

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

кафедра “Інформаційні системи та мережі”

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

до магістерської кваліфікаційної роботи на тему:

**«Інтелектуальна система освоєння іноземної мови з  
використанням статистичного аналізу»**

Студента групи САМ-21 Мироша Юрія Михайловича  
(група, шифр, прізвище та ініціали)

Керівник роботи \_\_\_\_\_ ( Христина Микіч )

Консультант \_\_\_\_\_ ( Тетяна ШЕПЕЛЬ )

Нормоконтроль \_\_\_\_\_ ( Андрій ВАСИЛЮК )

Завідувач кафедри ІСМ \_\_\_\_\_ ( Дмитро ДОСИН )

« 21 » \_\_\_\_\_ травня \_\_\_\_\_ 2025 р.

ЛЬВІВ - 2025

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Інститут комп’ютерних наук та інформаційних технологій  
Кафедра “Інформаційні системи та мережі”  
Спеціальність 124 “Системний аналіз”  
ОНП «Системний аналіз»**

**ЗАТВЕРДЖУЮ:**  
Зав. кафедри ІСМ \_\_\_\_\_  
“ 21 ” \_\_\_\_\_ травня \_\_\_\_\_ 2025р.

### ЗАВДАННЯ

на кваліфікаційну роботу студента групи САМ-21

Мирош Юрій Михайлович  
(прізвище, ім’я, по батькові)

1. Тема роботи: Інтелектуальна система освоєння іноземної мови з використанням статистичного аналізу

затверджена наказом за НУ «ЛП» від « 11 » \_\_\_\_\_ березня \_\_\_\_\_ 2025 р. № 864-4-06

2. Термін здачі студентом закінченої роботи: 12 травня 2025р.

3. Вихідні відомості до роботи: літературні джерела та джерела глобальної мережі Інтернет, методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 124 “Системний аналіз”

4. Зміст розрахунково-пояснювальної записки: (перелік питань, які належить розробити) проаналізувати літературні джерела, провести системний аналіз, обрати програмні засоби та технології для реалізації продукту, розробити програмний продукт, проаналізувати економічну доцільність роботи.

5. Перелік графічного матеріалу: схема дерева цілей, діаграма MAI, діаграми в нотації UML, ієрархія задач, знімки екрану

6. Перелік програмних продуктів, які належить використати в процесі розроблення роботи (проекту) Microsoft Office Professional 2019, Swagger, MS SQL Server, Postman, Visual Studio 2022

## 7. Консультування роботи, із зазначенням розділів роботи

Розділ	Консультанти	Підпис, дата	
		завдання видав	завдання отримав
Економічний	Шепель Т. П.		

8. Дата видачі завдання 24 лютого 2025р.Керівник \_\_\_\_\_  
(підпис)Завдання отримав \_\_\_\_\_  
(підпис)

## КАЛЕНДАРНИЙ ПЛАН

	Етапи кваліфікаційної роботи	Термін виконання	Примітки
1	Накопичення інформації	10.02.2025-15.03.2025	виконано
2	Дослідження та аналіз літературних джерел	15.03.2025-25.03.2025	виконано
3	Системний аналіз, побудова діаграм потоків даних, дерева цілей та ієрархії задач потоків даних, дерева цілей та ієрархії	25.03.2025-31.03.2025	виконано
4	Проектування та розробка структури системи	01.04.2025-10.04.2025	виконано
5	Розробка системи	01.04.2025-20.04.2025	виконано
6	Тестування та відлагоджування	21.04.2025-22.04.2025	виконано
7	Робота над пояснювальною запискою	23.04.2025-28.04.2025	виконано
8	Оформлення дипломної роботи	29.04.2025-01.05.2025	виконано

Студент \_\_\_\_\_  
(підпис)Керівник роботи \_\_\_\_\_  
(підпис)

## ЗМІСТ

<b>ВСТУП</b> .....	6
<b>РОЗДІЛ 1</b> .....	9
<b>Аналітичний огляд літературних та інших джерел</b> .....	9
<b>1.1. Методологічні засади досліджень</b> .....	9
<b>1.2. Попит на вивчення іноземних мов</b> .....	10
<b>1.3. Статистичний аналіз</b> .....	11
<b>1.4. Аналіз відомих систем</b> .....	12
<b>РОЗДІЛ 2</b> .....	19
<b>Системний аналіз та обґрунтування проблемної ситуації</b> .....	19
<b>2.1. Проблема розроблення системи освоєння іноземної мови з використанням статистичного аналізу</b> .....	19
<b>2.2. Системний аналіз об'єкта дослідження та предметної області</b> .....	21
<b>2.3. Використання UML задля конструювання моделі системи</b> .....	29
<i>2.3.1. Діаграма варіантів використання</i> .....	31
<i>2.3.2. Діаграма класів</i> .....	34
<i>2.3.3. Діаграма станів</i> .....	36
<i>2.3.4. Діаграма діяльності</i> .....	38
<i>2.3.5. Діаграма розгортання</i> .....	39
<i>2.3.6. Діаграма послідовності</i> .....	41
<b>2.4. Мета розроблення інформаційної системи</b> .....	42
<b>2.5. Місце застосування системи</b> .....	43
<b>2.6. Постановка та обґрунтування проблеми</b> .....	44
<b>2.7. Очікувані ефекти від впровадження системи</b> .....	45
<b>2.8. Розроблення концептуальної моделі системи</b> .....	47
<b>Висновок до другого розділу</b> .....	49
<b>РОЗДІЛ 3</b> .....	50
<b>Методи та засоби вирішення проблеми</b> .....	50
<b>3.1. Вибір та обґрунтування методів розв'язання задачі</b> .....	50
<b>3.2. Вибір та обґрунтування засобів розв'язання задачі</b> .....	52
<b>3.3. Технічні характеристики обраних програмних засобів розроблення</b> .....	56
<b>Висновки до третього розділу</b> .....	61
<b>РОЗДІЛ 4</b> .....	62

	5
<b>Практична реалізація</b> .....	<b>62</b>
<b>4.1. Описання реалізації завдання</b> .....	<b>62</b>
4.1.1. <i>Загальні відомості про програму</i> .....	62
4.1.2. <i>Структура бази даних</i> .....	62
4.1.3. <i>Функції програми</i> .....	67
<b>4.2. Аналіз отриманих результатів</b> .....	<b>68</b>
<b>Висновки до четвертого розділу</b> .....	<b>77</b>
<b>РОЗДІЛ 5</b> .....	<b>79</b>
<b>Економічне обґрунтування доцільності роботи</b> .....	<b>79</b>
<b>5.1. Економічна характеристика проектного рішення</b> .....	<b>79</b>
<b>Висновки до п'ятого розділу</b> .....	<b>82</b>
<b>ВИСНОВКИ</b> .....	<b>84</b>
<b>СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ</b> .....	<b>86</b>
<b>АНОТАЦІЯ</b> .....	<b>89</b>
<b>ABSTRACT</b> .....	<b>92</b>
<b>ДОДАТКИ</b> .....	<b>95</b>
<b>Додаток А</b> .....	<b>95</b>

## ВСТУП

У сучасну епоху глобалізації та швидкого розвитку технологій володіння іноземними мовами, зокрема англійською, стало однією з ключових компетенцій для успішного навчання, побудови кар'єри та особистісного розвитку. Зростаюче значення англійської мови у сфері вищої освіти, підтверджене останніми змінами в освітніх політиках, підкреслює необхідність впровадження інноваційних підходів до процесу вивчення мов. Хоча традиційні методи навчання мають свої сильні сторони, вони часто не враховують індивідуальні особливості, темп засвоєння матеріалу та потреби кожного учня, що створює актуальний запит на сучасні технологічні рішення, спрямовані на оптимізацію та персоналізацію навчального процесу.

Це дослідження зосереджене на розробці інтелектуальної системи для вивчення іноземних мов, яка використовує методи статистичного аналізу для підвищення ефективності навчального процесу. Вона також надає персоналізовані рекомендації, спрямовані на покращення результатів. Застосування статистичних методів дозволяє не лише точно оцінювати успішність навчання, але й створювати індивідуальні стратегії, які сприяють досягненню максимального результату. Такий інноваційний підхід відповідає сучасним викликам у сфері освіти та ефективно вирішує практичні проблеми, з якими стикаються учні на шляху до оволодіння іноземною мовою на високому рівні.

Цінність цього дослідження полягає в його здатності інтегрувати традиційні методи навчання з вимогами сучасного світу, де ключову роль відіграє аналіз даних. Розроблена система автоматизує процес моніторингу прогресу учнів, адаптує навчальні матеріали відповідно до їхніх потреб і надає зворотний зв'язок на основі детального аналізу даних. Такий підхід дозволяє переосмислити традиційні підходи до вивчення мов, роблячи їх більш ефективними та персоналізованими. Це дослідження робить вагомий внесок у розвиток освітніх технологій, пропонуючи масштабоване, універсальне та результативне рішення, яке може бути впроваджене в різних навчальних середовищах. Воно сприяє підвищенню доступності та якості мовної освіти, відповідаючи на сучасні виклики у сфері навчання.

*Актуальність теми.* На офіційному сайті МОН України 13 липня 2019 року було опубліковано інформацію про плани щодо активного просування іноземних мов, зокрема англійської, у закладах вищої освіти. У документі зазначено, що планується організація мовних курсів та інтенсивів для студентів, а також запровадження нових вимог до рівня володіння англійською мовою. Згідно з цими вимогами, знання англійської на рівні B1 або вище стане обов'язковим для вступу до університету, а рівень B2 – необхідним для випуску. Крім того, пропонується викладання окремих профільних дисциплін англійською мовою, а також регулярна перевірка знань у межах встановлених часових рамок. Одним із ключових нововведень є обов'язкове тестування на національному рівні (ЗНО) з англійської мови після завершення першого рівня освіти.

Зважаючи на ці зміни, можна зробити висновок, що англійська мова стає не лише важливим інструментом для підвищення конкурентоспроможності на ринку праці, але й необхідною умовою для здобуття вищої освіти. Це значно розширює аудиторію потенційних користувачів освітніх систем, орієнтованих на вивчення англійської мови, і підкреслює її ключову роль у сучасному освітньому процесі.

*Мета і задачі дослідження.* Метою цього дослідження є створення інтелектуальної системи для вивчення англійської та інших іноземних мов, яка використовує методи статистичного аналізу для оптимізації навчального процесу з урахуванням індивідуальних особливостей кожного користувача. Розроблена система автоматизуватиме процес моніторингу прогресу, адаптуватиме навчальні матеріали відповідно до поточного рівня знань користувача та надаватиме персоналізовані рекомендації для покращення результатів. Використання статистичних методів дозволить забезпечити точну оцінку ефективності навчання, а також розробляти індивідуальні стратегії, які сприятимуть досягненню максимальних результатів у вивченні або вдосконаленні знань іноземної мови. Такий підхід спрямований на підвищення якості та ефективності мовної освіти, відповідаючи сучасним викликам у сфері навчання.

*Задачі дослідження.* Для досягнення поставленої мети були сформульовані такі

завдання, які необхідно виконати в рамках дослідження:

- провести аналітичний огляд предметної області та вивчити наукові й технічні джерела, пов'язані з темою;
- провести системний аналіз об'єкта, що досліджується, описати його предметну область, визначити цілі інтелектуальної системи та побудувати відповідні діаграми для моделювання;
- обрати оптимальні інструменти та методи, необхідні для реалізації системи;
- розробити детальний план впровадження запропонованої системи;
- визначити та описати функціональні можливості створеної системи;
- обґрунтувати економічну доцільність розробки та надати її опис.

*Об'єкт дослідження:* процес освоєння іноземної мови за допомогою використання статистичного аналізу.

*Предмет дослідження:* методи та засоби створення інтелектуальної системи освоєння іноземної мови з використанням статистичного аналізу.

*Практичне значення одержаних результатів* розробка програмного забезпечення, а саме веб-сайту тестувань згенерованих за допомогою статистичного аналізу матеріалів іноземною мовою.

*Апробація роботи.* Результат роботи були показані у 99 випуску Міжнародної наукової інтернет-конференції на тему "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення" із публікацією тез:

Мирош Ю. М. Інтелектуальна система освоєння іноземної мови з використанням статистичного аналізу/Мирош Ю. М. Микіч Х. І. – матеріали 99 видання Міжнародної наукової інтернет-конференції на тему "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення" (14-15 травня 2025 р. м. Ополе, Польща).

## РОЗДІЛ 1

### Аналітичний огляд літературних та інших джерел

#### 1.1. Методологічні засади досліджень

Освоєння іноземної мови є складним і трудомістким процесом, що вимагає від людини значної самовіддачі, наполегливості та зусиль. У сучасних умовах існує безліч методів і технік, які сприяють підвищенню ефективності цього процесу, дозволяючи раціонально використовувати час і ресурси. Як і в будь-якій іншій діяльності, важливо починати з базових принципів. Дослідження у сфері «Засвоєння другої мови» виділяють три основні аспекти, які є ключовими для успішного навчання:

1. Прагнути сприймати інформацію (слухати, читати) новою мовою, намагаючись її зрозуміти.
2. Згодом переходити до створення власного мовного продукту (писати, говорити).
3. Аналізувати свої помилки та вдосконалюватися, враховуючи їх.

Ці три складові є фундаментом будь-якої мовної практики, і ефективний план навчання має забезпечувати їх збалансоване поєднання. Чим більше ви слухаєте та читаєте (сприймаєте інформацію), чим активніше говорите та пишете (відтворюєте), а також чим ретельніше аналізуєте свої помилки та працюєте над ними, тим швидше вдосконалюються ваші мовні навички.

Мотиви, які спонукають людей до вивчення іноземних мов, можуть бути різноманітними. Серед основних причин можна виділити такі:

- Бажання покращити соціальний статус і зменшити упередження, які часто виникають через незнання іншої мови.
- Високий попит на знання додаткових мов серед роботодавців, особливо для високооплачуваних вакансій.
- Можливість глибше пізнати культуру, яка давно цікавить, а також знайомство з новими людьми.
- Розширення світогляду та підвищення впевненості у собі завдяки новим

знанням.

- Використання спілкування як інструменту навчання, що дозволяє не лише вдосконалити мовні навички, а й розвинути комунікативні здібності.
- Практична користь під час подорожей за кордон, де знання мови може стати незамінним.
- Розширення професійних можливостей у таких сферах, як уряд, медицина, бізнес, юриспруденція тощо.
- Покращення пам'яті, що є важливим у сучасному ритмі життя.
- Глибше розуміння власної мови, яке часто приходить під час вивчення нової.
- Доступ до мистецтва, музики, кіно, театру, філософії та науки, які стають ближчими завдяки знанню іноземної мови.
- Вивчення іноземних мов також регламентується на законодавчому рівні [1].

Таким чином, вивчення мови не лише відкриває нові можливості, але й сприяє всебічному розвитку особистості.

## **1.2. Попит на вивчення іноземних мов**

Якщо розглянути ситуацію з вивченням мов у світі, то на сьогодні 1,2 мільярда людей займаються вивченням іноземних мов, причому дві третини з них обирають англійську, прагнучи отримати кращу роботу та вищий дохід [2].

Аналізуючи глобальну статистику, можна визначити п'ять найпоширеніших мов у світі за кількістю людей, які ними володіють, включаючи як носіїв, так і тих, хто вивчив ці мови. Наприклад, англійською мовою користуються близько 1,5 мільярда людей, з яких лише 25% є носіями, а решта 75% опанували її як другу мову. У випадку китайської мови, серед 1,1 мільярда користувачів лише 11% є носіями. Для хінді цей показник становить приблизно 30% з 650 мільйонів людей, які нею спілкуються. Іспанська мова має 420 мільйонів користувачів, з яких 22% є носіями. Французька ж демонструє найвищий відсоток тих, хто вивчив її як другу мову, – 79% із загальної

кількості 370 мільйонів користувачів.

Крім того, серед населення Європейського Союзу двоє з трьох осіб (66%) володіють хоча б однією іноземною мовою.

Зважаючи на ці дані, можна зробити висновок, що прагнення до вивчення нової мови або вдосконалення вже наявних знань є цілком обґрунтованим. Окрім очевидних переваг, таких як розширення кар'єрних можливостей і підвищення рівня доходу, знання іноземної мови стає сучасною необхідністю. Це здійснює вплив не тільки на сприйняття людини в суспільстві, але й на її загальний рівень освіченості, адже значна частина світової інформації не перекладається миттєво на рідну мову. Таким чином, володіння іноземною мовою надає більше доступу до ширшого спектру знань, ресурсів і можливостей, сприяючи особистісному та професійному розвитку.

### **1.3. Статистичний аналіз**

У сьогоденнішньому світі аналіз контенту має велике застосування в різноманітних галузях науки, демонструючи свою важливість у дослідницькій діяльності. Ця сфера охоплює численні наукові роботи, серед яких однією з найбільш цитованих є дослідження Laver, Benoit і Garry (2003) [3], виконане в рамках політології. У цій роботі порівнюється ефективність традиційних методів аналізу контенту з інноваційним підходом, заснованим на статистичному аналізі слів.

Традиційний метод, що передбачає ручний відбір слів, потребує значних зусиль, часу та ресурсів. Натомість автоматизований підхід, який базується на машинному аналізі контенту, є більш ефективним, надійним і відтворюваним. Однак, незважаючи на переваги, автоматизовані алгоритми для аналізу фраз можуть бути дорогими у впровадженні та вимагати регулярного оновлення. Крім того, їх доступність для мов, які суттєво відрізняються від англійської, залишається обмеженою, що створює додаткові виклики.

Одним із найпоширеніших методів аналізу контенту є використання статистичного розподілу виділених слів. У цій галузі існує велика кількість веб-ресурсів і наукових статей, присвячених застосуванню статистичного аналізу.

Наприклад, система wordfreq, описана у роботі Mehmet F. Dicle та Betul Dicle (2018) [4], була розроблена для підтримки дослідників у реалізації їхніх проєктів, пов'язаних із аналізом контенту.

Таким чином, статистичний аналіз слів виступає ключовим інструментом у дослідженнях контенту, забезпечуючи високу ефективність, точність і надійність, особливо в автоматизованих системах. Його застосування дозволяє значно оптимізувати процес аналізу, відкриваючи нові можливості для наукових досліджень і практичних завдань.

#### 1.4. Аналіз відомих систем

Ідея вивчення іноземних мов, як і використання статистичного аналізу для аналізу текстів, книг, статей чи веб-сайтів, далеко не нова. Саме тому важливо розглянути доступні альтернативи та інструменти, які можуть бути корисними в цій сфері.

Одним із таких інструментів є система wordfreq, яка використовується для статистичного аналізу тексту. Основний принцип її роботи полягає в завантаженні веб-сторінки або локального файлу та створенні статистичного розподілу слів [4].

Синтаксис використання wordfreq виглядає наступним чином:

``wordfreq using filename [min length (integer) nonumbers nogrammar nowww noscommon clear append]``, де:

- `filename` – ім'я файлу, який буде оброблятися;
- `min length (integer)` – мінімальна кількість символів у слові для включення його до статистичного розподілу (за замовчуванням – 0);
- `nonumbers` – видалення слів, що містять цифри;
- `nogrammar` – виключення слів, які є частиною загальної граматики (наприклад, "is" або "are");
- `nowww` – видалення слів, пов'язаних із http або html;
- `clear` – очищення даних у пам'яті;
- `append` – додавання нового статистичного розподілу до існуючого.

Система може працювати як із веб-адресами (починаючи з http або https), так і з локальними файлами будь-якого формату. Для всіх файлів обробляється їхній ASCII-вміст.

Процес роботи wordfreq включає кілька етапів:

1. Завантаження веб-сторінки або локального файлу.
2. Підготовка статистичного розподілу всіх унікальних слів у тексті.
3. Видалення символів, які не входять до ASCII-таблиці, та заміна їх пробілами.
4. Розбиття тексту на слова за допомогою пробілів як розділювачів.

Однак при аналізі веб-сайтів можуть виникати труднощі через наявність реклами або службових тегів, які додають до списку так звані "недослова" (наприклад, назви змінних, функцій чи класів). Для зручності користувачеві надаються чотири різні списки, які дозволяють виключати слова, що містять цифри, пов'язані з http або html, або інші небажані елементи.

Окрім інструментів для статистичного аналізу, існують також популярні додатки для вивчення іноземних мов. Одним із найвідоміших є Duolingo.

Duolingo став частиною хвилі безкоштовної освіти, яка поширилася в Інтернеті на початку 2010-х років. Його розробники використали можливості Інтернету та мобільних пристроїв, щоб зробити навчання доступним для широкої аудиторії [5].

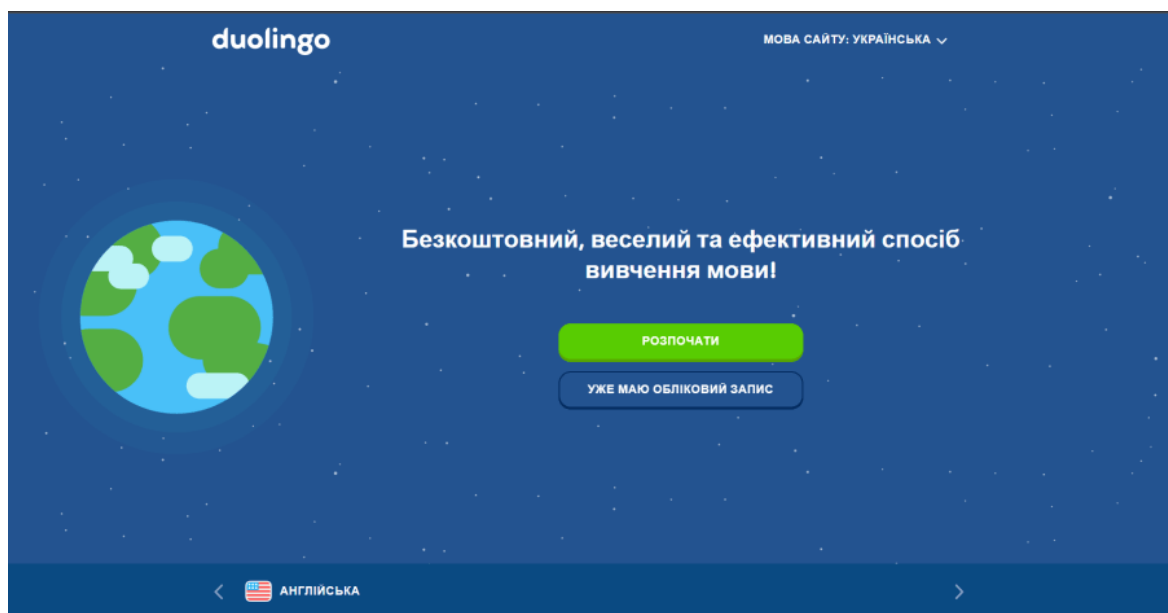
Ще до офіційного запуску Duolingo здобув популярність. У листопаді 2011 року була запущена приватна бета-версія, і за місяць у списку очікування зареєструвалися 300 000 людей. Протягом першого року додаток завантажили понад 10 мільйонів користувачів.

На відміну від традиційних підходів, Duolingo використовує елементи гейміфікації, щоб зробити навчання цікавим і мотивуючим. Користувачі можуть змагатися з друзями або іншими учасниками, проходячи курси та отримуючи нагороди за досягнення.

Ще однією перевагою Duolingo є використання даних для оптимізації навчального процесу. Додаток регулярно проводить тести, щоб визначити, як швидко

користувачі засвоюють матеріал, і коригує курси для полегшення навчання.

