pro дещо вищий за середнє значення за рахунок більш оптимістичних оцінок окремих критеріїв, наприклад, розпізнавання замість згадування та відповідності аудиторії та сценарію. GPT-5.1 Thinking у більшості випадків демонструє оцінки, що ближчі до експертних, з невеликим відхиленням у питанні доступності й мінімалістичного дизайну.

Щодо видимості стану системи, обидві моделі виставили оцінку 4, повністю узгоджену з оцінкою експерта. Коментарі моделей підкреслюють

чіткість заголовків та кнопок, але звертають увагу на відсутність повного набору станів UI та повідомлень про статус дії.

За узгодженістю і стандартами, було отримано максимальні оцінки (5) від GPT-5.1 Thinking та Gemini 2.0 Pro. Вони відповідають оцінці експерта, що демонструє високу здатність моделей розпізнавати стилістичну цілісність, узгодженість шрифтів, кольорів та відступів.

Щодо запобігання помилкам, то оцінки моделей дещо відрізняються: GPT-5.1 Thinking – 4, Gemini 2.0 Pro – 4,5. В обох випадках моделі правильно виявили потенційні ризики помилок, проте Gemini відзначила більш оптимістичний прогноз щодо користувацьких помилок.

Стосовно естетики та мінімалізму, GPT-5.1 Thinking оцінив дизайн вище (5) ніж експерт (4,5), тоді як Gemini 2.0 Pro поставила 4,5. Різниця пояснюється суб'єктивною оцінкою простору, пропорцій і когнітивного навантаження, яке моделі сприймають трохи інакше.

Щодо розпізнавання замість згадування, GPT-5.1 Thinking – 4, Gemini 2.0 Pro – 5, експерт – 4,5. Цей критерій показує, що Gemini надає трохи більш позитивну оцінку інтуїтивності елементів керування.

Доступність (WCAG) моделі було оцінено так: GPT-5.1 Thinking – 2, Gemini 2.0 Pro – 3, експерт – 2,5. Обидві моделі ідентифікували проблеми з контрастністю та відсутністю фокус-станів, проте Gemini оцінила доступність трохи вище, що свідчить про різне трактування стандартів WCAG.

Повна відповідність оцінок моделей та експерта (5) щодо сучасності інтерфейсу, що підтверджує здатність моделей коректно визначати трендові стилістичні рішення.

А відповідність аудиторії та сценарію GPT-5.1 Thinking оцінила на 4, Gemini 2.0 Pro – 5, експерт – 4. Модель Gemini оцінює відповідність більш оптимістично, особливо виділяючи ефективність СТА і структуру Landing Page.

На рисунку 3.6 зображено графік оцінок для першого інтерфейсу.

