

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка
Навчально-науковий інститут філології
Кафедра української мови та прикладної лінгвістики

Створення додатка для вивчення української мови на основі текстів пісень

Кваліфікаційна робота
на здобуття ОС «бакалавр»
студента IV курсу
галузі знань 03 «Гуманітарні науки»,
спеціальність 035 «Філологія»
спеціалізації 035.10 «Прикладна
лінгвістика », ОПП «Прикладна
(комп'ютерна) лінгвістика
та англійська мова»,
Пломодела Євгена Володимировича

Науковий керівник:
к.техн.н., доц. Микола КОСТИКОВ

«Допущено до захисту»
Протокол засідання кафедри
української мови та прикладної лінгвістики
від «5» 06 2026 року № 16

завідувач кафедри _____
к.філол.н., доц. Сергій РІЗНИК

КИЇВ - 2026

ЗМІСТ	2
Анотація	4
Вступ	6
1 Аналіз предметної області та сучасних рішень	8
1.1 Аналіз методів вивчення мови за допомогою цифрових технологій	8
1.2 Аналіз існуючих застосунків для вивчення мов та мовних корпусів	9
1.3 Виявлення проблем та постановка задачі дослідження	12
2 Проєктування інформаційної системи	15
2.1 Формування функціональних вимог до системи	15
2.2 Обґрунтування вибору технологій розробки	17
2.3 Проєктування архітектури та бази даних застосунку	20
3. Реалізація та тестування програмного продукту	25
3.1 Реалізація серверної логіки та бази даних корпусу пісень	25
3.2 Реалізація функціоналу навчання та користувацького інтерфейсу	27
3.3 Тестування системи та аналіз результатів	29
Висновки	43
Список використаних джерел	47
Додатки	49

АНОТАЦІЯ

У дипломній роботі використано можливості застосування корпусних технологій для створення сучасної програми вивчення української мови. Приділено увагу використанню текстів українських пісень як джерела мовного матеріалу. Проведено аналіз сучасних підходів до цифрового навчання мов, принципів побудови корпусів текстів та особливостей їх інтеграції в освітні програмні системи.

У роботі створено концепцію програмного застосунку для вивчення української мови з використанням корпусного аналізу, автоматизованого пошуку мовних одиниць і формування навчальних завдань на основі реального мовного матеріалу. Наукова новизна дослідження полягає у використанні структурованого корпусу текстів українських пісень як основи для персоналізованого навчання. Запропоновано модель зберігання та обробки текстових даних із використанням реляційної бази даних. Це забезпечує ефективне опрацювання великих обсягів мовного матеріалу.

Практичним результатом роботи стало створення програмного застосунку, реалізованого засобами мови програмування Python та системи керування базами даних PostgreSQL. Розроблена система забезпечує пошук і аналіз текстового матеріалу, створення інтерактивних навчальних вправ, контроль результатів навчання та накопичення статистичних даних про діяльність користувачів. Проведене тестування підтвердило працездатність програмного продукту та доцільність використання корпусу українських пісень як навчального ресурсу для розвитку мовних компетентностей.

Ключові слова: програмний застосунок, корпус текстів, українські пісні, цифрове навчання мов, обробка природної мови, Python, PostgreSQL, навчальна система.

ABSTRACT

The thesis utilizes the possibilities of applying corpus technologies to the development of a modern application for learning the Ukrainian language. Particular attention is paid to the use of Ukrainian song lyrics as a source of language material. An analysis of contemporary approaches to digital language learning, the principles of text corpus construction, and the specifics of their integration into educational software systems was conducted.

The study presents the concept of a software application for learning the Ukrainian language using corpus analysis, automated retrieval of linguistic units, and the generation of educational tasks based on authentic language material. The scientific novelty of the research lies in the use of a structured corpus of Ukrainian song lyrics as a foundation for personalized learning. A model for storing and processing textual data using a relational database is proposed. This approach ensures the efficient processing of large volumes of language material.

The practical outcome of the research is the development of a software application implemented using the Python programming language and the PostgreSQL database management system. The developed system provides text search and analysis, the creation of interactive learning exercises, monitoring of learning outcomes, and the accumulation of statistical data on user activity. The conducted testing confirmed the functionality of the software product and demonstrated the feasibility of using a corpus of Ukrainian song lyrics as an educational resource for the development of language competencies.

Keywords: software application, text corpus, Ukrainian song lyrics, digital language learning, natural language processing, Python, PostgreSQL, educational system.

ВСТУП

У сучасних умовах цифрової трансформації освіти особливої актуальності набуває розробка програмних засобів, що сприяють ефективному вивченню мов. Використання інформаційних технологій у мовній освіті дозволяє підвищити доступність навчальних матеріалів, персоналізувати навчальний процес, забезпечити інтерактивну взаємодію з користувачем та підвищити мотивацію до навчання. Одним із перспективних напрямів є застосування пісенних текстових матеріалів, які поєднують емоційне сприйняття, культурний контекст та природне мовне середовище.

Українські пісні є цінним джерелом лексичних, граматичних і стилістичних конструкцій, що можуть бути використані для формування мовної компетентності. Використання корпусу текстів українських пісень у навчальному застосунку дозволяє поєднати освітню складову з культурним контекстом, зробити процес навчання більш цікавим та наближеним до реального мовного середовища. Проте більшість сучасних мовних платформ не орієнтовані на український мовний контент або не використовують спеціалізовані корпуси текстів для адаптивного навчання.

Актуальність теми дипломної роботи полягає у необхідності створення сучасного програмного застосунку для вивчення української мови з використанням структурованого корпусу текстів українських пісень, реалізованого на базі реляційної бази даних. Такий підхід дозволяє забезпечити ефективний пошук, аналіз мовного матеріалу, формування навчальних завдань та відстеження прогресу користувачів.

У демонстраційному наповненні системи передбачено генератор пісень, який використовується для формування навчального текстового матеріалу. Навчальний словник, сформований після автоматичної обробки текстів, охоплює 324 лексичні одиниці. Результати роботи генератора та подальшої лематизації можуть потребувати ручної верифікації.

Метою дипломної роботи є розробка програмного застосунку для вивчення

української мови на основі корпусу текстів українських пісень із використанням мови програмування Python та системи керування базами даних PostgreSQL.

Для досягнення поставленої мети необхідно виконати такі завдання:

- проаналізувати сучасні методи цифрового вивчення мов та існуючі програмні рішення;
- дослідити можливості використання текстових корпусів у навчальних системах;
- обґрунтувати вибір технологій розробки;
- спроектувати архітектуру програмного застосунку;
- розробити структуру бази даних для зберігання корпусу українських пісень;
- реалізувати серверну логіку, модулі аналізу тексту та навчальний функціонал;
- провести тестування розробленої системи та оцінити її ефективність.

Об'єктом дослідження є процес цифрового вивчення української мови.

Предметом дослідження є методи, засоби та програмні технології створення застосунку для вивчення мови на основі корпусу текстів українських пісень.

Практичне значення роботи полягає у створенні програмного продукту, який може бути використаний як допоміжний інструмент для самостійного вивчення української мови, розширення словникового запасу та розвитку мовних навичок.

Структура дипломної роботи складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків. У першому розділі розглянуто предметну область та існуючі рішення, у другому - описано проектування системи, у третьому - реалізацію та результати тестування програмного застосунку.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА СУЧАСНИХ РІШЕНЬ

1.1 Аналіз методів вивчення мови за допомогою цифрових технологій

Сучасний розвиток цифрових технологій суттєво трансформував підходи до вивчення мов. Традиційні методи навчання, засновані на друкованих матеріалах та аудиторних заняттях, поступово доповнюються електронними засобами навчання, які забезпечують гнучкість, доступність та індивідуалізацію освітнього процесу. Цифрові технології дозволяють організувати навчання у зручному форматі, адаптувати складність матеріалу до рівня користувача та забезпечити постійний доступ до навчальних ресурсів [3, с. 45-61; 4, с. 38-49].

Одним із найбільш поширених методів цифрового навчання є використання інтерактивних навчальних систем. Такі системи дозволяють реалізувати вправи на засвоєння нової лексики, розвиток навичок читання, аудіювання, письма та контролю знань. Важливою перевагою інтерактивних платформ є наявність миттєвого зворотного зв'язку, автоматизованої перевірки відповідей та можливості повторення матеріалу. Це сприяє більш ефективному засвоєнню мовного матеріалу та формуванню стійких мовних навичок [5; 6].

Суттєву роль у цифровому вивченні мови відіграє мультимедійний підхід, який поєднує текстову, аудіо- та візуальну інформацію. Застосування мультимедійних і мобільних засобів сприяє кращому запам'ятовуванню нових слів, розвитку асоціативного мислення та підвищенню зацікавленості користувача [7; 8]. Особливо ефективним є використання матеріалів, що відображають природний мовний контекст, оскільки це дозволяє засвоювати лексику не ізольовано, а в ситуаціях її реального або навчально змодельованого використання; пісенний матеріал додатково посилює запам'ятовування завдяки ритму та мелодії [9].

Окремим перспективним напрямом є корпусний підхід до вивчення мови. Він передбачає використання структурованих текстових баз даних, які містять мовний матеріал, систематизований за певними ознаками [10; 11]. Корпусний підхід дозволяє аналізувати частотність уживання слів, контекст їх використання, типові мовні конструкції та стилістичні особливості. Це створює додаткові можливості

для формування навчальних вправ, адаптації контенту до рівня користувача та персоналізації процесу навчання [12; 13].

Таким чином, сучасні цифрові методи вивчення мови забезпечують ефективне поєднання інтерактивності, мультимедійності та аналітичних можливостей. Їх використання створює передумови для розробки сучасних програмних застосунків, здатних підвищити ефективність навчання та зробити процес засвоєння мовного матеріалу більш зручним і результативним.

1.2 Аналіз існуючих застосунків для вивчення мов та мовних корпусів

На сучасному етапі розвитку цифрової освіти існує значна кількість програмних застосунків, орієнтованих на вивчення іноземних мов. Більшість таких систем використовують інтерактивний підхід, поєднуючи теоретичний матеріал, практичні вправи, тести, систему повторення та відстеження прогресу користувача. Аналіз існуючих рішень дозволяє визначити їх сильні сторони, недоліки та сформулювати вимоги до власного програмного продукту.

Одним із найпоширеніших застосунків для вивчення мов є Duolingo. Даний сервіс реалізує навчання у формі коротких інтерактивних вправ, тестів, аудіозавдань та елементів гейміфікації. Система дозволяє відстежувати прогрес користувача, підтримує адаптацію складності завдань та має зручний інтерфейс. Основною перевагою є висока залученість користувачів, однак недоліком є обмежене використання автентичного контенту та шаблонність частини вправ [14].



Рисунок 1.1 – Логотип Duolingo

Ще одним популярним рішенням є LingQ, який орієнтований на навчання через читання та роботу з автентичними текстами. Застосунок дозволяє користувачу читати матеріали, виділяти нові слова, формувати словникові списки та повторювати лексику. Перевагою є можливість працювати з реальними текстами, однак функціонал сервісу частково обмежений платною підпискою [15].