Станом на сьогодні Duolingo має понад 500 мільйонів зареєстрованих користувачів, з яких 42 мільйони є активними щомісяця. Додаток пропонує 95 курсів на 38 мовах, а також має веб-версію (Рис. 1.1), де користувачі можуть зареєструватися та розпочати навчання.



*Рис. 1.1 Початкова сторінка веб-сайту Duolingo*

Для вивчення іноземних мов цей сайт пропонує інтерфейс, оформлений у вигляді гри (Рис. 1.2), що значно полегшує процес сприйняття матеріалу та підвищує зацікавленість серед звичайних користувачів.

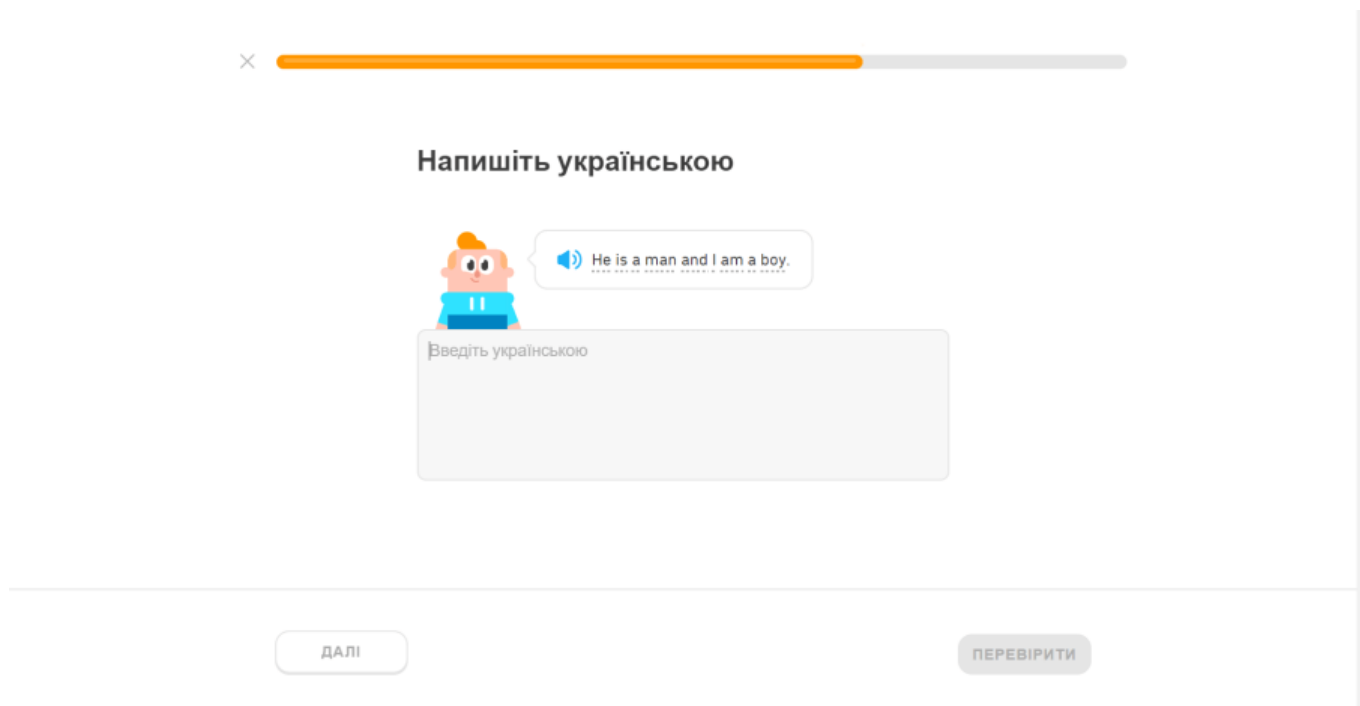


Рис. 1.2. Одне з завдань для навчання

Крім того, на платформі доступна система змагань (Рис. 1.3), яка додатково стимулює користувачів до активного вивчення нового матеріалу та сприяє регулярності у навчанні.

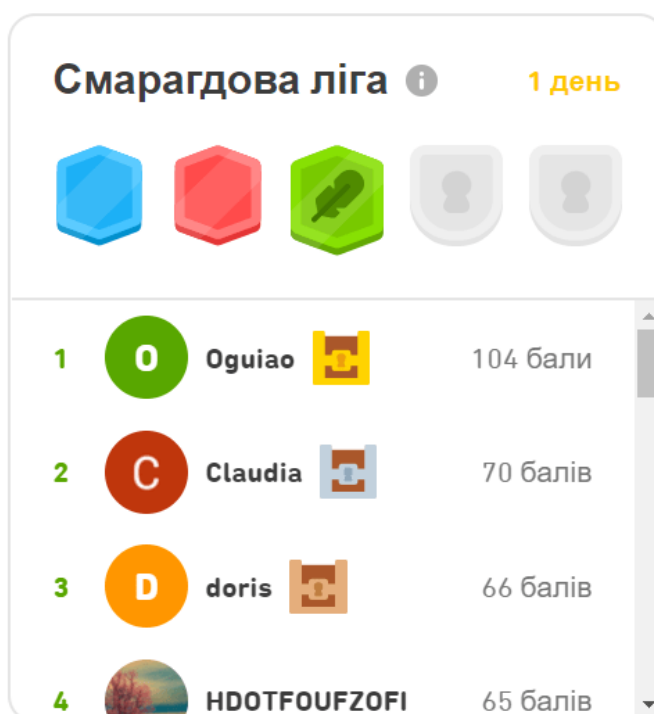


Рис. 1.3 Розподіл за місцями й лігами

До недоліків платформи можна віднести низький рівень складності завдань, який

не перевищує рівня B1.

Ще одним прикладом системи-аналога для вивчення іноземних мов є веб-сайт «test-english». Хоча він спеціалізується лише на одній мові, його навчальна цінність є досить високою.

На стартовій сторінці веб-сайту відображаються основні доступні функції та можливості для користувачів (Рис. 1.4).

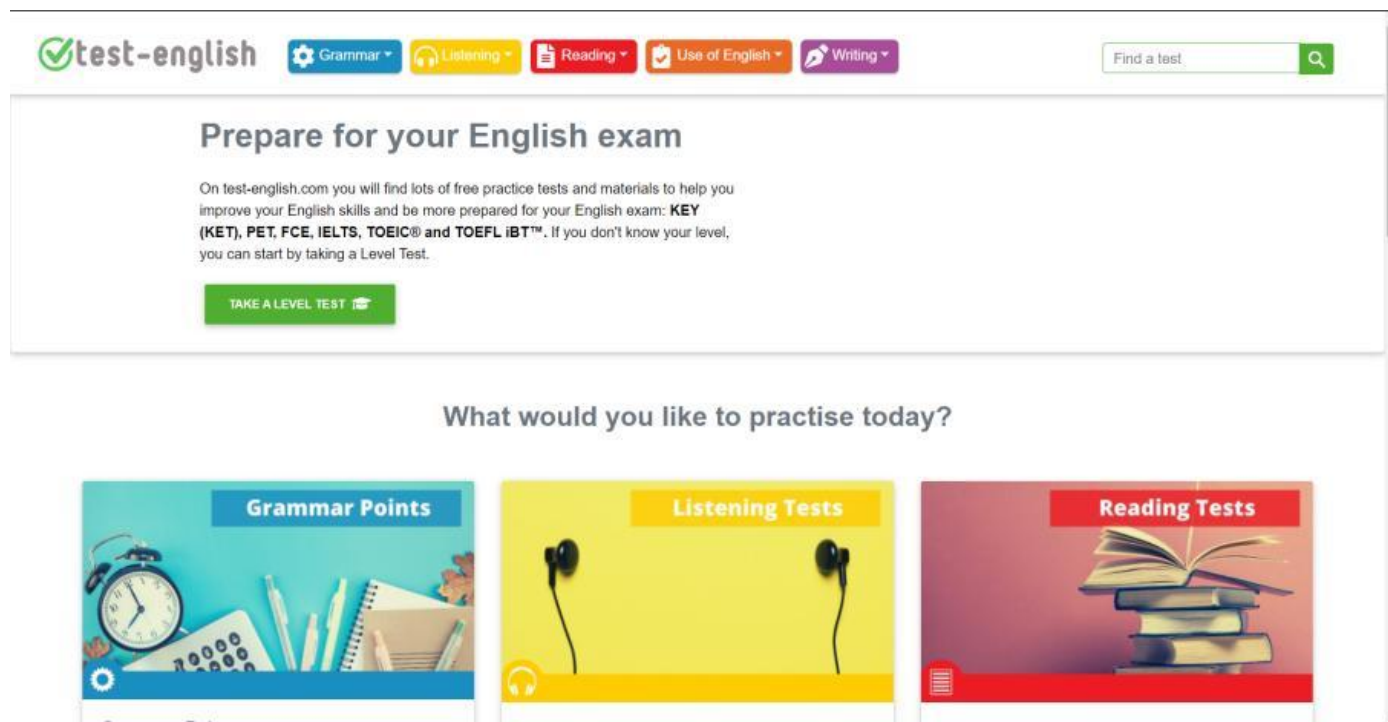


Рис. 1.4 Стартова сторінка веб-сайту «test-english»

Особливістю цієї системи є її всебічний підхід до вивчення мови, який охоплює багато аспектів навчання мови, зокрема: граматику, вимову, лексику, аудіювання та письмо. Вона не обмежується окремими компонентами, а пропонує комплексний підхід, що сприяє глибокому засвоєнню матеріалу. Завдяки такій багатогранності веб-сайт здобув популярність як серед тих, хто прагне вдосконалити свої навички в іноземних мовах, так і серед викладачів англійської мови, які використовують його як додатковий навчальний ресурс.

Функціонал сайту побудований на основі інтерактивних тестувань (Рис. 1.5), які дозволяють користувачам не лише перевіряти свої знання, але й отримувати детальні пояснення щодо правильності кожної відповіді (Рис. 1.6). Такий підхід робить процес

навчання більш зрозумілим, інтерактивним і ефективним, сприяючи кращому засвоєнню матеріалу та підвищенню мотивації до навчання.

The screenshot shows the 'test-english' website interface. At the top, there are navigation buttons for Grammar, Listening, Reading, Use of English, and Writing. The main content area is titled 'Grammar » B1+ Grammar lessons and exercises » Indirect questions'. Below this, there is a section for 'Exercise 1' with a progress indicator showing 'Exercises: 1 2 3'. The exercise instructions are: 'Choose the correct forms to complete the following questions. There are direct and indirect questions.' There are seven numbered questions, each with a dropdown menu for the answer:

- 1 I don't know the answer. How many times  to tell you?
- 2 I don't know  so unhappy.
- 3 Do you have any idea when  ?
- 4 How many people  for the party?
- 5 I wonder .
- 6 Did she tell you how often  ?
- 7 I'd like to know  for the experiment.

Рис. 1.5 Приклад тестування

The screenshot shows a hint for question 3. The question is: '3 Do you have any idea when  ✓?'. Below the question, there is a hint box that says: '⇨ In indirect questions the order of the sentence is subject + verb.'

Рис. 1.6 Підказки після тестування

Дана система є ефективним інструментом для освоєння англійської мови, проте її вузька спеціалізація лише на одну мову може бути сприйнята як обмеження.

### Висновок до першого розділу

У першому розділі було проведено детальне дослідження, яке включало огляд і аналіз наявних літературних джерел, а також розгляд основних причин, методів і

засобів вивчення іноземних мов, доступних через мережу Інтернет. Особливу увагу було приділено аналізу трьох систем-аналогів, які представляють різні підходи до навчання мов: одна з них базується на використанні статистичного аналізу, а дві інші орієнтовані на традиційні методи вивчення іноземних мов. Ці системи є одними з найпопулярніших варіантів для мовного навчання, що активно використовуються користувачами.

Кожна з розглянутих систем має свої переваги та недоліки, серед яких можна виділити такі ключові аспекти:

- обмежений спектр застосування, що не дозволяє охопити широку аудиторію;
- низький рівень складності завдань, який не завжди відповідає потребам користувачів із різним рівнем знань;
- вузька спеціалізація, зосереджена лише на одній мові, що обмежує можливості для багатомовного навчання.

Таким чином, розробка більш досконалої системи для вивчення іноземних мов із застосуванням статистичного аналізу є актуальним завданням. Така система здатна вирішити існуючі проблеми, запропонувати нові функціональні можливості та забезпечити продуктивний і ефективний процес навчання, адаптований до потреб сучасних користувачів.

## РОЗДІЛ 2

### Системний аналіз та обґрунтування проблемної ситуації

#### 2.1. Проблема розроблення системи освоєння іноземної мови з використанням статистичного аналізу

Освоєння іноземної мови є комплексним і багаторівневим процесом, який потребує відчутних зусиль, часу та систематичного підходу. Однак, навіть за наявності сучасних технологій, багато людей стикаються з труднощами у вивченні мов через низку проблем, які можна вирішити за допомогою інноваційних підходів, зокрема використання статистичного аналізу.

Основні проблеми, які виникають у процесі розуміння й осягнення іноземної мови, включають:

1. Індивідуальні особливості навчання: Кожна людина має свій унікальний стиль навчання, рівень сприйняття інформації та швидкість освоєння матеріалу. Конвенційні методи навчання доволі часто не беруть у розрахунок цих особливостей, які приводять до зменшення ефективності й продуктивності навчання.
2. Відсутність персоналізованого підходу: Більшість існуючих систем навчання пропонують однаковий підхід для всіх користувачів, не враховуючи їхній рівень знань, прогрес чи слабкі сторони. Це може демотивувати учнів і уповільнити процес навчання.
3. Складність вимірювання прогресу: Багато студентів не мають чіткого уявлення про свій прогрес у вивченні мови. Відсутність об'єктивних показників може викликати відчуття застою та зниження мотивації.
4. Неефективне використання даних: Сучасні системи навчання часто не використовують дані про поведінку користувачів, їхні помилки та успіхи для адаптації навчального процесу. Це призводить до втрати можливостей для оптимізації навчання.
5. Монотонність навчального процесу: Конвенційні методи навчання часто бувають нудними та одноманітними, що знижує інтерес до навчання. Відсутність

інтерактивності та адаптивності робить процес менш привабливим для користувачів.

6. Відсутність зворотного зв'язку: Багато систем не надають користувачам детальний аналіз їхніх помилок та рекомендацій щодо покращення. Це ускладнює розуміння того, на чому потрібно зосередитися для досягнення кращих результатів.

Можливості вирішення проблеми

Враховуючи вищезазначені проблеми, розробка системи освоєння іноземної мови з використанням статистичного аналізу є перспективним рішенням. Така система дозволить:

- Персоналізувати навчальний процес: Використання статистичного аналізу дозволить адаптувати навчальні матеріали до індивідуальних потреб кожного користувача. Система зможе аналізувати прогрес, визначати слабкі місця та пропонувати відповідні завдання.
- Вимірювати прогрес за допомогою об'єктивних показників: Завдяки збору та аналізу даних про те, наскільки вдало виконують завдання користувачі, система зможе надавати чіткі метрики, які відображатимуть реальний прогрес у вивченні мови.
- Оптимізувати навчальний процес: Аналізуючи дані про поведінку користувачів, система зможе визначати найбільш ефективні методи навчання та автоматично їх впроваджувати.
- Забезпечити інтерактивність та мотивацію: Використання гейміфікації, інтерактивних завдань та адаптивного контенту зробить рутину навчання цікавішою та більш захоплюючою.
- Надавати зворотний зв'язок: Система зможе автоматично аналізувати помилки користувачів та надавати рекомендації щодо їх виправлення, що сприятиме більш ефективному засвоєнню матеріалу.

Етапи розробки системи

1. Аналіз потреб користувачів: Проведення дослідження цільової аудиторії

для визначення основних проблем, з якими зустрічаються люди коли вивчать іноземні мови. Аналіз існуючих рішень на ринку для виявлення їхніх недоліків.

2. Проектування системи: Розробка структури системи, яка включатиме модулі для збору даних, аналізу прогресу, адаптації навчальних матеріалів та надання зворотного зв'язку. Створення прототипу інтерфейсу користувача.

3. Розробка алгоритмів статистичного аналізу: Створення алгоритмів, які будуть аналізувати дані про успішність користувачів, визначати закономірності та адаптувати навчальний процес відповідно до отриманих результатів.

4. Створення інтерфейсу користувача: Розробка й розготання зручного та інтуїтивно зрозумілого інтерфейсу, який забезпечить легкий доступ до навчальних матеріалів, статистики прогресу та рекомендацій.

5. Тестування системи: Повноцінне тестування системи на справжніх користувачах та бажаючих зацікавлених осіб для виявлення недоліків та їх усунення. Оцінка того наскільки система ефективна, якщо порівнювати зі звичними методами навчання.

6. Впровадження та підтримка: Запуск системи у використання, забезпечення технічної підтримки та регулярне оновлення функціоналу на основі зворотного зв'язку від користувачів.

Розробка такої системи дозволить значно підвищити ефективність процесу вивчення іноземної мови, роблячи усі етапи більш персоналізованим, орієнтованими на взаємодію та з прицілом на результат.

## **2.2. Системний аналіз об'єкта дослідження та предметної області**

Системний аналіз — це науковий метод, що являє собою послідовну сукупність методів і засобів, спрямованих на встановлення зв'язків між змінними та елементами досліджуваної системи.

Системний аналіз набув широкого застосування з моменту впровадження комп'ютерів у різні бізнес-операції. Його використання дозволяє розробляти альтернативні плани розвитку для створюваних систем. Завдяки повному

використанню можливостей інформаційних технологій цей метод довів свою ефективність у вирішенні різноманітних завдань.

Основною метою застосування системного аналізу є підвищення впевненості у прийнятті рішень. Крім того, він дозволяє розширити спектр можливих варіантів розвитку системи, забезпечуючи вибір оптимального шляху для подальшого дослідження.

Виконання системного аналізу передбачає побудову дерева цілей (Рис. 2.1).

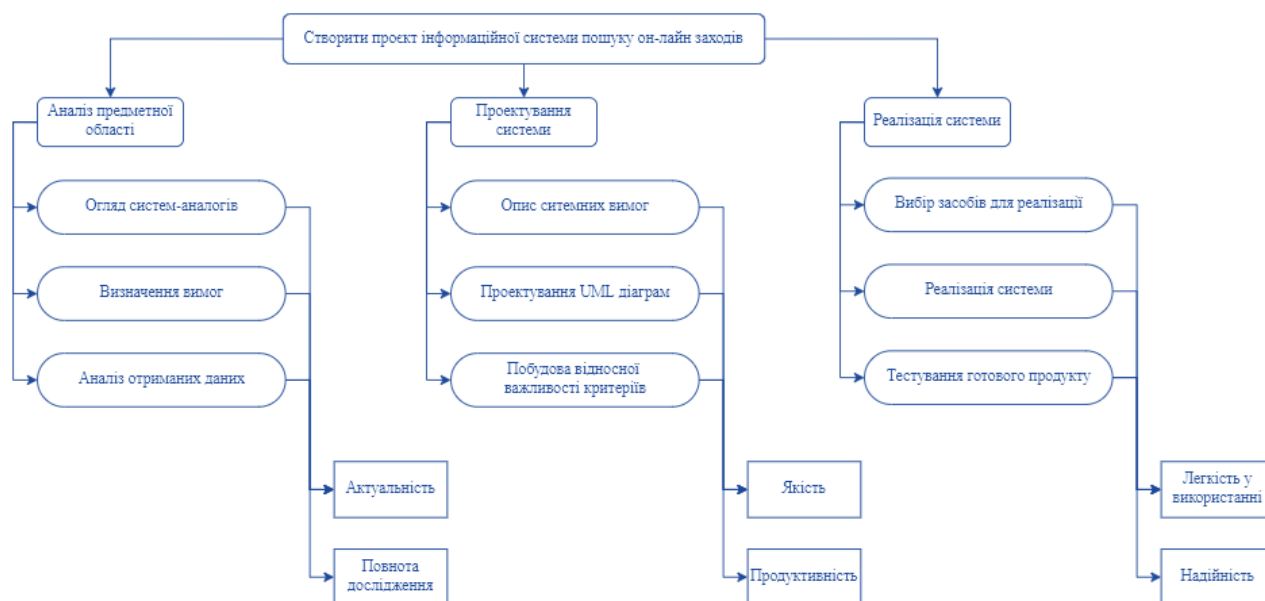


Рис. 2.1 Дерево цілей

Дерево цілей складається з кількох рівнів: головна мета, аспекти цієї мети, підаспекти, а також критерії якості функціонування системи.

#### Основні аспекти:

- **Аналіз предметної області**
- **Проектування системи**
- **Реалізація системи**

#### Аналіз предметної області

Цей аспект є ключовим для успішного виконання системного аналізу, оскільки помилки на цьому етапі можуть негативно вплинути на точність подальших фаз.

#### Аспекти аналізу предметної області:

- **Аналіз систем-аналогів:** необхідно дослідити існуючі рішення, щоб

отримати детальне уявлення про процес, а також оцінити їхні переваги та недоліки.

- **Визначення вимог:** встановлення критеріїв для системи, які забезпечать відповідність очікуванням користувачів.
- **Аналіз отриманих даних:** ретельне вивчення зібраної інформації для формування чіткої картини кінцевого вигляду системи та її основних критеріїв.

#### **Критерії для аналізу предметної області:**

- **Актуальність:** використання лише сучасної інформації, враховуючи швидкий розвиток технологій.
- **Повнота дослідження:** ретельний відбір і аналіз даних для всебічного розуміння процесу.

#### **Проектування системи**

На цьому етапі важливо визначити ключові завдання, функціональні процеси та сформулювати вимоги до системи.

#### **Критерії для проектування:**

- **Якість:** забезпечення високого рівня функціональності системи.
- **Продуктивність:** оптимізація роботи системи для досягнення ефективності.

#### **Підцілі для проектування:**

- **Опис системних вимог:** чітке формулювання критеріїв і виявлення основних проблем, пов'язаних із розробкою.
- **Проектування UML-діаграм:** використання спеціалізованої мови для документування та візуалізації моделей системи.
- **Побудова відносної важливості критеріїв:** аналіз і формування ієрархії критеріїв для коректної реалізації системи.

#### **Реалізація системи**

Цей аспект є вирішальним, оскільки від нього залежить, чи буде система зручною та корисною для користувачів.

### Критерії для реалізації:

- **Легкість у використанні:** забезпечення інтуїтивного та зручного інтерфейсу.

- **Надійність:** гарантування стабільної роботи системи без збоїв.

### Підцілі для реалізації:

- **Вибір засобів для реалізації:** аналіз доступних технологій і вибір найсучасніших рішень, які відповідають поставленим завданням.

- **Реалізація системи:** створення системи відповідно до встановлених вимог і функціональних потреб.

- **Тестування готового продукту:** перевірка прототипу для виявлення багів чи невідповідностей, а також їхнє усунення.

Для вибору типу розроблюваної інформаційної системи були визначені наступні критерії:

- **Актуальність (К1)** — відповідність сучасним вимогам і тенденціям.

- **Достовірність (К2)** — точність і надійність інформації, що надається системою.

- **Якість (К3)** — високий рівень виконання функцій системи.

- **Наочність (К4)** — зручність і зрозумілість представлення даних.

- **Коректність (К5)** — правильність роботи системи без помилок.

- **Швидкодія (К6)** — оперативність виконання запитів і обробки даних.

