

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Система для автоматичної оцінки рівня знань
студентів із використанням нейронних мереж»

на здобуття освітнього ступеня бакалавра
зі спеціальності 122 Комп'ютерні науки
(код, найменування спеціальності)
освітньо-професійної програми Штучний інтелект
(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання
на відповідне джерело*

(підпис) Ірина ЖУРАВЕЛЬ
Ім'я, ПРІЗВИЩЕ здобувача

Виконав: здобувачка вищої освіти гр.ШІД-42
Ірина ЖУРАВЕЛЬ

Керівник: Ольга ЗІНЧЕНКО
д.т.н, професор

Рецензент: _____

Київ 2025

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут інформаційних технологій

Кафедра Штучного інтелекту

Ступінь вищої освіти Бакалавр

Спеціальність 122 Комп'ютерні науки

Освітньо-професійна програма Штучний інтелект

ЗАТВЕРДЖУЮ

Завідувач кафедри Штучного інтелекту

_____ Ольга ЗІНЧЕНКО

«_____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Журавель Ірина Валентинівна
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Розробка системи для автоматичної оцінки рівня знань студентів із використанням нейронних мереж

керівник кваліфікаційної роботи Ольга ЗІНЧЕНКО д.т.н., професор,
(Ім'я, ПРИЗВИЩЕ науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «25» лютого 2025 р. № 56

2. Строк подання кваліфікаційної роботи «23» травня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література

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

1. Огляд сучасних технологій для розробки

5. Перелік графічного матеріалу: презентація

6. Дата видачі завдання «25» лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	25.02-05.11.25	Виконано
2	Вивчення літератури та додаткових матеріалів для створення додатку	05.11-12.11.25	Виконано
3	Дослідження баз даних та існуючих систем оцінювання	13.11-19.11.25	Виконано
4	Створення бази даних, структури додатку та реєстрації користувачів	20.11-25.11.25	Виконано
5	Інтеграція моделі Mistral-7B для автоматичної перевірки відповідей	27.11-03.12.25	Виконано
6	Перевірка функціональності додатку та усунення помилок	04.12-10.12.25	Виконано
7	Оформлення роботи: вступ, висновки, реферат	11.12-20.12.25	Виконано
8	Розробка демонстраційних матеріалів	21.12-23.05.25	Виконано

Здобувачка вищої освіти

(підпис)

Ірина ЖУРАВЕЛЬ

(Ім'я, ПРІЗВИЩЕ)

Керівник

кваліфікаційної роботи

(підпис)

Ольга ЗІНЧЕНКО

(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавр: 64 стор., 20 рис., 1 таблиця, 22 джерел.

Мета роботи: Розробка вебзастосунку для автоматичної оцінки рівня знань студентів із використанням нейронних мереж і сучасних мовних моделей.

Об'єкт дослідження: Процес перевірки знань студентів на основі відповідей на тестові завдання з використанням технологій штучного інтелекту.

Предмет дослідження: Інтелектуальна система оцінювання, що поєднує вебінтерфейс, базу даних, моделі нейронних мереж та алгоритми обробки природної мови (NLP) для аналізу відповідей студентів.

Короткий зміст роботи: У роботі розроблено та реалізовано вебсистему на основі Flask-фреймворку для автоматичного оцінювання знань. Система дозволяє створювати тести, збирати відповіді студентів та автоматично аналізувати їх. Архітектура реалізована з використанням Python, SQLAlchemy та зовнішніх мовних моделей (LLM) для оцінки відповідей. Проведено тестування з використанням реальних відповідей студентів, проаналізовано точність та ефективність системи. У проєкті також реалізовано особистий кабінет, збереження результатів і сучасний вебінтерфейс.

КЛЮЧОВІ СЛОВА: АВТОМАТИЧНЕ ОЦІНЮВАННЯ, НЕЙРОННІ МЕРЕЖІ, ШТУЧНИЙ ІНТЕЛЕКТ, NLP, FLASK, PYTHON, SQLALCHEMY, ГЕНЕРАЦІЯ ТЕСТІВ, ОСВІТНІ ТЕХНОЛОГІЇ.

ЗМІСТ

ВСТУП	9
1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ У СФЕРІ ОСВІТНІХ ТЕХНОЛОГІЙ	11
1.1 Огляд систем автоматичного оцінювання знань студентів	11
1.2 Порівняльна характеристика інтелектуальних освітніх платформ (Khan Academy, Coursera, Moodle, Gradescope)	15
1.3 Переваги та недоліки використання штучного інтелекту в освіті	23
Висновки до розділу 1	27
2 ПРОЄКТУВАННЯ СИСТЕМИ АВТОМАТИЧНОГО ОЦІНЮВАННЯ ЗНАНЬ	29
2.1 Архітектура запропонованої системи оцінювання	29
2.2 Вибір моделі нейронної мережі для аналізу відповідей студентів	33
2.3 Алгоритм оцінювання: генерація, інтерпретація та виставлення балів	38
Висновок до розділу 2	43
3 РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОЇ СИСТЕМИ	45
3.1 Технологічний стек, середовище розробки та компоненти системи	45
3.2 Реалізація модуля тестування та оцінювання відповідей	52
3.3 Аналіз результатів: точність, об'єктивність і порівняння з традиційними методами оцінювання	59
Висновок до розділу 3	62
ВИСНОВКИ	64
ПЕРЕЛІК ПОСИЛАНЬ	66
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (ПРЕЗЕНТАЦІЯ)	68

ВСТУП

У сучасних умовах цифровізації освіти особливого значення набувають інтелектуальні системи, здатні автоматизувати ключові процеси навчання, зокрема — оцінювання знань студентів. Зростаюча кількість онлайн-курсів, дистанційного навчання та змішаних форматів створює нові виклики для викладачів і навчальних закладів, які потребують об'єктивних, масштабованих та ефективних інструментів контролю знань. Традиційні методи тестування, що базуються на ручному аналізі відповідей, часто є суб'єктивними, трудомісткими та неспроможними забезпечити оперативний зворотний зв'язок. З огляду на стрімкий розвиток штучного інтелекту, зокрема нейронних мереж та методів обробки природної мови (NLP), з'являється можливість створення систем, здатних не лише генерувати навчальні матеріали, а й аналізувати письмові відповіді студентів з урахуванням змісту, логіки та повноти. Такі підходи відкривають перспективи впровадження автоматизованого оцінювання, яке мінімізує людський фактор, забезпечує однакові критерії для всіх студентів та дозволяє масштабувати освітній процес.

Актуальність теми обумовлена необхідністю створення сучасної системи, що поєднує інтелектуальні технології з освітніми практиками, з метою підвищення ефективності навчального процесу та оптимізації роботи викладачів. Автоматизація оцінювання дозволяє значно зекономити час, зменшити ймовірність помилок і забезпечити об'єктивний аналіз знань.

Метою даної кваліфікаційної роботи є розробка системи для автоматичної оцінки рівня знань студентів із використанням нейронних мереж та технологій обробки природної мови.

Об'єктом дослідження є процес оцінювання знань студентів у контексті цифрового освітнього середовища.

Предметом дослідження виступає інтелектуальна система, що реалізує аналіз текстових відповідей за допомогою моделей машинного навчання та методів NLP.

Для досягнення поставленої мети були визначені такі завдання:

- проаналізувати сучасні рішення у сфері автоматичного оцінювання зі
- дослідити методи NLP та архітектури нейронних мереж, що застосовуються для аналізу тексту;
- спроектувати архітектуру системи автоматичного оцінювання;
- реалізувати веб-застосунок на основі Flask із можливістю створення тестів, збору відповідей та виставлення балів;
- інтегрувати мовну модель (LLM) для інтелектуального аналізу відповідей студентів;
- провести тестування системи й оцінити її ефективність за метриками точності, узгодженості та об'єктивності оцінювання.

Методи дослідження: аналіз наукових публікацій та існуючих рішень, побудова архітектури клієнт-серверної системи, використання бібліотек Flask, SQLAlchemy, TensorFlow/Keras та інтеграція мовних моделей через API або локальні інтерфейси.

Наукова новизна полягає у створенні адаптивної системи оцінювання, яка поєднує вебінтерфейс, базу даних та мовні моделі, здатні аналізувати відповіді студентів з використанням сучасних методів ШІ.

Практичне значення розробки полягає в можливості впровадження системи в навчальні заклади для об'єктивного, масштабованого та швидкого оцінювання знань, що підвищує ефективність освітнього процесу та зменшує навантаження на викладачів.

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ У СФЕРІ ОСВІТНІХ ТЕХНОЛОГІЙ

1.1 Огляд систем автоматичного оцінювання знань студентів

Автоматичне оцінювання знань є одним із ключових напрямів у розвитку сучасних освітніх технологій, оскільки воно дозволяє не лише прискорити процес перевірки результатів навчання, а й забезпечити високу об'єктивність, незалежність від людського фактору та однакові умови для всіх студентів. Завдяки автоматизації освітні заклади отримують змогу ефективно масштабувати процес контролю знань на великі потоки студентів, що є критично важливим у період розширення дистанційного та змішаного навчання. З розвитком штучного інтелекту, зокрема технологій обробки природної мови та нейронних мереж, освітні системи набувають нових властивостей: вони стають здатними не просто перевіряти правильність відповідей, а й аналізувати зміст, логічну побудову думки, граматичну правильність, аргументованість та загальний рівень розуміння студентом навчального матеріалу. Такі підходи дозволяють максимально наблизити якість автоматичної перевірки до рівня людського експертного аналізу. Автоматичне оцінювання у поєднанні з інтелектуальними технологіями формує нову парадигму навчального процесу, де кожен студент отримує індивідуальний підхід, оперативний зворотний зв'язок та чітке розуміння своїх сильних і слабких сторін. У контексті сучасної цифрової освіти це є запорукою підвищення якості навчання, мотивації студентів і загального рівня освітньої аналітики.

Сучасні рішення у сфері автоматичного оцінювання представлені широким спектром платформ, починаючи від універсальних освітніх середовищ до спеціалізованих інструментів для аналізу письмових відповідей і кодів. Найбільш розповсюдженими є Moodle, Edmodo, Coursera, Khan Academy, Gradescope, Google Classroom, а також низка кастомних рішень, які розробляються безпосередньо для потреб конкретних навчальних закладів. Ці системи охоплюють різні типи навчального контенту, підтримують інтеграцію з базами даних студентів, дозволяють адаптувати типи завдань під конкретну дисципліну, а деякі —

застосовують алгоритми штучного інтелекту для класифікації та аналізу відкритих відповідей. Особливої актуальності набувають рішення, здатні працювати з великими обсягами даних у реальному часі, враховувати індивідуальні траєкторії навчання та формувати персоналізований зворотний зв'язок для кожного студента.

Moodle є однією з найпопулярніших систем у сфері дистанційного навчання, яка має відкритий код і широко використовується в університетах і школах по всьому світу. Вона надає широкі можливості для створення навчальних курсів, інтеграції мультимедійного контенту та організації тестування. Серед основних інструментів Moodle — генерація тестів із вибором відповіді, питаннями на відповідність, відкритими запитаннями, можливістю прив'язки до таймерів, обмеження спроб, автоматичного виставлення оцінок, а також гнучкого налаштування аналітики результатів. Система підтримує плагіни для розширення функціональності, у тому числі й ті, що працюють із модулями штучного інтелекту для оцінки відкритих відповідей.

Gradescope, розроблений у рамках Стенфордського університету, орієнтований насамперед на вищу освіту. Його головною особливістю є можливість ефективної перевірки письмових і навіть рукописних відповідей, які студенти завантажують у вигляді сканів. Система автоматично групує схожі відповіді за шаблонами, використовуючи елементи машинного навчання. Це дозволяє викладачеві не перевіряти кожен роботу окремо, а лише шаблони відповідей, що значно пришвидшує виставлення оцінок та зменшує ймовірність помилок. Gradescope також має функції для оцінювання програмного коду, підтримку аналітики по групах, таймінгів виконання і системи фідбеку для студентів.

Khan Academy — безкоштовна освітня платформа, яка пропонує адаптивне навчання на основі відеоуроків, інтерактивних вправ і тестів. Система використовує автоматичне оцінювання на кожному етапі навчання, надаючи студентам зворотний зв'язок у режимі реального часу. Вона формує індивідуальну траєкторію навчання, базуючись на прогресі, історії помилок і рівні складності

завдань, які опрацьовує студент. Алгоритми адаптації дозволяють не лише тестувати знання, а й пропонувати новий матеріал у зручній послідовності. Платформа підтримує аналітику успішності та інтеграцію з системами управління класами.

Coursera — глобальна платформа онлайн-освіти, яка співпрацює з провідними університетами та компаніями для створення онлайн-курсів. Її система оцінювання включає не лише автоматичні тести, а й перевірку проєктів, рефератів і відкритих відповідей через peer-review. Coursera активно використовує модулі машинного аналізу для базових типів тестів, однак більшість глибоких перевірок все ще потребують участі викладача або іншого користувача. Завдяки інтеграції з хмарними сервісами та мобільними додатками, платформа забезпечує гнучкий доступ до тестування, включно з таймерами, повторними спробами та динамічним формуванням питань залежно від попередніх результатів.

Google Classroom — це безкоштовна освітня платформа від компанії Google, яка дозволяє викладачам створювати курси, завдання та тести, інтегруючи їх із сервісами Google Docs, Google Sheets, Google Meet та Google Forms. У межах автоматичного оцінювання найбільш активно використовується Google Forms — інструмент для створення тестів із можливістю вказати правильні відповіді, автоматично виставляти оцінки та надавати пояснення. Платформа підтримує синхронізацію з календарем, аналітику успішності, а також можливість обміну навчальними матеріалами. Особливо зручною Google Classroom стала у період пандемії COVID-19, коли вона слугувала основою для дистанційного навчання у багатьох країнах. Її простота у використанні, інтеграція з іншими сервісами Google та можливість спільної роботи над документами роблять її ефективним рішенням для щоденного використання.

Попри високу функціональність та широке впровадження у навчальний процес, більшість існуючих систем автоматичного оцінювання залишаються обмеженими у своєму функціоналі. Основний підхід до оцінювання полягає у перевірці завдань із закритою структурою: тестів з вибором правильної відповіді,

відповідності, або коротких текстових відповідей. Такі підходи дозволяють швидко виставити оцінки, що особливо важливо при роботі з великими масивами студентських даних, однак вони не враховують глибину розуміння навчального матеріалу, здатність до критичного аналізу, узагальнення, формулювання власної позиції та її аргументації. Відсутність можливості адекватно оцінити ці складові робить традиційні системи автоматичного оцінювання менш релевантними у контексті гуманітарних, соціальних та аналітичних дисциплін. Особливо проблемним залишається аналіз відкритих текстових відповідей, есе, розгорнутих міркувань або письмових пояснень. У більшості випадків ці завдання оцінюються вручну викладачами, що потребує значних часових ресурсів, призводить до перевантаження педагогічного складу та, що не менш важливо, містить елемент суб'єктивності. Частина платформ намагається автоматизувати цей процес шляхом використання шаблонних алгоритмів — пошуку ключових слів або словоформ, однак ці інструменти не здатні враховувати глибинний контекст, синтаксичну структуру, семантику речень та логічні переходи між тезами, що суттєво обмежує їх аналітичні можливості.

Додатковою проблемою є брак персоналізації в оцінюванні: більшість систем не враховують індивідуальні особливості студента, його прогрес у навчанні, типові помилки або стиль мислення. Це унеможливорює формування гнучкої навчальної траєкторії на основі результатів тестування. У свою чергу, впровадження підходів на основі штучного інтелекту, нейромереж та обробки природної мови (NLP) відкриває нові перспективи для вирішення зазначених проблем. Зокрема, трансформерні архітектури на кшталт BERT, GPT та їх модифікації дозволяють аналізувати повноцінні текстові відповіді з урахуванням смислового наповнення, стилістики, логіки викладення та навіть емоційної тональності.

Застосування таких моделей дозволяє створювати більш гнучкі, адаптивні та інтелектуальні системи автоматичного оцінювання, здатні імітувати експертний підхід викладача з високим рівнем точності. Це, у свою чергу, сприяє формуванню

ефективного, об'єктивного та персоналізованого освітнього процесу, що відповідає викликам сучасної цифрової епохи. У цьому контексті дедалі більшої актуальності набувають рішення, що базуються на технологіях обробки природної мови (NLP) та нейронних мережах, зокрема трансформерних архітектурах (наприклад, BERT, GPT). Такі моделі здатні аналізувати повноцінні текстові відповіді, враховувати зміст, логіку міркувань, стилістичну послідовність і навіть тональність. Застосування подібних підходів дозволяє створювати більш гнучкі й інтелектуальні системи оцінювання, що наближують якість автоматичного аналізу до рівня експертної людської перевірки. Отже, впровадження таких технологій є наступним логічним кроком у розвитку освітніх платформ, орієнтованих на якісну перевірку знань у широкому розумінні цього поняття..