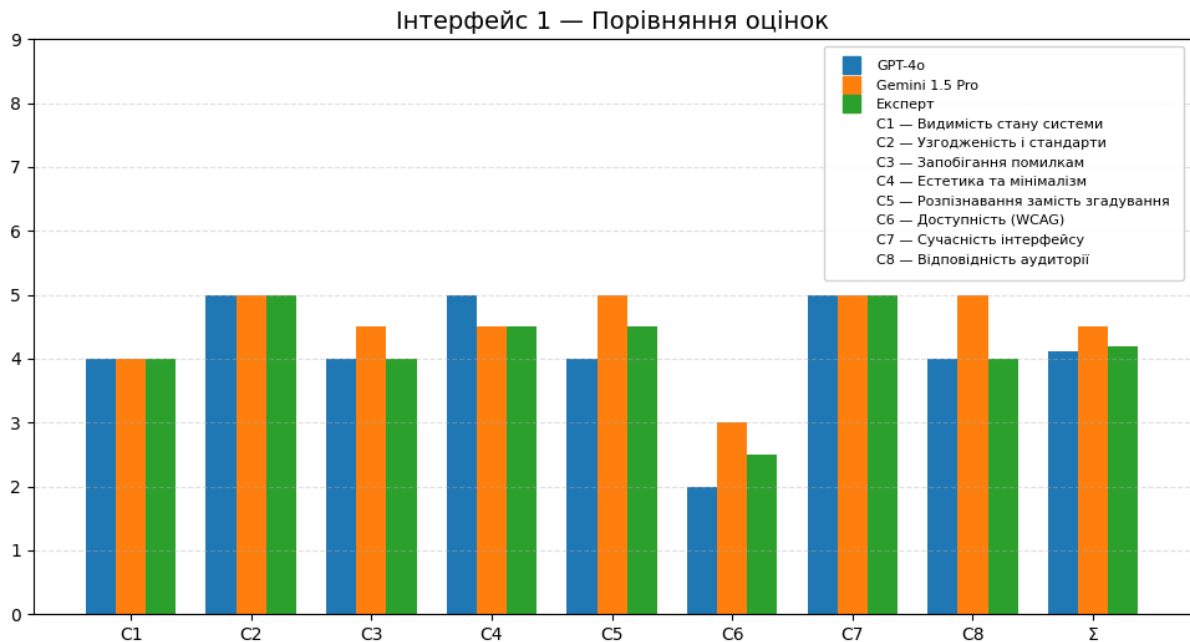


Рисунок 3.6 – Оцінки для першого інтерфейсу

3.4.2 Другий інтерфейс

Як можна побачити з таблиці 3.3, обидві моделі продемонстрували практично ідентичні оцінки, які відрізняються від оцінок експерта. Це свідчить про узгодженість висновків моделей при роботі з інтерфейсами низької якості, проте відображає певну оптимістичність моделей щодо деяких аспектів інтерфейсу (розпізнавання замість згадування у Gemini).

Стосовно видимості стану системи, GPT-5.1 Thinking і Gemini 2.0 Pro відзначають відсутність зворотного зв'язку, системних повідомлень та індикаторів стану, що ускладнює взаємодію користувача. Оцінки моделей (2) трохи вищі за експертну (1,5), оскільки моделі враховують мінімальну наочність через базові HTML-елементи.

В узгодженості й стандартах, обидві моделі та експерт оцінюють інтерфейс як хаотичний й застарілий (1). Моделі правильно визначили відсутність єдиної типографії, сітки і консистентності кольорів.

Моделі та експерт відзначають, що інтерфейс не захищає користувача від помилок: відсутні підказки, підписи для полів та валідація. Оцінки моделей повністю узгоджені з експертом (2).

GPT-5.1 Thinking і Gemini виставили мінімальні оцінки (1) у естетиці та мінімалізмі, підтверджуючи значний візуальний шум і відсутність структурної ієрархії, що ускладнює сприйняття контенту. Експерт теж виставив 1.

У розпізнаванні замість згадування GPT-5.1 Thinking виставила 2, Gemini 2.0 Pro – 2,5, експерт – 2. Моделі врахували наявність базових елементів, проте Gemini оцінила інтуїтивність трохи оптимістичніше через стандартну форму пошуку, що трохи покращує сприйняття елементів керування.

Щодо доступності (WCAG), моделі й експерт оцінюють доступність як критично низьку (1), зазначаючи проблеми з контрастністю тексту, відсутність alt-текстів та клавіатурної навігації.

У сучасності інтерфейсу оцінки моделей і експерта однакові (1). Визначено застарілі практики HTML 4.0, табличну верстку та відсутність адаптивного дизайну.

У відповідності аудиторії та сценарію GPT-5.1 Thinking поставила 3, Gemini – 2, експерт – 2. Моделі відзначають часткову релевантність контенту, але звертають увагу на слабку презентацію і застарілий стиль, що знижує ефективність користувацького сценарію.

На рисунку 3.7 зображено графік оцінок для першого інтерфейсу.

Моделі в цілому демонструють високу точність оцінки UX у порівнянні з експертними оцінками, відхилення по окремих критеріях зазвичай не перевищують 0,5 бала. Gemini 2.0 Pro проявляє тенденцію до більш оптимістичних оцінок у таких аспектах, як розпізнавання елементів керування та відповідність інтерфейсу аудиторії. GPT-5.1 Thinking демонструє більш консервативний підхід, ближчий до оцінок експерта. Аналіз коментарів моделей дозволяє виділити спільні напрямки для покращення інтерфейсів: підвищення контрастності, додавання видимих фокус-станів, уточнення формулювань СТА, забезпечення доступності через alt-атрибути для зображень і SVG-елементів.

