

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

«Вебдодаток для вивчення іноземних мов»

Виконала студентка групи ПЗк-21-1
МАЧУШНИК Олена Олексіївна

Керівник роботи:
ЧИЖМОТРЯ Олена Геннадіївна

Рецензент:
САВІЦЬКИЙ Роман Святославович

Житомир – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

Зав. кафедри інженерії
програмного забезпечення
Тетяна ВАКАЛЮК
«26» лютого 2024 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **МАЧУШНИК Олена Олексіївна**

Керівник роботи: **ЧИЖМОТРЕ Олена Геннадіївна**

Тема роботи: **«Вебдодаток для вивчення іноземних мов»**,

затверджена Наказом закладу вищої освіти від **«23» лютого 2024 р., №74/с**

Вихідні дані для роботи: розробка вебдодатку для вивчення іноземних мов

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів,
що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Чижмотря О.Г.	27.02.2024 р.	27.02.2024 р.
2	Чижмотря О.Г.	27.02.2024 р.	27.02.2024 р.
3	Чижмотря О.Г.	27.02.2024 р.	27.02.2024 р.

РЕФЕРАТ

Кваліфікаційна робота складається з програмного продукту та пояснювальної записки. Пояснювальна записка містить опис розробленого вебдодатку з вивчення іноземних мов.

Текстова частина викладена на 81 сторінці друкованого тексту. Пояснювальна записка містить 19 сторінок додатків. Список використаних джерел містить 30 найменувань та займає дві сторінки. У роботі наведено 76 рисунків та 17 таблиць. Загальний обсяг роботи – 100 сторінок.

Метою роботи є створення вебдодатку для вивчення іноземних мов, що забезпечує інтерактивне та ефективне навчання з використанням сучасних технологій. Мета включає розробку інтуїтивно зрозумілого інтерфейсу користувача, впровадження адаптивних методів навчання, що відповідають різним рівням знань користувачів, а також проведення тестування додатку для забезпечення його безперебійної роботи.

У роботі визначено основні завдання на розробку вебдодатку для вивчення іноземних мов та обґрунтовано вибір технологій розробки, описано структуру програмного продукту та роботу алгоритмів, вказано результати тестування вебдодатку.

Ключові слова: ВЕБДОДАТОК, ВИВЧЕННЯ ІНОЗЕМНИХ МОВ, PHP, HTML, CSS, JAVASCRIPT, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ВЕБ-РОЗРОБКА.

					ІПЗ.КР.Б – 121 – 24 – ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		О.О.Мачушник			Вебдодаток для вивчення іноземних мов Пояснювальна записка	Літ.	Арк.	Аркушів
Керівник		О.Г.Чижмотря					3	100
						Житомирська політехніка, група ІПЗк-21-1		
Рецензент		Р.С. Савіцький						
Зав. каф.		Т.А.Вакалюк						

ABSTRACT

The qualification work consists of a software product and an explanatory note. The explanatory note contains a description of the developed web application for learning foreign languages.

The text part is laid out on 81 pages of printed text. The explanatory note contains 19 pages of annexes. The list of used sources contains 30 names and occupies two pages. The work contains 76 figures and 17 tables. The total volume of work is 100 pages.

The goal of the work is to create a web application for learning foreign languages that provides interactive and effective learning using modern technologies. The goal includes developing an intuitive user interface, implementing adaptive learning methods that suit different levels of user knowledge, and testing the application to ensure it runs smoothly.

The work defines the main tasks for the development of a web application for learning foreign languages and substantiates the choice of development technologies, describes the structure of the software product and the operation of algorithms, and indicates the results of testing the web application.

Keywords: WEB APPLICATION, LEARNING FOREIGN LANGUAGES, PHP, HTML, CSS, JAVASCRIPT, SOFTWARE, WEB DEVELOPMENT.

					ІІЗ.КР.Б – 121 – 24 – ІІЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

ЗМІСТ

РЕФЕРАТ	3
ABSTRACT	4
ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	6
ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ, МЕТОДІВ ТА ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧІ.....	9
1.1. Постановка задачі	9
1.2. Аналіз існуючого програмного забезпечення.....	11
1.3. Обґрунтування вибору інструментальних засобів	16
Висновки до першого розділу	17
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	18
2.1. Технічне завдання на розробку системи	18
2.2. Аналіз вимог до програмного продукту.....	19
2.3. Алгоритм роботи та стани програмної системи.....	23
2.4. Об'єктно-орієнтована модель системи.....	25
2.5. Взаємодія об'єктів системи	27
2.6. Розробка бази даних системи	30
Висновки до другого розділу.....	36
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З СИСТЕМОЮ ДОДАТКУ	37
3.1. Порядок встановлення та налаштування системи.....	37
3.2. Розробка основних функцій вебдодатку	38
3.3. Опис інструкції користувача	46
3.4. Тестування роботи програмного продукту	73
Висновки до третього розділу	77
ВИСНОВКИ.....	78
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	79
ДОДАТКИ.....	81

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ПЗ – програмне забезпечення;

HTML – HyperText Markup Language, мова розмітки для створення та структурування вебсторінок;

CSS – Cascading Style Sheets, мова стилів, що використовується для опису зовнішнього вигляду вебсторінок;

PHP – Hypertext Preprocessor, серверна мова програмування, використовується для створення динамічних вебсторінок;

DOM – Document Object Model, об'єктна модель документа;

БД – база даних;

SQL – Structured Query Language, мова програмування, що використовується для управління базами даних;

MVC – Model-View-Controller, архітектурний шаблон, що розділяє програму на три взаємодіючі компоненти: модель, вид та контролер;

CRUD – Create, Read, Update, Delete, основні операції для роботи з базами даних, які включають створення, читання, оновлення та видалення даних;

UML – Unified Modeling Language, уніфікована мова моделювання, що використовується для створення діаграм.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

ВСТУП

Актуальність теми.

У сучасному світі вивчення іноземних мов набуває важливого значення з різних точок зору. Це необхідно для кар'єрного зростання, оскільки багато компаній працюють у міжнародному середовищі. Вміння спілкуватися мовами інших країн відкриває нові можливості для міжнародних переїздів та інтеграції в інші культури.

Вивчення іноземних мов не лише збагачує культурний досвід, але й надає доступ до різноманітних інформаційних ресурсів і можливостей у мережі Інтернет. Подорожі в сучасному світі відкривають нові горизонти, а знання іноземних мов стає ключем до глибшого розуміння та оцінки інших культур і традицій. Вивчаючи конкретну мову, люди дізнаються історію відповідної країни та її культуру.

Крім того, знання іноземних мов відкриває безмежні можливості для використання Інтернету. Люди, які володіють іноземними мовами, можуть легко спілкуватися онлайн з мешканцями інших країн, розширюючи своє коло спілкування та знаходячи нових друзів. Це сприяє культурній толерантності, міжнародному співробітництву та розвитку особистих і професійних навичок [13].

Метою кваліфікаційної роботи є дослідження сучасних алгоритмів та методів реалізації вебдодатків для вивчення іноземних мов, аналіз архітектур сучасних онлайн-платформ та розробка вебдодатку з інтуїтивно зрозумілим інтерфейсом користувача.

Для досягнення поставленої мети розробки вебдодатку, необхідно виконати наступні завдання:

- здійснити порівняльний аналіз існуючих аналогів;
- провести детальний аналіз вимог вебдодатку;
- обрати технології для реалізації вебдодатку;
- розробити структуру бази даних;

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

5. розробити UML діаграми для візуалізації процесів і взаємодії компонентів системи;
6. написати програмний код для сторони користувача і для панелі адміністратора;
7. провести функціональне тестування вебдодатку.

Об'єктом дослідження є процес проєктування та розробки онлайн-платформи для вивчення іноземних мов, який включає аналіз потреб користувачів, розробку функціональних вимог, проєктування архітектури системи, впровадження програмного забезпечення та його подальше тестування.

Предметом дослідження є використання мов програмування HTML, CSS, JavaScript та PHP для створення вебдодатку, зокрема дослідження їх взаємодії, можливостей інтеграції, а також способів гарантування безпеки даних і високої якості досвіду користувача.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ, МЕТОДІВ ТА ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧІ

1.1. Постановка задачі

Вебдодаток для вивчення іноземних мов – це програма, яка надає користувачам широкий спектр навчальних матеріалів і ресурсів для опанування іноземних мов. Вебдодаток зберігається на віддаленому сервері та працює за допомогою мережі Інтернет, використовуючи інтерфейс браузера.

Використання навчальної веб-платформи пришвидшує процес навчання та робить його інтерактивним, ефективним і доступним.

Вебдодаток запускаються без необхідності встановлення локально на комп'ютер, користувачі можуть вивчати нову інформацію на різних пристроях у будь-який час та будь-якому місці. Це дозволяє людям бути більш гнучкими в організації свого навчального процесу.

Для роботи вебпрограми необхідні вебсервер і база даних. Вебсервер керує обробкою запитів від клієнтів, зокрема, приймає запити на доступ до вебсторінок, обробляє їх і відправляє відповіді назад користувачам.

База даних зберігає всю необхідну інформацію, яку використовує вебпрограма. Вона забезпечує ефективне збереження, доступ, зміну та управління даними, дозволяючи зберігати великі обсяги інформації з високою швидкістю і надійністю. Вебпрограма забезпечує одночасний доступ для багатьох користувачів до однієї версії програми. Це означає, що користувачам не потрібно встановлювати додаток на свої пристрої, тим самим це забезпечує простоту оновлення та обслуговування програмного забезпечення, оскільки всі зміни виконуються на серверній стороні і миттєво стають доступними для всіх користувачів [14].

Вебдодаток для вивчення іноземних мов має переваги зниження витрат на навчання, підвищення швидкості та ефективності вивчення матеріалу, забезпечуює вільний доступ до всієї інформації.

Метою проекту, що розроблюється є створення функціоналу для постійного навчання користувачів іноземним мовам, починаючи від розширення

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

словникового запасу і закінчуючи до граматичних правил та аудіювання. Реалізація вебдодатку дозволить користувачам інтерактивно вивчати іноземні мови за допомогою читання статей, проходження тестів, опрацювання аудіо- та відеоматеріалів, а також читати актуальні статті блогу та замовляти товари пов'язані із іноземними мовами.

Основні вимоги до вебдодатку для ефективного функціонування:

1. дизайн сайту має бути у світлих тонах, використовуючи кольори та логотип компанії;
2. інтуїтивний інтерфейс користувача, проста та логічна навігація для зручності знаходження потрібної інформації;
3. забезпечення швидкодії та високої шкодкості завантаження сторінок;
4. адаптивний дизайн для різних пристроїв;
5. можливість проходження інтерактивних уроків з граматики, аудіювання, читання та письма із виконанням закріплюючого тесту.

Панель адміністратора повинна бути доступна лише після успішної авторизації користувача та перевірки наявності відповідних прав доступу. Для забезпечення ефективної роботи адміністраторів та зручного управління додатком, адмінпанель повинна відповідати наступним вимогам:

1. обмеження прав на перегляд, редагування та видалення даних відповідно до посади;
2. можливість перегляду, редагування та видалення навчальних матеріалів, тестів та відповідей до них;
3. можливість перегляду, редагування та видалення постів блогу;
4. можливість перегляду, редагування та видалення категорій товарів, товарів і замовлень.

Після того як дані змінюються в панелі адміністратора, вони автоматично оновлюються на стороні клієнта.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2. Аналіз існуючого програмного забезпечення

Сучасний ринок вебдодатків для вивчення іноземних мов характеризується високою якістю і широким асортиментом продуктів, спрямованих на різні вікові групи та рівні володіння мовами. Аналіз існуючого програмного забезпечення в галузі вивчення іноземних мов дозволяє виявити ключові напрямки та особливості розвитку цього сегменту. Детально розглянемо кілька сучасних вебдодатків, призначених для навчання іноземним мовам.

Duolingo - <https://www.duolingo.com/>

Duolingo (рис. 1.1) - це одна з найпопулярніших інтерактивних платформ для самостійного вивчення іноземних мов. Вебсайт спрямований на розвиток навичок читання, аудіювання, говоріння та письма через інтерактивні уроки із використанням анімацій персонажів.

Уроки на Duolingo базуються на використанні граматики та лексики в контексті реальних ситуацій. Кожен користувач має можливість починати з базових рівнів складності, які поступово зростають у складності і обсязі з кожним успішно пройденим завданням.

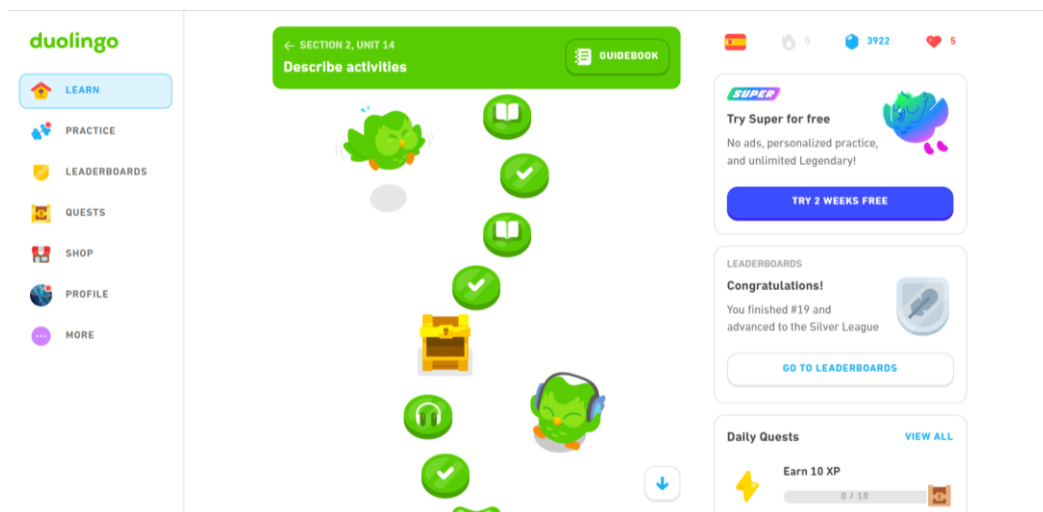


Рисунок 1.1 – Головна сторінка вебсайту Duolingo

Сервіс спрямований на те, щоб під час уроків користувачі могли взяти участь у перекладі вебсайтів, статей та інших документів (рис. 1.2). Він також пропонує ігрове дерево навичок, яке відстежує прогрес кожного користувача, і словниковий розділ для практики вже вивчених слів.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 1.2 – Проходження тестів на вебсайті Duolingo

Користувачі Duolingo заробляють монети (рис. 1.3), що служать мотивацією для активного навчання, пройшовши уроки та виконавши завдання високої якості. Ці монети можна використовувати для розблокування додаткових вправ, отримання віртуальних нагород або участі в рейтингових змаганнях з іншими користувачами.



Рисунок 1.3 – Отримання винагороди після тестів на вебсайті Duolingo

Duolingo використовує метод навчання, що базується на аналізі великої кількості статистичних даних. На кожному етапі система враховує, які питання викликають у користувачів труднощі і які помилки вони роблять. Ці дані потім агрегуються та використовуються для машинного навчання. Таким чином створюються індивідуальні уроки, що адаптовані до потреб кожного користувача [15].

BBC Learning English - <https://www.bbc.co.uk/learningenglish>

BBC Learning English (рис. 1.4) – це інноваційний ресурс, який надає можливості для навчання англійської мови через різноманітні платформи і формати. Він пропонує широкий вибір відеоуроків, аудіоподкастів, статей та інших мультимедійних матеріалів.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

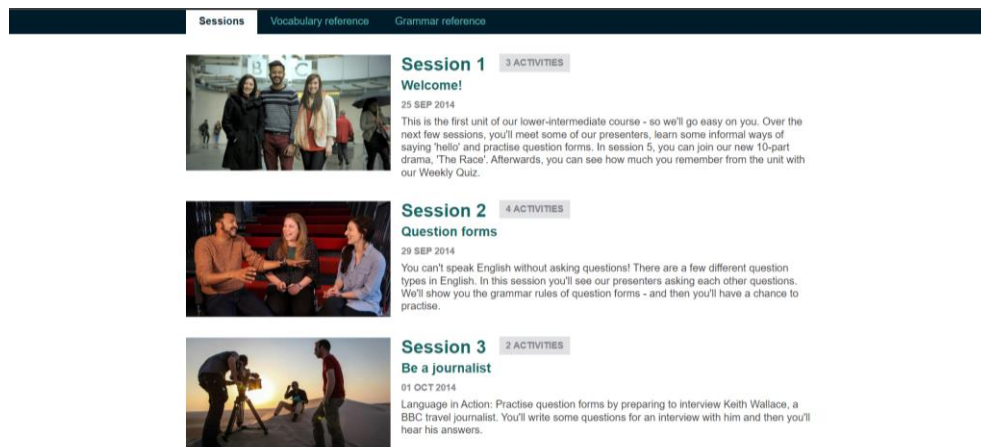


Рисунок 1.4 – Перелік матеріалів на вебсайті BBC Learning English

Вони включають курси для різних рівнів англійської мови, від початкового до продвинутого, а також спеціалізовані тематичні курси, які охоплюють від академічної англійської до підготовки до міжнародних іспитів. Користувачі мають змогу закріпити нові знання після вивчення матеріалу через проходження коротких тестів по темі (рис. 1.5).

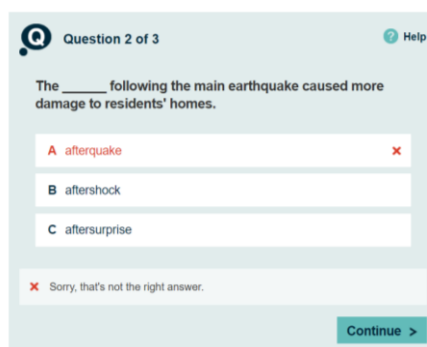


Рисунок 1.5 – Проходження тестів на вебсайті BBC Learning English

BBC Learning English використовує актуальні події і новини для створення матеріалів, що дозволяє користувачам навчатися мови через контекст реальних подій.

British Council LearnEnglish - <https://learnenglish.britishcouncil.org/>

British Council LearnEnglish (рис. 1.6) – це офіційний мультимедійний ресурс, розроблений British Council для навчання англійської мови, який пропонує широкий спектр навчальних матеріалів, включаючи відеоуроки, аудіоподкасти, інтерактивні вправи, статті та ігри.

Choose a grammar lesson



Adjectives and prepositions

Do you know how to use adjectives with prepositions like *interested in* or *similar to*? Test what you know with interactive exercises and read the explanation to help you.

447



Adjectives ending in '-ed' and '-ing'

Do you know the difference between *bored* and *boring*? Test what you know with interactive exercises and read the explanation to help you.

205



Articles: 'a', 'an', 'the'

Do you know how to use *a*, *an* and *the*? Test what you know with interactive exercises and read the explanation to help you.

Рисунок 1.6 – Перелік матеріалів на вебсайті British Council LearnEnglish

Даний ресурс пропонує матеріали для учнів на всіх рівнях від початкового до просунутого, а також спеціалізовані курси для підготовки до іспитів, академічної англійської та професійного використання. Кожний навчальний матеріал містить тест на закріплення знань (рис. 1.7).

Grammar A1-A2: Adjectives and prepositions: 1

Choose the correct word.

I'm really proud ___ you!

☒ of

☐ to

☐ with

7 items remaining

Finish Try again

Рисунок 1.7 – Проходження тестів на вебсайті British Council LearnEnglish

Ці три платформи для вивчення іноземних мов мають велику кількість спільних рис, таких як аудіо та відеоуроки, граматичні ресурси, інтерактивні вправи, графічні інструменти та тести для оцінки знань. Кожен з них також має свої унікальні особливості, наприклад, Duolingo використовує систему нагород для стимулювання користувачів, BBC Learning English ставить акцент на реальні ситуації та British Council LearnEnglish спрямований на практичні тематики.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

Порівняння існуючих аналогів

Критерії	Duolingo	BBC Learning English	British Council LearnEnglish
Тип навчальних ресурсів	Інтерактивні вправи, аудіоподкасти, читання діалогів	Відеоуроки, аудіоподкасти, текстові матеріали, інтерактивні вправи	Відеоуроки, аудіоподкасти, текстові матеріали, інтерактивні вправи
Кількість мов	40+	1	1
Адаптованість до рівнів	Так	Так	Так
Методи навчання	Гра, машинне навчання, самостійне навчання	Відеоуроки, читання, аудіоподкасти	Відеоуроки, читання, аудіоподкасти
Доступність платформи	Веб-сайт, мобільні додатки	Веб-сайт, мобільні додатки	Веб-сайт, мобільні додатки
Соціальна взаємодія	Форуми, групи користувачів	Спільнота користувачів на вебсайті	Спільнота користувачів на вебсайті

Отже, було вирішено розробити власний вебдодаток для вивчення іноземних мов. У даному проєкті увагу слід приділити наступним важливим критеріям, що забезпечать якісний досвід користувачам:

- простий та зручний інтерфейс: необхідно створити інтуїтивно зрозумілий інтерфейс, що дозволить користувачам швидко освоїти управління та навігацію по вебдодатку.
- чіткість, послідовність та наповненість навчального процесу: потрібно створити насичений навчальний процес, де користувач буде виконувати

різноманітні завдання, проходити рівні та взаємодіяти з різними елементами навчального матеріалу.

1.3. Обґрунтування вибору інструментальних засобів

Вибір інструментів для розробки є критично важливим кроком у створенні будь-якого веб-сайту.

Для створення клієнтської сторони вебдодатка для вивчення іноземних мов було вирішено використовувати HTML, CSS, JavaScript та JQuery.

HTML – це мова розмітки гіпертекстових документів, яка дозволяє структурувати та організовувати вміст кожної вебсторінки, визначаючи заголовки, абзаци, зображення та інші елементи. CSS – це мова стилів, що використовується для створення візуального оформлення вебсторінок, дозволяючи змінювати кольори, шрифти, макет і загальний вигляд сайту [16].

JavaScript є важливою мовою сценаріїв для підвищення інтерактивності вебсайту та швидкості реагування. JQuery – це платформа, розроблена на основі JavaScript, яка дозволяє легше маніпулювати з DOM та використовувати Ajax [17].

Далі необхідно вибрати серверну мову програмування для розробки вебдодатку для вивчення іноземних мов, яка повинна включати розширені можливості для обробки користувацьких запитів та взаємодію з базою даних.

Таблиця 1.2

Порівняння серверних мов програмування

Мова програмування	Синтаксис і читабельність	Взаємодія з базами даних	Асинхронність і швидкодія
PHP	Простий для освоєння	Підтримка багатьох типів баз даних, включаючи MySQL, PostgreSQL.	Менша підтримка для асинхронного програмування порівняно з Node.js.
Python	Чистий і лаконічний	Підтримка різних типів баз даних, з	Можливість асинхронного

	синтаксис, що підходить для початківців та досвідчених	включенням SQLite, PostgreSQL.	програмування з використанням aiohttp, asyncio.
JavaScript (Node.js)	Модульний синтаксис, заснований на JavaScript, іноді вимагає більшої уваги до деталей.	Підтримка різних баз даних, з включенням MongoDB, MySQL, PostgreSQL.	Основна підтримка асинхронного програмування, що дозволяє забезпечити високу швидкодію.

Враховуючи усі фактори кожної мови програмування, для розробки вебдодатку для вивчення іноземних мов була обрана серверна мова програмування PHP. Так як дана мова є нескладною, з великою кількістю доступних ресурсів і документації та забезпечує легку інтеграцію з багатьма типами баз даних.

Висновки до першого розділу

У першому розділі було загально сформульовано завдання, визначено основні вимоги до вебдодатку, проаналізовано існуючі вебплатформи для вивчення іноземних мов з метою аналізу їхнього функціоналу. Враховуючи визначені вимоги до програмного продукту та результати дослідження існуючих аналогів, було вирішено використовувати HTML, CSS, JavaScript, JQuery та PHP для розробки вебдодатку для вивчення іноземних мов.

Отже, можна зробити висновок, що перший розділ кваліфікаційної роботи містить важливу інформацію щодо аналізу наявних на ринку навчальних онлайн платформ, що дозволяє зробити правильний вибір напрямків розробки та забезпечити якість та ґрунтовність навчального процесу.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1. Технічне завдання на розробку системи

1) Назва системи: Навчальний вебдодаток «LingoUni».

Повне найменування системи: Вебдодаток для вивчення іноземних мов.

Коротке найменування системи: Вебдодаток, Програма, Програмне забезпечення.

2) Призначення системи:

Вебдодаток для вивчення іноземних мов призначений для забезпечення користувачів ефективними засобами самостійного вивчення мов через інтерактивні навчальні ресурси. Він надає доступ до різноманітних навчальних матеріалів, включаючи аудіювання, читання, граматику, письмо та подкасти, що дозволяє всебічно покращувати мовні навички. Користувачі можуть взаємодіяти з матеріалами, проходити закріплюючі тести та відслідковувати свій прогрес за допомогою індивідуальної статистики.

3) Вимоги до системи:

Програмне забезпечення має відповідати таким функціональним характеристикам:

- надійність та стабільність роботи;
- доступність через будь-який веббраузер;
- інтуїтивно зрозумілий та зручний інтерфейс користувача;
- вбудована система навчальних матеріалів для різних рівнів володіння мовою;
- можливість інтеграції з зовнішніми ресурсами для додаткового матеріалу;
- система відстеження прогресу користувачів та статистика навчання;
- надання інтерактивних вправ та тестів для закріплення знань;
- можливість адаптації під різні платформи та пристрої;

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

- функціонал зворотного зв'язку для отримання пропозицій від користувачів;
- інтуїтивно зрозумілий інструмент для додавання, редагування та видалення лінгвістичними таблицями, таблицями блогу та товарів;
- наявність декількох ролей.

4) Вимоги до інтерфейсу:

Інтерфейс додатку для вивчення іноземних мов має бути інтуїтивно зрозумілим і легким у використанні навіть для новачків. На головній сторінці користувачі повинні мати швидкий доступ до основних функцій додатку, таких як навчальні матеріали, блоги, магазин та кабінет. Важливо, щоб дизайн інтерфейсу був з акцентом на зручність навігації. Розділи повинні бути логічно організовані, а кнопки та посилання – зрозумілими і легко розпізнаваними. Необхідно використати кольорову гаму, яка не втомлює очі, забезпечує комфортне тривале користування додатком.

Важливим є також адаптивний дизайн, який дозволить коректно відображати інтерфейс на різних пристроях – від смартфонів до настільних комп'ютерів. Кожна сторінка додатку повинна містити зрозумілі заголовки, чіткі іконки та достатньо простору між елементами.

Інтерактивні елементи, такі як кнопки для лайків, коментарів, додавання в кошик, улюблені категорії та категорії вивчити пізніше, повинні миттєво реагувати на дії користувача.

2.2. Аналіз вимог до програмного продукту

Вебдодаток для вивчення іноземних мов повинен також задовольняти високорівневі вимоги, забезпечуючи користувачам зручність, ефективність та надійність.

Бізнес-вимоги включають забезпечення конкурентоспроможності продукту на ринку, що досягається через надання унікальних функцій, високоякісного контенту та сучасного дизайну. Також важливим є підтримка різних моделей монетизації, включаючи продаж навчальних матеріалів та товарів з фірмовим логотипом.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

Вимоги користувачів для вебдодатку включають можливість реєстрації нових облікових записів, входу в систему, а також відновлення забутого пароля в разі необхідності. Крім того, користувачі повинні мати можливість редагувати свою особисту інформацію, таку як дані і пароль для забезпечення безпеки та зручності використання. Важливим аспектом є також можливість перегляду особистої статистики навчання, що дозволяє користувачам відстежувати свій прогрес та досягнення в процесі вивчення іноземних мов. Також користувачі повинні мати можливість вивчати навчальних матеріалів і проходити закріплюючі тести.

Для зручності користувачів вебдодаток має забезпечити можливість перегляду історії замовлень і оформлення нових замовлень на навчальні матеріали, що дозволяє їм легко отримувати необхідний навчальний матеріал.

Додатково, важливим є функціонал пошуку навчальних матеріалів за фільтрами, що дозволяє користувачам швидко знаходити інформацію за певними критеріями, а також використання сторінки пошуку для отримання доступу до всього навчального контенту в одному місці.

Функціональні вимоги включають забезпечення авторизації користувачів у системі з можливістю збереження необхідної інформації в базі даних та за допомогою файлів cookies. Авторизованим користувачам повинен бути відкритий доступ до персоналізованого функціоналу, такого як зміна особистих даних і управління своїм профілем і зберігання даних у безпечному середовищі.

Нефункціональні вимоги до вебдодатку для вивчення іноземних мов включають забезпечення швидкодії та ефективної роботи системи. Оптимізація завантаження сторінок та відповідей сервера повинна забезпечувати час відповіді не більше 6 секунд для основних сторінок, а складні запити не повинні перевищувати 12 секунд.