1.2 Порівняльна характеристика інтелектуальних освітніх платформ (Khan Academy, Coursera, Moodle, Gradescope)

Інтелектуальні освітні платформи є рушієм глибоких трансформацій у сучасному освітньому середовищі, адже вони дозволяють не лише оптимізувати навчальний процес, а й значно підвищити його ефективність, гнучкість та персоналізацію. Актуальність їх використання зумовлена необхідністю швидкого реагування на виклики масової дистанційної освіти, потребою у якісному оцінюванні знань великої кількості студентів та інтеграції цифрових технологій у щоденну педагогічну практику. Такі платформи поєднують у собі інструменти для створення навчального контенту, адміністрування курсів, комунікації між учасниками освітнього процесу, а також автоматичного оцінювання та збору статистики щодо результатів навчання. Особливо важливою є інтеграція алгоритмів штучного інтелекту, які дозволяють адаптувати матеріали під потреби кожного студента, аналізувати його помилки, формувати персоналізовані траєкторії навчання та надавати зворотний зв'язок у реальному часі. Інтелектуальні освітні платформи можуть охоплювати як базові функції систем управління навчанням (LMS), так і спеціалізовані модулі для глибокого аналізу письмових відповідей, автоматичного розпізнавання тексту, перевірки коду або

формування індивідуального навантаження. Вони стають інструментом не лише для викладача, а й для адміністрацій закладів освіти, дозволяючи приймати обґрунтовані рішення на основі освітньої аналітики.

У цьому підрозділі здійснено детальний порівняльний аналіз (табл.1) найбільш популярних платформ: Khan Academy, Coursera, Moodle та Gradescope з точки зору їх технічних характеристик, можливостей адаптації, інтеграції AI-компонентів та ефективності в освітньому середовищі.

Khan Academy — безкоштовна освітня онлайн-платформа (рис. 1.1), яка надає доступ до великого обсягу навчального контенту, включаючи відеоуроки, інтерактивні вправи, міні-тести, практичні завдання й математичні симуляції. Вона розрахована переважно на шкільну аудиторію, однак також використовується студентами та дорослими слухачами для самостійного вивчення матеріалу. Основною перевагою платформи є її адаптивна система навчання, яка динамічно реагує на відповіді студента та формує індивідуальну траєкторію на основі рівня засвоєння тем.

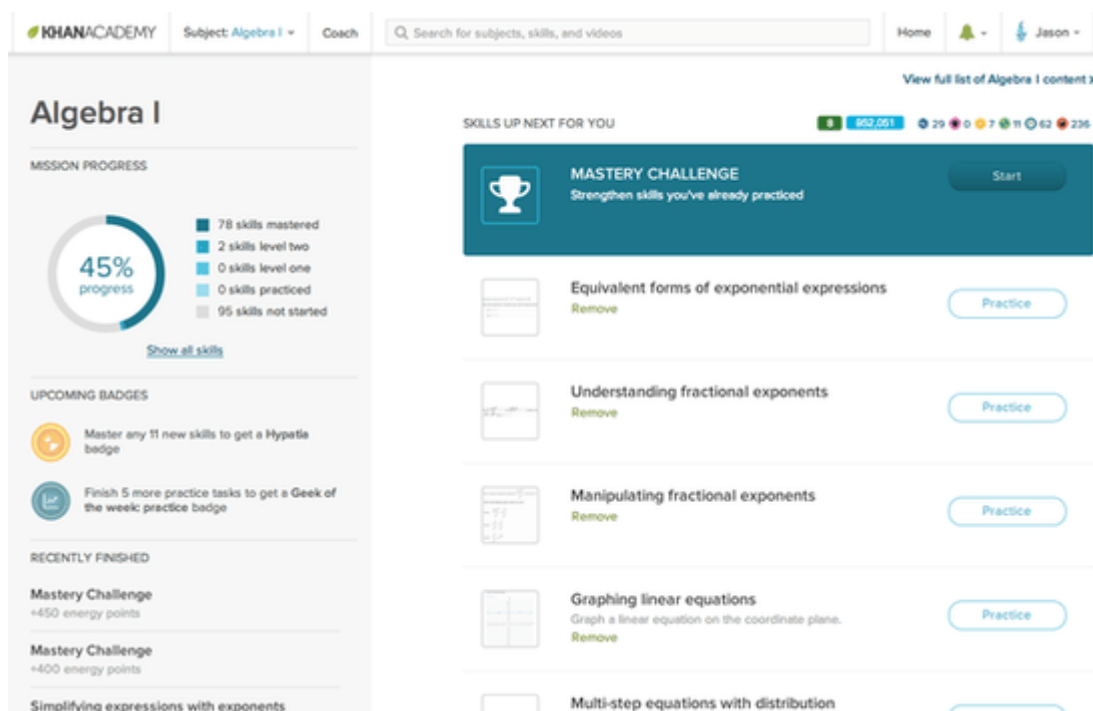


Рисунок 1.1 - Khan Academy — безкоштовна освітня онлайн-платформа

Система автоматично відстежує прогрес, надає миттєвий фідбек, підказки та візуалізує результати у вигляді діаграм і шкал, що сприяє кращому розумінню та мотивації користувача. Крім цього, Khan Academy пропонує панель аналітики для викладачів, яка дозволяє бачити рівень активності кожного студента, виявляти прогалини в знаннях та давати персоналізовані рекомендації. Платформа підтримує гейміфікаційні елементи (значки, рівні, нагороди), що робить навчання більш захоплюючим. Попри відсутність складних AI-модулів, система все ж реалізує базову адаптивність і аналітику, що відповідає ключовим вимогам до сучасних інтелектуальних платформ для самонавчання. Вона також підтримує інтеграцію з Google Classroom, що дозволяє включати матеріали Khan Academy до навчальних курсів у шкільних та університетських середовищах.

Coursera — комерційна освітня платформа (рис. 1.2), яка співпрацює з провідними університетами, інститутами та компаніями по всьому світу (Stanford, Yale, Google, IBM тощо). Вона надає доступ до тисяч сертифікованих онлайн-курсів, спеціалізацій, професійних програм та навіть повноцінних ступеневих програм (бакалаврських і магістерських). Coursera підтримує багаторівневу систему оцінювання знань, яка включає автоматичні тести з миттєвим фідбеком, есе з перевіркою методом реєр-review, а також проєктні роботи з подальшим захистом. Особливістю платформи є наявність адаптивного контенту, який змінюється залежно від прогресу студента.

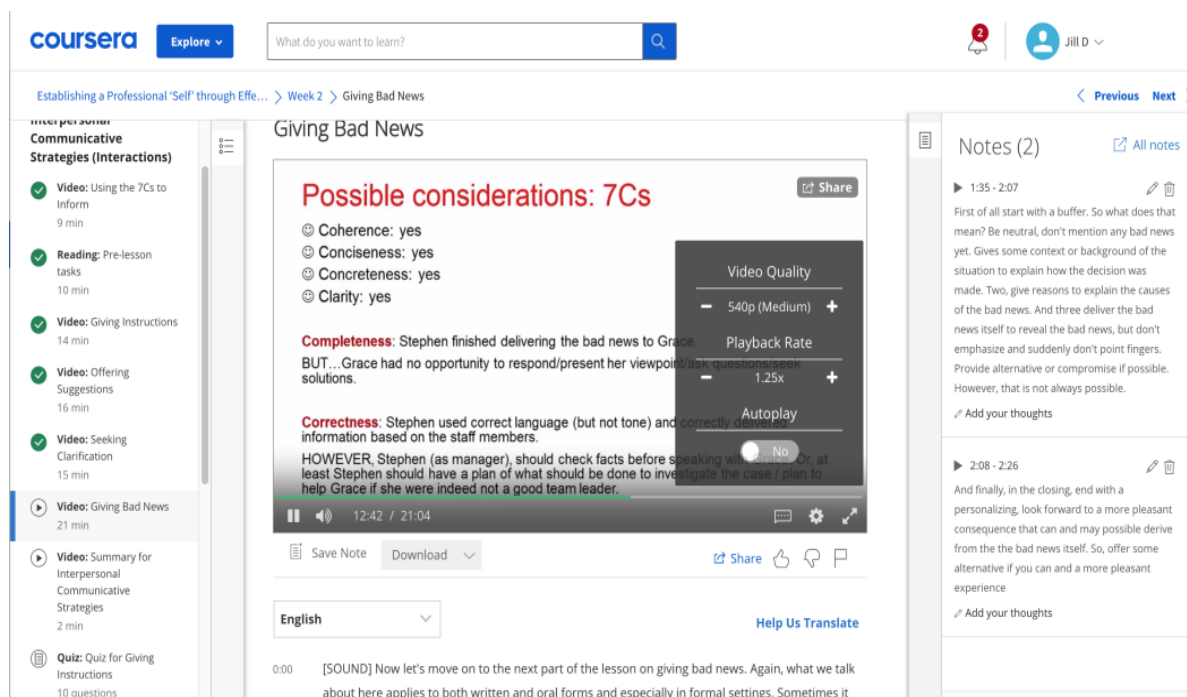


Рисунок 1.2 - Coursera — комерційна освітня платформа

Платформа активно використовує алгоритми штучного інтелекту для аналізу прогресу, рекомендацій курсів, виявлення слабких місць у знаннях, а також для формування індивідуальної освітньої траєкторії. Coursera підтримує інтеграцію з мобільними пристроями, має API для корпоративного впровадження, аналітичні панелі для викладачів та адміністраторів, а також можливості сертифікації, валідації навичок і зв'язку з ринком праці (Coursera for Business). Платформа надає гнучкі можливості для самонавчання та формального здобуття освіти, що робить її універсальним інструментом у сучасному освітньому середовищі.

Moodle — це одна з найпоширеніших у світі систем управління навчанням (LMS) (рис. 1.3) з відкритим програмним кодом, яка активно використовується в університетах, школах та корпоративному навчанні. Вона дозволяє створювати, адмініструвати, наповнювати та контролювати навчальні курси в електронному середовищі з урахуванням індивідуальних особливостей конкретного закладу освіти. Moodle має модульну архітектуру, що дає змогу підключати додаткові компоненти (плагіни) для розширення функціоналу, включаючи модулі для

тестування, збору статистики, інтерактивних форумів, відеоконференцій та синхронізації з іншими сервісами.

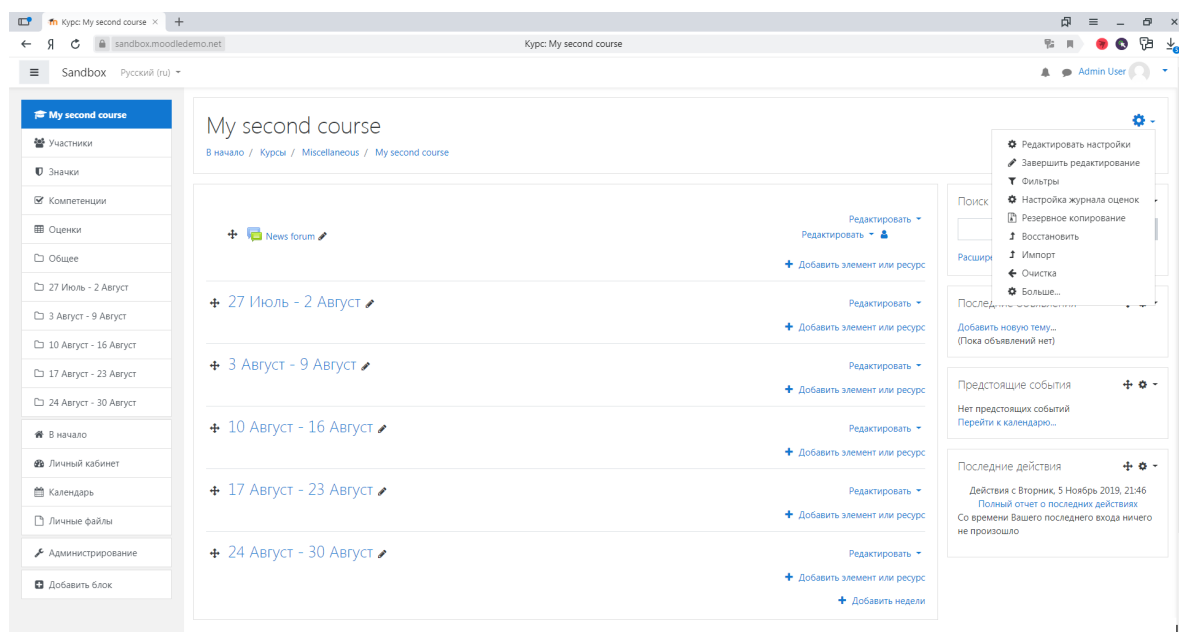


Рисунок 1.3 - Moodle — система управління навчанням (LMS)

Серед основних переваг Moodle — повна кастомізація дизайну та логіки курсу, підтримка SCORM та xAPI-модулів, налаштування прав доступу на будь-якому рівні користувачів (адміністратор, викладач, студент, асистент), а також підтримка інтернаціоналізації з перекладом інтерфейсу на десятки мов. Moodle підтримує різні типи оцінювання: від стандартних тестів з автоматичною перевіркою до індивідуальних письмових робіт з виставленням оцінки вручну, використанням рубрик або шкал. Завдяки API Moodle може інтегруватися з системами електронного журналу, аналітичними платформами, LDAP та SSO-автентифікацією. Окрему увагу приділено аналітичним інструментам: викладачі можуть аналізувати результати навчання за темами, бачити прогрес студентів, відстежувати активність і навіть прогнозувати ймовірність неуспішного складання курсу. Moodle також дозволяє організовувати адаптивне навчання — система може змінювати сценарій проходження курсу в залежності від відповідей студента. У сукупності ці функції роблять Moodle потужним рішенням для

підтримки повноцінного навчального циклу з можливістю глибокого аналізу освітніх даних та інтелектуального управління освітнім процесом.

Gradescope — спеціалізована платформа, яка орієнтована на автоматизацію процесу оцінювання (рис. 1.4) письмових робіт, тестів, кодів і графічних відповідей, і є особливо ефективною в галузі технічних, математичних, інженерних і природничих наук. Основною особливістю Gradescope є підтримка оцінювання як сканованих паперових робіт, так і цифрових подань, з можливістю гнучкого формування шаблонів перевірки.

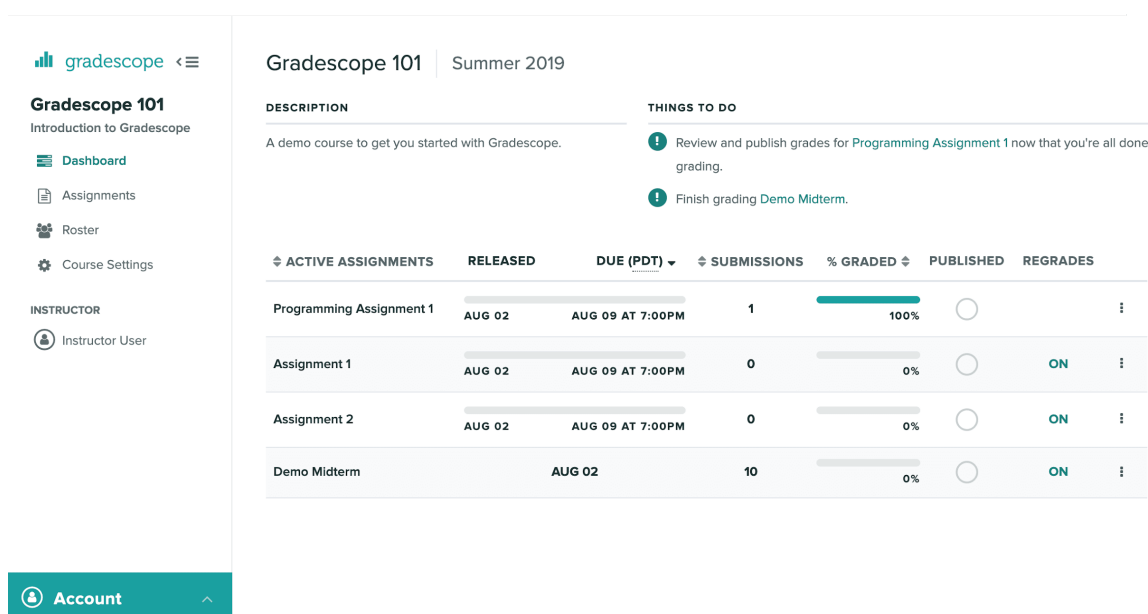


Рисунок 1.4 - Gradescope — спеціалізована платформа, яка орієнтована на автоматизацію процесу оцінювання

Платформа дозволяє викладачам завантажувати відповіді студентів у вигляді PDF або зображень, після чого система автоматично ідентифікує зони з відповідями, групує подібні варіанти та пропонує єдину схему оцінювання для схожих відповідей. Це суттєво скорочує час перевірки великих потоків студентів. Gradescope також підтримує багаторівневі рубрики, можливість виставлення балів за частину завдання, коментування окремих фрагментів відповіді та створення індивідуальних звітів про помилки та прогрес студентів. Інтеграція з AI-алгоритмами дозволяє платформі навчатися на основі прикладів оцінених відповідей, що дає змогу автоматизувати навіть відкриті запитання. Крім того,

Gradescope має функціонал для перевірки програмного коду, з можливістю запуску тестів і аналізу результатів. Вона також підтримує інтеграцію з популярними LMS, такими як Moodle, Canvas, Blackboard, та забезпечує безперервну синхронізацію результатів оцінювання з електронними журналами. Gradescope стає незамінним інструментом у випадках, коли необхідно обробити велику кількість складних письмових робіт, зберігаючи при цьому високу точність, прозорість оцінювання та можливість надання зворотного зв'язку кожному студентові.