Обидві моделі коректно визначають слабкі сторони інтерфейсів, включаючи хаотичний дизайн, низьку доступність і відсутність системних повідомлень.

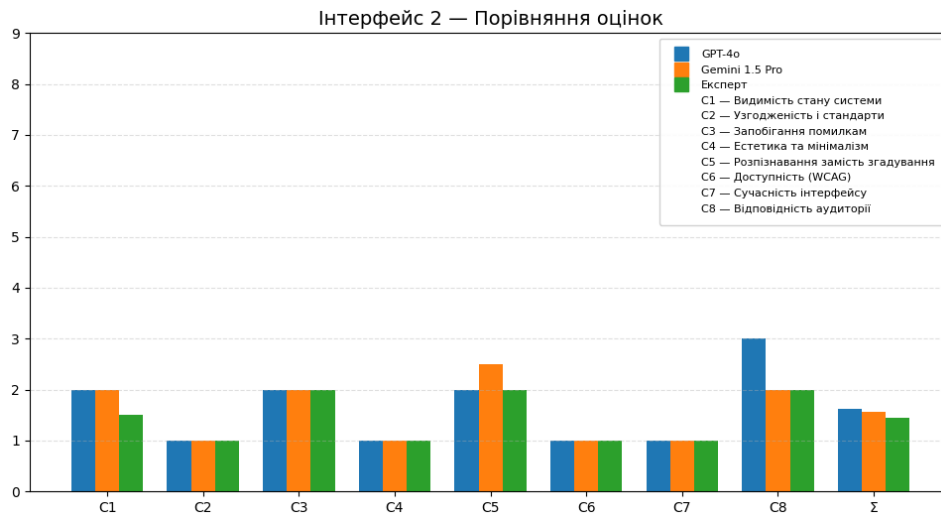


Рисунок 3.7 – Оцінки для другого інтерфейсу

Незважаючи на невеликі відмінності в оцінках, їхні висновки загалом узгоджуються з експертними, що підтверджує ефективність MLLM як інструменту автоматизованого UX-аудиту навіть при аналізі застарілих і низькоякісних інтерфейсів. Разом із тим, для того щоб моделі давали ще більш досконалі та точні оцінки, необхідно реалізувати етап 4 вдосконаленого методу отримання оцінки інтерфейсів від MLLM – апробацію, коли моделі критично аналізують одна одну й коригують власні оцінки на основі взаємної критики. Такий підхід дозволить підвищити узгодженість і точність оцінок.

3.5 Проведення етапу апробації моделями

На цьому етапі результати, отримані від різних MLLM, взаємно аналізуються та піддаються критичній перевірці. Моделі оцінюють відповіді одна одної, виявляють логічні, методологічні недоліки, уточнюють аргументацію і, за потреби, коригують свої первинні оцінки. Моделі GPT-5.1 Thinking надається відповідь Gemini 2.0 Pro і навпаки.

Критика відповіді від моделі Gemini 2.0 Pro:

«Перший інтерфейс: запобігання помилкам пропонується оцінити як 4 через неочевидність мобільного меню і відсутність валідації форм. Розпізнавання замість згадування знижене через універсальні тексти кнопок, оцінка – 4,5.

Відповідність аудиторії та сценарію в цілому ефективна, однак відсутність персоналізованих повідомлень і trust elements трохи знижує релевантність, тому пропонується оцінка 4,5.

Другий інтерфейс: критичною проблемою залишається розпізнавання елементів керування, оскільки іконки та посилання не забезпечують очікувану зрозумілість, тому оцінку пропонується скоригувати до 2. Інші критерії залишаються без змін, оскільки відповідають описаним недолікам і не потребують додаткової корекції.»