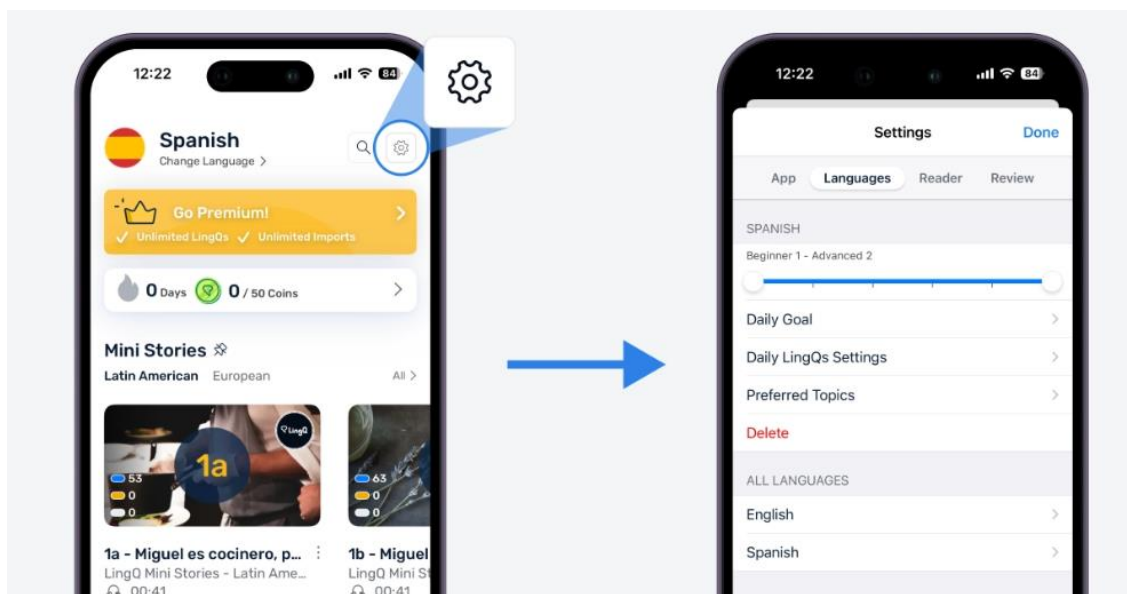


Рисунок 1.2 – Інтерфейс LingQ

Серед систем, що використовують музичний та мультимедійний контент, варто відзначити LyricsTraining. Даний сервіс пропонує навчання на основі популярних музичних композицій, поєднуючи прослуховування аудіо- або відеоматеріалу з інтерактивними завданнями. Основний принцип роботи полягає у відтворенні пісні та заповненні користувачем пропущених слів у тексті, що відображається на екрані. Такий формат дозволяє не лише покращувати навички аудіювання, а й розвивати швидкість сприйняття мовлення, розширювати словниковий запас та закріплювати знання граматичних конструкцій у реальному мовному контексті.

Проте система має певні обмеження. Зокрема, сервіс орієнтований переважно на популярні міжнародні мови, а підтримка українського мовного контенту є обмеженою. Також LyricsTraining не забезпечує повноцінного корпусного аналізу

текстів, не дозволяє виконувати глибокий пошук лексичних одиниць, аналіз частоти вживання слів чи формувати персоналізовані навчальні добірки на основі текстової бази. Це обмежує можливості системи як комплексного інструменту для системного вивчення мови [16].

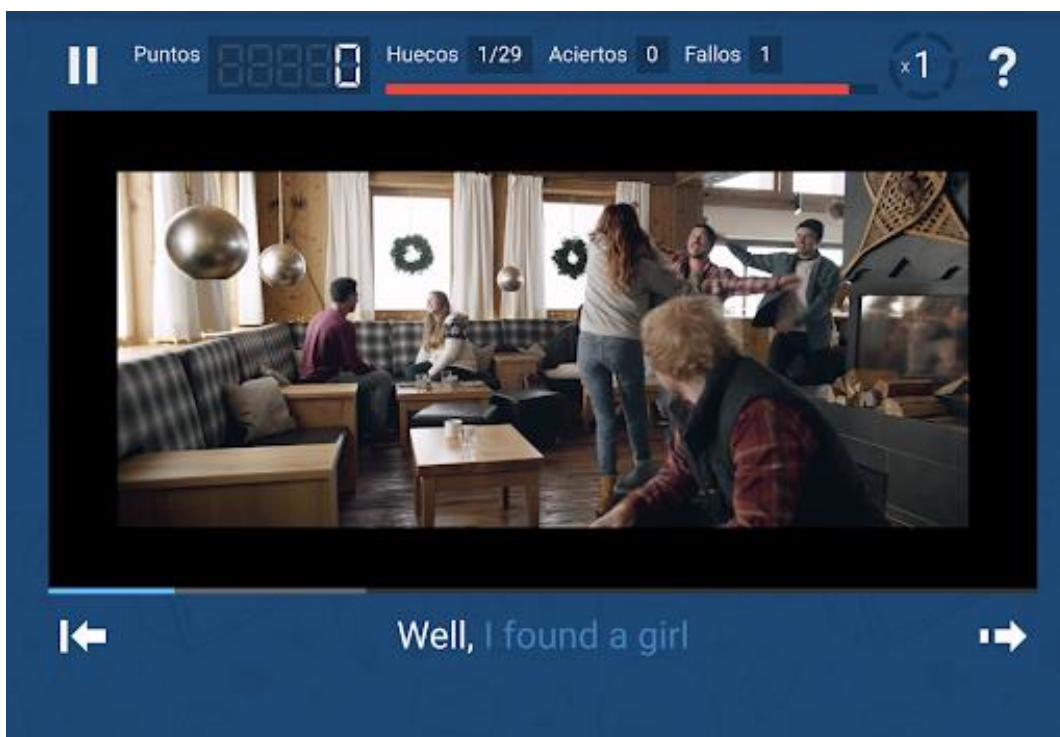


Рисунок 1.3 – Інтерфейс LyricsTraining

Окремої уваги заслуговують мовні корпуси як інструмент аналізу текстів. Корпусні системи дозволяють здійснювати пошук слів, аналіз частоти вживання, виявлення мовних конструкцій та контекстів використання. Такі рішення активно застосовуються у лінгвістичних дослідженнях, однак у більшості випадків не адаптовані під потреби навчальних застосунків для широкого кола [10; 17].

Проведений аналіз показав, що сучасні мовні застосунки мають значний функціонал, проте більшість із них не враховують особливості українського мовного контенту, не використовують спеціалізовані корпуси українських пісень та не поєднують мовне навчання з культурним контекстом. Це підтверджує доцільність розробки власного програмного застосунку, який поєднуватиме переваги інтерактивного навчання, мультимедійного контенту та корпусного аналізу текстів.

1.3 Виявлення проблем та постановка задачі дослідження

Проведений аналіз сучасних цифрових методів вивчення мови та існуючих програмних рішень показав, що, незважаючи на значний розвиток освітніх технологій, існує низка проблем, які знижують ефективність мовного навчання. Більшість сучасних застосунків орієнтовані на універсальні методики навчання, що не завжди враховують мовні, культурні та індивідуальні особливості користувачів. У результаті навчальний процес часто стає формальним, недостатньо гнучким і не забезпечує глибокого засвоєння матеріалу.

Однією з основних проблем є недостатня адаптація існуючих платформ до українського мовного середовища. Більшість популярних сервісів зосереджені на вивченні англійської та інших поширених мов, тоді як україномовний контент представлений обмежено. Це зменшує можливість використання рідномовного матеріалу як бази для навчання та ускладнює процес залучення користувачів до вивчення української мови. Крім того, навіть у разі наявності українського інтерфейсу, зміст навчальних матеріалів часто не враховує специфіку мовного середовища та культурного контексту.

Іншою суттєвою проблемою є недостатнє використання автентичних навчальних матеріалів, які відображають реальне мовне середовище. Значна частина сучасних систем використовує штучно створені тексти, стандартні вправи та універсальні сценарії навчання. Такий підхід не забезпечує достатнього рівня залученості користувача, не формує практичних навичок роботи з живою мовою та не сприяє розвитку мовного чуття.

Окремим недоліком сучасних рішень є обмежене використання корпусного підходу. Більшість освітніх платформ не мають структурованих текстових баз даних, які дозволяли б аналізувати лексику, частоту вживання слів, типові мовні конструкції та створювати персоналізовані вправи на основі реального мовного матеріалу. Відсутність корпусу як основи навчання обмежує можливості системи щодо адаптації контенту під рівень користувача та ускладнює побудову індивідуальної траєкторії навчання.

Ще однією проблемою є недостатній рівень персоналізації навчального

процесу. Більшість існуючих застосунків пропонують однаковий набір вправ для всіх користувачів, не враховуючи їхні інтереси, рівень підготовки та темп засвоєння матеріалу. Це призводить до зниження ефективності навчання та втрати мотивації.

Також варто відзначити недостатню інтеграцію культурного контенту в процес мовного навчання. Використання текстів українських пісень як навчального матеріалу має значний потенціал, оскільки поєднує мовну практику, емоційне сприйняття та ознайомлення з національною культурою. Пісенні тексти можуть підтримувати запам'ятовування фраз і лексики, однак у демонстраційних системах згенерований пісенний матеріал потребує додаткової редакторської та лінгвістичної перевірки [9]. Проте подібний підхід практично не реалізований у сучасних цифрових застосунках.

Для більш наочного узагальнення виявлених проблем доцільно подати їх у вигляді таблиці.

Таблиця 1.1 – Основні проблеми сучасних систем вивчення мов

№	Проблема	Опис	Наслідки
1	Відсутність україномовного контенту	Обмежена кількість матеріалів українською мовою	Зниження доступності навчання
2	Штучні навчальні матеріали	Використання неавтентичних текстів	Відсутність реального мовного контексту
3	Відсутність корпусного підходу	Немає структурованої бази текстів	Неможливість аналізу лексики
4	Низька персоналізація	Однакові вправи для всіх користувачів	Зниження ефективності навчання
5	Слабка інтеграція культури	Відсутність культурного контенту	Низька мотивація користувача
6	Обмежений аналіз текстів	Відсутність статистики та аналітики	Складність побудови навчальних сценаріїв

У зв'язку з виявленими проблемами постає необхідність розробки програмного застосунку для вивчення української мови, який поєднуватиме сучасні підходи до організації навчального процесу та використання цифрових технологій.

Такий застосунок повинен забезпечувати:

- створення структурованого корпусу текстів українських пісень у базі даних;
- можливість пошуку, фільтрації та аналізу текстового матеріалу;
- інтерактивну взаємодію користувача з текстом (зокрема роботу зі словами);
- формування навчальних вправ на основі реальних текстів;
- персоналізацію навчального процесу;
- збереження та відображення прогресу користувача;
- інтеграцію культурного контенту у процес навчання.