Таблиця 1 - Порівняльна таблиця демонструє ключові відмінності між платформами

Платформа	Тип доступу	Адаптивність	Автоматичне оцінювання	Підтримка AI/NLP	Типи контенту	Особливості
Khan Academy	Безкоштовна	Так	Так	Обмежена	Відео, вправи, тести	Візуалізація, гейміфікація
Coursera	Платна/Безкошт.	Частково	Так + peer-review	Частково	Курси, проекти	Сертифікація, API інтеграція
Moodle	Безкоштовна	Обмежено	Так	Через плагіни	Курси, тести, SCORM	Повна кастомізація
Gradescope	Умовно безкошт.	Ні	Так (письмові/код)	Частково	Письмові роботи	Групування, класифікація

У результаті проведеного порівняльного аналізу можна зробити висновок, що кожна з розглянутих платформ має унікальні особливості та переваги, які забезпечують її ефективне застосування в певному освітньому контексті.

Moodle є оптимальним рішенням для закладів вищої освіти завдяки можливості глибокої кастомізації, підтримці великої кількості користувачів, модульній архітектурі та інтеграції з внутрішніми інформаційними системами навчального закладу. Її сильні сторони — масштабованість, надійність і гнучкість.

Khan Academy — ідеальний вибір для самостійного навчання, а також для використання у шкільній практиці, де важливими є адаптивність, простота візуального представлення матеріалу та гейміфікаційні елементи, що сприяють залученню молодшої аудиторії до навчального процесу.

Coursera надає унікальну можливість здобути формальну сертифіковану освіту від провідних університетів світу, поєднуючи академічну якість з технологічною інфраструктурою платформи. Вона є однією з небагатьох систем, що інтегрує гнучкі траєкторії навчання, ринкову орієнтацію та валідацію навичок.

Gradescope значно підвищує ефективність перевірки письмових робіт, особливо в технічних дисциплінах, де необхідна об'єктивність, точність і масштабованість при обробці великої кількості ручних відповідей. Платформа поєднує аналітичні інструменти з алгоритмами машинного навчання, що забезпечує автоматизацію оцінювання без втрати якості.

Вибір платформи залежить від конкретних освітніх цілей, типу контенту (текстовий, візуальний, інтерактивний), рівня навчання (шкільна, вища або професійна освіта), а також технічних вимог до гнучкості, масштабованості, інтеграції та аналітичних можливостей. Важливу роль відіграє також підтримка штучного інтелекту та інструментів обробки природної мови, що дозволяє реалізовувати персоналізоване навчання, адаптацію матеріалу та інтелектуальне оцінювання знань. Універсального рішення не існує, оскільки кожна платформа має свою специфіку, сильні сторони та обмеження. Проте саме завдяки цій різноманітності з'являється можливість гнучко комбінувати інструменти, будувати гібридні системи та створювати новітні освітні екосистеми, які орієнтовані на глибоку аналітику, якісне зворотне оцінювання та ефективне управління освітнім процесом. Інтелектуальні освітні платформи є ключем до побудови освіти майбутнього — адаптивної, доступної, ефективної та заснованої на даних.

1.3 Переваги та недоліки використання штучного інтелекту в освіті

Використання штучного інтелекту (ШІ) в освіті відкриває нові горизонти для організації, модернізації та переосмислення навчального процесу на всіх його етапах — від підготовки навчальних матеріалів до підсумкового оцінювання. ШІ забезпечує автоматизацію рутинних завдань, таких як перевірка тестів і контрольних, формування звітів про успішність, аналітика поведінки студентів, а

також дає змогу здійснювати персоналізоване навчання з урахуванням сильних і слабких сторін кожного студента. Завдяки технологіям ШІ можна реалізувати адаптивні курси, віртуальних помічників, автоматичне генерування контенту відповідно до потреб і запитів студентів, а також використовувати чат-ботів для зворотного зв'язку в реальному часі. У системах управління освітою ШІ дає можливість адміністраторам приймати обґрунтовані рішення на основі аналізу великих обсягів освітніх даних.

Водночас активне впровадження ШІ супроводжується комплексом викликів, які охоплюють не лише технічну сторону реалізації, а й глибокі соціальні, етичні, правові та педагогічні аспекти. Зокрема, для ефективного впровадження інтелектуальних систем навчальні заклади стикаються з проблемою забезпечення належної інфраструктури — це стосується як доступу до високошвидкісного інтернету, так і наявності сучасного обладнання, серверних потужностей та технічної підтримки. Фінансові витрати на закупівлю, ліцензування, інтеграцію й обслуговування ШІ-рішень можуть бути надмірними для багатьох освітніх закладів, особливо в країнах із обмеженим бюджетом. У такому випадку виникає потреба у державній підтримці або міжнародному грантовому фінансуванні. Соціальні виклики стосуються, насамперед, рівня довіри до рішень, які приймаються ШІ. Оскільки алгоритми можуть бути складними для розуміння, студенти та викладачі нерідко відчують скепсис щодо їхньої справедливості, особливо в питаннях оцінювання. Це підсилюється відсутністю прозорості в логіці прийняття рішень, що вимагає створення інтерпретованих та пояснюваних моделей.

Педагогічні виклики пов'язані з необхідністю адаптації традиційних методик викладання до нових умов цифрового середовища, що включає як зміну ролі викладача, так і трансформацію самого освітнього контенту. Викладачам потрібно опановувати нові цифрові інструменти, переглядати методи взаємодії зі студентами, а також формувати нові компетентності, зокрема в сфері цифрової етики та кібербезпеки. Окрему увагу слід приділити правовим аспектам —

використання ШІ у сфері освіти потребує чітко визначених норм щодо захисту персональних даних студентів, правил авторства на контент, відповідальності за результати, отримані за участю або під контролем ШІ, а також механізмів оскарження рішень, прийнятих автоматизованими системами. Упровадження ШІ в освіту вимагає не лише технічної компетентності, а й нового педагогічного мислення, готовності до інституційних і нормативних змін, формування цілісної політики цифрової трансформації та розробки етичних рамок взаємодії людини й машини в навчальному процесі.

Переваги використання ШІ в освіті:

Автоматизація оцінювання: ШІ дозволяє автоматично перевіряти великі обсяги завдань, зокрема тести, відкриті відповіді, есе та програмний код. Це скорочує час на перевірку, зменшує навантаження на викладачів і підвищує ефективність освітнього процесу.

Персоналізація навчання: Інтелектуальні системи можуть адаптувати навчальний контент під індивідуальні потреби студента, враховуючи його темп засвоєння знань, попередні помилки та рівень підготовки. Завдяки цьому створюється ефект "навчання з репетитором" у цифровому середовищі.

Аналітика успішності: ШІ дозволяє проводити глибокий аналіз результатів навчання, виявляти тенденції, прогалини у знаннях, ризики неуспішності та формувати звіти для студентів, викладачів і адміністраторів.

Доступність освіти: Завдяки автоматизованим платформам навчання стає доступним для широкого кола слухачів незалежно від географічного розташування. ШІ дозволяє масштабувати освітній процес без втрати якості.

Мотивація та інтерактивність: Штучний інтелект може бути інтегрований у гейміфіковані платформи, віртуальних помічників, чат-ботів, що підвищує зацікавленість студентів у навчанні та створює нові формати взаємодії.

Недоліки та виклики впровадження ШІ в освіту:

Обмежена здатність до глибокого розуміння: Незважаючи на досягнення в галузі обробки природної мови, сучасні моделі ШІ все ще не можуть повністю

розуміти контекст, іронію, емоції чи міждисциплінарні зв'язки в навчальних текстах, що може призводити до помилок у оцінюванні.

Ризики упередженості: Моделі машинного навчання можуть відтворювати й навіть підсилювати упередження, що містяться в навчальних даних, що загрожує об'єктивності оцінювання.

Недостатня прозорість алгоритмів: Студенти й викладачі можуть не розуміти логіку, за якою приймаються рішення щодо оцінок. Це може призводити до недовіри до систем ІІ та необхідності створення інтерпретованих моделей.

Залежність від технічної інфраструктури: Використання ІІ вимагає стабільного інтернет-зв'язку, сучасного обладнання та спеціалізованого програмного забезпечення, що може бути проблемою для малозабезпечених шкіл або країн із обмеженими ресурсами.

Етичні та правові аспекти: Використання ІІ в освіті ставить питання захисту персональних даних студентів, академічної доброчесності, прозорості алгоритмів, авторських прав і відповідальності за прийняті ІІ рішення.

Отже, штучний інтелект має колосальний потенціал змінити систему освіти, зробивши її не лише доступнішою, ефективнішою та індивідуалізованішою, а й гнучкішою та більш орієнтованою на потреби конкретного студента. Застосування ІІ сприяє переходу від традиційної моделі масового навчання до персоналізованої, де кожен студент отримує освітню траєкторію, яка враховує його здібності, стиль мислення та темп опанування знань.

Однак впровадження інтелектуальних технологій має відбуватися виважено — з урахуванням технічних, соціальних, етичних і правових ризиків. Важливо забезпечити не лише якість алгоритмів, а й їхню пояснюваність і прозорість. Потрібно також передбачити механізми людського контролю над автоматизованими рішеннями, особливо у процесі оцінювання, де від справедливості залежить довіра до системи в цілому. Важливим є дотримання етичних принципів: захист персональних даних, забезпечення рівного доступу до технологій, уникнення дискримінації та підтримка цифрової інклюзії. Освітні

системи майбутнього мають поєднувати в собі силу ШІ та гуманістичні цінності, де технології є не заміною викладача, а його інтелектуальним партнером.

Лише гармонійне поєднання автоматизації освітніх процесів із гуманістичними засадами навчання здатне забезпечити повноцінну інтеграцію штучного інтелекту в педагогічну практику. Алгоритмічні системи мають виступати не заміником, а доповненням до діяльності викладача — розширюючи його можливості, підвищуючи ефективність навчального процесу та надаючи нові інструменти для аналітики, моніторингу та підтримки студента. Педагогічна мудрість, етична відповідальність та емоційна складова взаємодії залишаються фундаментом освіти, на який мають спиратися будь-які цифрові інновації. Саме тому стратегія впровадження ШІ повинна базуватися на принципах прозорості, пояснюваності, захисту прав студентів та збереження людського контролю над ключовими рішеннями. Такий підхід дозволить не лише уникнути ризиків, а й перетворити ШІ на потужний каталізатор розвитку освітньої галузі в епоху цифрової трансформації..

Висновки до розділу 1

У першому розділі було проведено детальний аналіз сучасного стану освітніх технологій, орієнтованих на автоматичне оцінювання знань студентів, із фокусом на можливості та обмеження систем, які використовують елементи штучного інтелекту. Розглянуто функціональні особливості та архітектуру чотирьох провідних інтелектуальних освітніх платформ: Moodle, Khan Academy, Coursera та Gradescope. Було здійснено їхню порівняльну характеристику за такими критеріями, як адаптивність, підтримка ШІ, типи контенту, можливості аналітики та персоналізації навчання. Під час дослідження було встановлено, що більшість сучасних платформ уже інтегрують базові інструменти ШІ — автоматичне оцінювання тестів, генерацію рекомендацій, адаптацію контенту до користувача. Проте повноцінна інтеграція технологій обробки природної мови та глибоке семантичне оцінювання відкритих відповідей досі залишається

малодослідженою і технічно складною задачею. Аналіз наявних рішень також засвідчив, що у сфері оцінювання відкритих завдань існує значна потреба в інноваціях, орієнтованих на глибоке розуміння текстів, логіки міркувань, аргументації та критичного мислення студентів.

У підрозділі 1.3 було проаналізовано переваги та недоліки використання ІІІ в освіті. До ключових переваг належать автоматизація, персоналізація, аналітика та доступність. Водночас було визначено низку викликів, серед яких: технічні обмеження, етичні ризики, складність інтерпретації рішень ІІІ, а також потреба в новому педагогічному мисленні та нормативному забезпеченні.

Узагальнюючи викладене, можна зробити висновок, що штучний інтелект відкриває широкі перспективи для трансформації освіти, але його впровадження повинно супроводжуватися комплексним підходом, що охоплює як технічні, так і соціально-педагогічні аспекти. Необхідно розробляти інтелектуальні системи оцінювання, які будуть не лише ефективними, але й прозорими, етичними та орієнтованими на підтримку студента в навчальному процесі. Саме на таких принципах має ґрунтуватися подальше проектування системи, що розглядається в наступному розділі.

2 ПРОЄКТУВАННЯ СИСТЕМИ АВТОМАТИЧНОГО ОЦІНЮВАННЯ ЗНАНЬ

2.1 Архітектура запропонованої системи оцінювання

Запропонована система автоматичного оцінювання знань студентів є багаторівневою інтелектуальною платформою нового покоління, яка поєднує у собі технології клієнт-серверної архітектури, реляційного зберігання даних, глибокого навчання (нейронних мереж), модулів природної мовної обробки (NLP), а також сучасного адаптивного веб-інтерфейсу для зручної та інтуїтивної взаємодії з користувачами. Вона орієнтована на комплексну автоматизацію процесу перевірки знань, охоплюючи як структуровані типи завдань (множинний вибір, відповідність, істина/неправда), так і неструктуровані (розгорнуті письмові відповіді, есе, фрагменти коду тощо). Основною метою є забезпечення високого рівня об'єктивності, прозорості, масштабованості, а також адаптивності оцінювання відповідно до індивідуальних особливостей студентів і предметної специфіки. Система дозволяє інтегрувати аналітику прогресу, надавати миттєвий зворотний зв'язок, автоматично генерувати звітність і сприяти персоналізованому навчанню на основі аналізу результатів кожного користувача.

Система забезпечує повноцінну цифрову екосистему, що охоплює весь цикл перевірки знань — від створення, редагування та адаптації тестів відповідно до предметної специфіки, збору відповідей і попередньої обробки текстів до глибокого семантичного аналізу, формування висновків і повноцінної генерації звітної документації. Архітектура системи базується на мікросервісному підході з підтримкою масштабування на рівні модулів, а також інтеграції з іншими навчальними системами, CRM, аналітичними інструментами та електронними журналами. У контексті складних завдань система здатна оцінювати не лише текстові відповіді, а й графічні об'єкти, структуровані дані, математичні вирази та фрагменти коду. Центральне місце в системі займає інтелектуальний помічник викладача, що не лише автоматизує перевірку знань, а й здатен аналізувати стилі

навчання, адаптувати завдання до рівня студента та виявляти потенційні зони ризику у знаннях для подальшої корекції навчального плану. Така система створює передумови для впровадження безперервного формативного оцінювання та підвищення якості освіти завдяки точному, своєчасному та персоналізованому зворотному зв'язку.

Архітектура системи складається з таких основних компонентів:

Клієнтський інтерфейс (frontend) — реалізований у вигляді веб-інтерфейсу, через який викладачі можуть створювати тести, переглядати результати та управляти сесіями оцінювання, а студенти — проходити тести, переглядати власний прогрес і отримувати зворотний зв'язок. Інтерфейс створено з використанням HTML, CSS, JavaScript і бібліотеки Bootstrap для забезпечення адаптивності.

Серверна частина (backend) — побудована на основі мікрофреймворку Flask (Python). Вона відповідає за логіку додатку, обробку запитів, взаємодію з базою даних, виклик моделей машинного навчання та надання відповідей клієнтському інтерфейсу.

База даних — реалізована з використанням SQLite (або PostgreSQL для масштабованих версій). База зберігає інформацію про користувачів, створені тести, відповіді студентів, результати оцінювання, а також журнали активності.

Модуль машинного навчання — включає модель на основі нейронної мережі, яка відповідає за обробку відкритих відповідей студентів. Для реалізації моделі використовується TensorFlow або Keras. У базовій версії модель виконує класифікацію текстових відповідей за рівнем повноти, логіки та відповідності правильній відповіді. Для відкритих питань застосовується NLP-підхід з використанням ембеддингів слів (наприклад, Word2Vec або BERT).