Крім того, програма має стабільно підтримувати одночасну роботу 120 користувачів без втрати продуктивності. Важливо, щоб система була готова масштабуватися при необхідності, здатна збільшувати обсяг обробки

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

користувацьких запитів без негативного впливу на якість обслуговування користувачів.

Системні вимоги повинні включати наступне:

- використання веб-сервера з підтримкою PHP версії не нижче 7.0 і MySQL версії 5.6 або вище;
- доступ до Інтернету із швидкістю від 2 Mbps для стабільної взаємодії з сервером і обміну даними;
- мінімальна оперативна пам'ять на сервері не менше 1 Гб;
- веббраузер рекомендовано використовувати від Chrome 86, Firefox 82, Safari 12, або Edge 86 і вище;
- роздільна здатність екрану не менше 1280x800 пікселів для зручного перегляду контенту.

На основі дослідженої інформації, можна зробити діаграму варіантів використання. Діаграма використання (use case diagram) – це тип діаграми UML, яка представляє взаємодію між учасниками і системою, тобто вона надає візуальне представлення того, як користувачі взаємодіють із системою [18].

Дана діаграма ілюструє ключові функції та дії, які користувачі можуть виконувати через веб-додаток, такі як реєстрація, вхід у систему, перегляд статистики, навчання та проходження тестів (рис. 2.1).

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

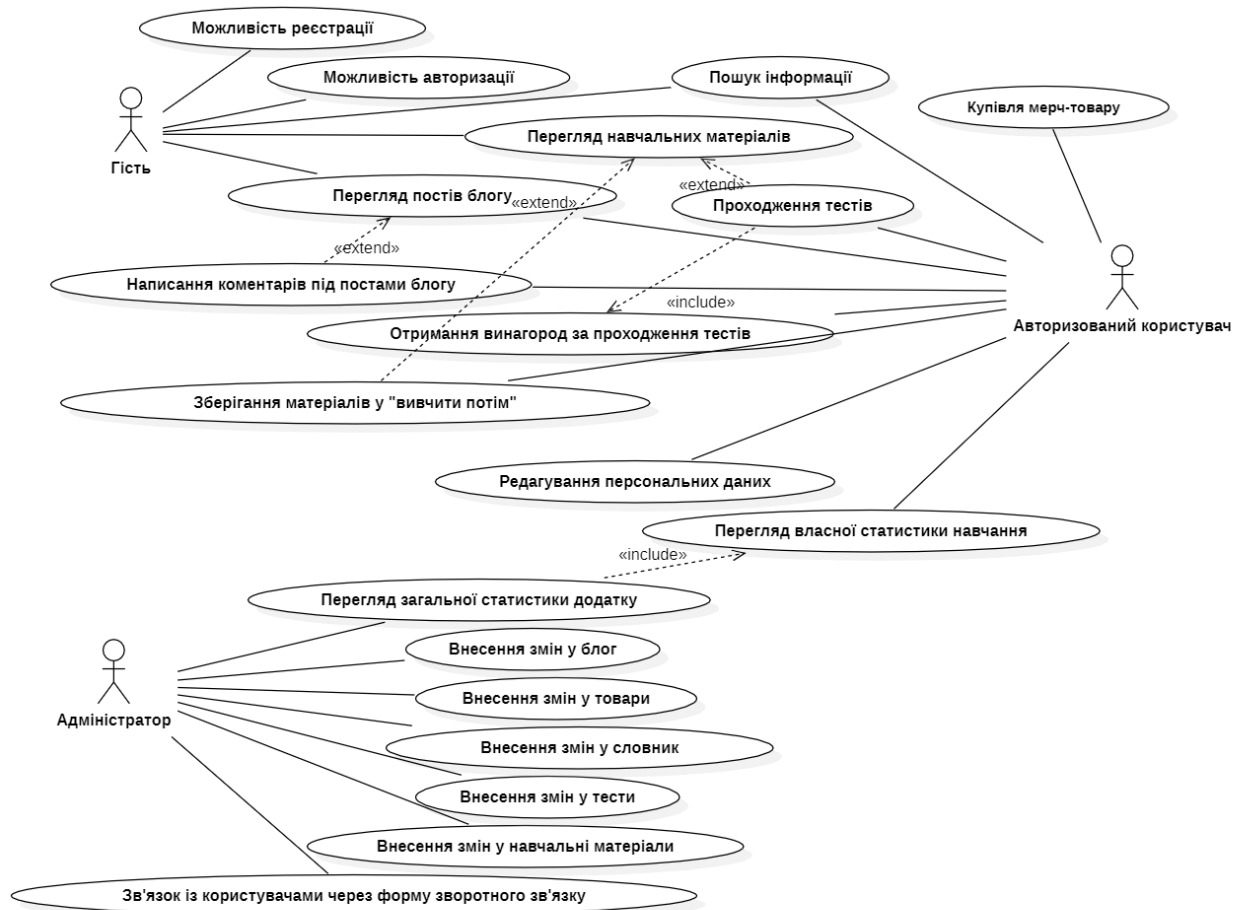


Рисунок 2.1 – Діаграма варіантів використання

У даній діаграмі використання гість має можливість зареєструватися та авторизуватися для отримання доступу до основного функціоналу. Усі користувачі, включаючи авторизованих, можуть використовувати пошукові функції, переглядати навчальні матеріали та читати пости у блозі. Авторизовані користувачі отримують додаткові можливості, такі як покупка товарів у інтернет-магазині, проходження тестів, додавання матеріалів до списку «Вивчити потім», коментування постів у блозі, редагування особистої інформації та перегляд власної статистики навчання.

Адміністратор сайту має повний доступ до управління ресурсами сайту. Він може переглядати загальну статистику, зокрема тенденції реєстрації користувачів. Крім того, адміністратор має можливість вносити зміни у блог, товари, словник, тести та навчальні матеріали. Для зв'язку з користувачами адміністратор використовує форму зворотного зв'язку, що сприяє ефективній комунікації з користувачами сайту.

2.3. Алгоритм роботи та стани програмної системи

Діаграма діяльності є блок-схемою, що ілюструє послідовність дій, вказуючи, як одна діяльність призводить до іншої. Вона демонструє потік процесів, відображаючи взаємозв'язок між різними етапами і показуючи, як виконання однієї дії спричиняє наступні кроки в процесі. Ця діаграма дозволяє краще зрозуміти логіку і послідовність дій, що виконуються в рамках вебдодатку, а також дає змогу виявити можливі точки зупинки або взаємодії між різними компонентами системи [19].

Діаграма процесу реєстрації та авторизації на вебдодатку (рис. 2.2) починається з того, що гість переходить на сторінку входу. Якщо у гостя є акаунт, він вводить свої дані для входу. В разі, якщо акаунт відсутній, система відображає форму реєстрації. Гість вводить необхідні дані для реєстрації, після чого система додає нового користувача до бази даних.

Далі користувач вводить свої дані для входу, і система проводить валідацію цих даних. Якщо введені дані є правильними, система авторизує користувача і відображає його особистий кабінет. В іншому випадку, якщо дані введені неправильно, гість знову повинен ввести дані для входу, і процес повторюється.

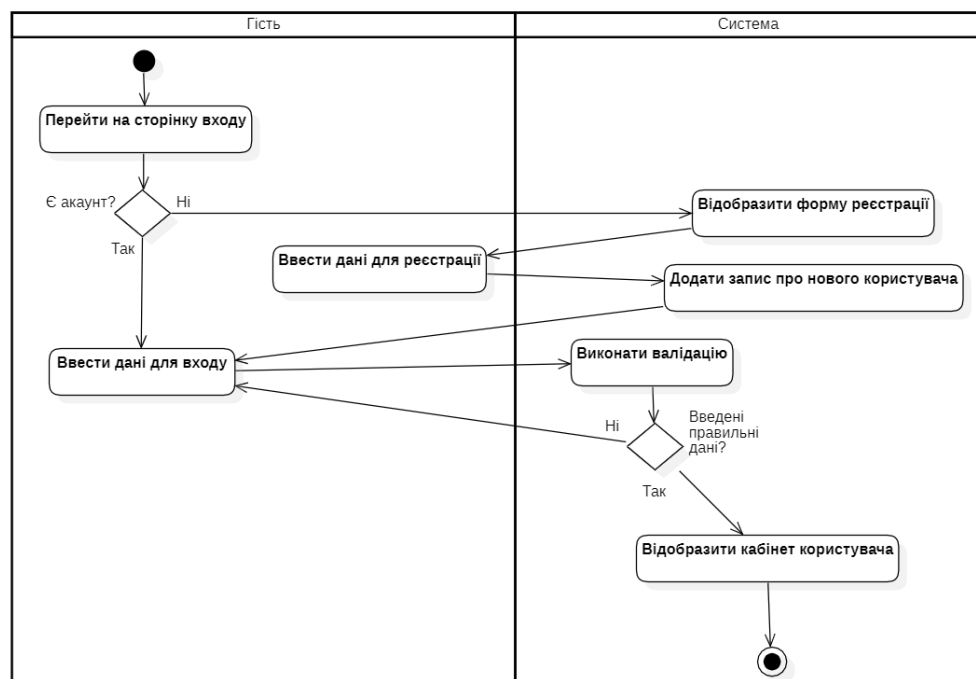


Рисунок 2.2 – Діаграма процесу реєстрації та авторизації

Процес вивчення мов на вебдодатку з вивчення іноземних мов (рис. 2.3) починається з того, що авторизований користувач виконує пошук навчального матеріалу за заданими фільтрами. Якщо пошук не дав результатів, система відображає повідомлення про відсутність результатів, і процес завершується. У випадку, якщо пошук був успішним, система відображає отримані результати.

Користувач переглядає відображені матеріали та обирає конкретний навчальний матеріал для вивчення нової лінгвістичної інформації. Після ознайомлення з матеріалом, користувач має можливість поставити лайк, а також може пройти закріплюючий тест, щоб перевірити свої знання.

Результати тестування, зокрема кількість правильних відповідей, фіксуються і зберігаються у базі даних для подальшого аналізу і відображення в особистій статистиці користувача.

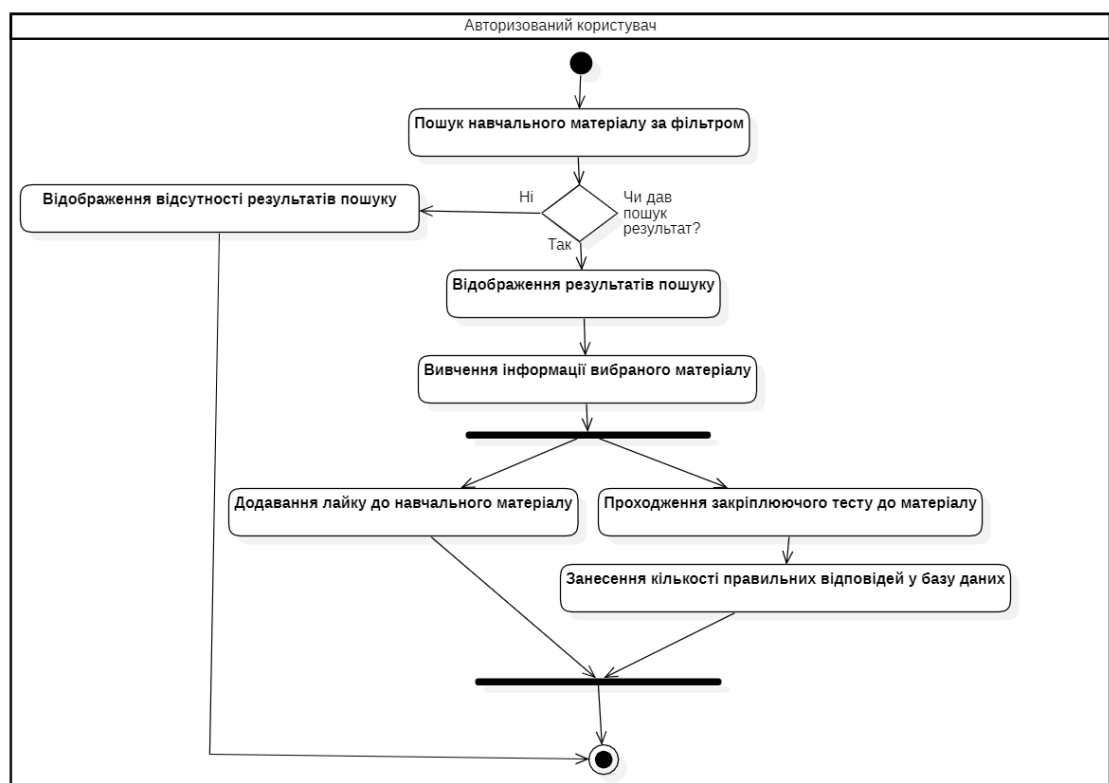


Рисунок 2.3 – Діаграма процесу вивчення мов

Процес покупок в інтернет-магазині (рис. 2.4) у вебдодатку з вивчення іноземних мов починається з того, що користувач виконує пошук товару за заданими фільтрами. Якщо пошук не дав результатів, система відображає повідомлення про відсутність результатів, і процес завершується. У разі успішного пошуку система відображає отримані результати.

Користувач переглядає результати пошуку і обирає конкретний товар для детального перегляду. Після ознайомлення з інформацією про товар, користувач може додати його до кошика. Коли всі необхідні товари додані до кошика, користувач відкриває кошик для перегляду.

На цьому етапі користувач має можливість перевірити актуальність товарів у кошику. Якщо товари більше не відповідають потребам, користувач може оновити кошик, видаливши або додавши нові позиції. Після цього, користувач переходить до етапу оформлення замовлення.

Оформлене замовлення отримує адміністратор вебсайту для подальшої обробки. Адміністратор переглядає всі замовлення, отримані від користувачів, і обробляє вибрані замовлення, забезпечуючи їх вчасну доставку і задоволення потреб клієнтів.

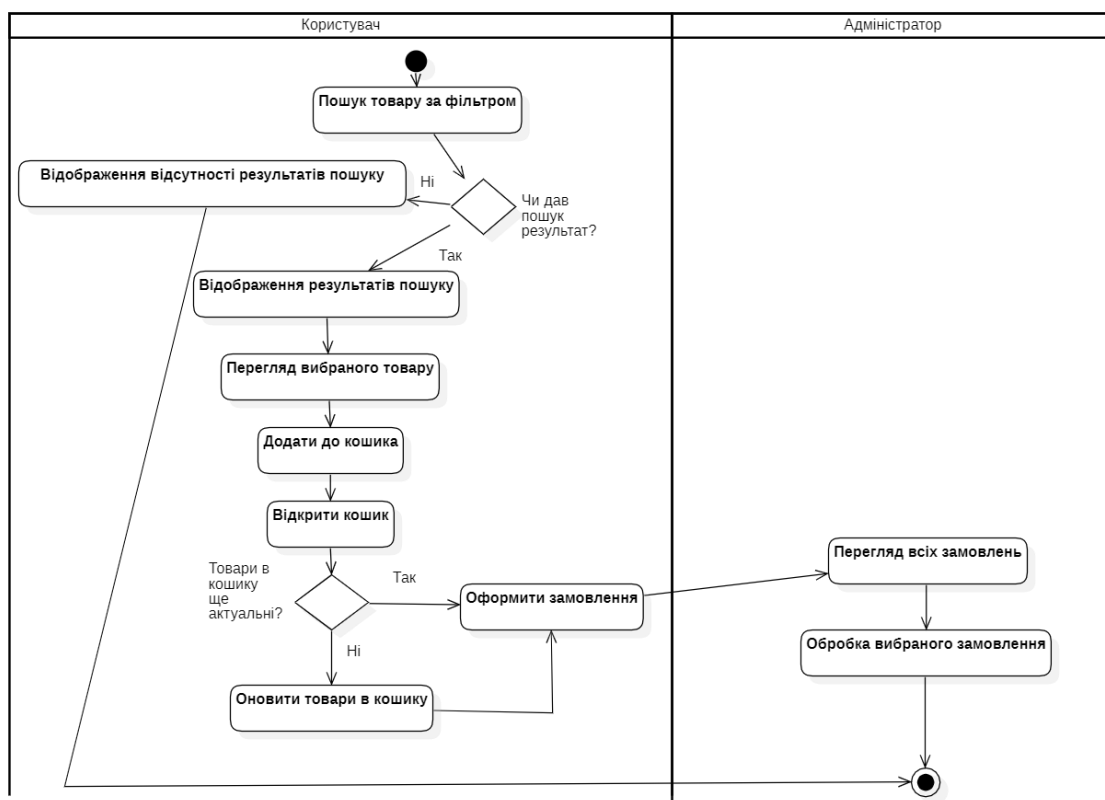


Рисунок 2.4 – Діаграма процесу покупок в інтернет-магазині

2.4. Об'єктно-орієнтована модель системи

Об'єктно-орієнтована модель системи представлена у вигляді діаграми класів (рис. 2.5). Діаграма класів наочно відображає структуру та зв'язки між класами в системі, що дозволяє зрозуміти їх взаємодію та залежності. Вона

містить інформацію про атрибути класів, методи, які вони реалізують, та типи асоціацій між ними [20].

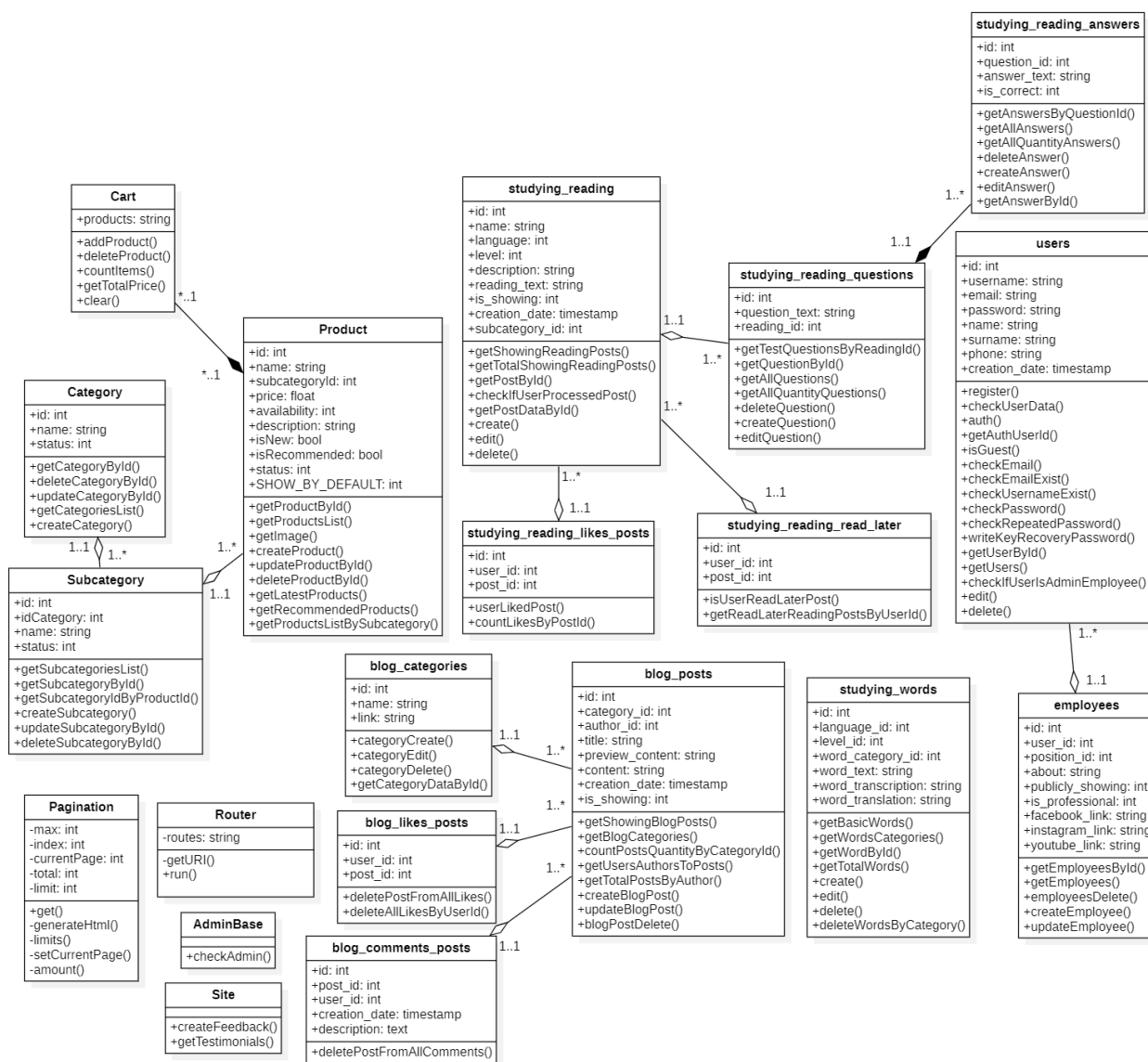


Рисунок 2.5 – Діаграма класів

Діаграма класів вебдодатку для вивчення іноземних мов розкриває ключові модулі, кожен з яких відіграє критичну роль у його функціонуванні. Модуль користувачів зберігає важливі дані, такі як ідентифікатори, особисту інформацію та зв'язки з іншими частинами класами, такими як `studying_reading_likes_posts` і `blog_comments_posts`. Він є основою для всіх інших функцій, включаючи взаємодію зі співробітниками через таблицю `employees`, що містить дані про їхні посади та соціальні мережі.

Модуль `studying_reading` відповідає за навчальний матеріал для читання, зокрема за матеріали з визначеними атрибутами, такими як назва, мова, рівень

складності та текст. Ця частина системи є центральною для процесу вивчення мови через читання, і вона взаємодіє з іншими компонентами, включаючи таблиці `studying_reading_questions` і `studying_reading_answers`, що забезпечують постановку питань та надання відповідей на них.

`Blog_posts` зберігає дані про блоги, включаючи категорії, авторів та контент, що сприяє активному обговоренню серед користувачів. Цей модуль взаємодіє з таблицями `blog_likes_posts` та `blog_comments_posts`, що дозволяє відстежувати популярність та здійснювати коментування.

Не менш важливим є інтернет-магазин, який надає можливість користувачам здійснювати покупки. Таблиці `Product`, `Category` та `Subcategory` дозволяють організовувати товари за категоріями та управляти корзиною покупок через `Cart`.

2.5. Взаємодія об'єктів системи

Діаграма послідовності – це графічне зображення, яке ілюструє взаємодії між об'єктами у певній системі, упорядковані за часом. Вона представляє послідовність повідомлень, які передаються між об'єктами для виконання конкретного сценарію.

Основна мета діаграми послідовності полягає у візуалізації порядку виконання дій і взаємодій у часі, що дозволяє краще зрозуміти і задокументувати роботу системи. Діаграма послідовності також допомагає виявити можливі проблеми у взаємодії об'єктів і оптимізувати процеси [21].

У діаграмі послідовності (рис. 2.6) вивчення іноземної мови у вебдодатку спочатку користувач починає взаємодію з інтерфейсом, ініціюючи подію переходу до навчальних матеріалів. Інтерфейс користувача (`view`) обробляє цю подію та передає її контролеру (`controller`). Контролер запитує модель (`model`) для отримання даних про доступні навчальні матеріали. Модель, у свою чергу, повертає контролеру необхідні дані, після чого контролер завантажує відображення списку навчальних матеріалів та передає цю інформацію у `view`. Інтерфейс користувача відображає користувачу список доступних навчальних матеріалів.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

Після цього користувач обирає конкретний навчальний матеріал для вивчення, знову ініціюючи подію переходу, яку обробляє view. Подія передається контролеру, який запитує у моделі деталі про вибраний матеріал та пов'язаний з ним тест. Модель повертає ці дані контролеру, який завантажує відображення вибраного матеріалу та передає його у view. Інтерфейс користувача відображає користувачу деталі вибраного навчального матеріалу та пов'язаний тест.

Користувач проходить тест, відповідаючи на запитання. Після завершення тесту відповіді користувача передаються до контролера. Модель обраховує результат та записує його у базу даних. Після успішного запису модуль повертає підтвердження контролеру. Контролер оновлює інтерфейс користувача, відображаючи оновлену статистику навчання, що включає результати останнього тесту.

Таким чином, діаграма послідовності ілюструє повний цикл взаємодії користувача з вебдодатком під час вивчення іноземної мови, починаючи від вибору навчального матеріалу до проходження тесту та відображення результатів.

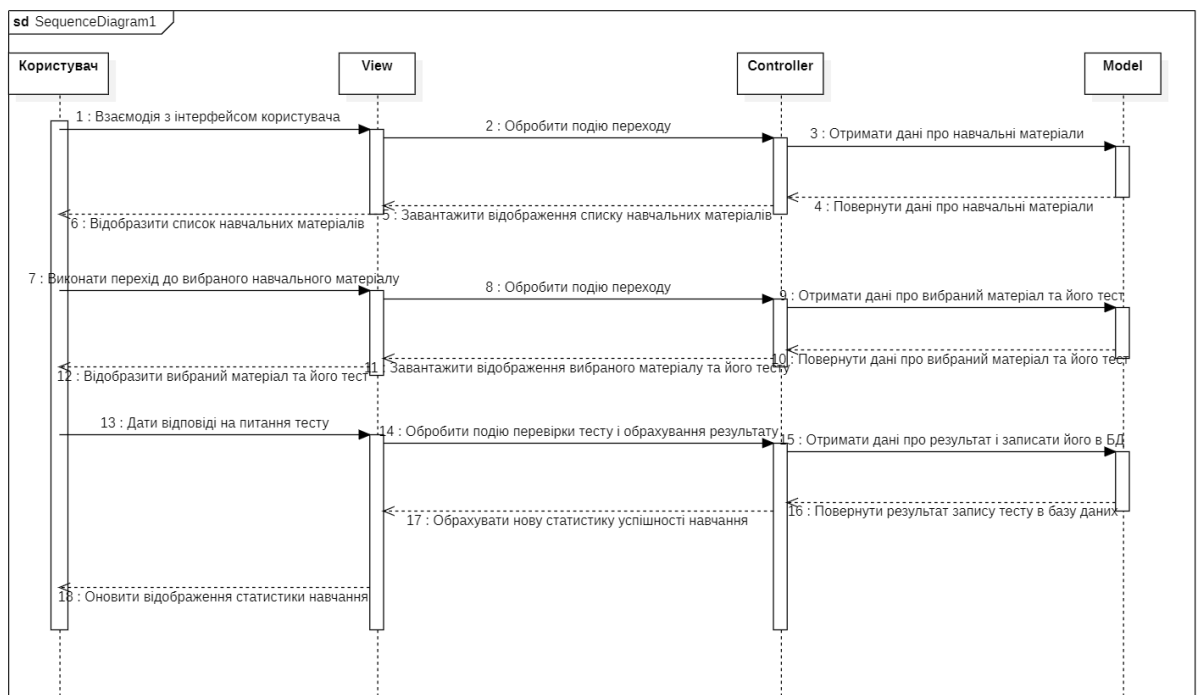


Рисунок 2.6 – Діаграма послідовності вивчення іноземної мови

Діаграма кооперації використовується для показу взаємодії об'єктів, вона відображає як об'єкти взаємодіють для виконання завдань у конкретному варіанті використання. Діаграма кооперації корисна для зображення простих взаємодій між меншою кількістю об'єктів [22].

У діаграмі кооперації (рис. 2.7) вивчення іноземної мови у вебдодатку відбувається наступне: спочатку у процесі вивчення мови користувач ініціює взаємодію з додатком через пошук навчальних матеріалів. Користувач вводить запит, і додаток здійснює пошук відповідних матеріалів у базі даних. Після завершення пошуку система відображає список знайдених навчальних матеріалів. Користувач переглядає цей список, обираючи матеріал, який хоче вивчити.

Обравши навчальний матеріал, користувач отримує детальну інформацію про нього та починає вивчення. На цьому етапі додаток відображає зміст навчального матеріалу, дозволяючи користувачу опановувати нову лінгвістичну інформацію. Після ознайомлення з матеріалом користувач має можливість пройти закріплюючий тест, який перевіряє засвоєні знання.

Під час проходження тесту користувач відповідає на питання, а система обробляє ці відповіді, перевіряє їх правильність та оновлює статистику користувача у базі даних. Після завершення тесту користувач може додати лайк до навчального матеріалу.

