



УДК 004.42:004.774.6

MODULAR APPROACH TO CREATING A WEB APPLICATION FOR LEARNING ENGLISH

МОДУЛЬНИЙ ПІДХІД ДО СТВОРЕННЯ ВЕБДОДАТКУ ДЛЯ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ

Ivasiuk H.P. / Івасюк Г.П.*s.ph-m.s., as.prof. / к.ф.-м.н., доц.*

ORCID: 0000-0003-4381-3071

Fratavchan T.M. / Фратавчан Т.М.*s.ph-m.s., as.prof. / к.ф.-м.н., доц.*

ORCID: 0000-0003-1076-0794

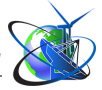
Nikita A.V. / Нікіта А.В.*Yuriy Fedkovych Chernivtsi National University,**Chernivtsi, Kotsiubynskoho 2, 58002**Чернівецький національний університет ім.Ю.Федьковича,**м. Чернівці, Коцюбинського 2, 58002*

Анотація. Робота присвячена розробці вебдодатка для вивчення англійської мови, який враховує найкращі практики розробки вебдодатків. Використані технології включають React та Next.js для фронтенду, NestJS та TypeORM для бекенду, а також TypeScript для підвищення надійності коду. Система авторизації реалізована за допомогою JWT для забезпечення безпеки та гнучкості доступу. Додаток містить функціонал для управління тестами, вивчення граматики, збільшення словникового запасу, а також конструктор речень.

Ключові слова: вебдодаток, React, Next.js, NestJS, TypeORM, TypeScript, JWT, фронтенд, бекенд.

Вступ.

Вебдодатки для вивчення іноземних мов стали популярними інструментами для навчання в умовах цифрової епохи. Вони надають користувачам зручний доступ до матеріалів, вправ та інтерактивних функцій, що допомагають у швидкому освоєнні іноземних мов. Однак з ростом кількості користувачів, контенту та обсягів даних, що зберігаються у базах даних, виникає необхідність у їхній оптимізації. Оптимізація вебдодатка для вивчення мов стає критично важливою для забезпечення стабільної роботи додатка навіть при значному навантаженні. Це не лише покращує швидкість завантаження і взаємодії з додатком, а й дозволяє ефективно обробляти великі масиви даних, що є необхідним для надання персоналізованих навчальних шляхів для кожного користувача.



Метою роботи є розробка та впровадження вебдодатку для вивчення англійської мови, що поєднує в собі сучасні методи навчання та передові технології. Вебдодаток для вивчення англійської мови має на меті поєднати сучасні методи навчання з новітніми технологіями. Додаток складається з трьох основних блоків: граматичної частини, словникової частини та системи тестування. Граматична частина містить довідковий матеріал з текстом, прикладами та ілюстраціями, структурований за тематикою і складністю. Словникова частина надає можливість переглядати картки з словами, слухати їх вимову та проходити тести на знання слів. Система тестування включає різноманітні тести, зокрема новий тест "Конструктор речень", де користувачам потрібно скласти речення із запропонованих слів.

У процесі виконання роботи були використані різні методи дослідження: аналіз джерел та схожих додатків, проектування системи, програмування, тестування.

Огляд існуючих додатків.

Існує чимало альтернативних вебдодатків для вивчення іноземних мов, таких як Duolingo, Memrise, Babbel та інші.

Duolingo є одним з найпопулярніших додатків для вивчення мов. Він використовує ігровий підхід, де користувачі виконують різноманітні вправи, щоб заробити бали та досягти нового рівня. Duolingo пропонує багато мов, але його основна увага зосереджена на початкових рівнях володіння іноземною мовою.

Memrise також використовує ігровий підхід, але зосереджується на запам'ятовуванні слів та фраз за допомогою карток і мнемонічних методів. Хоча Memrise є ефективним інструментом для розширення словникового запасу, він менш підходить для вивчення граматики та побудови речень.

Babbel пропонує курси, створені мовними експертами, з акцентом на практичне використання мови. Додаток Babbel не є безкоштовним, що може бути обмеженням для деяких користувачів.