Інтеграційний API-шар — забезпечує взаємодію між модулями, а також потенційну інтеграцію з іншими платформами (наприклад, Moodle, Google Classroom). Він дозволяє викликати модель оцінювання через REST API,

обмінюватися даними з зовнішніми LMS та забезпечувати підключення аналітичних модулів.

Модуль авторизації та безпеки — реалізує механізми реєстрації, входу, шифрування паролів, розмежування ролей (студент/викладач/адміністратор) та захисту від зовнішніх атак. Забезпечує збереження конфіденційності даних згідно з GDPR та іншими стандартами.

Система підтримує принцип модульності, що забезпечує можливість гнучкого масштабування функціональності, оновлення окремих компонентів без зупинки всієї системи, а також інтеграції новітніх технологічних рішень на основі потреб навчального середовища. Наприклад, модульна структура дозволяє підключати різні моделі машинного навчання — як загального призначення, так і вузькоспеціалізовані, для оцінювання відкритих відповідей, перевірки програмного коду, розпізнавання графічних об'єктів, інтерпретації математичних формул і навіть технічних креслень. Крім того, система може адаптувати алгоритми до предметних областей — з урахуванням специфіки навчальних дисциплін, рівня складності та стилю викладання. Це дозволяє не тільки масштабувати платформу під різні типи навчальних закладів (школи, коледжі, університети), але й забезпечити високу якість оцінювання у міждисциплінарному навчанні, включно з гуманітарними, технічними, природничими, інженерними й художніми напрямками. Такий підхід також сприяє персоналізації навчального процесу, дозволяючи адаптувати інструменти перевірки знань до індивідуального стилю студента, що є критично важливим для сучасної освіти.

Архітектура системи підтримує динамічну зміну конфігурацій бази даних, що критично важливо при обробці великих обсягів даних у масштабованих освітніх середовищах. Вона дозволяє автоматичне перемикання між локальними та віддаленими сховищами, з підтримкою таких функцій, як кластеризація, реплікація, резервне копіювання та балансування навантаження. При переході до серверного середовища з використанням PostgreSQL або MySQL відкриваються можливості для реалізації вертикального й горизонтального масштабування,

підвищення продуктивності за рахунок розподілених запитів, а також підключення до систем резервування на рівні контейнеризації (Docker, Kubernetes). Архітектура передбачає гнучку інтеграцію з хмарними середовищами — AWS, Heroku, Google Cloud, DigitalOcean — які забезпечують автоматичне масштабування, моніторинг у реальному часі, керування контейнерами, CI/CD-пайплайни, швидке оновлення окремих компонентів та високу відмовостійкість. Це особливо важливо для використання системи в національному масштабі або у разі значного зростання кількості користувачів. Інтеграційний API-шар виконує роль адаптивного шлюзу, реалізованого з дотриманням стандартів REST, OpenAPI та підтримкою JSON- і XML-форматів обміну. Він дозволяє безшовно підключати систему до LMS (Moodle, Google Classroom, Canvas), CRM-систем, аналітичних платформ (Tableau, Power BI), національних електронних освітніх реєстрів тощо. Універсальність API дає змогу будувати повноцінну екосистему навчання, не змінюючи бізнес-логіку інших цифрових компонентів і забезпечуючи масштабованість та стандартизованість даних між підсистемами.

Запропонована архітектура забезпечує не лише високу відмовостійкість і масштабованість, але й повну гнучкість у налаштуванні відповідно до потреб конкретного навчального закладу чи навіть окремої навчальної програми. Її адаптивність дозволяє інтегрувати нові освітні модулі, сучасні інтерфейси взаємодії, системи аналітики та сторонні освітні сервіси, що значно підвищує рівень персоналізації освіти та якість навчального процесу. Платформа може бути легко розгорнута як у локальному середовищі, так і в хмарній інфраструктурі з підтримкою контейнеризації, автошардингу, CDN та централізованого моніторингу. Завдяки використанню відкритих стандартів API, система забезпечує прозору інтеграцію з національними реєстрами, електронними журналами, ERP- та CRM-системами закладів освіти, що відкриває можливості для комплексної цифровізації процесів навчання й оцінювання. Також представлена загальна архітектура системи (рис. 2.1). У наступних підрозділах буде докладно розглянуто

процес вибору моделі нейронної мережі, її навчання, застосовані методи обробки природної мови, формування результатів оцінювання, адаптивну логіку формування тестів, а також аспекти UI/UX для різних типів користувачів — викладачів, студентів і адміністраторів.

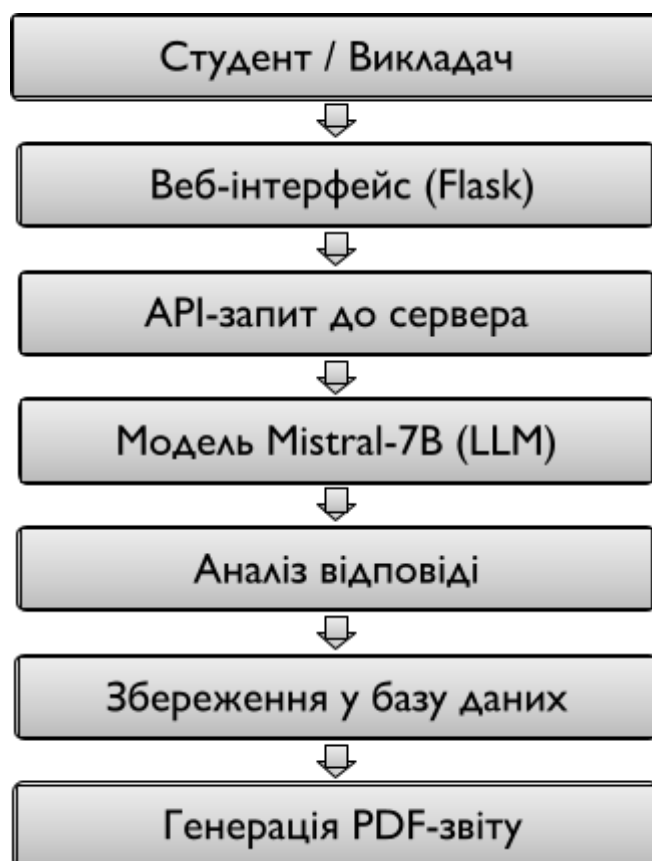


Рисунок 2.1 – Загальна архітектура системи

2.2 Вибір моделі нейронної мережі для аналізу відповідей студентів

Для реалізації модуля оцінювання відкритих текстових відповідей у системі було обрано підхід, що базується на сучасних трансформерних архітектурах мовних моделей, які на сьогодні вважаються найефективнішими в задачах обробки природної мови (Natural Language Processing, NLP). Ці моделі здатні виконувати складні семантичні інтерпретації тексту, виявляти логічні зв'язки між реченнями, враховувати контекст, граматику, стилістику та логіку викладення думки. Однією з головних вимог до моделі було забезпечення гнучкого підходу до

аналізу відповідей студентів, які можуть бути як короткими, так і розгорнутими, містити часткові знання або альтернативні формулювання правильної відповіді. Саме тому вибір трансформерних моделей, здатних обробляти відкриті питання та генерувати якісний текстовий фідбек, став пріоритетом.

Серед потенційних варіантів були розглянуті такі сучасні моделі глибокого навчання:

BERT (Bidirectional Encoder Representations from Transformers) (рис. 2.2) — глибока двонаправлена мовна модель, розроблена компанією Google. Її архітектура заснована на механізмі самоуваги (self-attention), який дозволяє моделі одночасно враховувати контекст слів зліва і справа. BERT є потужним інструментом для задач класифікації тексту, виявлення тональності, пошуку інформації та семантичного аналізу. У контексті оцінювання знань студентів BERT можна використовувати для класифікації відповідей на "правильно/неправильно", "повна/часткова/неповна" тощо. Проте BERT не є генеративною моделлю і не може створювати пояснення до оцінки, що обмежує її застосування у завданнях з потребою у зворотному зв'язку.

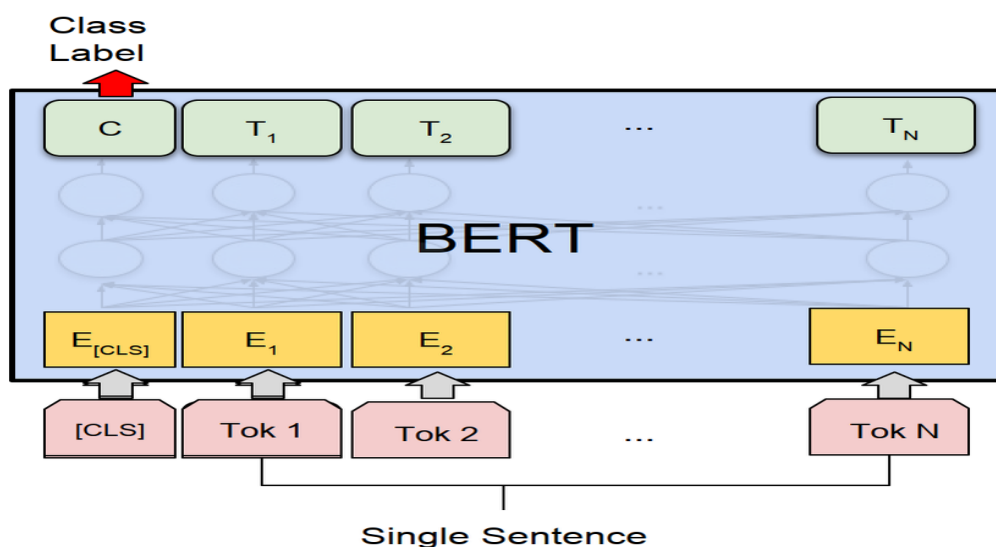


Рисунок 2.2 – Алгоритм роботи BERT

GPT (Generative Pretrained Transformer) (рис. 2.3) — серія моделей від компанії OpenAI, спеціалізованих на генерації зв'язного тексту. GPT навчається у режимі autoregressive, що дозволяє їй створювати продовження тексту на основі

вхідної інструкції. Завдяки потужності трансформерної архітектури GPT здатна інтерпретувати відкриті відповіді студентів, надавати логічні коментарі, виявляти змістові недоліки або неправильні формулювання. Однак базові версії GPT вимагають значних обчислювальних ресурсів та часто використовуються через API, що ускладнює локальне впровадження в навчальних закладах.

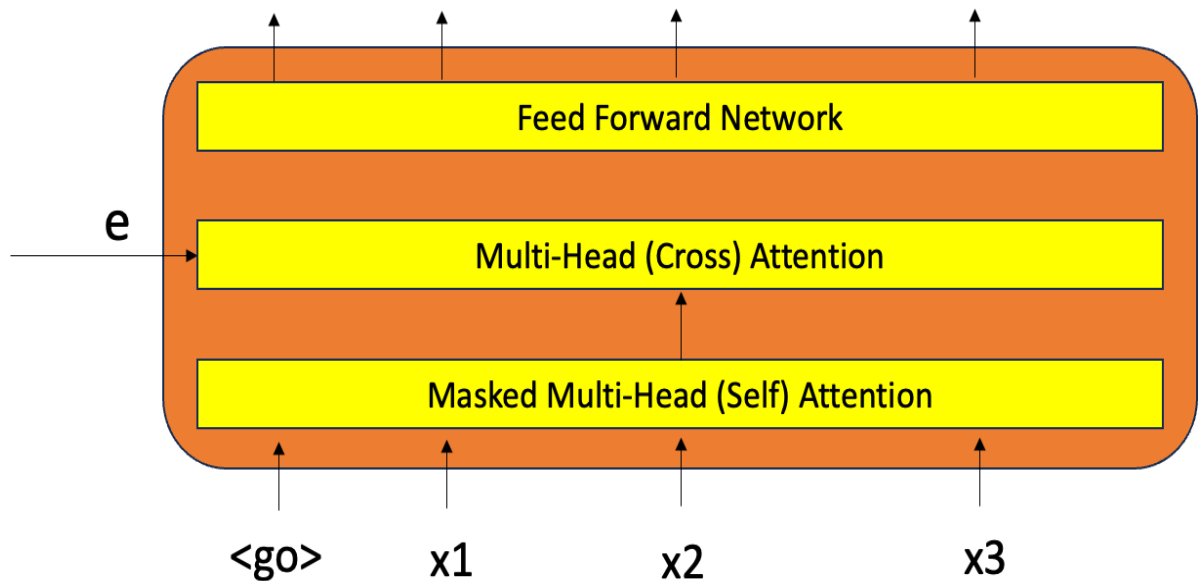


Рисунок 2.3 – Алгоритм роботи GPT

Mistral-7B Instruct (рис. 2.4) — одна з новітніх open-source моделей великого розміру (7 мільярдів параметрів), адаптована до виконання інструкцій (instruction-tuned). Розроблена для швидкої генерації точних і стислих відповідей на запити користувача, вона чудово підходить для інтерактивних освітніх сценаріїв. У нашому проєкті Mistral-7B Instruct показала відмінну продуктивність при оцінюванні письмових відповідей студентів, демонструючи здатність не лише оцінювати точність, але й пояснювати, чому відповідь правильна, частково правильна або хибна. Важливою перевагою є можливість локального розгортання моделі з використанням квантизації, що значно знижує вимоги до обчислювальних ресурсів.

У дипломному проєкті реалізовано локальний запуск моделі Mistral-7B Instruct за допомогою бібліотеки llama-cpp-python, що дозволяє уникнути

залежності від сторонніх API та забезпечити автономність системи в навчальному середовищі. Для запуску використовувалася квантизована версія моделі у форматі Q4_K_M, що дозволяє працювати з великою мовною моделлю навіть на середньостатистичному комп'ютері з 16 ГБ оперативної пам'яті та сучасним процесором з підтримкою AVX-інструкцій.

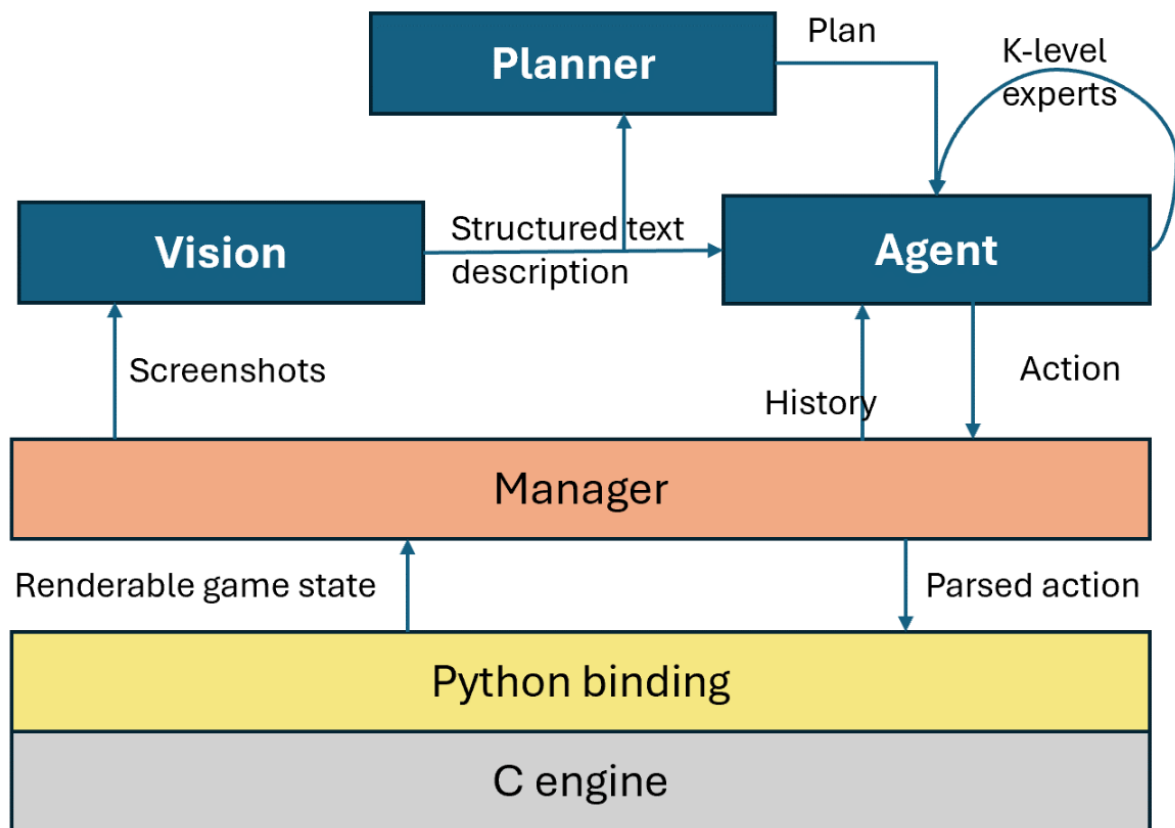


Рисунок 2.4 – Алгоритм роботи Mistral-7B

Архітектура взаємодії із моделлю побудована на базі інструкційного запиту (instruction prompt), який має такий формат:

1. Інструкція: "Оціни відповідь студента на питання..."
2. Формулювання запитання
3. Власне відповідь студента

У відповідь модель формує текстовий блок, що містить:

- числову оцінку (наприклад, "Оцінка: 7/10");
- пояснення причин, чому саме така оцінка виставлена ("Коментар: Частково правильна відповідь, бракує згадки про веб-застосування");

- додаткові рекомендації щодо вдосконалення відповіді (за потреби).