Таким чином, основним завданням дипломної роботи є проєктування та реалізація сучасного програмного застосунку для вивчення української мови на основі корпусу текстів українських пісень із використанням Python та PostgreSQL. Реалізація такого підходу дозволить підвищити ефективність навчального процесу, зробити його більш цікавим, доступним та адаптивним до потреб користувача, а також сприятиме популяризації української мови та культури у цифровому середовищі.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Формування функціональних вимог до системи

На етапі проєктування програмного застосунку було сформовано перелік функціональних вимог, які визначають основні можливості системи та логіку її роботи. Формування вимог здійснювалося на основі аналізу предметної області, особливостей цифрового вивчення мови, а також з урахуванням поставленої мети дипломного проєкту - створення інформаційно-навчальної системи для роботи з корпусом українськомовних пісенних текстів. Функціональні вимоги визначають набір сервісів, які повинні бути реалізовані в системі незалежно від конкретного технологічного стеку [18; 19].

Першим ключовим компонентом системи є модуль корпусу пісень. Даний модуль повинен забезпечувати повний цикл роботи з текстовим корпусом: додавання нових пісень до бази даних, редагування наявних записів, видалення застарілих або помилкових записів, а також відображення загального списку пісень. Крім цього, система повинна надавати можливість перегляду повного тексту пісні, здійснювати пошук за назвою, виконавцем, жанром, роком та окремими словами тексту. Це забезпечить користувачу зручний доступ до мовного матеріалу та створить основу для подальшого навчального використання.

Другим важливим компонентом є модуль лінгвістичного аналізу текстів. Система повинна підтримувати базові операції текстової обробки: токенизацію тексту, нормалізацію словоформ, підрахунок частоти вживання слів, виявлення стоп-слів та ключових лексичних одиниць. Додатково передбачається можливість пошуку повторюваних слів і словосполучень, а також збереження результатів аналізу в базі даних. Це дозволить використовувати мовний матеріал не лише як текстову базу, а і як джерело для побудови навчальних вправ та аналітики.

Третім функціональним блоком є навчальний модуль. Він повинен забезпечувати автоматичне формування списку нових слів на основі тексту обраної пісні, генерацію навчальних карток для запам'ятовування лексики, створення вправ типу «встав пропущене слово», а також вправ на співставлення слова та його

значення. Система має надавати можливість позначати слова як вивчені, зберігати результати проходження вправ та відображати індивідуальний прогрес користувача. Реалізація такого функціоналу підвищить ефективність навчального процесу та забезпечить персоналізацію навчання.

Окремо було визначено вимоги до модуля користувачів. Система повинна підтримувати реєстрацію нових користувачів, автентифікацію та вхід до особистого кабінету. У межах профілю користувача має бути доступна інформація про історію проходження вправ, кількість вивчених слів, успішність виконання завдань та загальний прогрес навчання. Це дозволить зробити навчальний процес більш структурованим і контрольованим.

Важливою складовою функціональних вимог є організація сховища даних. Система має забезпечувати структуроване зберігання пісенного матеріалу, підтримувати зв'язки між сутностями (пісні, виконавці, жанри, слова, вправи, користувачі), зберігати результати аналізу текстів і навчальний прогрес. Після нормалізації текстів та вилучення службових одиниць у навчальному словнику системи має бути доступно 324 лексичні одиниці для вивчення. Така структура відповідає загальним принципам проєктування інформаційних систем і баз даних [20].

Крім того, система повинна мати зручний вебінтерфейс та програмний інтерфейс для обміну даними між модулями. Інтерфейс має забезпечувати навігацію між каталогом пісень, сторінками аналізу текстів, навчальними вправами та особистим кабінетом користувача. Загалом система повинна працювати локально, бути придатною для демонстрації під час захисту дипломної роботи, мати модульну архітектуру та підтримувати можливість подальшого розширення функціоналу.

Таким чином, сформовані функціональні вимоги визначають основу майбутньої системи та забезпечують реалізацію повноцінного програмного продукту, що поєднує можливості корпусного сховища, лінгвістичного аналізу та інтерактивного навчання української мови.

2.2 Обґрунтування вибору технологій розробки

Після формування функціональних вимог було обрано сучасний стек технологій, який забезпечує зручність розробки, достатню продуктивність, надійність зберігання даних та можливість подальшого розширення функціоналу системи. Вибір технологій здійснювався з урахуванням специфіки проекту: роботи зі структурованим корпусом пісенного матеріалу, генератором пісень, навчальним функціоналом, текстовим аналізом та вебінтерфейсом для взаємодії з користувачем.

Основною мовою програмування для створення системи обрано Python. Дана мова є однією з найпоширеніших для розробки прикладних програмних рішень, вебзастосунків та систем обробки даних. Її перевагами є зрозумілий синтаксис, велика кількість бібліотек, підтримка різних стилів програмування та зручність інтеграції з інструментами для роботи з базами даних і текстовою інформацією. В офіційній документації Python зазначено, що мова підтримує ефективні структури даних високого рівня та є придатною для швидкої розробки застосунків. Це робить її доцільним вибором для реалізації інформаційно-навчальної системи лінгвістичного спрямування [21].



Рисунок 2.1 – Логотип Python

Для побудови серверної частини застосунку було обрано фреймворк FastAPI. Він позиціонується як сучасний високопродуктивний вебфреймворк для створення

API на основі стандартних підказок типів Python. Його використання дозволяє реалізувати структуровану серверну архітектуру, забезпечити модульність програмного коду, автоматичне формування OpenAPI-документації та зручну роботу з HTTP-запитами. REST API найкращим чином забезпечує вимогу програмного обміну даними між каталогом пісень, модулем аналізу, навчальним модулем та вебінтерфейсом [22].



Рисунок 2.2 – Логотип FastAPI

Для організації взаємодії між програмним кодом та базою даних було обрано SQLAlchemy 2.0. Це SQL toolkit та ORM, що дозволяє працювати з реляційними базами даних через об'єктну модель. Використання ORM-рівня спрощує підтримку коду, дозволяє зменшити кількість дубльованих SQL-запитів і полегшує опис складних зв'язків між сутностями системи. Для даного дипломного проєкту це особливо важливо, оскільки структура бази даних включає пов'язані таблиці пісень, виконавців, слів, користувачів, вправ і результатів навчання. Обрана ORM-бібліотека відповідає вимозі підтримки складних зв'язків між сутностями та спрощення CRUD-операцій [23].



Рисунок 2.3 – Логотип SQLAlchemy

Як систему керування базами даних було обрано PostgreSQL. Це реляційна СКБД, яка підтримує транзакційність, зовнішні ключі, індексацію, широкий набір типів даних та повноцінну роботу зі складними схемами. PostgreSQL дозволить реалізувати вимогу структурованого зберігання пісенного матеріалу, пов'язаних сутностей і словника з 324 лексичних одиниць, а також забезпечить швидкий пошук, фільтрацію, цілісність даних та можливість подальшого аналітичного опрацювання [24].



Рисунок 2.4 – Логотип PostgreSQL

Для формування вебсторінок у межах серверного підходу доцільно використати шаблонізатор Jinja2. Його застосування дозволяє генерувати HTML-сторінки на стороні сервера без необхідності створення окремого складного клієнтського застосунку. Jinja2 дозволить реалізувати вимогу зручного вебінтерфейсу для каталогу пісень, сторінки аналізу тексту, вправ та особистого кабінету при відносно простій архітектурі системи [25].



Рисунок 2.5 – Логотип Jinja2

Для формування статистики у дашборді було використано бібліотеку Chart.js. Застосування даної бібліотеки дозволило реалізувати вимогу наочного відображення прогресу користувача, результатів вправ і стану словника без необхідності використання складних інструментів візуалізації [26].



Рисунок 2.6 – Логотип chart.js

Отже, вибраний стек технологій є обґрунтованим як з технічної, так і з практичної точки зору. Він дозволяє реалізувати повноцінний програмний застосунок для вивчення української мови на основі корпусу текстів українських пісень, забезпечити зручну роботу з базою даних, підтримку навчального функціоналу та можливість подальшого розвитку системи [23].

2.3 Проектування архітектури та бази даних застосунку

На етапі проектування інформаційної системи було сформовано загальну архітектуру програмного застосунку та логічну структуру бази даних. Основною метою даного етапу є побудова такої структури системи, яка забезпечить зручну взаємодію між користувачем, серверною логікою, модулями аналізу тексту та базою даних корпусу пісень. Архітектура застосунку повинна бути модульною, масштабованою та придатною для подальшого розвитку [19; 27].

Для реалізації дипломного проєкту було обрано клієнт-серверну архітектуру. У межах даного підходу користувач взаємодіє із системою через вебінтерфейс, який надсилає запити до серверної частини. Серверний застосунок обробляє

запити, виконує бізнес-логіку, взаємодіє з базою даних, модулем лінгвістичного аналізу та навчальним модулем, після чого повертає результат користувачу. Такий підхід дозволяє чітко розділити рівні відповідальності системи, спростити підтримку коду та забезпечити зручне тестування окремих компонентів.

До основних компонентів архітектури системи належать:

- вебінтерфейс користувача;
- серверна частина застосунку;
- модуль роботи з корпусом пісень;
- модуль лінгвістичного аналізу;
- навчальний модуль;
- модуль автентифікації;
- база даних.

Вебінтерфейс забезпечує взаємодію користувача з системою: перегляд каталогу пісень, запуск аналізу тексту, проходження вправ та перегляд статистики. Серверна частина реалізує обробку запитів, маршрутизацію, логіку роботи модулів та взаємодію з базою даних. Модуль корпусу відповідає за зберігання текстів пісень, виконавців та жанрів. Модуль аналізу виконує обробку тексту та формує аналітичні дані. Навчальний модуль генерує вправи та відстежує прогрес користувача.

Для більш наочного представлення структури системи доцільно подати схему архітектури, яка демонструє взаємозв'язок між клієнтською частиною, сервером, API, модулями аналізу та базою даних.

