

**Тягунова М. Ю.**, кандидат технічних наук, доцент,  
доцент кафедри комп'ютерних систем та мереж  
Національного університету «Запорізька політехніка»  
ORCID: 0000-0002-9166-5897

**Киричек Г. Г.**, кандидат технічних наук, доцент,  
доцент кафедри комп'ютерних систем та мереж  
Національного університету «Запорізька політехніка»  
ORCID: 0000-0002-0405-7122

**Костецький Д. В.**, студент кафедри комп'ютерних систем  
та мереж  
Національного університету «Запорізька політехніка»  
ORCID: 0009-0001-7020-942X

### РОЗРАХУНОК ЕКОНОМІЇ ЧАСУ ПРИ СТВОРЕННІ ТЕСТУ ЗА ДОПОМОГОЮ ШТУЧНОГО ІНТЕЛЕКТУ

Дистанційне навчання розвивається вже багато десятиліть, але найбільшу увагу було привернуто до нього з початку поширення пандемії COVID-19, коли у всьому світі було введено карантинні обмеження. Завдяки цьому, процес інтеграції цього типу навчання у звичне очне було прискорено [1]. Адже саме дистанційне навчання, завдяки його гнучкості, допомогло продовжити виконувати навчальний план та навчатись майже за графіком, який створюєш сам собі. Але перелічуючи переваги дистанційного навчання, в нього також є і негативні впливи на організм, а саме: погіршення зору, постави, психологічні аспекти.

Дистанційне навчання має багато переваг над звичайним відвідуванням аудиторій тим, що вивчати матеріали можна будь-де, а якщо щось було не зрозуміло, то можна переглянути його ще раз, також лекції й завдання можна завантажити на пристрій і тим самим навчатися без світла або інтернету [2].

Саме під час дистанційного навчання опитування у вигляді тестування стає все більш розповсюдженим та актуальним. Розрахунок економії часу при створенні тестів з використанням штучного інтелекту є основним аспектом даної роботи. Використовуючи алгоритми штучного інтелекту, було оптимізовано процес створення тестів, зменшуючи час і зусилля, необхідні викладачам. Це передбачає розробку математичної моделі для кількісної оцінки часу, збереженого при використанні штучного інтелекту порівняно з традиційними методами створення тестів вручну. Результати цього розрахунку дають цінну інформацію про підвищення ефективності, яку пропонує штучний інтелект в освітніх установах, що, зрештою, сприятиме вдосконаленню технологічних методологій навчання.

У даній роботі розроблено математичну модель розрахунку економії часу при створенні тестів за допомогою штучного інтелекту у порівнянні із власноручною розробкою тестів викладачем. Проведене дослідження показало, що з таким підходом можна досягти економії часу до 62,5%.

Також у роботі наведено реалізацію створення тестів обома запропонованими варіантами (зі штучним інтелектом та без нього) у системі, яка реалізована з використанням фреймворку Laravel.

Отримані результати підтверджують доцільність та ефективність використання ШІ при розробці тестів.

Ключові слова: дистанційне навчання, тестування, процес створення тестів, штучний інтелект.

**Tiahunova M. Yu., Kyrychek H. H., Kostetskyi D. V. Time savings calculation in test creation using Artificial Intelligence**

Distance learning has been evolving for many decades, but it garnered significant attention with the onset of the COVID-19 pandemic when quarantine restrictions were implemented worldwide. As a result, the integration of this type of learning into traditional face-to-face education was expedited. Distance learning, with its flexibility, allowed for the continuation of the curriculum and learning almost on a self-paced schedule. However, while listing the advantages of distance learning, it also has negative impacts on health, such as deteriorating eyesight, posture, and psychological aspects.

Distance learning offers numerous advantages over traditional classroom attendance, as materials can be studied anywhere, and if something is not understood, it can be reviewed again. Additionally, lectures and assignments can be downloaded to devices, allowing learning without the need for light or the internet.