Приклад запиту й відповіді:

Запит: Оціни відповідь студента на питання *"Що таке HTTP?"*.

Відповідь студента: *"Це протокол для передачі файлів."*

Відповідь моделі: Оцінка: 7/10. Коментар: *Частково правильна відповідь. HTTP справді є протоколом, однак він призначений для передачі гіпертексту у мережі Інтернет, зокрема для роботи вебсайтів.*

Такий формат відповіді дозволяє формувати обґрунтований фідбек для студента, який отримує не лише оцінку, але й текстове пояснення, що підвищує прозорість і навчальну цінність процесу перевірки. У той самий час викладач має змогу бачити логіку оцінювання та втручатися за потреби. Це значно перевершує функціональність традиційних тестових систем, які переважно базуються на жорстких відповідностях без врахування контексту та логіки мислення. Параметри запуску моделі були ретельно підібрані з урахуванням доступних обчислювальних ресурсів і вимог до продуктивності. Для забезпечення швидкої та стабільної роботи в середовищі університетського серверу або персонального комп'ютера було обрано такі технічні параметри:

Формат квантизації: Q4_K_M — це один із найпоширеніших форматів квантизації для великих мовних моделей, який зменшує обсяг оперативної пам'яті, необхідної для запуску, при незначній втраті точності. Цей формат дозволяє запускати Mistral-7B на комп'ютерах без графічного процесора (GPU), що особливо актуально для освітніх закладів із обмеженими ресурсами.

Локальний запуск моделі — модель запускала без використання зовнішніх API (таких як OpenAI або Hugging Face API), що гарантує повну автономність, конфіденційність обробки відповідей студентів, а також незалежність від інтернет-з'єднання. Такий підхід особливо важливий у навчальних закладах, де питання безпеки даних і приватності відіграють ключову роль.

Інтеграція з Flask через спеціалізований ендпоінт POST /evaluate — модель вбудована у backend-сервер системи та приймає POST-запити з параметрами:

текст запитання та відповідь студента. Відповідь повертається у форматі JSON з полями `score`, `comment`, що полегшує подальше збереження в базу даних, візуалізацію в інтерфейсі та генерацію PDF-звітів. Додатково були протестовані конфігурації із багатопоточністю, кешуванням відповідей для повторних звернень, а також логуванням усіх запитів для подальшого аналізу та перевірки якості.

Вибір моделі Mistral-7B Instruct у межах даного дипломного проекту є не лише технічно обґрунтованим, а й стратегічно виваженим. По-перше, модель забезпечує достатню точність оцінювання, демонструючи високу узгодженість результатів із оцінками, що виставляються викладачами. По-друге, її можливість локального запуску без потреби у підключенні до сторонніх сервісів дозволяє зберігати конфіденційність даних студентів і гарантувати стабільність роботи навіть в умовах обмежених інфраструктурних ресурсів. По-третє, модель підтримує генерацію інтерпретованих відповідей, що підвищує прозорість та об'єктивність освітнього процесу. Нарешті, архітектурна сумісність із технологічним стеком системи (Flask, REST API, SQLite) робить її ідеальним компонентом для інтеграції без додаткових витрат на адаптацію. Усі ці фактори разом формують міцну основу для використання Mistral-7B Instruct як ядра інтелектуальної освітньої платформи нового покоління..

2.3 Алгоритм оцінювання: генерація, інтерпретація та виставлення балів

Процес оцінювання студентських відповідей у межах запропонованої системи реалізовано на основі концепції інструкційного запиту до великої мовної моделі (LLM), що базується на принципах сучасної обробки природної мови, глибокого семантичного аналізу та генеративного підходу до створення тексту. LLM (Large Language Model) – це нейромережева система, натренована на величезних корпусах текстів, яка здатна не лише обробляти мовну інформацію, а й генерувати логічно узгоджені, граматично правильні та змістовно релевантні тексти у відповідь на задані інструкції. У контексті освітньої діяльності це відкриває нові можливості для інтелектуального оцінювання знань студентів. На

відміну від традиційних систем тестування, які зазвичай опираються на жорсткі шаблони, ключові слова або бінарну логіку перевірки (правильно/неправильно), запропонований підхід передбачає комплексний аналіз відповіді. Модель вивчає логіку викладення, здатність студента структурувати думку, наявність аргументів, точність формулювань, використання термінології, зв'язність речень та повноту розкриття теми. Важливо, що оцінювання здійснюється з урахуванням контексту, а не лише ізольованих фрагментів тексту, що дозволяє уникнути механістичного підходу й підвищує об'єктивність оцінки. Мовна модель здатна адаптуватися до різних стилів і рівнів складності відповідей, завдяки чому система однаково ефективно працює як з простими поняттями, так і зі складними аналітичними завданнями. Це робить LLM універсальним інструментом для освітніх цілей, здатним охопити як гуманітарні, так і технічні дисципліни. В результаті система не просто виставляє оцінку, а виконує функцію інтелектуального асистента, що підсилює навчальний процес через надання зворотного зв'язку, рекомендацій і дидактичних підказок.

Алгоритм оцінювання побудований як послідовність взаємопов'язаних інтелектуальних етапів, кожен із яких виконує специфічну функцію — від формування запиту до генерації результату та його фіксації в базі даних. На відміну від традиційних підходів, де перевірка зводиться до зіставлення з фіксованим шаблоном, ця система здійснює повноцінний семантичний аналіз тексту. У процесі роботи модель оцінює не лише наявність ключових термінів, а й глибину розкриття теми, логічну структуру відповіді, аргументованість, стилістику викладення. Такий підхід дозволяє формувати багатовимірну оцінку знань, яка включає як числову складову, так і пояснення до неї. Крім того, алгоритм реалізований у спосіб, що забезпечує масштабованість — система може бути легко адаптована під інші предметні області, типи тестування (відкриті питання, есе, задачі на логіку тощо), та навіть під індивідуальні потреби навчальних закладів. Можливість інтеграції з LMS-платформами, аналітичними модулями та зовнішніми API розширює функціонал і дозволяє створити

повноцінну екосистему цифрового оцінювання з високим рівнем автоматизації та гнучкості.

Основні етапи алгоритму:

1. Формування запиту (Prompt engineering):

- Система формує структурований запит для LLM у форматі:
- "Оціни відповідь студента на питання: '<питання>'. Відповідь студента: '<відповідь>'."
- Цей формат дозволяє моделі зрозуміти інструкцію та забезпечує послідовний стиль відповіді незалежно від тематики запитання.

2. Передача даних у LLM:

- Запит передається на сервер моделі (локально через llama-cpp або через API), де активується механізм генерації відповіді.

3. Аналіз тексту моделю:

- Модель виконує семантичний та контекстуальний аналіз відповіді.
- Визначаються ключові поняття, логіка викладення, відповідність очікуваній правильній відповіді.
- У разі потреби модель фіксує неповноту, поверховість або змішування понять.

4. Генерація оцінки:

- Модель формує текст у структурі: Оцінка: X/10. Коментар: ...
- Числова частина (від 0 до 10) виставляється згідно з відповідністю до очікуваної відповіді та складності запитання.
- Коментар пояснює причини, що вплинули на оцінку: правильність, аргументація, повнота.

5. Формування відповіді системи:

- Сформована оцінка перетворюється у JSON-формат, наприклад:

```
{
  "score": 7,
```

"comment": "Відповідь частково правильна. Відсутня згадка про веб-контекст протоколу HTTP."

}

- Цей формат дозволяє легко відображати результати у вебінтерфейсі та записувати їх до бази даних.

6. Збереження та відображення результату:

- Оцінка зберігається у базі даних, асоціюється з ID студента та ID запитання.
- Вона стає доступною в особистому кабінеті викладача та студента.
- Додатково генерується PDF-звіт із результатами тестування, де відображаються всі відповіді, бали та коментарі.

Переваги такого алгоритму:

- Прозорість: викладач і студент бачать не лише оцінку, а й її обґрунтування.
- Автоматизація: усі етапи — від генерації до збереження — виконуються без ручного втручання.
- Адаптивність: алгоритм працює незалежно від предметної області — гуманітарні, технічні, аналітичні запитання.
- Гнучкість: систему легко налаштувати під різні шкали оцінювання, критерії та формати відповіді.

Схематично зображено внутрішні етапи обробки текстової відповіді студентів у межах трансформерної моделі Mistral-7B (рис. 2.5). Алгоритм починається з формування інструкційного запиту (prompt), що включає питання та відповідь студента. Далі відбувається токенізація – перетворення тексту на окремі елементи (слова, символи, частини слів). На наступному етапі токени трансформуються у вектори ознак, які враховують контекст. Потім модель аналізує зв'язки між словами у реченнях за допомогою механізму self-attention. Завершальним етапом є генерація оцінки та пояснення, що надалі повертається у форматі тексту та інтегрується у систему. Такий підхід дозволяє реалізувати глибоке семантичне оцінювання з високим рівнем точності.



Рисунок 2.5 - Внутрішні етапи обробки текстової відповіді

Розроблений алгоритм забезпечує комплексну, багатовимірну та обґрунтовану оцінку рівня знань студента. Він враховує не лише формальну відповідність відповіді правильному шаблону, а й логічну послідовність викладення, повноту розкриття теми, наявність ключових понять, аргументованість, граматичну правильність та стилістичну якість. Завдяки поєднанню глибокої семантичної обробки тексту та здатності нейромереж до генеративного моделювання, система забезпечує високу точність і послідовність оцінювання. Крім того, наявність пояснювальних коментарів до кожної оцінки сприяє формуванню зворотного зв'язку між студентом і викладачем, що має позитивний педагогічний ефект.

У результаті створюється нова якість освітнього оцінювання, яка об'єднує інтелектуальну потужність штучного інтелекту, гнучкість цифрових платформ і гуманістичні принципи навчального процесу. Такий підхід здатен радикально змінити роль оцінювання — з інструменту контролю воно перетворюється на засіб діагностики, розвитку та підтримки індивідуальної освітньої траєкторії кожного студента.

Висновок до розділу 2

У другому розділі дипломної роботи було здійснено детальний аналіз архітектурних особливостей, вибору та інтеграції нейронної моделі для оцінювання рівня знань студентів. Розглянуто можливі підходи до обробки природної мови, зокрема моделі BERT, GPT і Mistral-7B, а також обґрунтовано вибір саме Mistral-7B Instruct як оптимального рішення. Підтверджено, що ця модель здатна забезпечити якісну генерацію оцінок із поясненнями, при цьому працюючи в локальному середовищі без залежності від зовнішніх API.

Описано алгоритм оцінювання, який охоплює всі етапи – від формування інструкційного запиту до збереження результатів та генерації зворотного зв'язку у форматі JSON. Запропонована схема дозволяє реалізувати прозору, масштабовану та адаптивну систему оцінювання, яка враховує семантику, логіку та глибину розуміння змісту відповідей.

В межах розділу було подано візуальні схеми, які відображають внутрішню логіку роботи трансформерної моделі, підкреслюючи складність та інтелектуальну гнучкість запропонованого підходу. Таким чином, розділ 2 формує міцне підґрунтя для подальшої реалізації системи в рамках розділу 3, де буде представлено практичну частину – програмну реалізацію та результати тестування..

3 РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОЇ СИСТЕМИ

3.1 Технологічний стек, середовище розробки та компоненти системи

Для реалізації системи автоматичного оцінювання знань студентів було обрано сучасний та ефективний технологічний стек, який поєднує інструменти для веброзробки, обробки природної мови, генерації PDF-звітів та локального розгортання великих мовних моделей. Основною метою при виборі технологій було досягнення високої продуктивності, стабільності роботи, зручності підтримки та розширення системи, а також забезпечення можливості впровадження її в умовах реального освітнього закладу без необхідності у дорогому серверному обладнанні або платних підписках. Архітектура проєкту реалізована як повноцінний веборієнтований застосунок, у якому кожен структурний елемент виконує свою чітко визначену роль. Клієнтська частина (frontend) відповідає за взаємодію з користувачем: введення даних, перегляд результатів, навігацію між розділами та візуальне представлення функцій. Вона забезпечує інтуїтивно зрозумілий інтерфейс як для студентів, так і для викладачів. Серверна частина (backend) обробляє бізнес-логіку застосунку: перевірку прав доступу, обробку форм, облік результатів, виклики до нейронної моделі та підготовку даних для бази.

База даних виконує роль постійного сховища навчальної інформації: тестів, питань, результатів, коментарів моделі, профілів користувачів. Застосування реляційної структури забезпечує збереження цілісності зв'язків між сутностями та дозволяє швидко витягати потрібну інформацію для аналізу. Модуль взаємодії з мовною моделлю реалізований як окрема серверна служба, що приймає запити через API, формує інструкційний prompt, надсилає його до моделі Mistral-7B, отримує результат (оцінку та пояснення) й передає назад у систему для подальшого збереження та відображення користувачам. Така модульна організація забезпечує високий рівень масштабованості: кожен компонент можна замінити, оновити або винести в окремий контейнер чи сервер без порушення загальної

логіки системи. Ізольованість компонентів також спрощує налагодження та тестування, а логічне розмежування відповідальностей відповідає принципам сучасної інженерії програмного забезпечення. Завдяки цьому архітектурному рішенню система демонструє стабільність, гнучкість та готовність до впровадження в різноманітних освітніх сценаріях. Окрему увагу було приділено вибору технологій із відкритим вихідним кодом, що дозволяє будь-якому навчальному закладу адаптувати й впровадити систему без додаткових витрат, а також вносити зміни у відповідності до своїх внутрішніх освітніх і адміністративних потреб. Обрані технології демонструють гармонійне поєднання сучасного веброзробницького підходу та інноваційних засобів штучного інтелекту на основі трансформерних моделей.

Основні компоненти:

Frontend (інтерфейс користувача): реалізовано з використанням HTML, CSS і JavaScript. Для стилізації застосовано фреймворк Bootstrap, що забезпечує адаптивний і сучасний вигляд інтерфейсу. Інтерфейс поділений на два типи кабінетів: студентський (рис. 3.1) та викладацький (рис. 3.2), з відповідним набором функцій(переглянути тести (рис. 3.3), перегляд результатів (рис. 3.4).

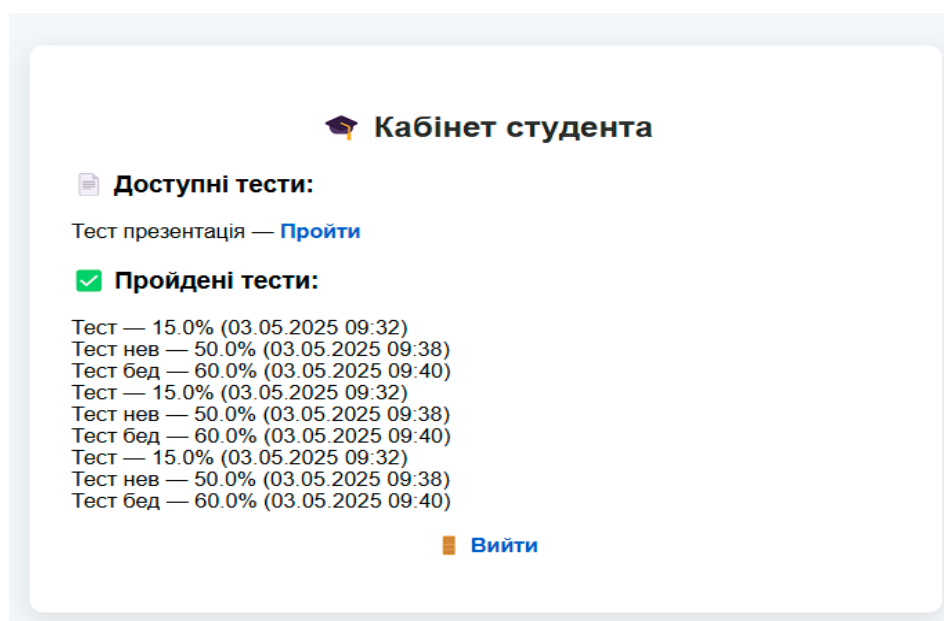


Рисунок 3.1 – Кабінет студента

Backend (серверна логіка): побудований на основі мікрофреймворку Python – Flask. Сервер відповідає за обробку запитів, аутентифікацію, взаємодію з базою даних, обробку відповідей студентів, виклик мовної моделі та формування результатів оцінювання.