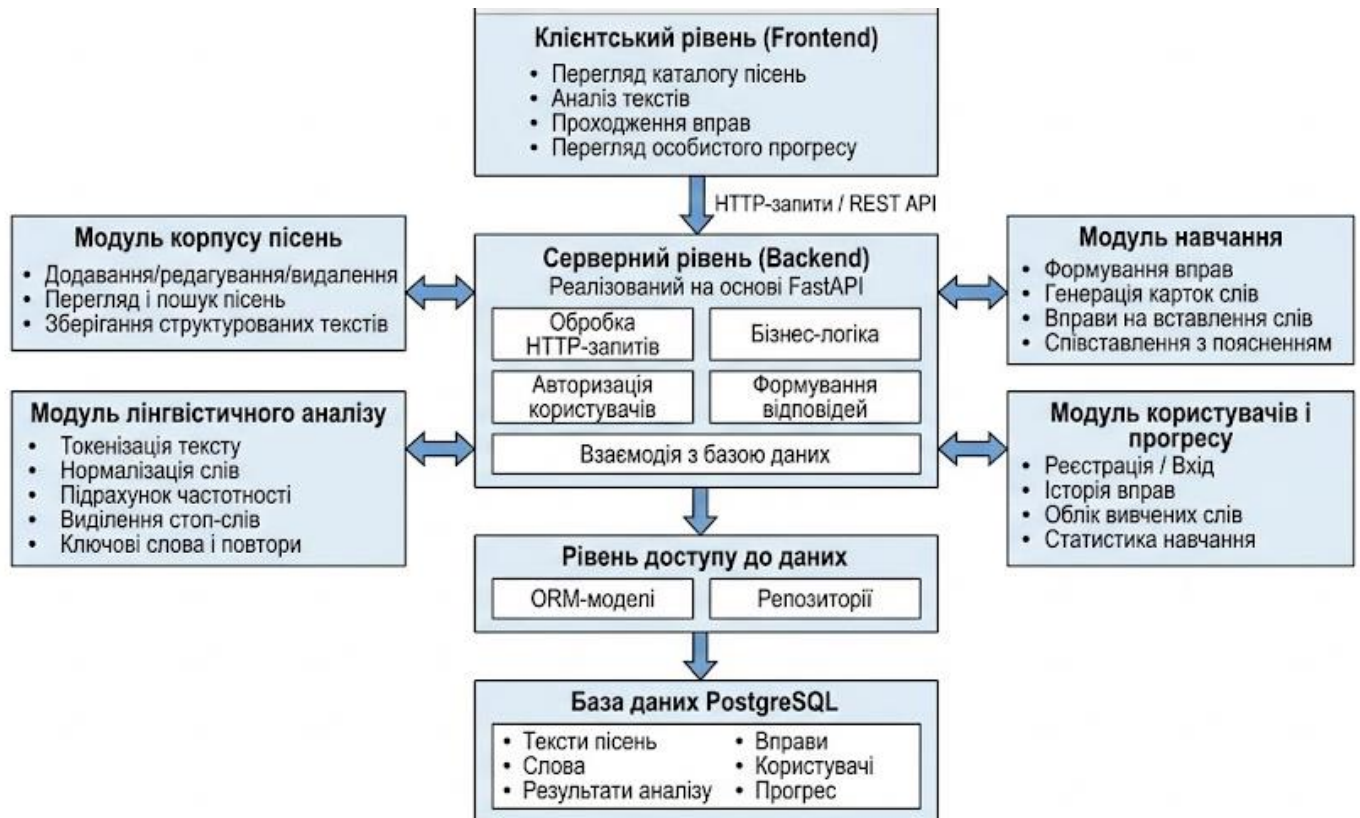


Рисунок 2.7 – Загальна архітектура програмного застосунку

Проектування бази даних є важливим етапом створення системи, оскільки саме база даних забезпечує зберігання корпусу текстів українських пісень, результатів аналізу та навчального прогресу користувачів. Для даного проєкту обрано реляційну модель даних, реалізовану на базі PostgreSQL. Вибір реляційної моделі узгоджується з вимогами до цілісності, нормалізації та зв'язності даних [20; 27].

Основними сутностями бази даних є:

- користувачі;
- пісні;
- виконавці;
- жанри;
- тексти пісень;
- слова;
- результати аналізу;
- вправи;

– прогрес користувачів.

Для забезпечення цілісності даних між таблицями передбачено зв'язки типу «один до багатьох» та «багато до багатьох». Наприклад, одна пісня може містити багато слів, а одне слово може зустрічатися у багатьох піснях. Користувач може проходити багато вправ, а одна вправа може бути доступна різним користувачам. Така структура дозволяє уникнути дублювання інформації та забезпечує гнучкість при роботі з корпусом.

У таблиці 2.1 доцільно подати перелік основних сутностей бази даних, їх призначення та ключові поля.

Таблиця 2.1 – Основні сутності бази даних застосунку

Сутність	Призначення	Основні поля	Основні зв'язки
users	Зберігання облікових записів користувачів	id, email, username, full_name, password_hash	1:M з user_progress, user_exercise_attempts, user_known_words
artists	Зберігання даних про виконавців, авторів, композиторів	id, name, country	M:N з songs через song_artists
genres	Зберігання жанрів пісень	id, name, description	1:M з songs
songs	Основна таблиця корпусу пісень	id, title, year, genre_id, educational_summary, difficulty_level	M:1 з genres, 1:M з song_lines, M:N з artists, M:N з words
song_artists	Проміжна таблиця зв'язку пісень і виконавців/авторів	id, song_id, artist_id, role	M:1 до songs, M:1 до artists
song_lines	Построкове зберігання тексту пісні	id, song_id, line_number, text	M:1 до songs

Продовження таблиці 2.1

words	Словник лексичних одиниць корпусу	id, token, lemma, part_of_speech, is_stop_word	M:N з songs через song_words, M:N з users через user_known_words
song_words	Зв'язок між піснями та словами з частотними характеристиками	id, song_id, word_id, frequency, relative_frequency	M:1 до songs, M:1 до words
analysis_results	Результати лінгвістичного аналізу пісень	id, song_id, total_tokens, unique_tokens, top_keywords, summary	M:1 до songs
exercises	Навчальні вправи, сформовані на основі пісень	id, source_song_id, exercise_type, prompt, payload, correct_answer	M:1 до songs, M:1 до users
user_exercise_attempts	Історія виконання вправ користувачами	id, user_id, exercise_id, user_answer, is_correct, score	M:1 до users, M:1 до exercises

Для візуального представлення структури бази даних доцільно використати ER-діаграму, яка відображає таблиці, атрибути та зв'язки між ними. Вона приведена в Додатку А.

Таким чином, спроектована архітектура системи та структура бази даних забезпечують реалізацію всіх функціональних вимог дипломного проєкту. Запропонована модель дозволяє ефективно працювати з корпусом текстів українських пісень, реалізувати навчальний функціонал, виконувати текстовий аналіз та забезпечувати зручну взаємодію користувача з системою.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ

3.1 Реалізація серверної логіки та бази даних корпусу пісень

Практична реалізація дипломного проєкту передбачала створення серверної частини програмного застосунку, яка забезпечує основну бізнес-логіку системи, взаємодію між користувачем і базою даних, обробку текстового матеріалу та виконання навчальних сценаріїв. Серверна частина була реалізована з використанням FastAPI та мови програмування Python, що дозволило побудувати сучасну, модульну та зручну для підтримки архітектуру програмного продукту [22; 27].

Розробка серверної логіки здійснювалася за принципом розподілу функціональності на окремі логічні модулі. Такий підхід дозволив чітко розмежувати відповідальність між компонентами системи та забезпечити зручність подальшого тестування і супроводу. Основу серверної частини становлять модулі маршрутизації, обробки запитів, роботи з корпусом текстів, автентифікації користувачів, текстового аналізу та навчального функціоналу.

Маршрутизація запитів була реалізована через систему HTTP-ендпоінтів, які відповідають за доступ до основних функцій застосунку. Зокрема, створено маршрути для перегляду списку пісень, отримання повного тексту композиції, пошуку за назвою або словами тексту, запуску аналізу тексту, формування вправ, реєстрації та авторизації користувача. Така організація забезпечує зручну взаємодію між вебінтерфейсом і серверною частиною системи.

Окрему увагу було приділено реалізації модуля роботи з корпусом пісень. У межах системи реалізовано механізми додавання нових записів до бази даних, оновлення наявної інформації, видалення записів та пошуку за різними параметрами. Для кожної пісні передбачено збереження таких атрибутів, як назва, виконавець, жанр, рік випуску та текстовий фрагмент. Це дозволяє сформувати структурований корпус мовного матеріалу, придатний для навчального використання.

Для реалізації лінгвістичного функціоналу було створено окремий модуль аналізу тексту. Його основними завданнями є розбиття тексту на окремі слова, нормалізація слів, підрахунок частоти вживання лексичних одиниць та виділення ключових слів. Результати аналізу можуть використовуватися як для побудови навчальних вправ, так і для надання користувачу додаткової інформації щодо лексичних особливостей пісні.

Навчальний модуль реалізовано як окремий компонент серверної логіки, який взаємодіє з корпусом пісень і результатами текстового аналізу. На основі тексту композицій система формує список нових слів, створює вправи на заповнення пропусків, словникові картки та завдання на перевірку знань. Такий підхід дозволяє зробити процес навчання інтерактивним та адаптивним до індивідуальних потреб користувача.

Важливим елементом серверної логіки є модуль автентифікації та управління користувачами. У системі реалізовано реєстрацію нових користувачів, механізм входу в особистий кабінет, збереження історії виконаних вправ та відображення прогресу навчання. Для захисту облікових записів використовується безпечне зберігання паролів у зашифрованому вигляді, що підвищує рівень надійності системи.

Для зберігання даних було використано PostgreSQL. Застосування реляційної бази даних дозволило реалізувати структуроване зберігання корпусу текстів українських пісень, результатів аналізу, вправ та даних користувачів. На відміну від текстових файлів, база даних забезпечує швидкий пошук, підтримку зв'язків між таблицями, цілісність інформації та можливість подальшого масштабування системи. Технічні можливості PostgreSQL безпосередньо відповідають цим вимогам [24].

Для взаємодії між серверною логікою та базою даних було використано ORM-бібліотеку SQLAlchemy. Це дозволило побудувати об'єктну модель сутностей системи та спростити реалізацію CRUD-операцій. Використання ORM підвищило читабельність коду, зменшило ризик помилок при роботі з SQL-запитами та забезпечило гнучкість у внесенні змін до структури проєкту.

Використання SQLAlchemy забезпечує узгодженість між моделями предметної області та схемою бази даних [23].

У межах реалізації також було виконано первинне наповнення та перевірку бази даних корпусу українськомовних пісенних текстів. У системі передбачено генератор пісень, який використовується для формування навчального текстового матеріалу, а після токенізації, нормалізації та вилучення службових одиниць сформовано словник із 324 лексичних одиниць, доступних для вивчення. Такі дані не можна вважати повністю верифікованим корпусом: можливі неточності у формулюваннях, повтори, некоректні леми або помилки перекладу, що потребує ручної перевірки та подальшого доопрацювання.

Таким чином, реалізована серверна логіка та структура бази даних забезпечують стабільну роботу програмного застосунку, підтримку всіх основних функціональних модулів та можливість подальшого розширення системи відповідно до поставлених завдань дипломного проєкту.

3.2 Реалізація функціоналу навчання та користувацького інтерфейсу

Після реалізації серверної логіки та бази даних наступним етапом практичної частини дипломного проєкту стала розробка функціоналу навчання та користувацького інтерфейсу застосунку. Основною метою даного етапу було створення зручного механізму взаємодії користувача з корпусом українськомовних пісенних текстів, а також реалізація навчальних сценаріїв, що дозволяють ефективно засвоювати нову лексику та мовні конструкції [1; 28].

Користувацький інтерфейс застосунку реалізовано у вигляді вебсистеми, що забезпечує доступ до всіх основних функцій через браузер. Такий підхід є доцільним для дипломного проєкту, оскільки дозволяє спростити демонстрацію програмного продукту, забезпечити зручну навігацію між модулями та зробити систему доступною без додаткового встановлення клієнтського програмного забезпечення.