During distance learning, testing in the form of quizzes becomes increasingly common and relevant. The time savings calculation in test creation using artificial intelligence is an important aspect of our study. By leveraging AI algorithms, we aim to streamline the test creation process, reducing the time and effort required from instructors. This involves developing a mathematical model to quantify the amount of time saved when utilizing AI compared to traditional manual test creation methods. The results of this calculation will provide valuable insights into the efficiency gains offered by AI in educational settings, ultimately contributing to the advancement of technology-enhanced learning methodologies.

This study developed a mathematical model to calculate time savings in test creation using artificial intelligence compared to manual test creation by instructors. The research showed that this approach can achieve time savings of more than 60%.

The study also presents the implementation of test creation using both proposed methods (with and without AI) in a system developed using the Laravel framework. The results obtained confirm the feasibility and effectiveness of using AI in test development.

Key words: distance learning, testing, test creation process, Artificial Intelligence.

**Постановка проблеми.** Якщо розглядати навчання в загальному, то створення тестів є важливою частиною цього процесу, бо за допомогою їх можна зрозуміти, наскільки добре засвоєно матеріал студентами, і що потрібно пропрацювати більш детально. У звичному представленні для створення тесту викладачу потрібно витратити час на те, щоб створити питання і підібрати до нього варіанти відповідей. Цей процес, окрім витрати часу, також сильно втомлює. Але на заміну цьому, є кращий варіант – це генерування тестів за допомогою штучного інтелекту.

**Аналіз останніх досліджень та публікацій.** Провівши аналіз організації процесу навчання та особливо реалізації контролю знань під час дистанційної освіти [3-6], стає зрозумілим, що одним із найпоширеніших видів опитування у такій формі є саме тестування. Розробка тестів базується на різних методиках [7-10], кожна з яких має свої переваги та недоліки. Проте на заміну стандартних підходів, напрацьованих роками, у різних сферах життя все частіше застосовуються інструменти штучного інтелекту [11].

**Мета статті** – обґрунтування доцільності використання інструментів штучного інтелекту для створення тестів за рахунок розробки математичної моделі для розрахунку часу, витраченого на створення тесту з використанням штучного інтелекту і без нього.

**Виклад основного матеріалу.** Розглянемо різні варіанти створення тестів:

– коли викладач працює над тестом, він витрачає велику купу часу, бо йому потрібно, опираючись на текст його лекцій, розробити питання та варіанти відповідей до нього. При такому варіанті роботи, присутній індивідуальний підхід, бо викладач може регулювати складність питань і тест буде особливим. Проте цей процес також має певні обмеження у вигляді робочого часу, за який потрібно зробити велику купу іншої роботи, і його може не вистачити на написання якісного тесту, особливо коли у викладача велика кількість дисциплін;

– коли для створення тесту використовуються інструменти штучного інтелекту (ШІ), тоді викладачу потрібно лише написати коректну назву і докладно описати, що повинен перевірити тест: що буде створено, якої складності, і вказати кількість питань. Після цього тест буде згенеровано автоматично за лічені секунди, і після перевірки некоректні або нецікаві питання можна буде просто видалити.

У таблиці 1 показано порівняння цих двох підходів.

Таблиця 1

**Порівняння створення тестів двома підходами**

Варіант	Створення викладачем	Автоматична генерація
Час	Повільно	Швидко
Точність запитань	Залежить від кваліфікації викладача і уважності	Залежить від опису запиту
Складність створення	Залежить від досвіду викладача та його підготовки	Миттєва генерація, але потрібен контроль якості
Зворотній зв'язок	Детальний і контекстний зворотний зв'язок	Просто генерується без пояснень
Різноманітність запитань	Залежить від уяви викладача	Максимально унікальні питання
Інновації та технології	Залежить від інтересу викладача в нових методах і технологіях	Використовує передові технології

З таблиці видно, що як при створенні тестів самостійно викладачем, так і зі застосуванням інструментів штучного інтелекту є свої переваги та обмеження.