Загалом, хоча існують багато альтернативних додатків для вивчення мов,



розроблений нами додаток пропонує комбінацію різноманітних методів та сучасних технологій, що робить його ефективним інструментом для вивчення англійської мови. Він може бути більш доступним завдяки можливості користувачів створювати власні тести та поділитися ними з іншими.

Аналізуючи різноманітні дослідження, можна зробити висновок, що вебдодатки можуть не лише доповнювати традиційні методи вивчення мов, але й частково замінювати їх. Наприклад, дослідження Duolingo показало, що учні, які пройшли курси іспанської та французької мов, досягли такого ж рівня знань, як і студенти університетських курсів, але вдвічі швидше. Інше дослідження, опубліковане New York Times, вказує на те, що такі додатки є дуже корисними для початківців і дозволяють досягти рівня B1/B2.

Таким чином, попит на такі додатки постійно зростає, а актуальність вивчення іноземних мов, зокрема англійської, залишатиметься високою ще довгий час. Вебдодаток, який пропонує комплексний підхід до вивчення англійської мови, є важливим інструментом у цьому контексті, оскільки він задовольняє сучасні потреби користувачів, надаючи їм зручний та ефективний спосіб вивчення мови.

Аналіз вибраних технологій.

При розробці такого додатка важливо враховувати:

- масштабованість – підтримка великої кількості користувачів,
- продуктивність – швидке завантаження та обробка даних,
- гнучкість – легке розширення функціоналу,
- безпеку – захист даних користувачів.

Тому його реалізація базується на стеку технологій: React, Next.js, Node.js, TypeScript, NestJS та TypeORM [1–6].

Особливості розробки додатка.

Архітектура додатку побудована на клієнт-серверній моделі. Фронтенд реалізований на React із використанням Next.js для серверного рендерингу та покращення продуктивності. Серверна частина розроблена на NestJS, що забезпечує модульну структуру та підтримку архітектурного патерну MVC.



Взаємодія між клієнтом і сервером здійснюється через REST API.

Функціональні можливості:

- аутентифікація та авторизація (JWT);
- управління користувачами та їхніми рівнями доступу;
- модуль навчальних матеріалів із можливістю адаптації під користувача;
- відстеження прогресу навчання та надання рекомендацій;
- інтерактивні вправи з автоматичною перевіркою.

База даних побудована на MySQL, для роботи з якою використовується TypeORM. Забезпечено оптимізацію запитів для підвищення продуктивності.

Використовується шифрування паролів через bcrypt, а також політика безпечного зберігання даних.

Додаток забезпечує ефективну взаємодію користувачів з навчальними матеріалами, а також використовує сучасні вебтехнології для підвищення продуктивності та безпеки.

Опис сторінки персонального кабінету.