Основною точкою взаємодії користувача із системою є каталог пісень. На головній сторінці користувач отримує доступ до списку композицій, доступних у

корпусі. Для кожної пісні відображаються базові метадані: назва, виконавець, жанр, рік випуску та короткий опис. Додатково реалізовано функцію пошуку та фільтрації, що дозволяє швидко знаходити пісні за назвою, автором або ключовими словами тексту.

Після вибору конкретної пісні користувач переходить на сторінку детального перегляду, де відображається повний текст композиції. Однією з ключових особливостей інтерфейсу є реалізація інтерактивного тексту: кожне слово пісні є клікабельним. При натисканні на слово система виконує звернення до серверної частини та відображає додаткову інформацію: переклад, базову форму слова, можливе тлумачення та мовні особливості. Такий підхід дозволяє користувачу вивчати лексику безпосередньо в контексті реального тексту, що позитивно впливає на ефективність навчання. Робота зі словом у контексті відповідає сучасним підходам до засвоєння лексики, де значення розглядається не ізольовано, а через уживання у мовному матеріалі [28; 29].

Важливою частиною функціоналу є можливість додавання обраної пісні до персонального списку навчання. Після натискання відповідної кнопки система автоматично аналізує текст композиції, визначає список нових слів та формує персональний навчальний модуль. Таким чином, кожна пісня стає окремим тематичним блоком для подальшого вивчення.

На основі лексики обраної пісні система генерує декілька типів навчальних активностей. Першим видом є флеш-картки, які використовуються для запам'ятовування нових слів. Користувачу відображається слово з тексту пісні, а після взаємодії - переклад або пояснення. Такий формат сприяє кращому засвоєнню нової лексики.

Другим видом навчальних вправ є завдання на вставлення пропущених слів у рядках пісні. Користувачеві пропонується фрагмент тексту з пропущеним словом, яке необхідно відновити. Це дозволяє закріплювати лексику та одночасно розвивати розуміння контексту.

Третім видом функціоналу є вправи на співставлення слів та їх значень або перекладів. Такі вправи сприяють формуванню асоціативних зв'язків і покращують

швидкість пригадування вивченого матеріалу. Окрім цього, система підтримує режим повторення раніше вивчених слів, що дозволяє закріплювати знання у довготривалій пам'яті. Для словникової роботи важливими є повторення, варіювання завдань і поєднання форми слова з його значенням [30].

Для персоналізації навчального процесу реалізовано систему обліку прогресу користувача. У межах особистого кабінету користувач може переглядати:

- список пісень, доданих до навчання;
- перелік нових та вивчених слів;
- результати виконаних вправ;
- загальний прогрес проходження матеріалу.

Система автоматично зберігає інформацію про пройдені вправи, кількість правильних відповідей, складні слова та рівень засвоєння матеріалу. Це дозволяє адаптувати подальші вправи до поточного рівня користувача та формувати більш ефективний сценарій навчання.

Особлива увага була приділена зручності інтерфейсу. Навігація між каталогом пісень, сторінками текстів, вправами та особистим кабінетом реалізована максимально просто та зрозуміло. Інтерфейс розроблено з урахуванням принципів доступності та логічної структури сторінок, що підвищує комфорт використання системи.

Таким чином, у межах дипломного проєкту було реалізовано повноцінний навчальний функціонал та вебінтерфейс, що забезпечують інтерактивне вивчення української мови на основі текстів пісень. Запропонований підхід поєднує зручність використання, персоналізацію навчального процесу та ефективні методи засвоєння лексичного матеріалу.

3.3 Тестування системи та аналіз результатів

Після завершення етапів проєктування та реалізації програмного застосунку було проведено комплексне тестування розробленої системи. Основною метою цього етапу стала перевірка коректності роботи всіх функціональних модулів, виявлення можливих помилок у взаємодії між компонентами системи, оцінка

стабільності роботи застосунку та підтвердження його придатності до практичного використання. Оскільки дипломний проєкт являє собою інформаційно-навчальну систему для вивчення української мови на основі корпусу текстів українських пісень, тестування охоплювало як технічні аспекти роботи програмного продукту, так і перевірку зручності його використання з боку кінцевого користувача.

У межах тестування було поставлено декілька основних завдань. По-перше, необхідно було перевірити правильність роботи серверної логіки та обробки запитів. По-друге, важливо було переконатися у коректності збереження, вибірки та оновлення даних у базі даних. По-третє, потрібно було протестувати навчальний функціонал, пов'язаний із додаванням пісень до персонального навчального списку, формуванням набору нових слів, створенням флеш-карток та генерацією вправ. По-четверте, слід було оцінити роботу користувацького інтерфейсу, логіку переходів між сторінками та зручність навігації. Таким чином, тестування мало комплексний характер і стосувалося усіх рівнів програмного застосунку.

На першому етапі було проведено функціональне тестування. Його сутність полягала в перевірці відповідності фактичної поведінки системи раніше сформованим функціональним вимогам. Кожна основна функція застосунку перевірялася окремо. Зокрема, перевірялася можливість перегляду каталогу пісень, відкриття конкретної пісні, відображення її повного тексту, натискання на окремі слова та отримання довідкової інформації про них.

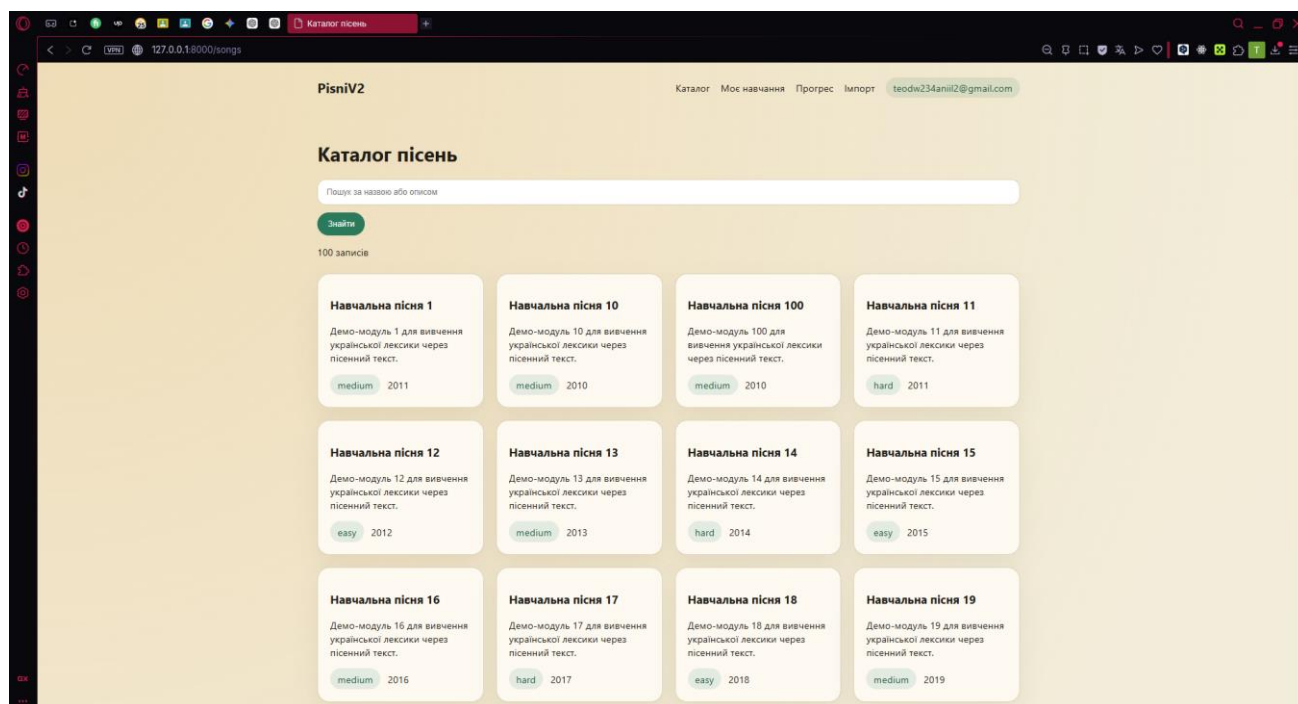


Рисунок 3.1 – Сторінка каталогу пісень та вибір композиції

Окремо було протестовано механізм інтерактивної взаємодії з текстом пісні. Однією з ключових особливостей системи є можливість натискання на будь-яке слово у тексті для отримання його перекладу, базової форми та додаткового пояснення. У процесі перевірки було підтверджено, що система визначає обране слово, виконує запит до серверної частини та відображає результат користувачу без затримок і збоїв, однак якість автоматичного морфологічного аналізу потребує додаткової перевірки.

PisniV2 Каталог Моє навчання Прогрес Імпорт teodw234anil2@gmail.com

Навчальна пісня 100

Демо-модуль 100 для вивчення української лексики через пісенний текст.

medium Фольк-поп

Додати до навчання Флеш-картки Cloze Matching

- 1 Сонце тихо сходить над старими дахами
- 2 і навчас слухати у рядку 1
- 3 Українське слово веде нас до змісту 100
- 4 Місто кличе далі у долинах літа
- 5 щоб слова ожили у рядку 2
- 6 Українське слово веде нас до змісту 101

Нові слова

слово	2
веде	2
нас	2
змісту	2
рядку	2
українське	2
слухати	1
місто	1

Повторення

у	частота: 3
до	частота: 2
змісту	частота: 2
долинах	частота: 1
слухати	частота: 1
старими	частота: 1
рядку	частота: 2
українське	частота: 2

Рисунок 3.2 – Сторінка тексту та взаємодії з піснею

кличе

Lemma: кличий

POS: adjective

Stop-word: ні

Пояснення: Слово 'кличий' розпізнане як adjective.

Переклад: shout out

Частота в пісні: 1

Приклади

- 4. Місто кличе далі у долинах літа

Рисунок 3.3 – Модальне вікно з некоректним перекладом

Рисунок 3.3 демонструє не лише роботу модального вікна, а й обмеження автоматичної лематизації: словоформа «кличе» була помилково визначена як прикметник, хоча у наведеному контексті це дієслівна форма. Такі помилки можуть виникати через омонімію, неповноту словникових даних або обмеження алгоритму

автоматичного морфологічного аналізу [31]. На практиці вони не блокують основний сценарій навчання, однак результати лематизації та перекладу мають перевірятися вручну або уточнюватися за допомогою якіснішого морфологічного аналізатора.