- **Функціональність (К7)** — здатність системи виконувати необхідні завдання.

- **Надійність (К8)** — стабільність і безперебійність роботи системи.

Для визначення типу інформаційної системи було обрано **метод аналізу ієрархій**. Згідно з цим методом, розглянуто чотири типи інформаційних систем:

1. **Експертна система (А1)** — націлена на використання знань експертів для прийняття рішень й аналізу на їх базі.

2. **На базі штучного інтелекту (А2)** — націлена на машинне навчання для адаптації та вдосконалення процесу здобування знань.

3. **На основі статистичного аналізу (A3)** — базується на обробці та аналізі статистичних даних для оптимізації процесу.

4. **На основі готових даних (A4)** — використовує вже існуючі бази даних для забезпечення функціональності.

Щоб перейти до наступного етапу в процесі аналізу потрібно побудувати матриці для попарних порівнянь базуючись на наперед визначених критеріях якості, а також матриці для попарних порівнянь альтернатив. У межах даного етапу здійснюється обчислення власних чисел (ВЧ), а також власних векторів (ВВ).

Для виконання цих розрахунків була використана шкала, що відображає відносну важливість пріоритетів щодо один одного, яка дозволяє визначити експертну оцінку кожного критерію та альтернативи. Шкала представлена у таблиці 2.1.

Цей підхід забезпечує структуровану оцінку критеріїв і альтернатив, дозволяючи визначити їхню відносну важливість у контексті поставленої мети. Результати розрахунків допомагають обрати оптимальний варіант для реалізації системи, враховуючи всі ключові аспекти та вимоги.

Таблиця 2.1

### Шкала відносної важливості пріоритетів

Значення	Якісна характеристика
1	Рівноцінні елементи
2	Несуттєвий пріоритет
3	Слабкий пріоритет
4	Помірний пріоритет
5	Значний пріоритет
6	Істотний пріоритет
7	Сильний пріоритет
8	Дуже сильний пріоритет
9	Безумовний пріоритет

Щоб побудувати матрицю порівнянь критеріїв необхідно використати формули для обчислення ВЧ (2.1) та ВВ (2.2).

$$ВЧ = \sqrt[n]{\prod_{j=1}^n a_{ij}} \quad (2.1)$$

$$w_i = \frac{w_i}{\sum_{i=1}^n w_i} \quad (2.2)$$

де  $w_i$  – це ВЧ, а  $\sum_{i=1}^n w_i$  – сума цих ВЧ, що обчислюються для конкретного критерію.

Матриця критеріїв, що наведена далі ілюструє важливість кожного з перелічених вище критеріїв у відношенні до інших.

A =	K1	1	2	1/3	1/4	1/5	1/6	1/7	1/8
	K2	1/2	1	1/4	1/3	1/5	1/6	1/7	1/8
	K3	3	4	1	2	1/2	3	1	1/2
	K4	4	3	1/2	1	1	1/3	1/4	1/5
	K5	5	5	2	1	1	1/4	1/3	1/4
	K6	6	6	1/3	3	4	1	1/2	1/3
	K7	7	7	1	4	3	2	1	1/2
	K8	8	8	2	5	4	3	2	1

$$w_i = (0,45; 0,35; 1,85; 0,75; 1,05; 2,35; 2,15; 2,45)$$

$$w_i = (0,027; 0,021; 0,145; 0,083; 0,105; 0,141; 0,180; 0,298)$$

Наступним етапом є побудова матриць парних порівнянь альтернатив для кожного з визначених критеріїв, а також відносно головної мети.

Цей процес дозволяє оцінити, наскільки кожна альтернатива відповідає конкретному критерію, і визначити її відносну важливість у контексті досягнення головної цілі.

Критерій «Актуальність»

	A1	A2	A3	A4
A1	1	3	1/2	1/4
A2	1/3	1	1/5	1/7
A3	2	5	1	1/3
A4	4	7	3	1

$$w_i = (0.39, 0.33, 1.73, 0.66)$$

$$w_i = (0.04, 0.03, 0.16, 0.06)$$

## Критерій «Достовірність»

	A1	A2	A3	A4
A1	1	1/2	1/3	1/5
A2	2	1	1/4	1/6
A3	3	4	1	1/2
A4	5	6	2	1

$$BЧ = (0.45, 0.35, 1.85, 0.75)$$

$$BВ = (0.05, 0.04, 0.18, 0.07)$$

## Критерій: «Якість»

	A1	A2	A3	A4
A1	1	1/3	1/5	1/7
A2	3	1	1/4	1/6
A3	5	4	1	1/3
A4	7	6	3	1

$$BЧ = (0.55, 0.45, 2.05, 0.85)$$

$$BВ = (0.06, 0.05, 0.20, 0.08)$$

## Критерій: «Наочність»

	A1	A2	A3	A4
A1	1	1/2	1/4	1/6
A2	2	1	1/3	1/5
A3	4	3	1	1/2
A4	6	5	2	1

$$BЧ = (0.65, 0.55, 2.25, 0.95)$$

$$BВ = (0.07, 0.06, 0.22, 0.09)$$

Критерій: «Коректність»

	A1	A2	A3	A4
A1	1	1/3	1/5	1/7
A2	3	1	1/4	1/6
A3	5	4	1	1/3
A4	7	6	3	1

ВЧ = (0.75, 0.65, 2.45, 1.05)

ВВ = (0.08, 0.07, 0.24, 0.10)

Критерій: «Швидкодія»

	A1	A2	A3	A4
A1	1	1/2	1/3	1/4
A2	2	1	1/2	1/3
A3	3	2	1	1/2
A4	4	3	2	1

ВЧ = (0.85, 0.75, 2.65, 1.15)

ВВ = (0.09, 0.08, 0.26, 0.11)

Критерій: «Функціональність»

	A1	A2	A3	A4
A1	1	1/3	1/4	1/6
A2	3	1	1/2	1/4
A3	4	2	1	1/3
A4	6	4	3	1

ВЧ = (0.95, 0.85, 2.85, 1.25)

ВВ = (0.10, 0.09, 0.28, 0.12)

Критерій: «Надійність»

	A1	A2	A3	A4
A1	1	1/2	1/3	1/5
A2	2	1	1/2	1/4
A3	3	2	1	1/2
A4	5	4	2	1

$$BЧ = (1.05, 0.95, 3.05, 1.35)$$

$$BВ = (0.11, 0.10, 0.30, 0.13)$$

На цьому етапі можна побудувати матрицю порівнянь альтернатив на основі раніше обчислених власних чисел (ВЧ) та власних векторів (ВВ). Ця матриця дозволяє інтегрувати всі отримані дані, враховуючи відносну важливість кожного критерію та альтернативи, і провести остаточний аналіз.

	Узагальнені пріоритети								
	0.027	0.021	0.145	0.083	0.105	0.141	0.180	0.298	
A1	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.11	0.091
A2	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10	0.081
A3	0.16	0.18	0.20	0.22	0.24	0.26	0.28	0.30	0.243
A4	0.06	0.07	0.08	0.09	0.10	0.11	0.12	0.13	0.113

Згідно з результатами, отриманими за допомогою методу аналізу ієрархій, найбільший пріоритет, рівний 0,243, має А3 — система на основі статистичного аналізу.

Таким чином, можна підсумувати, що саме такий тип інформаційної системи є оптимальним вибором для реалізації поставленої мети. Система на основі статистичного аналізу забезпечує ефективну обробку даних, їх аналіз та використання для досягнення головної цілі — створення системи освоєння іноземної мови з використанням статистичного аналізу.

### 2.3. Використання UML задля конструювання моделі системи

Уніфікована мова для моделювання (UML) включає набір взаємопов'язаних

діаграм. Вона створена задля підтримки розробників систем і програмного забезпечення у визначенні, конструюванні, візуалізації та документуванні програмних систем. Крім того, UML може застосовуватися для бізнес-моделювання [6].

UML відіграє важливу роль у розробці об'єктно-орієнтованого програмного забезпечення. Використовуючи графічні позначення, UML допомагає розробляти мржливі проекти, оцінювати архітектуру програмного забезпечення та документувати його [6].

Мета UML полягає у забезпеченні стандартної нотації, яку можуть використовувати та розуміти всі розробники систем. Вона була розроблена для широкого спектра застосувань, включаючи автоматизацію виробництва програмного забезпечення, покращення його якості, зменшення витрат і скорочення часу виходу на ринок. UML допомагає управляти складністю систем, які з кожним роком стають дедалі масштабнішими.

Основні цілі UML:

- Надати користувачам мову візуального моделювання для створення та обміну значущими моделями.
- Забезпечити можливість розширення та спеціалізації основних концепцій.
- Бути незалежною від конкретних мов програмування та процесів розробки.
- Надати основу для повного розуміння мови моделювання.

UML включає широкий спектр діаграм, що дозволяє враховувати різні перспективи системи. У процесі розробки програм задіяно багато зацікавлених сторін, кожна з яких орієнтована на окремі аспекти системи, що потребують різного рівня деталізації.

Типи діаграм UML:

Структурні діаграми

Ілюструють статичну структуру системи, її компоненти на різних рівнях

абстракції та взаємозв'язки між ними. Основні типи структурних діаграм:

- Діаграми класів;
- Діаграми компонентів;
- Діаграми розгортання;
- Діаграми об'єктів;
- Діаграми пакетів;
- Діаграми композитів;
- Діаграми кооперацій.

Діаграми поведінки відображають динаміку взаємодії компонентів системи, представлену як послідовність змін упродовж часу. Основні типи діаграм поведінки:

- Діаграми прецедентів;
- Діаграми активності;
- Діаграми станів;
- Діаграми комунікації;
- Діаграми послідовності;
- Діаграми взаємодії;
- Діаграми синхронізації.

UML є потужним інструментом для моделювання систем, забезпечуючи стандартизований підхід до їхнього проектування, аналізу та документування. Завдяки широкому спектру діаграм і концепцій, UML дозволяє враховувати різні аспекти системи, що робить її універсальним рішенням для розробників.

### *2.3.1. Діаграма варіантів використання*

Діаграма варіантів використання в UML забезпечує загальний огляд взаємодії користувачів системи (акторів) із системою, дозволяючи візуалізувати ключові аспекти її функціонування. Вона є корисним інструментом для командної роботи, оскільки сприяє обговоренню та візуалізації таких елементів:

- Сценарії: взаємодія системи або програмного забезпечення з організаціями, особами чи іншими системами.

- Завдання: дії, які система або програма дозволяють виконувати користувачам (акторам).
- Область діяльності системи: межі функціональності системи.

Діаграма варіантів використання не перевантажена деталями, що дозволяє надати схематичний огляд взаємодій між акторами, системою та варіантами використання. Вона є ефективним доповненням до текстового опису, допомагаючи візуалізувати структуру взаємодій.

На діаграмі:

- Варіанти використання позначаються овалами з назвами.
- Актори представлені фігурками.
- Участь акторів у системі показана лініями, що з'єднують їх із варіантами використання.
- Кордон системи відображається прямокутником, який охоплює варіанти використання.

Призначення діаграми варіантів використання:

- Ілюстрація взаємодії між системою та користувачами.
- Уточнення функціональних вимог системи.
- Визначення контексту системи та її вимог.
- Розробка моделі основного процесу.

Основні елементи діаграми варіантів використання:

1. Актори: Зовнішні користувачі, які взаємодіють із системою. Актори можуть бути особами, організаціями або іншими системами, що мають зв'язок із аналізованою системою.
2. Система: Набір дій і взаємодій між акторами та системою. Система може бути представлена як сценарій, що описує її функціональність.
3. Цілі: Завершальні результати, яких прагнуть досягти за допомогою варіантів використання. Діаграма відображає процеси та шляхи, які ведуть до досягнення цих цілей.
4. Асоціації: Зв'язки, що встановлюються між акторами та суміжними

варіантами використання. Вказують, які актори пов'язані з певними варіантами використання.

5. Межі системи: Визначені області, які встановлюють межі для варіантів використання системи, а такі, що знаходяться за межами системи, вважаються зовнішніми.

6. Пакети: Засоби для групування елементів діаграми в логічні кластери. Використовуються для організації діаграми та спрощення її сприйняття.

- Існують наступні види відношень в даній діаграмі:
- Узагальнення: Use Case "Пройти тестування" може бути узагальнений для різних типів тестів (наприклад, граматики, лексики, читання).
- Включення: Use Case "Ввести логін та пароль" включає в себе Use Case "Перевірити логін та пароль". Use Case "Обрати кількість тестів" включає в себе Use Case "Генерувати тести".
- Розширення: Use Case "Аналізувати результати" може розширювати Use Case "Отримати результати тестування".
- Асоціації: Актор "Користувач" асоційований з усіма Use Case, які стосуються взаємодії з системою.

Присутні на діаграмі наступні актори: Користувач, система

Функції користувача:

- Ввести логін та пароль.
- Вибрати книгу.
- Обрати кількість тестів.
- Пройти тестування.
- Отримати результати тестування.
- Аналізувати результати.
- Завершити використання веб-сайту.

Функції системи:

- Перевірити логін та пароль.

- Генерувати тести.

Діаграму зображено на рисунку 2.2.

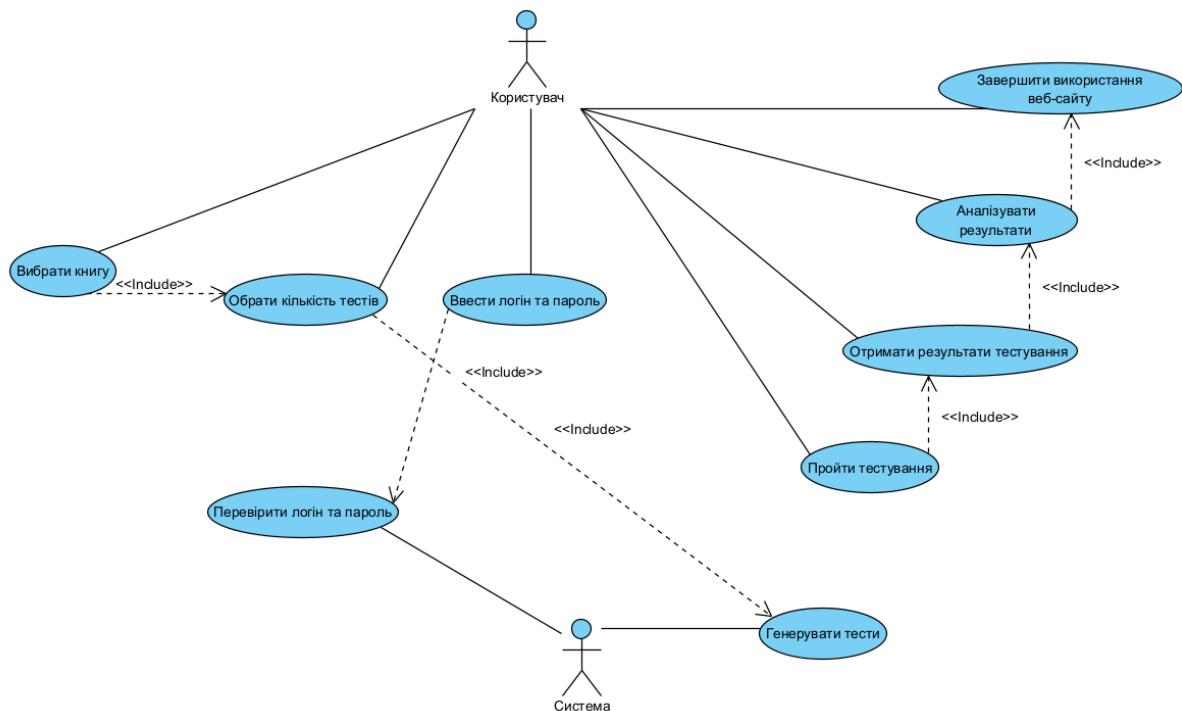


Рис. 2.2 Діаграма варіантів використання

### 2.3.2. Діаграма класів

Діаграма класів є візуальним інструментом, який використовується для моделювання та представлення структури об'єктно-орієнтованих систем. Вона відображає статичну архітектуру системи, демонструючи її основні складові елементи: класи, методи, атрибути та відношення між об'єктами [6].

Клас виступає як шаблон для створення об'єктів. Класи та об'єкти тісно пов'язані, і їх не можна розглядати окремо. У контексті об'єктно-орієнтованого програмування класи є основою, оскільки саме вони використовуються для створення об'єктів.

Клас визначає характеристики об'єкта, але сам по собі не є об'єктом. Класи задають типи для об'єктів, а об'єкти є інстанціями цих класів. Усі об'єкти, створені за допомогою одного класу, мають однакові атрибути (властивості) та поведінку

(методи).

Структура класу:

Клас складається з трьох основних частин:

- Назва класу – розміщується у верхній частині визначення класу.
- Атрибути – відображаються у другій частині. Тип атрибута – вказується після двокрапки. Атрибути представлені у вигляді змінних.
- Методи (операції) – показуються у третій частині. Методи представляють послуги, які клас надає. Тип даних, що повертає метод, вказується після двокрапки. Тип даних параметрів методу зазначається після їхніх назв, розділений двокрапкою.

Діаграму класів відображено на рисунку 2.3.

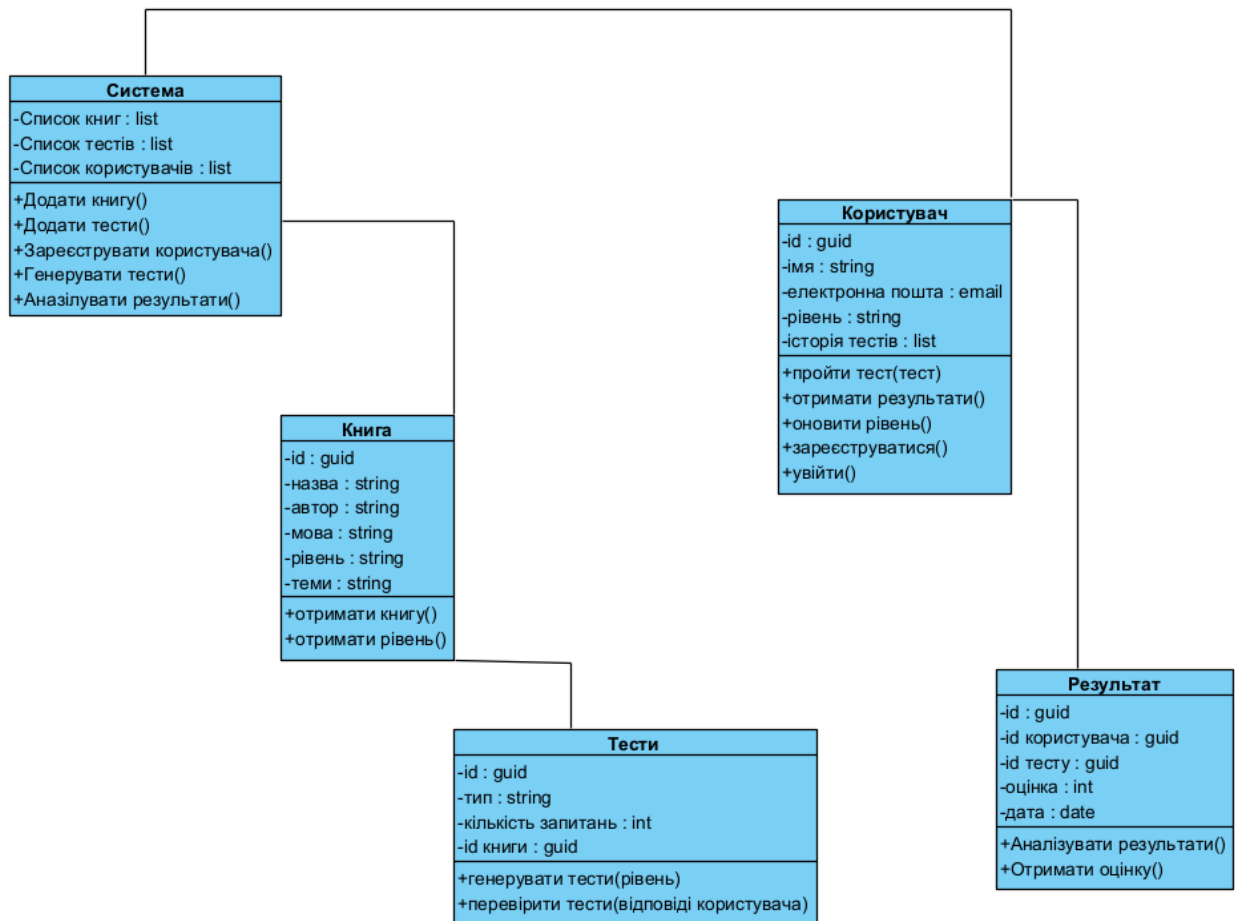


Рис. 2.3 Діаграма класів

Більш детальний опис діаграми подано в описі далі.

Книга:

- Атрибути: ID, Назва, Автор, Мова, Рівень, Теми
- Методи: ОтриматиТеми(), ОтриматиРівень()

Тест:

- Атрибути: ID, Тип, КількістьЗапитань, Рівень, КнигаID
- Методи: ГенеруватиТест(рівень),

ПеревіритиТести(ВідповідіКористувача)

Користувач:

- Атрибути: ID, Ім'я, ЕлектроннаПошта, Рівень, ІсторіяТестів
- Методи: ПройтиТест(Тест), ОтриматиРезультати(), ОновитиРівень(),

зареєструватися, увійти

Результат:

- Атрибути: ID, КористувачID, ТестID, Оцінка, Дата
- Методи: АналізуватиРезультати(), ОтриматиОцінку()

Система

- Атрибути: СписокКниг, СписокТестів, СписокКористувачів
- Методи: ДодатиКнигу(Книга), ДодатиТест(Тест),

ЗареєструватиКористувача(Користувач), ГенеруватиТести(Книга),

АналізуватиРезультати(Користувач)

### 2.3.3. Діаграма станів

Діаграма станів показує динаміку системи через серію станів, які проходить об'єкт у відповідь на різні події протягом його життєвого циклу. Вона також демонструє реакції об'єкта на ці події [7].

Основні елементи діаграми станів:

1. Стан: Визначає період у житті об'єкта, коли він задовольняє певні критерії, виконує конкретні дії або очікує на зовнішні події.
2. Подія: Специфічна подія, яка може спричинити зміну стану об'єкта.

3. Перехід: Процес зміни станів, який відбувається, коли певні умови та події задовольняються. Перехід призводить до виконання дій та переходу об'єкта до наступного стану.

4. Дія: Операції або обчислення, які виконуються об'єктом. Дії можуть включати створення, видалення інших об'єктів або відправлення сигналів.

Особливості діаграми станів: Вона дозволяє наочно показувати поведінку об'єкта протягом його життєвого циклу.

Діаграма станів є корисною для моделювання складних систем, де важливо враховувати зміни станів об'єктів у відповідь на події.

Діаграму станів для розроблюваної системи представлено на рисунку 2.4.