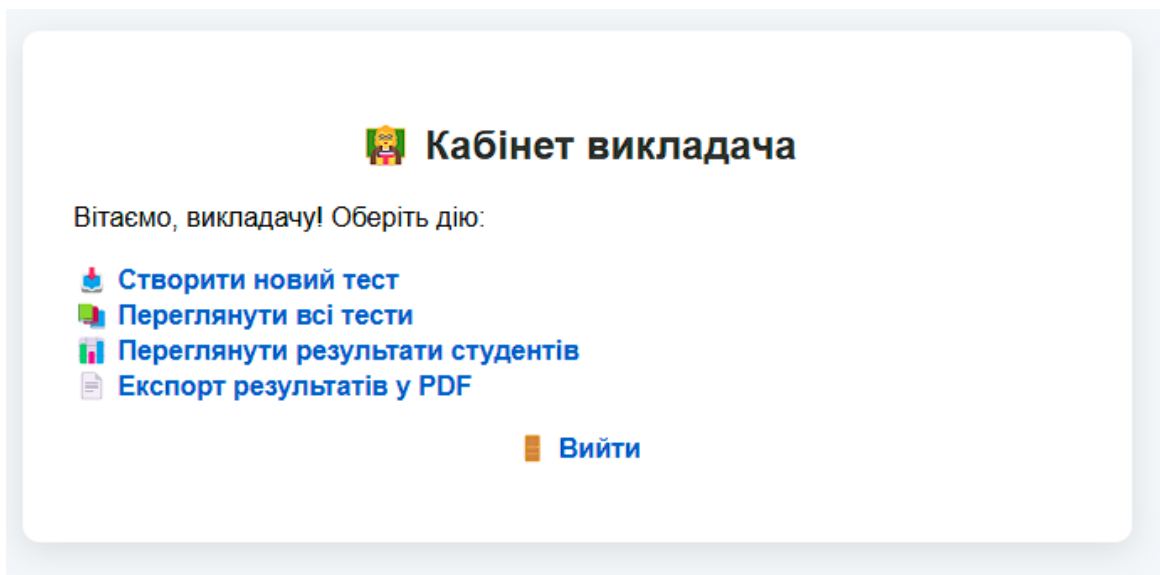


Рисунок 3.2 – Кабінет викладача

База даних: застосовано SQLite як легку реляційну базу для зберігання інформації про користувачів, тести, відповіді, оцінки та згенеровані звіти. База структурована на основі ключових сутностей: Student, Teacher, Test, Question, Answer, Result.

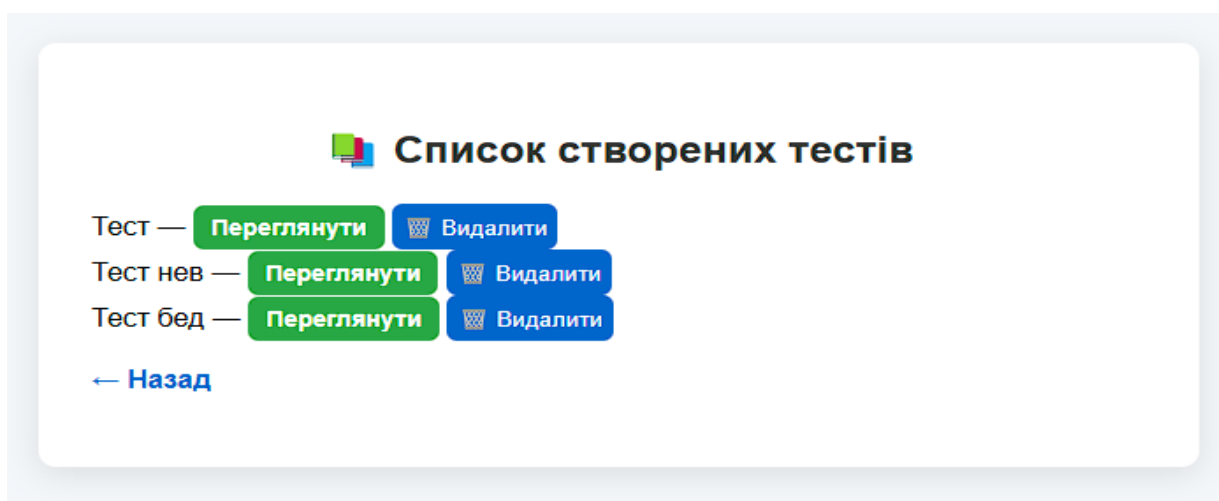


Рисунок 3.3 – Список тестів

Мовна модель: Mistral-7B Instruct у форматі Q4_K_M, розгорнута локально з використанням бібліотеки llama-cpp-python. Запуск відбувається через окремий Python-процес, що приймає запити від Flask-сервера через HTTP API.

Результати студентів					
Студент	Тест	Оцінка	Спроби	Дата	Дія
student	Тест бед	60.0%	1 з 1	03.05.2025 09:40	✓ Дозволити ще
student	Тест бед	100.0%	2 з 2	03.05.2025 09:39	✓ Дозволити ще
student	Тест нев	50.0%	3 з 1	03.05.2025 09:38	✓ Дозволити ще
student	Тест нев	100.0%	4 з 2	03.05.2025 09:36	✓ Дозволити ще
student	Тест	15.0%	5 з 1	03.05.2025 09:32	✓ Дозволити ще

← Назад до кабінету

Рисунок 3.4 – Результати тестів

PDF-звітність: реалізована за допомогою бібліотеки ReportLab. Дозволяє формувати повноцінні звіти з результатами тестування, які включають питання, відповіді студентів, оцінки, коментарі моделі та загальний підсумок.

Інструменти розробки:

- Python 3.10
- Flask 2.x
- SQLite 3
- llama-cpp-python
- HTML5/CSS3/JavaScript (Bootstrap 5)

- Jinja2 (для шаблонізації вебсторінок)
- VS Code / JupyterLab (середовище розробки)

Така архітектура забезпечує легкість у налаштуванні, оскільки всі компоненти працюють у межах зрозумілих і добре документованих середовищ, які не потребують складної конфігурації або специфічного обладнання. Система підтримує масштабування як по вертикалі (заміна SQLite на повноцінну серверну базу даних на зразок PostgreSQL), так і по горизонталі (рознесення компонентів на окремі сервери, підключення кешування, балансування навантаження). Це відкриває можливості для розгортання як у малих академічних середовищах, так і у великих університетських кампусах. Завдяки чітко визначеній ролі кожного компонента — від мовної моделі до генератора звітів — система дозволяє розділяти права доступу та функціонал між студентами, викладачами й адміністраторами, що відповідає принципам ролевої безпеки та модульності. Інтеграція зі сторонніми сервісами, такими як навчальні платформи (LMS), хмарні сховища чи бібліотеки, також можлива через стандартні REST API або webhook-механізми.

Усі інструменти, використані для розробки системи, мають відкритий вихідний код і вільну ліцензію, що усуває фінансові бар'єри на етапі впровадження. Це робить запропоновану систему не лише технічно ефективною, але й соціально доступною — вона придатна до адаптації, впровадження і вдосконалення силами самих викладачів або студентських команд у межах внутрішніх проєктів.

Загальна схема компонентів розробленої системи, що візуалізує архітектурну побудову та взаємодію між її ключовими модулями (рис. 3.5). Система складається з шести базових компонентів, кожен з яких відповідає за критично важливу функцію в процесі оцінювання знань студентів.

Першим елементом є веб-інтерфейс, реалізований засобами Flask і HTML/CSS. Він відповідає за зручність взаємодії користувачів із системою,

включаючи реєстрацію, авторизацію, вибір тестів, подання відповідей і перегляд результатів.

Другим компонентом є мовна модель Mistral-7B, інтегрована через llama.cpp. Вона виконує роль "інтелектуального ядра" системи: аналізує текстові відповіді, виставляє оцінки, генерує розгорнуті коментарі до кожного результату. Модель працює в локальному середовищі та отримує вхідні інструкції у вигляді запитів (prompts).

База даних (SQLite або PostgreSQL) забезпечує централізоване зберігання всієї інформації про користувачів, тести, відповіді та оцінки. Вона є критичним елементом для забезпечення цілісності та збереження навчальної аналітики.

Наступним важливим блоком є генератор звітів, який реалізовано на основі бібліотеки ReportLab. Цей модуль формує зручні для друку PDF-файли, які містять структуровану інформацію про результати тестування з коментарями, питаннями, відповідями та підсумковими оцінками.

Компонент панелі користувачів, побудований з використанням HTML та Jinja, відповідає за візуальне відображення персональних кабінетів викладача і студента. Тут користувачі мають змогу створювати тести, проходити оцінювання та переглядати історію результатів.

Завершальним елементом є API для AI-запитів, реалізований на Python із використанням маршрутизації Flask. Саме цей модуль здійснює передачу даних від системи до мовної моделі та приймає її відповіді для подальшої обробки.

Кожен компонент є логічно ізольованим, але взаємопов'язаним. У сукупності вони забезпечують повний цикл оцінювання: від створення тестів і введення відповідей — до генерації пояснень, виставлення балів, збереження результатів і формування звітів. Завдяки модульності та гнучкій архітектурі система може бути легко адаптована або розширена під потреби конкретного навчального закладу.



Рисунок 3.5 – Структура компонентів системи автоматичного оцінювання.

3.2 Реалізація модуля тестування та оцінювання відповідей

Модуль тестування є центральним елементом системи, який виконує ключову роль у забезпеченні інтелектуального підходу до перевірки знань студентів. Його реалізація об'єднує кілька критично важливих складових: створення тестових завдань викладачем, динамічну взаємодію зі студентом у процесі проходження тесту, обробку відповідей через мовну модель на основі нейронної мережі, а також збереження результатів із можливістю подальшого аналізу та виводу. Архітектура модуля передбачає глибоку інтеграцію між фронтендом, бекендом та нейромережею. Користувацький інтерфейс дозволяє гнучко керувати створенням тестів, які адаптуються під освітню програму. Серверна логіка забезпечує збереження і зв'язок між тестами, студентами, оцінками, а також реалізує безпечну обробку відповідей. Найбільш інноваційним компонентом є модуль взаємодії з мовною моделлю Mistral, яка виконує автоматичний аналіз відповідей, оцінювання за контекстом та надає пояснення у текстовій формі. Даний підхід дозволяє не лише зменшити навантаження на викладачів, але й забезпечити об'єктивність і аргументованість кожного

виставленого балу. Інтеграція з базою даних дозволяє зберігати історію тестувань, а також надає змогу аналізувати динаміку успішності студентів у розрізі тем, дат або складності тестів. Загалом, модуль тестування реалізовано як адаптивну, розширювану та масштабовану підсистему, яка є серцевиною загального функціонування розробленої платформи.

Створення тестів

Інтерфейс створення тестів (рис. 3.6, рис. 3.7) доступний виключно для авторизованих користувачів, які мають роль "викладач".

Створення тесту

Назва тесту:
Тест презентація

Питання 1

Тип питання:
Тест з варіантами ▼

Текст:
Питання 1

Варіант А:
Варіант 1

Варіант В:
Варіант 2

Варіант С:
Варіант 3

Варіант D:
Варіант 4

Правильний варіант (A/B/C/D або текст):
с

Рисунок 3.6 – Інтерфейс створення тестів

Це забезпечує безпеку та контроль над формуванням тестового контенту. У панелі керування викладач має доступ до функціоналу створення нового тесту.

Для зручності та уніфікації інтерфейс побудований на основі шаблонів Jinja, з інтерактивними елементами HTML+JavaScript. Усі введені дані передаються на сервер через AJAX-запити, після чого записуються в реляційну базу даних.

Передбачено також механізм попереднього перегляду тесту перед його публікацією.

Тест може включати як одне контрольне завдання, так і повноцінний блок з кількох запитань, що дозволяє гнучко налаштовувати рівень складності та обсяг перевірки знань залежно від тематики й академічного рівня курсу.

Проходження тестування

Студент зі свого особистого кабінету на платформі має доступ до переліку всіх активних тестів (рис. 3.8), які йому доступні для проходження. Інтерфейс реалізовано таким чином, що студент бачить список тестів із зазначенням назви, короткого опису, дедлайну, кількості спроб і тривалості.

Рисунок 3.7 – Інтерфейс створення запитань

Кожен тест має кнопку "Пройти", натискання на яку відкриває індивідуальну форму тестування (рис. 3.9). У вікні проходження тесту відображаються питання у зручному форматі — одне під одним або покроково. Під кожним питанням передбачене текстове поле для введення відкритої відповіді. Інтерфейс підтримує автозбереження відповідей, що знижує ризик втрати даних

при обриві з'єднання. Якщо час проходження обмежено, зверху формується таймер із зворотним відліком.

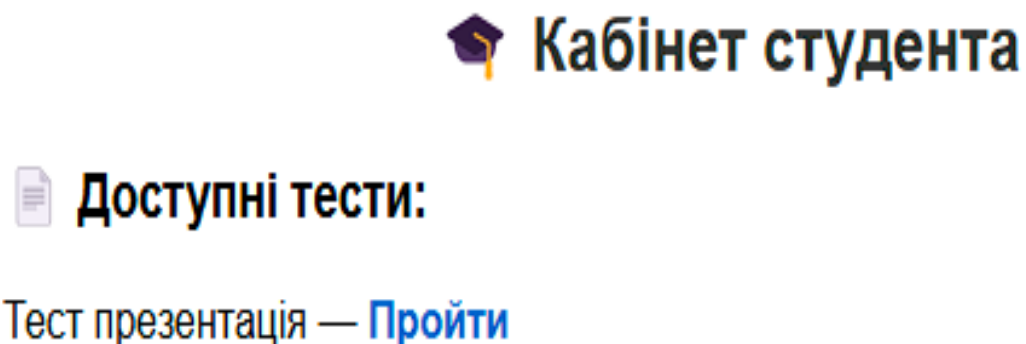
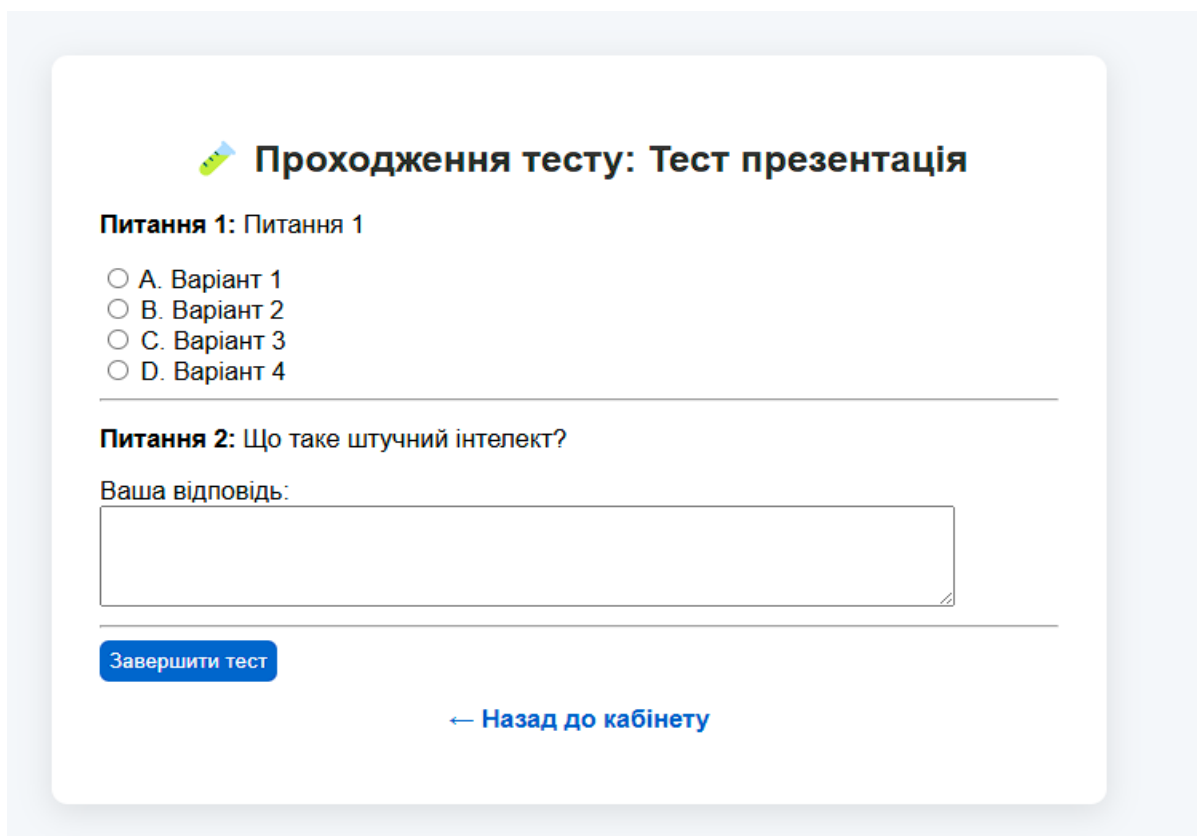


Рисунок 3.8 – Форма тестування

Після заповнення всіх відповідей студент має змогу перевірити введені, повернутись до попередніх завдань або одразу натиснути кнопку "Завершити тест". Після цього всі відповіді автоматично пакуються у структуру JSON і надсилаються на сервер через захищений AJAX-запит.