Критика відповіді від моделі GPT-5.1 Thinking:

«Перший інтерфейс: оцінку GPT-5.1 Thinking за критерієм Естетика та мінімалізм пропонується скоригувати до 4,5. Насичений градієнт Hero-секції трохи ускладнює візуальну легкість мінімалізму. Оцінку за Доступність скоригувати до 2,5.

Другий інтерфейс: За критерієм Відповідність аудиторії та сценарію пропонується скоригувати до 2. Непрофесійний дизайн й іконка «Skull» для медичних товарів знижують довіру B2B-аудиторії до критичного рівня.»

Загальні відкореговані оцінки наводяться в таблиці 3.4.

Таблиця 3.4 – Оцінки після апробації

Інтерфейс	Критерій	Відкореговані оцінки		
		GPT-5.1 Thinking	Gemini 2.0 Pro	Експерт
1	2	3	4	5

Продовження таблиці 3.4

1	2	3	4	5
Інтерфейс 1	Видимість стану системи	4	4	4
	Узгодженість і стандарти	5	5	5
	Запобігання помилкам	4	4	4
	Естетика та мінімалізм	4,5	4,5	4,5
	Розпізнавання замість згадування	4	4,5	4,5
	Доступність (WCAG)	2,5	3	2,5
	Сучасність інтерфейсу	5	5	5
	Відповідність аудиторії й сценарію	4	4,5	4
Інтерфейс 2	Видимість стану системи	2	2	1,5
	Узгодженість і стандарти	1	1	1
	Запобігання помилкам	2	2	2
	Естетика та мінімалізм	1	1	1
	Розпізнавання замість згадування	2	2	2
	Доступність (WCAG)	1	1	1
	Сучасність інтерфейсу	1	1	1
	Відповідність аудиторії й сценарію	2	2	2

У таблиці 3.5 наведено оновлені середні оцінки за всіма критеріями.

Таблиця 3.5 – Середні оцінки за всіма критеріями для обох інтерфейсів

Інтерфейс	GPT-5.1 Thinking	Gemini 2.0 Pro	Експерт
Інтерфейс 1	4,13	4,30	4,19
Інтерфейс 2	1,50	1,50	1,44

Середні оцінки після апробації демонструють, що результати моделей стали ближчими один до одного та до експертних оцінок: для першого

інтерфейсу показники GPT-5.1 Thinking та Gemini 2.0 Pro зросли та вирівнялися, для другого – підтвердили низьку якість інтерфейсу, зберігаючи узгодженість між моделями і експертом.

Проведення апробації виявилось корисним: воно підвищило точність автоматизованого UX-аудиту, дозволило виявити і скоригувати недоліки первинної оцінки моделей і підтвердило ефективність кроку критики й самокорекції для більш комплексного аналізу інтерфейсів.

3.6 Перспективи подальших досліджень

Отримані результати демонструють ефективність використання MLLM для автоматизованого UX-аудиту, проте відкриваються численні напрямки для подальшого дослідження.

Варто дослідити розширені методи апробації, коли моделі не лише критикують одна одну, але спільно генерують рекомендації для покращення дизайну, з врахуванням когнітивних особливостей користувачів.

Перспективним є застосування моделей до більш складних і динамічних інтерфейсів, включно з вебзастосунками й мобільними застосунками, інтерактивними панелями та системами із змінним контентом, де важлива реакція на дії користувача в реальному часі.

Також цікавим напрямком є інтеграція MLLM із інструментами автоматизованого тестування доступності й юзабіліті. Можна створити повноцінну систему підтримки UX-дизайну, здатну оцінювати і пропонувати оптимальні зміни інтерфейсу.

Нарешті, можливе дослідження комбінованого підходу, коли оцінки моделей поєднуються з аналізом поведінки реальних користувачів, що дозволить більш точно виявляти проблеми й прогнозувати ефективність запропонованих рішень.