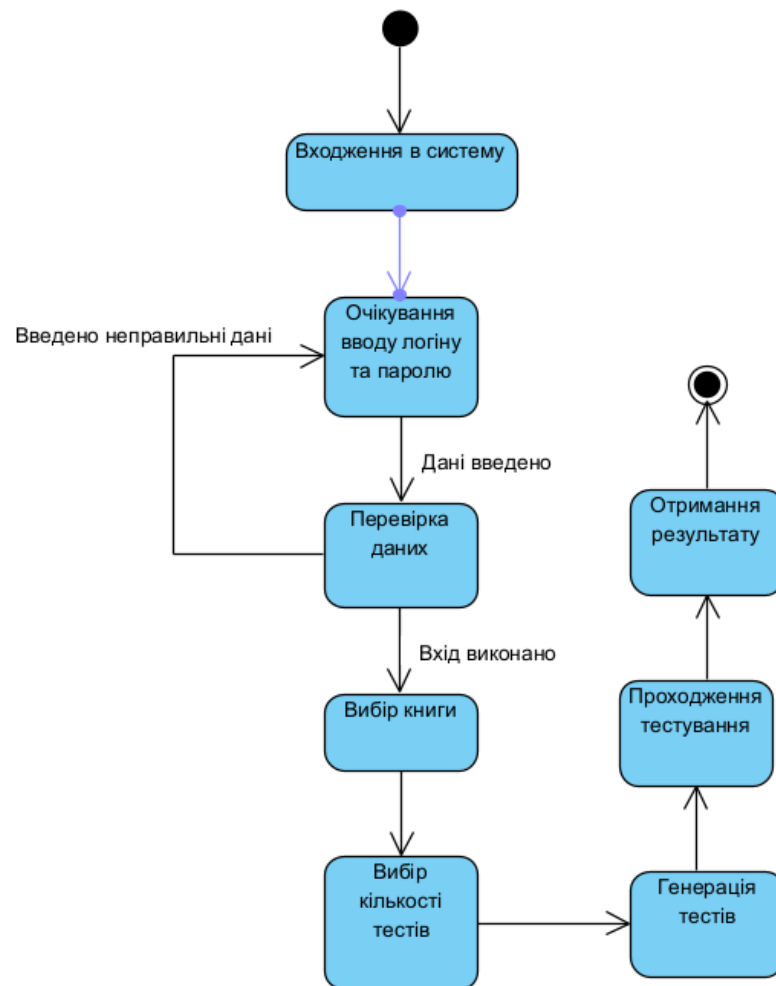


Рис. 2.4 UML Діаграма станів

Діаграма станів відображає процес взаємодії користувача із системою.

Користувач починає з входу в систему, вводячи логін і пароль, які перевіряються на правильність. Після успішної авторизації користувач обирає книгу, кількість тестів, проходить тестування та отримує результати. У разі помилки авторизації користувач повертається до введення логіна і пароля.

#### *2.3.4. Діаграма діяльності*

Діаграма діяльності в UML є важливим інструментом для моделювання динаміки системи, оскільки вона детально описує переходи між різними діями. Ця діаграма фактично є деталізованою блок-схемою, яка ілюструє послідовність того як будуть виконуватися дії.

Основні характеристики діаграми діяльності:

- Вона демонструє, як різні дії взаємодіють між собою для надання певних послуг.
- Діаграма відображає послідовність операцій, які необхідно виконати для досягнення певної події або результату.
- Операції синхронізуються для досягнення остаточної мети, що дозволяє візуалізувати логіку процесу.

Діаграма показана на рисунку 2.5.



Рис. 2.5 Діаграма діяльності

Вона показує послідовність дій користувача в системі. Користувач починає з введення логіна та пароля, після чого вибирає книгу, налаштовує кількість тестів, проходить тестування та отримує результати. На цій діаграмі всі дії виконуються послідовно, без розгалужень чи умовних переходів.

### 2.3.5. Діаграма розгортання

Діаграма розгортання в UML використовується для моделювання фізичних аспектів об'єктно-орієнтованих систем. Вона ілюструє, як виконуються вузли обробки та які компоненти функціонують на цих вузлах. Ця діаграма є структурною і призначена для представлення статичної структури системи.

Використання діаграми розгортання:

- Визначення взаємодії з іншими системами: дозволяє зрозуміти, як система інтегрується з зовнішніми компонентами.
- Оцінка надійності системи: допомагає аналізувати стабільність і безперебійність роботи системи.
- Розуміння підключення та взаємодії: визначає, хто і як буде підключатися до системи та взаємодіяти з нею.
- Визначення програмного забезпечення: описує операційні системи, протоколи та інше програмне забезпечення, яке буде використовуватися.
- Аналіз апаратних і програмних засобів: дозволяє зрозуміти, з якими засобами працюватимуть користувачі.
- Планування контролю системи: забезпечує управління та моніторинг системи після її розгортання.
- Забезпечення безпеки: допомагає врахувати аспекти захисту даних і доступу до системи.

Діаграму розгортання для система освоєння іноземної мови з використанням статистичного аналізу представлено на рисунку 2.6.

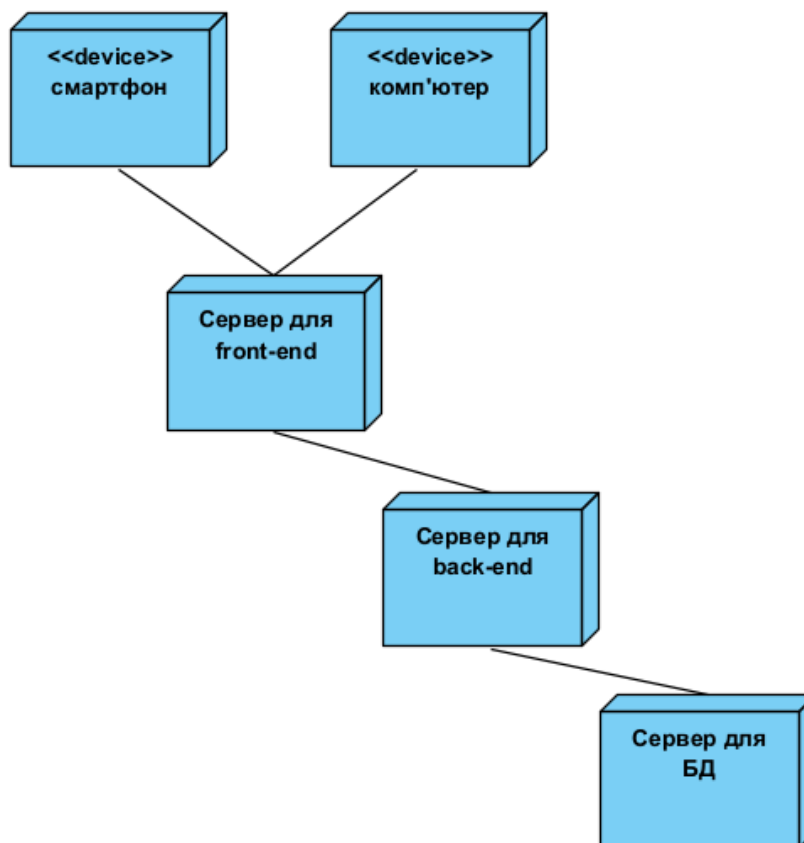


Рис. 2.6 Діаграма розгортання

### 2.3.6. Діаграма послідовності

В UML для того щоб продемонструвати процеси та взаємодії між об'єктами, а також те, як ці процеси розгортаються у часі використовується діаграма послідовності. Вона слугує інструментом для візуалізації порядку виконання дій і передачі повідомлень між учасниками взаємодії.

Основні функції діаграми послідовності:

- відображення взаємодій у межах співпраці;
- ілюстрація високорівневих взаємодій між системою та користувачами, підсистемами або зовнішніми системами.

Застосування діаграми послідовності включає:

- моделювання взаємодій між об'єктами всередині системи;
- відображення взаємодій між окремими екземплярами об'єктів;
- створення моделей взаємодій між об'єктами;

- візуалізацію загальних взаємодій, включаючи всі можливі сценарії обміну даними.

Діаграма представлена на рисунку 2.7.

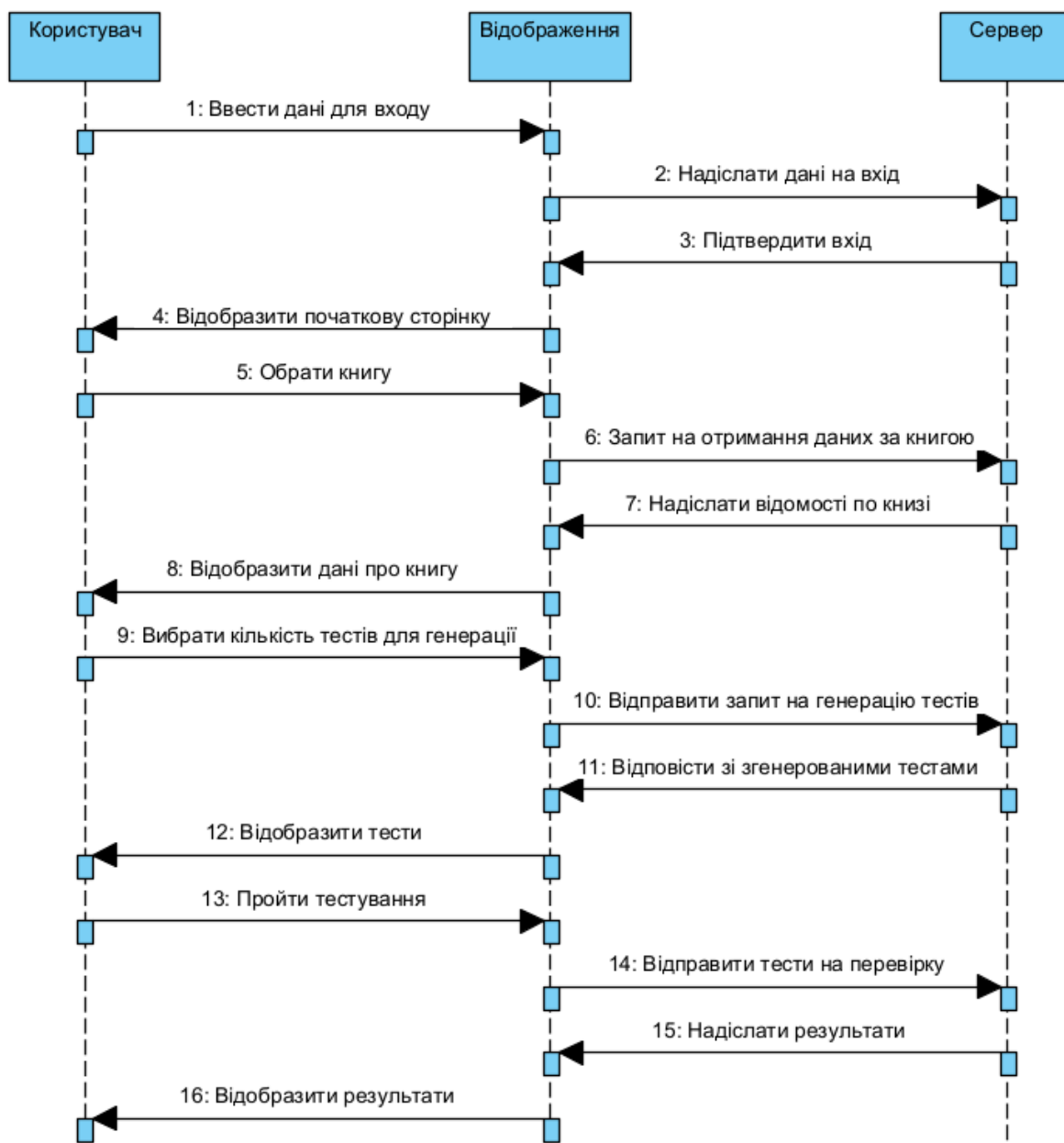


Рис. 2.7. Діаграма послідовності

## 2.4. Мета розроблення інформаційної системи

Метою створення системи є вирішення проблеми, пов'язаної з ефективним освоєнням іноземної мови, й зокрема автоматизація процесу створення тестів для оцінки знань користувачів на основі статистичного аналізу їхніх результатів. Система

забезпечує персоналізований підхід до навчання, мінімізуючи витрати часу та людських ресурсів.

Мета для категорій користувачів:

Користувач — це особа, яка використовує систему для спрощення процесу вивчення матеріалу. Система дозволяє користувачу обрати книгу, пройти тестування, отримати результати та проаналізувати свої знання. Завдяки автоматизованому створенню тестів на основі обраної книги, користувач може зосередитися на своїх слабких місцях і покращити рівень володіння мовою. Система також надає можливість отримати повний аналіз результатів для наступного вдосконалення.

Адміністратор — це особа, яка відповідає за управління системою та її користувачами. Система дозволяє кері адміністратору додавати книги, редагувати їхній контент, а також автоматично генерувати тести на основі книг за допомогою статистичного аналізу. Він може переглядати результати тестування користувачів, аналізувати їхню продуктивність та надавати рекомендації для подальшого навчання. Крім того, система забезпечує можливість адміністрування користувачів, включаючи їх додавання, редагування та видалення.

## **2.5. Місце застосування системи**

Система може бути застосована в різних сферах, де необхідно ефективно освоювати іноземну мову, проводити тестування знань та аналізувати результати для подальшого вдосконалення. Основні місця застосування включають:

1. Освітні установи: Наприклад школи, університети, мовні курси. Система може використовуватися для навчання студентів, автоматизації процесу створення тестів, оцінки знань та надання персоналізованих рекомендацій для покращення рівня володіння мовою.

2. Корпоративне навчання: Система може бути впроваджена для навчання співробітників, які працюють у міжнародному середовищі, де знання іноземної мови є критично важливим. Вона дозволяє проводити регулярне тестування, аналізувати прогрес та адаптувати навчальні матеріали до потреб працівників.

3. Самостійне навчання: Система може бути використана людьми, які самостійно вивчають іноземну мову. Вона забезпечує доступ до книг, тестів та аналізу результатів, що допомагає ефективно організувати процес навчання.

4. Мовні школи та центри сертифікації: Система може бути застосована для підготовки до сертифікаційних іспитів (наприклад, TOEFL, IELTS, DELF, Goethe-Zertifikat), автоматизуючи процес створення тестів та аналізу результатів.

5. Онлайн-платформи навчання: Система може бути інтегрована в онлайн-платформи для дистанційного навчання, забезпечуючи користувачів доступом до навчальних матеріалів, тестів та результатів у зручному форматі.

## **2.6. Постановка та обґрунтування проблеми**

Освоєння іноземної мови є важливим аспектом особистісного та професійного розвитку, особливо в умовах глобалізації, де знання мов відкриває доступ до повністю нових можливостей у роботі, навчанні, та відпочинку. Проте традиційні методи навчання часто стикаються з низкою проблем, які знижують ефективність процесу:

- Нестача персоналізації: стандартні навчальні програми не враховують індивідуальні потреби та слабкі сторони кожного учня.
- Складність оцінки прогресу: користувачі часто не мають доступу до детального аналізу своїх результатів, що ускладнює визначення напрямків для вдосконалення.
- Витрати часу: ручне створення тестів та аналіз результатів займає багато часу, що може бути неефективним для викладачів та адміністраторів.
- Мотивація: відсутність інтерактивного підходу може знижувати зацікавленість у навчанні.

Система "Інтелектуальна система освоєння іноземної мови з використанням статистичного аналізу" допоможе вирішити ці проблеми завдяки таким функціям:

- Автоматизація процесу навчання: система генерує тести на основі книг та аналізує результати, що дозволяє користувачам зосередитися на навчанні, а не на організаційних аспектах.

- Персоналізований підхід: статистичний аналіз результатів тестування дозволяє адаптувати навчальні матеріали до потреб кожного користувача.
- Ефективність: автоматизоване створення тестів та аналіз результатів значно скорочує витрати часу для викладачів та адміністраторів.
- Мотивація: інтерактивний процес навчання та детальний аналіз результатів сприяють підвищенню зацікавленості користувачів.

Проте, під час розроблення та впровадження системи можуть виникнути певні труднощі:

1. Фінансування: початкові інвестиції у розробку системи можуть бути значними, особливо для інтеграції статистичного аналізу та автоматизації.
2. Технічна реалізація: система потребує залучення кваліфікованих спеціалістів для розробки алгоритмів генерації тестів та аналізу результатів.
3. Стабільність роботи: важливо забезпечити безперебійну роботу системи, особливо при великій кількості користувачів.
4. Конкуренція: на ринку існують аналогічні системи, такі як Duolingo, Babbel, Rosetta Stone, які мають значний досвід і популярність.

Серед аналогів варто виділити Duolingo, Babbel, Rosetta Stone, Memrise. Ці системи пропонують інтерактивне навчання, але не завжди враховують індивідуальні потреби користувачів або автоматизують аналіз результатів на основі зібраних й проаналізованих статистичних даних.

Враховуючи вищезазначене, розроблення та впровадження системи "Інтелектуальна система освоєння іноземної мови з використанням статистичного аналізу" є доцільним. Система допоможе вирішити ключові проблеми, пов'язані з навчанням іноземної мови, забезпечить персоналізований підхід до навчання та автоматизує процес оцінки знань, що зробить її корисною для освітніх установ, корпоративного навчання та індивідуальних користувачів.

## **2.7. Очікувані ефекти від впровадження системи**

Для повного розуміння впливу впровадження системи на увесь процес освоєння

іноземної мови важливо проаналізувати ключові аспекти, які активізуються після її запровадження. Система спрямована на автоматизацію навчання, персоналізацію процесу та підвищення ефективності оцінки знань користувачів. Це посприяє тому, що процес освоєння матеріалу буде супроводжений інтерактивними активностями, що будуть яскравіше відображати прогрес.

Ключові аспекти:

Фінансовий аспект відображає вплив системи на фінансові результати організацій, які її використовують. Наприклад, мовні школи можуть залучити більше клієнтів завдяки автоматизації та персоналізації навчання.

1. Економічний аспект відображає потребу в початкових інвестиціях для розробки та впровадження системи, та загальну оцінку ефективності цих інвестицій у довгостроковій перспективі. Система дозволяє зменшити витрати на ручне створення тестів та аналіз результатів.

2. Часовий аспект демонструє, що система значно скорочує час, який потрібен для генерування тестів, опрацювання результатів та підготовки навчальних матеріалів. Це дозволяє користувачам і викладачам зосередитися на навчанні, а не на організаційних аспектах.

3. Технологічний аспект покращує якість навчального процесу за допомогою використання модерних технологій, таких як статистичний аналіз, автоматизація та інтерактивні інструменти.

4. Науково-технічний аспект показує, що система є інноваційним та технологічним рішенням, що поєднує статистичний аналіз із навчанням. Це сприяє розвитку нових підходів до вивчення мов та оцінки знань.

5. Соціальний аспект сприяє підвищенню доступності якісного навчання для широкого кола користувачів, включаючи студентів, працівників та індивідуальних учнів. Вона також підвищує мотивацію до навчання завдяки персоналізованому підходу.

Оцінка ефектів, що здійснюють ефект чи відносяться до системи, відображена у таблиці 2.2.

Таблиця 2.2

**Оцінка ефектів інтелектуальної системи**

Ціль	Ефект	Одиниці виміру	Значення оцінки
Наукова	Освітній	Високий рівень ефекту	Важливе значення
Технічна	Науково-технічний	Дуже високий рівень ефекту	Ключове значення
Економічна	Економічний	Високий рівень ефекту	Важливе значення
Фінансова	Фінансовий	Середній рівень ефекту	Суттєве значення
Соціальна	Соціальний	Високий рівень ефекту	Ключове значення
Часова	Часовий	Дуже високий рівень ефекту	Важливе значення

**2.8. Розроблення концептуальної моделі системи**

Види вхідної інформації:

- Дані для реєстрації та входу: Ім'я користувача, пароль, електронна пошта;
- Рівень володіння мовою (A1, A2, B1, B2, C1, C2);
- Історія тестів.

Дані про книги:

- Назва книги;
- Автор;
- Мова книги;
- Рівень складності;
- Теми, які охоплює книга;

Дані про тести:

- Тип тесту (граматика, лексика, читання, аудіювання);
- Кількість запитань;
- Рівень складності;
- Відповіді користувача.

Результати тестування:

- Оцінка;
- Дата проходження тесту;
- Аналіз сильних і слабких сторін користувача.

Вихідні дані:

- Доступ до книг: Користувач може переглядати список доступних книг, обирати їх для навчання.
- Згенеровані тести: Система автоматично створює тести на основі обраної книги та рівня складності.
- Результати тестування: Оцінка користувача.

Система пропонує книги, тести або теми для вдосконалення знань на основі результатів тестування.

Опис вхідної інформації:

- Дані для реєстрації та входу: Інформація вводиться користувачем під час створення облікового запису або входу в систему.
- Вимоги до пароля: Не менше 11 символів, не менше однієї великої літери, і хоч одна цифра.
- Дані про користувача: адреса електронної скриньки повинна відповідати формату (наявність символу "@" та домену). Також вони включають особисту інформацію та рівень володіння мовою, який оновлюється чи залишається минулим після проходження тестів.
- Дані про книги: Включають навчальні матеріали, які використовуються для створення тестів.
- Дані про тести: Включають тип тесту, кількість запитань та відповіді користувача, які аналізуються системою.

Опис вихідної інформації:

- Доступ до книг: Користувач отримує доступ до списку книг, які відповідають його рівню володіння мовою.
- Згенеровані тести: Тести створюються автоматично на основі обраної книги та рівня складності.
- Результати тестування: Користувач отримує оцінку та детальний аналіз своїх результатів.

Концептуальна модель системи включає вхідну інформацію (дані користувачів,

книги, тести) та вихідну інформацію (результати тестування, рекомендації, доступ до навчальних матеріалів). Система автоматизує процес навчання, забезпечуючи персоналізований підхід до освоєння іноземної мови.

### **Висновок до другого розділу**

У даному розділі було проведено вичерпний системний аналіз розроблюваної інтелектуальної системи освоєння іноземної мови з використанням статистичного аналізу. Описано основну проблему, яку вирішує система, та проведено аналіз ієрархій для вибору типу системи. За допомогою UML-діаграм здійснено детальний опис системи, за допомогою діаграм станів, діяльності, розгортання та класів. Представлено ключову мету створення системи, її сферу застосування та обґрунтування необхідності розробки та впровадження. Визначено очікувані ефекти, що очікуються від впровадження такої системи, які охоплюють економічний, соціальний, технологічний та інші аспекти. Сформульовано концептуальну модель системи, яка включає основні задачі, архітектуру, вхідні та вихідні дані. Також описано вимоги до системи, які забезпечують її ефективне функціонування та досягнення поставлених цілей.

## РОЗДІЛ 3

### Методи та засоби вирішення проблеми

#### 3.1. Вибір та обґрунтування методів розв'язання задачі

Створення інтелектуальної системи для освоєння іноземної мови вимагає застосування сучасних підходів, які забезпечують високу ефективність навчання, адаптивність до потреб користувача та можливість аналізу великих масивів даних. Одним із ключових методів, який використовується задля розв'язання задачі, є статистичний аналіз. Цей підхід дозволяє виявляти закономірності у навчальному процесі, оптимізувати навчальні матеріали та адаптувати систему до рівня знань користувача.

Основою для вибору статистичного аналізу є його здатність працювати з великими обсягами даних, отриманих під час взаємодії користувача із системою. У процесі навчання система накопичує значну кількість інформації, яка включає дані про успішність виконання завдань, частоту помилок, час, витрачений на опрацювання матеріалу, та рівень засвоєння різних аспектів мови, таких як лексика, граматики чи вимова. Використання статистичних методів дозволяє систематично аналізувати ці дані, визначати ключові закономірності та тенденції, а також виявляти тонкі місця у процесі навчання для кожного користувача. Наприклад, аналіз частоти помилок у конкретних граматичних конструкціях може сигналізувати про необхідність додаткового повторення відповідного матеріалу, а оцінка часу виконання завдань допомагає виявити складність окремих етапів навчання та адаптувати їх до індивідуальних потреб.