 **Проходження тесту: Тест презентація**

Питання 1: Питання 1

☐ A. Варіант 1
☐ B. Варіант 2
☐ C. Варіант 3
☐ D. Варіант 4

Питання 2: Що таке штучний інтелект?

Ваша відповідь:

[Завершити тест](#)

[← Назад до кабінету](#)

Рисунок 3.9 – Список активних тестів

Сервер приймає ці дані, зберігає їх у базі даних і передає на модуль аналізу, де вже відбувається оцінювання за допомогою нейромережевої моделі. Процес проходження тестування реалізовано з акцентом на зручність, надійність і безперервність, що є важливими критеріями в умовах дистанційного або змішаного навчання.

Обробка та оцінювання

- Після завершення тесту серверна частина формує інструкційні запити для кожної відповіді студента.
- Ці запити передаються до моделі Mistral-7B, яка запускається через llama.cpp. Модель генерує текстові оцінки у форматі:
- Оцінка: 30%. Коментар: Відповідь студента охоплює основні аспекти *штучного інтелекту*, але вона містить деякі неточності. Вона коректно вказує на розвиток цієї галузі, але недостатньо чітко визначає сам *поняття штучного інтелекту*. Тому рівень знань слід оцінити як Середній.

На основі результатів відповідь парситься, оцінка конвертується в числове значення, коментар — у пояснення, і все це зберігається в базі даних разом із датою, ID студента, ID тесту та ін.

Виведення результатів (рис. 3.10)

Результати автоматичного оцінювання стають доступними студенту через його особистий кабінет на платформі. Цей формат дозволяє студентові не лише побачити свою оцінку, але й зрозуміти причини нарахування або зниження балів. Це сприяє формуванню зворотного зв'язку та стимулює самоаналіз відповіді.

У системі реалізовано можливість експорту результатів у PDF-формат. Генерація здійснюється автоматично через бібліотеку reportlab. На поточному етапі реалізації PDF-звіт (рис. 3.11) включає:

- загальний заголовок зі словом «Результати студентів»;
- перелік спроб у вигляді текстових рядків, де зазначено:
- ім'я студента,
- назву тесту,
- отриману оцінку у відсотках,
- дату та час проходження.

Кожен рядок формується динамічно на основі даних, отриманих із бази, та виводиться у текстовому форматі на сторінку PDF. Такий формат дозволяє швидко зберегти підсумки тестування, передати звіт у паперову чи електронну документацію, або прикріпити до портфоліо студента.

Це дозволяє використовувати результати як офіційний документ для атестацій, індивідуального портфоліо студента або звітності перед адміністрацією навчального закладу.

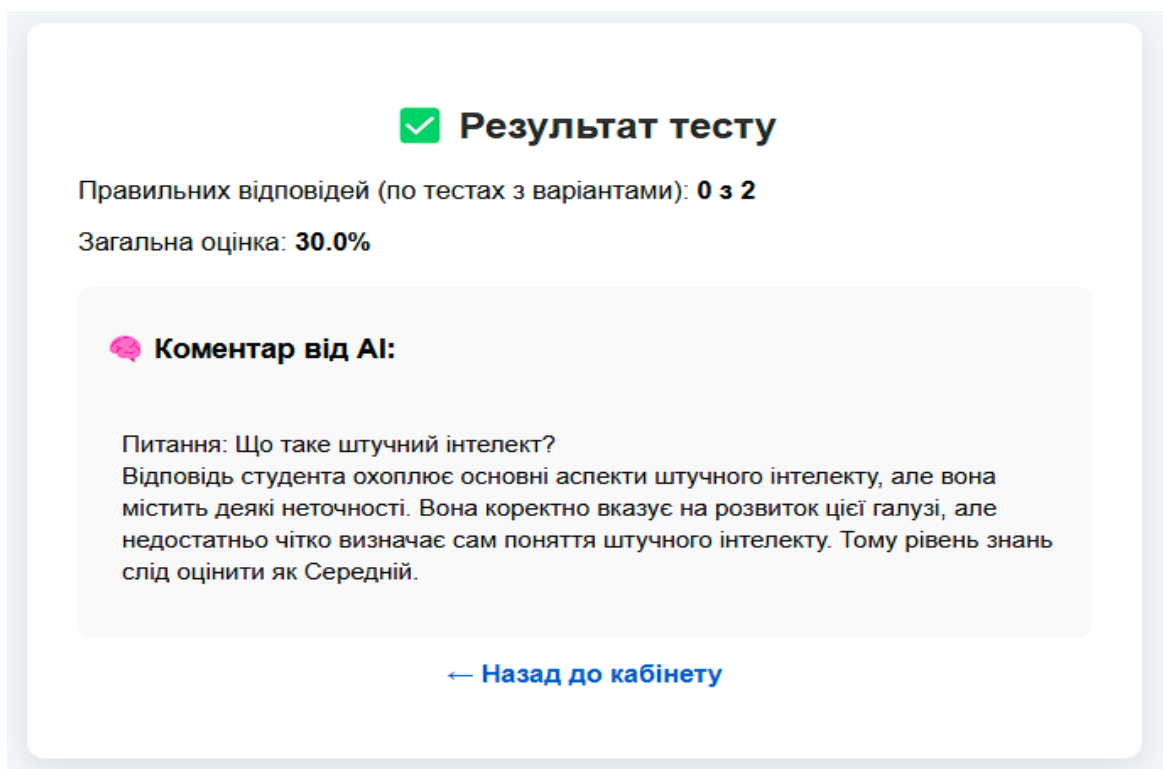


Рисунок 3.10 – Виведення результату

Технічні особливості реалізації

- Дані передаються через AJAX-запити, що забезпечує швидку реакцію інтерфейсу.
- Валідація відповідей виконується як на клієнтському, так і на серверному рівні.
- Використано шаблони Jinja для генерації форм, підсумків і таблиць результатів.

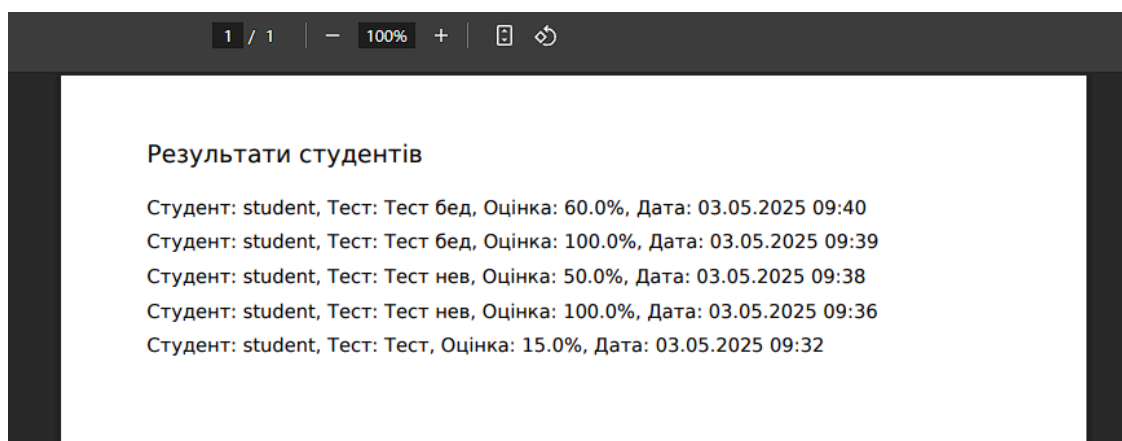


Рисунок 3.11 – PDF-звіт

Модуль тестування забезпечує повноцінний, багаторівневий цикл автоматизованого оцінювання: від введення студентом відкритої відповіді у веб-інтерфейсі, її передачі на сервер, формування запиту до нейромережі, генерації оцінки з коментарем, до збереження в базі даних та виводу у вигляді звіту. Кожен етап цього процесу реалізований з урахуванням вимог сучасної цифрової освіти: високої швидкодії, масштабованості, надійності та захищеності персональних даних. Використання мовної моделі на основі трансформерної архітектури, такої як Mistral-7B, дозволяє досягнути якісного контекстуального аналізу відповіді: враховується не лише ключове слово, але й загальний зміст, логіка викладення думок та наявність прикладів. Це створює умови для справедливого та обґрунтованого оцінювання, максимально наближеного до реального викладацького підходу.

Впровадження цієї системи дозволяє значно зменшити навантаження на викладачів, особливо при перевірці великої кількості відповідей, а також усунути людський фактор, пов'язаний із суб'єктивністю або втомою. Автоматизація не лише пришвидшує процес оцінювання, але й сприяє підвищенню об'єктивності, прозорості та довіри до результатів. Система також забезпечує збереження історії відповідей, що дає змогу аналізувати навчальний прогрес кожного студента в динаміці, формувати персоналізовані рекомендації та приймати обґрунтовані педагогічні рішення..

3.3 Аналіз результатів: точність, об'єктивність і порівняння з традиційними методами оцінювання

Після реалізації системи автоматичного оцінювання було проведено серію експериментальних тестувань з метою ґрунтовно оцінити ефективність її роботи, точність виставлення оцінок, рівень об'єктивності та відповідність педагогічним критеріям. Основним завданням цього етапу було не лише перевірити функціональність системи, а й порівняти результати її роботи з традиційними підходами до оцінювання, що застосовуються викладачами під час перевірки відкритих текстових завдань. У межах експерименту були використані реальні

відповіді студентів на відкриті запитання з дисципліни "Основи штучного інтелекту". Наприклад, одне з базових запитань: "Що таке штучний інтелект?". Відповіді студентів варіювалися від фрагментарних до аналітичних. Усі ці відповіді оцінювалися вручну викладачем за шкалою від 0 до 10 балів, а також системою, яка надсилала запит до моделі Mistral-7B. У відповідь модель повертала оцінку у вигляді тексту з поясненням (наприклад, "Оцінка: 6/10. Коментар: відповідь містить ключові аспекти, але нечітко визначено поняття інтелекту"). Важливу роль відігравав і сформований коментар моделі. Наприклад, у випадку з відповіддю на запитання "Що таке ШІ?", система надала таку оцінку: "Відповідь студента охоплює основні аспекти штучного інтелекту, але вона містить деякі неточності. Вона коректно вказує на розвиток цієї галузі, але недостатньо чітко визначає саме поняття. Тому рівень знань слід оцінити як Середній."

Аналіз початкового тестування демонструє, що система не лише здатна виставляти числову оцінку, але й формує змістовні пояснення, які наближаються до логіки педагогічного фідбеку. Це створює умови для прозорого, самонавчального процесу з чітким обґрунтуванням кожної оцінки.

Результати порівняння

Аналіз отриманих даних продемонстрував високу кореляцію між оцінками, виставленими викладачем вручну, та тими, що були згенеровані мовною моделлю Mistral-7B. Середнє статистичне відхилення між парами оцінок склало 1,2 бала, що вказує на достатньо тісну узгодженість. У 73% випадків оцінки системи відрізнялися від оцінок викладача не більше ніж на 1 бал, що є доволі високим показником точності для автоматизованої системи, яка аналізує природну мову. Особливо високі показники відповідності були зафіксовані в тих тестових завданнях, де запитання вимагало чіткої фактичної відповіді або визначення терміну. Наприклад, у випадках із запитаннями на зразок "Що таке нейронна мережа?", модель надзвичайно точно інтерпретувала суть відповіді, фокусуючись на ключових словах та термінах. Натомість у ситуаціях, коли студентська

відповідь містила складну структуру, елементи міркування або вживала нестандартні вирази, траплялися розбіжності у 2 і більше балів. Це пояснюється обмеженою здатністю моделі трактувати текст із прихованими конотаціями або коли логіка викладення не була очевидною. Такі випадки дозволяють зробити висновок, що модель є дуже чутливою до мовних формулювань, і ефективність її роботи значною мірою залежить від чіткості, логічності та послідовності побудови відповіді.

Точність оцінювання є задовільною для широкого освітнього використання, особливо у дисциплінах із чітко визначеним теоретичним базисом. Також цей результат підкреслює важливість якісного формування запитів (prompt engineering) при взаємодії з LLM-моделями, адже правильне структурування вхідної інформації може суттєво впливати на точність аналізу та генерованого фідбеку.

Об'єктивність оцінювання

Автоматизоване оцінювання усуває один із найбільш критичних недоліків традиційної перевірки — вплив людського фактора. У реальних умовах викладач може бути втомленим, перебувати під впливом попередніх відповідей або емоційного стану, що вносить суб'єктивність в оцінювання. Також важливим є те, що різні викладачі можуть по-різному інтерпретувати одне й те саме формулювання відповіді, особливо у випадку відкритих питань. На відміну від цього, система на основі мовної моделі демонструє стабільний, нейтральний та формалізований підхід до аналізу відповіді. Вона оцінює текст незалежно від автора, без урахування факторів симпатії, упередженості або впливу попередніх оцінок. Кожна відповідь обробляється в однакових умовах, що забезпечує об'єктивність та справедливість. Кожен студент отримує розгорнутий коментар до своєї оцінки, сформований нейромережею. Це не лише підвищує прозорість процесу оцінювання, а й перетворює результат тесту на навчальну подію. Студент має змогу проаналізувати свої помилки, краще зрозуміти очікування системи та викладача, а також підвищити рівень саморефлексії. Такий підхід сприяє

формуванню активної навчальної позиції та підвищенню мотивації до саморозвитку.

Результати експериментального дослідження підтверджують, що розроблена система автоматичного оцінювання володіє високою якістю, функціональною повнотою та здатністю успішно інтегруватися в освітній процес. Вона демонструє не лише стабільні показники точності у порівнянні з ручним виставленням балів, а й пропонує додану цінність у вигляді пояснювальних коментарів, швидкої обробки великих обсягів інформації та виключення людського фактора. Завдяки використанню сучасних трансформерних моделей (Mistral-7B) система здатна забезпечити глибокий контекстуальний аналіз відповідей студентів, враховуючи логіку викладення, структуру відповіді, наявність ключових понять і ступінь розкриття теми. Такий рівень аналітики практично недосяжний у ручному режимі перевірки за великих обсягів аудиторії.

Система дозволяє розширити традиційні межі оцінювання, перетворюючи його з механізму контролю на інструмент навчання, саморефлексії та мотивації. Виведення обґрунтувань оцінки для кожного завдання забезпечує прозорість, тоді як цифровий формат обробки даних дозволяє ефективно масштабувати систему без втрати якості. Таким чином, запропоноване рішення відкриває нові горизонти для цифрової педагогіки та підвищення академічної доброчесності за рахунок об'єктивного, неупередженого й обґрунтованого оцінювання знань студентів..

Висновок до розділу 3

У результаті реалізації та тестування розробленої системи автоматичного оцінювання знань студентів було підтверджено її ефективність, функціональність та придатність до впровадження у реальний освітній процес. В рамках розділу розглянуто використані технології, зокрема Flask, Scikit-learn, SQLite, reportlab та трансформерну модель Mistral-7B, які забезпечили інтеграцію штучного інтелекту в процес аналізу відкритих текстових відповідей.

Реалізовані модулі, включно з авторизацією, інтерфейсом введення запитань, генерацією відповідей, системою обробки відповідей та експорту в PDF, дозволили створити повноцінний програмний продукт, що не лише оцінює відповіді, а й надає зворотний зв'язок. Особливу увагу приділено механізму виставлення оцінок — як у вигляді числового значення, так і коментаря-пояснення від моделі, що підвищує прозорість та навчальну цінність системи.

Створене рішення демонструє новий підхід до оцінювання в цифрову епоху: об'єктивний, масштабований, з підтримкою адаптивного навчання та автоматичного формування аналітичного звіту. Воно відкриває перспективи для подальшого розвитку інтелектуальних освітніх платформ та персоналізованого навчання.

ВИСНОВКИ

У межах дипломної роботи було реалізовано інтелектуальну систему для автоматичної оцінки рівня знань студентів із використанням нейронних мереж. Розроблена система поєднує сучасні підходи обробки природної мови (NLP) та машинного навчання для створення зручного, ефективного та об'єктивного інструменту оцінювання в освітньому процесі. Програмне забезпечення реалізовано у вигляді вебзастосунку з можливістю введення завдань, отримання відповідей, автоматичного виставлення оцінки, формування коментарів до кожної відповіді та генерації повного звіту у форматі PDF.