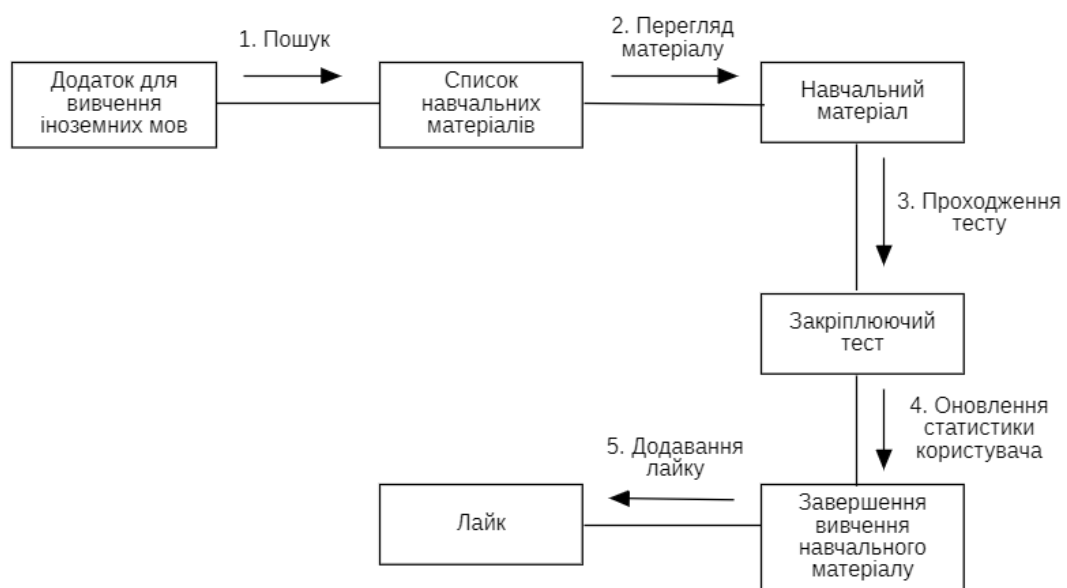


Рисунок 2.7 – Діаграма кооперації вивчення іноземної мови

2.6. Розробка бази даних системи

База даних вебдодатку для вивчення іноземних мов реалізована за допомогою MySQL та містить більше 60 таблиць. На рисунку 2.8 показана схема бази даних. Детальніші рисунки головних таблиць бази даних, включаючи їх структуру, взаємозв'язки та атрибути, наведені у Додатку А.

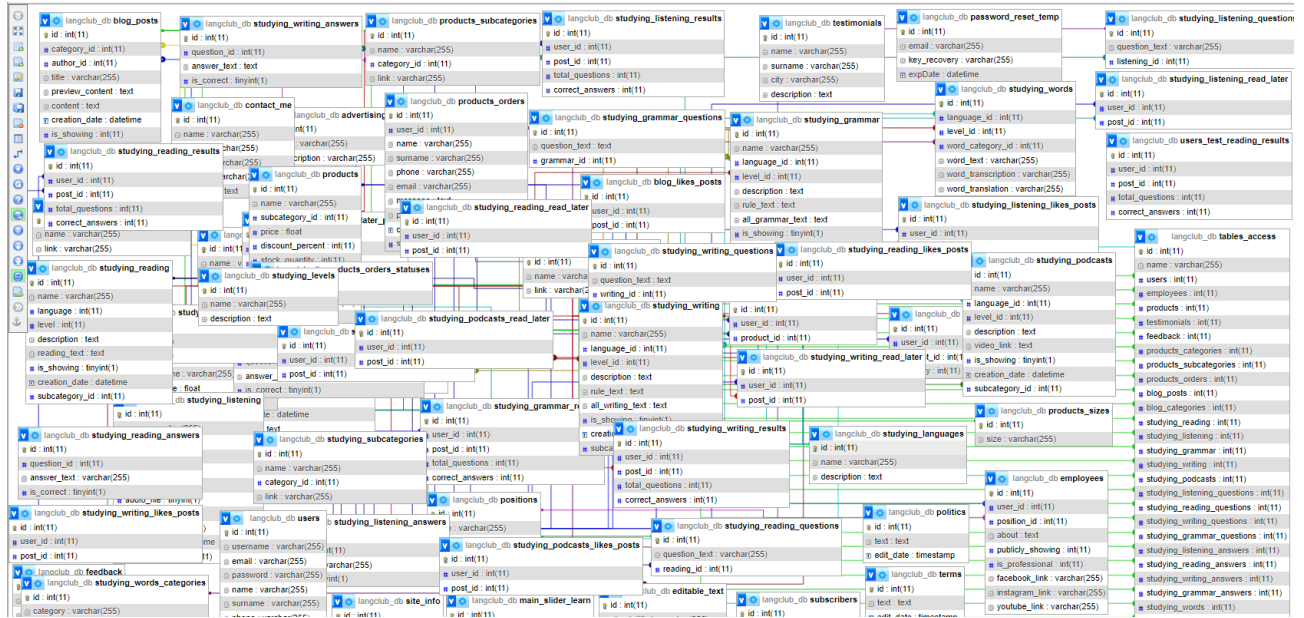


Рисунок 2.8 – Структурна схема БД

Опис ключових таблиць бази даних включає в себе структуру і основні атрибути, необхідні для зберігання та організації даних.

Таблиця «studying_reading» зберігає дані про навчальні матеріали читання.

Таблиця 2.1

Опис полів таблиці «studying_reading»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	—	ID навчального матеріалу читання
name	varchar	—	—	Назва
language	int	—	+	Мова
level	int	—	+	Рівень
description	text	—	—	Опис
reading_text	text	—	—	Навчальний текст
is_showing	tinyint	—	—	Чи відображається

creation_date	datetime	–	–	Дата створення
subcategory_id	int	–	+	Підкатегорія

Таблиця «studying_reading_read_later» зберігає дані про навчальні матеріали читання, які користувачі бажають вивчити пізніше.

Таблиця 2.2

Опис полів таблиці «studying_reading_read_later»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID матеріалу для пізнішого опрацювання
user_id	int	–	+	Користувач
post_id	int	–	+	Пост

Таблиця «studying_reading_likes_posts» зберігає дані про навчальні матеріали читання, які користувачі вподобали і поставили лайк.

Таблиця 2.3

Опис полів таблиці «studying_reading_likes_posts»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID вподобаного матеріалу
user_id	int	–	+	Користувач
post_id	int	–	+	Пост

Таблиця «studying_reading_questions» зберігає дані про питання до відповідних навчальних матеріалів читання.

Таблиця 2.4

Опис полів таблиці «studying_reading_questions»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID питання
question_text	varchar	–	–	Текст питання
reading_id	int	–	+	Навчальний матеріал

Таблиця «studying_reading_answers» зберігає дані про відповіді на питання до відповідних навчальних матеріалів читання.

Таблиця 2.5

Опис полів таблиці «studying_reading_answers»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID питання
question_id	int	–	+	Текст питання
answer_text	varchar	–	–	Текст відповіді
is_correct	tinyint	–	–	Чи правильно

Таблиця «studying_reading_results» зберігає дані про кількість балів набраних користувачами при проходженні закріплюючих тестів до навчальних матеріалів читання.

Таблиця 2.6

Опис полів таблиці «studying_reading_results»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID результату тексту
user_id	int	–	+	Користувач
post_id	int	–	+	Навчальний матеріал
total_questions	int	–	–	Кількість питань
correct_answers	int	–	–	Кількість правильних питань

Таблиця «studying_words» зберігає дані про слова, які представляються користувачу як словник.

Таблиця 2.7

Опис полів таблиці «studying_words»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID слова

language_id	int	—	+	Мова
level_id	int	—	+	Рівень
word_category_id	int	—	+	Категорія слова
word_text	varchar	—	—	Текст слова
word_transcription	varchar	—	—	Транскрипція слова
word_translation	varchar	—	—	Переклад слова

Таблиця «blog_posts» містить дані про пости блогу.

Таблиця 2.8

Опис полів таблиці «blog_posts»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	—	ID поста
category_id	int	—	+	Категорія
author_id	int	—	+	Автор
title	varchar	—	—	Назва
preview_content	text	—	—	Короткий текст
content	text	—	—	Вміст поста
creationg_date	datetime	—	—	Дата створення
is_showing	int	—	—	Чи відображається

Таблиця «blog_comments_posts» зберігає коментарі до відповідних постів блогу.

Таблиця 2.9

Опис полів таблиці «blog_comments_posts»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	—	ID коментаря
post_id	int	—	+	Пост
user_id	int	—	+	Користувач
creation_date	datetime	—	—	Дата створення

description	text	—	—	Текст коментаря
-------------	------	---	---	-----------------

Таблиця «products_cart» зберігає товари, які користувачі додали в кошик.

Таблиця 2.10

Опис полів таблиці «products_cart»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	—	ID запису товару в кошику
user_id	int	—	+	Користувач
product_id	int	—	+	Товар
quantity	int	—	—	Кількість

Таблиця «products_orders» містить замовлення товарів із інтернет-магазину користувачами.

Таблиця 2.11

Опис полів таблиці «products_orders»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	—	ID замовлення
user_id	int	—	+	Користувач
name	varchar	—	—	Ім'я
surname	varchar	—	—	Прізвище
phone	varchar	—	—	Телефон
email	varchar	—	—	Пошта
message	text	—	—	Повідомлення
products	text	—	—	Товари
creation_date	datetime	—	—	Дата створення
status_id	int	—	+	Статус

Таблиця «users» містить інформацію про зареєстрованих користувачів вебдодатку.

Таблиця 2.12

Опис полів таблиці «users»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID користувача
username	varchar	–	–	Нікнейм
email	varchar	–	–	Пошта
password	varchar	–	–	Пароль
name	varchar	–	–	Ім'я
surname	varchar	–	–	Прізвище
phone	varchar	–	–	Телефон
creation_date	datetime	–	–	Дата створення

Таблиця «employees» містить інформацію про зареєстрованих на вебсайті співробітників компанії.

Таблиця 2.13

Опис полів таблиці «employees»

Назва	Тип даних	ПК	ЗК	Опис поля
id	int	+	–	ID співробітника
user_id	int	–	+	Користувач
position_id	int	–	+	Посада
about	text	–	–	Про співробітника
publicly_showing	int	–	–	Чи відображається
is_professional	int	–	–	Чи професіонал
facebook_link	varchar	–	–	Посилання на facebook
instagram_link	varchar	–	–	Посилання на instagram
youtube_link	varchar	–	–	Посилання на youtube

Висновки до другого розділу

У другому розділі кваліфікаційної роботи було розглянуто і проаналізовано технічне завдання на розробку вебдодатку для вивчення іноземних мов. Проведено детальний аналіз вимог до програмного продукту, охоплюючи як функціональні, так і нефункціональні вимоги. Розроблено алгоритм роботи системи та проаналізовано можливі стани програмної системи, що дозволило визначити ключові аспекти її функціонування.

Створено об'єктно-орієнтовану модель вебдодатку, яка включає опис основних об'єктів та їх взаємодію. Взаємодія об'єктів системи була проілюстрована за допомогою UML діаграм, що демонструють послідовність і кооперацію між різними компонентами системи під час виконання основних процесів.

Особлива увага була приділена розробці бази даних системи, визначено структуру ключових таблиць, їх атрибути та зв'язки. У результаті було створено комплексну технічну базу для подальшої розробки та впровадження системи для вивчення іноземних мов, яка включає всі необхідні елементи для її успішного функціонування.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З СИСТЕМОЮ ДОДАТКУ

3.1. Порядок встановлення та налаштування системи

Вебдодаток для вивчення іноземних мов використовує архітектурний шаблон MVC (Model-View-Controller), що сприяє його структурованості та ефективності. У цьому шаблоні модель (Model) відповідає за обробку даних, представлення (View) відповідає за відображення інформації користувачеві, а контролер (Controller) керує взаємодією між моделлю та представленням, обробляючи вхідні дані від користувача та виконуючи відповідні дії. Кожен компонент шаблону вебдодатку розділений на відокремлені об'єкти, що сприяє зручному розширенню, підтримці та управлінню кодом [24].

Структура вебдодатку (рис. 3.1) організована для зручного управління та підтримки. У папці components знаходяться компоненти, такі як пагінація, кошик, перевірка прав адміністратора. Папка config містить конфігураційні файли для налаштування додатку, включаючи підключення до бази даних і роутинг. У папці controllers зберігаються контролери для обробки вхідних запитів та взаємодії з моделями даних і представленнями. Папка models містить моделі для роботи з даними та базою даних. Папка phpmailer включає файли для надсилання електронних листів з вебдодатку [25].

Папка styles містить файли CSS для оформлення веб-сторінок, а папка views зберігає представлення, які відповідають за відображення даних користувачам у вигляді HTML сторінок [27].

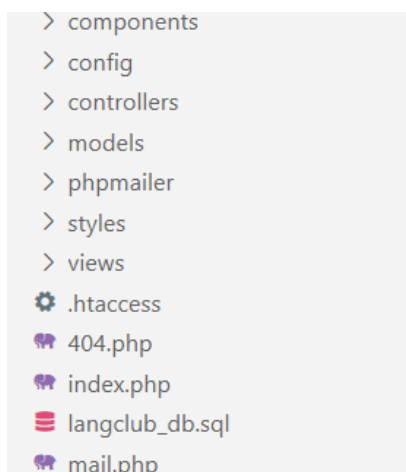


Рисунок 3.1 – Файлова структура вебдодатку

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

Ця структура сприяє організації та зрозумілості коду вебдодатку для вивчення іноземних мов, полегшує роботу для можливості подальшого доповнення коду і забезпечує його масштабованість.

3.2. Розробка основних функцій вебдодатку

Зайшовши на вебсайт, користувач в першу чергу авторизується. Вхід на сайт (рис. 3.2) реалізований через заповнення форми, яка містить пошту, пароль, код безпеки і можливість поставлення відмітки про запам'ятовування даних входу при наступних авторизаціях на відповідному комп'ютері.

Код безпеки генерується випадковим чином через функцію `mt_rand()` і записується у файл `cookies`. Коли користувач вводить код безпеки і натискає кнопку входу відбувається перевірка всіх даних на правильність за допомогою функції `checkUserData()`.

```
$securityCode = mt_rand(1000, 9999);
setcookie('securityCode', $securityCode, time() + (30 * 24 * 60 * 60), '/');

$default_email = '';
$default_password = '';
$isChecked = false;

if (isset($_COOKIE['emailRememberLogin']) && isset($_COOKIE['passwordRememberLogin'])) {
    $isChecked = true;
    $default_email = $_COOKIE['emailRememberLogin'];
    $default_password = $_COOKIE['passwordRememberLogin'];
}

if(isset($_POST['submit'])) {
    $email = $_POST['email'];
    $password = $_POST['password'];
    $code = $_POST['code'];
    $errors = [];
    if(isset($_COOKIE['securityCode']) && $code != $_COOKIE['securityCode']) {
        $errors[] = 'Неправильний код безпеки ';
    }
    else {
        $userId = User::checkUserData($email, $password);
        if($userId == false) {
            $errors[] = 'Неправильні дані для входу на сайт';
        }
        else {
            if (isset($_POST['rememberLoginDataCheckbox'])) {
                setcookie('emailRememberLogin', $email, time() + (30 * 24 * 60 * 60), '/');
                setcookie('passwordRememberLogin', $password, time() + (30 * 24 * 60 * 60), '/');
            }
            User::auth($userId);
            header("Location: /cabinet/");
        }
    }
}

require_once('views/user/login.php');
```

Рисунок 3.2 – Функція вихонання входу на вебсайт

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

Якщо користувач забув пароль, то є можливість його відновити за допомогою використання пошти (рис. 3.3). Користувачу необхідно ввести зареєстровану електронну адресу, на яку прийде повідомлення із згенерованим кодом за допомогою функцій rand(), uniqid(), strlen(), mod5(). Щоб проаналізувати справжність кодів, порівнюється значення початкового коду відновлення, яке для безпеки було занесено в базу даних завдяки функції writeKeyRecoveryPassword(), та кодом який вводить користувач.

```
public function actionForgotPassword() {
    $site_categories = Site::getSiteCategories();
    $studying_categories = Site::getStudyingCategories();
    $studying_subcategories = Site::getStudyingSubcategories();
    $products_categories = Site::getProductsCategories();
    $products_subcategories = Site::getProductsSubcategories();
    $blog_categories = Site::getBlogCategories();
    $site_info = Site::getSiteInfo();
    $advertising_info = Site::getAdvertisingInfo();

    $errors = [];

    if(isset($_POST['submit'])) {
        $email = $_POST['email'];

        if(!User::checkEmail($email)) {
            $errors[] = 'Неправильний email';
        }

        if(!User::checkEmailExist($email)) {
            $errors[] = 'Такий email не зареєстрований';
        }

        if ($errors == false) {
            $key_recovery = md5(2418*2 + strlen($email));

            $addKey = substr(md5(uniqid(rand(),1)),3,10);
            $key_recovery = $key_recovery . $addKey;

            setcookie('emailToRecoverPassword', $email, time() + (30 * 24 * 60 * 60), '/');

            User::writeKeyRecoveryPassword($email, $key_recovery);

            $typeOfEmail = "Email recovery";
            require_once('mail.php');
            header("Location: /user/enter-key-recovery-password");
        }
    }
}
```

Рисунок 3.3 – Функція відновлення забутого паролю

У всіх частинах вебдодатку, які відповідають за відображення контенту, застосовується єдиний підхід до отримання та представлення даних. Після підключення до бази даних за допомогою методу Db::getConnection(),

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

контролери отримують необхідну інформацію з відповідних таблиць, враховуючи критерії сортування чи фільтрації (рис. 3.4). Отриманий результат передається у представлення, де дані відображаються за допомогою циклічних вставок PHP, організованих з використанням HTML тегів.

```
class Grammar {
    const SHOW_BY_DEFAULT = 10;

    public static function getShowingGrammarPosts($page = 1) {
        $page = intval($page);
        $offset = ($page - 1) * self::SHOW_BY_DEFAULT;

        $db = Db::getConnection();

        $result = $db->query('SELECT * FROM studying_grammar WHERE is_showing=1 ORDER BY
            creation_date DESC LIMIT ' . self::SHOW_BY_DEFAULT . ' OFFSET ' . $offset);

        $grammar_posts = [];

        if ($result && $result->rowCount() > 0) {
            $i = 0;
            while ($row = $result->fetch()) {
                $grammar_posts[$i]['id'] = $row['id'];
                $grammar_posts[$i]['name'] = $row['name'];
                $grammar_posts[$i]['language_id'] = $row['language_id'];
                $grammar_posts[$i]['level_id'] = $row['level_id'];
                $grammar_posts[$i]['description'] = $row['description'];
                $grammar_posts[$i]['rule_text'] = $row['rule_text'];
                $grammar_posts[$i]['all_grammar_text'] = $row['all_grammar_text'];
                $grammar_posts[$i]['subcategory_id'] = $row['subcategory_id'];
                $grammar_posts[$i]['creation_date'] = $row['creation_date'];
                $i++;
            }
        }

        return $grammar_posts;
    }
}
```

Рисунок 3.4 – Функція зчитування даних у моделі Grammar

У вебдодатку реалізований функціонал лайків (рис. 3.5) через використання JQuery та AJAX. Функція toggleHeartIcon() призначена для зміни вигляду та взаємодії іконки серця, що використовується для лайку на сторінці вебдодатку. При кліці на іконку спершу перевіряється наявність класу fi-rs-thumbs-up. Якщо він присутній, це означає, що користувач щойно натиснув на лайк і він активується. У такому випадку відбувається зміна класів іконки з fi-rs-thumbs-up на fi-rs-thumbs-up-color, що змінює колір іконки на червоний, що показує, що лайк додано. Далі виконується асинхронний запит до сервера за

допомогою AJAX методом POST до файлу addGrammarLike.php, в якому передаються ідентифікатор користувача та ідентифікатор поста. Після успішного виконання запиту, дані про кількість лайків, які повертаються з сервера, відображаються в іконці.

```
function toggleHeartIcon(icon, user_id, post_id) {
    if (icon.classList.contains('fi-rs-thumbs-up')) {
        icon.classList.remove('fi-rs-thumbs-up');
        icon.classList.add('fi-rs-thumbs-up-color');

        $.ajax({
            url: "../../controllers/ajax/addGrammarLike.php",
            method: "POST",
            data: {user_id: user_id, post_id: post_id},
            success: function(data) {
                icon.innerHTML = " " + data;
            }
        });
    } else {
        icon.classList.remove('fi-rs-thumbs-up-color');
        icon.classList.add('fi-rs-thumbs-up');
        icon.innerHTML = "<?php echo Grammar::countLikesByPostId($post['id']); ?>";

        $.ajax({
            url: "../../controllers/ajax/removeGrammarLike.php",
            method: "POST",
            data: {user_id: user_id, post_id: post_id},
            success: function(data) {
                icon.innerHTML = " " + data;
            }
        });
    }
}
```

Рисунок 3.5 – Функція зміни лайків

Функціональність фільтрації даних на вебсторінках навчальних матеріалів реалізована з використанням різних інтерактивних елементів (рис. 3.6). Наприклад, HTML елементи startYearInput (початковий рік) та endYearInput (кінцевий рік), при кожному введенні нового значення в ці поля запускається функція setAllFilters(), що оновлює вміст фільтрів.

Також відбувається обробка кліку по елементу з ідентифікатором is-new-filter-link, який змінює значення isNewChecked на основі вибраного стану чекбокса. Після зміни значення також викликається setAllFilters() для оновлення результатів фільтрації.

Для обрання декількох варіантів вибору фільтрації одночасно, наприклад обрання декількох мов, відповідає частина коду за обробку кліків на елементи з

класом `.language-filter-link`, які представляють собою набір чекбоксів для вибору мов фільтрації. При кожному кліку оновлюється масив `activeLanguages`, який містить значення обраних мов, і знову викликається `setAllFilters()` для поновлення вмісту сторінки з урахуванням нових фільтрів.

```
startYearInput.addEventListener('input', function() {
    startYearInputValue = startYearInput.value;
    setAllFilters();
});

endYearInput.addEventListener('input', function() {
    endYearInputValue = endYearInput.value;
    setAllFilters();
});

$("#is-processed").click(function(e) {
    showProcessed = document.querySelector('input[name="process"]:checked').value;
    setAllFilters();
});

$("#is-new-filter-link").click(function(e) {
    isNewChecked=false;
    var checkbox = document.getElementById('is-new-filter-link');

    if (checkbox.checked) {
        isNewChecked=true;
    }
    setAllFilters();
});

$(".language-filter-link").click(function(e) {

    activeLanguages = [];
    var checkboxes = document.querySelectorAll('.language-filter-link');

    var checkedValues = [];

    checkboxes.forEach(function(checkbox) {
        if (checkbox.checked) {
            activeLanguages.push(checkbox.value);
        }
    });

    setAllFilters();
});
```

Рисунок 3.6 – Код отримання вибраних даних для фільтрації

Функція `setAllFilters()` викликається при зміні будь-якого з фільтрів або параметрів (рис. 3.7). Вона формує об'єкт `data` з поточними значеннями фільтрів і виконує Аjax-запит до сервера за адресою `../controllers/ajax/grammarCategory.php`. У цьому запиті передаються всі важливі параметри, включаючи активні категорії (`activeCategories`), значення років (`startYearInputValue`, `endYearInputValue`), мови (`activeLanguages`), рівні

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

(activeLevels), статус обробки (showProcessed), стан чекбокса "Новий" (isNewChecked), поточну сторінку (page) та ідентифікатор користувача (userId). Після успішного отримання відповіді від сервера функція оновлює вміст на сторінці, замінюючи вміст елемента з ідентифікатором updated-data-filtered новим вмістом data.

```
var activeCategories = [];
var activeLanguages = [];
var activeLevels = [];
var showProcessed = "all";
var isNewChecked = false;
var startYearInput = document.getElementById('year-start-filter');
var startYearInputValue = startYearInput.value;
var endYearInput = document.getElementById('year-end-filter');
var endYearInputValue = endYearInput.value;
var page = 1;

function setAllFilters() {
    var action = 'fetch_data';
    var categoriesJSON = JSON.stringify(activeCategories);
    var languagesJSON = JSON.stringify(activeLanguages);
    var levelsJSON = JSON.stringify(activeLevels);
    var startYearInputValueJSON = JSON.stringify(startYearInputValue);
    var endYearInputValueJSON = JSON.stringify(endYearInputValue);
    var userId = <?php echo isset($_SESSION['userLoggedIn']) ? $_SESSION['userLoggedIn'] : '0'; ?>;

    $.ajax({
        url: "../../controllers/ajax/grammarCategory.php",
        method: "POST",
        data: { action: action, activeCategories:categoriesJSON, startYearInputValue:startYearInputValue,
            endYearInputValue:endYearInputValue, activeLanguages:languagesJSON, activeLevels:levelsJSON,
            showProcessed:showProcessed, isNewChecked:isNewChecked, page:page, userId:userId},
        success: function(data) {
            $('#updated-data-filtered').html(data);
        }
    });
}
```

Рисунок 3.7 – Код фільтрації граматичних навчальних матеріалів

Користувачі із правами адміністратора мають доступ до адмінпанелі, де можна виконувати CRUD операції над таблицями відповідно до тих доступів до кожної таблиці, які вони мають (рис. 3.8).

						id	name	
☐		Изменить		Копировать		Удалить	1	немає доступу
☐		Изменить		Копировать		Удалить	2	тільки читати
☐		Изменить		Копировать		Удалить	3	читати і оновлювати
☐		Изменить		Копировать		Удалить	4	читати, додавати і оновлювати
☐		Изменить		Копировать		Удалить	5	читати, додавати, оновлювати, видаляти
☐		Изменить		Копировать		Удалить	6	тільки видаляти
☐		Изменить		Копировать		Удалить	7	тільки оновлювати
☐		Изменить		Копировать		Удалить	8	тільки додавати

Рисунок 3.8 – Перелік варіантів доступів до таблиць

Наприклад, функція для обробки запитів до адмінпанелі для відображення списку слів (рис. 3.9) спочатку перевіряє, чи авторизований користувач має права адміністратора за допомогою методу `checkAdmin()`. Якщо користувач не є адміністратором, він перенаправляється на головну сторінку адмінпанелі за допомогою функції `header("Location: /admin")`.

Далі функція отримує ідентифікатор авторизованого користувача за допомогою методу `checkLogged()` класу `User` і отримує список доступів до таблиць за допомогою методу `getTableAccessesList($userId)` класу `User`.

Якщо у адміністратора немає доступу до таблиці навчання слів (`$access_data['studying_words']` не дорівнює 1), то виконується перенаправлення на головну сторінку адмінпанелі. Якщо є доступ, то функція отримує всі слова для відображення на сторінці, викликаючи методи `getAllWords()` і `getTotalWords()`.

Наступним кроком є включення файлу представлення `words.php`, який відповідає за відображення списку слів та пагінації в панелі адміністратора. Функція завершується з поверненням значення `true`, що підтверджує успішне виконання функції.

```
public function actionWords($page = 1) {
    self::checkAdmin();
    $userId = User::checkLogged();
    $access_data = User::getTableAccessesList($userId);

    if($access_data['studying_words']==1) header("Location: /admin");

    $words = Words::getAllWords($page);
    $total_words = Words::getTotalWords();

    $page = preg_replace('/^[^0-9]/i', ' ', $page);

    $pagination = new Pagination($total_words, $page, Words::SHOW_BY_DEFAULT, 'page-');

    require_once('views/admin/words.php');
    return true;
}
```

Рисунок 3.9 – Метод відображення записів таблиці в адмінпанелі

Функція `actionWordsUpdate()` (рис. 3.10) призначена для редагування слова в адмінпанелі. Вона включає в себе всі необхідні кроки для перевірки прав доступу, отримання даних для редагування слова. Обробка поданої форми зі

змінами відправляється методом POST, функція отримує введені дані з форми і викликає метод edit() для збереження змін у базі даних. Після успішного редагування слова відбувається перенаправлення на сторінку зі списком слів за допомогою header("Location: /admin/words").

```
public function actionWordsUpdate($id) {
    self::checkAdmin();

    $userId = User::checkLogged();
    $access_data = User::getTableAccessesList($userId);

    if($access_data['studying_words'] != 3 && $access_data['studying_words'] != 4
    && $access_data['studying_words'] != 5 && $access_data['studying_words'] != 7) {

        header("Location: /admin");
    }

    $languages = Site::getAllLearningLanguages();
    $levels = Site::getAllLearningLanguageLevels();
    $categories = Words::getWordsCategories();

    $word = Words::getWordById($id);

    if(isset($_POST['submit'])) {
        $result = false;

        $language_id = $_POST['language_id'];
        $level_id = $_POST['level_id'];
        $word_category_id = $_POST['word_category_id'];
        $word_text = $_POST['word_text'];
        $word_transcription = $_POST['word_transcription'];
        $word_translation = $_POST['word_translation'];

        Words::edit($id, $language_id, $level_id, $word_category_id, $word_text, $word_transcription, $word_translation);

        header("Location: /admin/words");
    }

    require_once('views/admin/wordsUpdate.php');
    return true;
}
```

Рисунок 3.10 – Метод оновлення наявного запису таблиці в адмінпанелі

Функція actionWordsDelete() (рис. 3.11) є частиною контролера і відповідає за обробку запиту на видалення слова з бази даних. Починаючи з перевірки прав доступу адміністратора, вона також перевіряє доступ користувача до видалення слова за допомогою getTableAccessesList(). Якщо доступ є, функція викликає статичний метод delete() з класу Words. Крім того, функція обробляє можливі помилки під час видалення слова, забезпечуючи відповідні повідомлення користувачу про успішність або невдачу операції. Завдяки цьому, функція actionWordsDelete() гарантує безпечне і контрольоване видалення даних з бази, підтримуючи цілісність системи.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public function actionWordsDelete($id) {

    self::checkAdmin();

    $userId = User::checkLogged();
    $access_data = User::getTableAccessesList($userId);

    if($access['studying_words'] != 5 && $access['studying_words'] != 6){
        header("Location: /admin");
    }

    Words::delete($id);
    header("Location: /admin/words");

    return true;
}

```

Рисунок 3.11 – Метод actionWordsDelete()

Функція delete() у класі Words (рис. 3.12) відповідає за виконання SQL запиту до бази даних для видалення запису з таблиці studying_words за його ідентифікатором. Вона використовує метод getConnection() для отримання з'єднання з базою даних, після чого формує SQL запит з параметром :id, який зв'язується з реальним значенням ідентифікатора за допомогою методу bindParam(). Після підготовки запиту метод execute() виконує його і повертає результат, який вказує на успішність операції видалення.