Методи кластеризації та регресійного аналізу є особливо корисними для групування користувачів за рівнем знань, стилем навчання або іншими характеристиками, що дозволяє створити персоналізовані траєкторії навчання. Наприклад, кластеризація може допомогти виділити групи користувачів, які мають схожі проблеми у вивченні мови, та запропонувати для них спеціально розроблені навчальні програми. Регресійний аналіз, у свою чергу, дозволяє прогнозувати прогрес користувача, враховуючи його попередні результати, що дає змогу системі динамічно

змінювати складність завдань або додавати нові елементи для підтримання мотивації. Крім того, статистичний аналіз може бути використаний для оцінки ефективності різних навчальних стратегій, таких як повторення матеріалу з певною періодичністю або використання інтерактивних вправ для закріплення знань. Це дозволяє системі постійно вдосконалюватися, адаптуючи методи навчання до потреб кожного користувача та забезпечуючи максимальну ефективність процесу освоєння іноземної мови.

Для реалізації поставленої задачі доцільно використовувати такі методи статистичного аналізу, як кореляційний аналіз, кластеризація та байєсівські моделі. Кореляційний аналіз дозволяє визначити взаємозв'язок між різними параметрами навчального процесу, наприклад, між частотою повторення матеріалу та рівнем засвоєння знань. Кластеризація допомагає групувати користувачів за рівнем знань, стилем навчання або іншими характеристиками, що сприяє створенню адаптивних навчальних програм. Байєсівські моделі, у свою чергу, дозволяють враховувати невизначеність у процесі навчання, що особливо важливо для прогнозування результатів.

Важливим аспектом є інтеграція статистичних методів із сучасними технологіями машинного навчання. Наприклад, використання нейронних мереж у комбінації зі статистичним аналізом дозволяє створювати системи, які не лише адаптуються до індивідуальних потреб користувача, але й прогнозують його поведінку у процесі освоєння матеріалу. Це відкриває можливість для розробки інтелектуальних алгоритмів, які автоматично коригують навчальні матеріали та завдання відповідно до рівня підготовки користувача.

Отже, використання статистичного аналізу в інтелектуальній системі для вивчення іноземної мови є цілком виправданим і логічним рішенням. Завдяки цьому підходу можна досягти високого рівня адаптивності системи, що дозволяє враховувати особливості й побажання кожного користувача, його темп навчання та слабкі сторони. Статистичний аналіз відкриває можливості для персоналізації навчального процесу, що робить його не лише ефективним, але й комфортним для

користувача. Адже система, яка аналізує дані про прогрес і помилки, може пропонувати саме ті завдання, які потрібні для подолання труднощів і швидшого досягнення результату.

Крім того, поєднання статистичних методів із сучасними технологіями машинного навчання створює потужний інструмент для розробки інноваційної системи, яка постійно вдосконалюється. Машинне навчання допомагає системі не лише адаптуватися до потреб користувача, але й прогнозувати його подальші кроки у навчанні, пропонуючи оптимальні стратегії для досягнення поставлених цілей. Така інтеграція технологій дозволяє створити систему, яка не просто навчає, а й активно підтримує користувача на кожному етапі, роблячи навчання максимально результативним і приємним. У результаті, це забезпечує не лише глибоке засвоєння знань, але й підвищує мотивацію до вивчення, трансформуючи процес освоєння мови на захоплючу, ефективну і продуктивну подорож.

### **3.2. Вибір та обґрунтування засобів розв'язання задачі**

Інтелектуальна система для вивчення іноземної мови з використанням статистичного аналізу найдоцільніше реалізувати у вигляді веб-сайту, оскільки основна взаємодія із тим, хто користуватиметься відбуватиметься через мережу Інтернет. Такий підхід забезпечує зручність і доступність використання системи, дозволяючи працювати з нею на різних пристроях — телефонах, планшетах, ноутбуках чи комп'ютерах — незалежно від платформи, на якій запускається сайт.

Архітектура системи складається з трьох основних компонентів: front-end, back-end і база даних.

Front-end відповідає за те як буде виглядати сайт та за взаємодію з користувачем. Для його реалізації варто розглянути популярні технології, такі як Angular, React і Vue.js:

1. Angular — платформа для створення ефективних і масштабованих односторінкових додатків, побудована на TypeScript. Вона пропонує компонентну архітектуру, що ідеально підходить для масштабованих проєктів,

вбудовані бібліотеки для роботи з реактивними формами, маршрутизацією та зв'язком між клієнтом і сервером, широкий набір інструментів для зручної розробки та тестування продукту. Angular підтримує зворотну сумісність, що дозволяє легко оновлювати фреймворк до нових версій. Станом на 2022 рік понад 1,7 мільйона розробників використовують Angular [8].

2. React — гнучка та ефективна JavaScript-бібліотека для створення інтерфейсів користувача. Вона дозволяє поступово інтегруватися в проект, додаючи інтерактивність до HTML-коду або використовуючи її для складних додатків. React забезпечує декларативний підхід до розробки, що полегшує створення динамічних інтерфейсів [9].

3. Vue.js — JavaScript-фреймворк для створення інтерфейсів користувача. Його декларативна та компонентна моделі програмування базуються на HTML, CSS і JavaScript, що дозволяє швидко та ефективно розробляти інтерфейси будь-якої складності [10].

Back-end — це серверна частина веб-сайту, яка відповідає за приймання, обробку даних, виконання обчислень і складних маніпуляцій. Вибір технології для реалізації back-end залежить від вимог проекту, але серед найпопулярніших рішень можна виділити Django, .NET і Laravel:

1. Django — високорівневий веб-фреймворк, написаний на Python, який сприяє швидкій розробці та чистому дизайну. Основні переваги Django це зручність і швидкість розробки завдяки вбудованим інструментам і бібліотекам. Розроблений досвідченими програмістами, він вирішує більшість типових завдань веб-розробки без необхідності додаткових зусиль. Підтримка масштабованості та безпеки, що робить його ідеальним для створення складних веб-додатків [11].

2. .NET — безкоштовне кросплатформне середовище розробки з відкритим кодом, яке дозволяє створювати різноманітні типи додатків. Для веб-розробки використовується розширення ASP.NET, яке надає основний інструментарій для обробки веб-запитів мовою програмування C#. Бібліотеки для популярних

шаблонів, таких як MVC (Model-View-Controller). Вбудовану систему автентифікації та авторизації, яка включає стандартні методи для обробки входу в систему, підтримку зовнішніх сервісів (Google, Facebook тощо) і багатофакторну автентифікацію. Високу продуктивність і надійність, що робить.NET популярним вибором для корпоративних рішень [12].

3. Laravel — PHP-фреймворк із виразним і елегантним синтаксисом, який забезпечує зручність і ефективність розробки. Основні переваги Laravel це те що він надає готову структуру для створення додатків, дозволяючи зосередитися на основних завданнях. Підтримує модульне та інтеграційне тестування, що спрощує перевірку функціональності. Має продуману систему *dependence injection* і високий рівень абстракції, що полегшує проектування бази даних. Підходить для створення як невеликих, так і масштабованих веб-додатків [13].

База даних — це місце для зберігання опрацьованих даних, які можуть бути отримані, змінені або додані в будь-який момент за допомогою запитів, процедур чи транзакцій. Для визначення оптимальної бази даних у межах даного проекту розглядаються варіанти, доступні в хмарному сервісі Amazon RDS.

Служба реляційних баз даних Amazon (Amazon RDS) — це сервіс, частина загального клауд провайдера AWS, що спрощує налаштування, управління та масштабування лише реляційних баз даних у хмарі [14]. Серед найпопулярніших варіантів, доступних у Amazon RDS, виділяються: Amazon Aurora, MariaDB та MSSQL Server.

Amazon Aurora — механізм реляційної бази даних, що забезпечує керування у всій мірі, сумісний із MySQL і PostgreSQL. Основні переваги якої це висока продуктивність і масштабованість, що робить її ідеальною для великих проектів, сумісність із MySQL і PostgreSQL, що дозволяє використовувати плюси кожної з систем, максимальний обсяг даних для зберігання — до 128 тебібайтів, що забезпечує підтримку великих обсягів інформації, розподілене сховище даних, яке гарантує надійність і безпеку [15].

MariaDB — одна з найпопулярніших реляційних баз даних із відкритим кодом.

Основні характеристики якої це висока продуктивність і стабільність, що робить її надійним вибором для різних типів проектів, відкритість і підтримка декількох версій баз даних у середовищі Amazon RDS, підтримка широкого спектра функцій, включаючи складні запити, транзакції та реплікацію даних [16].

MicrosoftSQL Server — одним з перших виборів для корпоративних ІТ-середовищ. Основні особливості це – серверний механізм баз даних — основний модуль, який забезпечує контроль над зберіганням, обробкою та безпекою даних, підтримка реляційного механізму для управління файлами бази даних, таблицями, індексами та сторінками, можливість створення та виконання збережених процедур, функцій, тригерів та інших об'єктів бази даних, підтримка бізнес-аналітики та обробки транзакцій, що робить її ідеальною для складних корпоративних рішень.

Окрім основних інструментів, у процесі розробки веб-сайту використовуються технології, які є невід'ємною частиною сучасного веб-додатка, такі як:

1. **HTML** — основа структури веб-контенту, яка визначає значення та організацію елементів на веб-сторінці. HTML є базовим будівельним блоком Інтернету, що дозволяє створювати та публікувати контент, з'єднувати сторінки через гіпертекстові посилання та працювати у поєднанні з CSS і JavaScript для стилізації та функціональності [17].

2. **CSS** — мова стилів, яка визначає зовнішній вигляд веб-документів. CSS дозволяє задавати, як елементи веб-сторінки мають виглядати на різних носіях (екрані, папері тощо). Це стандартизована мова, яка забезпечує гнучкість у дизайні та адаптивність інтерфейсів. Хоча існували версії CSS1, CSS2.1 і CSS3, офіційної версії CSS4 так і не було створено [18].

3. **Bootstrap** — найпопулярніший фреймворк для створення адаптивних і мобільно-орієнтованих веб-сайтів. Розроблений командою Twitter, Bootstrap пропонує готові шаблони дизайну, які можна використовувати або налаштовувати. Він забезпечує сумісність із сучасними браузерами та дозволяє швидко створювати інтерфейси, які коректно відображаються на різних пристроях й не потребують глибоких знань технології як такої [19][20].

Ці технології забезпечують функціональність, естетичність і доступність веб-додатків на різних платформах та пристроях.

### **3.3. Технічні характеристики обраних програмних засобів розроблення**

Для створення інформаційної системи у вигляді веб-сайту, яка використовує статистичний аналіз для вивчення іноземних мов, необхідно провести ретельний аналіз доступних програмних засобів, що відповідають поставленим цілям. Такий аналіз включає оцінку функціональних можливостей, продуктивності, зручності використання, а також відповідності технічним і бізнес-вимогам проекту.

Оскільки система передбачає використання мережі Інтернет і різноманітних браузерів, було вирішено створити веб-сайт, який включатиме:

- Зовнішню частину сайту (front-end) — інтерфейс, що відображається в браузері та з яким взаємодіє користувач.
- Серверну частину (back-end) — компонент, який обробляє запити користувача, що надходять із браузера, виконує дії з отриманими даними та повертає необхідну інформацію.
- Базу даних — сховище, яке зберігатиме всю необхідну інформацію, таку як дані про користувачів, книги тощо.

Вибір веб-сайту як форми реалізації системи обумовлений його універсальністю та зручністю. Такий підхід має переваги для всіх сторін:

- Користувачі отримують миттєвий доступ до системи без обмежень, пов'язаних із платформою чи пристроєм.
- Власник/адміністратор/менеджер системи отримує просте управління сайтом через інтерфейс адміністратора, що спрощує адміністрування та підтримку.

Таким чином, веб-сайт є оптимальним рішенням для реалізації системи, забезпечуючи доступність, зручність і ефективність як для користувачів, також й для менеджерів.

Після аналізу доступних варіантів для реалізації front-end частини веб-сайту було

обрано Angular.

Angular має багату історію розвитку. Вперше представлений Google у 2009 році як основа на базі JavaScript для створення мобільних і настільних веб-додатків, Angular зазнав значних змін. У 2016 році Google повністю переписав і перевипустив Angular, щоб відповідати зростаючим вимогам сучасного Інтернету [21].

На сьогодні Angular пропонує такі переваги:

1. Функціонал із коробки — Angular забезпечує готові інструменти та бібліотеки для швидкого старту розробки.
2. TypeScript — використання цієї мови програмування додає типізацію, що підвищує надійність і зручність коду.
3. Послідовність — Angular є цілісним адаптивним фреймворком, який забезпечує єдиний підхід до створення компонентів, модулів і служб, на відміну від більш гнучкого, але менш структурованого React.
4. Продуктивність — оптимізовані механізми рендерингу та обробки даних забезпечують високу швидкість роботи додатків.
5. Підтримка та гнучке оновлення — Angular активно підтримується Google, що гарантує регулярні оновлення та вдосконалення.
6. Модульна структура — дозволяє розробникам легко організувати код і масштабувати додатки.
7. Angular Material — набір готових компонентів і модулів інтерфейсу користувача, які відповідають принципам Material Design від Google, що спрощує створення сучасних і зручних інтерфейсів.

Серед недоліків Angular можна виділити такі аспекти:

- Високий поріг входу — для початку роботи з Angular необхідно витратити значну кількість часу на вивчення його концепцій, структури та інструментів.
- Надмірність — деякі компоненти фреймворку перенасичені декоративним кодом, що може ускладнювати розробку та збільшувати обсяг роботи.

Для реалізації серверної частини було обрано ASP.NET Core базуючись на .NET 8, що забезпечує надійну та сучасну основу для веб-сайту.

ASP.NET Core — це мультиплатформне, високопродуктивне середовище з можливістю брати участь в його розробці, що дає змогу створювати сучасні додатки з доступом до Інтернету та підтримкою хмарних технологій таких як Azure, AWS, тощо. [22].

Основні можливості ASP.NET Core:

1. Створення веб-додатків, служб, застосунків Інтернету речей та мобільних серверних додатків.
2. Використання улюблених інструментів для розробки на Windows, Linux і macOS.
3. Розгортання додатків як локально, так і в хмарі.
4. Успадкування найкращих властивостей від .NET Core.
5. Переваги ASP.NET Core, якщо взяти до порівняння й інші існуючі альтернативи: легкість у тестуванні, що спрощує перевірку функціональності додатків, попередньо інтегровані сучасні фреймворки, які прискорюють розробку.
6. Підтримка хмарних систем конфігурацій, що допомагає забезпечити гнучкість у розгортанні й налаштуванні.
7. Стабільна та інтегрована система dependency injection, яка спрощує управління залежностями.
8. Високопродуктивний, легкий і компонентний постачальник запитів HTTP, що забезпечує швидкість і ефективність обробки запитів.
9. Можливість розміщення додатків на різних платформах, таких як IIS, Nginx, Docker тощо.

Та й загалом ASP.NET Core – потужний інструментом для створення серверної частини веб-додатків, забезпечуючи високу продуктивність, гнучкість і підтримку сучасних технологій.

MSSQL Server — це засіб для роботи з реляційними базами даних, розроблений

корпорацією Microsoft [23].

Переваги MSSQL Server:

1. Просте встановлення — як і інші продукти Microsoft, MSSQL Server має зручний та інтуїтивно зрозумілий інтерфейс для встановлення, що спрощує процес налаштування.
2. Покращена продуктивність — сервер забезпечує ефективне стиснення та шифрування даних, що покращує функції зберігання та пошуку інформації.
3. Безпека — MSSQL Server вважається одним із найбезпечніших серверів баз даних завдяки складним алгоритмам шифрування, які роблять його практично невразливим для атак, оскільки зловмисникам важко пройти всі рівні безпеки.
4. Декілька варіацій — MSSQL Server пропонує різні версії для різних потреб: Enterprise — для масштабних організацій із високими вимогами до продуктивності. Standard — для малих і середніх компаній. Workgroup — для малих підприємств. Express — безкоштовна версія, ідеальна для навчальних цілей. Developer — містить усі функції для активної розробки та тестування.
5. Відмінний механізм відновлення даних — сервер пропонує широкий набір функцій для збереження та відновлення даних, що забезпечує надійність у разі збоїв.

C# — це об'єктно-орієнтована мова програмування, яка надає змогу розробникам розробляти різноманітні безпечні та надійні програми, що працюють у середовищі.NET [24].

Переваги використання C#:

1. Швидкий час розробки — завдяки статичній типізації, зрозумілому синтаксису, великій кількості бібліотек і функціональним можливостям високого рівня, C# дозволяє швидко створювати програми.
2. Висока масштабованість — природа C# забезпечує створення надійних продуктів, які легко налаштовуються та адаптуються до змін.
3. Об'єктно-орієнтованість — ця властивість робить C# високоефективною

та надзвичайно гнучкою мовою програмування, що дозволяє створювати складні та структуровані програми.

4. Гнучка система навчання — C# є мовою високого рівня, яку легко вивчати та розуміти. Вбудовані функції спрощують її використання навіть для початківців.

5. Велика спільнота — C# популярна мова програмування, що регулярно входить у різні топи за використанням у світі, що забезпечує доступ до значної кількості ресурсів, документації та підтримки від досвідчених розробників.

Visual Studio 2022 було обрано як середовище розробки для реалізації проекту, оскільки воно надає найновітніші можливості для розробки програм на багатьох мовах, зокрема C#. Головною особливістю Visual Studio 2022 є перехід на 64-розрядну систему, що робить її першою і єдиною IDE такого типу [25].

Переваги Visual Studio 2022:

1. Продуктивність — перехід на 64-розрядну архітектуру значно збільшує швидкодію середовища, дозволяючи працювати з великими проектами без втрати ефективності.

2. Сучасність — це найновіше доступне середовище розробки, яке підтримує створення сучасних мобільних і настільних додатків, адаптивних веб-інтерфейсів.

3. Інноваційність — серед нововведень:

4. Доповнення коду за допомогою штучного інтелекту.

5. Спільні сеанси написання коду в реальному часі.

6. Зручна робота з системами контролю версій, хмарними середовищами та CI/CD.

7. IntelliCode — потужний алгоритм автоматичного довершення коду, який враховує контекст, включаючи назви змінних, функцій тощо, що значно прискорює процес розробки.

8. Безкоштовна хмарна інтеграція — забезпечує відмінну інтеграцію з хмарною платформою Azure, що спрощує розгортання додатків у хмарі.

9. Вбудована підтримка контролю версій Git [26] — дозволяє легко керувати репозиторіями та інтегруватися з популярними системами контролю версій.

Недоліки Visual Studio 2022:

- Розміри — середовище займає значний обсяг пам'яті на пристрої.
- Значна навантаження на пристрій — потребує потужного обладнання для ефективної роботи, що може бути проблемою для слабких систем.

Тому цілком закономірно, що VS 2022 є найкращим вибором для розробників, які працюють із.NET і C#, забезпечуючи сучасні інструменти, високу продуктивність і інноваційні функції, хоча її використання може вимагати потужних ресурсів пристрою.

### **Висновки до третього розділу**

Цей розділ містить, відомості про те, як проведено аналіз і здійснено вибір методів і засобів для створення веб-сайту. Також у ньому розглянуто основні компоненти та етапи розробки застосунку, зокрема можливі альтернативи для ключових частин веб-додатку: front-end, back-end і бази даних. Було детально описано переваги й недоліки кожного з елементів, а також розглянуто сторонні рішення, які зазвичай використовуються у розробці веб-сайтів.

У результаті аналізу програмних засобів було сформовано такий вибір:

1. Angular — для реалізації зовнішньої (лицевої) частини сайту (front-end).
2. ASP.NET Core — для розробки серверної частини застосунку (back-end).
3. MSSQL Server — для зберігання даних додатку.

Цей набір інструментів забезпечує сучасний, продуктивний і надійний підхід до створення веб-додатку, що відповідає вимогам проекту.

## РОЗДІЛ 4

### Практична реалізація

#### 4.1. Описання реалізації завдання

##### 4.1.1. Загальні відомості про програму

Інтелектуальна платформа «TestsByBook» створена для того, щоб зробити процес вивчення іноземних мов більш зручним і ефективним. Вона працює шляхом аналізу текстів книг і поділу їх на корисні фрагменти, які можна використовувати для подальшого читання та навчання. Це не лише допомагає отримати ключову інформацію про обрану книгу, але й сприяє вдосконаленню знань англійської мови.

Розробка веб-сайту здійснена із застосуванням таких технологій:

- Angular, HTML і CSS — для реалізації графічного інтерфейсу (front-end), що забезпечує інтуїтивно зрозумілу та зручну взаємодію користувачів із системою.
- ASP.NET Core і C# — для серверної частини (back-end), яка відповідає за обробку запитів, виконання операцій із даними та забезпечення стабільної роботи системи.
- AWS RDS MSSQL Server — як система управління базами даних (СУБД), що гарантує надійне зберігання інформації про користувачів, книги та результати тестувань.

Платформа «TestsByBook» об'єднує сучасні технологічні рішення, створюючи потужний інструмент для вивчення англійської мови через інтерактивну взаємодію з текстами книг.

##### 4.1.2. Структура бази даних

Щоб реалізувати веб-сайту інтелектуальної системи, яка використовує статистичний аналіз для вивчення іноземних мов, застосовується СУБД Microsoft SQL Server. База даних має назву, що відповідає назві сайту, — «TestsByBook».

Структура бази даних включає такі таблиці:

1. Authors — зберігає інформацію про авторів книг.

2. Books — містить дані про книги, доступні в системі.
3. Sentences — зберігає речення з книг, які використовуються для аналізу.
4. Words — містить слова, виділені з текстів для статистичного аналізу.
5. Tests — зберігає інформацію про тести, створені на основі книг.
6. Answers — містить відповіді на запитання тестів.
7. AspNetUsers — таблиця, яка зберігає дані про користувачів системи, створена за допомогою вбудованої системи автентифікації ASP.NET Core.

Структуру бази даних візуалізовано на рисунку. 4.1, де відображено зв'язки між таблицями та їхні основні атрибути.