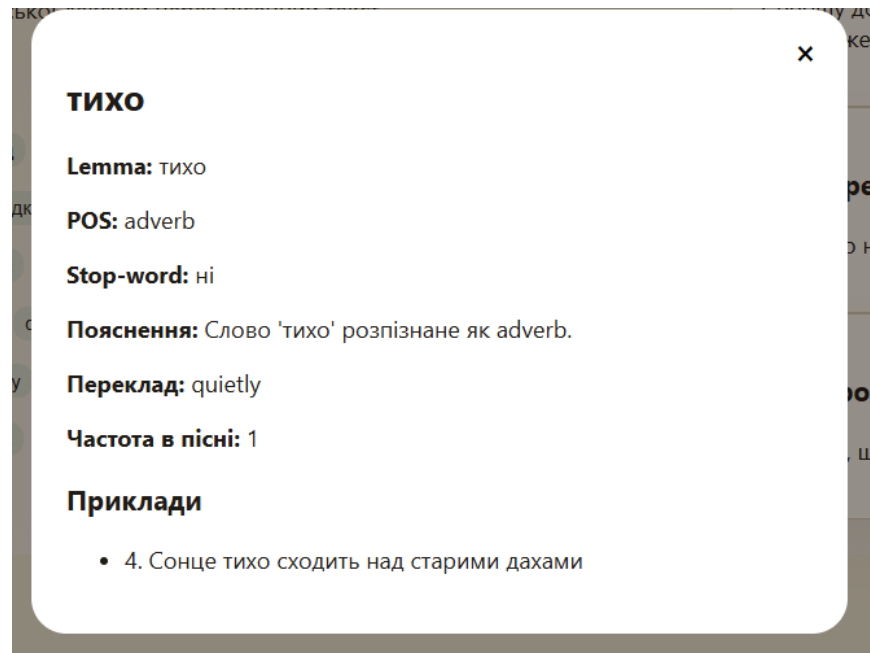


Рисунок 3.4 – Модальне вікно з коректним перекладом

На рисунку 3.4 зафіксовано коректні приклади роботи модуля: система визначає частотність слів, відокремлює службові одиниці та формує список нових слів для подальших вправ. Наведений приклад доцільно розглядати як виявлений дефект, який підтверджує подальшого доопрацювання лінгвістичного модуля.

У ході функціонального тестування особливу увагу було приділено перевірці логіки роботи головного користувацького сценарію. Даний сценарій передбачає, що користувач входить до системи, відкриває каталог пісень, обирає конкретну композицію, переглядає текст, взаємодіє з окремими словами та додає пісню до персонального навчального набору. Після цього він переходить до виконання вправ, сформованих саме на основі обраної пісні. При перевірці цього сценарію було встановлено, що система коректно зберігає зв'язок між користувачем і вибраною піснею, формує перелік слів для вивчення та дозволяє працювати з навчальними активностями у межах одного тематичного модуля.

Після цього було протестовано механізм додавання пісні до навчання. На

цьому етапі перевірялося, чи правильно система створює запис про зв'язок між користувачем і вибраною піснею, чи виділяє список лексичних одиниць, які мають використовуватися у вправах, і чи зберігає цей набір для подальшої роботи. Перевірка показала, що після додавання пісні до навчання вона з'являється в особистому списку користувача, а на її основі стають доступними відповідні навчальні завдання.



Рисунок 3.5 – Додавання пісні до персонального навчального списку

Значний обсяг тестування було присвячено вправам. Для флеш-карток перевірялися правильність формування пари «слово – переклад», коректність послідовності показу карток, можливість переходу між елементами набору та збереження факту вивчення слова. Під час тестування було встановлено, що картки формуються коректно на основі тексту пісні, а взаємодія з ними є зручною та зрозумілою для користувача.

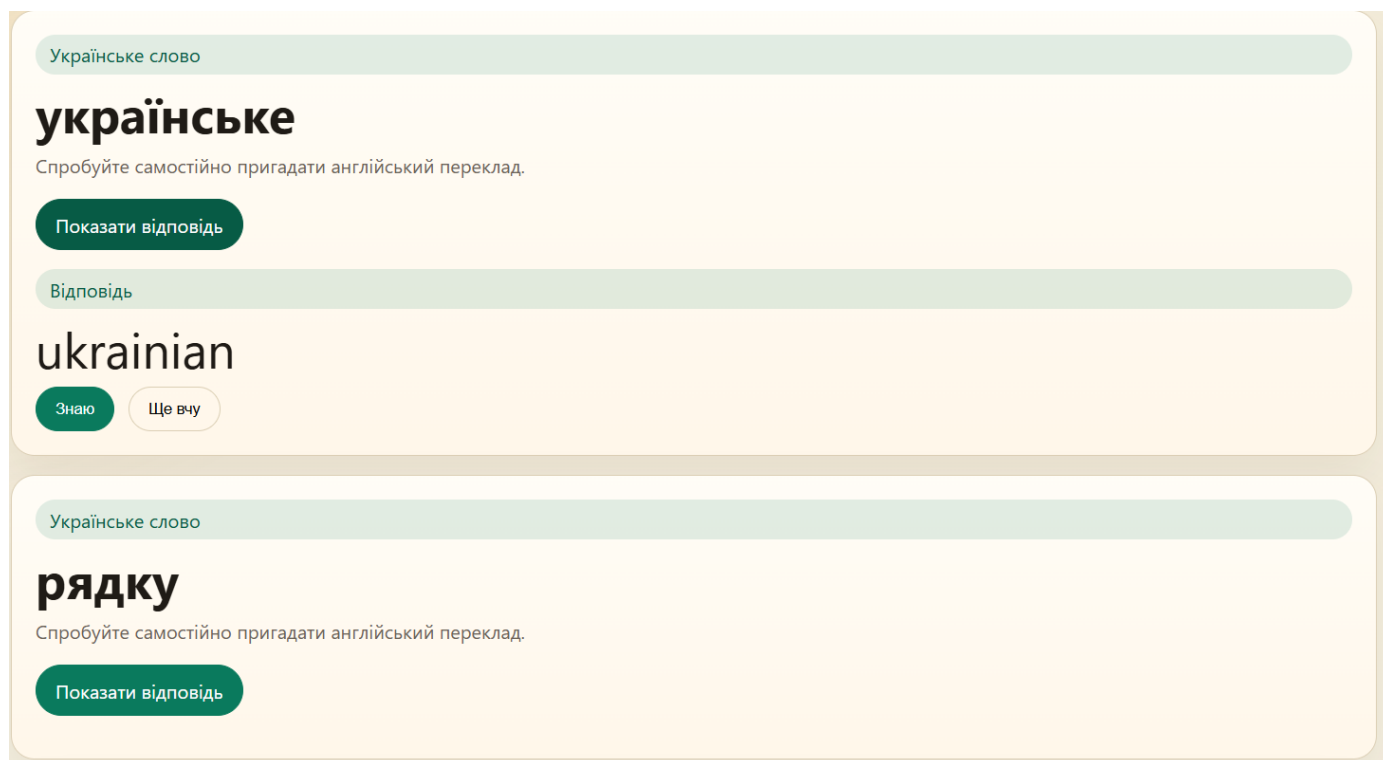


Рисунок 3.6 – Інтерфейс навчальних флеш-карток

Для навчальних вправ тестувалася правильність формування завдань на основі слів із тексту пісні, коректність порівняння введеної відповіді з очікуваним значенням, а також адекватність повідомлень про правильну або неправильну відповідь. Особливістю даного типу вправ є те, що користувачу пропонується окреме слово з пісні англійською мовою, яке необхідно перекласти українською мовою. Таким чином, перевіряється знання лексики, сформованої на основі реального тексту пісні.

Результати перевірки показали, що вправи формуються коректно, система правильно оцінює введений переклад та зберігає результати виконання. У разі правильної відповіді користувач отримує відповідне підтвердження, а у випадку помилки система повідомляє про неправильний переклад і дозволяє повторити спробу.

Переклад слів

city

місто

Перевірити

місто — Вірно

ukrainian

Введіть переклад англійською

Перевірити

Рисунок 3.7 – Виконання вправи на переклад слова з пісні

Після виконання вправ користувач має можливість перейти до інших типів навчальних завдань. Зокрема, передбачено реалізацію вправ на пошук відповідної пари (matching), де необхідно співставити слово та його переклад. На даний момент цей функціонал перебуває в стадії розробки та представлений у вигляді демонстраційної версії.

Matching вправи по пісні #10

Співстав слова і переклади.

- слово — word
- веде — knowledgeable
- нас — us
- змісту — content

Позначити як виконано

Рисунок 3.8 – Демонстраційна версія вправи типу matching

Окремо було виконано перевірку обліку прогресу користувача. Система повинна була зберігати, які слова вже були переглянуті, які вправи виконані, які результати отримані та який відсоток матеріалу по кожній пісні вважається

пройденим. У ході тестування було встановлено, що після виконання вправ відповідні дані коректно зберігаються у базі даних, а особистий кабінет користувача відображає оновлену статистику.

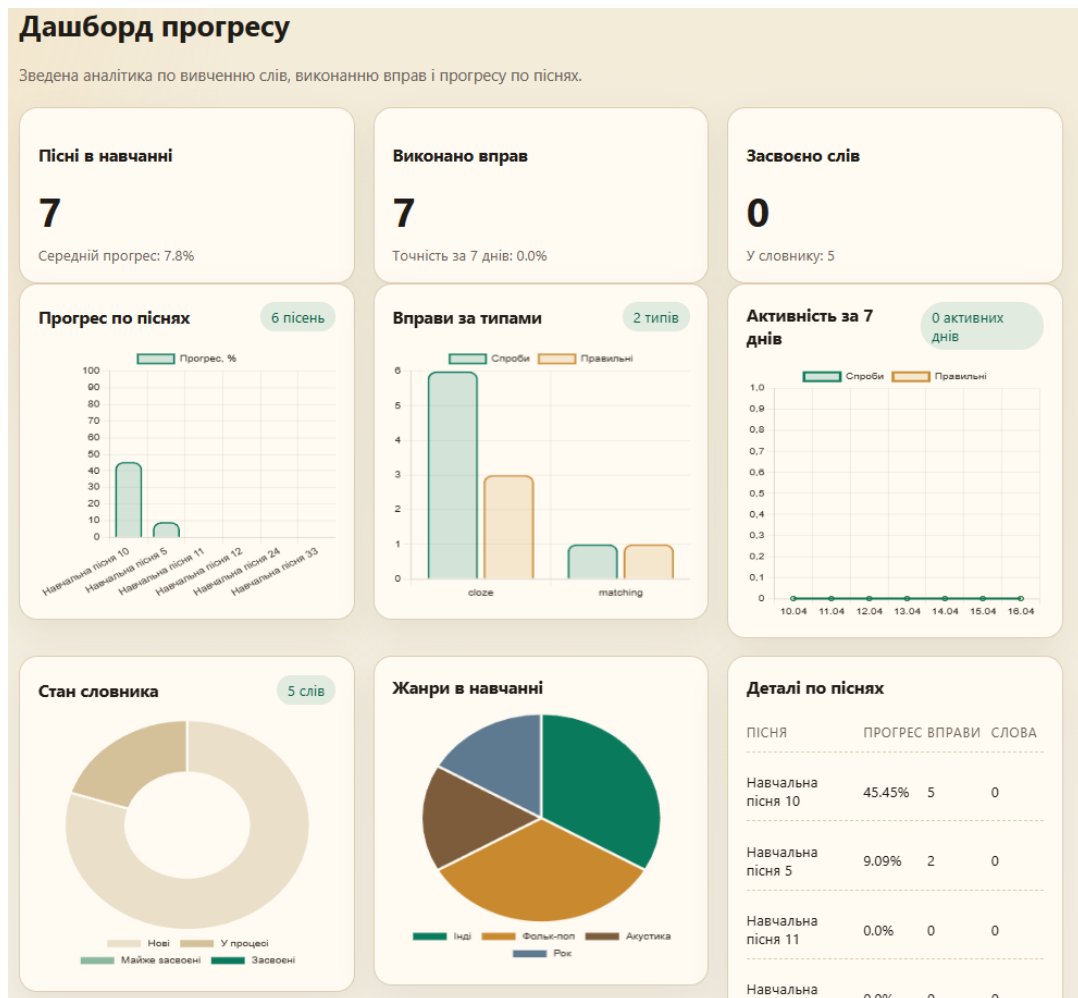


Рисунок 3.9 – Відображення статистики прогресу користувача

Окремим напрямом стало тестування модуля автентифікації користувачів. Було перевірено процес реєстрації нового користувача, введення коректних і некоректних даних, вхід у систему під існуючим обліковим записом, а також перевірка доступу до персональних сторінок. Під час тестування було підтверджено, що система не допускає створення дубльованих облікових записів з однаковими ідентифікаційними даними, коректно обробляє неправильні логін або пароль та не дозволяє неавторизованому користувачу отримати доступ до особистого кабінету.