Розробимо математичну модель для розрахунку економії часу при створенні тесту викладачем з використанням ШІ і без.

При цьому зауважимо, що час, який витрачає викладач на те, щоб розробити тест, залежить від того, який у нього стаж і якої складності потрібно зробити тест. Тому будемо орієнтуватися на середньостатистичного викладача і розробку тесту середньої складності.

Припустимо, що  $T_B$  – це час створення одного запитання тесту викладачем, розрахований у хвиликах,  $N_{заг}$  – загальна кількість питань у тесті. Нехай час створення запиту для створення автоматичного тесту ШІ задамо як  $T_{зап}$ . При цьому вважатимемо, що  $T_{зап} = T_B$ . Нехай час, витрачений викладачем на перевірку автоматично створеного ШІ одного запитання задано  $T_{пер}$ . Припустимо, що  $T_{пер} = 0,2 * T_B$ . Нехай час, який буде витрачено викладачем на корегування цих запитань, задано  $T_{кор}$ . Припустимо, що  $T_{кор} = 0,5 * T_B$ , а кількість запитань, які потребують корегування,  $N_{кор}$  дорівнюватиме  $N_{кор} = 0,35 * N_{заг}$ .

У такому випадку загальний час, витрачений викладачем на розробку  $N_{заг}$  питань тесту самостійно без використання ШІ, розраховується за формулою 1:

$$T_{в_заг} = T_B * N_{заг} \quad (1)$$

А цей же час, витрачений на розробку  $N_{заг}$  питань тесту з використанням штучного інтелекту розраховується за формулою 2:

$$T_{\text{ші\_заг}} = T_{\text{зап}} + T_{\text{кор}} * N_{\text{кор}} + T_{\text{пер}} * N_{\text{заг}} \quad (2)$$

Оскільки час на генерацію запитань ШІ дуже малий, то не враховуємо це у загальному розрахунку.

Якщо загальний витрачений час  $N_{\text{заг}}$ , переписати з урахуванням наведених вище припущень, то отримаємо, що вираз за формулою 2 перетворюється у вираз за формулою 3:

$$T_{\text{ші\_заг}} = T_{\text{в}} + 0,2 * T_{\text{в}} * N_{\text{заг}} + 0,5 * T_{\text{в}} * 0,35 * N_{\text{заг}} \quad (3)$$

Спростивши вираз за формулою 3, отримаємо:

$$\begin{aligned} T_{\text{ші\_заг}} &= T_{\text{в}} + 0,2 * T_{\text{в}} * N_{\text{заг}} + 0,175 * T_{\text{в}} * N_{\text{заг}} = T_{\text{в}} + 0,375 * T_{\text{в}} * N_{\text{заг}} = \\ &= (1 + 0,375 * N_{\text{заг}}) * T_{\text{в}} \end{aligned} \quad (4)$$

Це означає, що чим більше буде створено завдань у тесті, тим меншим буде вплив часу на створення запиту до ШІ у загально витраченому часі для створення усіх завдань для тесту.

Розрахуємо відсоток зекономленого часу  $E$  при використанні ШІ для створення тестів за формулою 5.

$$E = \frac{T_{\text{в\_заг}} - T_{\text{ші\_заг}}}{T_{\text{в\_заг}}} * 100\% \quad (5)$$

Перетворимо формулу 5 з урахування формул 1 та 4, отримаємо результат, представлений формулою 6:

$$\begin{aligned} E &= \frac{T_{\text{в\_заг}} - T_{\text{ші\_заг}}}{T_{\text{в\_заг}}} * 100\% = \frac{T_{\text{в}} * N_{\text{заг}} - T_{\text{в}} + 0,375 * T_{\text{в}} * N_{\text{заг}}}{T_{\text{в}} * N_{\text{заг}}} * 100\% = \\ &= \left( 0,625 - \frac{1}{N_{\text{заг}}} \right) * 100\% \end{aligned} \quad (6)$$

Таким чином, із формули 6 видно, що чим більшою буде кількість запитань у тесті, тим ефективність застосування ШІ буде вище. При цьому у будь-якому разі вона не перевищуватиме 62,5%.