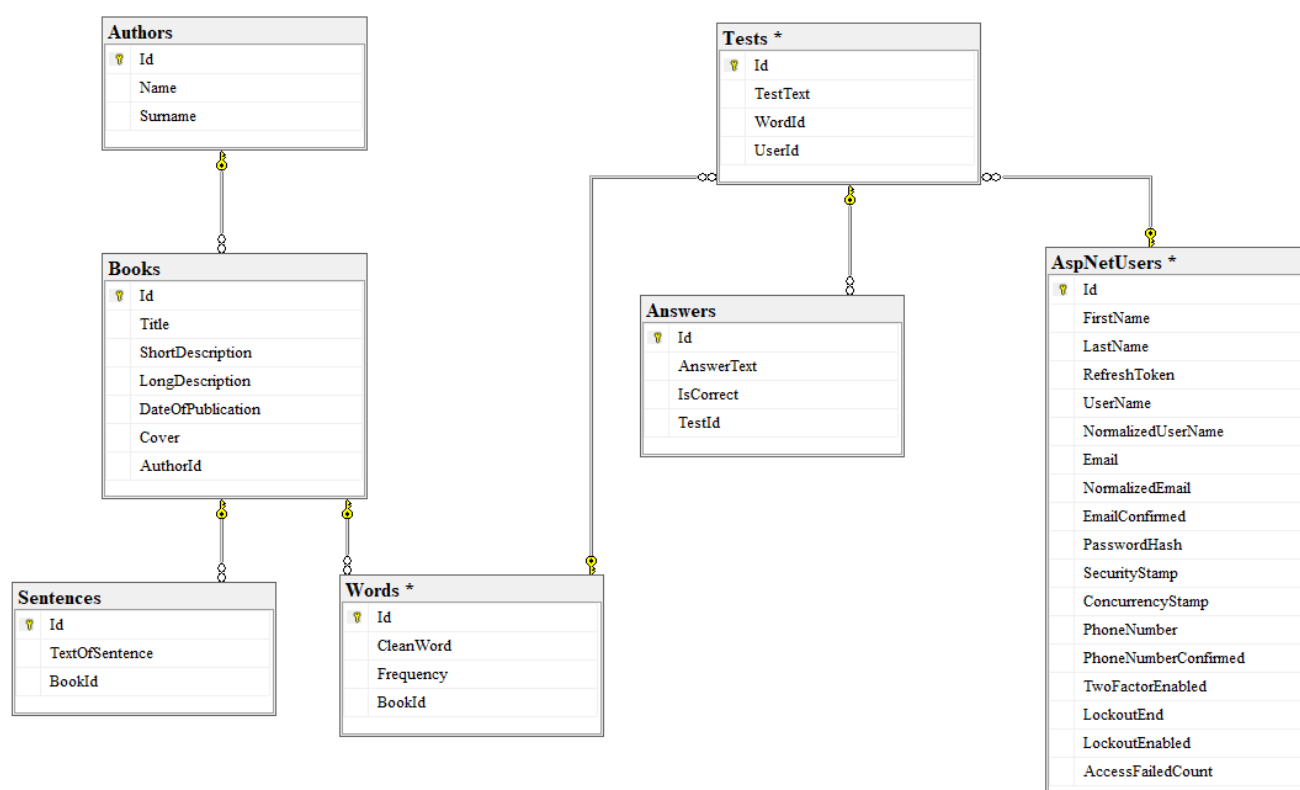


Рис. 4.1 Структура бази даних «TestsByBook»

Згідно зі структурою бази даних «TestsByBook», призначення та атрибути кожної таблиці розглянуто окремо:

1. Authors — таблиця, що зберігає дані про авторів книг. Вона містить такі атрибути: ідентифікатор (ID), ім'я автора, прізвище автора. (Рис. 4.2).
2. Books — таблиця, що зберігає відомості про книги. Її атрибути:

ідентифікатор (ID), заголовок книги, короткий опис, довгий опис, дата публікації, обкладинка книги, ідентифікатор автора (зв'язок із таблицею Authors). (Рис. 4.3).

3. Sentences — таблиця, що зберігає спеціально відібрані речення з книг. Вона включає такі атрибути: ідентифікатор (ID), текст речення, ідентифікатор книги (зв'язок із таблицею Books). (Рис. 4.4).

4. Words — таблиця, що зберігає спеціально відібрані слова з книг. Її атрибути: ідентифікатор (ID), чисте слово (без зайвих символів), статистика появи слова в книзі, ідентифікатор книги (зв'язок із таблицею Books). (Рис. 4.5).


5. Tests — таблиця, що зберігає всі згенеровані системою тести. Вона містить такі атрибути: ідентифікатор (ID), текст тесту, ідентифікатор слова (зв'язок із таблицею Words), ідентифікатор користувача (зв'язок із таблицеюAspNetUsers). (Рис. 4.6).

6. Answers — таблиця, що зберігає всі можливі відповіді до тестів. Її атрибути: ідентифікатор (ID), текст відповіді, маркер правильності відповіді (булевий тип), ідентифікатор тесту (зв'язок із таблицею Tests). (Рис. 4.7).


7. AspNetUsers — таблиця, що зберігає дані про користувачів системи. Вона включає такі атрибути: ідентифікатор (ID), ім'я користувача, прізвище користувача, маркер оновлення, ім'я користувача у системі, нормалізована версія імені користувача, електронна адреса, нормалізована версія електронної адреси, хеш паролю, охоронний штамп, паралельний штамп, номер телефону, маркер підтвердження номера телефону, маркер необхідності двоступінчастої аутентифікації, час блокування користувача, маркер доступності блокування, кількість невдалих спроб авторизації. (Рис. 4.8).

	Column Name	Data Type	Allow Nulls
▶	Id	int	<input type="checkbox"/>
	Name	varchar(40)	<input type="checkbox"/>
	Surname	varchar(40)	<input type="checkbox"/>


Рис. 4.2 Таблиця Authors

	Column Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	Title	varchar(40)	<input type="checkbox"/>
	ShortDescription	varchar(255)	<input type="checkbox"/>
	LongDescription	text	<input type="checkbox"/>
	DateOfPublication	date	<input type="checkbox"/>
	Cover	text	<input checked="" type="checkbox"/>
	AuthorId	int	<input type="checkbox"/>


*Рис. 4.3 Таблица Books*

	Column Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	TextOfSentence	text	<input type="checkbox"/>
	BookId	int	<input checked="" type="checkbox"/>

*Рис. 4.4 Таблица Sentences*

	Column Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	CleanWord	varchar(50)	<input type="checkbox"/>
	Frequency	real	<input type="checkbox"/>
	BookId	int	<input type="checkbox"/>

*Рис. 4.5 Таблица Words*

	Column Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	TestText	text	<input type="checkbox"/>
	WordId	int	<input type="checkbox"/>
	UserId	nvarchar(450)	<input type="checkbox"/>

*Рис. 4.6 Таблица Tests*

	Column Name	Data Type	Allow Nulls
▶	Id	int	<input type="checkbox"/>
	AnswerText	varchar(255)	<input checked="" type="checkbox"/>
	IsCorrect	bit	<input type="checkbox"/>
	TestId	int	<input type="checkbox"/>

*Рис. 4.7 Таблиця Answers*

	Column Name	Data Type	Allow Nulls
▶	Id	nvarchar(450)	<input type="checkbox"/>
	FirstName	varchar(50)	<input type="checkbox"/>
	LastName	varchar(50)	<input type="checkbox"/>
	RefreshToken	varchar(50)	<input checked="" type="checkbox"/>
	UserName	nvarchar(256)	<input checked="" type="checkbox"/>
	NormalizedUserName	nvarchar(256)	<input checked="" type="checkbox"/>
	Email	nvarchar(256)	<input checked="" type="checkbox"/>
	NormalizedEmail	nvarchar(256)	<input checked="" type="checkbox"/>
	EmailConfirmed	bit	<input type="checkbox"/>
	PasswordHash	nvarchar(MAX)	<input checked="" type="checkbox"/>
	SecurityStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>
	ConcurrencyStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>
	PhoneNumber	nvarchar(MAX)	<input checked="" type="checkbox"/>
	PhoneNumberConfirmed	bit	<input type="checkbox"/>
	TwoFactorEnabled	bit	<input type="checkbox"/>
	LockoutEnd	datetimeoffset(7)	<input checked="" type="checkbox"/>
	LockoutEnabled	bit	<input type="checkbox"/>
	AccessFailedCount	int	<input type="checkbox"/>

*Рис. 4.8 Таблиця AspNetUsers*

Побудовані відношення між таблицями бази даних забезпечують ефективно структурування даних. Вони представлені різними типами зв'язків між первинними та зовнішніми ключами таблиць. Структура бази даних (Рис. 4.1) містить 6 зв'язків:

1. Authors – Books — забезпечує зв'язок "один до багатьох" між таблицями авторів і книг. Один автор може мати багато книг у системі. Атрибут, що визначає зв'язок: AuthorId.

2. Books – Sentences — забезпечує зв'язок "один до багатьох" між таблицями книг і речень. Одна книга може містити багато речень. Атрибут, що визначає зв'язок: BookId.

3. Books – Words — забезпечує зв'язок "один до багатьох" між таблицями книг і слів. Одна книга може містити багато слів. Атрибут, що визначає зв'язок: BookId.

4. Words – Tests — забезпечує зв'язок "один до багатьох" між таблицями слів і тестів. Для одного слова може бути створено багато тестів. Атрибут, що визначає зв'язок: WordId.

5. Tests – Answers — забезпечує зв'язок "один до багатьох" між таблицями тестів і відповідей. Для одного тесту існує декілька відповідей. Атрибут, що визначає зв'язок: TestId.

6. AspNetUsers – Tests — забезпечує зв'язок "один до багатьох" між таблицями користувачів і тестів. Для одного користувача може зберігатися багато тестів. Атрибут, що визначає зв'язок: UserId.

Ці зв'язки забезпечують логічну організацію даних у системі «TestsByBook», дозволяючи ефективно управляти інформацією про авторів, книги, речення, слова, тести, відповіді та користувачів.

#### *4.1.3. Функції програми*

Веб-сайт «TestsByBook» надає користувачам можливість ознайомитися з основними причинами, чому варто вивчати нову мову.

На початковому етапі користувач може перейти на сторінку з книгами, де доступна інформація про обкладинку книги, автора, рік випуску та короткі відомості про саму книгу. Для цієї функціональності реєстрація в системі не є обов'язковою.

Сайт підтримує два типи користувачів:

- Звичайний користувач — має можливість зареєструватися у системі, після чого в нього з'являється доступ до основного функціоналу — тестувань.
- Адміністратор — має розширені права для управління системою.

Зареєстрований користувач може здійснити авторизацію, щоб отримати доступ до функціоналу сайту. Після входу до системи користувач отримує доступ до тестувань, де він може:

- Вибрати одну з раніше переглянутих книг. Коротка інформація про книгу буде відображена нижче.
- Вказати необхідну кількість тестів, які будуть згенеровані випадковим чином.
- Пройти тестування, після чого відправити свої відповіді на перевірку.

Результати тестування надаються у вигляді сповіщення, що дозволяє користувачеві оцінити свої знання та прогрес у вивченні мови.

«TestsByBook» забезпечує зручний і інтерактивний спосіб покращення знань англійської мови через роботу з книгами та тестами, адаптуючи функціонал до потреб різних типів користувачів.

#### **4.2. Аналіз отриманих результатів**

Даний підрозділ демонструє контрольний приклад веб-сайту, створеного для інтелектуальної системи освоєння іноземної мови з використанням статистичного аналізу, яка носить назву «TestsByBook».

На головній сторінці веб-сайту користувачам пропонується список причин, які мотивують до вивчення нових мов (Рис. 4.9). Як приклад обрано англійську мову, оскільки вона є однією з найпоширеніших і затребуваних мов у світі.

Крім того, на сторінці розташована панель навігації, яка забезпечує доступ до ключових функцій системи та визначає основні можливості взаємодії користувача із сайтом. Завдяки панелі навігації користувачі можуть легко перемикатися між різними розділами, такими як каталог книг, тестування, реєстрація та авторизація.

Цей приклад ілюструє зручність і функціональність інтерфейсу, що робить систему інтуїтивно зрозумілою та ефективною для використання. Такий підхід сприяє комфортному освоєнню іноземних мов і забезпечує позитивний досвід роботи з платформою.

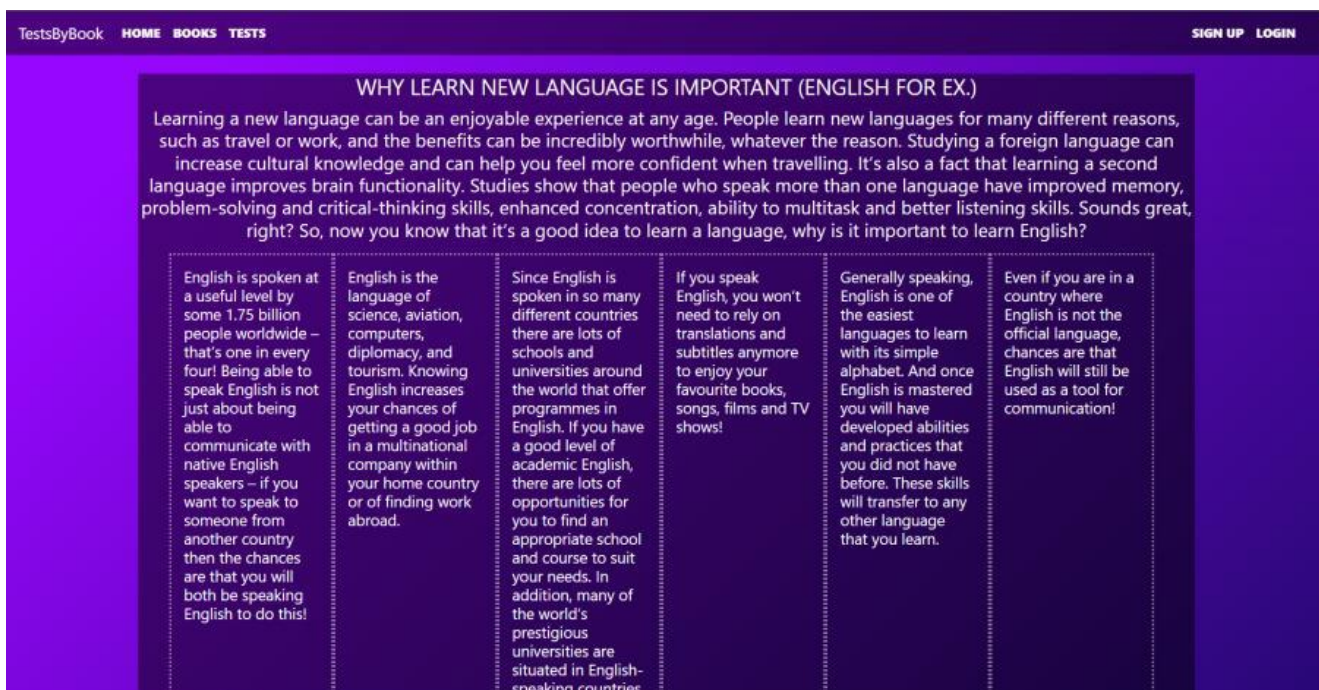


Рис. 4.9 Домашня сторінка веб-сайту

Панель навігації веб-сайту «TestsByBook» є важливим компонентом інтерфейсу, який забезпечує швидкий і зручний доступ до основних функцій системи. Розглянемо детальніше кожен її елемент:

- Назва веб-сайту — це перший елемент панелі, який відображає назву «TestsByBook». Натискання на цей елемент дозволяє користувачеві повернутися на головну сторінку сайту. При цьому відбувається оновлення даних системи, що забезпечує актуальність відображеної інформації.
- Розділ «BOOKS» — другий елемент панелі, який надає доступ до сторінки з усіма доступними книгами. У цьому розділі користувачі можуть переглядати список книг, ознайомлюватися з інформацією про автора, рік видання, короткий опис книги, а також бачити її обкладинку (Рис. 4.10).

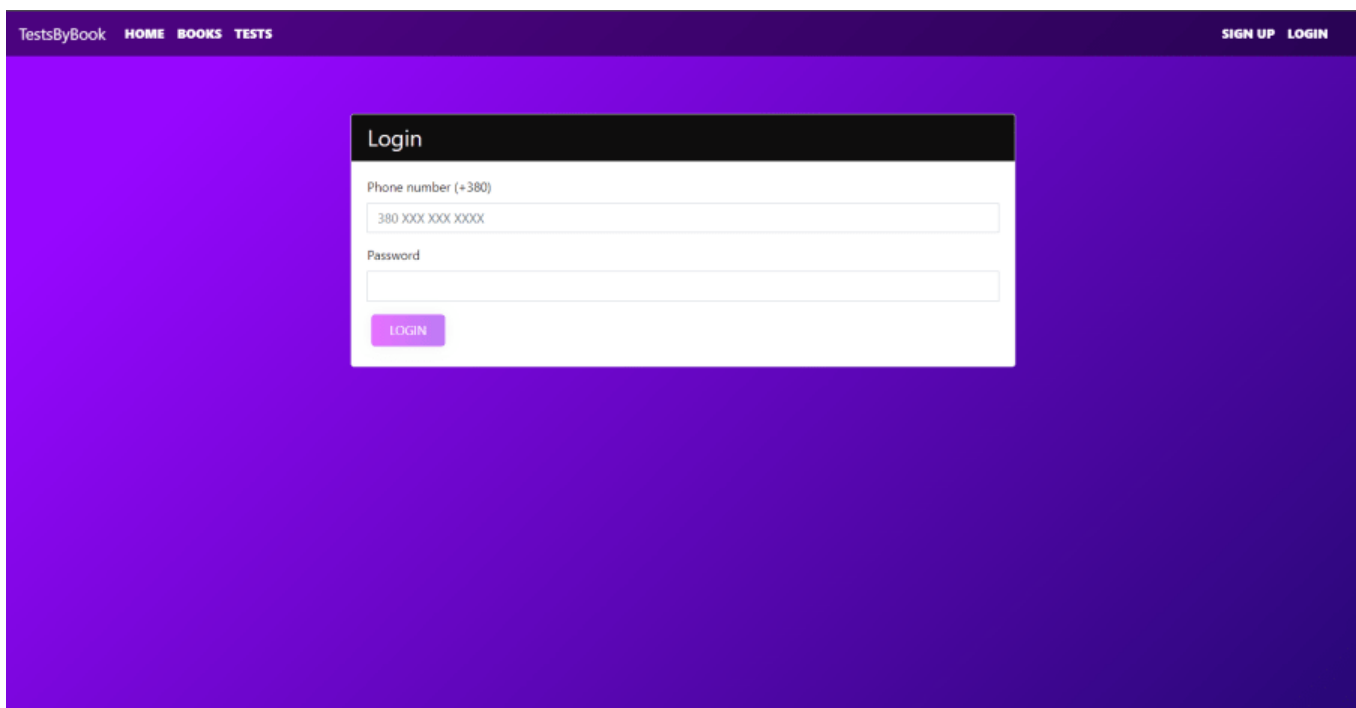
Ці елементи панелі навігації створюють інтуїтивно зрозумілий інтерфейс, який дозволяє користувачам легко переміщатися між розділами веб-сайту. Завдяки цьому забезпечується комфортна взаємодія з платформою та швидкий доступ до необхідної інформації, що сприяє ефективному використанню системи для вивчення іноземних мов.



Рис. 4.10 Сторінка «BOOKS» веб-сайту

На сторінці «BOOKS» користувач може переглядати доступні книги, гортати список, переглядати обкладинки, читати короткий опис кожної книги, дізнаватися ім'я автора та рік першої публікації.

Після ознайомлення з книгами користувач має можливість перейти до сторінки тестів. Якщо вхід у систему ще не здійснено, користувача зустріне форма входу (Рис. 4.11), яка дозволяє авторизуватися для отримання доступу до функціоналу тестування.

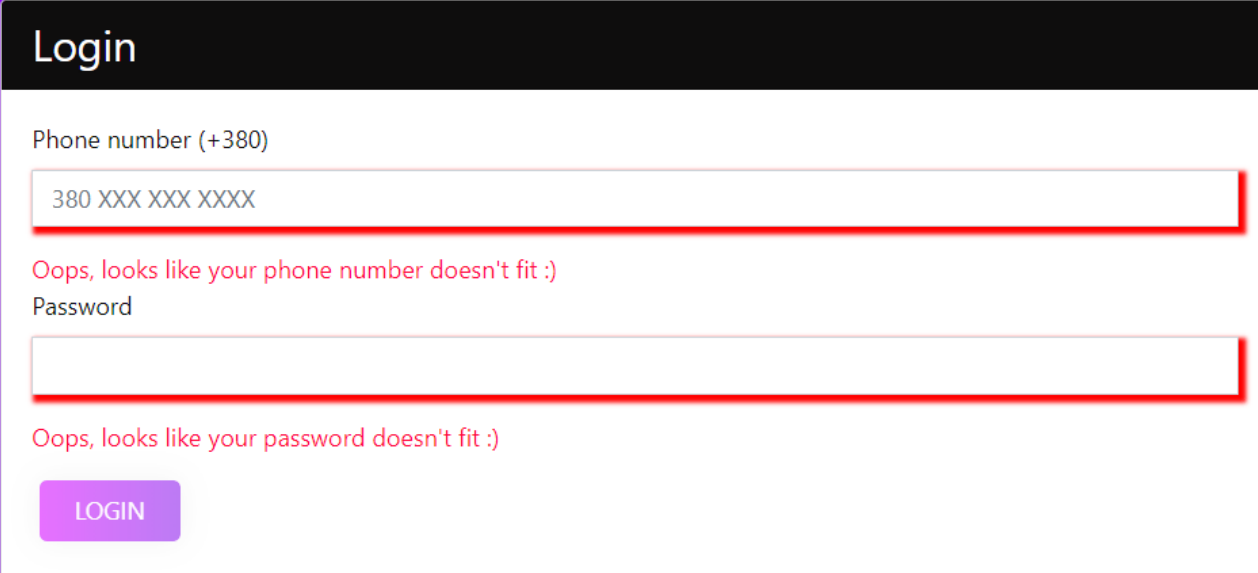


*Рис. 4.11 Сторінка, що забезпечує вхід до системи*

Користувач також має можливість перейти до форми входу, вибравши пункт «LOGIN» на панелі навігації.

На сторінці входу представлена інтерактивна форма, яка запрошує користувача ввести номер телефону, зазначений під час реєстрації, та свій пароль. Якщо введені дані не відповідають встановленому формату або містять помилки, система автоматично відобразить повідомлення про помилку під відповідним полем, яке було заповнене некоректно (Рис. 4.12).

Ця функція забезпечує комфортний і безпечний процес авторизації, надаючи користувачам можливість оперативно виправляти помилки та отримувати доступ до системи без зайвих труднощів. Такий підхід сприяє підвищенню зручності використання веб-сайту, а також гарантує захист даних користувачів.

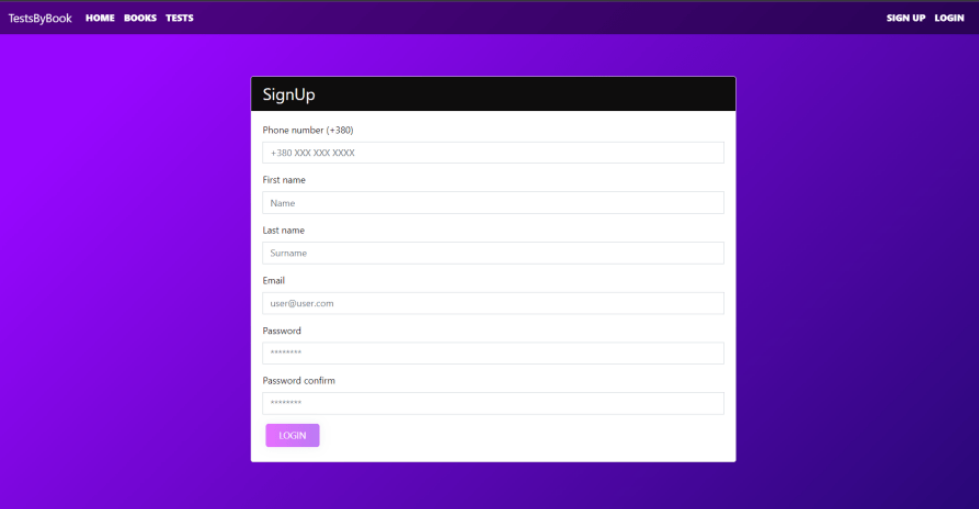


The image shows a login form titled "Login" on a black background. Below the title, there are two input fields. The first is labeled "Phone number (+380)" and contains the text "380 XXX XXX XXXX". Below this field is a red error message: "Oops, looks like your phone number doesn't fit :)". The second input field is labeled "Password" and is empty. Below it is another red error message: "Oops, looks like your password doesn't fit :)". At the bottom left of the form is a purple button labeled "LOGIN".