```

public static function delete($id) {
    $db = Db::getConnection();
    $sql = 'DELETE FROM studying_words WHERE id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);
    return $result->execute();
}

```

Рисунок 3.12 – Метод delete()

3.3. Опис інструкції користувача

Спочатку було обрано розмітку сторінок зі стилями, і на основі цього було розроблено повноцінний вебдодаток для вивчення іноземних мов [29].

Головна сторінка вебдодатку (рис. 3.13) включає навігаційне меню та основну інформацію про навчальну платформу. Основний рекламний акцент робиться на розвиток навичок говоріння та читання іноземними мовами. Крім того, користувачам повідомляється про наявність інтернет-магазину, де можна придбати товари лінгвістичної тематики.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

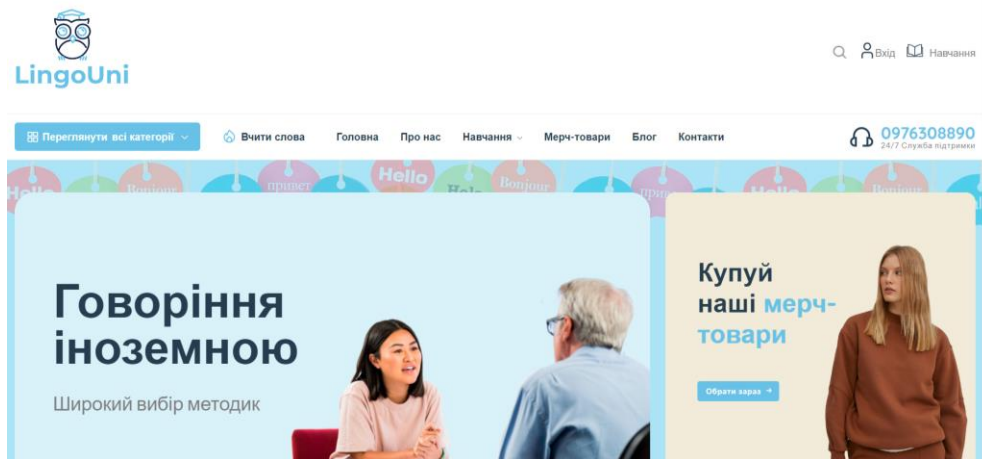


Рисунок 3.13 – Головна сторінка вебдодатку

Крім цього, на головній сторінці також присутні посилання на навчальні сторінки (рис. 3.14), присвячені граматиці, письму та словниковому запасу. Також відображаються рекомендовані товари з інтернет-магазину, включаючи їхні фото, назви, ціни. Авторизовані користувачі мають можливість додавати товари до кошика, тоді як неавторизовані користувачі можуть лише перейти до детального опису товару.

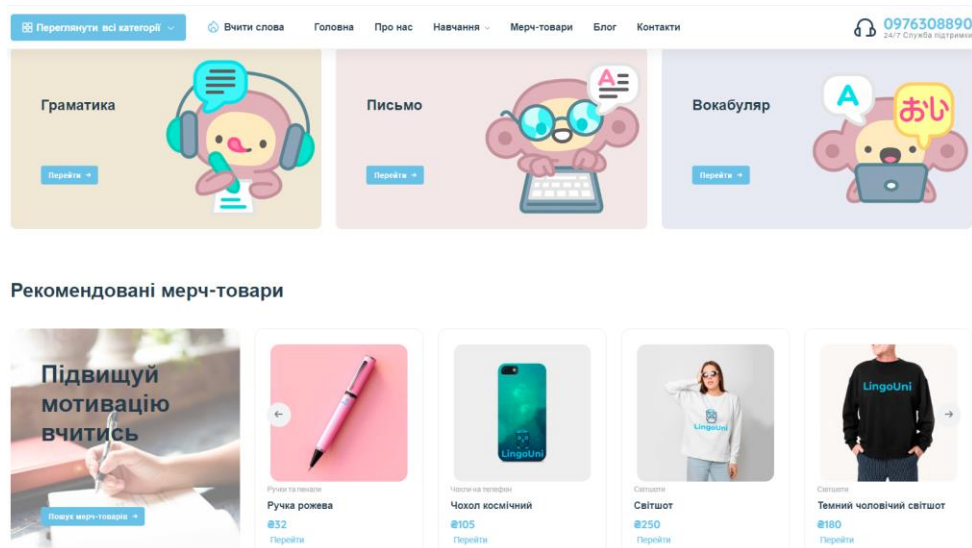


Рисунок 3.14 – Головна сторінка вебдодатку: середня частина

На нижній частині головної сторінки вебсайту розміщений функціональний блок для підписки на розсилку новин та оновлень (рис. 3.15). Цей блок дозволяє користувачам легко підписатися, ввівши свою електронну адресу в спеціально відведене поле.

Після введення електронної адреси користувача система автоматично перевіряє, чи вже існує підписка на цю адресу. Якщо така адреса вже підписана,

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

користувач отримає відповідне повідомлення про помилку. Якщо ж адреса ще не зареєстрована, система додасть її до списку підписників, і користувач побачить підтвердження успішної підписки. Всі повідомлення про помилки або успішну реєстрацію відображаються біля поля введення.

Під блоком підписки розміщені посилання (рис. 3.15), схожі на ті, які знаходяться у верхньому навігаційному меню. Ці посилання надають швидкий доступ до важливих розділів сайту. Додатково, тут розташовані посилання на сторінки з інформацією про політику конфіденційності та правила та умови користування сайтом.

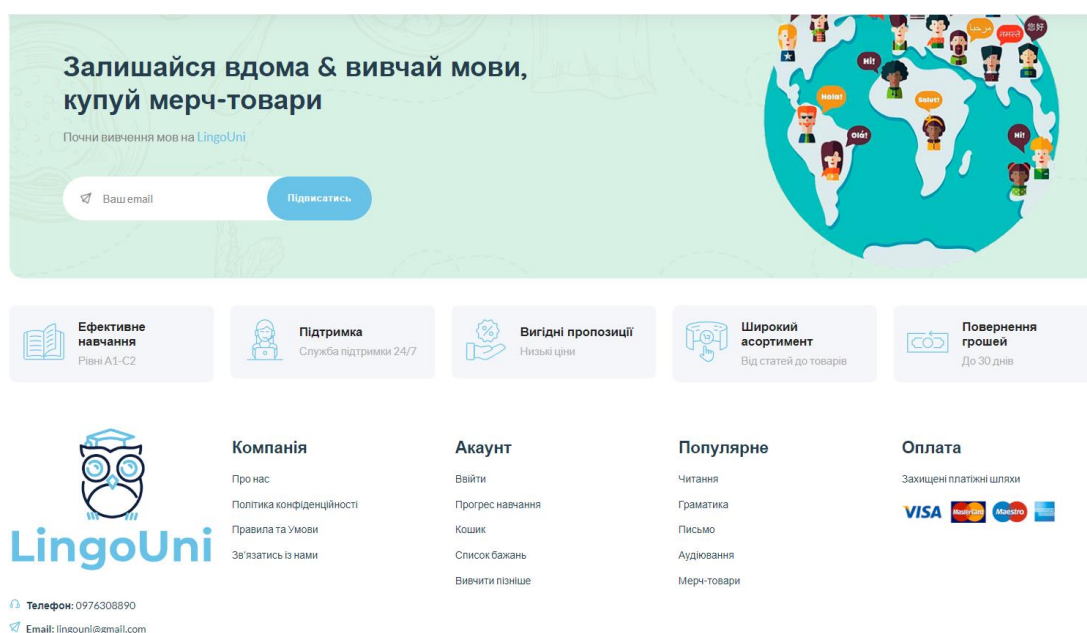


Рисунок 3.15 – Головна сторінка вебдодатку: нижня частина

Для того, щоб повною мірою використовувати функціонал вебдодатку для вивчення іноземних мов, користувачу необхідно увійти на сайт. Якщо у користувача немає облікового запису, він може зареєструватися.

Процес реєстрації (рис. 3.16) передбачає введення таких даних: ім'я, прізвище, електронна пошта, номер телефону, пароль, підтвердження паролю та перенесення цифр згенерованого випадковим чином коду безпеки із сусіднього блоку біля поля введення.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

Створити акаунт

Вже є акаунт? [Увійти](#)

ipzk211_moo

Олена

Мачушник

dev8only@gmail.com

+380976309002

.....

.....

6158

6158

☒ Я погоджуюсь із Правилами & Умовами.

Зареєструватись

Рисунок 3.16 – Процес реєстрації користувача

Після введення всіх необхідних даних користувач повинен поставити галочку, що він погоджується із правилами та умовами сайту. Тільки після цього стає доступною кнопка реєстрації, і користувач може завершити процес реєстрації за умови коректного введення всіх даних. Якщо користувач введе занадто короткий пароль або електронну адресу, яка вже зареєстрована, система виведе відповідне повідомлення про помилку. Користувач зможе завершити реєстрацію тільки після виправлення всіх помилок (рис. 3.17).

- Паролі не співпадають
- Пароль повинен містити цифри, великі і малі літери та не повинен бути коротше 6-ти символів
- Такий email вже використовується

Зареєструватись

Рисунок 3.17 – Помилки при реєстрації користувача

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

Після успішної реєстрації користувач може увійти на сайт, ввівши електронну пошту, пароль та згенерований випадковим чином код безпеки (рис. 3.18). Цей код генерується при кожному завантаженні сторінки, забезпечуючи додатковий рівень безпеки. Якщо користувач не бажає вводити свої дані щоразу при вході, він може поставити галочку біля опції «Запам'ятати». Це дозволить системі автоматично зберігати та підставляти електронну пошту і пароль у відповідні поля при наступних входах. Таким чином, користувачеві залишиться лише ввести код безпеки і натиснути кнопку входу, що значно спрощує процес авторизації та економить час.



Переглянути всі категорії

Вчити слова Головна Про нас Навчання Мерч-товари Блог Контакти

0976308890
24/7 Служба підтримки

Вхід

Немає акаунту? [Зареєструватись](#)

dev8only@gmail.com

.....

7210 7210

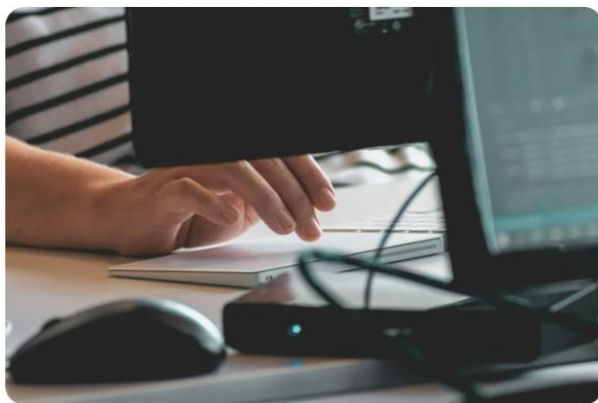
☒ Запам'ятати [Забули пароль?](#)

Ввійти

Рисунок 3.18 – Процес входу користувача

Іноді користувачі забувають свої паролі, і для таких випадків існує сторінка відновлення паролю (рис. 3.19). Дана можливість забезпечує зручний і безпечний спосіб відновлення доступу до акаунту із виключенням можливості злому сторінки сторонніми. Для того щоб відновити пароль, користувачу необхідно ввести електронну пошту, яка була використана при реєстрації, і натиснути на кнопку «Надіслати код». Після цього на вказану електронну адресу буде надіслано лист із унікальним кодом для підтвердження особи. Введення цього коду на сторінці відновлення дозволить користувачу встановити новий пароль, який забезпечить подальший доступ до акаунту.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		



Відновлення паролю

Згадали пароль? [Вхід](#)

dev8only@gmail.com

Надіслати код

Рисунок 3.19 – Процес відновлення паролю: введення пошти

Після натискання кнопки «Надіслати код», на вказану електронну пошту користувача надсилається лист (рис. 3.20) із випадково згенерованим кодом для відновлення доступу. Код призначений для одноразового використання і діє протягом обмеженого часу, що забезпечує додатковий рівень безпеки.

Цей код є дуже довгим і його неможливо вгадати, адже він призначений для підтвердження ідентичності користувача і гарантування безпеки процесу відновлення паролю. Код генерується випадковим чином, що робить його унікальним для кожного запиту. Після отримання коду користувач копіює його та вводить в спеціальне поле на вебсайті.

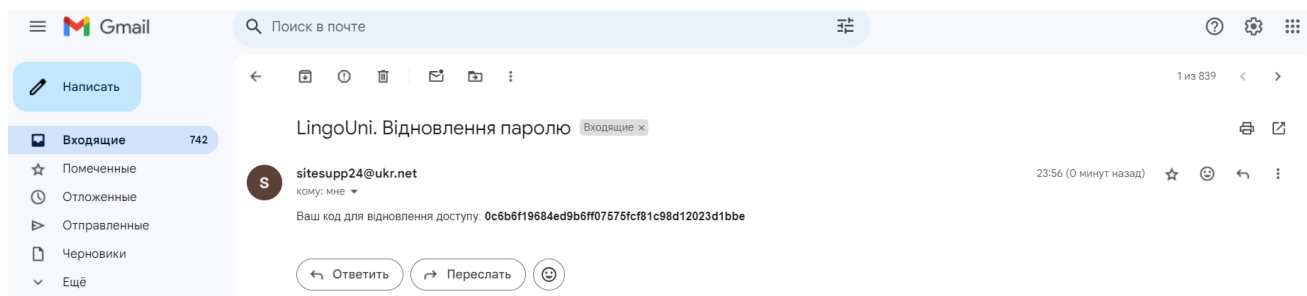
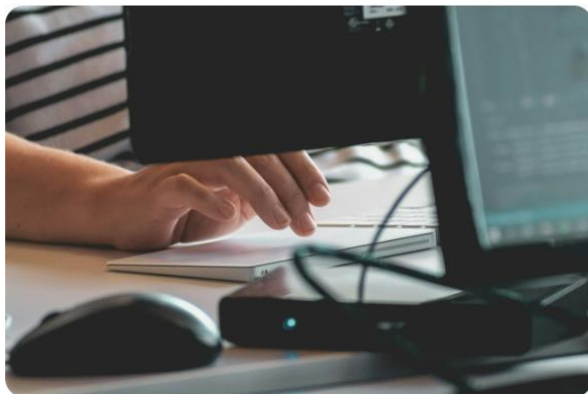


Рисунок 3.20 – Процес відновлення паролю: лист відновлення

Перевірка коду відновлення на сайті дозволяє завершити процедуру відновлення паролю і отримати знову доступ до свого акаунту. Користувачу буде запропоновано створити новий пароль для входу (рис. 3.21), який потім можна змінити в особистому кабінеті.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		



Введіть код відновлення з ПОШТИ

Доступ відновлено. Ваш новий пароль для входу: x2OVaYJt53

0c6b6f19684ed9b6ff07575fcf81c98d12023d1bbe

Перевірити

Рисунок 3.21 – Процес відновлення паролю: перевірка коду

Після успішної авторизації користувача в системі відбувається автоматичне перенаправлення на його особистий кабінет. У кабінеті користувача спочатку відображається категорія загальної інформації (рис. 3.22), яка надає огляд того, що саме можна робити та які функції доступні в цьому кабінеті.

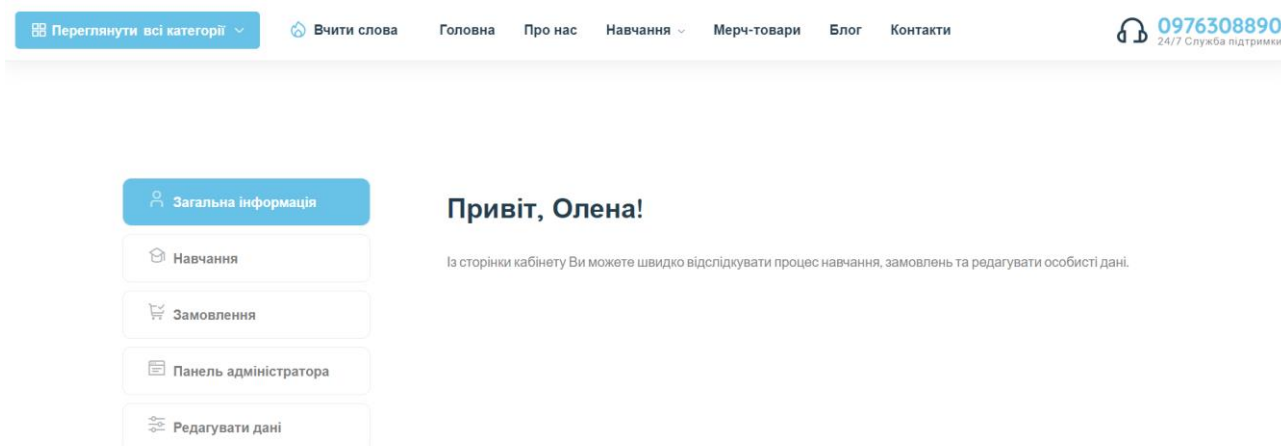


Рисунок 3.22 – Кабінет користувача: загальна інформація

У розділі «Навчання» в кабінеті користувача (рис. 3.23) відображається детальна інформація про його прогрес у вивченні іноземної мови. Користувач може переглянути кількість пройденого матеріалу у розділах читання, аудіювання, граматики та письма. Також доступні статистичні дані про проходження тестів за різними темами та середній відсоток успішності у цих тестах. Ця інформація дозволяє користувачам систематично відстежувати свій прогрес.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 3.23 – Кабінет користувача: навчання

У розділі «Замовлення» в кабінеті користувача (рис. 3.24) знаходиться повний перелік всіх замовлень, які він зробив на сайті. Для кожного замовлення вказані дата його оформлення та поточний статус. Адміністратор сайту встановлює статуси замовлень, які можуть бути «в обробці», «опрацьовано» або «завершено». Таке відображення замовлень дозволяє користувачам легко відстежувати стан своїх замовлень.

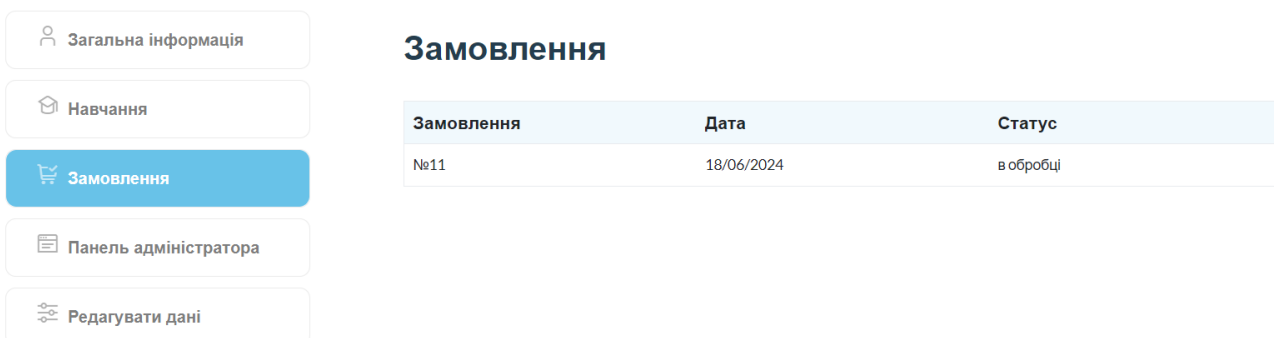


Рисунок 3.24 – Кабінет користувача: замовлення

Розділ «Панель адміністратора» в кабінеті (рис. 3.25) надає користувачам, які мають права адміністратора в системі, доступ до спеціального функціоналу для управління вебсайтом. Цей розділ відображається тільки для тих користувачів, які є працівниками даної компанії та мають права адміністратора.

Загальна інформація

Навчання

Замовлення

Панель адміністратора

Редагувати дані

Панель адміністратора

У панелі адміністратора можна редагувати, оновлювати, додавати, видаляти записи та керувати правами доступу користувачів.

[Перейти](#)

Рисунок 3.25 – Кабінет користувача: панель адміністратора

У особистому кабінеті користувача доступна можливість редагувати особисті дані (рис. 3.26), такі як ім'я, прізвище, телефон, електронну пошту і пароль. Це дозволяє користувачам оновлювати свою контактну інформацію і змінювати пароль для гарантування безпеки свого облікового запису.

Загальна інформація

Навчання

Замовлення

Панель адміністратора

Редагувати дані

Редагувати дані

Ім'я

Олена

Прізвище

Мачушник

Ім'я користувача

oleeppaa

Телефон

+380976309002

Email

dev8only@gmail.com

Новий пароль

Повторити новий пароль

Зберегти

Рисунок 3.26 – Кабінет користувача: редагувати дані

У вебдодатку доступна сторінка з правилами та умовами (рис. 3.27), де користувачі можуть ознайомитися з усіма важливими аспектами використання сервісу, а саме їх правами та обмеженнями.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 3.27 – Сторінка «Правила та умови»

У вебдодатку також існує сторінка «Політика конфіденційності» (рис. 3.28), де користувачі можуть ознайомитися з важливими аспектами збору, зберігання та використання їхніх особистих даних. На цій сторінці розглядаються правила обробки даних і права користувачів щодо захисту їхніх особистих даних.

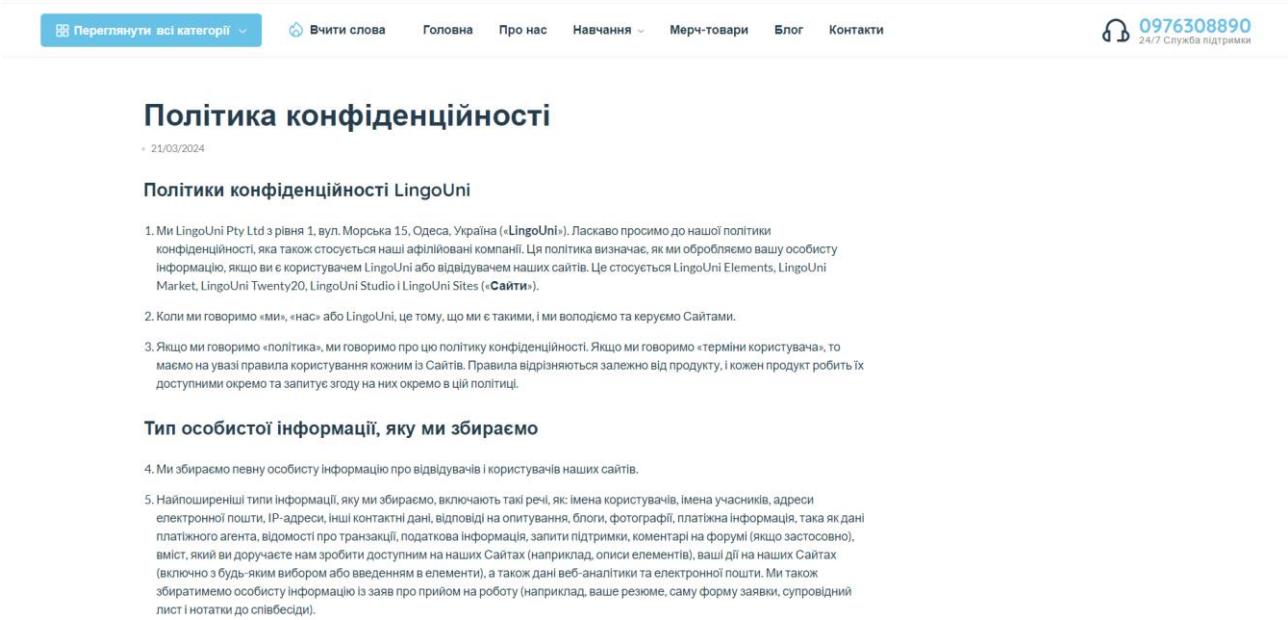


Рисунок 3.28 – Сторінка «Політика конфіденційності»

На сторінці «Про нас» (рис. 3.29) представлена інформація про лінгвістичну компанію. Користувачі можуть ознайомитися з короткою історією компанії, її цілями і цінностями. Крім того, на цій сторінці розміщується

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

інформація про ключових співробітників і команду компанії, що дозволяє краще зрозуміти професійний склад людей, які розробляють навчальні матеріали для вивчення іноземних мов.

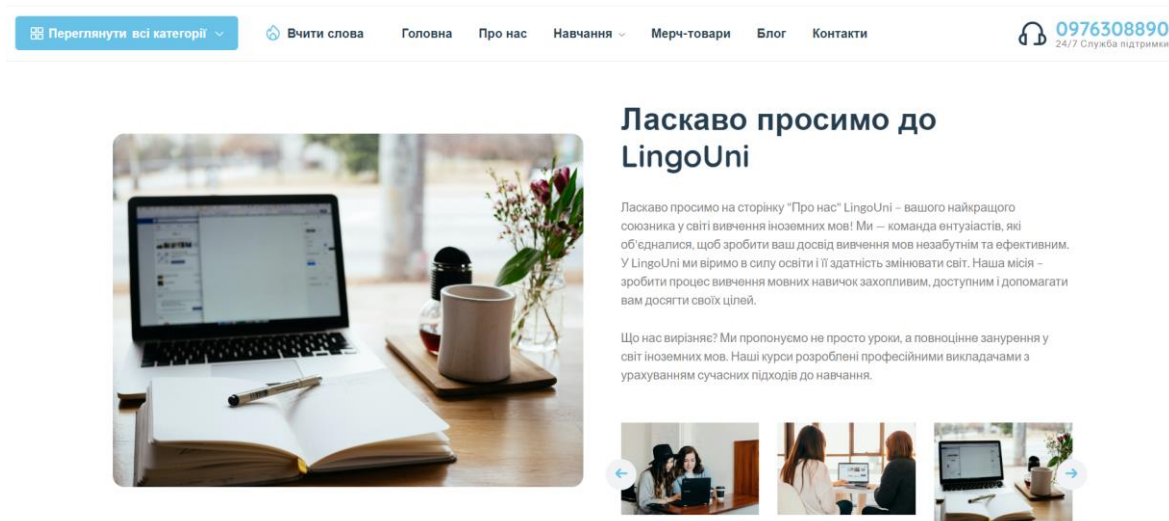


Рисунок 3.29 – Сторінка «Про нас»

На сторінці контактів (рис. 3.30) доступна інформація про розташування компанії, і її контактні дані. Також, на цій сторінці розміщена форма зворотного зв'язку, через яку користувачі можуть надсилати повідомлення представникам компанії. У формі необхідно вказати ім'я, прізвище, електронну пошту, телефон та тему повідомлення і саме повідомлення.

Форма зворотного зв'язку спрощує процес комунікації та забезпечує швидке вирішення питань або отримання необхідної інформації від компанії.

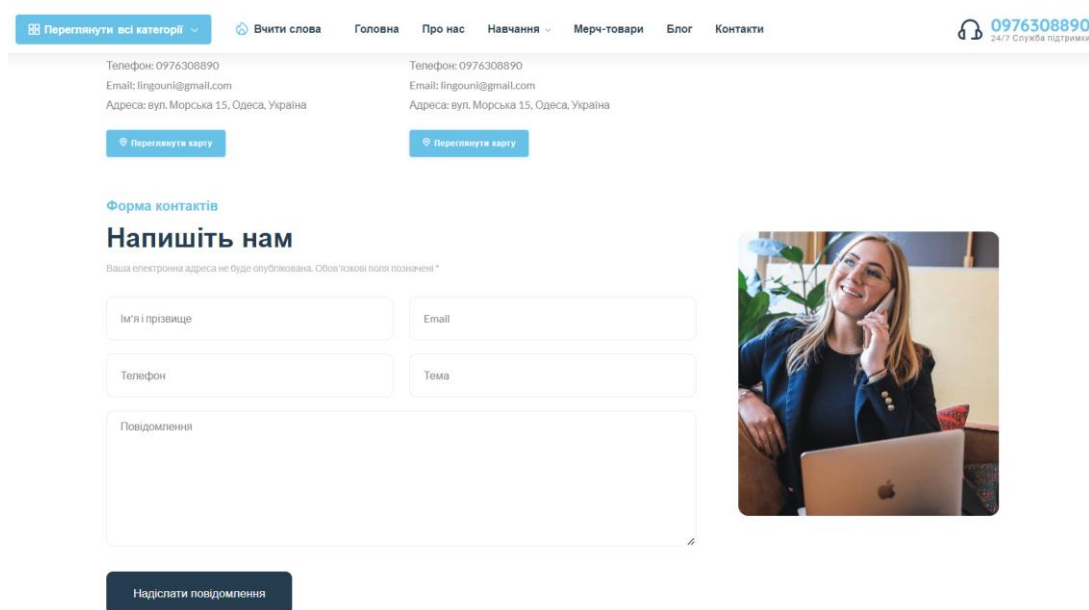


Рисунок 3.30 – Сторінка «Контакти»

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

На сайті реалізований функціонал блогу (рис. 3.31), на якому можна знайти цікаву інформацію про різні аспекти мовознавства, пов'язані з темами подорожей, оглядами навчальних матеріалів, рекомендаціями експертів та актуальними новинами з цієї сфери. Цей розділ призначений для того, щоб користувачі могли отримувати цікаву та корисну інформацію, яка допомагає їм в процесі навчання та підтримує мотивацію до вивчення іноземних мов.

Також авторизовані користувачі можуть додавати лайки до посту. Після того, як авторизований користувач додав лайк, картинка лайку відображається червоним кольором.

Авторизовані користувачі мають можливість додати пост до списку для прочитання пізніше, натиснувши на кнопку «Читати пізніше». Після натиснення цієї кнопки відобразатиметься повідомлення «Додано до читати пізніше», що підтверджує успішне додавання посту до списку.

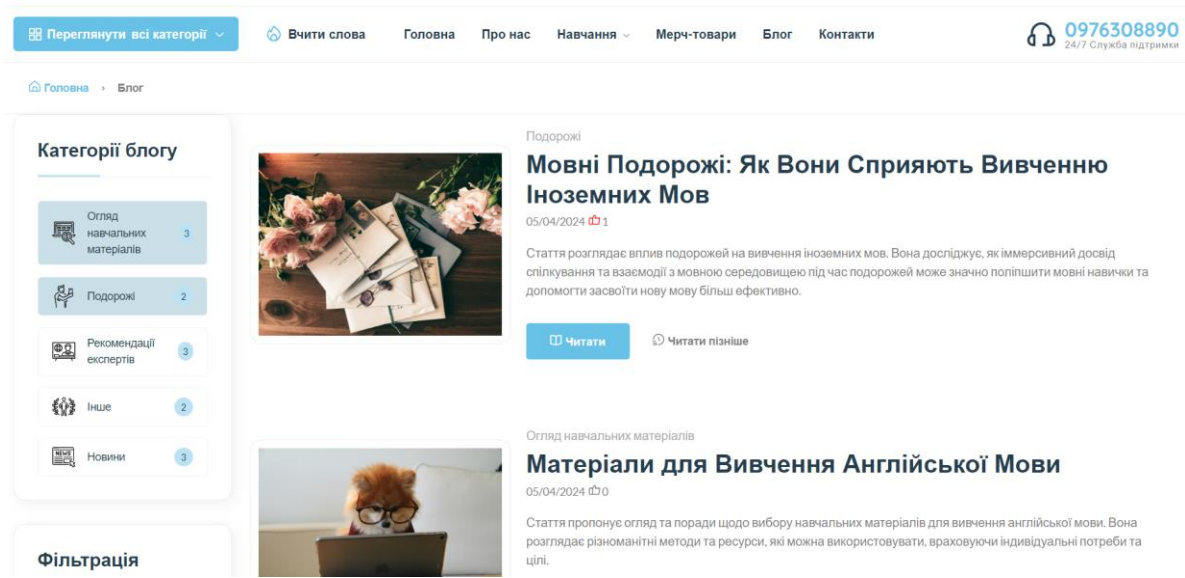


Рисунок 3.31 – Сторінка блогу

На сторінці блогу користувачі мають можливість зручно фільтрувати публікації за різними критеріями, такими як категорія блогу, дата, автор і новизна постів (рис. 3.32). Це дозволяє швидко знаходити потрібну інформацію. Функція фільтрації робить процес пошуку контенту більш ефективним.

Фільтрація

Від:

03.08.2017

До:

05.04.2024

Автор

☒ Олена Мачушник
☐ Єва Петрусенко
☐ Степан Лисенко

Інше

☒ Нові

Категорії блогу

Огляд навчальних матеріалів
3

Подорожі
2

Рекомендації експертів
3

Інше
2

Новини
3

Рисунок 3.32 – Фільтрація на сторінці блогу

Після натискання на назву посту користувач переходить на сторінку з обраним постом блогу (рис. 3.33). На цій сторінці відображається заголовок посту, фото та повний текст матеріалу. Такий формат дає можливість користувачам зручно прочитати вміст посту не відволікаючись на іншу інформацію.

Збоку біля вмісту посту розташований блок «Про пост», де представлена інформація про автора посту, дату його написання та кількість лайків. Авторизовані користувачі можуть додавати лайки до посту.

Переглянути всі категорії
Вчити слова
Головна
Про нас
Навчання
Мерч-товари
Блог
Контакти
0976308890
24/7 Служба підтримки

Револьюційні Рекорди: Найшвидше Вивчення Мов

Вивчення іноземних мов є складним і часом тривалим процесом, проте деякі особи досягають неймовірних результатів у короткий термін. У цій статті ми розглянемо найновіші досягнення у вивченні іноземних мов, зосереджуючись на найшвидшому рекорді вивчення мови та його секретах.

Історія Рекорду У світі вивчення іноземних мов багато людей змагаються за рекорд швидкого та ефективного освоєння нової мови. Один із найвражаючих рекордів належить Джонафану Грейвсу з Великобританії, який зміг опанувати іспанську мову вражаючим терміном у всього лише три місяці. Його досягнення було визнано як найшвидше вивчення іноземної мови за всю історію.

Секрети Швидкого Вивчення Мови 1. Іммерсія в Мовне Середовище: Головний секрет успіху Грейвса полягав у тому, що він створив іммерсивне мовне середовище, оточившись іспанською мовою на всі 100%. Він спілкувався лише іспанською, дивився іспанські фільми, слухав іспанську музику та підписався на іспанські новини.

Про пост

Єва Петрусенко

05/04/2024 10:04

1

Рисунок 3.33 – Сторінка посту блогу

Після вмісту посту розташований блок із коментарями (рис. 3.34), де користувачі можуть обговорювати матеріал. Авторизовані користувачі мають

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

можливість залишати свої коментарі та опубліковувати їх для загального перегляду іншими користувачами. Це забезпечує інтерактивність і залученість, дозволяючи користувачам залишати думки щодо прочитаного посту.

Залишіть коментар

Коментар

Надіслати

Коментарі



18/06/2024 00:38
Цікава стаття

Олена Мачушник

×

Рисунок 3.34 – Сторінка посту блогу: блок із коментарями

Якщо користувач натисне на ім'я та прізвище автора в блоці «Про пост», відбудеться перенаправлення на сторінку, де зібрані всі пости, написані цим автором (рис. 3.35). Це зручно для користувачів, яким подобається манера викладення інформації конкретного автора, оскільки вони можуть легко знайти і прочитати інші матеріали цього автора.

Автор: Степан Лисенко

Огляд навчальних матеріалів



Ресурси для вивчення іспанської мови
08/06/2020 1
Іспанська мова, з її багатою історією та широким поширенням, є однією з найпопулярніших мов у світі. Чи ви новачок, що тільки починає свій шлях до опанування іспанської мови, чи досвідчений учень, що шукає можливості для вдосконалення своїх навичок, у цій статті ми розглянемо найкращі навчальні ресурси та матеріали, які допоможуть вам досягти своїх цілей.