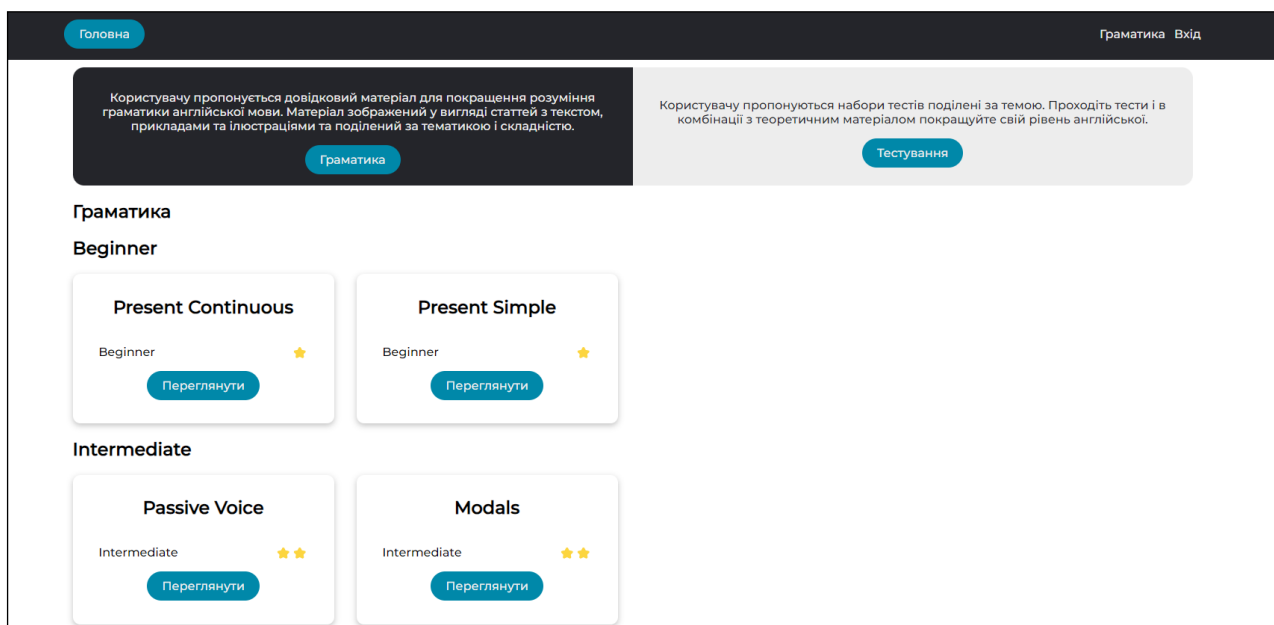
Сторінка персонального кабінету дозволяє користувачам переглядати свою інформацію, результати тестів та керувати створеними тестами. Ця сторінка також включає адмінпанель для адміністраторів, де вони можуть керувати реченнями та тестами. Для управління різними станами використовується хук `useState`, наприклад, інформація про користувача, результати тестів, доступні теми, поточний вибраний тест та речення. Хук `useEffect` використовується для завантаження даних з API під час ініціалізації React-компонента. Після першого рендерингу компонента викликається функція для отримання даних, яка надсилає кілька запитів до API для отримання інформації про користувача, його результати тестів, теми та створені тести. У всіх запитах використовується токен для аутентифікації та авторизації.

React-компонент забезпечує обробку результатів тестів через API-запити. Користувачі можуть переглядати свої результати у персональному кабінеті, де відображаються досягнення у різних типах тестів.

Можна виділити наступні основні елементи сторінки: інформація про



користувача, адмінпанель, керування тестами, додавання нових речень, досягнення.