*Рис. 4.12 Неправильно введені дані*

Якщо користувач ще не зареєстрований у системі, він може скористатися опцією «SIGN UP», доступною на панелі навігації.

При виборі цього пункту відкривається сторінка з формою для реєстрації (Рис. 4.14). У цій формі користувач має можливість ввести дані, що потрібні для реєстрації нового облікового запису, такі як ім'я, номер телефону, електронна пошта, прізвище, та пароль.



The image shows a user registration form titled "SignUp" on a white background. The form is set against a dark blue background. At the top left of the page is the text "TestsByBook HOME BOOKS TESTS" and at the top right is "SIGN UP LOGIN". The registration form includes the following fields: "Phone number (+380)" with the placeholder "+380 XXX XXX XXXX", "First name" with the placeholder "Name", "Last name" with the placeholder "Surname", "Email" with the placeholder "user@user.com", "Password" with a masked input (dots), and "Password confirm" with a masked input (dots). A purple "LOGIN" button is located at the bottom of the form.

*Рис. 4.14 Сторінка реєстрації користувача*

За результатами успішної реєстрації або входу до системи користувач

автоматично повертається на головну сторінку, а панель навігації змінює свій вигляд (Рис. 4.17). У ній додається опція щоб вийти з акаунту, яка дозволяє завершити сеанс у будь-який момент. Крім того, система виконує вихід із облікового запису після 20 хвилин бездіяльності, забезпечуючи додатковий рівень безпеки.

Після авторизації користувач отримує доступ до сторінки тестувань. Для переходу до неї необхідно вибрати опцію «TESTS» у панелі навігації (Рис. 4.18). На сторінці тестувань користувач може налаштувати параметри тесту, обрати книгу, з якою він бажає працювати, і згенерувати тестове завдання.

Ця функціональність надає інтерактивний і персоналізований підхід до вивчення іноземної мови, зокрема англійської. Вона дозволяє користувачеві адаптувати процес тестування відповідно до власних потреб і цілей, роблячи навчання більш ефективним і зручним.

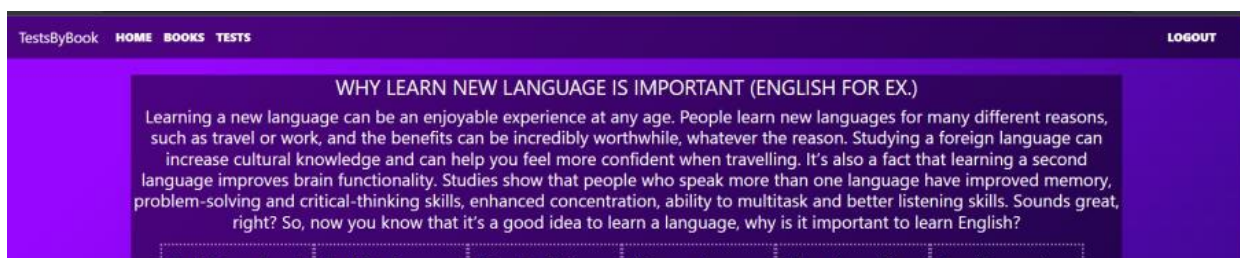


Рис. 4.17 Вигляд системи після вдалого логіну

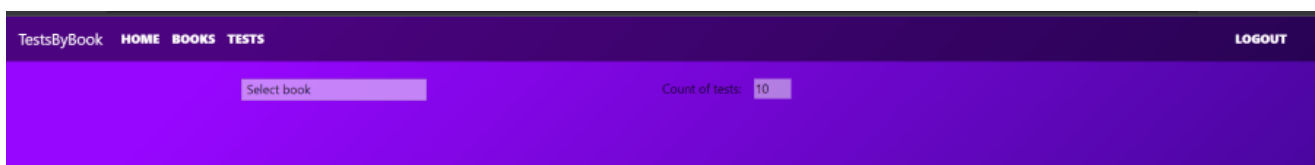


Рис. 4.18 Сторінка тестування

Меню вибору книг на сторінці тестувань містить випадаючий список, у якому відображаються всі доступні на даний момент книги.

Після вибору однієї з опцій користувач отримує розширену інформацію про вибрану книгу, включаючи її опис, автора, рік публікації та обкладинку. Крім того, користувачеві надається можливість вказати бажану кількість тестів, які будуть згенеровані системою на основі вибраної книги (Рис. 4.19).



Рис. 4.19 Інформація про книгу і можливість створення тестів

На наступному етапі, після вибору книги, користувач обирає бажану кількість тестів через випадające меню та натискає кнопку «Get tests». Після цього система автоматично генерує відповідне тестування (Рис. 4.20).

Згенеровані тести створюються на основі вибраної книги та включають завдання, спрямовані на покращення знань іноземної мови. Кожен тест складається з питань, сформованих за допомогою статистичного аналізу тексту книги. Це дозволяє користувачеві ефективно засвоювати нову лексику, вдосконалювати граматичні навички та поглиблювати знання мови в інтерактивному форматі.

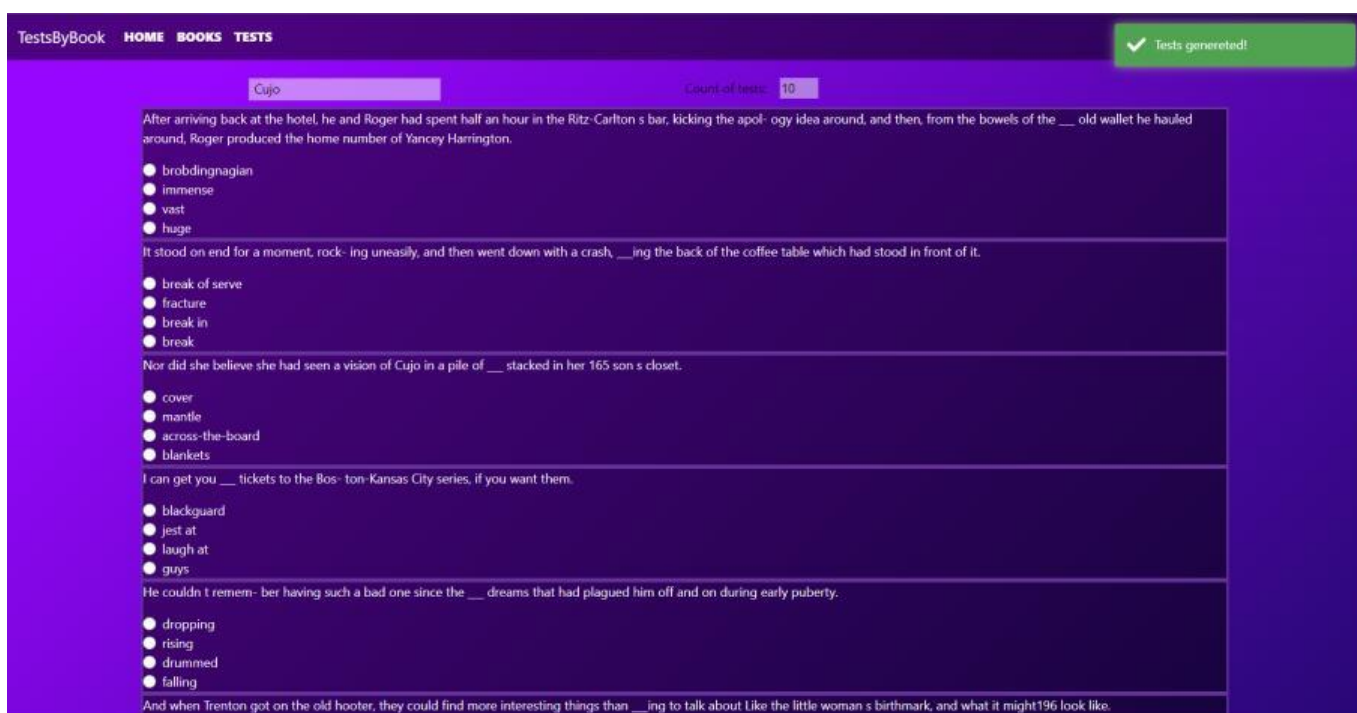


Рис. 4.20 Згенеровані тести

На цьому етапі користувач сайту переходить безпосередньо до процесу

навчання, виконуючи згенеровані тести. Для кожного запитання він має можливість обрати відповіді, які вважає правильними.

Коли всі запитання тесту отримують відповіді, користувач може відправити їх на перевірку, натиснувши кнопку «Check».

Після завершення перевірки система генерує повідомлення з результатами тестування (Рис. 4.21). У цьому повідомленні відображається кількість правильних і неправильних відповідей, що дає змогу користувачеві оцінити свій рівень знань і прогрес у вивченні іноземної мови. Такий підхід дозволяє не лише перевіряти результати, а й визначати аспекти, які потребують додаткової уваги та вдосконалення.

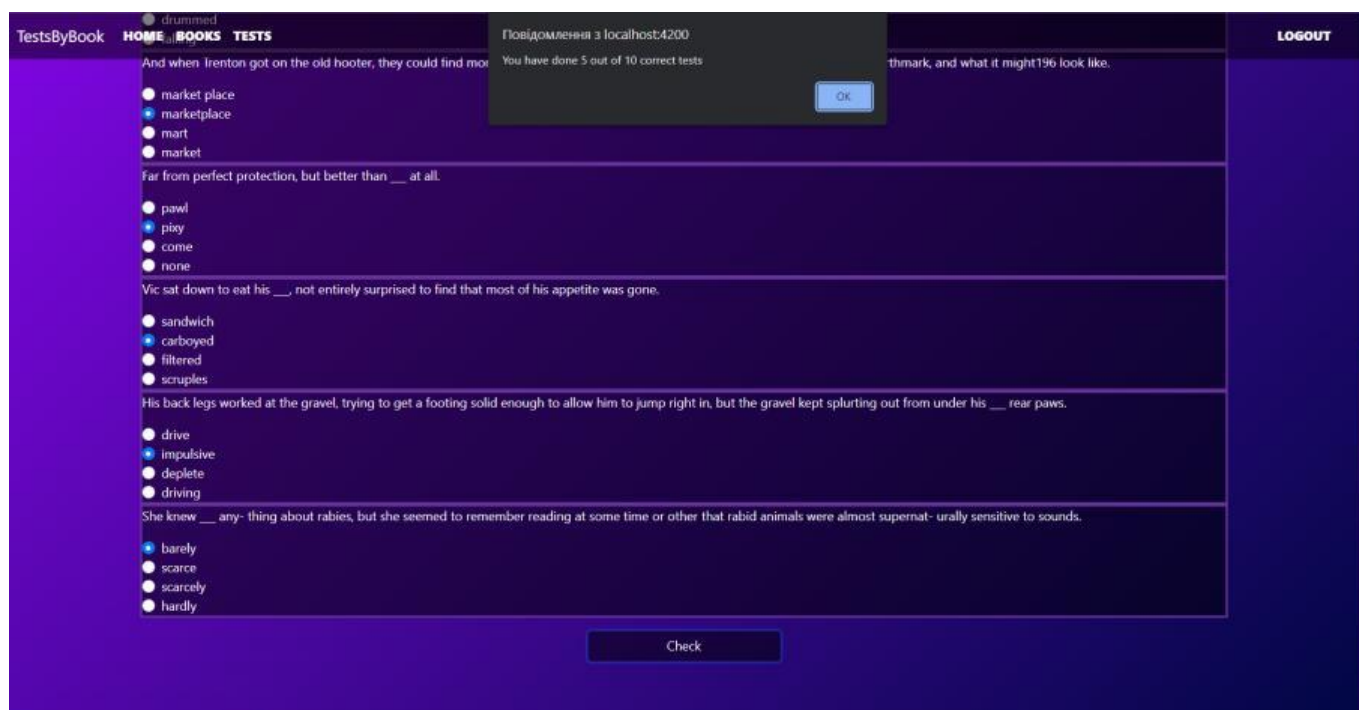


Рис. 4.21 Пройдені й перевірені тести

Для користувачів із роллю адміністратор доступна додаткова функція — можливість додавати книги до системи (Рис. 4.22).

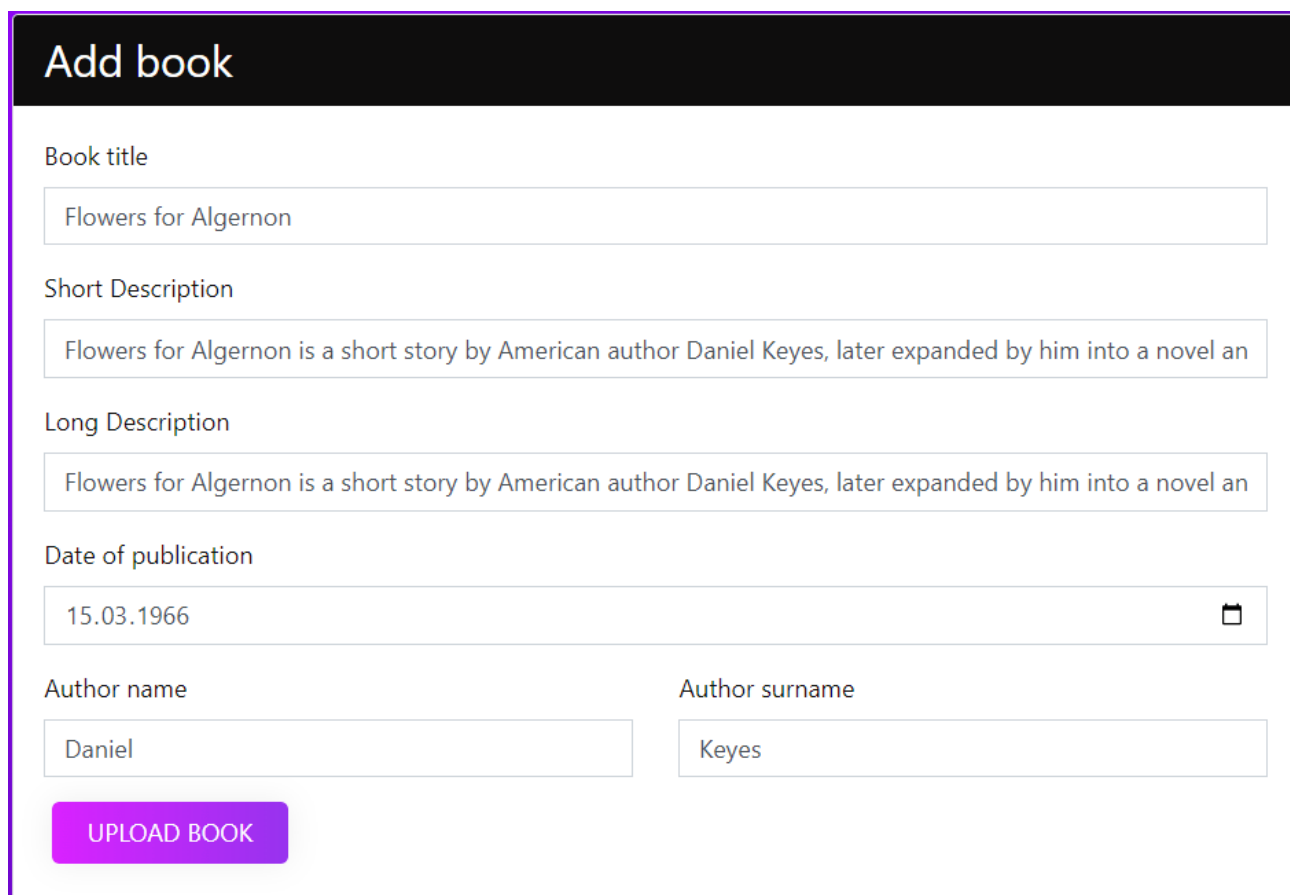
На відповідній сторінці розміщена форма, яка дозволяє адміністратору ввести необхідні дані про книгу, зокрема:

1. Назву книги.
2. Короткий опис.

3. Довгий опис.
4. Дату публікації.
5. Прізвище та ім'я власника твору.

Після того, як адміністратор заповнить форму, він має можливість натиснути кнопку «UPLOAD BOOK», щоб додати книгу до системи.

Ця функція дозволяє адміністраторам розширювати бібліотеку доступних навчальних матеріалів, забезпечуючи користувачів новими ресурсами для вивчення мови та проходження тестування. Завдяки цьому система постійно оновлюється, підтримуючи актуальність контенту та сприяючи більш ефективному процесу навчання.



**Add book**

Book title  
Flowers for Algernon

Short Description  
Flowers for Algernon is a short story by American author Daniel Keyes, later expanded by him into a novel an

Long Description  
Flowers for Algernon is a short story by American author Daniel Keyes, later expanded by him into a novel an

Date of publication  
15.03.1966

Author name  
Daniel

Author surname  
Keyes

**UPLOAD BOOK**

*Рис. 4.22 Сторінка з можливістю додавання книг*

Після того як адміністратор додає або оновлює дані, введені через форму, відповідна інформація автоматично додається змінюється у віддаленому сховищі

даних. Ці зміни одразу відображаються на веб-сайті, забезпечуючи актуальність даних для всіх користувачів (Рис. 4.23).

Ця функція гарантує, що додані або оновлені книги стають доступними для перегляду та використання в тестуваннях без затримок, забезпечуючи зручність і ефективність управління контентом системи.



*Рис. 4.23 Результат додавання нової книги*

### **Висновки до четвертого розділу**

Розділ описує детально як було розроблено веб-сайт «TestsByBook». Представлено структуру бази даних, створеної для роботи системи, з детальним роз'ясненням усіх таблиць, їхніх атрибутів і зв'язків між ними. Окрім цього, розроблено інструкцію для користувачів, яка пояснює, як взаємодіяти із системою. У ній описано основні можливості та призначення платформи, а також технічні вимоги для її використання, включаючи апаратне забезпечення, організацію роботи та необхідне програмне забезпечення.

Наприкінці розділу проведено детальний аналіз прикладу роботи веб-сайту. У ньому описано всі функціональні можливості, які система надає користувачам, зокрема перегляд доступних книг, реєстрацію та авторизацію, генерацію тестів,

проходження тестувань. Крім того, розглянуто додаткові функції для адміністраторів, такі як можливість додавання нових книг до бібліотеки.

Цей розділ демонструє, як розроблений веб-сайт виступає зручним і ефективним інструментом для вивчення іноземних мов. Використовуючи інтерактивний підхід і сучасні технології, система забезпечує комфортне та продуктивне навчання, адаптоване до потреб користувачів.

## РОЗДІЛ 5

### Економічне обґрунтування доцільності роботи

#### 5.1. Економічна характеристика проектного рішення

Сферою застосування системи є освітня галузь, корпоративне навчання, мовні школи, онлайн-платформи для самостійного навчання та підготовка до міжнародних сертифікаційних іспитів.

Саме тому метою роботи є створення інтелектуальної системи освоєння іноземної мови з використанням статистичного аналізу, яка дозволить автоматизувати процес навчання, персоналізувати підхід до користувачів та підвищити ефективність освоєння мов. Система буде реалізована у вигляді веб-застосунку, що забезпечить доступність для широкого кола користувачів через мережу Інтернет. Основний функціонал включає реєстрацію користувачів, вибір навчальних матеріалів, автоматичне створення тестів на основі книг, проходження тестування та аналіз результатів із наданням персоналізованих рекомендацій.

Потенційними споживачами запропонованої системи є:

- Освітні установи: школи, університети, мовні курси, які прагнуть автоматизувати процес навчання та оцінки знань студентів.
- Мовні школи: організації, які надають послуги з вивчення іноземних мов і потребують інструментів для створення тестів та аналізу результатів.
- Індивідуальні користувачі: люди, які самостійно вивчають іноземні мови та потребують інструментів для оцінки своїх знань і вдосконалення навичок.

Розроблення запропонованої системи сприятиме впровадженню новітніх технологій у процес вивчення іноземних мов. Система використовує статистичний аналіз для створення тестів і аналізу результатів, що дозволяє адаптувати навчальні матеріали до потреб кожного користувача. Оскільки на ринку існує обмежена кількість систем, які поєднують статистичний аналіз із навчанням, запропоноване рішення має значний потенціал для впровадження в освітній та корпоративний сектори.

Далі наведено те як було здійснено обчислення доцільності системи:

## Розрахунок витрати на оплату праці

Фах розробника	Кількість розробників, чол.	Час роботи, дні	Денна заробітна плата розробника, грн.	Витрати на оплату праці, грн.
ВА	1	70	3 616,30	253 141
Дата аналітик	1	45	3 847,10	173 120
Дизайнер	1	10	2 626,90	26 269
Frontend розробник	2	70	4 162,90	582 806
Backend програміст	2	70	4 162,90	582 806
Тестувальник	2	60	3 269,40	392 328
<b>Разом:</b>	9	325	21 685,50	2 010 470

Таблиця 5.2

## Розрахунок витрати на сплату єдиного соціального внеску

Спеціаліст	Сума основної заробітної плати, грн	Сума додаткової заробітної плати, грн	Разом витрати на оплату праці, грн	Сума ЄСВ, грн
Бізнес аналітик	253 141	-	253 141	55 691,02
Спеціаліст із аналізу даних	173 120	-	173 120	38 086,4
Дизайнер	26 269	-	26 269	5 779,18
Front-end розробник	582 806	-	582 806	128 217,32
Back-end програміст	582 806	-	582 806	128 217,32
Тестувальник	392 328	-	392 328	86 312,16
<b>Разом:</b>	2 010 470	-	2 010 470	442303,40

Таблиця 5.3

## Розрахунок витрат на куповані вироби

№ п/п	Найменування купованих виробів	Марка, тип	Кількість на розробку, шт.	Ціна за одиницю, грн.	Сума витрат, грн.	Сума витрат з урахуванням транспортно-заготівельних витрат, грн.
1	Jira	Atlassian	1	8 370	8 370	0
2	AWS	Amazon	2	17 520	35 040	0
3	Github	Microsoft	1	4 752	4 752	0
4	Figma	Figma	1	1999	1999	
<i>Всього</i>						50 161

### Кошторис витрат на розробку проектного рішення

Найменування елементів витрат	Сума витрат, грн.
Витрати на розробку проектного рішення, у т.ч.:	
витрати на оплату праці	2 010 470
сплата єдиного соціального внеску	442 303,40
витрати на додаткові вироби, що закупаються	50 161
накладні витрати	603 141
інші витрати	201 047
Витрати на відлагодження і дослідну експлуатацію системи	13 000
<i>Всього</i>	<i>3 320 122,40</i>

Таблиця 5.5

### Визначення комплексного показника якості проектованої системи (аналога)

Показники	Числове значення показників, бали		Відносний показник якості, $C_i$	Коефіцієнт вагомості $q_i$	$C_i * q_i$
	Аналог	Проект. прогр. продукт			
Функціональність	9	9	1,00	0,20868	0,20868
Надійність	7	8	1,14	0,19182	0,21922
Зручність	6	8	1,33	0,19701	0,26268
Продуктивність	6	8	1,33	0,20802	0,27736
Підтримуваність	4	8	2,00	0,19448	0,38896
Всього	32	41	6,81	1	1,35690

Таблиця 5.6

### Розрахунок витрат на підготовку даних для роботи на ЕОМ

Категорія персоналу	Чисельність співробітників і-ої категорії, чол.	Час роботи співробітників і-ої категорії, год.	Середньо-годинна ЗП співробітника і-ої категорії, грн.	Витрати на підготовку даних, грн.
Проектне рішення				
Бізнес аналітик	1	10	618,25	6182,5
Спеціаліст із аналізу даних	1	10	657,71	6577,1
Дизайнер	1	10	449,09	4490,9
Front-end розробник	2	10	711,7	14234
Back-end розробник	2	10	711,7	14234

Тестувальник	2	10	558,95	11179
<i>Всього</i>	9	60	3707,4	56897,5
Аналог				
Бізнес аналітик	1	15	618,25	9273,75
Дизайнер	1	15	449,09	6736,35
Front-end розробник	2	15	711,7	21351
Back-end розробник	2	15	711,7	21351
Тестувальник	2	15	558,95	16768,5
<i>Всього</i>	8	75	3049,69	75480,6

Таблиця 5.7

### Показники економічної ефективності проектного рішення

Найменування показників	Одиниці вим.	Значення показників	
		Аналог	Проектне рішення
1. Капітальні вкладення	грн.	-	3 307 122,40
2. Ціна придбання	грн.	4 850 000	4 067 760
3. Річні експлуатаційні витрати	грн.	229 321,80	172 852,5
4. Ціна споживання	грн.	5 348 608	4 443 588
5. Економічний ефект в сфері експлуатації	грн.	-	122 779,67
6. Додатковий економічний ефект в сфері експлуатації	грн.	-	438 618,09
7. Економічний ефект в сфері проектування	грн.	-	905 020
8. Додатковий економічний ефект в сфері проектування	грн.	-	1 486 976,82
9. Термін окупності витрат на проектування рішення	роки	-	2,2
10. Коефіцієнт конкурентоспроможності		-	1,130

### Висновки до п'ятого розділу

Даний розділ містить розрахунок витрат на розробку та впровадження інтелектуальної системи освоєння іноземної мови з використанням статистичного аналізу. Аналіз включав обчислення комплексного показника якості, витрат на експлуатацію, вартості споживання проектного рішення та оцінку економічної ефективності. Загальна сума витрат на створення проектного рішення склала 3 307 122,40 грн., при цьому показник якості у порівнянні з аналогічними системами

становить 1,36. Значення конкурентоспроможності системи дорівнює 1,130, що вказує на її перевагу над існуючими рішеннями.