Розглянемо це на прикладі. Припустимо, що на створення одного запиту викладач витрачає в середньому 5 хвилин, і тест складається із 25 запитань. Тоді економія часу у відсотках при застосуванні ШІ складатиме  $(0,625 - 0,04) * 100\% = 58,5\%$ , що означає, що викладач витратить на створення тесту на 58,5% менше часу.

Перерахуємо ці ж значення за формулами 1 та 4. Час, який викладач витратив на створення 25 питань тесту самостійно, без використання штучного інтелекту дорівнюватиме  $T_{\text{в\_заг}} = 5 * 25 = 125$  хв.. Загальний час, витрачений на створення 25 питань тесту з використанням ШІ дорівнюватиме  $T_{\text{ші\_заг}} = 5 + 0,375 * 5 * 25 = 51,875$  хв. Тоді час, який зекономить викладач дорівнюватиме  $125 - 51,875 = 73,125$  хв., що і є 58,5% від витраченого часу без використання ШІ. На рисунку 4.13 за допомогою графіка зображено порівняння витрат часу цих двох підходів.

Таким чином, розроблена математична модель є адекватною. Слід також зазначити, що припущення, застосовані у даній моделі базуються на опитуванні викладачів ВНЗ спеціальності 123 Комп'ютерна інженерія.

Проведене практичне тестування функціоналу виявило, що система створила тести з ідеальною якістю, які не потребують жодних корекцій варіантів відповідей. Загалом, для генерації тестів та перевірки відповідей на 25 питань знадобилося всього близько 20 хвилин.

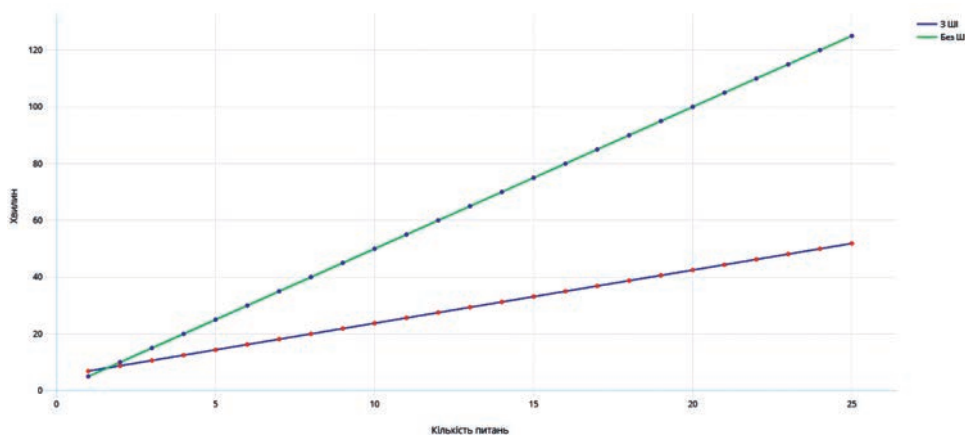


Рис. 1. Залежність витраченого часу від кількості питань

Реалізація системи, яка дозволяє створити тести викладачем або за допомогою ШІ виконана за допомогою фреймворка Laravel. На рисунках 2-3 зображено логіку, в якій виконується дві дії при створенні питання для тесту. В якості параметрів передається номер тесту, до якого додається питання, його назва,