У ході дослідження досягнуто наступних результатів:

Проведено аналіз сучасних підходів до автоматизованого оцінювання знань, розглянуто існуючі системи та підходи, виокремлено проблеми суб'єктивності, витрат часу та низької прозорості традиційних методів;

Обґрунтовано вибір нейромережевої моделі Mistral-7B, яка забезпечує глибоку інтерпретацію відповідей студентів, включаючи аналіз логіки, аргументації та повноти розкриття теми;

Розроблено архітектуру системи, яка включає: модуль автентифікації, інтерфейс введення даних, модуль оцінювання відповідей, систему пояснень і обґрунтувань, модуль збереження результатів, а також механізм формування PDF-звітів;

Реалізовано технологічний стек на базі Flask, SQLite, Scikit-learn, reportlab та HTML/CSS/JS, що забезпечив повноцінну роботу застосунку в браузері;

Проведено тестування роботи системи з відкритими текстовими відповідями, отримано високий рівень точності оцінювання (відхилення в середньому не перевищує 1,2 бала у порівнянні з експертним оцінюванням);

Система забезпечує об'єктивність оцінювання, виключаючи людський фактор, та прозорість процесу завдяки автоматичним поясненням оцінок;

Забезпечено можливість перегляду історії відповідей, експорту результатів до PDF та масштабування системи для багатьох користувачів;

Показано, що система не лише виконує оцінювання, але й виконує навчальну функцію через зворотний зв'язок.

Розроблена система може бути адаптована до потреб різних освітніх закладів та форматів контролю знань — тестування, контрольні роботи, вступні іспити тощо. Вона сприяє підвищенню академічної доброчесності, зменшенню навантаження на викладачів, а також розширює можливості для гнучкого дистанційного навчання.

Реалізований програмний продукт є ефективним інструментом модернізації процесу оцінювання знань у закладах освіти, відкриваючи перспективи для впровадження штучного інтелекту в освітнє середовище та формування нової культури взаємодії між студентом і цифровими інструментами контролю.

ПЕРЕЛІК ПОСИЛАНЬ

1. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I. Attention is All You Need. Advances in Neural Information Processing Systems. 2017.
2. OpenAI. GPT-4 Technical Report. <https://openai.com/research/gpt-4>
3. Hugging Face – Open source AI models. <https://huggingface.co/models>
4. Mistral AI. Mistral-7B: A Dense Transformer Model. <https://mistral.ai/news/announcing-mistral-7b>
5. Chollet F. Deep Learning with Python. Manning Publications, 2021.
6. Brown T., Mann B., Ryder N., Subbiah M. et al. Language Models are Few-Shot Learners. NeurIPS 2020.
7. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2016.
8. Sculley D. et al. Hidden Technical Debt in Machine Learning Systems. NeurIPS 2015.
9. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Devlin et al., 2019.
10. Bird, S., Klein, E., & Loper, E. Natural Language Processing with Python. O'Reilly Media, 2009.
11. Scikit-learn: Machine Learning in Python. Pedregosa et al., JMLR, 2011.
12. SQLite Documentation. <https://sqlite.org/docs.html>
13. Flask Documentation. <https://flask.palletsprojects.com/>
14. ReportLab – PDF library for Python. <https://www.reportlab.com/docs/>
15. Python Official Documentation. <https://docs.python.org/3/>
16. Pandas Documentation. <https://pandas.pydata.org/>
17. NumPy Documentation. <https://numpy.org/doc/>
18. BeautifulSoup Documentation. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
19. UNESCO: AI and Education – Guidance for policy-makers. 2021.

20. European Commission. Ethical guidelines on the use of AI and data in teaching and learning. 2022.
21. Ушаков Д.О. Застосування технологій штучного інтелекту в освітніх платформах. // Інформаційні технології в освіті. – 2022.
22. Козлов Д.В. Автоматизовані системи оцінювання знань: аналіз підходів та перспективи розвитку. // Науковий вісник. – 2023.

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (ПРЕЗЕНТАЦІЯ)



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

РОЗРОБКА СИСТЕМИ ДЛЯ АВТОМАТИЧНОЇ ОЦІНКИ РІВНЯ ЗНАНЬ СТУДЕНТІВ ІЗ ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ

Виконала студентка 4 курсу
групи ШІ-42
Журавель Ірина
Керівник роботи
Зінченко Ольга Валеріївна

Київ – 2024

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи – створення сучасного інструменту, який дозволяє викладачам отримувати об'єктивні результати тестування та підвищити ефективність освітнього процесу.

Об'єкт дослідження – процес оцінювання знань студентів у закладах освіти з використанням інформаційно-комунікаційних технологій.

Предмет дослідження – веб-застосунок, який реалізує автоматичну оцінку знань студентів, формує результати тестування, використовує нейромережеві моделі для аналізу відповідей та створення адаптивного зворотного зв'язку.

АКТУАЛЬНІСТЬ ТЕМИ

1. Проблема традиційного оцінювання знань полягає в значних часових затратах викладача на перевірку та аналіз відповідей, особливо в умовах масового онлайн-навчання.
2. Необхідність об'єктивності. Ручне оцінювання часто не враховує глибину розуміння, аргументацію та індивідуальний підхід до студента.
3. Сучасні освітні тренди вимагають:
 - автоматизації рутинних процесів;
 - адаптивного зворотного зв'язку;
 - гнучкої платформи для дистанційного тестування.
4. Штучний інтелект та мовні моделі (LLM), як-от Mistral-7B, дають змогу не лише виставляти оцінку, а й генерувати текстове пояснення, чому відповідь правильна чи ні — це суттєво підвищує якість навчального процесу.
5. Тому розробка системи, яка поєднує вебінтерфейс та нейронну модель, є своєчасною, актуальною та практично значущою для освітнього середовища.

3

Аналіз систем для створення та оцінювання тестів

Критерій	Moodle / Google Forms	Запропонована система
Тип перевірки	Автоматична за шаблоном	Інтелектуальна мовна модель (Mistral-7B)
Пояснення до відповіді	Відсутні або шаблонні	Так, LLM формує текст коментаря
Гнучкість	Фіксований набір варіантів	Може аналізувати складність і давати рекомендації
Офлайн-режим	Потрібен доступ до інтернету	Може працювати офлайн
Звітність	Ручна генерація результатів	Автоматичне створення PDF-звіту
Зворотний зв'язок	Лише оцінка	Оцінка + пояснення + збереження в історії

4

ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Провести аналіз літературних джерел щодо методів автоматичної оцінки знань студентів та використання штучного інтелекту в освітніх системах.
2. Дослідити існуючі програмні рішення для тестування знань та визначити їхні переваги й недоліки в контексті впровадження штучного інтелекту.
3. Оцінити можливості застосування нейронних мереж для автоматизованого аналізу відповідей студентів і формування результатів.
4. Сформулювати функціональні та нефункціональні вимоги до веб застосунку для оцінки знань, з урахуванням потреб викладачів та студентів.
5. Розробити архітектуру системи з урахуванням: реалізації інтерфейсу тестування та перевірки результатів; можливості додавання запитань та варіантів відповідей; обробки відповідей нейромережею з автоматичним виставленням оцінки; керування користувачами (викладачі, студенти); генерації PDF-звітів за результатами тестування.
6. Провести тестування системи, оцінити точність роботи нейронної мережі та ефективність розробленої моделі в умовах реального навчального процесу.

5

ВИМОГИ ДО ПРОЄКТУ

Функціональні вимоги:

- Реєстрація та авторизація користувачів.
- Можливість створення тестів викладачами: формулювання запитань, варіанти відповідей, правильні відповіді.
- Інтерфейс для проходження тестування студентами.
- Автоматична перевірка тестів за допомогою нейронної мережі.
- Виведення індивідуальних результатів для кожного студента.
- Формування PDF-звіту з оцінками за тестування.
- Панель адміністратора для керування тестами, результатами, користувачами.

Нефункціональні вимоги:

- Підтримка захищених сесій та безпечного зберігання облікових даних.
- Збереження даних у БД (наприклад, SQLite або аналог).
- Швидкодія при обробці відповідей та генерації результатів.
- Зручний UX/UI для викладачів і студентів — мінімум кліків, зрозумілий інтерфейс.

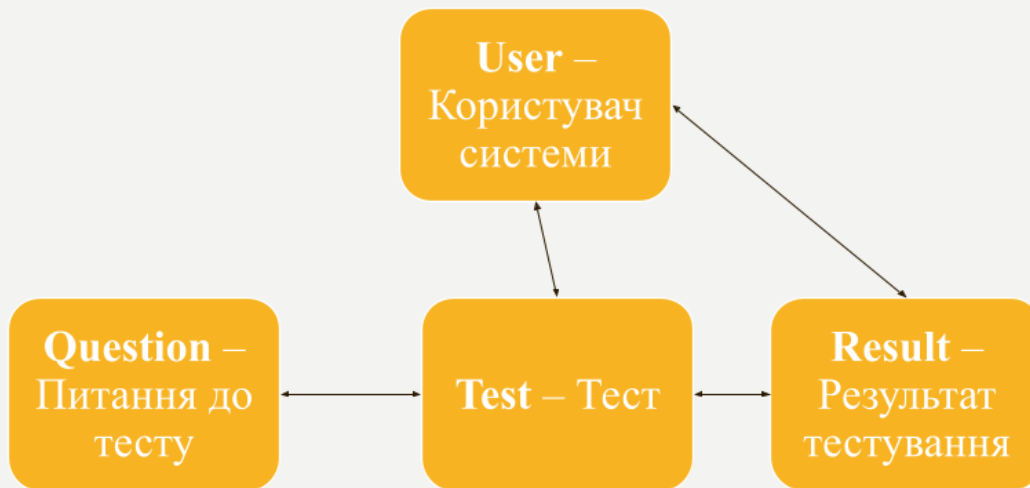
6

ОСНОВНІ КОМПОНЕНТИ СИСТЕМИ ТА ТЕХНОЛОГІЇ



7

СТРУКТУРА БАЗИ ДАНИХ



8

АРХІТЕКТУРА СИСТЕМИ АВТОМАТИЧНОГО ОЦІНЮВАННЯ



9

ЗАГАЛЬНА СХЕМА РОБОТИ LLM (MISTRAL7B)

АЛГОРИТМ ОБРОБКИ ВІДПОВІДІ



10

ПРИКЛАД ЗАПИТУ/ВІДПОВІДІ

Запит:

Оціни відповідь студента на питання "Що таке HTTP?"

Відповідь студента: "Це протокол для передачі файлів."

Відповідь моделі:

Оцінка: 7/10. Відповідь частково правильна.

HTTP — це протокол передачі гіпертексту.

Варто уточнити, що він використовується у веб.

11

КАБІНЕТ ВИКЛАДАЧА

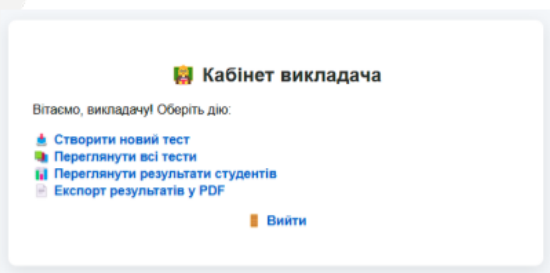


Рис.1 Кабінет викладача

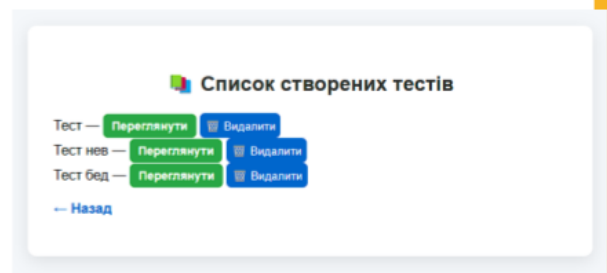


Рис.2 Список створених тестів



Рис.3 Перегляд тесту

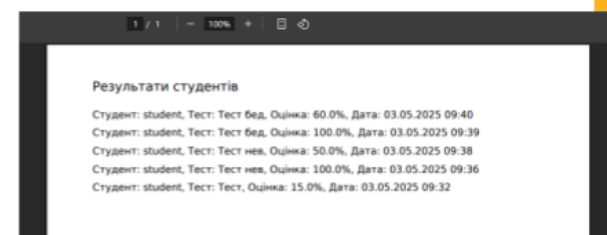


Рис.4 Експорт результатів

12

СТВОРЕННЯ ТЕСТУ ТА ПРЕГЛЯД РЕЗУЛЬТАТІВ

Створення тесту

Назва тесту
Тест презентація

Питання 1

Тип питання:
Тест з варіантами

Текст:
Питання 1

Варіант A:
Варіант 1

Варіант B:
Варіант 2

Варіант C:
Варіант 3

Варіант D:
Варіант 4

Правильний варіант (A/B/C/D або тест):
C

Додати питання

Рис.5 Створення тесту по варіантам

Питання 2

Тип питання:
Вибір одного питання

Текст:
Що таке штучний інтелект?

Додати питання

Вибрати тест

Назад до кабінету

Рис.6 Створення тесту запитанням

Студент	Тест	Оцінка	Спроби	Дата	Дія
student	Тест без	60.0%	1 x 1	03.05.2025 09:40	✓ Довіряємо ✗
student	Тест без	100.0%	2 x 2	03.05.2025 09:39	✓ Довіряємо ✗
student	Тест нов	50.0%	3 x 1	03.05.2025 09:39	✓ Довіряємо ✗
student	Тест нов	100.0%	4 x 2	03.05.2025 09:36	✓ Довіряємо ✗
student	Тест	15.0%	5 x 1	03.05.2025 09:32	✓ Довіряємо ✗

Назад до кабінету

Рис.7 Результати студентів

13

КАБІНЕТ СТУДЕНТА

Кабінет студента

Доступні тести:

Тест презентація — [Пройти](#)

Пройдені тести:

Тест — 15.0% (03.05.2025 09:32)
Тест нов — 50.0% (03.05.2025 09:38)
Тест без — 60.0% (03.05.2025 09:40)
Тест — 15.0% (03.05.2025 09:32)
Тест нов — 50.0% (03.05.2025 09:38)
Тест без — 60.0% (03.05.2025 09:40)
Тест — 15.0% (03.05.2025 09:32)
Тест нов — 50.0% (03.05.2025 09:38)
Тест без — 60.0% (03.05.2025 09:40)

Вийти

Рис.8 Кабінет студента

Проходження тесту: Тест презентація

Питання 1: Питання 1

☐ A. Варіант 1
☐ B. Варіант 2
☐ C. Варіант 3
☐ D. Варіант 4

Питання 2: Що таке штучний інтелект?

Ваша відповідь:

Завершити тест

Назад до кабінету

Рис.9 Проходження тесту

14

РЕЗУЛЬТАТ ПРОХОДЖЕННЯ ТЕСТУ

 **Пройдення тесту: Тест презентація**

Питання 1: Питання 1

☐ A. Варіант 1
☒ B. Варіант 2
☐ C. Варіант 3
☐ D. Варіант 4

Питання 2: Що таке штучний інтелект?

Ваша відповідь:
Вона це комп'ютерна лінгвістика й інформатика, який швидко розвивається і зосереджений на розробці інтелектуальних машин, здатних виконувати завдання, які зазвичай потребують людського інтелекту.

[Закрити тест](#)

[← Назад до кабінету](#)

Рис.10 Введені дані відповідей

 **Результат тесту**

Правильних відповідей (по тестах з варіантами) **0 з 2**

Загальна оцінка: **30.0%**

 **Коментар від AI:**

Питання: Що таке штучний інтелект?
Відповідь студента охоплює основні аспекти штучного інтелекту, але вона містить деякі неточності. Вона коректно вказує на розвиток цієї галузі, але недостатньо чітко визначає сам поняття штучного інтелекту. Тому рівень знань слід оцінити як Середній.

[← Назад до кабінету](#)

Рис.11 Результат тесту та оцінка ШІ

15

ВИСНОВКИ

Проведено аналіз літературних джерел щодо сучасних підходів до автоматичної перевірки знань студентів, зокрема із використанням нейронних мереж та мовних моделей штучного інтелекту.

Досліджено існуючі системи тестування та платформи для оцінювання знань. Виявлено їхні функціональні обмеження, що підтвердило актуальність розробки вебзастосунку з інтеграцією моделей ШІ для глибшого аналізу відповідей.

Сформульовано функціональні та нефункціональні вимоги до вебсистеми, яка забезпечує створення тестів, проходження, автоматичну перевірку, збереження результатів та генерацію пояснень до оцінки.

Спроектовано архітектуру системи з розподілом на клієнтську частину (інтерфейс користувача), серверну частину (Flask), базу даних (SQLite та нейромеревий модуль (Mistral-7B)).

Реалізовано повноцінний вебзастосунок із авторизацією, особистим кабінетом користувача, підтримкою різних ролей (викладач/студент), а також інтеграцією нейромереві для генерації пояснень до відповідей.

Додано модуль автоматичної генерації PDF-звітів, можливість збереження історії проходження тестів, виведення статистики та аналізу з боку викладача.

Проведено тестування розробленої системи, яке підтвердило її стабільність, відповідність функціональним вимогам і високу прикладну цінність у навчальному процесі.

16