Рисунок 3.10 – Сторінка реєстрації користувача

Рисунок 3.11 – Сторінка логіну користувача

Наступним важливим етапом стало тестування бази даних. Оскільки основою застосунку є корпус текстів українських пісень, реалізований не у вигляді звичайних текстових файлів, а як повноцінна реляційна структура, необхідно було перевірити цілісність, узгодженість та коректність зв'язків між таблицями. Для цього виконувалась перевірка додавання нових пісень, збереження метаданих, вибірки записів за різними параметрами, оновлення даних, а також видалення записів. Додатково перевірялись зв'язки між таблицями користувачів, пісень, слів, вправ та результатів навчання.

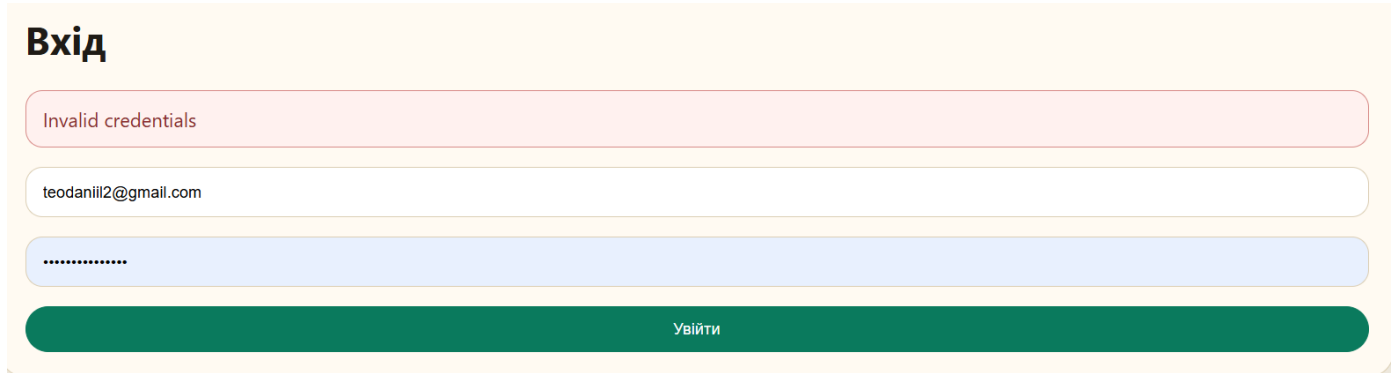
Під час тестування бази даних було встановлено, що система коректно працює з основними сутностями, а використання зовнішніх ключів дозволяє уникати появи логічно некоректних записів. Наприклад, неможливо зберегти результат вправи без існуючого користувача або без існуючої пісні, до якої ця вправа належить. Це підтверджує правильність обраної моделі даних та доцільність використання реляційної СКБД для реалізації корпусного підходу у навчальному застосунку.

Поряд із функціональним тестуванням було проведено тестування інтерфейсу користувача. Його метою стала перевірка того, наскільки зручно користувач може взаємодіяти із системою без додаткових пояснень. Оцінювалися зрозумілість структури сторінок, зручність переходів між розділами, логічність розташування кнопок, інформативність повідомлень та загальна послідовність сценаріїв взаємодії. У результаті було встановлено, що застосунок забезпечує достатньо просту та логічну модель навігації: користувач може без труднощів перейти від каталогу пісень до конкретної композиції, далі - до блоку навчання, а потім - до особистого кабінету з прогресом.

Оскільки система є вебзастосунком, було виконано перевірку коректності роботи інтерфейсу в межах стандартного браузерного середовища. Тестувалися завантаження сторінок, відображення тексту пісень, робота форм введення даних, натискання на інтерактивні елементи, а також реакція системи на некоректні дії користувача. Особливо важливою була перевірка обробки помилок, коли користувач вводить порожні або некоректні дані, намагається перейти до неіснуючої пісні або виконує неавторизовану дію. У таких випадках система повинна не лише залишатися працездатною, а й коректно повідомляти про причину помилки.

Ще одним напрямом стало тестування стійкості системи до типових помилок введення. Наприклад, перевірялося, як застосунок поводить себе при спробі повторної реєстрації вже наявного користувача, при введенні надто короткого пароля, при порожньому полі пошуку, при спробі відкрити сторінку вправ без додавання пісні до навчання. Така перевірка є важливою, оскільки в реальному використанні

користувачі не завжди діють за «ідеальним» сценарієм, тому система повинна бути готовою до некоректних або неповних дій.



The image shows a web login interface. At the top left, the word 'Вхід' (Login) is displayed in bold. Below it, a red rectangular box contains the text 'Invalid credentials'. Underneath this box is an input field for an email address, which contains the text 'teodanil2@gmail.com'. Below the email field is a password field, represented by a blue bar with dots. At the bottom of the form is a green button with the text 'Увійти' (Login).

Рисунок 3.12 – Повідомлення системи при некоректному введенні даних

Окрім перевірки правильності роботи було проведено тестування швидкодії на базовому рівні. Для цього оцінювався час відкриття основних сторінок застосунку, швидкість виконання пошуку по корпусу пісень, швидкість відображення тексту пісні та швидкість відповіді системи при натисканні на слово. Також перевірялося, наскільки швидко формується набір вправ після додавання пісні до навчання. За підсумками спостереження було встановлено, що при локальному запуску системи всі основні операції виконуються достатньо швидко для комфортного використання. Це свідчить про те, що обрана архітектура є придатною для демонстраційного та практичного використання в межах дипломного проєкту.

Для систематизації результатів тестування доцільно було сформувати набір тестових сценаріїв. Кожен сценарій містив вхідні умови, дії користувача, очікуваний результат та фактичний результат виконання. Такий підхід дозволив не лише структуровано перевірити систему, а й зручно зафіксувати підсумки тестування. У межах дипломної роботи можна подати кілька основних сценаріїв у вигляді таблиці.

Таблиця 3.1 – Основні сценарії тестування системи

№	Назва сценарію	Вхідні умови	Дії користувача	Очікуваний результат	Фактичний результат
1	Перегляд каталогу пісень	Система запущена	Відкрити головну сторінку	Відображається список пісень	Відповідає
2	Відкриття сторінки пісні	Є записи в БД	Натиснути на пісню	Відкривається сторінка з текстом	Відповідає
3	Інтерактивне слово	Відкрита пісня	Натиснути на слово	Відображається переклад і інформація	Відповідає
4	Додавання пісні до навчання	Користувач авторизований	Натиснути "Додати до навчання"	Пісня додається в список користувача	Відповідає
5	Повторне додавання пісні	Пісня вже додана	Натиснути повторно	Не створюється дубль	Відповідає
6	Перегляд "Мої пісні"	Є додані пісні	Перейти в розділ	Відображається список пісень	Відповідає
7	Генерація флеш-карток	Пісня додана	Відкрити вправи	Створюються картки зі словами	Відповідає
8	Проходження флеш-карток	Є картки	Натискати "далі"	Показуються слова і переклад	Відповідає
9	Вправа "пропущене слово"	Є вправи	Ввести відповідь	Система перевіряє правильність	Відповідає
10	Вправа "matching"	Є вправи	З'єднати пари	Система перевіряє відповідність	Відповідає
11	Збереження результатів	Виконана вправа	Завершити вправу	Дані записуються в БД	Відповідає
12	Оновлення прогресу	Є результати	Перейти в dashboard	Оновлюється статистика	Відповідає
13	Реєстрація користувача	Новий користувач	Заповнити форму	Створюється обліковий запис	Відповідає
14	Реєстрація з помилкою	Некоректні дані	Ввести неправильний email	Відображається помилка	Відповідає
15	Вхід у систему	Є користувач	Ввести логін і пароль	Успішний вхід	Відповідає
16	Невірний логін	Неправильні дані	Ввести неправильний пароль	Відображається помилка	Відповідає
17	Доступ до dashboard	Користувач увійшов	Перейти в кабінет	Відображається статистика	Відповідає
18	Імпорт TXT	Є файл	Завантажити файл	Пісня додається	Відповідає

Продовження таблиці 3.1

19	Імпорт CSV	Є файл	Завантажити CSV	Додається кілька пісень	Відповідає
20	Імпорт JSON	Є файл	Завантажити JSON	Дані імпортуються	Відповідає
21	Помилка імпорту	Некоректний файл	Завантажити файл	Виводиться помилка	Відповідає
22	Перевірка БД	Є записи	Виконати запити	Дані коректні	Відповідає
23	Обробка помилок	Некоректні дії	Ввести пусті поля	Система показує повідомлення	Відповідає
24	Швидкодія	Система працює	Відкрити сторінку	Сторінка завантажується швидко	Відповідає

Після завершення тестування було виконано аналіз отриманих результатів. Загалом перевірка показала, що реалізований програмний застосунок коректно виконує основні функції, передбачені дипломним проєктом. Система забезпечує роботу з корпусом текстів українських пісень, підтримує інтерактивне вивчення слів у контексті, дозволяє додавати пісні до персонального навчального списку, генерує вправи та зберігає результати користувача. Усі основні модулі взаємодіють між собою узгоджено, а структура бази даних забезпечує зберігання інформації.

Водночас тестування дозволило виявити окремі аспекти, які можуть бути покращені в майбутньому. До них можна віднести розширення словникової бази для більшої кількості слів, удосконалення механізму автоматичного добору нових слів для вивчення, покращення адаптивності інтерфейсу під різні розміри екранів, а також впровадження складніших типів вправ. Окремо слід зазначити, що генератор пісень і модуль автоматичного аналізу можуть створювати помилки у текстах, лемах, частинах мови або перекладах; тому ці компоненти потребують ручної верифікації результатів і подальшого доопрацювання [31].