ВИСНОВКИ

Таким чином, у кваліфікаційній роботі досліджено моделі штучного інтелекту, зокрема, мультимодальні великі мовні моделі, у задачі оцінки продуктивності користувацького інтерфейсу. Вирішено такі завдання:

- проведено аналіз сучасних підходів до оцінки продуктивності користувацьких інтерфейсів, визначено їхні основні обмеження, що базуються на суб'єктивних або ресурсозатратних методах, що підтвердило актуальність знаходження способів автоматизації;

- здійснено огляд літературних джерел, що стосуються використання моделей машинного навчання й комп'ютерного зору для візуального аналізу UI та прогнозування показників юзабіліті, що стало підґрунтям для вибору технологічної бази;

- сформовано та модифіковано систему критеріїв, що є придатною для автоматизованої оцінки продуктивності інтерфейсу на основі його візуального подання;

- обрано конкретні моделі штучного інтелекту (GPT-5.1 Thinking і Gemini 2.0 Pro), придатні для реалізації поставленої задачі, зокрема для виявлення складних елементів, оцінки естетичної якості, відповідності UI-правилам;

- розроблено покроковий алгоритм автоматизованої оцінки продуктивності інтерфейсу, що передбачає вхідні візуальні дані (скріншоти та верстку), їхню обробку обраними моделями ШІ, отримання кількісної оцінки за заданими критеріями і апробацію моделями їхніх відповідей;

- розроблено програмний прототип системи, що реалізує запропонований метод оцінки, забезпечуючи можливість введення тестових даних та отримання кількісної оцінки, що було продемонстровано на прикладі аналізу двох інтерфейсів;

- проведено експериментальне дослідження, порівнявши результати роботи розробленого підходу з експертними оцінками, що підтвердило

потенційну точність і ефективність використання MLLM у оцінці продуктивності інтерфейсів, і розробленого власного методу оцінки.

Наукова новизна роботи полягає у розробці низки критеріїв оцінки, котрі підходять для MLLM, і власного підходу до автоматичної оцінки продуктивності UI, який використовує інтеграцію можливостей MLLM для аналізу комплексних UX/UI-критеріїв.

Практична цінність дослідження полягає в тому, що розроблений підхід пропонує надійний і швидкий засіб для інтеграції у промислові інструменти UX-аналітики. Це може дати суттєво скоротити час оцінювання, підвищити об'єктивність результатів та забезпечити можливість масштабованого аналізу інтерфейсів, сприяючи підвищенню якості та функціональної ефективності цифрових продуктів.

Результати роботи апробовано у вигляді 2 тез доповідей під час IX Міжнародної науково-практичної конференції «Development of science: theories, methodology, practice and technologies» [40] та X Міжнародної науково-практичної конференції «Current issues, modern achievements and innovations of science» [41].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Grigera, J., Espada, J. P., & Rossi, G. (2023). AI in user interface design and evaluation. *IT professional*, 25(2), 20-22.
2. Nielsen, J., & Molich, R. (1990, March). Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 249-256).
3. Ivory, M. Y., & Hearst, M. A. (2001). The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys (CSUR)*, 33(4), 470-516.
4. Luera, R. A., Rossi, R., Dernoncourt, F., Basu, S., Kim, S., Mukherjee, S., ... & Kveton, B. (2025). Mllm as a ui judge: Benchmarking multimodal llms for predicting human perception of user interfaces. arXiv preprint arXiv:2510.08783. (дата звернення 28.10.2025)
5. GPT-4o vs. Gemini 1.5 Pro vs. Claude 3 Opus: Multimodal AI Model Comparison. URL: <https://encord.com/blog/gpt-4o-vs-gemini-vs-claude-3-opus>. (дата звернення 28.10.2025)
6. GPT-4 Turbo vs. Claude 3 Opus vs. Google Gemini 1.5 Pro – Best Generative AI LLMs Compared. URL: <https://www.kommunicate.io/blog/gpt4-vs-claude-3-vs-gemini> (дата звернення 28.10.2025)
7. Guerino, G., Rodrigues, L., Capeleti, B., Mello, R. F., Freire, A., & Zaina, L. (2025, September). Can GPT-4o Evaluate Usability Like Human Experts? A Comparative Study on Issue Identification in Heuristic Evaluation. In *IFIP Conference on Human-Computer Interaction* (pp. 381-402). Cham: Springer Nature Switzerland.
8. Soui, M., & Haddad, Z. (2023). Deep learning-based model using DensNet201 for mobile user interface evaluation. *International Journal of Human-Computer Interaction*, 39(9), 1981-1994.
9. Mao, Y. (2019). User interface evaluation with machine learning methods (Doctoral dissertation).
10. Biva, J. T., & Järvi, J. (2025). AI in Usability Testing.