Читати Додано до читати пізніше

Рекомендації експертів



Майстерність Аудіювання
03/08/2017 2
У цій статті ми дослідимо експертні поради щодо сприйняття аудіоматеріалів на іноземній мові. Ви дізнаєтеся про стратегії та підходи, які допоможуть вам зрозуміти всі слова та суть аудіо, що слугуватиме важливим елементом вивчення мови.

Читати Читати пізніше

Рисунок 3.35 – Сторінка написаних постів вибраного автора

Вебдодаток також має інтернет-магазин, де пропонуються товари, вироблені цією лінгвістичною компанією. У магазині продаються різноманітні товари, такі як робочі зошити, словники, одяг, прикраси, сумки, рюкзаки та канцелярія, всі з фірмовим логотипом компанії (рис. 3.36).

Для ілюстрації товарів використовувалися фото, створені за допомогою логотипу компанії та макетів, що забезпечує професійний і привабливий вигляд кожного продукту в каталозі [26].

Авторизовані користувачі можуть додавати товари в кошик для подальшого придбання, а також до категорії улюблених. Це забезпечує зручність під час покупок, дозволяючи користувачам легко зберігати і переглядати ті товари, які вони планують купити або які їм сподобалися.

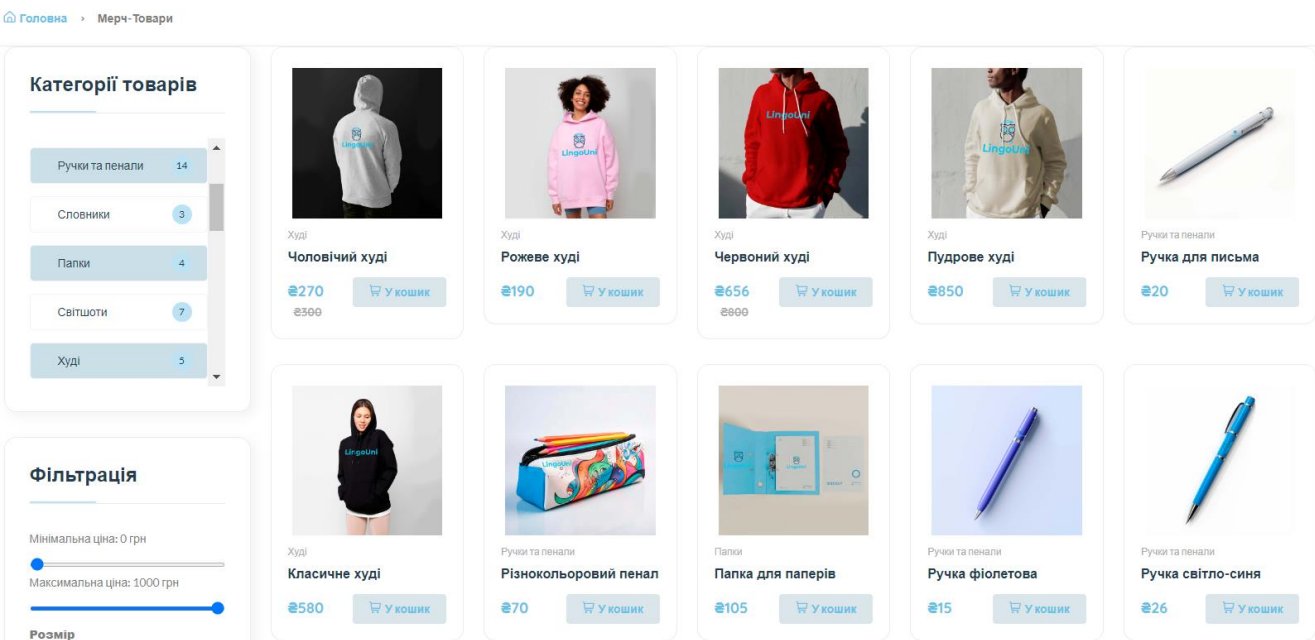


Рисунок 3.36 – Сторінка інтернет-магазину

Також на сторінці магазину користувачі мають можливість зручно фільтрувати товари за різними критеріями, такими як категорія товару, ціна, розмір і рекомендація товарів (рис. 3.37).

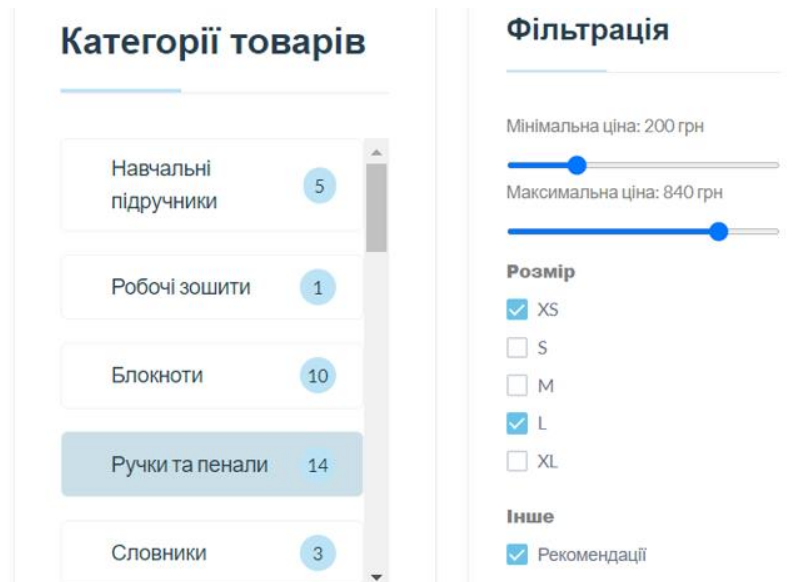


Рисунок 3.37 – Фільтрація на сторінці товарів

Якщо користувач бажає переглянути товари, які він додав до категорії улюблених, він може перейти за посиланням «Товари» у головному меню. Це відкриває сторінку із улюбленими товарами авторизованого користувача (рис. 3.38), що дозволяє швидко знаходити всі збережені позиції, забезпечуючи зручний доступ до улюблених товарів для їх подальшого перегляду або придбання.

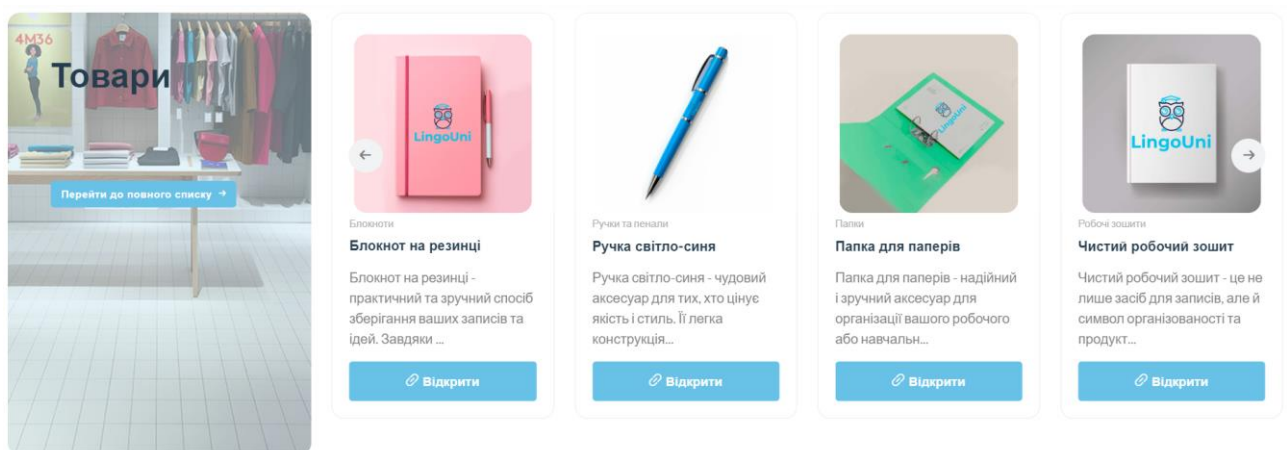


Рисунок 3.38 – Улюблені товари авторизованого користувача

Коли користувач переходить на сторінку товару (рис. 3.39), він має можливість переглянути всю детальну інформацію про цей товар, включаючи опис та характеристики. На цій сторінці користувач також може вибрати бажану кількість товару і додати його в кошик для подальшого придбання.

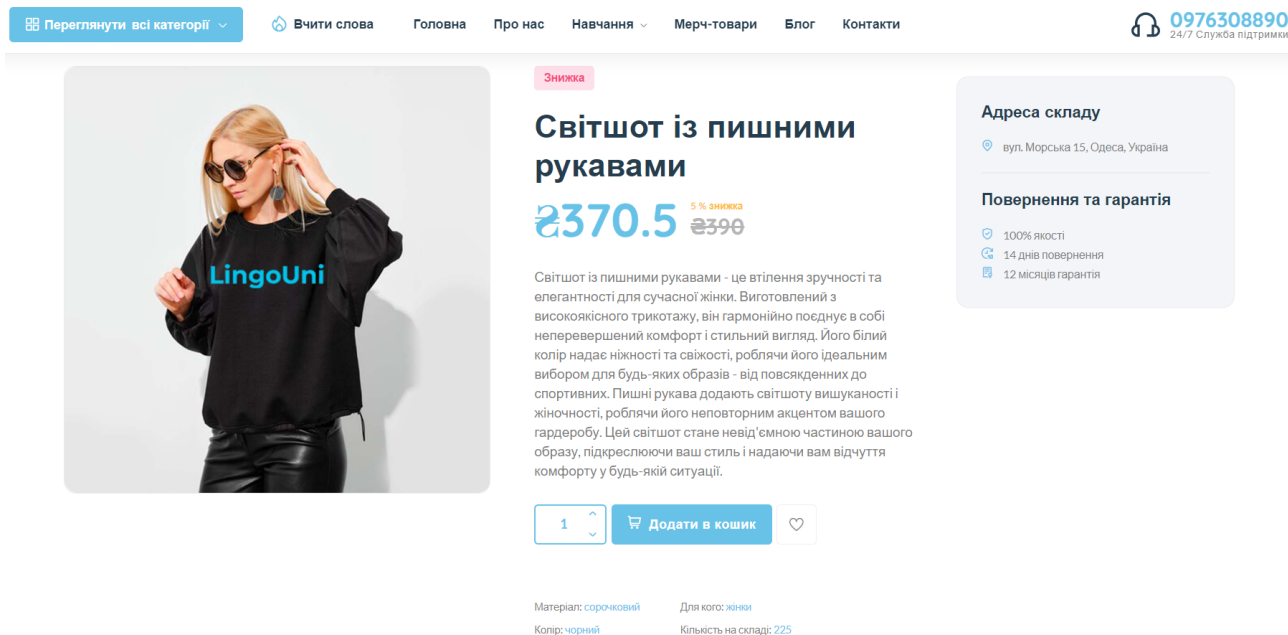


Рисунок 3.39 – Сторінка товару

Нижче на сторінці вибраного товару розташований блок із схожими продуктами, які також належать до тієї самої категорії, що і обраний товар (рис. 3.40). Цей блок дозволяє користувачам швидко знайти альтернативні варіанти або додаткові товари, які можуть їх зацікавити.

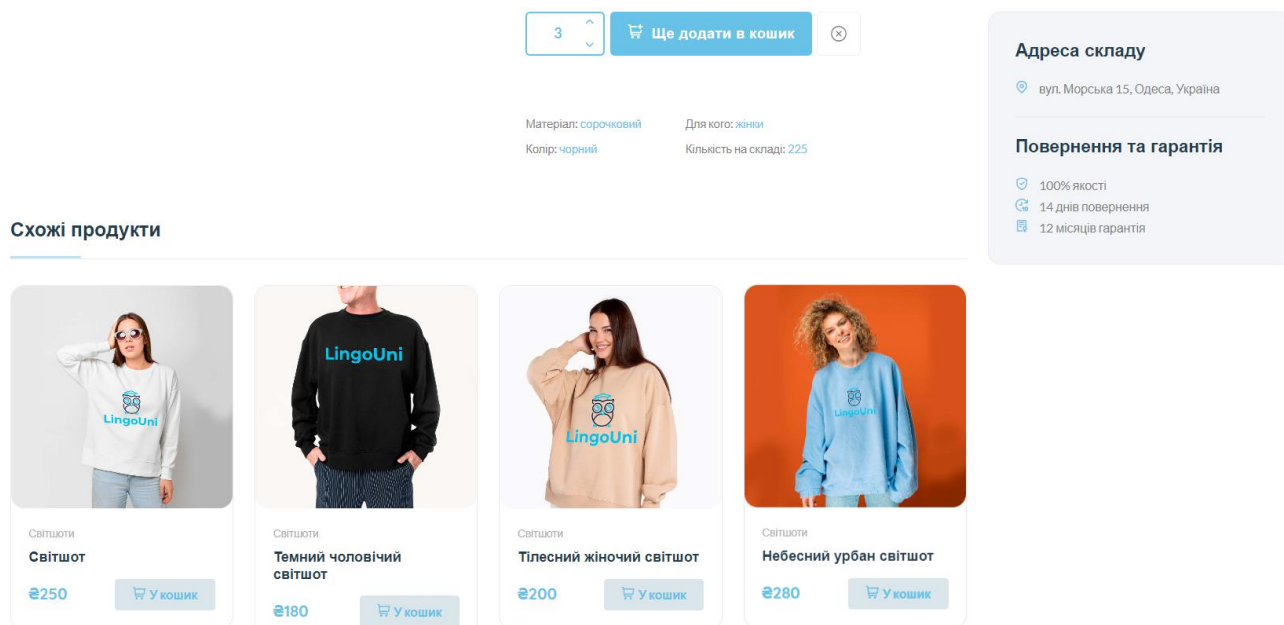


Рисунок 3.40 – Сторінка товару: схожі товари

Після того, як користувач обрав товари, він може перейти в кошик, де відображається детальна інформація про обрані товари (рис. 3.41). В кошику користувач може переглянути кількість кожного товару, його ціну, а також загальну суму замовлення. Тут є можливість змінювати кількість товару,

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

видаляти певні позиції або очистити кошик повністю. Також користувач може перейти до оформлення замовлення, щоб завершити процес покупки.

Ваш кошик

Ви маєте **наступні** товари у кошику:

Очистити кошик

Товар	Ціна	Кількість	Загалом	Видалити
 Світшот із пишними рукавами	€370.5	3	€ 1170	
 Чистий робочий зошит	€175	1	€ 175	

← Продовжити покупки

Загалом

€ 1345

Перейти до оформлення →

Рисунок 3.41 – Сторінка кошику

Для оформлення замовлення користувач має ввести свої дані або дані людини, яка буде отримувати замовлення (рис. 3.42). В повідомленні необхідно вказати місце доставки.

Форма замовлення

Оформлення замовлення

Після оформлення замовлення воно буде оброблене і надіслано.

Олена

Мачушник

dev@gmail.com

+380976309123

Нова пошта, м. Житомир, відділення №1

Надіслати повідомлення



Рисунок 3.42 – Сторінка оформлення замовлення

Вебдодаток для вивчення іноземних мов пропонує користувачам можливість розширювати свій словниковий запас та вивчати нові іноземні слова. У додатку доступний словник (рис. 3.43), який включає перелік слів відповідно до обраної мови і категорії слів. Користувачі можуть шукати необхідні слова, переглядати їх транскрипцію і переклад.

Словник			Оберіть мову: Англійська мова	Оберіть рівень: A1 (Початковий)	Оберіть категорію: Сім'я
Слово	Транскрипція	Переклад			
mother	['mʌðər]	мати			
father	['fɑːðər]	батько			
sister	['sɪstər]	сестра			
brother	['brʌðər]	брат			
family	['fæmɪli]	сім'я			
baby	['beɪbi]	немовля			
parents	['perənts]	батьки			
child	[tʃaɪld]	дитина			
children	['tʃɪldrən]	діти			
husband	['hʌzbənd]	чоловік			
wife	[waɪf]	дружина			

Рисунок 3.43 – Сторінка вивчення слів: словник

Після того як користувач ознайомився зі словником і вивчив необхідні слова, вебдодаток надає можливість закріпити отримані знання за допомогою тесту. Тест генерується випадковим чином і містить питання з обраної категорії слів, мови та її рівня (рис. 3.44). Користувач може перевірити свої знання, відповідаючи на питання та переглядаючи правильні відповіді після завершення кожного питання.

Вправи

Оберіть мову:
Англійська мова

Оберіть рівень:
C1 (Просунутий)

Оберіть категорію:
Хобі та захоплення

Який переклад слова archaeology?

ентомологія	☐
археологія	☒
ботаніка	☐

Далі

Рисунок 3.44 – Сторінка вивчення слів: вправи

Вебдодаток розроблений із доступом до різноманітних навчальних матеріалів з аудіювання, читання, граматики, письма та подкастів. Кожен з цих розділів містить інформацію, яка сприяє якісному покращенню знань іноземної

мови у користувачів. Крім теоретичного матеріалу, до кожного з них також включений закріплюючий тест, який дозволяє перевірити рівень розуміння вивченого матеріалу. Результати цих тестів зберігаються в системі і відображаються у вигляді навчальної статистики кожного користувача.

У вебдодатку навчальні матеріали організовані у вигляді переліку всіх постів (рис. 3.45), який дозволяє користувачам зручно ознайомлюватись з різноманітними темами. Кожен пост має можливість додавання лайків, а також користувачі можуть скористатись функцією «Опрацювати пізніше», щоб відкласти ознайомлення з постами на більш зручний час. Біля вже опрацьованих матеріалів відображається зелена галочка.

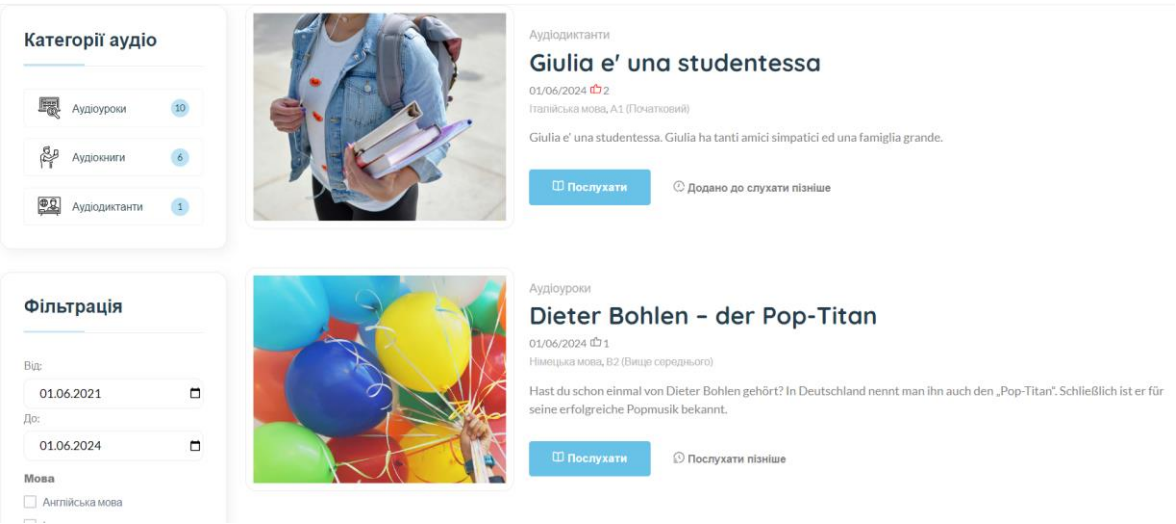


Рисунок 3.45 – Відображення навчальних матеріалів

Всі навчальні матеріали мають фільтрацію за датою, мовою, рівнем мови, опрацьованістю та новизною (рис. 3.46).

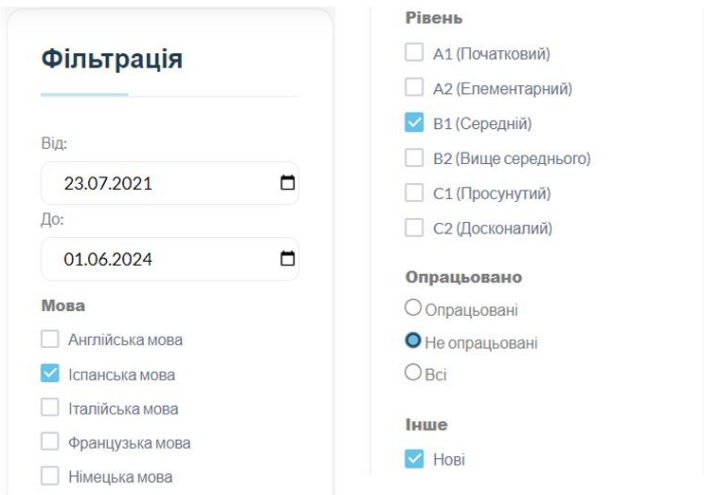


Рисунок 3.46 – Фільтрація для навчальних матеріалів

Кожен аудіоматеріал з категорії навчання містить можливість відтворення аудіо та відображення тексту аудіо (рис. 3.47). Після прослуховування аудіо користувач може пройти закріплюючий тест, який допомагає перевірити розуміння і усвідомлення вивченого матеріалу. Це сприяє ефективній аудіо підготовці, розвиває навички розуміння на слух і закріплює лексичний матеріал.

Аудіоуроки

Die Hornisgrinde



0:00 / 1:19

Показати текст аудіо

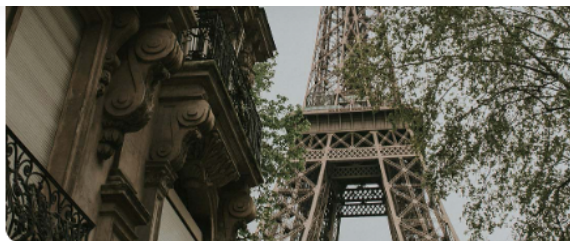
Закріплюючий тест

1. Wo liegt Baden-Württemberg?

Baden-Württemberg liegt im Nordwesten von Deutschland.	☐
Der Schwarzwald liegt im Süden von Deutschland.	☐

Рисунок 3.47 – Навчання: аудіювання

Навчальні матеріали для читання включають в себе текст читання (рис. 3.48), який сприяє розвитку навичок розуміння прочитаного. Крім тексту для читання, користувачі також мають можливість пройти закріплюючий тест, який допомагає перевірити розуміння тексту.



Ah, le mois de décembre : ses illuminations, la couronne de l'Avent pour les catholiques, et le fameux calendrier rempli de chocolat pour les enfants... Tous ces préparatifs rendent chaque année l'atmosphère festive et enjouée. Il n'y a qu'à observer la métamorphose des rues commerçantes pour en être convaincu. Finies les façades grises et ternes, terminée la morosité ambiante : les rues se parent de guirlandes scintillantes et les vitrines rivalisent d'imagination pour leur décoration. Les passants semblent moins pressés, et s'attardent plus facilement devant une devanture ou dans les rayons. Les dernières semaines avant Noël se déroulent dans une ambiance joyeuse et féérique. Le 24 décembre est le jour du réveillon. C'est l'occasion de fêter Noël en famille, en petit comité. Un sapin fraîchement coupé est installé dans la cuisine ou le salon, et décoré à l'aide de guirlandes colorées et de boules de Noël. Les cadeaux seront déposés à son pied, le soir du 24 ou le 25 décembre au matin. Le repas est servi sur une table resplendissante, recouverte de bougies et de petits santons. Le lendemain, le 25 décembre, est le jour de Noël. C'est un jour de fête où l'on invite la famille plus éloignée à partager le repas de midi. C'est souvent un repas fastueux et long : il n'est pas rare de sortir de table vers 16 ou 17 heures !

Закріплюючий тест

1. A quelle date fête t'on le réveillon de Noël en France?

le soir du 25 décembre	☐
le soir du 24 décembre	☐
le midi du 25 décembre	☐
le 23 décembre	☐

2. Trouver dans le texte un synonyme de devanture.

table	☐
réveillon	☐
vitrine	☐

Рисунок 3.48 – Навчання: читання

На сторінці для вивчення граматики та письма користувач зустрічає рамку з відображеним правилом граматики чи письма (рис. 3.49). Під цією рамкою розміщений опис правила, який детально пояснює його застосування та особливості. Після цього доступний закріплюючий тест, який дозволяє користувачеві перевірити своє розуміння та навички, пов'язані з вивченим матеріалом. Тест складається із запитань, які спрямовані на закріплення та перевірку засвоєної теорії.

Еcriture Créative en Français



Principales règles pour l'écriture créative : 1. Laissez libre cours à votre imagination et adoptez des idées non conventionnelles. 2. Utilisez un langage vivant et descriptif pour évoquer des émotions et des images. 3. Expérimentez avec différentes structures narratives et points de vue. 4. Montrez, ne racontez pas : impliquez le lecteur à travers des détails et des expériences sensorielles. 5. Révisez et modifiez votre travail pour affiner votre message et améliorer la clarté et la cohérence.

Pour écrire de manière créative, plongez dans vos pensées et émotions. Laissez votre plume danser sur la page, peignant des images avec des mots. Voici un exemple d'un court texte créatif : Le soleil se couchait sous l'horizon, peignant le ciel de teintes orangées et rosées. Les vagues murmuraient des secrets sur le rivage tandis que la brise marine dansait entre mes cheveux. Perdu dans l'instant, j'ai fermé les yeux et senti la chaleur du soleil disparaître sur ma peau. En cet instant fugace, j'ai trouvé du réconfort dans la beauté du monde qui m'entourait, un moment suspendu dans le temps.

Закріплюючий тест

1. Quest-ce qui distingue l'écriture créative des autres formes d'écriture?

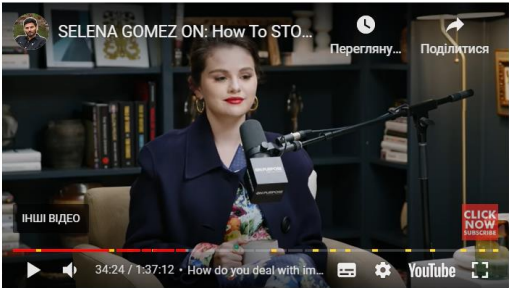
L'expression de l'imagination et de l'originalité	☐
---	-----------------------

Рисунок 3.49 – Навчання: письмо

На сторінці навчальних подкастів (рис. 3.50) користувач зустрічає відео з подкастом, яке можна відтворити безпосередньо на сторінці. Відео надає можливість одночасно слухати аудіо та переглядати візуальну інформацію, якщо вона доступна.

Із зірками

How to stop insecurity



Про пост

- Із зірками
- Англійська мова
- C2 (Досконалий)
- 02/06/2024 22:16
- 0

Рисунок 3.50 – Навчання: подкасти

Зверху на сторінці «Навчання» користувач може бачити індивідуальну статистику всіх пройдених матеріалів та тих, які він відклав для подальшого перегляду (рис. 3.51). Ця інформація дозволяє оцінити його прогрес у вивченні матеріалів.

Навчання

Головна > Навчання

Статистика всіх, пройдених та відкладених навчальних статей
персональна статистика користувача

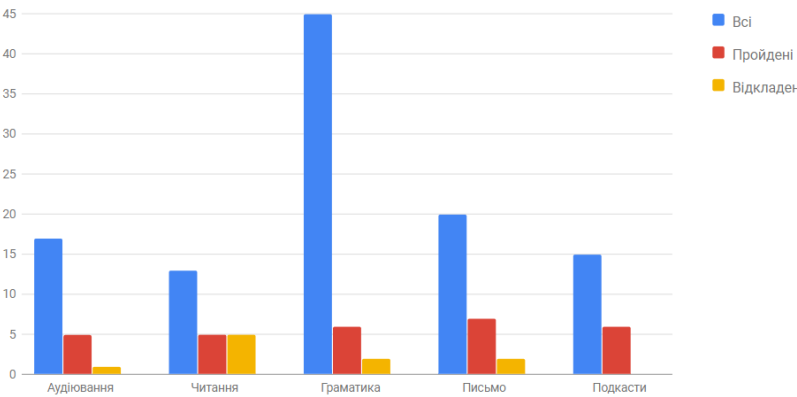


Рисунок 3.51 – Навчання: індивідуальна статистика користувача

Нижче на сторінці «Навчання» реалізовано відображення матеріалів з категорії «Вивчити пізніше» для кожного навчального матеріалу (рис. 3.52). Це дає користувачам зручну можливість відкладати для майбутнього вивчення ті

матеріали, які вони вважають цікавими або корисними, але поки не готові їх вивчати в даний момент.

Матеріали із категорії "Вивчити пізніше"

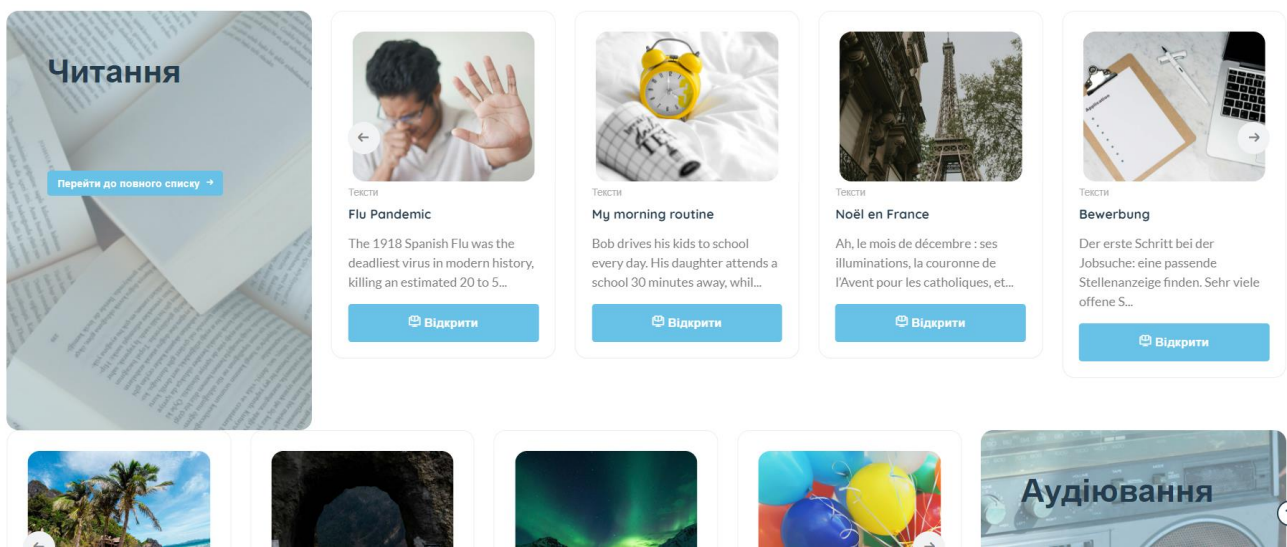


Рисунок 3.52 – Матеріали із категорії «Вивчити пізніше»

На вебсайті існує можливість здійснювати пошук по всім навчальним матеріалам, блогу та товарам (рис. 3.53). Користувачі можуть використовувати цей інструмент для швидкого доступу до потрібної інформації. Результати пошуку відображаються у вигляді списку посилань на знайдені ресурси, що дозволяє з легкістю знаходити важливу інформацію без зайвих зусиль.

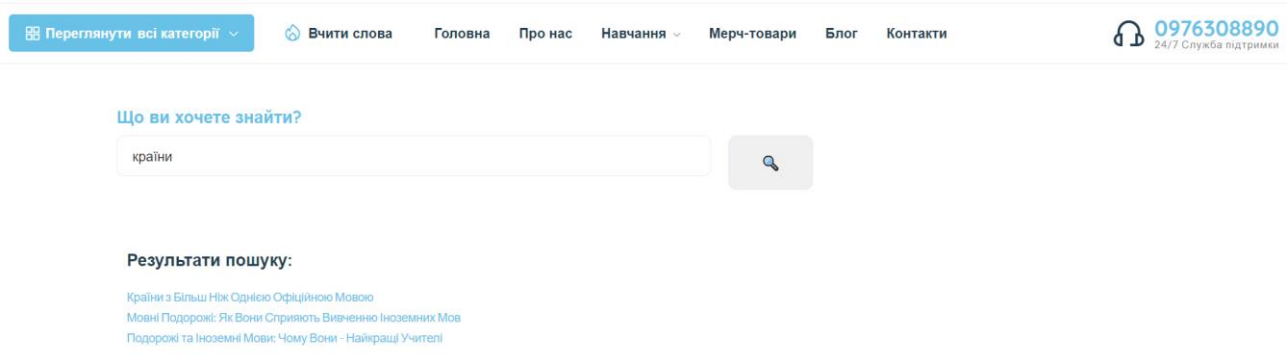


Рисунок 3.53 – Сторінка пошуку

У панелі адміністратора вебдодатка користувач з відповідними правами може переглядати різні таблиці даних і проводити з ними різноманітні операції, такі як створення, оновлення і видалення. Кожен співробітник має обмежений доступ лише до тих таблиць, які відповідають його функціональним обов'язкам. Наприклад, головний лінгвіст має доступ лише до таблиць, пов'язаних з

навчальними матеріалами, в той час як інші таблиці залишаються недоступними для його перегляду. Таким чином гарантується безпека даних і ефективне управління ресурсами в системі.

На головній сторінці адмінпанелі вебдодатка відображається загальна статистика щодо кількості реєстрацій користувачів за декілька останніх років (рис. 3.54). Ця інформація дозволяє адміністраторам отримати уявлення про динаміку зростання або спаду кількості нових користувачів на платформі.



Рисунок 3.54 – Панель адміністратора: головна сторінка

Після того, як адміністратор обрав конкретну таблицю в адмінпанелі вебдодатка, він отримує можливість переглянути всі записи, що містяться в цій таблиці (рис. 3.55). Це означає, що він може отримати повний список даних, які були збережені в базі даних системи для цієї конкретної таблиці.

Панель адміністратора

Управління таблицею читання (Створити новий)

№	Назва	Мова	Рівень	Опис	Дії
1	London	Англійська мова	B1 (Середній)	London, the capital of England and the United Kingdom, is renowned for its rich history and international allure. Rita and Joanne embarked on their first overseas journey to London, eager to explore its iconic landmarks...	Редагувати Видалити
2	My morning routine	Англійська мова	A2 (Елементарний)	Bob drives his kids to school every day. His daughter attends a school 30 minutes away, while his son's school is near Bob's job. His daughter is in the sixth grade and his son is in the second grade; both are good students. During the drive, his daughter sings, and his son sleeps.	Редагувати Видалити
3	The House	Англійська мова	A1 (Початковий)	The Smiths live in a house with a living room where they watch TV, a kitchen where the father cooks, and a dining room where they eat. The house has two bedrooms for sleeping and a closet for clothes. There's one bathroom where they brush their teeth. Outside, they have a garden where John and Sarah play with their dog.	Редагувати Видалити
4	Flu Pandemic	Англійська мова	C1 (Просунутий)	The 1918 Spanish Flu was the deadliest virus in modern history, killing an estimated 20 to 50 million people worldwide. It struck during World War I, causing more deaths among U.S. servicemen than the war itself. The pandemic initially began mildly in spring 1918 but became severe in fall 1918, likely exacerbated by wartime conditions. A third wave hit in winter 1919 and a smaller fourth wave in spring 1920.	Редагувати Видалити
5	Fahrenheit 451	Англійська мова	C2 (Досконалий)	Burning brought him immense pleasure, reveling in the destruction and transformation of objects. With determination, he ignited the house, engulfing it in a raging fire that painted the evening sky with fiery hues. Returning to the firehouse, he reflected on his actions with a grim satisfaction, knowing the fiery smile would linger in his memory.	Редагувати Видалити