а також варіанти відповідей у вигляді масиву. Першою дією створюється запис в базі даних, в якій зберігається назва питання, після чого береться його ідентифікатор, який використовується для ідентифікації при додаванні питань в базу даних.

```
3 usages  ± DimaK0st+1*
13 class QuestionService
14 {
15     no usages  ± DimaK0st
16     public function __construct(private VariantService $variantService, private QuestionRepository $questionRepository)
17     {
18     }
19     ± DimaK0st+1
20     public function create(CreateQuestionRequest $request)
21     {
22         $question = $this->questionRepository->create($request->getTestId(), $request->getTitle());
23         $this->variantService->create($request->getVariants(), $question->id);
24
25         return $question;
26     }
27 }
```

Рис. 2. Приклад логіки створення питання

```
13 class QuestionRepository
14 {
15     ± Dima Kostetskiy +1
16     public function create(int $testId, string $title)
17     {
18         $question = new Question();
19         $question->test_id = $testId;
20         $question->title = $title;
21         $question->save();
22
23         return $question;
24     }
25 }
26 }
27 }
28 }
29 }
30 }
31 }
32 }
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }
```

Рис. 3. Приклад створення питання

Приклад моделі питання відображено на рисунку 4. Зазвичай в Laravel моделі використовуються для обміну інформацією з базою даних. У попередніх прикладах було показано те, як в них передаються дані, й за допомогою методу save вони зберігаються. Але моделі також дозволяють визначати відносини між таблицями. Це означає, що при належному налаштуванні при виконанні запиту до бази даних варіанти відповідей автоматично додаються до результуючого об'єкту. Використання відносин також ефективно скорочує час, оскільки дані отримуються за допомогою одного запиту.

```
15 class Question extends Model
16 {
17     use HasFactory;
18
19     protected $with = ['variants'];
20
21     ± DimaK0st
22     public function test()
23     {
24         return $this->belongsTo(related: Test::class, foreignKey: 'test_id', ownerKey: 'id');
25     }
26     ± DimaK0st
27     public function variants()
28     {
29         return $this->hasMany(related: Variant::class, foreignKey: 'question_id', localKey: 'id');
30     }
31 }
```

Рис. 4. Приклад моделі питання

Наступним розглянемо створення тесту з використанням ШІ. Для цього було встановлено бібліотеку «jumbaeric/laragpt», яка має зрозумілу документацію та надає функції для легкої роботи з ChatGpt Api. На рисунку 5 зображено, як функція отримує дані від користувача, а саме назву тестування, його опис та кількість питань для генерування. Оскільки запит займає певний час, в Laravel можна зробити спеціальний запит, який виконується асинхронно, тобто користувач лише надсилає дані, і після того, як код їх опрацює, тест буде додано до курсу.

```
public function generateWithAi(CreateTestRequest $request)
{
    $createTestDto = CreateTestDto::fromRequest($request);

    GetDataFromAi::dispatch($createTestDto);

    return true;
}
```

Рис. 5. Логіка запуску асинхронного створення тесту за допомогою штучного інтелекту

На рисунку 6 зображено логіку асинхронної команди, яка задає початкові налаштування і описує те, як повинен функціонувати штучний інтелект, після генерується запит від користувача, в якому всі дані збираються в масив і передаються статичній функції бібліотеки, яка відправить запит на ChatGpt до API, і поверне результат.

```
public function handle()
{
    $result = Laragpt::chat([
        'model' => 'gpt-3.5-turbo', // required
        'messages' => [ // required
            [
                'role' => 'system',
                'content' => "Ви помічник, який створив тести за темою і описом, які передаються і повертаються у вигляді JSON а наступною структурою:
                (\\title*: \\*, \\description*: \\*, \\course_id*: 1, \\count*: 1, \\questions*:
                [\\question*: \\*, \\options*: [\\answer*: \\*, \\correct*: 0]])."
            ],
            [
                'role' => 'user',
                'content' => "Тема тесту - " . $this->createTestDto->title . ", опис - " .
                $this->createTestDto->description . ", кількість питань - " . $this->createTestDto->count . ", "
            ],
        ],
    ]);

    $questionsDto = new GeneratedAiQuestionsDto($result);

    return $this->testService->createFromAi($this->createTestDto, $questionsDto);
}
```