11. Emirhan, A., & Jieun, H. (2024). AI-Based UX Assessment: The Role of GPT-4 Vision in UX/UI Comparison and Heuristic Evaluation. 한국 HCI 학회 학술대회, 272-276.
12. Wu, J., Peng, Y. H., Li, X. Y. A., Swearngin, A., Bigham, J. P., & Nichols, J. (2024, October). UIClip: a data-driven model for assessing user interface design. In Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology (pp. 1-16).
13. Baechler, G., Sunkara, S., Wang, M., Zubach, F., Mansoor, H., Etter, V., ... & Sharma, A. (2024). Screenai: A vision-language model for ui and infographics understanding. arXiv preprint arXiv:2402.04615. (дата звернення 20.10.2025).
14. Namoun, A., Alrehaili, A., Nisa, Z. U., Almoamari, H., & Tufail, A. (2024). Predicting the usability of mobile applications using AI tools: the rise of large user interface models, opportunities, and challenges. Procedia Computer Science, 238, 671-682.
15. Pathak, H., & Kapoor, R. (2025). Computer Vision for UI Testing: Leveraging Image Recognition and AI to Validate Elements and Layouts. Authorea Preprints. (дата звернення 22.10.2025).
16. Brdnik, S., Heričko, T., & Šumak, B. (2022). Intelligent user interfaces and their evaluation: a systematic mapping study. Sensors, 22(15), 5830.
17. Bohdan N., Tvoroshenko I., Gorokhovatskyi V., and Kobylin O. (2025) Development of a hybrid method to enhance context memory for a chatbot application based on large language models, International Journal of Academic Information Systems Research, 9(10), pp. 7-18.
18. Suprun A., Tvoroshenko I., Gorokhovatskyi V., and Yakovleva O. (2025) Development and research of a method for the combined use of large language models for text generation, International Journal of Academic and Applied Research, 9(10), pp. 249-263.
19. Gorokhovatskyi V., Chmutov Y., Tvoroshenko I., and Kobylin O. (2025) Reducing computational costs by compressing the structural description in image classification methods, Advanced Information Systems, vol. 9, no. 1, pp. 5-12.

20. Gorokhovatskyi V., Tvoroshenko I., Yakovleva O., Hudáková M., and Gorokhovatskyi O. (2024) Application a committee of Kohonen neural networks to training of image classifier based on description of descriptors set, *IEEE Access*, vol. 12, pp. 73376-73385.

21. Gorokhovatskyi V., Tvoroshenko I., Yakovleva O., and Hudáková M. (2025) Image description compression in classification structural methods, *IEEE Access*, vol. 13, pp. 43631-43641.

22. Yakovleva O., Matúšová S., Tvoroshenko I., and Isaiev Y. (2024) Visitor counting based on video stream analysis from surveillance cameras to solve various business problems, *Verejná správa a regionálny rozvoj ekonómia, manažment a marketing*, XX(1), pp. 67-87.

23. Gorokhovatskyi V., Tvoroshenko I., Yakovleva O. (2024) Transforming image descriptions as a set of descriptors to construct classification features, *Indonesian Journal of Electrical Engineering and Computer Science*, 33 (1), 113-125.

24. Lyashenko, V., Lyubchenko, V., Mohammad, A., Alveera, K., & Kobylín, O. (2016). The methodology of image processing in the study of the properties of fiber as a reinforcing agent in polymer compositions.

25. Yakovleva, O., Kovtunencko, A., Liubchenko, V., Honcharenko, V., & Kobylín, O. (2023, April). Face Detection for Video Surveillance-based Security System. In *COLINS* (3) (pp. 69-86).