6	1984	Англійська мова	C2 (Досконалий)	The passage illustrates the inevitability of punishment for thoughtcrime, emphasizing the totalitarian control exerted by the Thought Police. The protagonist grapples with the concept of crimethink, a mental barrier against dangerous thoughts enforced by the Party. Orwell portrays the manipulation of reality through doublethink, where individuals must accept contradictory beliefs without question.	Редагувати Видалити

Рисунок 3.55 – Відображення записів вибраної таблиці

Адміністратор має можливість створювати нові записи у вибраній таблиці (рис. 3.56). Для цього він вводить необхідні дані в поля, які визначені для кожного типу запису. Ця функціональність дозволяє адміністратору додавати нові дані безпосередньо до бази даних системи.

The screenshot shows the LingoUni admin interface. On the left is a sidebar with navigation links: Головна, Мерч-магазин, Блог, Словник, Читання, Аудіювання, Граматика, Письмо, Подкасти, and Загальне. The main content area is titled 'Панель адміністратора' and 'Управління таблицею товарів: створення запису'. The form includes fields for: Назва (Name), Підкатегорія (Subcategory) with a dropdown menu showing 'Навчальні підручники', Ціна (Price), Зніжок (Discount), На складі (In stock), and Опис (Description) with a text area labeled 'Type here'.

Рисунок 3.56 – Створення нового запису для вибраної таблиці

У адмінпанелі вебдодатка крім можливості створення нових записів, адміністратор може здійснювати оновлення і видалення вже існуючих записів. Оновлення дозволяє адміністратору змінювати вміст вибраного запису, виправляти помилки чи оновлювати інформацію, що стала застарілою (рис. 3.57).

Видалення дозволяє адміністратору видаляти записи, які більше не потрібні або були створені помилково, забезпечуючи таким чином підтримку чистоти даних і ефективне керування інформацією в системі.

The screenshot shows the LingoUni admin interface for updating a record. The sidebar is the same as in Figure 3.56. The main content area is titled 'Панель адміністратора' and 'Управління таблицею товарів: оновлення запису'. The form includes fields for: Назва (Name) with the value 'Світшот', Підкатегорія (Subcategory) with a dropdown menu showing 'Світшоти', Ціна (Price) with the value '250', Зніжок (Discount) with the value '0', На складі (In stock) with the value '223', and Опис (Description) with the text: 'Цей білий світшот створений для жінок із зручного трикотажного матеріалу. Він поєднує в собі комфорт і стиль, ідеальний вибір для повсякденного використання або спортивних заходів.'

Рисунок 3.57 – Оновлення наявного запису вибраної таблиці

3.4. Тестування роботи програмного продукту

Функціональне тестування допомагає перевірити систему на відповідність функціональним вимогам, воно гарантує, що програма належним чином задовольняє вимоги. Мета функціонального тестування полягає у перевірці кожної функції програми із очікуваним результатом, це включає тестування вхідних і вихідних даних програмного забезпечення, маніпулювання даними, взаємодію з користувачем і реакцію системи на різні сценарії та умови [30].

У першу чергу необхідно протестувати користувацьку сторону вебдодатку, а саме: працездатність усіх посилань, кнопок, слайдерів, полів для введення даних, можливості входу, реєстрації, проходження навчальних матеріалів і закріплюючих тестів, додавання лайків і «Вивчити пізніше» до постів, додавання товарів в кошик і в улюблені, перегляд навчальної статистики.

Таблиця 3.1

Тестування користувацької сторони вебдодатку

№	Опис тесту	Вхідні дані	Очікуваний результат	Фактичний результат	Статус
1	Перевірка посилань головного меню	Натискання на посилання в меню	Перехід на вказану сторінку вебсайту	Перехід на вказану сторінку вебсайту	Пройдено
2	Перевірка переходу за посиланнями в футері	Натискання на посилання у футері	Перехід на вказану сторінку вебсайту	Перехід на вказану сторінку вебсайту	Пройдено
3	Перевірка введення пошти для підписки	Ввести пошту у поле введення і натиснути «Підписатись»	Виведення тексту «Ви успішно підписались»	Виведення тексту «Ви успішно підписались»	Пройдено
4	Перевірка форми зворотного зв'язку	Ввести ім'я, прізвище, пошту,	Сторінка оновлюється, дані форми очищаються	Сторінка оновлюється, дані форми очищаються	Пройдено

		телефон, тему і повідомлення			
5	Перевірка відображення товарів інтернет-магазину і фільтрація	Задати довільний фільтр товарів	Відображаються товари відповідно до обраного фільтру	Відображаються товари відповідно до обраного фільтру	Пройдено
6	Перевірка додавання товару до улюблених	Натиснути на сердечко біля обраного товару	Товар додається на сторінку «Товари»	Товар додається на сторінку «Товари»	Пройдено
7	Перевірка додавання товару до кошику	Натиснути на «У кошик» біля обраного товару	Товар додається на сторінку кошику	Товар додається на сторінку кошику	Пройдено
8	Перевірка відображення окремого товару	Обрати товар і натиснути на нього	Відображається вся інформація про товар на окремій сторінці	Відображається вся інформація про товар на окремій сторінці	Пройдено
9	Перевірка оформлення замовлення	Оформити замовлення в кошику	Замовлення відображається у кабінеті користувача	Замовлення відображається у кабінеті користувача	Пройдено
10	Перевірка відображення постів блогу і фільтрація	Задати довільний фільтр постів блогу	Відображаються пости відповідно до обраного фільтру	Відображаються пости відповідно до обраного фільтру	Пройдено
11	Перевірка додавання посту блогу до «Читати пізніше»	Натиснути на кнопку «Читати пізніше» біля обраного посту	Кнопка змінила текст на «Додано до читати пізніше»	Кнопка змінила текст на «Додано до читати пізніше»	Пройдено

13	Додавання і прибирання лайку до поста блогу	Натиснути на картинку лайку послідовно два рази	При кожному кліку змінюється колір картинки і число біля неї	При кожному кліку змінюється колір картинки і число біля неї	Пройдено
14	Перевірка відображення окремого поста блогу	Обрати пост блогу і натиснути на нього	Відображається вся інформація поста на окремій сторінці	Відображається вся інформація поста на окремій сторінці	Пройдено
15	Перевірка відображення словника	Задати фільтр	Таблиця зі словами оновлюється залежно від заданого фільтру	Таблиця зі словами оновлюється залежно від заданого фільтру	Пройдено
16	Перевірка тестів до словника	Відповісти на питання тесту	Фон відповіді змінює колір на зелений або червоний	Фон відповіді змінює колір на зелений або червоний	Пройдено
17	Перевірка відображення навчальних матеріалів і фільтрації	Задати довільний фільтр навчального матеріалу	Відображаються навчальні матеріали відповідно до обраного фільтру	Відображаються навчальні матеріали відповідно до обраного фільтру	Пройдено
18	Перевірка додавання навчального матеріалу до «Вивчити пізніше»	Натиснути на кнопку «Вивчити пізніше» біля обраного навчального матеріалу	Кнопка змінила текст на «Додано до вивчити пізніше»	Кнопка змінила текст на «Додано до вивчити пізніше»	Пройдено
19	Додавання і прибирання лайку	Натиснути на картинку лайку	При кожному кліку	При кожному кліку	Пройдено

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						75
Змн.	Арк.	№ докум.	Підпис	Дата		

	до навчального матеріалу	послідовно два рази	змінюється колір картинки і число біля неї	змінюється колір картинки і число біля неї	
21	Перевірка відображення окремого навчального матеріалу	Обрати навчальний матеріал і натиснути на нього	Відображається вся інформація навчального матеріалу на окремій сторінці	Відображається вся інформація навчального матеріалу на окремій сторінці	Пройдено
22	Перевірка тестів до навчальних матеріалів	Відповісти на питання тесту і натиснути «Перевірити»	Фон відповіді змінює колір на зелений або червоний	Фон відповіді змінює колір на зелений або червоний	Пройдено

Також необхідно протестувати панель адміністратора вебдодатку, а саме: працездатність відображення всіх доступних таблиць та записів наявних у таблицях, створення нових записів, оновлення та видалення наявних записів.

Таблиця 3.2

Тестування панелі адміністратора вебдодатку

№	Опис тесту	Вхідні дані	Очікуваний результат	Фактичний результат	Статус
1	Перевірка відображення всіх доступних таблиць	Відкрити ліве бокове меню	Відображаються всі доступні таблиці	Відображаються всі доступні таблиці	Пройдено
2	Перевірка відображення всіх записів відповідно до таблиць	Обрати необхідну таблицю	Відображаються записи обраної таблиці	Відображаються записи обраної таблиці	Пройдено
3	Створення нових записів відповідно до таблиць	Введення даних про новий запис у форму введення	Новий запис додався до відображення всіх записів таблиці	Новий запис додався до відображення всіх записів таблиці	Пройдено

4	Оновлення наявних записів відповідно до таблиць	Редагування даних про наявний запис у форму введення	Наявний запис оновився у відображенні всіх записів таблиці	Наявний запис оновився у відображенні всіх записів таблиці	Пройдено
5	Видалення наявних записів відповідно до таблиць	Натиснути кнопку «Видалити»	Запис не відображається у відображенні всіх записів таблиці	Запис не відображається у відображенні всіх записів таблиці	Пройдено
6	Перевірка відображення статистики реєстрації користувачів	Перейти на головну сторінку адмінпанелі	Статистика коректно показує дані	Статистика коректно показує дані	Пройдено

Під час проведення тестування було уважно перевірено вебдодаток на наявність можливих помилок і недоліків. Завдяки цьому процесу було досягнуто високу якість вебсайту.

Висновки до третього розділу

У третьому розділі кваліфікаційної роботи було детально проаналізовано та описано інтерфейс розробленого вебдодатку, визначено основні елементи інтерфейсу та їх функціональні можливості. Проведено аналіз послідовності використання програмного додатку, описано роботу з головної сторінки до адмінпанелі, враховуючи всі етапи взаємодії користувача з додатком.

Також було проведено тестування розробленого програмного забезпечення з метою перевірки його працездатності та відповідності функціональним вимогам. В рамках тестування була уважно перевірена робота як панелі адміністратора, так і користувацької частини вебдодатку з урахуванням різноманітних сценаріїв використання. Отримані результати підтвердили високу якість розробленого програмного забезпечення і його здатність ефективно функціонувати у визначених умовах.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						77
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було розроблено вебдодаток для вивчення іноземних мов, спрямований на забезпечення користувачам ефективного та зручного інструменту для вивчення лінгвістичного матеріалу. Основні цілі та завдання проекту полягали у створенні вебдодатку, який міг би поєднувати в собі сучасні технології та найкращі практики розробки програмного забезпечення. Аналіз існуючих рішень на ринку надав змогу обрати оптимальні технології, такі як HTML, CSS, JavaScript, JQuery та PHP, щоб забезпечити високий рівень функціональності та ефективність роботи вебдодатку.

Детальний аналіз вимог до програмного продукту був здійснений з метою точного визначення функціональних і нефункціональних вимог, які повинні були бути задоволені для досягнення успішного впровадження програмного продукту. Була створена об'єктно-орієнтована модель системи, яка чітко відображує взаємодію різних компонентів програмного забезпечення.

Особлива увага була приділена розробці алгоритмів та оптимізації ключових аспектів функціонування системи, що дозволило досягти високої швидкодії та стабільності роботи додатку. У третьому розділі роботи детально був описаний інтерфейс вебдодатку, визначені основні елементи і їх функціональні можливості, що дозволило користувачам зручно та ефективно взаємодіяти з додатком.

Проведене функціональне тестування підтвердило високу якість програмного забезпечення і його відповідність вимогам. Результати тестування свідчать про ефективність розробленого вебдодатку, його здатність до ефективного функціонування.

Отже, у результаті виконання кваліфікаційної роботи був реалізований вебдодаток для вивчення іноземних мов, що відповідає всім встановленим вимогам та демонструє високий рівень функціональності.

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						78
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

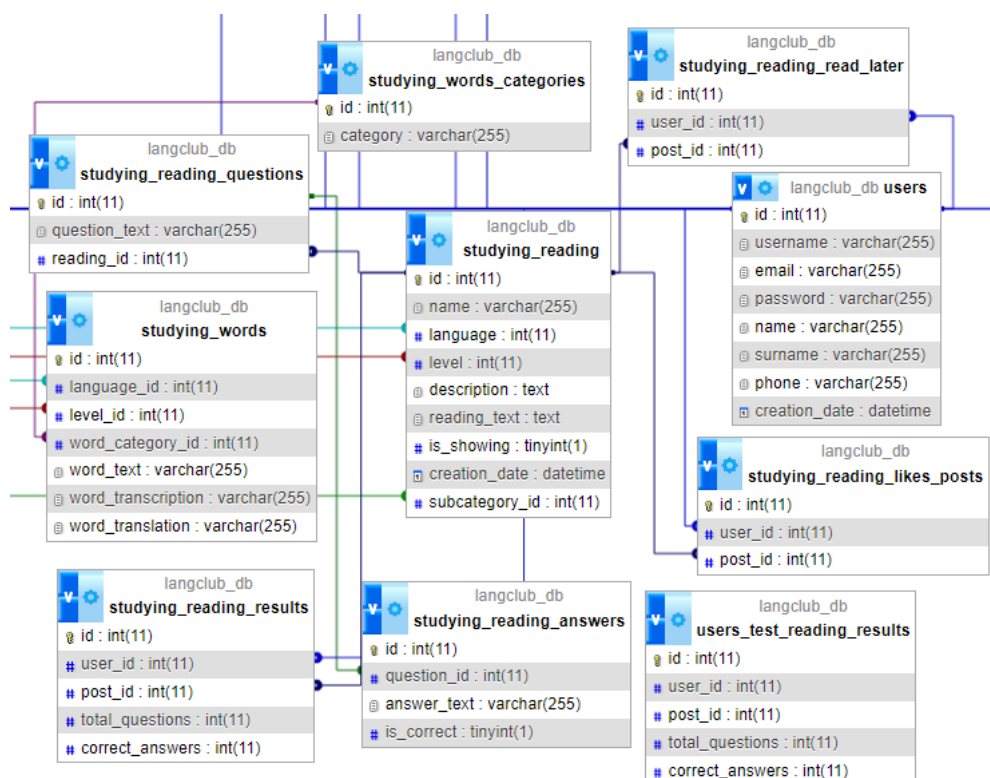
1. HTML and CSS: The Comprehensive Guide, 1st edition/Jurgen Wolf, 2023–814p.
2. Responsive Web Design with HTML5 and CSS, 4th edition/Ben Frain, 2022–498p.
3. Web Coding & Development All-in-one, 2nd edition / Paul McFedries, 2024–848p.
4. Javascript: The Definitive Guide, 7th edition / David Flanagan, 2020 – 704 p.
5. PHP Objects, Patterns, and Practice, 5th edition / Matt Zandstra, 2019 - 576 p.
6. PHP: The Right Way / Josh Lockhart, 2020 - 150 p.
7. PHP&MySQL:Server-side Web Development, 1st Edition/Jon Duckett,2022-672p.
8. PHP 8: The Modern Web Developer's Guide / Kameron Hussain, 2024 – 235 p.
9. W3Schools JavaScript [Електронний ресурс] – Режим доступу до ресурсу:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript>
10. jQuery Learning Center [Електронний ресурс] – Режим доступу до ресурсу:
<https://learn.jquery.com/>
11. PHP Documentation [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.php.net/manual/en/>
12. MySQL Documentation [Електронний ресурс] – Режим доступу до ресурсу:
<https://dev.mysql.com/doc/>
13. The Importance of Language Learning in a Globalized World [Електронний ресурс] – Режим доступу до ресурсу: <https://euro-lingo.com/blog/the-importance-of-language-learning-in-a-globalized-world/>
14. Web application [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.techtarget.com/searchsoftwarequality/definition/Web-application-Web-app>
15. Duolingo [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.duolingo.com/info>
16. Електронний HTML і CSS довідник [Електронний ресурс] – Режим доступу до ресурсу: <https://html-css.co.ua/>
17. Difference Between JavaScript and jQuery [Електронний ресурс] – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/difference-between-javascript-and-jquery/>

					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						79
Змн.	Арк.	№ докум.	Підпис	Дата		

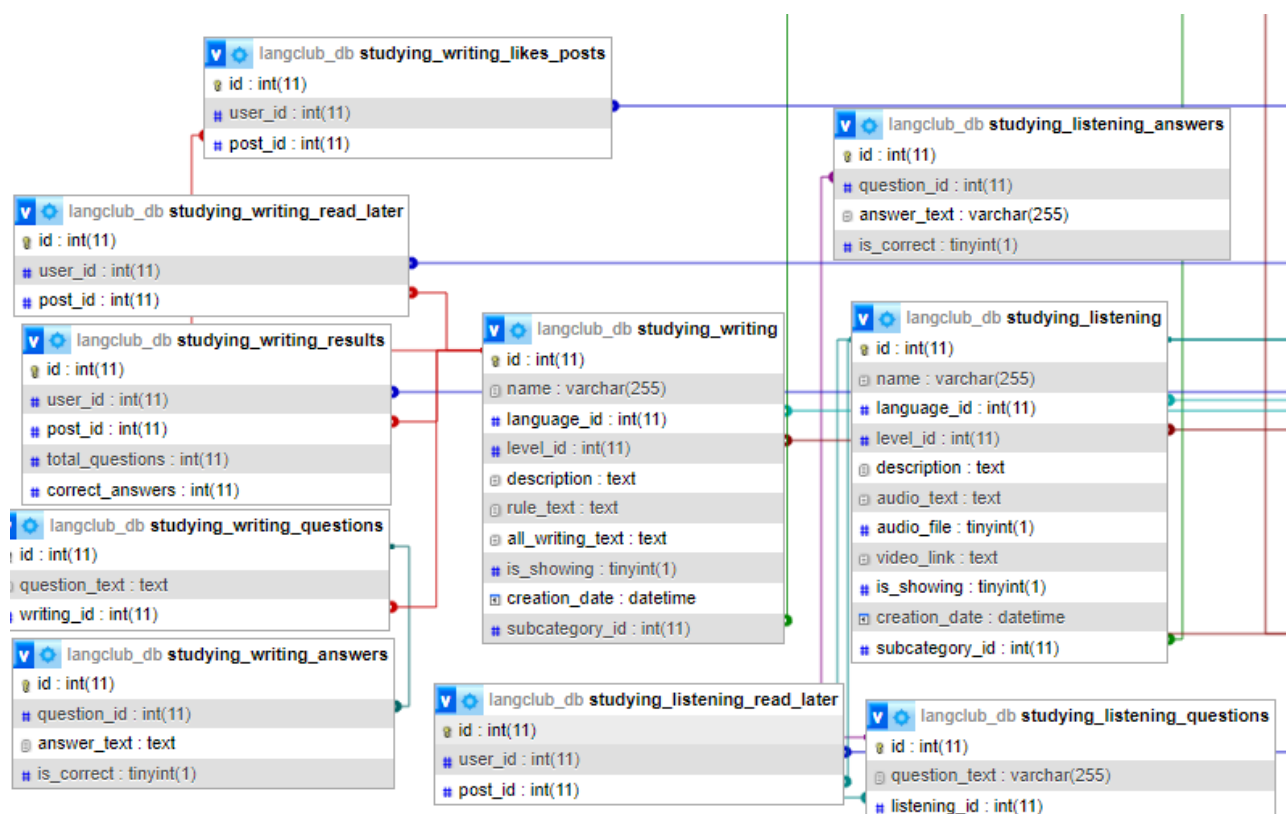
18. Use Case Diagrams [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.geeksforgeeks.org/use-case-diagram/>
19. Діаграма діяльності в UML [Електронний ресурс] – Режим доступу до ресурсу: <https://www.guru99.com/uk/uml-activity-diagram.html>
20. Class Diagram [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.geeksforgeeks.org/unified-modeling-language-uml-class-diagrams/>
21. Діаграма послідовності (Sequence Diagrams) [Електронний ресурс] – Режим доступу до ресурсу: <https://www.maxzosim.com/sequence-diagrams/>
22. What is UML Collaboration Diagram [Електронний ресурс] – Режим доступу до ресурсу: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml-collaboration-diagram/>
23. PHP MVC: the easy way [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.cloudways.com/blog/send-emails-in-php-using-phpmailer/>
24. MVC Design Pattern [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.geeksforgeeks.org/mvc-design-pattern/>
25. How to Send Emails in PHP Using PHPMailer Library [Електронний ресурс] – Режим доступу до ресурсу: <https://github.com/PHPMailer/PHPMailer>
26. Unblast mockups [Електронний ресурс] – Режим доступу до ресурсу:
<https://unblast.com/mockups/>
27. Multipurpose eCommerce HTML Template [Електронний ресурс] – Режим доступу до ресурсу: <https://themeforest.net/item/nest-multipurpose-ecommerce-html-template/33948410>
28. Unsplash [Електронний ресурс] – Режим доступу до ресурсу:
<https://unsplash.com/>
29. Flaticon [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.flaticon.com/>
30. Functional Testing – Software Testing [Електронний ресурс] – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/software-testing-functional-testing/>

ДОДАТКИ

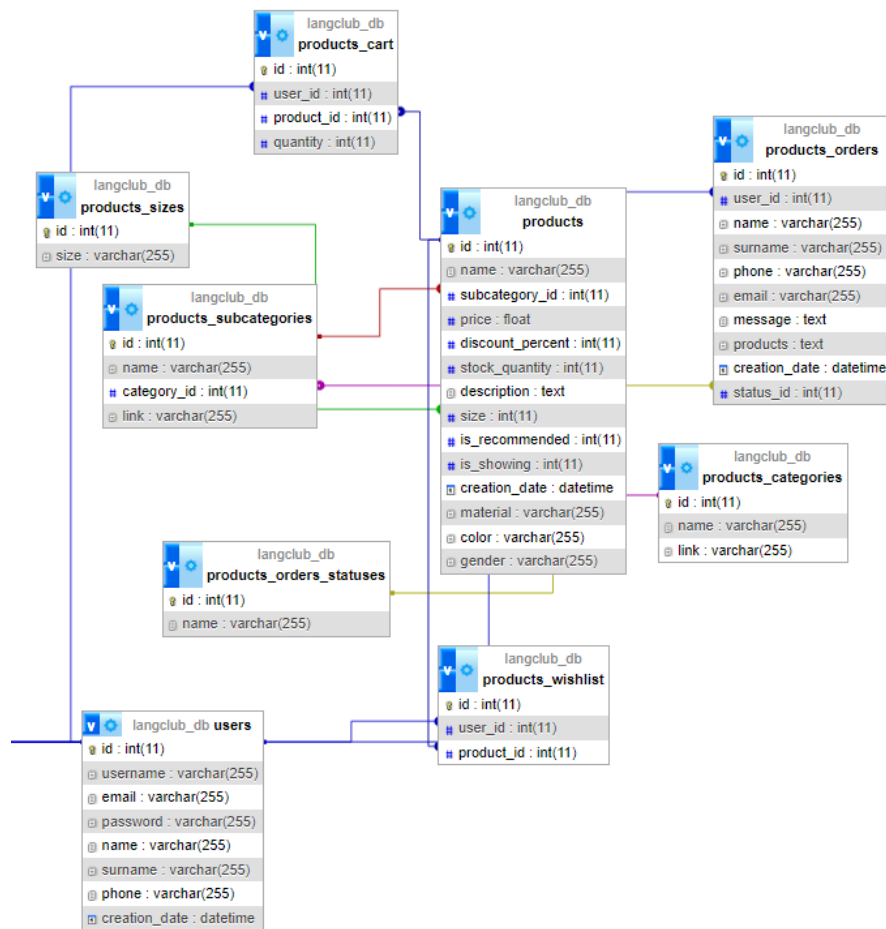
					ІПЗ.КР.Б – 121 – 24 – ПЗ	Арк.
						81
Змн.	Арк.	№ докум.	Підпис	Дата		



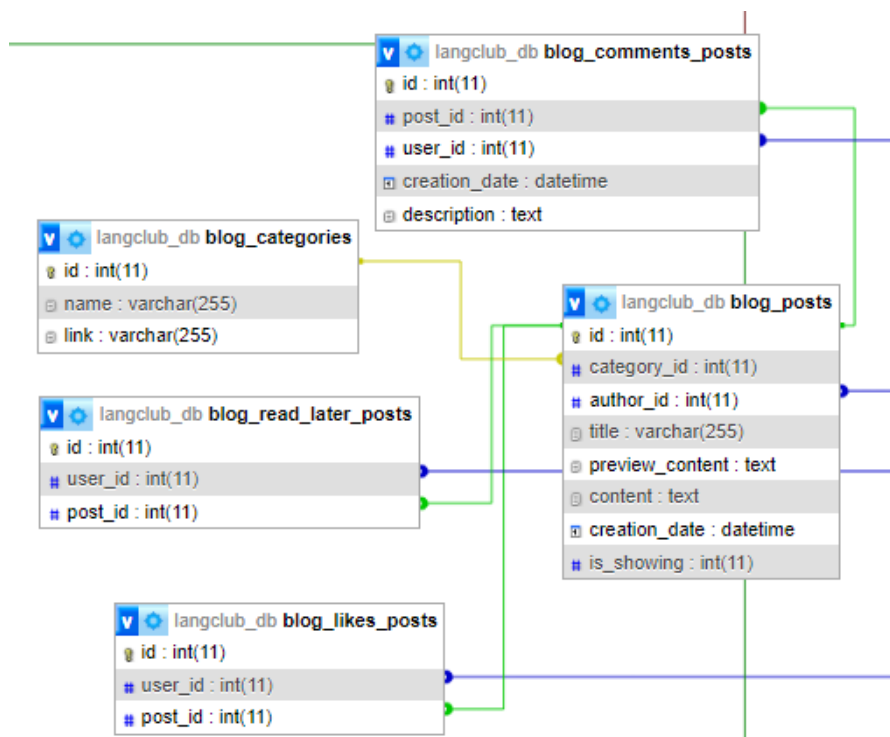
А.1 – Навчальні таблиці читання та словника в базі даних



А.2 – Навчальні таблиці письма та аудіювання в базі даних



А.3 – Таблиці інтернет-магазину



А.4 – Таблиці блогу

Модель «Words.php»