Отже, проведене тестування підтвердило працездатність розробленого програмного застосунку та відповідність його основних можливостей поставленим завданням. Аналіз результатів показав, що реалізована система може ефективно використовуватися як навчальний інструмент для вивчення української мови на основі корпусу текстів українських пісень.

ВИСНОВКИ

У межах виконання дипломного проєкту було досягнуто основної поставленої мети - розроблено програмний застосунок для вивчення української мови на основі корпусу текстів українських пісень. У процесі виконання роботи було здійснено повний цикл розробки інформаційної системи: від аналізу предметної області та сучасних цифрових рішень до проєктування архітектури, створення бази даних, реалізації програмного функціоналу та проведення комплексного тестування. Отримані результати підтверджують доцільність обраного підходу та практичну цінність створеного програмного продукту.

На початковому етапі роботи було проведено аналіз сучасних методів цифрового вивчення мов, досліджено особливості використання інтерактивних платформ, мультимедійного контенту та корпусного підходу в освітньому процесі. Проведений аналіз показав, що більшість існуючих рішень орієнтовані на універсальні сценарії навчання та не враховують повною мірою можливості використання автентичного українського мовного контенту. Окремо було встановлено, що тексти українських пісень є перспективною основою для створення навчального середовища, оскільки поєднують мовний матеріал, культурний контекст та емоційну залученість користувача.

У результаті аналізу предметної області було сформульовано основні проблеми, які потребували вирішення в межах дипломного проєкту. До таких проблем належали відсутність достатньої кількості спеціалізованих застосунків для вивчення української мови на основі автентичного контенту, недостатнє використання корпусного підходу у сучасних навчальних системах, а також відсутність інструментів, що поєднують інтерактивне вивчення слів у контексті, персоналізацію навчання та збереження прогресу користувача. Саме ці фактори стали підставою для розробки власного програмного рішення.

На етапі проєктування було сформовано функціональні та архітектурні вимоги до майбутньої системи. Було визначено, що застосунок повинен забезпечувати зберігання корпусу текстів українських пісень у структурованій

реляційній базі даних, підтримувати пошук і фільтрацію пісень, виконувати базовий лінгвістичний аналіз текстів, генерувати навчальні вправи та забезпечувати персоналізацію навчального процесу. Додатково було передбачено реалізацію вебінтерфейсу, системи реєстрації користувачів, збереження статистики та можливості подальшого розширення функціоналу.

Для реалізації поставлених завдань було обрано сучасний стек технологій, який включає мову програмування Python, фреймворк FastAPI, ORM-бібліотеку SQLAlchemy, систему керування базами даних PostgreSQL та шаблонізатор Jinja2. Обраний стек дозволив реалізувати модульну архітектуру, зручну взаємодію між компонентами системи та ефективну роботу з корпусом текстових даних.

У процесі практичної реалізації було створено серверну частину програмного застосунку, яка забезпечує маршрутизацію запитів, взаємодію з базою даних, логіку навчального функціоналу та обробку текстової інформації. Було реалізовано модуль корпусу пісень, який дозволяє зберігати, переглядати, редагувати та шукати тексти українських композицій. Важливою особливістю системи стало використання реляційної бази даних для зберігання корпусу пісень у структурованому вигляді, що є значно ефективнішим, ніж використання звичайних текстових файлів.

У межах проєкту було створено базу даних для зберігання україномовних пісенних текстів і передбачено генератор пісень. Для кожної композиції передбачено збереження назви, виконавця, жанру, року випуску та текстового фрагмента. Після нормалізації текстів у системі доступно 324 лексичні одиниці для вивчення. Така структура дозволяє виконувати пошук за різними критеріями, здійснювати аналіз лексики та формувати персоналізований навчальний контент. Корпус потребує подальшої лінгвістичної перевірки.

Особлива увага у роботі була приділена навчальному функціоналу системи. Реалізований застосунок дозволяє користувачу переглядати тексти пісень, натискати на окремі слова та отримувати їх переклад, пояснення або базову форму. Це забезпечує вивчення нової лексики безпосередньо в контексті реального мовного середовища, що є більш ефективним порівняно з традиційним

заучуванням слів без контексту.

Ключовою особливістю системи є можливість додавання обраної пісні до персонального навчального списку. Після цього система автоматично формує індивідуальний набір слів та вправ на основі тексту конкретної композиції. Кожна пісня фактично стає окремим тематичним модулем для вивчення. Це забезпечує високий рівень персоналізації навчального процесу та підвищує зацікавленість користувача.

У межах навчального модуля було реалізовано декілька типів вправ: флеш-картки для запам'ятовування нових слів, вправи на вставлення пропущених слів у рядках пісні, вправи на співставлення слів і значень, а також механізм повторення раніше вивченого матеріалу. Такий підхід дозволяє поєднати декілька методик навчання та підвищити ефективність засвоєння мовного матеріалу, оскільки вивчення лексики потребує повторюваних контактів зі словом і роботи з його значенням у контексті [28; 30].

Також у межах дипломного проєкту було реалізовано систему реєстрації користувачів, особистий кабінет та механізм відстеження навчального прогресу. Користувач має можливість переглядати перелік доданих до навчання пісень, список нових і вивчених слів, результати виконаних вправ та загальну статистику. Це дозволяє зробити процес навчання більш структурованим, контрольованим та мотивуючим.

Важливим завершальним етапом стало тестування системи. Було проведено комплексну перевірку функціональних модулів, бази даних, навчального функціоналу, системи авторизації та інтерфейсу користувача. Результати тестування підтвердили коректність роботи застосунку, стабільність взаємодії між компонентами системи та відповідність реалізованого функціоналу поставленим вимогам. Було встановлено, що система коректно виконує сценарії перегляду пісень, взаємодії зі словами, додавання пісень до навчання, проходження вправ та збереження прогресу.

Практична цінність дипломного проєкту полягає у створенні повноцінного програмного продукту, який може використовуватися як інструмент для

самостійного вивчення української мови. Запропонований підхід дозволяє поєднати сучасні інформаційні технології, корпусний аналіз текстів та інтерактивні методи навчання. Це робить процес вивчення мови більш природним, зручним та цікавим для користувача.

Окремою перевагою розробленого застосунку є можливість його подальшого розвитку. У перспективі система може бути розширена за рахунок:

- інтеграції аудіосупроводу пісень;
- автоматичного визначення складності лексики;
- розширення словникової бази;
- адаптації під мобільні пристрої;
- реалізації рекомендаційної системи для добору нових пісень;
- додавання соціальних функцій і спільного навчання.

У межах дипломного проєкту було розроблено та протестовано програмний застосунок для вивчення української мови на основі корпусу україномовних пісенних текстів. Усі поставлені завдання виконано, основну мету роботи досягнуто. Розроблений програмний продукт підтвердив свою працездатність, практичну цінність та перспективність використання у сфері цифрової мовної освіти. Водночас генератор пісень, словникова база та модуль автоматичної лематизації потребують подальшого доопрацювання, оскільки автоматично сформовані тексти й результати морфологічного аналізу можуть містити помилки. Застосунок може слугувати основою для подальших досліджень і вдосконалення освітніх інформаційних систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. The Handbook of Technology and Second Language Teaching and Learning / ed. by C. A. Chapelle, S. Sauro. Hoboken : Wiley-Blackwell, 2017.
2. Beatty K. Teaching and Researching Computer-Assisted Language Learning. 2nd ed. London : Pearson Education, 2010.
3. Бугай А. В. Інформаційні технології в системі сучасної освіти. 2023. С. 45–61.
4. Гончарова І. П. Використання інформаційних технологій у навчальному процесі. 2023. С. 38–49
5. Levy M., Stockwell G. CALL Dimensions: Options and Issues in Computer-Assisted Language Learning. New York : Routledge, 2006. 328 p
6. Stockwell G. Computer-Assisted Language Learning: Diversity in Research and Practice. Cambridge : Cambridge University Press, 2012. 231 p
7. Godwin-Jones R. Smartphones and language learning. Language Learning & Technology. 2017. Vol. 21, № 2. P. 3–17
8. Kukulska-Hulme A., Shield L. An overview of mobile assisted language learning: from content delivery to supported collaboration and interaction. ReCALL. 2008. Vol. 20, № 3. P. 271–289
9. Ludke K. M., Ferreira F., Overy K. Singing can facilitate foreign language learning. Memory & Cognition. 2014. Vol. 42, № 1. P. 41–52
10. Жуковська В. В. Вступ до корпусної лінгвістики : навч. посіб. Житомир : Вид-во ЖДУ ім. І. Франка, 2013. 142 с
11. Boulton A., Cobb T. Corpus use in language learning: a meta-analysis. Language Learning. 2017. Vol. 67, № 2. P. 348–393
12. Flowerdew L. Corpora and Language Education. Basingstoke : Palgrave Macmillan, 2012. 368 p. URL: <https://link.springer.com/book/10.1057/9780230355569> (дата звернення: 20.03.2026).
13. Multiple Affordances of Language Corpora for Data-driven Learning / ed. by A. Leńko-Szymańska, A. Boulton. Amsterdam : John Benjamins, 2015. 295 p
14. Duolingo : вебсайт. URL: <https://www.duolingo.com>

15. LingQ : вебсайт. URL: <https://www.lingq.com>
16. LyricsTraining : вебсайт. URL: <https://lyricstraining.com>
17. Beyond Concordance Lines: Corpora in Language Education / ed. by P. Pérez-Paredes, G. Mark. Amsterdam : John Benjamins, 2021. 255 p
18. Серверні WEB-технології : навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2023. 163 с.
19. Люшенко Л. А., Хіцко Я. В. Розробка та аналіз вимог до програмного забезпечення. Київ : КПІ ім. Ігоря Сікорського, 2020. 64 с.
20. Проектування інформаційних систем. Бази даних : навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2022. 112 с.
21. Python Software Foundation. Python Documentation : вебсайт. URL: <https://docs.python.org>
22. FastAPI Documentation : вебсайт. URL: <https://fastapi.tiangolo.com>
23. SQLAlchemy Documentation : вебсайт. URL: <https://docs.sqlalchemy.org>
24. PostgreSQL Global Development Group. PostgreSQL Documentation : вебсайт. URL: <https://www.postgresql.org/docs>
25. Pallets Projects. Jinja Documentation : вебсайт. URL: <https://jinja.palletsprojects.com>
26. Chart.js Documentation : вебсайт. URL: <https://www.chartjs.org>
27. Лосєв М. Ю., Федько В. В. Бази даних : навч.-практ. посіб. Харків : ХНЕУ ім. С. Кузнеця, 2018. 233 с.
28. Nation I. S. P. Learning Vocabulary in Another Language. 2nd ed. Cambridge : Cambridge University Press, 2013. 605 p.
29. Webb S., Nation P. How Vocabulary Is Learned. Oxford : Oxford University Press, 2017. 322 p.
30. Schmitt N., Schmitt D. Vocabulary in Language Teaching. 2nd ed. Cambridge : Cambridge University Press, 2020. 290 p.
31. Jurafsky D., Martin J. H. Speech and Language Processing : an Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models. 3rd ed., draft. 2026.

ДОДАТКИ

Додаток А