26. Lyubchenko, V., Veretelnyk, K., Kots, P., & Lyashenko, V. (2024). Digital image segmentation procedure as an example of an NP-problem.

27. Pomazan, V., Tvoroshenko, I., and Gorokhovatskyi, V. (2023). Development of an application for recognizing emotions using convolutional neural networks, *International Journal of Academic Information Systems Research*, 7(7), pp. 25-36.

28. Кобилін, О.А., & Творошенко, І.С. (2021). Методи цифрової обробки зображень: навч. посібник. Харків: ХНУРЕ.

29. Daradkeh, Y. I., Gorokhovatskyi, V., Tvoroshenko, I., & Zeghid, M. (2022). Tools for fast metric data search in structural methods for image classification, *IEEE Access*, vol. 10, pp. 124738-124746.

30. Gorokhovatskyi, V., & Tvoroshenko, I. (2024). An effective method for transforming an image description into a compact vector for classification.

31. Gorokhovatskyi V., Tvoroshenko I. (2023) Identification of visual objects by the search request. Int. scientific symp. «Intelligent Solutions-S». Computational intelligence. Decision making theory: proceedings of the international symposium, September 28, 2023, Kyiv-Uzhorod, Ukraine, 25-27.

32. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Zeghid M. (2024) Improving the effectiveness of image classification structural methods by compressing the description according to the information content criterion, *Computers, Materials & Continua*, vol. 80, no. 2, pp. 3085-3106.

33. Гороховатський В.О., Творошенко І.С., Чмутов Ю.В. (2022) Застосування систем ортогональних функцій для формування простору ознак у методах класифікації зображень, *Сучасні інформаційні системи*, 6(3), С. 5-12.

34. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Al-Dhaifallah M. (2022) Classification of Images Based on a System of Hierarchical Features, *Computers, Materials & Continua*, 72(1), pp. 1785-1797.

35. Yanholenko, O., Grinchenko, M., Rohovyi, M., Yakovleva, O., & Rogovyi, A. (2025). The model and method of intelligent planning of IT project team work. PhD Workshop on Artificial Intelligence in Computer Science at 9th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2025). *CEUR Workshop Proceedings Vol. 3403*. pp. 134-149. <https://doi.org/10.31110/COLINS/2025-3/010>

36. Yakovleva, O., Matúšová, S., & Táncošová, J. (2024, December 16-18). Investigation of LLMs for generating answers based on user-provided content to support educational and organizational processes. Abstracts of XVI International Scientific and Practical Conference «Modern and new technical trends that help humanity». Thessaloniki, Greece, Pp. 289-295. URL: <https://eu-conf.com/events/modern-and-new-technical-trends-that-help-humanity/>

37. Yakovleva, O., Nebeský, L., & Kirichenko, A. (2023). Using the GPT models for responses based on custom content to develop neural consultant for

university applicants. Abstracts of V International Scientific and Practical Conference «The world of modern technologies and inventions» Madrid, Spain. Pp. 172-178. URL: <https://eu-conf.com/ua/events/trends-in-science-regarding-the-creation-of-new-teaching-methods/>

38. Tailwind Toolbox – Landing Page Template. URL: <https://github.com/tailwindtoolbox/Landing-Page/tree/master> (дата звернення 12.11.2025)

39. Pacific Northwest X-Ray Inc. – Search for X-Ray Products. URL: <https://www2.pnwx.com> (дата звернення 12.11.2025)

40. Кульмінський Я. К., Любченко В. А. (2025) Застосування мультимодальних моделей ШІ для автоматизованої оцінки інтерфейсів користувача. IX Міжнародна науково-практична конференція «Development of science: theories, methodology, practice and technologies» (28–31 жовтня 2025 р., Париж, Франція), с. 76–80.

41. Kulminskyi Y. (2025) Modified method for automated evaluation of user interface performance. X International Scientific and Practical Conference «Current issues, modern achievements and innovations of science» (November 4–7 2025, Bilbao, Spain), pp.37-43.