```

<?php
class Words {
    const SHOW_BY_DEFAULT = 20;
    public static function getBasicWords() {
        $db = Db::getConnection();
        $result = $db->query('SELECT * FROM studying_words WHERE language_id=1 AND level_id=1 AND
word_category_id=1');
        $basic_words = [];
        if ($result && $result->rowCount() > 0) { $i = 0;
            while ($row = $result->fetch()) {
                $basic_words[$i]['id'] = $row['id'];
                $basic_words[$i]['language_id'] = $row['language_id'];
                $basic_words[$i]['level_id'] = $row['level_id'];
                $basic_words[$i]['word_category_id'] = $row['word_category_id'];
                $basic_words[$i]['word_text'] = $row['word_text'];
                $basic_words[$i]['word_transcription'] = $row['word_transcription'];
                $basic_words[$i]['word_translation'] = $row['word_translation'];
                $i++;}
            return $basic_words;}
        public static function getWordsCategories() {
            $db = Db::getConnection();
            $result = $db->query('SELECT * FROM studying_words_categories ');
            $words_categories = [];
            if ($result && $result->rowCount() > 0) { $i = 0;
                while ($row = $result->fetch()) {
                    $words_categories[$i]['id'] = $row['id'];
                    $words_categories[$i]['category'] = $row['category']; $i++;}}
            return $words_categories;}
        public static function getAllWords($page = 1) {
            $page = intval($page);
            $offset = ($page - 1) * self::SHOW_BY_DEFAULT;
            $db = Db::getConnection();
            $result = $db->query('SELECT * FROM studying_words LIMIT ' . self::SHOW_BY_DEFAULT . '
OFFSET ' . $offset); $words = [];
            if ($result && $result->rowCount() > 0) { $i = 0;
                while ($row = $result->fetch()) {
                    $words[$i]['id'] = $row['id'];
                    $words[$i]['language_id'] = $row['language_id'];
                    $words[$i]['level_id'] = $row['level_id'];
                    $words[$i]['word_category_id'] = $row['word_category_id'];
                    $words[$i]['word_text'] = $row['word_text'];
                    $words[$i]['word_transcription'] = $row['word_transcription'];
                    $words[$i]['word_translation'] = $row['word_translation']; $i++;}}
            return $words;}
        public static function getWordById($id) {
            $id = intval($id); $db = Db::getConnection();
            $sql = 'SELECT * FROM studying_words WHERE id = :id';

```

					ІПЗ.КР.Б – 121 – 24 – ІЗ	Арк.
						84
Змн.	Арк.	№ докум.	Підпис	Дата		

```

$result = $db->prepare($sql);
$result->bindParam(':id', $id, PDO::PARAM_INT);
$result->execute(); $word = [];
if ($result && $row = $result->fetch()) {
    $word['id'] = $row['id'];
    $word['language_id'] = $row['language_id'];
    $word['level_id'] = $row['level_id'];
    $word['word_category_id'] = $row['word_category_id'];
    $word['word_text'] = $row['word_text'];
    $word['word_transcription'] = $row['word_transcription'];
    $word['word_translation'] = $row['word_translation'];
}return $word;}

public static function getWordsCategoryById($id) {
    $id = intval($id); $db = Db::getConnection();
    $sql = 'SELECT * FROM studying_words_categories WHERE id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);
    $result->execute(); $category = [];
    if ($result && $row = $result->fetch()) {
        $category['id'] = $row['id']; $category['category'] = $row['category'];}
    return $category;}

public static function getTotalWords() {
    $db = Db::getConnection();
    $sql = 'SELECT count(id) AS count FROM studying_words ';
    $result = $db->prepare($sql); $result->execute();
    $row = $result->fetch(); return $row['count'];}

public static function create($language_id, $level_id, $word_category_id, $word_text,
$word_transcription, $word_translation) {
    $db = Db::getConnection();
    $sql = "INSERT INTO studying_words (language_id, level_id, word_category_id, word_text,
word_transcription, word_translation)
VALUES (:language_id, :level_id, :word_category_id, :word_text,
:word_transcription, :word_translation)";
    $result = $db->prepare($sql);
    $result->bindParam(':language_id', $language_id, PDO::PARAM_INT);
    $result->bindParam(':level_id', $level_id, PDO::PARAM_INT);
    $result->bindParam(':word_category_id', $word_category_id, PDO::PARAM_INT);
    $result->bindParam(':word_text', $word_text, PDO::PARAM_STR);
    $result->bindParam(':word_transcription', $word_transcription, PDO::PARAM_STR);
    $result->bindParam(':word_translation', $word_translation, PDO::PARAM_STR);
    if($result->execute()) {
        return $db->lastInsertId();}
    return 0;}

public static function edit($id, $language_id, $level_id, $word_category_id, $word_text,
$word_transcription, $word_translation) {
    $db = Db::getConnection(); $sql = "";
    $sql = "UPDATE studying_words SET language_id = :language_id, level_id = :level_id,
word_category_id = :word_category_id, word_text = :word_text, word_transcription =
:word_transcription, word_translation = :word_translation WHERE id = :id"; $result = $db-
>prepare($sql);
    $result->bindParam(':language_id', $language_id, PDO::PARAM_INT);

```

					ІІЗ.КР.Б – 121 – 24 – ІІЗ	Арк.
						85
Змн.	Арк.	№ докум.	Підпис	Дата		

```

$result->bindParam(':level_id', $level_id, PDO::PARAM_INT);
$result->bindParam(':word_category_id', $word_category_id, PDO::PARAM_INT);
$result->bindParam(':word_text', $word_text, PDO::PARAM_STR);
$result->bindParam(':word_transcription', $word_transcription, PDO::PARAM_STR);
$result->bindParam(':word_translation', $word_translation, PDO::PARAM_STR);
$result->bindParam(':id', $id, PDO::PARAM_INT);
return $result->execute();}

public static function delete($id) {
    $db = Db::getConnection();
    $sql = 'DELETE FROM studying_words WHERE id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);
    return $result->execute();}

public static function deleteWordsByCategory($id) {
    $db = Db::getConnection();
    $sql = 'DELETE FROM studying_words WHERE word_category_id = :id';
    $stmt = $db->prepare($sql);
    $stmt->bindParam(':id', $id, PDO::PARAM_INT);
    return $stmt->execute();}

public static function deleteCategory($id) {
    $db = Db::getConnection();
    $sql = 'DELETE FROM studying_words_categories WHERE id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);
    return $result->execute();}

public static function createCategory($category) {
    $db = Db::getConnection();
    $sql = "INSERT INTO studying_words_categories (category)
            VALUES (:category)";
    $result = $db->prepare($sql);
    $result->bindParam(':category', $category, PDO::PARAM_STR);
    if($result->execute()) {
        return $db->lastInsertId();}
    return 0;}

public static function editWordsCategory($id, $category) {
    $db = Db::getConnection(); $sql = "";
    $sql = "UPDATE studying_words_categories SET category = :category
    WHERE id = :id";
    $result = $db->prepare($sql);
    $result->bindParam(':category', $category, PDO::PARAM_STR);
    $result->bindParam(':id', $id, PDO::PARAM_INT);return $result->execute();}

```

Модель «Reading»

<?php

```

class Reading {
    const SHOW_BY_DEFAULT = 10;
    public static function getShowingReadingPosts($page = 1) {
        $page = intval($page); $offset = ($page - 1) * self::SHOW_BY_DEFAULT;
        $db = Db::getConnection();

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ІЗ | Арк. |
| | | | | | | 86 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

        $result = $db->query('SELECT * FROM studying_reading WHERE is_showing=1 ORDER BY
creation_date DESC LIMIT ' . self::SHOW_BY_DEFAULT . ' OFFSET ' . $offset); $reading_posts = [];
        if ($result && $result->rowCount() > 0) { $i = 0;
            while ($row = $result->fetch()) {
                $reading_posts[$i]['id'] = $row['id'];
                $reading_posts[$i]['name'] = $row['name'];
                $reading_posts[$i]['language'] = $row['language'];
                $reading_posts[$i]['level'] = $row['level'];
                $reading_posts[$i]['description'] = $row['description'];
                $reading_posts[$i]['reading_text'] = $row['reading_text'];
                $reading_posts[$i]['subcategory_id'] = $row['subcategory_id'];
                $reading_posts[$i]['creation_date'] = $row['creation_date'];
                $i++;} return $reading_posts;}

        public static function getTestQuestionsByReadingId($reading_id) {
            $db = Db::getConnection();
            $stmt = $db->prepare('SELECT * FROM studying_reading_questions WHERE reading_id =
:reading_id');
            $stmt->bindParam(':reading_id', $reading_id, PDO::PARAM_INT);
            $stmt->execute(); $test_questions = [];
            if ($stmt && $stmt->rowCount() > 0) {
                $test_questions = $stmt->fetchAll(PDO::FETCH_ASSOC);} return $test_questions;}

        public static function getAnswersByQuestionId($question_id) {
            $db = Db::getConnection();
            $stmt = $db->prepare('SELECT * FROM studying_reading_answers WHERE question_id =
:question_id');
            $stmt->bindParam(':question_id', $question_id, PDO::PARAM_INT); $stmt->execute();
            $test_answers = [];
            if ($stmt && $stmt->rowCount() > 0) {
                $test_answers = $stmt->fetchAll(PDO::FETCH_ASSOC);} return $test_answers;}

        public static function getTotalShowingReadingPosts() {
            $db = Db::getConnection();
            $sql = 'SELECT count(id) AS count FROM studying_reading WHERE is_showing=1';
            $result = $db->prepare($sql); $result->execute();
            $row = $result->fetch(); return $row['count'];}

        public static function getReadingSubcategories() { $db = Db::getConnection();
            $result = $db->query('SELECT * FROM studying_subcategories WHERE category_id = 2');
            $reading_subcategories = [];
            if ($result && $result->rowCount() > 0) { $i = 0;
                while ($row = $result->fetch()) {
                    $reading_subcategories[$i]['id'] = $row['id'];
                    $reading_subcategories[$i]['name'] = $row['name'];
                    $reading_subcategories[$i]['link'] = $row['link'];$i++;}}
            return $reading_subcategories;}

        public static function getReadingCategoryNameById($subcategory_id) {
            $db = Db::getConnection();
            $sql = "SELECT name FROM studying_subcategories WHERE id = :subcategory_id";
            $stmt = $db->prepare($sql);
            $stmt->bindParam(':subcategory_id', $subcategory_id, PDO::PARAM_INT);
            $stmt->execute(); $result = $stmt->fetch(PDO::FETCH_ASSOC);
            return isset($result['name']) ? $result['name'] : null;}

        public static function userLikedPost($user_id, $post_id) {

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 87 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

$db = Db::getConnection();
$stmt = $db->prepare("SELECT COUNT(*) FROM studying_reading_likes_posts WHERE user_id
= :user_id AND post_id = :post_id");
$stmt->bindParam(':user_id', $user_id);
$stmt->bindParam(':post_id', $post_id); $stmt->execute();
$count = $stmt->fetchColumn();
if ($count > 0) {return true; } else {return false; }}
public static function countLikesByPostId($post_id) { $db = Db::getConnection();
$sql = 'SELECT count(id) AS count FROM studying_reading_likes_posts WHERE
post_id=:post_id';
$stmt = $db->prepare($sql); $stmt->bindParam(':post_id', $post_id, PDO::PARAM_INT);
$stmt->execute(); $row = $stmt->fetch(); return $row['count'];}
public static function countPostsQuantityByCategoryId($category_id) {
$db = Db::getConnection();
$sql = 'SELECT COUNT(*) AS post_count FROM studying_reading WHERE subcategory_id =
:category_id'; $stmt = $db->prepare($sql);
$stmt->bindParam(':category_id', $category_id, PDO::PARAM_INT);
$stmt->execute(); $result = $stmt->fetch(PDO::FETCH_ASSOC);
return $result['post_count'];}
public static function getPostById($id) { $id = intval($id);
if ($id) { $db = Db::getConnection();
$result = $db->query('SELECT * FROM studying_reading WHERE id=' . $id);
$result->setFetchMode(PDO::FETCH_ASSOC); return $result->fetch();}
public static function isUserReadLaterPost($user_id, $post_id) {
$db = Db::getConnection();
$stmt = $db->prepare("SELECT COUNT(*) FROM studying_reading_read_later WHERE user_id
= :user_id AND post_id = :post_id"); $stmt->bindParam(':user_id', $user_id);
$stmt->bindParam(':post_id', $post_id); $stmt->execute();
$count = $stmt->fetchColumn();
if ($count > 0) { return true; } else {return false; }}
public static function getPostsMinCreationDate() { $db = Db::getConnection();
$sql = 'SELECT MIN(creation_date) AS min_date FROM studying_reading';
$stmt = $db->query($sql); $row = $stmt->fetch(PDO::FETCH_ASSOC);
$minDate = date('Y-m-d', strtotime($row['min_date'])); return $minDate;}
public static function getPostsMaxCreationDate() {
$db = Db::getConnection();
$sql = 'SELECT MAX(creation_date) AS max_date FROM studying_reading';
$stmt = $db->query($sql); $row = $stmt->fetch(PDO::FETCH_ASSOC);
$maxDate = date('Y-m-d', strtotime($row['max_date'])); return $maxDate;}
public static function checkIfUserProcessedPost($user_id, $post_id) {
$db = Db::getConnection();
$stmt = $db->prepare("SELECT COUNT(*) FROM studying_reading_results WHERE user_id =
:user_id AND post_id = :post_id"); $stmt->bindParam(':user_id', $user_id);
$stmt->bindParam(':post_id', $post_id); $stmt->execute();
$count = $stmt->fetchColumn();
if ($count > 0) { return true; } else {return false; }}
public static function getReadLaterReadingPostsByUserId($user_id) {
$db = Db::getConnection();
$result = $db->query('SELECT * FROM studying_reading_read_later WHERE user_id =
'.$user_id); $read_later_posts = [];
if ($result && $result->rowCount() > 0) { $i = 0;

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 88 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

        while ($row = $result->fetch()) {
            $read_later_posts[$i]['id'] = $row['id'];
            $read_later_posts[$i]['user_id'] = $row['user_id'];
            $read_later_posts[$i]['post_id'] = $row['post_id'];
            $i++;} return $read_later_posts;}

public static function getPostDataById($post_id) {$id = intval($post_id);
    if ($id) { $db = Db::getConnection();
        $result = $db->query('SELECT * FROM studying_reading WHERE is_showing=1 AND id=' . $id);
        $result->setFetchMode(PDO::FETCH_ASSOC); return $result->fetch();}}

public static function getQuestionById($question_id) { $id = intval($question_id);
    if ($id) { $db = Db::getConnection();
        $result = $db->query('SELECT * FROM studying_reading_questions WHERE id=' . $id);
        $result->setFetchMode(PDO::FETCH_ASSOC); return $result->fetch();}}

public static function getQuantityProcessedByUserReadingPosts($user_id) {
    $db = Db::getConnection();
    $stmt = $db->prepare("SELECT COUNT(*) AS count FROM studying_reading_results WHERE
user_id = :user_id");
    $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT);
    $stmt->execute(); $row = $stmt->fetch(); return $row['count'];}

public static function getAveragePercentageOfTestSuccess($user_id) {
    $db = Db::getConnection();
    $stmt = $db->prepare("SELECT * FROM studying_reading_results WHERE user_id =
:user_id"); $stmt->bindParam(':user_id', $user_id, PDO::PARAM_INT); $stmt->execute();
    $rows = $stmt->fetchAll(PDO::FETCH_ASSOC); $avg_percents_sum = 0;
    $total_tests = count($rows);
    foreach ($rows as $row) { if($row['total_questions'] != 0) {
        $avg_percents_sum += ($row['correct_answers'] / $row['total_questions']) *
100;} else { $avg_percents_sum += 0;}}
    if ($total_tests > 0) {
        return $avg_percents_sum / $total_tests; } else {return 0;}}

public static function getAllPosts($page = 1) {
    $page = intval($page);
    $offset = ($page - 1) * self::SHOW_BY_DEFAULT;
    $db = Db::getConnection();
    $result = $db->query('SELECT * FROM studying_reading LIMIT ' . self::SHOW_BY_DEFAULT
. ' OFFSET ' . $offset); $posts = [];
    if ($result && $result->rowCount() > 0) { $i = 0;
        while ($row = $result->fetch()) {
            $posts[$i]['id'] = $row['id'];
            $posts[$i]['name'] = $row['name'];
            $posts[$i]['language'] = $row['language'];
            $posts[$i]['level'] = $row['level'];
            $posts[$i]['description'] = $row['description'];
            $posts[$i]['reading_text'] = $row['reading_text'];
            $posts[$i]['subcategory_id'] = $row['subcategory_id'];
            $posts[$i]['creation_date'] = $row['creation_date'];
            $posts[$i]['is_showing'] = $row['is_showing'];
            $i++;} return $posts;}

public static function getAllTotalPosts() { $db = Db::getConnection();
    $result = $db->query('SELECT * FROM studying_reading '); $posts = [];
    if ($result && $result->rowCount() > 0) {$i = 0;