Додатково було визначено економічний ефект від експлуатації системи, який становить 122 779,67 грн., що підтверджує її ефективність та доцільність впровадження. Таким чином, розробка та реалізація запропонованої системи для освоєння іноземної мови із застосуванням статистичного аналізу є виправданою, оскільки вона демонструє високий рівень якості, конкурентоспроможність та економічну вигоду.

## ВИСНОВКИ

У ході роботи над магістерською роботою було досліджено етапи створення системи освоєння іноземної мови з використанням статистичного аналізу. Виконано огляд наукових публікацій, виявлено та обговорено проблематику, а також розроблено стратегії її подолання. Проведено системний аналіз, підібрано методи та засоби впровадження та, власне, реалізовано систему.

Під час того як розроблявся перший розділ було здійснено глибокий розгляд наукових публікацій та онлайн-ресурсів. Цей аналіз допоміг визначити існуючі платформи для вивчення іноземних мов, а також виявити їх можливі переваги та недоліки сторони. Отримані дані підкреслюють необхідність створення системи освоєння іноземної мови, яка використовує статистичний аналіз для автоматизації навчання, персоналізації процесу та підвищення ефективності оцінки знань користувачів.

У другому розділі кваліфікаційної роботи було здійснено комплексний аналіз системи. Описано ключові проблеми, які стосуються процесу вивчення іноземних мов, та проведено аналіз ієрархій. Вибрано відповідний тип системи. За допомогою UML-діаграм надано всебічний опис системи. Окреслено вимоги, які система має задовольняти, та представлено альтернативні шляхи вирішення існуючих проблем, пов'язаних із проектом. Визначено основну мету створення системи, її застосування та надано обґрунтування необхідності її розробки, тестування та впровадження. Показано основні очікування від впровадження системи, сформульовано її концептуальну модель, основні завдання та архітектуру, а також детально описано вимоги до системи.

Третій розділ присвячений аналізу технічних засобів, які використовуються для розробки веб-сайтів. Окрему увагу приділено технологіям, що застосовуються для розробки клієнтської (front-end) та серверної (back-end) частин. Для створення запланованої системи обрано основні інструменти: .NET, Angular, MSSQL Server. Вибір середовища для розробки припав на Visual Studio, зважаючи на його інтеграцію з іншими інструментами, зручність роботи та підтримку сучасних технологій.

Використання Angular для front-end частини забезпечує створення динамічного та інтерактивного інтерфейсу користувача, тоді як .NET у поєднанні з MSSQL Server дозволяє реалізувати надійну серверну частину та ефективну роботу з базою даних.

Четвертий розділ містить огляд розробленого програмного забезпечення, яке створено відповідно до визначених стандартів. Тут представлено ключові деталі про програму, її цільове призначення, основні функції, організацію бази даних, логічну структуру, використані технічні ресурси, а також процедури ініціалізації та завантаження. Описано також вхідні та вихідні параметри даних. Додатково проведено аналіз прикладу, який демонструє коректну роботу програми та її відповідність заданим критеріям.

У п'ятому розділі проведено аналіз витрат, пов'язаних із створенням та реалізацією системи. Розглянуто розрахунки комплексного показника якості, витрат на експлуатацію, вартості споживання запропонованого рішення, а також оцінено економічну ефективність. Окрім того, визначено економічний результат у процесі використання системи. Таким чином, розробка та впровадження системи виявляються обґрунтованими та доцільними.

## СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Концептуальні засади державної політики щодо розвитку англійської мови у сфері вищої освіти [Електронний ресурс]. Режим доступу: <https://mon.gov.ua/ua/news/mon-stvorilo-koncepciyu-rozvitku-anglijskoyi-v-universitetah-u-dodatku-riven-v1-obovyazkova-umova-vstupu-v2-vipusku-vikladannya-profilnih-disciplin-inozemnoyu-ta-movni-skriningi>.
2. The Guardian: Duolingo creator: «I wanted to create a way to learn languages for free» [Електронний ресурс]. Режим доступу: <https://www.theguardian.com/education/2014/aug/27/luis-von-ahn-ceo-duolingo-interview>.
3. The American Political Science Review. 1906, травень – щоміс. – 2003. Т. 97, № 2. С. 311–331.
4. The Stata Journal. 2001, липень – щоміс. – 2018. Т. 3, № 2. С. 379–386.
5. Duolingo Revenue and Usage Statistics [Електронний ресурс]. Режим доступу: <https://www.businessofapps.com/data/duolingo-statistics/>.
6. Авраменко В.С., Авраменко А.С. Проектування інформаційних систем: навчальний посібник. Черкаси, 2017. УДК 004.4 (075.8). ББК 32.973.2-018я73-1 А21. Режим доступу: <https://files.znu.edu.ua/files/Bibliobooks/Inshi72/0053479.pdf>.
7. Кучеров Д.П., Артамонов Є.Б. Інженерія програмного забезпечення: навчальний посібник. Київ, 2017. УДК 004.415/.416(075). ББК 3973.20-018.2я7 К959. Режим доступу: <https://core.ac.uk/download/286637943.pdf>.
8. Introduction to the Angular Docs [Електронний ресурс]. Режим доступу: <https://angular.io/docs>.
9. Посібник: знайомство з React [Електронний ресурс]. Режим доступу: <https://uk.reactjs.org/tutorial/tutorial.html>.
10. What is Vue? [Електронний ресурс]. Режим доступу: <https://vuejs.org/guide/introduction.html>.
11. Meet Django [Електронний ресурс]. Режим доступу: <https://www.djangoproject.com/>.

12. What is .NET? [Электронный ресурс]. Режим доступа: <https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet>.
13. Meet Laravel [Электронный ресурс]. Режим доступа: <https://laravel.com/docs/9.x#meet-laravel>.
14. Amazon Relational Database Service Documentation [Электронный ресурс]. Режим доступа: [https://docs.aws.amazon.com/rds/?id=docs\\_gateway](https://docs.aws.amazon.com/rds/?id=docs_gateway).
15. What is Amazon Aurora? [Электронный ресурс]. Режим доступа: [https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/CHAP\\_AuroraOverview.html](https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/CHAP_AuroraOverview.html).
16. MariaDB Foundation [Электронный ресурс]. Режим доступа: <https://mariadb.org/>.
17. HTML: HyperText Markup Language [Электронный ресурс]. Режим доступа: <https://developer.mozilla.org/en-US/docs/Web/HTML>.
18. CSS: Cascading Style Sheets [Электронный ресурс]. Режим доступа: <https://developer.mozilla.org/en-US/docs/Web/CSS>.
19. Get started with Bootstrap [Электронный ресурс]. Режим доступа: <https://getbootstrap.com/docs/5.1/getting-started/introduction/>.
20. Bootstrap [Электронный ресурс]. Режим доступа: <https://developer.mozilla.org/en-US/docs/Glossary/Bootstrap>.
21. 7 Benefits of Angular You Should Know if You Want to Build a Digital Product [Электронный ресурс]. Режим доступа: <https://www.netguru.com/blog/benefits-of-angular>.
22. Overview to ASP.NET Core [Электронный ресурс]. Режим доступа: <https://docs.microsoft.com/en-us/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0>.
23. MS SQL SERVER HISTORY AND ADVANTAGES [Электронный ресурс]. Режим доступа: <https://bytescout.com/blog/2014/09/ms-sql-server-history-and-advantages.html>.
24. What is C#? [Электронный ресурс]. Режим доступа:

<https://dotnet.microsoft.com/en-us/learn/csharp>.

25. Meet the Visual Studio family [Электронный ресурс]. Режим доступа: <https://visualstudio.microsoft.com/>.

26. Microsoft Learn. Git with Visual Studio [Электронный ресурс]. Режим доступа: <https://learn.microsoft.com/en-us/visualstudio/version-control/git-with-visual-studio?view=vs-2022>.

## АНОТАЦІЯ

Мирош Ю. М., Микіч Х. І. (керівник). Інтелектуальна система освоєння іноземної мови з використанням статистичного аналізу. Магістерська кваліфікаційна робота. – Національний університет «Львівська політехніка», Львів, 2025.

Розширена анотація.

У сучасному світі знання іноземних мов є важливим фактором особистісного та професійного розвитку [1]. Однак традиційні методи навчання часто стикаються з низкою проблем, таких як відсутність персоналізації, складність оцінки прогресу та низька мотивація учнів. Ці виклики стають особливо актуальними в умовах глобалізації, коли знання мов відкриває доступ до нових можливостей для багатьох аспектів життя від навчання й роботи до спілкування.

Система освоєння іноземної мови з використанням статистичного аналізу спрямована на те, щоб вирішувати такі проблеми за допомогою автоматизації процесу навчання, персоналізації підходу до кожного користувача та підвищення ефективності оцінки знань. Основний функціонал системи включає реєстрацію користувачів, вибір навчальних матеріалів (книг), автоматичне створення тестів на основі книг, проходження тестування та аналіз результатів із наданням персоналізованих рекомендацій.

Система освоєння іноземної мови, яка базується на використанні статистичного аналізу, розроблена для вирішення низки важливих завдань, пов'язаних із навчальним процесом. Її мета полягає у впровадженні автоматизації навчання, що дозволяє значно спростити процес засвоєння матеріалу, а також забезпечити персональний підхід до будь-якого користувача, маючи на увазі його рівень знань, темп навчання та особисті потреби. Крім того, система спрямована на підвищення точності та ефективності оцінки знань, що дає змогу отримувати більш об'єктивні результати та рекомендації для подальшого вдосконалення. Основний функціонал системи охоплює кілька ключових етапів. Спочатку користувачі проходять реєстрацію, що дозволяє створити персоналізований профіль для збереження даних про їхній прогрес. Далі вони мають можливість обирати навчальні матеріали, зокрема книги, які відповідають їхнім

інтересам та рівню володіння мовою. На основі обраних матеріалів система автоматично генерує тести, які допомагають перевірити засвоєння інформації. Після проходження тестування результати аналізуються, і користувач отримує детальні персоналізовані рекомендації, спрямовані на покращення слабких сторін та закріплення сильних навичок. Таким чином, система забезпечує комплексний підхід до навчання, роблячи його більш ефективним, зручним та адаптованим до потреб кожного користувача.

Система реалізована у вигляді веб-застосунку, що забезпечить доступність для великої кількості користувачів через мережу Інтернет. Веб-сайт дозволяє легко працювати з будь-яким типом пристроїв (телефон, планшет, ноутбук, комп'ютер) і робить користувача незалежним від операційної системи, на якій запускається сайт. Використання сучасних технологій, таких як Angular для front-end, .NET для back-end та MS SQL Server для роботи з базою даних, забезпечить надійність, продуктивність та інтерактивність системи.

Ефективність системи полягає у її можливості адаптовувати навчальні матеріали відповідно до потреб й запитів кожного користувача на основі статистичного аналізу результатів тестування. Це надає змогу зосередитися на тих сторонах учня, що потребують покращення, підвищуючи його рівень володіння мовою. Також це проявляється у її здатності забезпечувати динамічний зворотний зв'язок, що сприяє більш глибокому розумінню навчального прогресу та підвищенню мотивації користувачів. Крім того, система автоматизує процес створення тестів, що значно скорочує витрати часу для викладачів та адміністраторів.

Мета створення такої системи полягає у вирішенні проблем, що пов'язані із вивченням іноземних мов, а також у забезпеченні ефективного та персоналізованого навчання.

Для досягнення поставлених цілей необхідно виконати наступні задачі:

- вибрати тему;
- провести аналіз за темою предметної області, включаючи огляд літературних та інших джерел;

- здійснити системний аналіз розроблюваної системи, визначити її основну функцію та структуру;
- описати і проаналізувати комплекс вимог до системи;
- створити план проєкту;
- визначити засоби та методи, які використовуються для вирішення поставлених задач;
- реалізувати систему та навести результати виконання проєкту.

Об'єкт дослідження: автоматизація процесу освоєння іноземних мов з використанням статистичного аналізу.

Предмет дослідження: засоби та методи проєктування та розроблення системи освоєння іноземної мови з використанням статистичного аналізу.

**Ключові слова:** освоєння іноземної мови, статистичний аналіз, веб-сайт, інтернет-ресурс, автоматизація навчання.

Перелік використаних літературних джерел:

1. Кастельс М. Інтернет-галактика: Роздуми про Інтернет, бізнес і суспільство. Оксфорд: Oxford University Press, 2001.

## ABSTRACT

Myrosh Y. M., Mykich Kh. I. (supervisor). Intelligent System for Foreign Language Acquisition Using Statistical Analysis. Master's Qualification Thesis. – National University "Lviv Polytechnic," Lviv, 2025.

### Extended Abstract

In today's interconnected world, proficiency in foreign languages plays a crucial role in both personal growth and professional advancement [1]. Despite its importance, traditional approaches to language learning frequently face numerous obstacles, including a lack of tailored learning experiences, challenges in accurately measuring progress, and insufficient motivation among learners. These issues are further amplified in the era of globalization, where the ability to communicate in multiple languages serves as a gateway to a wide range of opportunities. From pursuing higher education and advancing in one's career to fostering meaningful cross-cultural interactions, language skills have become an essential asset in navigating the complexities of modern life.

The foreign language acquisition system, based on the use of statistical analysis, has been developed to address a range of critical tasks associated with the learning process. Its primary goal is to implement automation in language learning, significantly simplifying the material acquisition process while ensuring a personalized approach for each user. This personalization takes into account the user's knowledge level, learning pace, and individual needs. Additionally, the system aims to enhance the accuracy and efficiency of knowledge assessment, enabling more objective results and providing recommendations for further improvement.

The system's core functionality encompasses several key stages. First, users complete a registration process, which allows the creation of a personalized profile to store data about their progress. Afterward, users have the option to choose learning resources, such as books, that align with their personal interests and language proficiency. Using these selected materials, the system autonomously creates customized tests aimed at assessing the user's understanding of the content. After completing the tests, the results are analyzed, and the user receives detailed, personalized recommendations aimed at improving weaker areas and

reinforcing stronger skills.

In this way, the system provides a comprehensive approach to language learning, making it more effective, convenient, and adaptable to the needs of each individual user. The system is implemented as a web application, ensuring accessibility for a wide range of users via the Internet. The website is designed to function seamlessly across various devices (smartphones, tablets, laptops, desktops) and makes users independent of the operating system on which the site is accessed. The use of modern technologies, such as Angular for the front-end, .NET for the back-end, and MS SQL Server for database management, ensures the system's reliability, performance, and interactivity.

The system's effectiveness lies in its ability to adapt learning materials to the needs and requests of each user based on statistical analysis of test results. This enables the system to focus on the areas where the learner requires improvement, thereby enhancing their language proficiency. Additionally, the system provides dynamic feedback, which contributes to a deeper understanding of learning progress and boosts user motivation. In addition, the system streamlines the test creation process by automating it, which greatly minimizes the time and effort needed from teachers and administrators.

The primary objective of developing such a system is to overcome the common obstacles in foreign language learning while delivering a more efficient, tailored, and engaging educational experience for learners.

The goal of creating such a system is to address the challenges associated with foreign language learning and to provide effective and personalized education.

To accomplish the outlined objectives, the following tasks were carried out:

- Select the topic;
- Conduct an analysis of the subject area, including a review of literature and other sources;
- Perform a systematic analysis of the system under development, defining its primary function and structure;
- Describe and analyze the set of requirements for the system;
- Develop a project plan;

- Identify the tools and methods used to solve the outlined tasks;
- Implement the system and present the project results.

**Object of research:** automation of the foreign language acquisition process using statistical analysis.

**Subject of research:** tools and methods for designing and developing a foreign language acquisition system using statistical analysis.

**Keywords:** foreign language acquisition, statistical analysis, website, internet resource, learning automation.

**References:**

1. Castells, Manuel. 2001. *The Internet Galaxy: Reflections on the Internet, Business, and Society*. Oxford: Oxford University Press.

## ДОДАТКИ

### Додаток А

ParameterValidator.cs

```
public class ParameterValidator : IParameterValidator
{
    private readonly IStringValidator _stringValidator;

    public ParameterValidator(IStringValidator stringValidator)
    {
        _stringValidator = stringValidator;
    }

    public void ValidationEmail(string email)
    {
        _stringValidator.IsEmptySpace(email, "Email");
    }

    public void ValidationPhoneNumber(string phoneNumber)
    {
        var typeOfParam = "Phone number";
        _stringValidator.IsLength(13, phoneNumber, "Phone number");
        _stringValidator.IsLetters(phoneNumber, "Phone Number");
        _stringValidator.IsEmptySpace(phoneNumber, "Phone number");
        _stringValidator.IsStringContains(phoneNumber, "+380", "Phone number");
    }
}
```

TestsByBookContext.cs

```
namespace TestsByBook.DAL.Context
{
    public class TestsByBookContext : IdentityDbContext<TestsByBookUser>
    {
        public TestsByBookContext(DbContextOptions<TestsByBookContext> options) : base(options)
        {

```

```

}

public DbSet<Book> Books { get; set; }
public DbSet<Author> Authors { get; set; }
public DbSet<Sentence> Sentences { get; set; }
public DbSet<Word> Words { get; set; }
public DbSet<Test> Tests { get; set; }
public DbSet<Answer> Answers { get; set; }

protected override void OnModelCreating(ModelBuilder builder)
{
    base.OnModelCreating(builder);
}
}
}
}

```

TestsService.cs

```

namespace TestsByBook.BL.Services
{
    public class TestsService: ITestsService
    {
        private readonly UserManager<TestsByBookUser> _userManager;
        private readonly IUnitOfWork _database;
        private readonly IMapper _mapper;
        private readonly IWordSelector _wordSelector;
        private readonly IJwtToken _jwtToken;
        private const string WORD_REPLACER = "___";
        public TestsService(IUnitOfWork database,
            UserManager<TestsByBookUser> userManager,
            IMapper mapper,
            IWordSelector wordSelector,
            IJwtToken jwtToken)
        {
            _database = database;

```

```

_userManager = userManager;
_mapper = mapper;
_wordSelector = wordSelector;
_jwtToken = jwtToken;
}

```

```

public async Task<IEnumerable<TestDto>> GetTestsAsync(TestRequest testRequest)
{
var claims = _jwtToken.GetPrincipalFromExpiredToken(testRequest.Token)?.Claims;

```

```

if (claims == null)
{
throw new ArgumentException("Some trouble with your account");
}

```

```

var userId = claims.First().Value;
var user = await _userManager.FindByIdAsync(userId);

```

```

if (user == null)
{
throw new ArgumentException("Some trouble with your account");
}

```

```

var tests = await _database.TestRepository.GetTestsByUserIdAndBookId(userId,
testRequest.BookId).ToListAsync();

```

```

var existedTestsCount = tests.Count();
var additionalTestsCount = testRequest.CountOfTest - existedTestsCount;
IEnumerable<Test> generatedTests = new List<Test>();
if (additionalTestsCount > 0)
{
generatedTests = await GenerateNewTestsAsync(userId, testRequest.BookId, additionalTestsCount,
existedTestsCount);
await _database.SaveAsync();
}

```

```

    }

    tests.AddRange(generatedTests);

    return _mapper.Map<IEnumerable<Test>,
IEnumerable<TestDto>>(tests.Take(testRequest.CountOfTest));
    }

    private async Task<IEnumerable<Test>> GenerateNewTestsAsync(string userId, int bookId, int
takeCount, int skipCount)
    {
        var rand = new Random();
        var knownCountOfWords = 100;
        var additionalCountForNullableCase = 5;
        takeCount += additionalCountForNullableCase;
        skipCount = rand.Next(skipCount, knownCountOfWords - takeCount);

        var words = await
_database.WordRepository.GetWordsByBookId(bookId).Skip(skipCount).Take(takeCount).Select(x =>
x).ToListAsync();

        var allSentences = await _database.SentenceRepository.GetSentencesByBookId(bookId).Select(x
=> x.TextOfSentence).ToListAsync();

        var tests = new List<Test>();

        foreach (var word in words)
        {
            string sentence = allSentences.FirstOrDefault(x => x.Contains(word.CleanWord));

            var answersWords = await GetAnswersAsync(word.CleanWord);
            if (sentence != null)
            {

```

```

tests.Add(new Test()
{
    TestText = sentence.Replace(word.CleanWord, WORD_REPLACER),
    Answers = answersWords.Select(x => new Answer() { AnswerText = x, IsCorrect = x ==
word.CleanWord }).ToList(),
    WordId = word.Id,
    UserId = userId
});
}

allSentences.Remove(sentence);
}

await _database.TestRepository.AddTestsRangeAsync(tests.ToArray());

return tests;
}

private async Task<List<string>> GetAnswersAsync(string word)
{
    var res = (await _wordSelector.GetReletedWordsAsync(word)).ToList();

    res.Add(word);

    return res;
}

public async Task<IEnumerable<int>> CheckTestsAsync(IDictionary<int, string> testsAnswers)
{
    var ids = testsAnswers.Keys.ToArray();
    var tests = await _database.TestRepository.GetTestsRangeByIdsAsync(ids);

    var result = tests.Where(x => x.Answers.FirstOrDefault(y => y.AnswerText ==
testsAnswers[x.Id]).IsCorrect)

```

```
.Select(x => x.Id);
```

```
foreach (var testAnswer in testsAnswers)
```

```
{
```

```
await _database.TestRepository.DeleteByIdAsync(testAnswer.Key);
```

```
}
```

```
return result;
```

```
}
```

```
}
```

```
}
```