Рис. 6. Логіка асинхронної команди відправки запиту

Далі відповідь штучного інтелекту буде конвертовано за допомогою спеціальних класів DTO, які використовуються для зручної передачі даних між контролерами й сервісами проекту. На рисунку 7 зображено те, як інформацію спочатку представлено у вигляді звичайного рядка JSON. Ця відповідь передається в клас GeneratedAiQuestionsDto, де рядок розділяється на об'єкти, і на рисунку 8 зображено логіку перетворення цієї інформації. На рисунку 9 зображено результуючий об'єкт, який передається по проекту й з нього інформація зберігається в базу даних.

```
{ "questions": [ { "question": "Яка основна функція програми PocketTracer?", "options": [ { "answer": "Графічний дизайн", "correct": 0 }, { "answer": "Моделювання мереж", "correct": 1 }, { "answer": "Музичне створення", "correct": 0 }, { "answer": "Текстова переписка", "correct": 0 } ] }, { "question": "Яке призначення має PocketTracer?", "options": [ { "answer": "Фільмова монтаж", "correct": 0 }, { "answer": "Кулінарія", "correct": 0 }, { "answer": "Моделювання та симуляція мережі", "correct": 1 }, { "answer": "Малюнок на воді", "correct": 0 } ] }, { "question": "Які можливості надає програма PocketTracer?", "options": [ { "answer": "Спортивні прогнози", "correct": 0 }, { "answer": "Ігри віртуальної реальності", "correct": 0 }, { "answer": "Симуляція мережі та аналіз", "correct": 1 }, { "answer": "Геологічні дослідження", "correct": 0 } ] }, { "question": "Які дані можна аналізувати за допомогою PocketTracer?", "options": [ { "answer": "Тренди моди", "correct": 0 }, { "answer": "Економічні індикатори", "correct": 0 }, { "answer": "Поведінку мережі", "correct": 1 }, { "answer": "Астрономічні спостереження", "correct": 0 } ] }, { "question": "Які інструменти доступні в PocketTracer для редагування зображень?", "options": [ { "answer": "Мозаїка та колажі", "correct": 1 }, { "answer": "Графічні пензли", "correct": 0 }, { "answer": "Текстові стилі", "correct": 0 }, { "answer": "Музичні семпли", "correct": 0 } ] }, { "question": "Чому приділяється основна увага в PocketTracer?", "options": [ { "answer": "Ігрові фічі", "correct": 0 }, { "answer": "Дизайн інтерфейсу", "correct": 0 }, { "answer": "Моделювання мереж", "correct": 1 }, { "answer": "Формування новин", "correct": 0 } ] }, { "question": "Які області використання програми PocketTracer особливо популярні?", "options": [ { "answer": "Гастрономія", "correct": 0 }, { "answer": "Медицина", "correct": 0 }, { "answer": "Інформаційні технології", "correct": 1 }, { "answer": "Астрофізика", "correct": 0 } ] } ] }
```

Рис. 7. Інформація, яка надходить до API

```

class GeneratedAiQuestionsDTO {
    public array $questions = []; // Масив об'єктів QuestionDTO
    public function __construct(array|string $data)
    {
        if (is_string($data)) {
            $data = json_decode($data, associative: true);
        }
        $this->fill($data);
        return $this->questions;
    }
    private function fill(array $data)
    {
        foreach ($data['questions'] as $questionData) {
            $this->questions[] = new QuestionDTO($questionData);
        }
    }
}

class OptionDTO {
    public $variant;
    public $correct;
    public function __construct($data)
    {
        $this->variant = $data['answer'];
        $this->correct = $data['correct'];
    }
}

class QuestionDTO {
    public $question = '';
    public $options = [];
    public function __construct(array|string $data)
    {
        if (is_string($data)) {
            $data = json_decode($data, associative: true);
        }
        $this->fill($data);
        return [
            'question' => $this->question,
            'options' => $this->options
        ];
    }
    private function fill(array $data)
    {
        $this->question = $data['question'];
        foreach ($data['options'] as $optionData) {
            $this->options[] = new OptionDTO($optionData);
        }
    }
}

```

Рис. 8. Логіка перетворення інформації

```

App\Http\Dtos\GeneratedAiQuestionsDTO [#539 // app\Jobs\GetDataFromAi.php:
-questions: array:2 [
  0 => App\Http\QuestionDTO [#540
+question: "Яка основна функція програми PocketTracer?"
+options: array:4 [
  0 => App\Http\OptionDTO [#541
+variant: "Графічний дизайн"
+correct: 0
  ]
  1 => App\Http\OptionDTO [#542
+variant: "Моделювання мереж"
+correct: 1
  ]
  2 => App\Http\OptionDTO [#543
+variant: "Музичне створення"
+correct: 0
  ]
  3 => App\Http\OptionDTO [#544
+variant: "Текстова переписка"
+correct: 0
  ]
  ]
  ]
  1 => App\Http\QuestionDTO [#545
+question: "Яке призначення має PocketTracer?"
+options: array:4 [
  0 => App\Http\OptionDTO [#546
+variant: "Фільмова монтаж"
+correct: 0
  ]
  1 => App\Http\OptionDTO [#547
+variant: "Кулінарія"
+correct: 0
  ]
  2 => App\Http\OptionDTO [#548
+variant: "Моделювання та симуляція мережі"
+correct: 1
  ]
  ]
  ]
]

```

Рис. 9. Результуючий оброблений об'єкт

**Висновки з дослідження і перспективи подальших розвідок у цьому напрямі.** Таким чином, у роботі досліджено питання доцільності використання ШІ при розробці тестів. Завдяки розробленій математичній моделі можна розрахувати, що економія часу при використанні ШІ для створення тесту може сягати майже 62,5%. Це допомагає значно зменшити витрати часу викладача. Завдяки такій можливості, можна створювати тести різних рівнів складності, що дозволяє виявляти слабкі місця в знаннях та корегувати навчальний процес більш ефективно. Якщо порівняти два підходи, то видно, що при автоматичній генерації з використанням ШІ, продуктивність роботи викладача значно збільшиться.

#### Список використаних джерел:

1. Castro, Mayleen Dorcas B.; Tumibay, Gilbert M. A literature review: efficacy of online learning courses for higher education institution using meta-analysis. *Education and Information Technologies*, 2021, 26 (2). P. 1367-1385.
2. Alhazbi, Saleh; Hasan, Mahmood A. The role of self-regulation in remote emergency learning: comparing synchronous and asynchronous online learning. *Sustainability*, 2021, 13 (19). P. 11070.
3. Тягунова М.Ю., Лаврик В.Р. Віртуальна лабораторія як ефективне рішення під час дистанційної освіти. *Systems and Technologies*, 2023, 66 (2). С. 125-131.
4. Liu, Zi-Yu; Lomovtseva, Natalya; Korobeynikova, Elena. Online learning platforms: Reconstructing modern higher education. *International Journal of Emerging Technologies in Learning (iJET)*, 2020, 15 (13). P. 4-21.

- 
5. Van Wart, Montgomery, et al. Integrating students' perspectives about online learning: a hierarchy of factors. *International Journal of Educational Technology in Higher Education*, 2020, 17 (1). P. 53.
  6. Arief, Harefaan, et al. Implementation Entrepreneurship Education Online-Learning Program To Create Farmer Entrepreneur Through Urban Farming. In: *ICCD*. 2021. P. 102-106.
  7. Леонтьєва, О. О.; Ткачук, Г. В. Особливості створення тестів засобами онлайн-сервісу Kahoot. *Матеріали II Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених за тематикою «Сучасні комп'ютерні системи та мережі в управлінні»: збірка наукових праць/Під редакцією ГО Райко.–Херсон: ФОП Вишемирський ВС, 2019.–409 с., 2019, с. 158.*
  8. Sarsa, Sami, et al. Automatic generation of programming exercises and code explanations using large language models. In: *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1*. 2022. P. 27-43.
  9. Kaddoura, Sanaa; Popescu, Daniela Elena; Hemanth, Jude D. A systematic review on machine learning models for online learning and examination systems. *PeerJ Computer Science*, 2022, 8. P. e986.
  10. Zaadnoordijk, Lorian, et al. A global perspective on testing infants online: Introducing ManyBabies-AtHome. *Frontiers in Psychology*, 2021, 12, p. 703234.
  11. Seo, Kyoungwon, et al. The impact of artificial intelligence on learner–instructor interaction in online learning. *International journal of educational technology in higher education*, 2021, 18. P. 1-23.

#### References:

1. Castro, M. D. B., & Tumibay, G. M. (2021). A literature review: efficacy of online learning courses for higher education institution using meta-analysis. *Education and Information Technologies*, 26(2), 1367-1385.
2. Alhazbi, S., & Hasan, M. A. (2021). The role of self-regulation in remote emergency learning: comparing synchronous and asynchronous online learning. *Sustainability*, 13(19), 11070.
3. Tiahunova, M. Y., & Lavryk, V. R. (2023). Virtualna laboratorii yak efektyvne rishennia pid chas dystantsiinoi osvity. *Systems and Technologies*, 66 (2), 125-131.
4. Liu, Z. Y., Lomovtseva, N., & Korobeynikova, E. (2020). Online learning platforms: Reconstructing modern higher education. *International Journal of Emerging Technologies in Learning (iJET)*, 15(13), 4-21.
5. Van Wart, M., Ni, A., Medina, P., Canelon, J., Kordrostami, M., Zhang, J., & Liu, Y. (2020). Integrating students' perspectives about online learning: a hierarchy of factors. *International Journal of Educational Technology in Higher Education*, 17(1),
6. Arief, H., Soelton, M., Saratian, E. T. P., Tafiprios, T., Puspaningrum, A., & Mugiono, M. (2021, October). Implementation Entrepreneurship Education Online-Learning Program To Create Farmer Entrepreneur Through Urban Farming. In *ICCD* (Vol. 3, No. 1, pp. 102-106).
7. Leontieva, O. O., & Tkachuk, H. V. (2019). Osoblyvosti stvorennia testiv zasobamy onlain-servisu Kahoot. *Materialy II Vseukrainskoi naukovo-praktychnoi internet-konferentsii studentiv, aspirantiv ta molodykh vchenykh za tematykoiu «Suchasni kompiuterni systemy ta merezhi v upravlinni»: zbirka naukovykh prats/Pid redaktsiieiu HO Raiko. – Kherson: FOP Vyshemyrskyi VS, 2019. – 409 p., 158.*
8. Sarsa, S., Denny, P., Hellas, A., & Leinonen, J. (2022, August). Automatic generation of programming exercises and code explanations using large language models. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1* (pp. 27-43).
9. Kaddoura, S., Popescu, D. E., & Hemanth, J. D. (2022). A systematic review on machine learning models for online learning and examination systems. *PeerJ Computer Science*, 8, e986.
10. Zaadnoordijk, L., Buckler, H., Cusack, R., Tsuji, S., & Bergmann, C. (2021). A global perspective on testing infants online: Introducing ManyBabies-AtHome. *Frontiers in Psychology*, 12, 703234.
11. Seo, K., Tang, J., Roll, I., Fels, S., & Yoon, D. (2021). The impact of artificial intelligence on learner–instructor interaction in online learning. *International journal of educational technology in higher education*, 18, 1-23.