**Рисунок 1 - Головна сторінка додатку.
Досягнення**

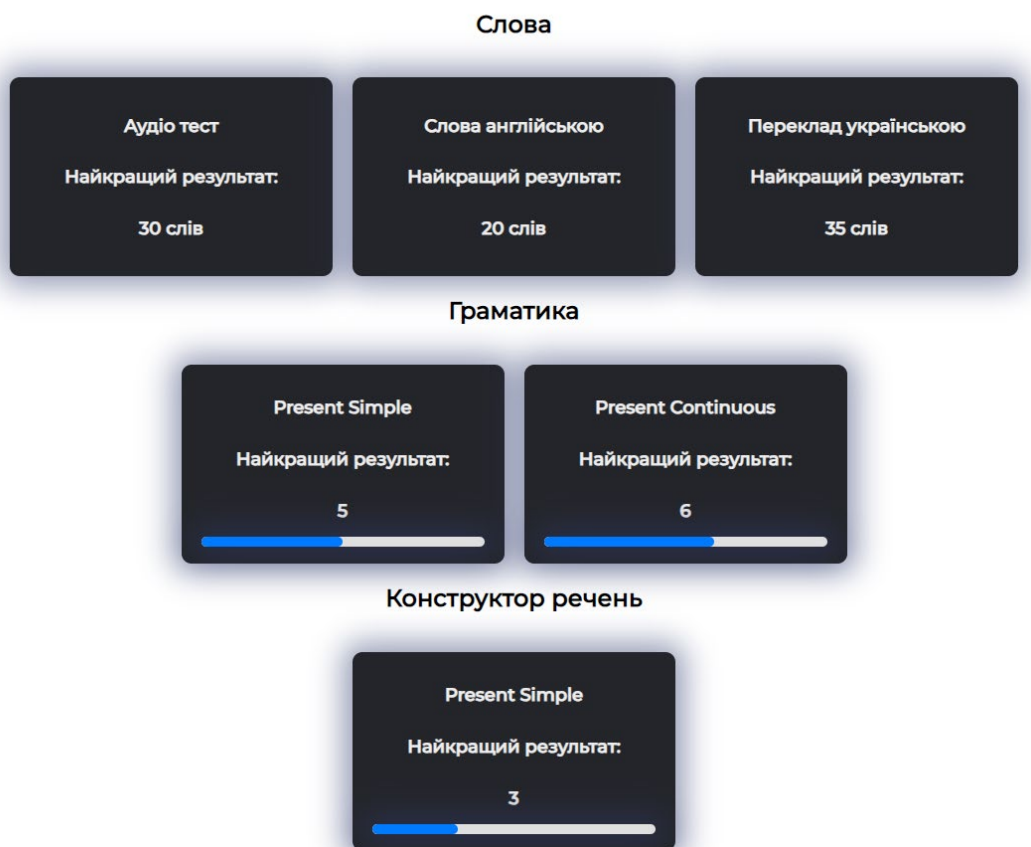


Рисунок 2 - Досягнення користувача в різних тестах і вправах.



Таким чином, сторінка персонального кабінету забезпечує зручний інтерфейс для перегляду особистої інформації та результатів, а також для управління тестами та реченнями, надаючи можливості як для звичайних користувачів-учнів, користувачів-вчителів, так і адміністраторів.

Рисунок 3 - Вигляд блоку для керування користувацькими тестами

Висновки.

У роботі показано приклад створення сучасного повноцінного вебдодатка для вивчення англійської мови. Використано сучасні технології та підходи, що забезпечили функціональність, масштабованість та безпеку додатка. На frontend частині використано React та Next.js, що забезпечує високий рівень інтерактивності, покращене SEO та продуктивність додатка. Використання багаторазових компонентів та автоматичне розділення коду сприяє кращій організації коду та швидкому завантаженню сторінок. На backend частині використано NestJS, що забезпечує модульну архітектуру та асинхронну обробку запитів, що дозволяє легко масштабувати додаток. TypeORM та MySQL забезпечують ефективну роботу з базою даних, а TypeScript зменшує кількість помилок на етапі розробки та полегшує підтримку коду. Система авторизації реалізована за допомогою JWT, забезпечуючи надійну та гнучку аутентифікацію та авторизацію користувачів. Використання ESLint автоматизує перевірку коду, підтримуючи його високу якість.



Література:

1. Кірупа Чіннатхамбі. Learning React: A Hands-On Guide to Building Web Applications Using React and Redux. – Longman (Pearson Education), 2018. – 350p.
2. Michele Riva. Real-World Next.js: Build scalable, high-performance, and modern web applications using Next.js, the React framework for production. – Packt Publishing, 2022. – 366 p.
3. J. Bell, G. Magolan, D. Guijarro, P. Housley, A. Peretti. Nest.js: A Progressive Node.js Framework. – Bleeding Edge Press, 2018. – 408 p.
4. Nathan Rozentals. Mastering TypeScript. – Packt, 2021. – 516 p.
5. M. Fowler. Patterns of Enterprise Application Architecture. – Addison-Wesley, 2003. – P.560.
6. R. Gupta. Hands-on Penetration Testing for Web Applications. – Bpb Publications, 2021. – 310 p.

Abstract. *The work examines how to build modern web applications using frontend and backend technologies. A web application for learning English was developed, which takes into account the best practices of web application development. The technologies used include React and Next.js for the frontend, NestJS and TypeORM for the backend, and TypeScript to enhance code reliability. The authorization system is implemented using JWT to ensure security and flexible access control. The application includes functionality for managing tests, learning vocabulary and grammar.*

Key words: *web application, React, Next.js, NestJS, TypeORM, TypeScript, JWT, frontend, backend.*