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 89 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

        while ($row = $result->fetch()) {
            $posts[$i]['id'] = $row['id'];
            $posts[$i]['name'] = $row['name'];
            $posts[$i]['language'] = $row['language'];
            $posts[$i]['level'] = $row['level'];
            $posts[$i]['description'] = $row['description'];
            $posts[$i]['reading_text'] = $row['reading_text'];
            $posts[$i]['subcategory_id'] = $row['subcategory_id'];
            $posts[$i]['creation_date'] = $row['creation_date'];
            $posts[$i]['is_showing'] = $row['is_showing']; $i++;}}
    return $posts;}

    public static function getAllQuestions($page = 1) {
        $page = intval($page); $offset = ($page - 1) * self::SHOW_BY_DEFAULT;
        $db = Db::getConnection();
        $result = $db->query('SELECT * FROM studying_reading_questions LIMIT ' .
self::SHOW_BY_DEFAULT . ' OFFSET ' . $offset); $questions = [];
        if ($result && $result->rowCount() > 0) { $i = 0;
            while ($row = $result->fetch()) {
                $questions[$i]['id'] = $row['id'];
                $questions[$i]['question_text'] = $row['question_text'];
                $questions[$i]['reading_id'] = $row['reading_id']; $i++;}}
            return $questions;}

    public static function getAllAnswers($page = 1) {
        $page = intval($page); $offset = ($page - 1) * self::SHOW_BY_DEFAULT;
        $db = Db::getConnection();
        $result = $db->query('SELECT * FROM studying_reading_answers LIMIT ' .
self::SHOW_BY_DEFAULT . ' OFFSET ' . $offset); $answers = [];
        if ($result && $result->rowCount() > 0) { $i = 0;
            while ($row = $result->fetch()) {
                $answers[$i]['id'] = $row['id'];
                $answers[$i]['question_id'] = $row['question_id'];
                $answers[$i]['answer_text'] = $row['answer_text'];
                $answers[$i]['is_correct'] = $row['is_correct'];
                $i++; }} return $answers; }

    public static function getAllQuantityAnswers() { $db = Db::getConnection();
        $sql = 'SELECT count(id) AS count FROM studying_reading_answers';
        $result = $db->prepare($sql); $result->execute();
        $row = $result->fetch(); return $row['count']; }

    public static function getAllQuantityQuestions() { $db = Db::getConnection();
        $sql = 'SELECT count(id) AS count FROM studying_reading_questions';
        $result = $db->prepare($sql); $result->execute();
        $row = $result->fetch(); return $row['count']; }

    public static function getAllQuantityPosts() { $db = Db::getConnection();
        $sql = 'SELECT count(id) AS count FROM studying_reading';
        $result = $db->prepare($sql); $result->execute();
        $row = $result->fetch(); return $row['count']; }

    public static function create($name, $language, $level, $subcategory_id, $description,
$reading_text, $is_showing) { $db = Db::getConnection();
        $sql = "INSERT INTO studying_reading (name, language, level, subcategory_id,
description, reading_text, is_showing) VALUES (:name, :language, :level, :subcategory_id,
:description, :reading_text, :is_showing)"; $result = $db->prepare($sql);

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 90 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

$result->bindParam(':name', $name, PDO::PARAM_STR);
$result->bindParam(':language', $language, PDO::PARAM_INT);
$result->bindParam(':level', $level, PDO::PARAM_INT);
$result->bindParam(':subcategory_id', $subcategory_id, PDO::PARAM_INT);
$result->bindParam(':description', $description);
$result->bindParam(':reading_text', $reading_text);
$result->bindParam(':is_showing', $is_showing, PDO::PARAM_INT);
if($result->execute()) {return $db->lastInsertId();} return 0;}

public static function edit($id, $name, $language, $level, $subcategory_id, $description,
$reading_text, $is_showing) {
    $db = Db::getConnection(); $sql = ""; $sql = "UPDATE studying_reading
    SET name = :name, language = :language, level = :level, subcategory_id =
    :subcategory_id, description = :description, reading_text = :reading_text, is_showing =
    :is_showing WHERE id = :id"; $result = $db->prepare($sql);
    $result->bindParam(':name', $name, PDO::PARAM_STR);
    $result->bindParam(':language', $language, PDO::PARAM_INT);
    $result->bindParam(':level', $level, PDO::PARAM_INT);
    $result->bindParam(':subcategory_id', $subcategory_id, PDO::PARAM_INT);
    $result->bindParam(':description', $description);
    $result->bindParam(':reading_text', $reading_text);
    $result->bindParam(':is_showing', $is_showing, PDO::PARAM_INT);
    $result->bindParam(':id', $id, PDO::PARAM_INT); return $result->execute();}

public static function delete($id) { $db = Db::getConnection();
    $sql = 'DELETE FROM studying_reading_likes_posts WHERE post_id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);$result->execute();
    $sql = 'DELETE FROM studying_reading_read_later WHERE post_id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);$result->execute();
    $sql = 'DELETE FROM studying_reading_results WHERE post_id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);$result->execute();
    $sql = 'SELECT id FROM studying_reading_questions WHERE reading_id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);$result->execute();
    $ids_questions = $result->fetchAll(PDO::FETCH_COLUMN, 0);
    foreach ($ids_questions as $id_question) { Reading::deleteQuestion($id_question);}
    $sql = 'DELETE FROM studying_reading WHERE id = :id'; $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT); return $result->execute();}

public static function deleteQuestion($id) { $db = Db::getConnection();
    $sql = 'DELETE FROM studying_reading_answers WHERE question_id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT); $result->execute();
    $sql = 'DELETE FROM studying_reading_questions WHERE id = :id';
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);return $result->execute();}

public static function createQuestion($reading_id, $question_text) {
    $db = Db::getConnection();
    $sql = "INSERT INTO studying_reading_questions (reading_id, question_text)
    VALUES (:reading_id, :question_text)"; $result = $db->prepare($sql);
    $result->bindParam(':question_text', $question_text, PDO::PARAM_STR);

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 91 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

$result->bindParam(':reading_id', $reading_id, PDO::PARAM_INT);
return $result->execute();}

public static function editQuestion($id, $question_text, $reading_id) {
$db = Db::getConnection(); $sql = "";
$sql = "UPDATE studying_reading_questions SET question_text = :question_text,
reading_id = :reading_id WHERE id = :id"; $result = $db->prepare($sql);
$result->bindParam(':question_text', $question_text, PDO::PARAM_STR);
$result->bindParam(':reading_id', $reading_id, PDO::PARAM_INT);
$result->bindParam(':id', $id, PDO::PARAM_INT);return $result->execute();}

public static function deleteAnswer($id) {
$db = Db::getConnection();$sql = 'DELETE FROM studying_reading_answers WHERE id = :id';
$result = $db->prepare($sql);
$result->bindParam(':id', $id, PDO::PARAM_INT); return $result->execute(); }

public static function getAllQuestionsAtOnce() { $db = Db::getConnection();
$result = $db->query('SELECT * FROM studying_reading_questions '); $questions = [];
if ($result && $result->rowCount() > 0) { $i = 0;
while ($row = $result->fetch()) { $questions[$i]['id'] = $row['id'];
$questions[$i]['question_text'] = $row['question_text'];
$questions[$i]['reading_id'] = $row['reading_id'];$i++;}}return $questions; }

public static function createAnswer($question_id, $answer_text, $is_correct) {
$db = Db::getConnection();
$sql = "INSERT INTO studying_reading_answers (question_id, answer_text, is_correct)
VALUES (:question_id, :answer_text, :is_correct)";
$result = $db->prepare($sql);
$result->bindParam(':answer_text', $answer_text, PDO::PARAM_STR);
$result->bindParam(':question_id', $question_id, PDO::PARAM_INT);
$result->bindParam(':is_correct', $is_correct, PDO::PARAM_INT);
return $result->execute();}

public static function editAnswer($id, $question_id, $answer_text, $is_correct) { $db =
Db::getConnection(); $sql = "";
$sql = "UPDATE studying_reading_answers
SET question_id = :question_id, answer_text = :answer_text, is_correct = :is_correct
WHERE id = :id"; $result = $db->prepare($sql);
$result->bindParam(':answer_text', $answer_text, PDO::PARAM_STR);
$result->bindParam(':question_id', $question_id, PDO::PARAM_INT);
$result->bindParam(':is_correct', $is_correct, PDO::PARAM_INT);
$result->bindParam(':id', $id, PDO::PARAM_INT); return $result->execute();}

public static function getAnswerById($answer_id) { $id = intval($answer_id);
if ($id) { $db = Db::getConnection();
$result = $db->query('SELECT * FROM studying_reading_answers WHERE id=' . $id);
$result->setFetchMode(PDO::FETCH_ASSOC);return $result->fetch();}}?>

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 92 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Контролер «WordsController»

```
<?php class WordsController {
    public function actionIndex() {
        $site_categories = Site::getSiteCategories();
        $studying_categories = Site::getStudyingCategories();
        $studying_subcategories = Site::getStudyingSubcategories();
        $products_categories = Site::getProductsCategories();
        $products_subcategories = Site::getProductsSubcategories();
        $blog_categories = Site::getBlogCategories();
        $site_info = Site::getSiteInfo();
        $advertising_info = Site::getAdvertisingInfo();
        $languages = Site::getAllLearningLanguages();
        $levels = Site::getAllLearningLanguageLevels();
        $words_categories = Words::getWordsCategories();
        $basic_words = Words::getBasicWords();
        require_once('views/words/index.php'); return true;}}?>
```

Контролер «ReadingController»

```
<?php class ReadingController {
    public function actionIndex($page = 1) {
        $site_categories = Site::getSiteCategories();
        $studying_categories = Site::getStudyingCategories();
        $studying_subcategories = Site::getStudyingSubcategories();
        $products_categories = Site::getProductsCategories();
        $products_subcategories = Site::getProductsSubcategories();
        $blog_categories = Site::getBlogCategories();
        $site_info = Site::getSiteInfo();
        $advertising_info = Site::getAdvertisingInfo();
        $languages = Site::getAllLearningLanguages();
        $levels = Site::getAllLearningLanguageLevels();
        $reading_subcategories = Reading::getReadingSubcategories();
        $page = preg_replace('/[^0-9]/i', ' ', $page);
        $posts = Reading::getShowingReadingPosts($page);
        $total_posts = Reading::getTotalShowingReadingPosts();
        $pagination = new Pagination($total_posts, $page, Reading::SHOW_BY_DEFAULT, 'page-');
        require_once('views/reading/index.php'); return true;}

    public function actionView($post_id) {
        $site_categories = Site::getSiteCategories();
        $studying_categories = Site::getStudyingCategories();
        $studying_subcategories = Site::getStudyingSubcategories();
        $products_categories = Site::getProductsCategories();
        $products_subcategories = Site::getProductsSubcategories();
        $blog_categories = Site::getBlogCategories();
        $site_info = Site::getSiteInfo();
        $advertising_info = Site::getAdvertisingInfo();
        $post = Reading::getPostById($post_id);
        if(!$post) header("Location: /404"); $userId = User::getAuthUserId();
        $test_questions = Reading::getTestQuestionsByReadingId($post_id);
        require_once('views/reading/view.php'); return true;}}?>
```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 93 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Лістинг view/words/index.php

```

<!DOCTYPE html>
<html class="no-js" lang="en"><head>
    <meta charset="utf-8" />
    <title>LingoUni - Вивчення слів</title>
    <meta http-equiv="x-ua-compatible" content="ie=edge" />
    <meta name="description" content="" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta property="og:title" content="" /><meta property="og:type" content="" />
    <meta property="og:url" content="" /><meta property="og:image" content="" />
    <link rel="shortcut icon" type="image/x-icon" href="/styles/frontend/imgs/theme/favicon.ico" />
    <link rel="stylesheet" href="/styles/frontend/css/main.css?v=5.6" />
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
</head><body>
    <?php require_once('views/layouts/header.php'); ?>
    <main class="main"><div class="page-header mt-30 mb-75">
        <div class="container"><div class="archive-header">
            <div class="row align-items-center"><div class="col-xl-5">
                <h1 class="mb-15">Вивчення слів</h1>
                <div class="breadcrumb"><a href="/" rel="nofollow"><i class="fi-rs-
home mr-5"></i>Головна</a><span></span> Вивчення слів</div></div></div></div></div>
            <div class="container"><div class="row"><div class="col-lg-9" style="margin: 0 auto;">
                <div class="shop-product-fillter mb-50 pr-30">
                    <div class="total-product">
                        <h2>Вправи</h2></div>
                    <div class="sort-by-cover mr-10">
                        <label for="language-selection">Оберіть мову:</label>
                        <select name="language-selection" id="language-selection">
                            <?php foreach ($languages as $language): ?>
                                <option value="<?php echo $language['id']; ?>"><?php echo
$language['name']; ?></option><?php endforeach; ?></select></div>
                        <div class="sort-by-cover mr-10">
                            <label for="level-selection">Оберіть рівень:</label>
                            <select name="level-selection" id="level-selection">
                                <?php foreach ($levels as $level): ?>
                                    <option value="<?php echo $level['id']; ?>"><?php echo
$level['name']; ?></option> <?php endforeach; ?></select></div>
                        <div class="sort-by-cover mr-10">
                            <label for="category-selection">Оберіть категорію:</label>
                            <select name="category-selection" id="category-selection">
                                <?php foreach ($words_categories as $category): ?>
                                    <option value="<?php echo $category['id']; ?>"><?php echo
$category['category']; ?></option>
                                <?php endforeach; ?></select></div></div>
                        <div id="test-content">
                            <button class="btn" id="startTestButton">Почати</button></div></div></div>
                    </div><div class="container"><div class="row">

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 94 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

<div class="col-lg-9" style="margin: 0 auto;">
    <div class="shop-product-fillter mb-50 pr-30"><div class="total-product">
        <h2>Словник</h2></div>
        <div class="sort-by-cover mr-10">
            <label for="language-vocabulary-selection">Оберіть мову:</label>
            <select name="language-vocabulary-selection" id="language-vocabulary-
selection"><?php foreach ($languages as $language): ?>
                <option value="<?php echo $language['id']; ?>"><?php echo
$language['name']; ?></option>
            <?php endforeach; ?></select></div>
            <div class="sort-by-cover mr-10">
                <label for="level-selection">Оберіть рівень:</label>
                <select name="level-vocabulary-selection" id="level-vocabulary-
selection">
                    <?php foreach ($levels as $level): ?>
                        <option value="<?php echo $level['id']; ?>"><?php echo
$level['name']; ?></option><?php endforeach; ?></select></div><div class="sort-by-cover mr-10">
                    <label for="category-selection">Оберіть категорію:</label>
                    <select name="category-vocabulary-selection" id="category-vocabulary-
selection">
                        <?php foreach ($words_categories as $category): ?>
                            <option value="<?php echo $category['id']; ?>"><?php echo
$category['category']; ?></option>
                        <?php endforeach; ?></select></div></div>
            <div id="vocabulary-content">
                <table class="table" style="text-align: center;">
                    <tr><th>Слово</th><th>Транскрипція</th><th>Переклад</th></tr>
                    <?php foreach ($basic_words as $word): ?><tr>
                        <td><?php echo $word['word_text']; ?></td>
                        <td><?php echo $word['word_transcription']; ?></td>
                        <td><?php echo $word['word_translation']; ?></td></tr>
                    <?php endforeach; ?></table></div></div></div></main>
        <?php require_once('views/layouts/footer.php'); ?><script>
let languageVocabularyValue = 1; let levelVocabularyValue = 1; let categoryVocabularyValue = 1;
$("#language-vocabulary-selection").change(function() {
    languageVocabularyValue = $(this).val();generateVocabulary();});
$("#level-vocabulary-selection").change(function() {
    levelVocabularyValue = $(this).val();generateVocabulary();});
$("#category-vocabulary-selection").change(function() {
    categoryVocabularyValue = $(this).val();generateVocabulary();});
function generateVocabulary() {
    $.ajax({ url: "../controllers/ajax/generateWordsVocabulary.php", method: "POST",
        data: { languageVocabularyValue: languageVocabularyValue,
levelVocabularyValue:levelVocabularyValue, categoryVocabularyValue:categoryVocabularyValue},
        success: function(data) {$("#vocabulary-content").html(data);}});}
let languageSelectionElement = document.getElementById("language-selection");
let levelSelectionElement = document.getElementById("level-selection");
let categorySelectionElement = document.getElementById("category-selection");
$("#startTestButton").click(function(e) {generateTest();});
$("#language-selection").change(function() {generateTest();});

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 95 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

$("#level-selection").change(function() {generateTest();});
$("#category-selection").change(function() {generateTest();});
function generateTest() { var languageSelection = languageSelectionElement.value;
var levelSelection = levelSelectionElement.value;
var categorySelection = categorySelectionElement.value;
$.ajax({url: "../controllers/ajax/generateWordsLearning.php",
method: "POST", data: { languageSelection: languageSelection,
levelSelection:levelSelection, categorySelection:categorySelection},
success: function(data) {$('#test-content').html(data);}});}
function continueTest() {
var languageSelectionElement = document.getElementById("language-selection");
var languageSelection = languageSelectionElement.value;
var levelSelectionElement = document.getElementById("level-selection");
var levelSelection = levelSelectionElement.value;
var categorySelectionElement = document.getElementById("category-selection");
var categorySelection = categorySelectionElement.value;
$.ajax({url: "../controllers/ajax/generateWordsLearning.php", method: "POST",
data: { languageSelection: languageSelection, levelSelection:levelSelection,
categorySelection:categorySelection},
success: function(data) { $('#test-content').html(data);}});}
function checkAnswerTest(e) { if (e.getAttribute('data-correct') == '1') {
let parentRow = e.closest('tr'); parentRow.style.backgroundColor = 'lightgreen';
} else {
let parentRow = e.closest('tr'); parentRow.style.backgroundColor = 'lightcoral';}}
</script>
<script src="/styles/frontend/js/vendor/modernizr-3.6.0.min.js"></script>
<script src="/styles/frontend/js/vendor/jquery-3.6.0.min.js"></script>
<script src="/styles/frontend/js/vendor/jquery-migrate-3.3.0.min.js"></script>
<script src="/styles/frontend/js/vendor/bootstrap.bundle.min.js"></script>
<script src="/styles/frontend/js/plugins/slick.js"></script>
<script src="/styles/frontend/js/plugins/jquery.syotimer.min.js"></script>
<script src="/styles/frontend/js/plugins/wow.js"></script>
<script src="/styles/frontend/js/plugins/slider-range.js"></script>
<script src="/styles/frontend/js/plugins/perfect-scrollbar.js"></script>
<script src="/styles/frontend/js/plugins/magnific-popup.js"></script>
<script src="/styles/frontend/js/plugins/select2.min.js"></script>
<script src="/styles/frontend/js/plugins/waypoints.js"></script>
<script src="/styles/frontend/js/plugins/counterup.js"></script>
<script src="/styles/frontend/js/plugins/jquery.countdown.min.js"></script>
<script src="/styles/frontend/js/plugins/images-loaded.js"></script>
<script src="/styles/frontend/js/plugins/isotope.js"></script>
<script src="/styles/frontend/js/plugins/scrollup.js"></script>
<script src="/styles/frontend/js/plugins/jquery.vticker-min.js"></script>
<script src="/styles/frontend/js/plugins/jquery.theia.sticky.js"></script>
<script src="/styles/frontend/js/plugins/jquery.elevatezoom.js"></script>
<script src="/styles/frontend/js/main.js?v=5.6"></script>
<script src="/styles/frontend/js/shop.js?v=5.6"></script>
</body></html>

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 96 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Лістинг view/reading/view.php

```
<!DOCTYPE html>
<html class="no-js" lang="en"><head>
  <meta charset="utf-8" />
  <title>LingoUni - Пост читання</title>
  <meta http-equiv="x-ua-compatible" content="ie=edge" />
  <meta name="description" content="" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <meta property="og:title" content="" />
  <meta property="og:type" content="" />
  <meta property="og:url" content="" />
  <meta property="og:image" content="" />
  <link rel="shortcut icon" type="image/x-icon" href="/styles/frontend/imgs/theme/favicon.ico" />
  <link rel="stylesheet" href="/styles/frontend/css/plugins/slider-range.css" />
  <link rel="stylesheet" href="/styles/frontend/css/main.css?v=5.6" />
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>
</head>
<body class="single-product">
  <?php require_once('views/layouts/header.php'); ?>
  <main class="main">
    <div class="page-header breadcrumb-wrap">
      <div class="container">
        <div class="breadcrumb">
          <a href="/" rel="nofollow"><i class="fi-rs-home mr-5"></i>Головна</a>
          <span></span> Пост читання</div></div></div>
        <div class="page-content mb-50">
          <div class="container">
            <div class="row">
              <div class="col-xl-11 col-lg-12 m-auto">
                <div class="row">
                  <div class="col-lg-9">
                    <div class="single-page pt-50 pr-30">
                      <div class="single-header style-2">
                        <div class="row">
                          <div class="col-xl-10 col-lg-12 m-auto">
                            <h6 class="mb-10"><a href="#"><?php echo
Reading::getReadingCategoryId($post['subcategory_id']); ?></a></h6>
                            <h2 class="mb-10"><?php echo $post['name'];
?></h2></div></div></div>
                            <figure class="single-thumbnail">
                              
                              </figure>
                            <div class="single-content"><div class="row">
                              <div class="col-xl-10 col-lg-12 m-auto">
                                <?php echo $post['reading_text']; ?>
                                <?php if(!User::isGuest()): ?>
                                  <h3 class="mb-10">Закріплючий тест</h3>
                                  <div class="test-container">
                                    <form id="test-form" action="" method="POST">
```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 97 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

        <?php $i = 1 ?>
        <?php foreach ($test_questions as
$question): ?>

            <div class="question-container">
                <div class="question-text
margin-top-bottom-small">

                    <?php echo $i++ . ". " .
htmlspecialchars($question['question_text']); ?>

                </div>
                <div>
                    <?php $answers =
Reading::getAnswersByQuestionId($question['id']);

                    if (!empty($answers)): ?>
                        <table>
                            <?php foreach
($answers as $answer): ?>

                                <tr>
                                    <td
style="width: 85%;"><?php echo htmlspecialchars($answer['answer_text']); ?></td>

                                    <td
style="text-align: center;">

                                        <input
t class="input-test-answer" type="radio" name="question<?php echo $question['id']; ?>"
value="<?php echo $answer['id']; ?>" data-correct="<?php echo $answer['is_correct']; ?>"
                                        </td>
                                </tr>
                            <?php endforeach;
                            ?></table>

                                <?php endif; ?>
                            </div></div>
                        <?php endforeach; ?>
                    <div id="test-results" class="margin-
top-bottom-small"></div>

                        <button id="check-answers-test"
class="btn" type="button" data-test="<?php echo $post['id']; ?>">Перевірити</button>
                        </form></div><?php endif; ?>
                    </div></div></div></div></div>
                <div class="col-lg-3 primary-sidebar sticky-sidebar pt-50">
                    <div class="widget-area">
                        <div class="sidebar-widget widget-category-2 mb-50">
                            <h5 class="section-title style-3 mb-20">Про пост</h5>
                            <div class="product-rate-cover margin-10">
                                <i class="fi fi-rs-school mr-10 text-brand"></i>
                                <span>
                                    <?php echo
Reading::getReadingCategoryId($post['subcategory_id']); ?>
                                </span></div>
                            <div class="product-rate-cover margin-10">
                                <i class="fi fi-rs-trophy mr-10 text-brand"></i>
                                <span>

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 98 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

        <?php echo
Site::getLanguageNameById($post['language']); ?>
    </span>
</div>
<div class="product-rate-cover margin-10">
    <i class="fi fi-rs-star mr-10 text-brand"></i>
    <span>
        <?php echo
Site::getLanguageLevelNameById($post['level']); ?>
    </span></div>
<div class="product-rate-cover margin-10">
    <i class="fi fi-rs-time-forward-ten mr-10 text-
brand"></i>
    <span>
        <?php $dateTime = DateTime::createFromFormat('Y-
m-d H:i:s', $post['creation_date']);
        $formattedDate = $dateTime->format('d/m/Y H:i');
        echo $formattedDate;?>
    </span>
</div>
<div class="product-rate-cover margin-10">
    <?php if(User::isGuest()): ?>
        <span class="font-small ml-5 fi-rs-thumbs-up" >
<?php echo Reading::countLikesByPostId($post['id'])?></span>
        <?php else: ?>
            <?php
if(!Reading::userLikedPost(User::getAuthUserId(), $post['id'])): ?>
                <span class="font-small ml-5 fi-rs-thumbs-up"
data-post-id="<?php echo $post['id']; ?>" data-user-id="<?php echo User::getAuthUserId(); ?>"
onclick="toggleHeartIcon(this, <?php echo User::getAuthUserId(); ?>, <?php echo $post['id'];
?>)"> <?php echo Reading::countLikesByPostId($post['id'])?></span>
            <?php else: ?>
                <span class="font-small ml-5 fi-rs-thumbs-up-
color" data-post-id="<?php echo $post['id']; ?>" data-user-id="<?php echo User::getAuthUserId();
?>" onclick="toggleHeartIcon(this, <?php echo User::getAuthUserId(); ?>, <?php echo $post['id'];
?>)"> <?php echo Reading::countLikesByPostId($post['id'])?></span>
            <?php endif; ?>
        <?php endif; ?></div></div></div></div></div>
    </div></div></div></main>
<?php require_once('views/layouts/footer.php'); ?>
<script>
function toggleHeartIcon(icon, user_id, post_id) {
    if (icon.classList.contains('fi-rs-thumbs-up')) {
        icon.classList.remove('fi-rs-thumbs-up');
        icon.classList.add('fi-rs-thumbs-up-color');
        $.ajax({ url:"../controllers/ajax/addReadingLike.php",
            method:"POST", data:{user_id:user_id, post_id:post_id},
            success:function(data){
                icon.innerHTML = " "+data;}}); } else {
        icon.classList.remove('fi-rs-thumbs-up-color'); icon.classList.add('fi-rs-thumbs-up');
        icon.innerHTML = "<?php echo Reading::countLikesByPostId($post['id']); ?>";
    }
}

```

| | | | | | | |
|------|------|----------|--------|------|---------------------------|------|
| | | | | | ІІЗ.КР.Б – 121 – 24 – ІІЗ | Арк. |
| | | | | | | 99 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

$.ajax({ url:"../controllers/ajax/removeReadingLike.php",
        method:"POST", data:{user_id:user_id, post_id:post_id},
        success:function(data){ icon.innerHTML = " "+data;}});}} </script><script>
document.getElementById('check-answers-test').addEventListener('click', function(event) {
    event.preventDefault(); let form = document.getElementById('test-form');
    let correctAnswers = 0; let totalQuestions = <?php echo count($test_questions); ?>;
    var userId = <?php echo $_SESSION['userLoggedIn'] ?>;var postId = <?php echo $post['id']; ?>;
    <?php foreach ($test_questions as $question): ?>
        var questionId = "<?php echo $question['id']; ?>";
        var selectedAnswer = form.querySelector('input[name="question" + questionId +
        ""]:checked');
        if (selectedAnswer) { let parentRow = selectedAnswer.closest('tr');
            if (selectedAnswer.getAttribute('data-correct') == '1') {correctAnswers++;
                parentRow.style.backgroundColor = 'lightgreen'; } else {
                parentRow.style.backgroundColor = 'lightcoral'; }}
        <?php endforeach; ?>
    let resultElem = document.getElementById('test-results');
    resultElem.innerHTML = "<i class='fi fi-rs-bell-ring mr-10 text-brand'></i> Ви відповіли
    правильно на " + correctAnswers + " з " + totalQuestions + " питань.";
    $.ajax({ url: "../controllers/ajax/insertAnswersReadingResults.php",
        method: "POST",data: { post_id: postId, total_questions: totalQuestions, correct_answers:
        correctAnswers, user_id: userId}, success: function(data) { }}));}};
</script>
<script src="/styles/frontend/js/vendor/modernizr-3.6.0.min.js"></script>
<script src="/styles/frontend/js/vendor/jquery-3.6.0.min.js"></script>
<script src="/styles/frontend/js/vendor/jquery-migrate-3.3.0.min.js"></script>
<script src="/styles/frontend/js/vendor/bootstrap.bundle.min.js"></script>
<script src="/styles/frontend/js/plugins/slick.js"></script>
<script src="/styles/frontend/js/plugins/jquery.syotimer.min.js"></script>
<script src="/styles/frontend/js/plugins/wow.js"></script>
<script src="/styles/frontend/js/plugins/slider-range.js"></script>
<script src="/styles/frontend/js/plugins/perfect-scrollbar.js"></script>
<script src="/styles/frontend/js/plugins/magnific-popup.js"></script>
<script src="/styles/frontend/js/plugins/select2.min.js"></script>
<script src="/styles/frontend/js/plugins/waypoints.js"></script>
<script src="/styles/frontend/js/plugins/counterup.js"></script>
<script src="/styles/frontend/js/plugins/jquery.countdown.min.js"></script>
<script src="/styles/frontend/js/plugins/images-loaded.js"></script>
<script src="/styles/frontend/js/plugins/isotope.js"></script>
<script src="/styles/frontend/js/plugins/scrollup.js"></script>
<script src="/styles/frontend/js/plugins/jquery.vticker-min.js"></script>
<script src="/styles/frontend/js/plugins/jquery.theia.sticky.js"></script>
<script src="/styles/frontend/js/plugins/jquery.elevatezoom.js"></script>
<script src="/styles/frontend/js/main.js?v=5.6"></script>
<script src="/styles/frontend/js/shop.js?v=5.6"></script>
</body></html>

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | ІПЗ.КР.Б – 121 – 24 – ПЗ | Арк. |
| | | | | | | 100 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |