

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»

КВАЛІФІКАЦІЙНА РОБОТА

Тема: «Вебсайт «NextLevel» для чіп-тюнінгу автомобілів з використанням багатокритеріальних фільтрів та рекомендаційної системи»

Ступінь вищої освіти – бакалавр
Спеціальність – 122 «Комп’ютерні науки»
Освітня програма «Комп’ютерні науки»

ПОЯСНЮВАЛЬНА ЗАПИСКА

Виконав: здобувач 4 курсу
групи КН-22

Ростислав КОВАЛЬОВ

Керівник: викладач кафедри
інформаційного менеджменту,
математики та статистики
Олег МУШИНСЬКИЙ

Засвідчую, що кваліфікаційна
робота оформлена відповідно до
ДСТУ 3008:2015 та не містить
запозичень з праць інших авторів
без відповідних посилань.

Здобувач: _____
(підпис)

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»**

ЗАТВЕРДЖУЮ:

завідувач кафедри комп'ютерних наук

_____ Сергій МІЧКІВСЬКИЙ

«____» _____ 20__ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Ковальов Ростислав Олександрович

Тема роботи	Вебсайт «NextLevel» для чіп-тюнінгу автомобілів з використанням багатокритеріальних фільтрів та рекомендаційної системи
Номер та дата наказу про затвердження теми	№ 133-7 від 22 грудня 2025 року
Коротка постановка завдання	Розробка вебсайту для чіп-тюнінгу автомобілів з використанням багатокритеріальних фільтрів та рекомендаційної системи
Посилання на джерела інформації (не більше п'яти найменувань, які рекомендує науковий керівник)	Курдупов О.Л Нескородева Т.В. Дослідження систем рекомендацій при створенні веб-сайту автосалону. Матеріали IV Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених «Прикладні інформаційні технології». Донецький національний університет імені Василя Стуса, 2023
Вимоги до кваліфікаційної роботи	Кваліфікаційна робота має передбачити теоретичне, системотехнічне або експериментальне дослідження складного спеціалізованого завдання або практичної проблеми в галузі комп'ютерних наук, яке характеризується комплексністю та невизначеністю умов і потребує застосування теорій і методів інформаційних технологій.

Дата видачі завдання 27 грудня 2025 р.

Керівник

Здобувач освітнього ступеня бакалавра

Олег МУШИНСЬКИЙ

Ростислав КОВАЛЬОВ

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання	Примітка
Підготовчий етап			
1	Вибір напряму дослідження	01.12.2025 р.	
2	Формування теми та призначення керівника	15.12.2025 р.	
3	Затвердження теми кваліфікаційної роботи	22.12.2025 р.	
4	Затвердження завдання на кваліфікаційну роботу	26.12.2025 р.	
Основний етап			
5	Розробка концепції кваліфікаційної роботи	23.01.2026 р.	
6	Підбір та вивчення джерел інформації з напряму дослідження. Огляд існуючих аналогів	13.03.2026 р.	
7	Затвердження розширеної постановки завдання. Підготовка та подання керівникові розділу 1 кваліфікаційної роботи	20.03.2026 р.	
8	Проектування. Підготовка та подання керівникові розділу 2 кваліфікаційної роботи	27.03.2026 р.	
9	Підготовка доповіді для експертизи стану виконання кваліфікаційної роботи (проміжний контроль)	30.03-04.04.2026 р.	
10	Реалізація. Підготовка та подання керівникові розділу 3 кваліфікаційної роботи	06.04.2026 р.	
11	Підготовка та подання керівнику першого варіанту всієї кваліфікаційної роботи	13.04.2026 р.	
12	Доопрацювання кваліфікаційної роботи з урахуванням зауважень керівника та представлення керівникові доопрацьованого варіанту кваліфікаційної роботи	20.04.2026 р.	
Завершальний етап			
13	Представлення рукопису для перевірки на плагіат	27.04-03.05.2026 р.	
14	Підготовка презентації та доповіді на передзахист	04.05-10.05.2026 р.	
15	Передзахист кваліфікаційної роботи	11.05-15.05.2026 р.	
16	Доопрацювання роботи за результатами передзахисту	18.05-05.06.2026 р.	
17	Експертиза роботи керівником та зовнішнім експертом	08.06-14.06.2026 р.	
18	Доопрацювання доповіді та презентації для захисту	08.06-14.06.2026 р.	
19	Захист кваліфікаційної роботи	15.06-21.06.2026 р.	

Керівник
Здобувач освітнього ступеня бакалавра

Олег МУШИНСЬКИЙ
Ростислав КОВАЛЬОВ

Ковальов Р.О. Вебсайт «NextLevel» для чіп-тюнінгу автомобілів з використанням багатокритеріальних фільтрів та рекомендаційної системи.

Пояснювальна записка кваліфікаційної роботи за спеціальністю 122 – Комп’ютерні науки (освітня програма – Комп’ютерні науки) СО Бакалавр. – ВНЗ «Університет економіки та права «КРОК», Навчально-науковий інститут інформаційних та комунікаційних технологій, кафедра комп’ютерних наук, Київ, 2026.

Розглядається проблема підбору прошивок до різноманітних моделей автомобілів для покращення їх швидкісних характеристик. Розроблено веб-застосунок конфігуратора, в якому автоматизовано підбір даних прошивок за їх характеристиками сумісності.

Ключові слова: веб-застосунок, конфігуратор, комп’ютер.

Kovalyov R.O. “NextLevel” Website for Car Chip Tuning Using Multi-Criteria Filters and a Recommendation System.

Explanatory note to the qualification thesis in specialty 122 – Computer Science (Educational Program – Computer Science), Bachelor’s degree. – “KROK” University of Economics and Law, Educational and Scientific Institute of Information and Communication Technologies, Department of Computer Science, Kyiv, 2026.

The paper addresses the problem of selecting firmware for various car models in order to improve their speed performance. A web-based configurator application has been developed, in which the selection of such firmware is automated according to their compatibility characteristics.

Keywords: web application, configurator, computer.

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1 ПОСТАНОВКА ЗАВДАННЯ НА РОЗРОБКУ ВЕБСАЙТ «NEXTLEVEL»	8
1.1 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.2 ВИЗНАЧЕННЯ ПОТЕНЦІЙНИХ КОНКУРЕНТНИХ ПЕРЕВАГ ПРОГРАМНОГО ПРОДУКТУ	10
1.3 ПОСТАНОВКА ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ	15
Висновки до розділу 1.....	17
РОЗДІЛ 2 ПРОЄКТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ «NEXTLEVEL»	18
2.1 МОДЕЛЮВАННЯ ПОВЕДІНКИ ПРОДУКТУ	18
2.2 МОДЕЛЮВАННЯ АРХІТЕКТУРИ ТА СТРУКТУРИ ПРОДУКТУ	24
2.3 ОПИС АЛГОРИТМІВ ТА МАТЕМАТИЧНИХ МОДЕЛЕЙ.....	26
Висновки до розділу 2.....	28
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ПРОДУКТУ «NEXTLEVEL»	30
3.1 ТЕХНОЛОГІЧНІ РІШЕННЯ ПРОГРАМНОГО ПРОДУКТУ	30
3.2 ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ.....	37
3.3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОЄКТУ.....	38
Висновки до розділу 3.....	47
ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАННЯ.....	50

ВСТУП

Актуальність теми. У сучасних умовах стрімкого розвитку автомобільних технологій та зростання попиту на індивідуалізацію транспортних засобів чіп-тюнінг стає популярним способом покращення технічних і швидкісних характеристик автомобілів. Водночас процес підбору прошивок для різних моделей автомобілів є складним і потребує врахування великої кількості параметрів сумісності, технічних характеристик двигуна та електронних систем керування. Неправильний вибір прошивки може призвести до зниження ефективності роботи автомобіля або навіть до технічних несправностей. Тому актуальним є створення веб-застосунку, який автоматизує процес підбору прошивок із використанням багатокритеріальних фільтрів та рекомендаційної системи.

Існуючі онлайн-ресурси та сервіси з підбору прошивок часто не забезпечують достатнього рівня автоматизації, не враховують усіх параметрів сумісності або не надають персоналізованих рекомендацій. Користувачам доводиться самотійно аналізувати технічні характеристики, порівнювати варіанти та консультуватися з фахівцями, що потребує значних витрат часу та спеціальних знань.

Мета дослідження. Розробка веб-застосунку «NextLevel» для підбору прошивок чіп-тюнінгу автомобілів із застосуванням багатокритеріальних фільтрів та рекомендаційної системи, що забезпечить автоматизований і зручний вибір оптимальних рішень відповідно до характеристик автомобіля та потреб користувача.

Завдання дослідження:

- проаналізувати наявні аналоги веб-застосунків і сервісів для підбору прошивок чіп-тюнінгу та визначити їх переваги й недоліки;

- розробити алгоритм багатокритеріального відбору прошивок з урахуванням параметрів сумісності;
- спроектувати структуру рекомендаційної системи;
- здійснити проектування та реалізацію веб-застосунку «NextLevel».

Об’єктом дослідження є процес підбору прошивок для чіп-тюнінгу автомобілів за характеристиками сумісності та продуктивності.

Предмет дослідження є методи та програмні засоби автоматизації підбору прошивок у веб-застосунку з використанням багатокритеріальної фільтрації та рекомендаційних алгоритмів.

Методи дослідження. Методи аналізу та узагальнення інформації, підходи до побудови алгоритмів багатокритеріального вибору, методи проектування інформаційних систем, а також алгоритми обробки та структуризації даних.

Практичне значення. Розроблений веб-застосунок «NextLevel» має практичну цінність для власників автомобілів і спеціалістів з чіп-тюнінгу, оскільки дозволяє швидко та обґрунтовано підібрати відповідну прошивку з урахуванням технічних характеристик автомобіля та побажань користувача, зменшуючи ризик помилкового вибору.

Структура роботи. Кваліфікаційна робота складається зі вступу, трьох розділів, висновків та списку використаних джерел. Пояснювальна записка містить таблиці та рисунки. Загальний обсяг роботи становить 52 сторінки, основний зміст викладено на 48 сторінках.

РОЗДІЛ 1

ПОСТАНОВКА ЗАВДАННЯ НА РОЗРОБКУ ВЕБСАЙТ «NEXTLEVEL»

1.1 Опис предметної області

У сфері автомобільного тюнінгу чіп-тюнінг є одним із найпоширеніших способів підвищення потужності двигуна, крутного моменту та покращення динамічних характеристик транспортного засобу. Процес чіп-тюнінгу передбачає зміну програмного забезпечення електронного блоку керування (ECU) автомобіля шляхом встановлення відповідної прошивки.

Підбір прошивки є складним завданням, оскільки необхідно враховувати модель автомобіля, тип і об'єм двигуна, покоління ECU, тип пального, наявність турбонаддуву, екологічні стандарти та інші технічні параметри. Неправильно підібрана прошивка може призвести до некоректної роботи двигуна або пошкодження його компонентів. Тому ключовим фактором є забезпечення повної технічної сумісності.

Кожна прошивка характеризується набором параметрів, які можуть бути:

- числовими (наприклад, об'єм двигуна, потужність, крутний момент);
- рядковими (марка, модель, тип двигуна, версія ECU);
- логічними (наявність турбіни, підтримка певного стандарту тощо).

Для визначення сумісності необхідно реалізувати алгоритми порівняння характеристик відповідно до їх типів даних:

- для числових параметрів - перевірка рівності або належності до допустимого діапазону значень;
- для рядкових - перевірка повного або часткового співпадіння;
- для логічних - перевірка відповідності істинності параметра.

Веб-застосунок-конфігуратор чіп-тюнінгу – це онлайн-інструмент, який надає користувачам можливість автоматизовано підібрати відповідну прошивку на основі введених характеристик автомобіля.

Вебсайт передбачає створення зручного інтерфейсу з каталогом запчастин, фільтрацією за маркою, моделлю, роком випуску автомобіля, а також можливістю замовлення чіп-тюнінгу онлайн. Додатково планується реалізувати систему зворотного зв'язку, онлайн-консультації та інтеграцію платіжних сервісів для прийому замовлень.

Таблиця 1.1 – Логічна матриця проекту

Структура проекту	Опис	Вимірювані показники	Спосіб вимірювання	Припущення
A. Глобальна ціль	Підвищення ефективності продажу автозапчастин та популяризація чіп-тюнінгу через онлайн-сервіс.	Зростання кількості клієнтів, збільшення прибутку, підвищення впізнаваності бренду.	Аналіз статистики продажів, кількість відвідувань сайту, відгуки користувачів.	Наявність попиту на автозапчастини та тюнінг; стабільна робота платформи.
B. Очікувані результати	Функціонуючий сайт з базою товарів, системою замовлень та консультацій.	Наявність онлайн-каталогу, кількість замовлень, відсоток повторних клієнтів.	Перевірка роботи сайту, звіти системи аналітики, опитування користувачів.	Підтримка технічної інфраструктури, наявність постачальників.
C. Продукти проекту	Вебсайт з адаптивним дизайном, каталогом автозапчастин, модулем замовлення чіп-тюнінгу.	Розроблений та протестований сайт, готовий до запуску.	Внутрішнє тестування функцій, відгуки тестових користувачів.	Достатнє фінансування, наявність розробників і фахівців.
D. Діяльність	Аналіз ринку, розробка дизайну, створення бази даних, програмування, тестування, запуск.	Завершення кожного етапу в запланований термін.	Проміжні звіти, контрольні списки виконання завдань.	Злагоджена робота команди, своєчасне отримання технічних ресурсів.

Проектований веб-застосунок «NextLevel» передбачає такі функції:

- пошук прошивок за маркою, моделлю, типом двигуна та іншими параметрами;
- багатокритеріальна фільтрація для уточнення результатів підбору;
- автоматична перевірка сумісності прошивки з параметрами автомобіля;
- рекомендаційна система для пропозиції оптимальних варіантів;
- відображення технічних характеристик і прогнозованого приросту потужності;
- збереження обраних конфігурацій у профілі користувача;
- розрахунок очікуваних змін технічних показників після чіп-тюнінгу.

Переваги використання веб-конфігуратора:

- зручність доступу з будь-якого пристрою, підключеного до мережі Інтернет;
- економія часу на аналіз технічної документації;
- мінімізація ризику помилки при виборі прошивки;
- структуроване подання інформації;
- підвищення рівня обґрунтованості прийнятого рішення.

1.2 Визначення потенційних конкурентних переваг програмного продукту

Аналіз існуючих онлайн-сервісів з підбору прошивок для чіп-тюнінгу показує, що більшість із них або орієнтовані виключно на професійні СТО, або не забезпечують достатнього рівня автоматизації процесу підбору.

Основними недоліками наявних рішень є:

- відсутність повної автоматичної перевірки сумісності;
- обмежені можливості багатокритеріальної фільтрації;
- відсутність персоналізованих рекомендацій;

- складний або перевантажений інтерфейс;
- орієнтація на вузький перелік моделей або конкретного постачальника.

Веб застосунок «NextLevel» серед конкурентів буде виділятися такими перевагами:

- Автоматизація процесу підбору. Система забезпечуватиме автоматичний аналіз введених параметрів автомобіля та формування переліку лише сумісних прошивок.
- Багатокритеріальна фільтрація. Передбачено реалізацію гнучкої системи фільтрів, що враховує технічні, експлуатаційні та індивідуальні параметри вибору.
- Рекомендаційна система. Алгоритми рекомендацій дозволять пропонувати користувачу оптимальні варіанти залежно від його цілей (збільшення потужності, економія пального, збалансований режим тощо).
- Інтуїтивний інтерфейс. Інтерфейс буде спроектовано з урахуванням принципів UX/UI-дизайну для забезпечення простоти використання навіть для не підготовлених користувачів.
- Незалежність від конкретного постачальника. Застосунок не буде обмежений асортиментом одного сервісного центру чи магазину, що забезпечить об'єктивність підбору.

Серед існуючих сервісів для підбору прошивок чіп-тюнінгу можна виділити кілька популярних платформ, що використовуються автомобільними майстернями та приватними користувачами. До найбільш відомих належать ECM Titanium Database, RaceChip та TuningFiles.

ECM Titanium Database (рис. 1.1) є професійним програмним забезпеченням для роботи з прошивками ECU. Воно надає доступ до великої бази даних карт двигуна та дозволяє виконувати детальний аналіз і редагування

параметрів прошивки. Однак даний інструмент орієнтований переважно на професійних фахівців і потребує спеціальних знань для використання.

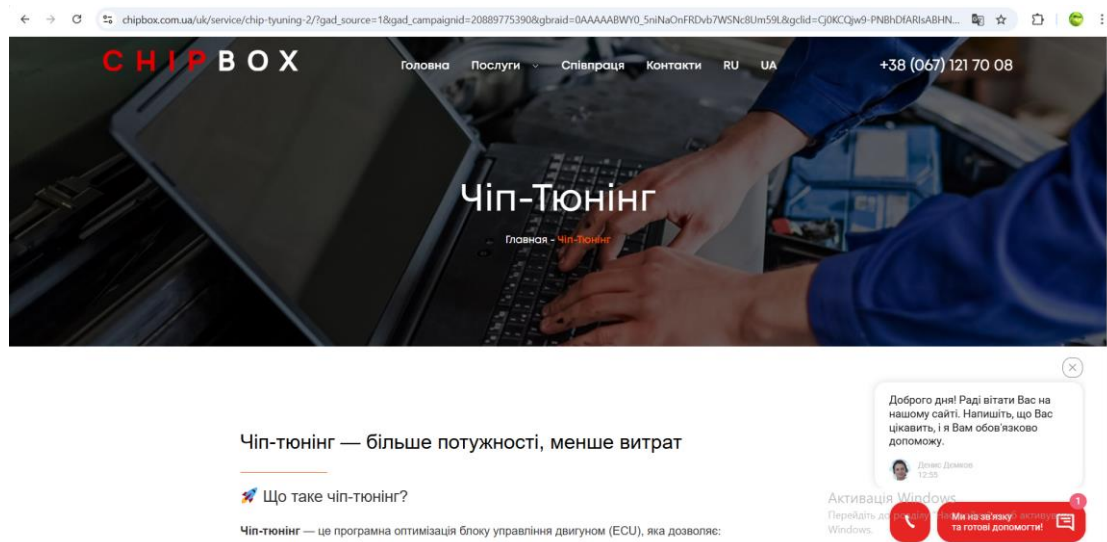


Рисунок 1.1 – Зовнішній вигляд головної сторінки сайту ПО ЕСМ Titanium

Джерело: [1]

Chipbox (рис. 1.2) пропонує готові рішення для збільшення потужності автомобіля за допомогою спеціальних модулів або налаштувань. Сервіс має зручний інтерфейс для користувачів, але можливості гнучкого підбору параметрів і детальної перевірки сумісності є обмеженими. Інтерфейс сайту є зручним та інтуїтивно зрозумілим: користувач може швидко знайти інформацію про свій автомобіль, обравши марку, модель і тип двигуна. Також ресурс містить приклади виконаних робіт, відгуки клієнтів і контактні дані для запису на послуги.

Сайт виконує не лише інформаційну, а й маркетингову функцію - він спрямований на залучення клієнтів, демонструючи переваги чіп-тюнінгу та експертність компанії у цій сфері. З функціональної точки зору сайт виконує одночасно інформаційну та комунікаційну ролі, забезпечуючи взаємодію між компанією та потенційними клієнтами через форму зворотного зв'язку та

контактні дані. Крім того, ресурс має виражену маркетингову спрямованість, оскільки презентує конкурентні переваги послуг і підкреслює експертність компанії у сфері автомобільного тюнінгу.

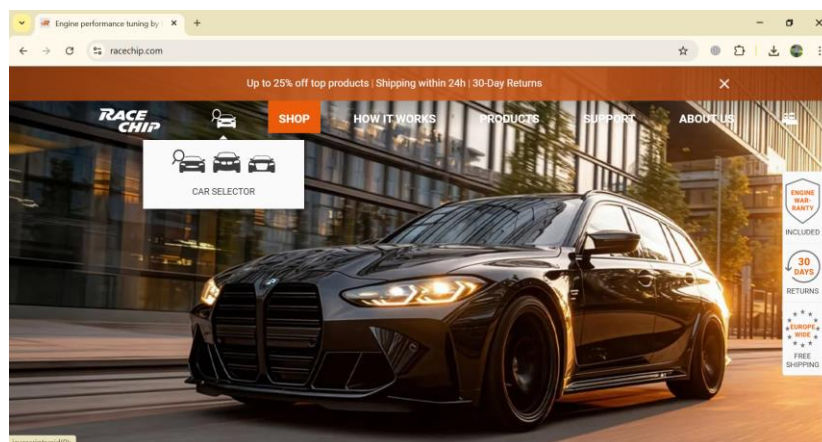


Рисунок 1.2 – Зовнішній вигляд головної сторінки сайту RaceChip

Джерело: [2]

TuningFiles (рис. 1.3) представляє онлайн-платформу для пошуку та завантаження файлів прошивок. Сервіс дозволяє знайти потрібну прошивку для конкретної моделі автомобіля, проте не забезпечує автоматичного аналізу сумісності та рекомендацій щодо оптимального варіанту.

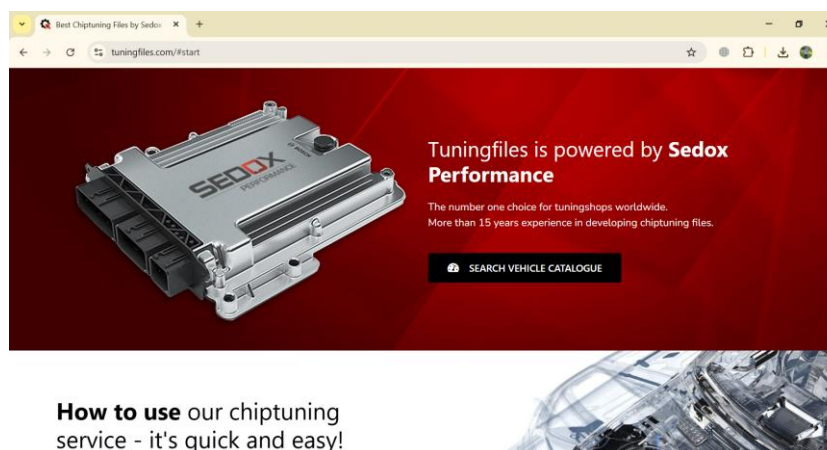


Рисунок 1.3 – Зовнішній вигляд головної сторінки сайту TuningFiles

Джерело: [3]

Для більш наочного порівняння основних характеристик розглянутих сервісів та розроблюваного веб-застосунку «NextLevel» проведемо порівняльний аналіз (табл. 1.2).

Таблиця 1.2 Порівняльний аналіз характеристик відомих сервісів

Характеристика	ECM Titanium	RaceChip	TuningFiles	NextLevel
Онлайн-використання	частково	так	так	так
Автоматична перевірка сумісності	ні	частково	ні	так
Багатокритеріальна фільтрація	ні	ні	частково	так
Рекомендаційна система	ні	ні	ні	так
Простий інтерфейс для користувача	ні	так	частково	так
Незалежність від постачальника	ні	ні	частково	так

На основі проведеного аналізу можна зробити висновок, що більшість існуючих рішень або орієнтовані на вузьку професійну аудиторію, або не забезпечують достатнього рівня автоматизації процесу підбору прошивок.

Розроблюваний веб-застосунок «NextLevel» має такі потенційні конкурентні переваги:

- автоматизований підбір сумісних прошивок на основі характеристик автомобіля;
- використання багатокритеріальної фільтрації для точнішого пошуку;
- реалізація рекомендаційної системи, яка пропонує оптимальні варіанти;
- простий та інтуїтивно зрозумілий інтерфейс;
- можливість використання сервісу без спеціальних технічних знань;
- незалежність від конкретного постачальника прошивок.

Таким чином, веб-застосунок «NextLevel» дозволяє підвищити ефективність підбору прошивок та зробити процес чіп-тюнінгу більш доступним для широкого кола користувачів.

1.3 Постановка завдання на кваліфікаційну роботу

Основним завданням кваліфікаційної роботи є вирішення проблеми автоматизованого підбору сумісних прошивок для чіп-тюнінгу автомобілів.

Вхідні дані:

- дані користувача (ім'я, email, пароль);
- характеристики автомобіля (марка, модель, рік випуску, тип двигуна, об'єм, тип пального тощо);
- база даних прошивок (назва, опис, параметри сумісності, технічні показники, прогнозований приріст потужності).

Вихідні дані:

- перелік сумісних прошивок;
- рекомендовані варіанти;
- детальна інформація про обрану прошивку;
- збережена конфігурація в профілі користувача.

Основні етапи розв'язання задачі:

- розробити алгоритм багатокритеріального підбору прошивок;
- спроектувати структуру бази даних для зберігання характеристик автомобілів та прошивок;
- реалізувати RESTful API для обробки та передачі даних;
- розробити рекомендаційну систему;
- спроектувати UI/UX-дизайн веб-застосунк.

Функціональні вимоги:

Реєстрація та авторизація користувачів

- можливість створення облікового запису;
- авторизація через email/пароль або соціальні мережі;
- відновлення пароля.
- Управління профілем користувача

- перегляд та редагування особистих даних;
- додавання інформації про автомобіль (марка, модель, двигун).

Каталог послуг чіп-тюнінгу

- відображення доступних послуг;
- детальний опис кожної послуги (приріст потужності, зміни витрати пального тощо).

Багатокритеріальна система фільтрації

- фільтрація за маркою автомобіля;
- фільтрація за моделлю та типом двигуна;
- фільтрація за бажаними параметрами (потужність, економічність, ціна).
- комбінування декількох критеріїв одночасно.

Рекомендаційна система

- генерація персоналізованих рекомендацій на основі введених даних;
- врахування історії переглядів і вибору користувача;
- пропозиція оптимальних варіантів тюнінгу.

Нефункціональні вимоги:

Продуктивність

- час завантаження сторінки не більше 2–3 секунд;
- обробка запитів у режимі реального часу;
- підтримка одночасної роботи великої кількості користувачів.

Масштабованість

- можливість розширення функціоналу;
- підтримка зростання кількості користувачів;

Юзабіліті (зручність використання)

- інтуїтивно зрозумілий інтерфейс;
- адаптивний дизайн для мобільних пристроїв;
- мінімальна кількість дій для виконання основних операцій.

Реєстрація та авторизація користувачів

- можливість створення облікового запису;
- авторизація через email/пароль або соціальні мережі;
- відновлення пароля. управління профілем користувача
- перегляд та редагування особистих даних.
- додавання інформації про автомобіль (марка, модель, двигун) ;
- каталог послуг чіп-тюнінгу;
- відображення доступних послуг;
- детальний опис кожної послуги (приріст потужності, зміни витрати пального тощо).

Висновки до розділу 1

Підбір прошивок для чіп-тюнінгу автомобілів потребує врахування великої кількості технічних параметрів і перевірки їх сумісності. Автоматизація цього процесу дозволяє зменшити ризик помилок та підвищити ефективність прийняття рішень.

Веб-застосунок «NextLevel» має конкурентні переваги завдяки реалізації багатокритеріальної фільтрації, рекомендаційної системи, автоматичної перевірки сумісності та зручного інтерфейсу.

Наступний розділ роботи буде присвячений проектуванню архітектури системи, моделюванню даних і розробці алгоритмів функціонування веб-застосунку.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ «NEXTLEVEL»

2.1 Моделювання поведінки продукту

Для опису поведінки веб-застосунку «NextLevel» використано UML-діаграми, які дозволяють наочно представити взаємодію користувача із системою, а також внутрішню логіку виконання основних процесів.

Хеш-таблиця є структурою даних, що дозволяє ефективно знаходити значення за ключем. Для цього застосовується хеш-функція, яка перетворює ключ у відповідний індекс у таблиці.

Переваги:

- забезпечує дуже швидкий доступ до даних (у середньому $O(1)$);
- проста у використанні.

Недоліки:

- ефективність залежить від якості хеш-функції;
- потребує більше пам'яті порівняно з іншими структурами.

Класифікація — це підхід у машинному навчанні, спрямований на віднесення нових об'єктів до певних категорій на основі попереднього навчання моделі.

Переваги:

- ефективна у задачах, де складно сформулювати чіткі правила;
- може досягати високої точності при наявності достатньої кількості даних.

Недоліки:

- потребує значних обсягів даних для навчання;
- результати іноді важко пояснити.

Кластеризація передбачає об'єднання об'єктів у групи відповідно до їх схожості без попередньо заданих міток.

Переваги:

- дозволяє працювати з немаркованими даними;
- допомагає виявляти приховані структури та залежності.

Недоліки:

- результати залежать від обраного алгоритму та його налаштувань;
- інтерпретація отриманих груп може бути складною.

Лінійне програмування – це метод оптимізації, який застосовується для знаходження найкращого розв’язку задач із лінійними обмеженнями та цільовою функцією.

Переваги:

- ефективно для задач із чітко заданими умовами;
- дозволяє отримати оптимальне рішення.

Недоліки:

- може бути складним при великій кількості змінних;
- не підходить для задач із нелінійними залежностями.

Основним актором системи є користувач, який взаємодіє з веб-застосунком через інтерфейс браузера. Додатковими акторами можуть виступати адміністратор системи та зовнішні сервіси (платіжні системи, API).

Варіанти використання:

- реєстрація та авторизація користувача;
- введення характеристик автомобіля;
- пошук прошивок;
- застосування фільтрів;
- отримання рекомендацій;
- перегляд детальної інформації;
- збереження конфігурації;
- оформлення замовлення.

Діаграма варіантів використання демонструє, що ключовою функцією системи є автоматизований підбір сумісних прошивок, навколо якого побудовані всі інші процеси.

Нижче подано діаграму взаємодії користувачів із системою.

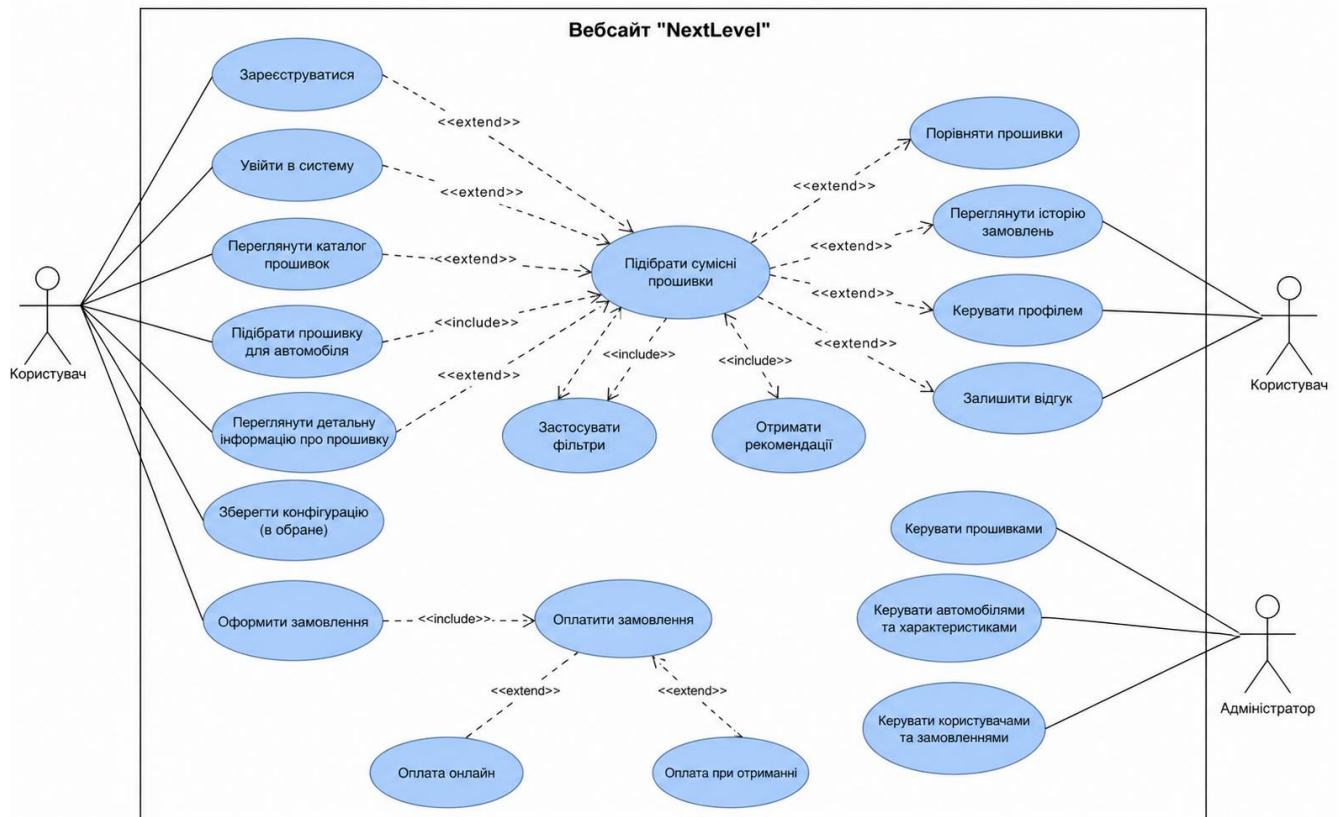


Рисунок 2.1 – Діаграма Use Case сайту NextLevel.

Джерело: розроблено автором

Таблиця 2.1 – Розбиття Use Case на Slice

Назва Slice	Короткий опис	Цінність для користувача / бізнесу	Пріоритет
Пошук запчастин (Search Parts)	Забезпечує можливість швидкого пошуку потрібної деталі за параметрами авто.	Економія часу користувача, зручність у виборі.	Високий
Перегляд товару (View Item)	Дозволяє переглядати фото, опис і характеристики запчастини.	Збільшує довіру клієнта до продукту.	Високий
Оформлення замовлення	Надає можливість додати товар у кошик і створити замовлення.	Сприяє продажам і збільшенню прибутку.	Високий
Оплата онлайн (Online Payment)	Реалізує процес оплати через інтегровану систему платежів.	Покращує досвід користувача, прискорює обробку замовлення.	Високий
Запит консультації (Request Consultation)	Дозволяє клієнту отримати поради спеціаліста.	Покращує сервіс та довіру.	Середній
Керування каталогом (Manage Catalog)	Функція адміністратора для додавання, редагування та видалення товарів.	Підтримує актуальність бази даних.	Середній
Обробка замовлень (Process Orders)	Адміністратор переглядає нові замовлення та змінює їх статус.	Забезпечує оперативну роботу магазину.	Високий

Діаграма діяльності відображає послідовність дій користувача при підборі прошивки:

1. Початок роботи із системою.
2. Введення параметрів автомобіля.
3. Обробка введених даних.
4. Перевірка сумісності прошивок.
5. Фільтрація результатів.
6. Формування списку варіантів.
7. Генерація рекомендацій.
8. Відображення результатів.
9. Вибір прошивки користувачем.

10. Завершення процесу.

У разі некоректного введення даних передбачено повернення до етапу введення.

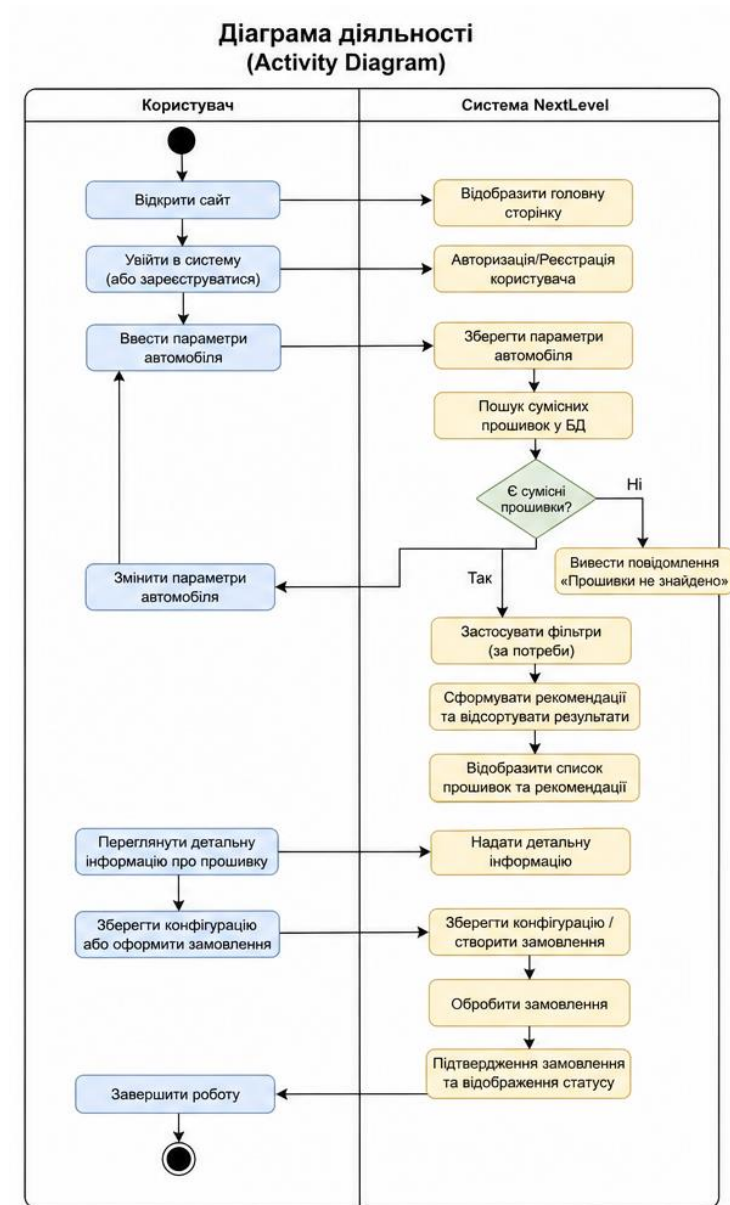


Рисунок 2.2 – Діаграма діяльності.

Джерело: розроблено автором

Діаграма послідовності описує взаємодію між компонентами системи:
Користувач → Frontend → Backend → База даних.

Основний сценарій:

1. Користувач вводить параметри автомобіля.
2. Frontend передає запит до Backend через API.
3. Backend виконує обробку запиту.
4. Здійснюється запит до бази даних.
5. Отримані дані проходять перевірку сумісності.
6. Застосовується алгоритм рекомендацій.
7. Результат повертається на Frontend.
8. Користувач отримує список відповідних прошивок.

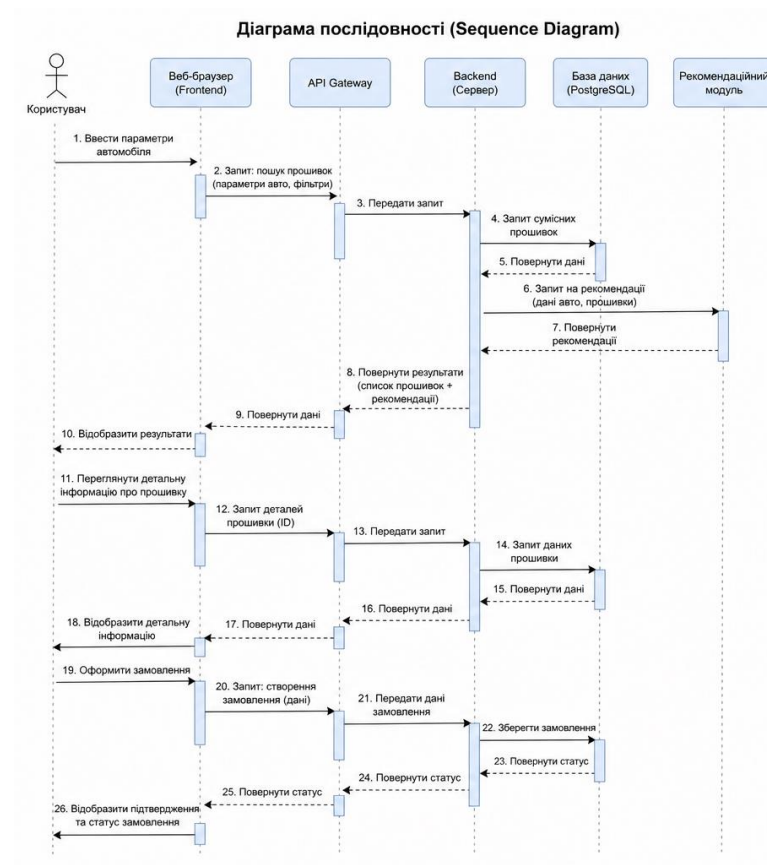


Рисунок 2.3 – Діаграма послідовності

Джерело: розроблено автором

2.2 Моделювання архітектури та структури продукту

Для побудови веб-застосунку «NextLevel» використано модульний підхід, що забезпечує масштабованість та зручність супроводу системи.

Основні класи системи:

- User (користувач)
 - id, name, email, password
- Car (автомобіль)
 - brand, model, year, engine_type, fuel_type
- Firmware (прошивка)
 - id, name, compatibility_params, power_gain
- Recommendation
 - score, type
- Order
 - id, user_id, firmware_id, status

Зв'язки між класами:

- User має багато Order;
- Car використовується для підбору Firmware;
- Recommendation пов'язана з Firmware;
- Order містить обрану прошивку.

Система складається з таких компонентів:

- Frontend (інтерфейс користувача)- реалізує взаємодію з користувачем.
- Backend (серверна логіка) - обробляє запити, реалізує алгоритми.
- Database (база даних)- зберігає інформацію про користувачів, авто та прошивки.
- Recommendation Engine - відповідає за формування рекомендацій.
- API Layer - забезпечує зв'язок між frontend і backend.

Для візуалізації архітектуру вебсайту було використано Модель С4. Вона складається з наступних рівнів.

Рівень 1 — Контекст системи. Система «NextLevel» взаємодіє з:

- користувачем;
- платіжними сервісами;
- зовнішніми базами даних (за потреби).

Рівень 2 — Контейнери. Система складається з:

- веб-клієнта (Frontend);
- серверної частини (Backend);
- бази даних;
- модуля рекомендацій.

Рівень 3 — Компоненти. Backend поділяється на:

- модуль обробки запитів;
- модуль логіки підбору;
- модуль рекомендацій;
- модуль авторизації.

Рівень 4 — Код. Реалізація здійснюється за допомогою:

- REST API;
- структурованих класів;
- ORM для роботи з БД.

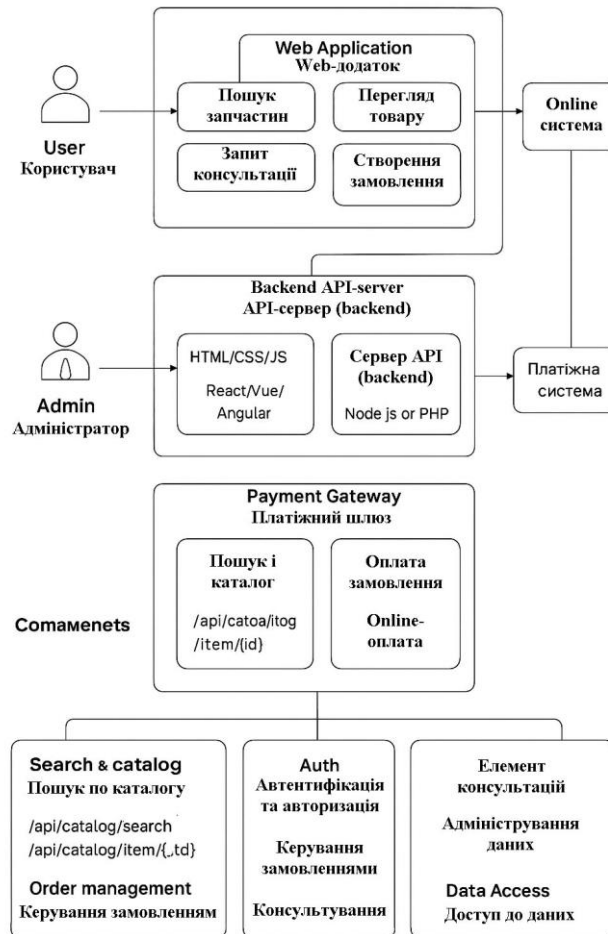


Рисунок 2.4 – Діаграма C4

2.3 Опис алгоритмів та математичних моделей

Алгоритм пошуку в глибину (DFS) застосовується для обходу графів або дерев шляхом послідовного переходу вздовж однієї гілки. Він рухається вперед доти, поки не досягне кінцевого вузла або не знайде потрібний елемент. Після цього відбувається повернення назад і перевірка інших можливих шляхів, поки не буде знайдено результат або не будуть перевірені всі варіанти.

Переваги:

- дозволяє швидко знайти потрібний елемент у разі його наявності;
- має просту логіку реалізації.

Недоліки:

- не завжди ефективний при необхідності обійти всі вузли;

- може витрачати час на дослідження зайвих гілок.

Алгоритм пошуку в ширину (BFS) здійснює обхід структури рівень за рівнем. Спочатку аналізуються всі вузли поточного рівня, і лише після цього відбувається перехід до наступного. Для організації процесу використовується черга.

Переваги:

- забезпечує повний перегляд усіх елементів (якщо вони існують);
- підходить для визначення таких характеристик, як глибина або ширина дерева.

Недоліки:

- може працювати повільніше при пошуку конкретного елемента;
- здатен обробляти зайві вузли, які не мають значення для результату.

Алгоритм підбору прошивок. Основний алгоритм базується на багатокритеріальному аналізі:

1. Отримання параметрів автомобіля;
2. Вибір всіх можливих прошивок;
3. Фільтрація за обов'язковими параметрами;
4. Перевірка сумісності;
5. Формування списку допустимих варіантів.

Алгоритм вибору комплектуючих базується на підході пошуку в ширину, де дані поетапно відсіюються за заданими критеріями. Основні етапи роботи алгоритму:

1. Отримання вхідних даних від користувача: тип комплектуючого, необхідні характеристики (за потреби) та вже вибрані елементи.
2. Виконання запиту до бази даних для отримання списку товарів відповідної категорії.
3. Первинна фільтрація — виключення варіантів, що не відповідають заданим параметрам.

4. Аналіз уже обраних компонентів для визначення характеристик, що впливають на сумісність.
5. Додаткова перевірка списку з урахуванням сумісності.
6. Повернення користувачеві остаточного списку відповідних комплектуючих.

Алгоритм багатокритеріальної фільтрації. Для кожної прошивки обчислюється відповідність параметрам:

- числові параметри — перевірка діапазону;
- текстові — співпадіння;
- логічні — true/false відповідність.

Результат — множина сумісних варіантів.

- **Алгоритм рекомендаційної системи.** Використовується підхід оцінювання:

$Score = w_1 \cdot P + w_2 \cdot E + w_3 \cdot C$, де P — приріст потужності, E — економія пального, C — сумісність, w — вагові коефіцієнти.

Користувачу пропонуються варіанти з максимальним значенням $Score$.

Обрані алгоритми забезпечують:

- простоту реалізації;
- високу швидкість роботи;
- можливість масштабування;
- гнучкість налаштування.

Алгоритми реалізуються на серверній частині та використовуються при кожному запиті користувача для формування результатів підбору.

Висновки до розділу 2

У другому розділі було виконано проектування веб-застосунку «NextLevel». Розроблено моделі поведінки системи, що відображають взаємодію користувача з програмним продуктом.

Побудовано архітектуру системи, яка включає клієнтську та серверну частини, базу даних та модуль рекомендацій. Це забезпечує масштабованість і зручність підтримки.

Описані алгоритми багатокритеріального підбору та рекомендацій дозволяють автоматизувати процес вибору прошивок і підвищити точність результатів.

Отримані результати створюють основу для реалізації програмного продукту, що буде розглянуто у наступному розділі.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ПРОГРАМНОГО ПРОДУКТУ «NEXTLEVEL»

3.1 Технологічні рішення програмного продукту

Після завершення етапу проєктування було виконано практичну реалізацію веб-застосунку «NextLevel», призначеного для автоматизованого підбору прошивок чіп-тюнінгу автомобілів із використанням багатокритеріальної фільтрації та рекомендаційного механізму.

Програмний продукт реалізовано за сучасною клієнт-серверною архітектурою, що передбачає поділ системи на окремі функціональні частини:

- клієнтська частина (Frontend);
- серверна частина (Backend);
- база даних;
- модуль рекомендаційної системи.

Такий підхід забезпечує масштабованість, зручність підтримки та можливість подальшого розширення функціоналу.

Frontend-частина реалізована як адаптивний вебінтерфейс, доступний через сучасні браузері на персональних комп'ютерах, планшетах та мобільних пристроях.

Інтерфейс користувача включає:

- головну сторінку сервісу;
- форму введення параметрів автомобіля;
- сторінку результатів пошуку;
- сторінку детального перегляду прошивки;
- особистий кабінет користувача;
- сторінку оформлення замовлення.

Інтерфейс побудовано з урахуванням принципів UX/UI-дизайну: простота використання, логічна структура елементів, швидкий доступ до основних функцій.

Backend забезпечує бізнес-логіку роботи застосунку та виконує:

- обробку запитів користувача;
- авторизацію та реєстрацію;
- пошук сумісних прошивок;
- застосування фільтрів;
- формування рекомендацій;
- збереження замовлень;
- взаємодію з базою даних.

Для обміну даними між клієнтом і сервером використовується REST API.

Для зберігання інформації використовується реляційна база даних.

Основними таблицями є:

- користувачі;
- автомобілі;
- прошивки;
- характеристики;
- конфігурації;
- замовлення.

База даних забезпечує швидкий пошук інформації, збереження історії дій користувачів та структурованість даних.

При введенні параметрів автомобіля система виконує:

- перевірку коректності введених даних;
- пошук відповідних моделей у базі;
- вибір сумісних прошивок;
- фільтрацію результатів;
- ранжування за рейтингом;

- виведення оптимальних варіантів.

Для підвищення якості вибору реалізовано рекомендаційну систему, яка враховує:

- тип двигуна;
- ціль користувача (потужність / економія / баланс);
- популярність прошивки;
- рейтинг користувачів;
- сумісність параметрів.

На етапі конструювання було виконано:

- створення структури проєкту;
- розробку інтерфейсних компонентів;
- підключення бази даних;
- реалізацію API;
- програмування логіки підбору;
- налаштування безпеки доступу;
- оптимізацію продуктивності.

У процесі реалізації веб-застосунку було використано ряд ключових програмних рішень, які забезпечують функціональність системи. Розглянемо основні фрагменти коду та їх призначення.

Обробка авторизації користувача:

```
@app.post("/login", response_class=HTMLResponse)
async def login(request: Request, email: str = Form(...), password:
str = Form(...)):
    conn = sqlite3.connect("database.db")
    cur = conn.cursor()
    cur.execute(
        "SELECT name, email, role FROM users WHERE email=? AND
password=?",
        (email, password)
```

```

)
user = cur.fetchone()
conn.close()
if user:
    request.session["user"] = {
        "name": user[0],
        "email": user[1],
        "role": user[2]
    }
return RedirectResponse("/profile", status_code=302)
return templates.TemplateResponse(
    request=request,
    name="login.html",
    context={"error": "Невірний email або пароль"}
)

```

Даний фрагмент коду реалізує процес авторизації користувача. Введені дані перевіряються у базі даних, після чого у разі успіху створюється сесія користувача. Це дозволяє зберігати інформацію про користувача під час роботи із сайтом.

Реєстрація користувача:

```

cur.execute("""
    INSERT INTO users (name, email, password)
    VALUES (?, ?, ?)
""", (name, email, password))

```

Цей фрагмент відповідає за додавання нового користувача до бази даних. Використання параметризованого запиту забезпечує захист від SQL-ін'єкцій.

Отримання замовлень користувача:

```

cur.execute(
    "SELECT firmware_name, status FROM orders WHERE user_name=?",
    (user["email"],)
)
orders = cur.fetchall()

```

Запит дозволяє отримати всі замовлення, пов'язані з конкретним користувачем. Дані відображаються у кабінеті користувача у вигляді списку.

Обробка покупки товару:

```
cur.execute("""
    INSERT INTO orders (user_name, firmware_name, status)
    VALUES (?, ?, ?)
    """, (user["email"], product, "нове"))
```

Фрагмент реалізує створення нового замовлення. Після натискання кнопки "Купити" інформація про товар додається до бази даних із відповідним статусом.

Динамічне відображення товарів (HTML шаблон):

```
{% for p in products %}
<div class="card">
    
    <h3>{{ p.name }}</h3>
    <p>{{ p.price }}</p>
</div>
{% endfor %}
```

Цей шаблон використовує Jinja2 для відображення списку товарів. Дані передаються із серверної частини, що дозволяє формувати сторінку динамічно.

Маршрутизація категорій товарів:

```
@app.get("/category/{name}")
```

```
async def category_page(request: Request, name: str):
```

Даний маршрут дозволяє відкривати різні категорії товарів (турбіни, інтеркулери тощо) залежно від параметра в URL. Це забезпечує гнучкість структури сайту.

Розглянуті фрагменти коду демонструють основні принципи роботи веб-застосунку: обробку запитів користувача, взаємодію з базою даних, управління сесіями та динамічне формування інтерфейсу. Використані підходи забезпечують надійність, безпеку та масштабованість програмного продукту.

Для зберігання даних у веб-застосунку «NextLevel» використовується реляційна база даних, що забезпечує структурованість, цілісність та ефективну обробку інформації. Основною метою бази даних є зберігання інформації про користувачів, автомобілі та прошивки для їх подальшого аналізу та підбору.

Основними сутностями бази даних є:

- Користувач (User). Містить інформацію про користувача веб-застосунку, зокрема електронну пошту, пароль, ім'я та збережені конфігурації. Дана сутність використовується для авторизації, персоналізації та збереження історії взаємодії із системою.
- Автомобіль (Car). Містить характеристики транспортного засобу, необхідні для підбору прошивок, такі як марка, модель, рік випуску, тип двигуна, об'єм двигуна, тип пального та інші параметри.
- Прошивка (Firmware). Містить інформацію про доступні прошивки для чіп-тюнінгу, включаючи назву, опис, технічні характеристики, параметри сумісності, а також очікуваний приріст потужності та інші показники.
- Конфігурація (Configuration). Містить інформацію про вибрані користувачем параметри автомобіля та відповідні підібрані прошивки. Дозволяє зберігати результати підбору для подальшого використання або порівняння.
- Характеристика (Attribute). Описує окремі параметри як автомобілів, так і прошивок (наприклад, тип двигуна, стандарт екології, наявність турбіни тощо). Містить назву, тип даних (числовий, текстовий, логічний) та опис.
- Параметр сумісності (Compatibility Parameter). Є розширенням характеристики та використовується для перевірки відповідності між автомобілем і прошивкою. Дана сутність дозволяє реалізувати гнучку систему порівняння значень параметрів.

Сутності бази даних пов'язані між собою таким чином:

- один користувач може мати багато конфігурацій;

- одна конфігурація пов'язана з одним автомобілем і може містити декілька прошивок;
- прошивки пов'язані з характеристиками та параметрами сумісності;
- характеристики використовуються як для автомобілів, так і для прошивок.

На рисунку 3.1 представлено ER-діаграму веб-застосунку «NextLevel», яка відображає основні сутності системи, їх атрибути та зв'язки між ними. Дана модель демонструє структуру бази даних і використовується як основа для подальшої реалізації сховища даних.

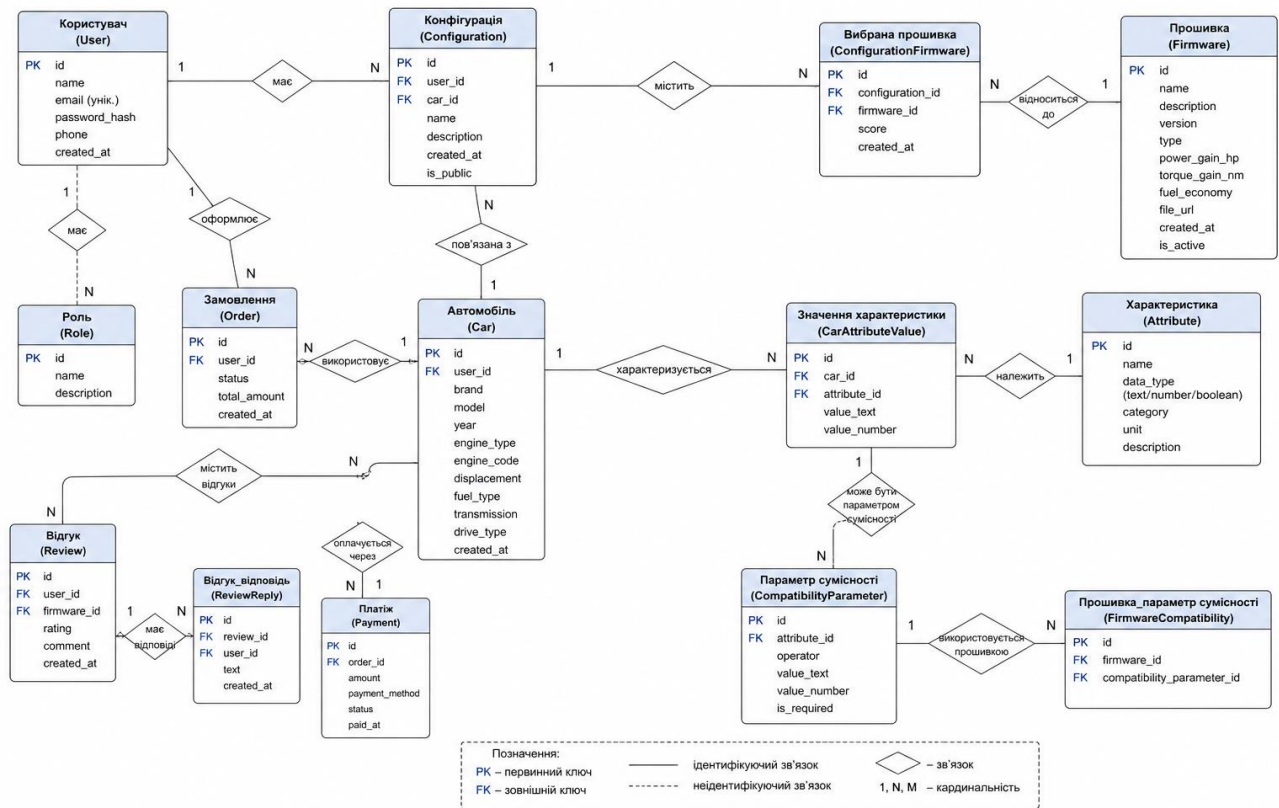


Рисунок 3.1 – ER-діаграма веб-застосунку «NextLevel»

3.2 Тестування програмного продукту

Після реалізації веб-застосунку було проведено комплексне тестування для перевірки працездатності всіх модулів системи.

За допомогою функціонального тестування було перевірено:

- реєстрацію користувача;
- вхід до системи;
- пошук прошивок;
- застосування фільтрів;
- відображення результатів;
- збереження конфігурацій;
- створення замовлення;
- роботу особистого кабінету.

За результатами тестування всі основні функції працюють коректно.

З використанням тестування інтерфейсу було перевірено коректність відображення сторінок у браузерях:

- Google Chrome;
- Mozilla Firefox;
- Microsoft Edge.

Також протестовано адаптивність інтерфейсу на різних роздільних здатностях екрана.

В рамках навантажувального тестування було змодельовано одночасну роботу декількох користувачів із системою. Серверна частина стабільно обробляла запити без втрати продуктивності.

При тестуванні безпеки було перевірено:

- захист авторизації;
- перевірку введених даних;
- обмеження доступу до адміністративної частини;

- захист від SQL-ін'єкцій;
- шифрування паролів.

За результатами перевірки критичних помилок виявлено не було.
Програмний продукт придатний до практичного використання.

3.3 Практична реалізація проєкту

Веб-застосунок «NextLevel» призначений для власників автомобілів, автосервісів та спеціалістів з чіп-тюнінгу.

Для використання сервісу необхідно:

1. Перейти на головну сторінку сайту;
2. Зареєструватися або увійти до системи;
3. Ввести характеристики автомобіля;
4. Обрати параметри підбору;
5. Переглянути список рекомендованих прошивок;
6. Ознайомитися з деталями;
7. Зберегти конфігурацію або оформити замовлення.

Застосування сервісу дозволяє:

- значно скоротити час підбору;
- знизити ризик помилкового вибору;
- автоматизувати процес аналізу сумісності;
- отримати персональні рекомендації;
- швидко оформити замовлення онлайн.

У майбутньому систему можна розширити шляхом:

- інтеграції з автосервісами;
- мобільного застосунку;
- підключення онлайн-оплати;
- використання штучного інтелекту;
- міжнародної локалізації.

У результаті виконання кваліфікаційної роботи було розроблено повнофункціональний веб-застосунок «NextLevel», який призначений для підбору, замовлення та придбання автомобільних прошивок і комплектуючих для тюнінгу.

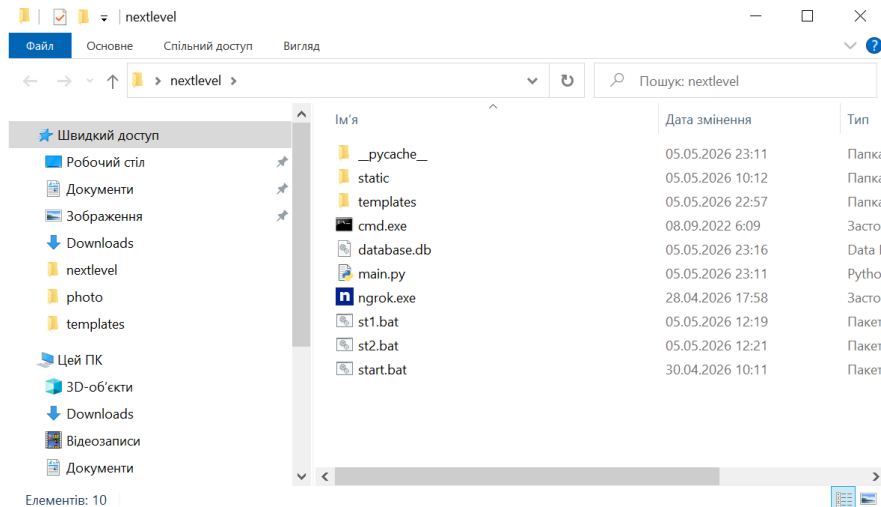


Рисунок 3.2 – Структура папок

Джерело: розроблено автором

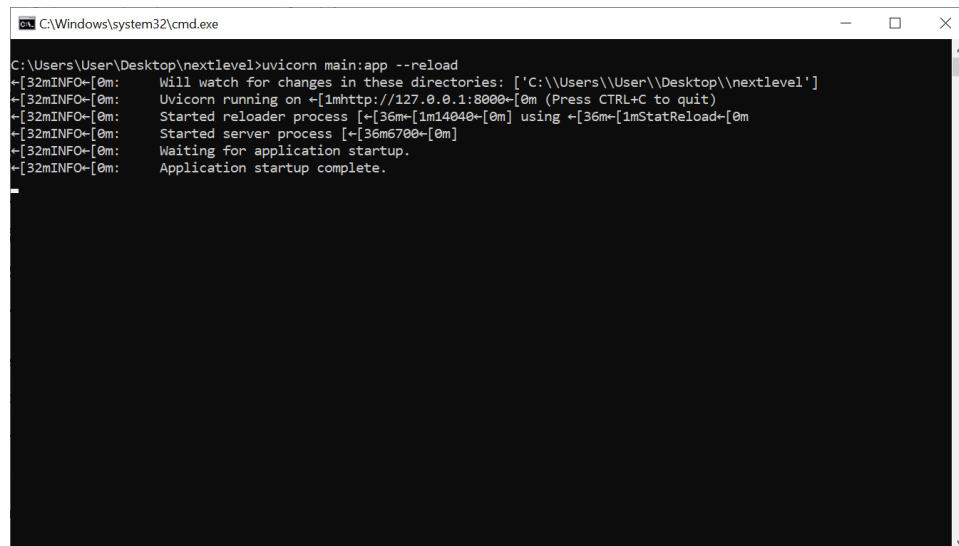


Рисунок 3.3 – Запуск серверної частини Unicorn

Джерело: розроблено автором

Розроблений програмний продукт реалізовано з використанням сучасних технологій веб-розробки. Серверна частина побудована на мові програмування Python із використанням фреймворку FastAPI, що забезпечує високу продуктивність, асинхронну обробку запитів та зручність розробки. Для зберігання даних використовується реляційна база даних SQLite, яка забезпечує простоту розгортання та достатню швидкодію для локального використання.

Зовнішній вигляд головної сторінки веб-застосунку наведено на рисунку 3.4. Головна сторінка виконує роль початкової точки взаємодії користувача із системою. Через неї користувач отримує доступ до основних можливостей сервісу, зокрема до реєстрації, авторизації, пошуку автомобільних прошивок, перегляду товарів для тюнінгу та оформлення замовлень. Інтерфейс головної сторінки має бути зрозумілим для користувача, оскільки саме з нього починається робота із застосунком.

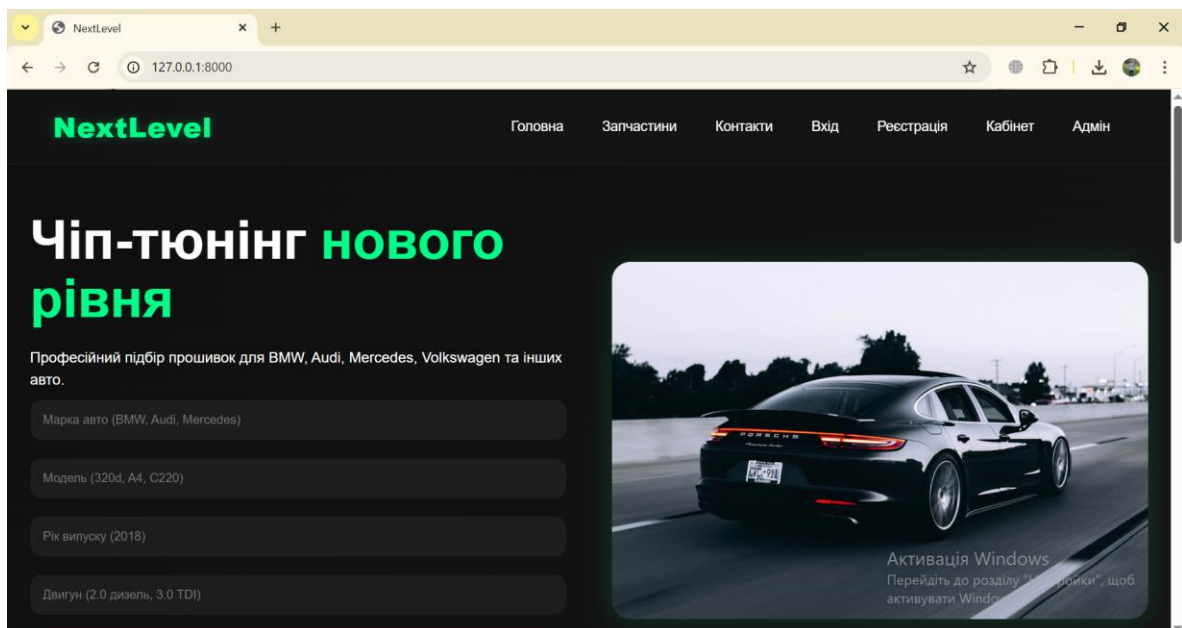


Рисунок 3.4 – Зовнішній вигляд головної сторінки сайту

Джерело: розроблено автором

Клієнтська частина реалізована за допомогою HTML, CSS та шаблонізатора Jinja2, що дозволяє динамічно відображати дані та забезпечує зручну взаємодію користувача із системою.

Функціональні можливості веб-застосунку включають:

- реєстрацію та авторизацію користувачів;
- підбір автомобільних прошивок за параметрами (марка, модель, рік тощо);
- перегляд каталогу запчастин для тюнінгу (турбіни, інтеркулери, вихлопні системи);
- перехід до окремих категорій товарів із детальним списком позицій;
- оформлення замовлення як прошивок, так і фізичних товарів;
- перегляд історії замовлень у кабінеті користувача;
- адміністративну панель для керування даними.

На рисунку 3.5 представлено сторінку реєстрації користувача. Її призначення полягає у створенні облікового запису, необхідного для подальшого використання персоналізованих функцій системи. Після реєстрації користувач отримує можливість зберігати обрані конфігурації, оформлювати замовлення та переглядати історію власних дій у системі. Наявність механізму реєстрації також дозволяє розмежувати доступ до функцій для звичайних користувачів і адміністратора.

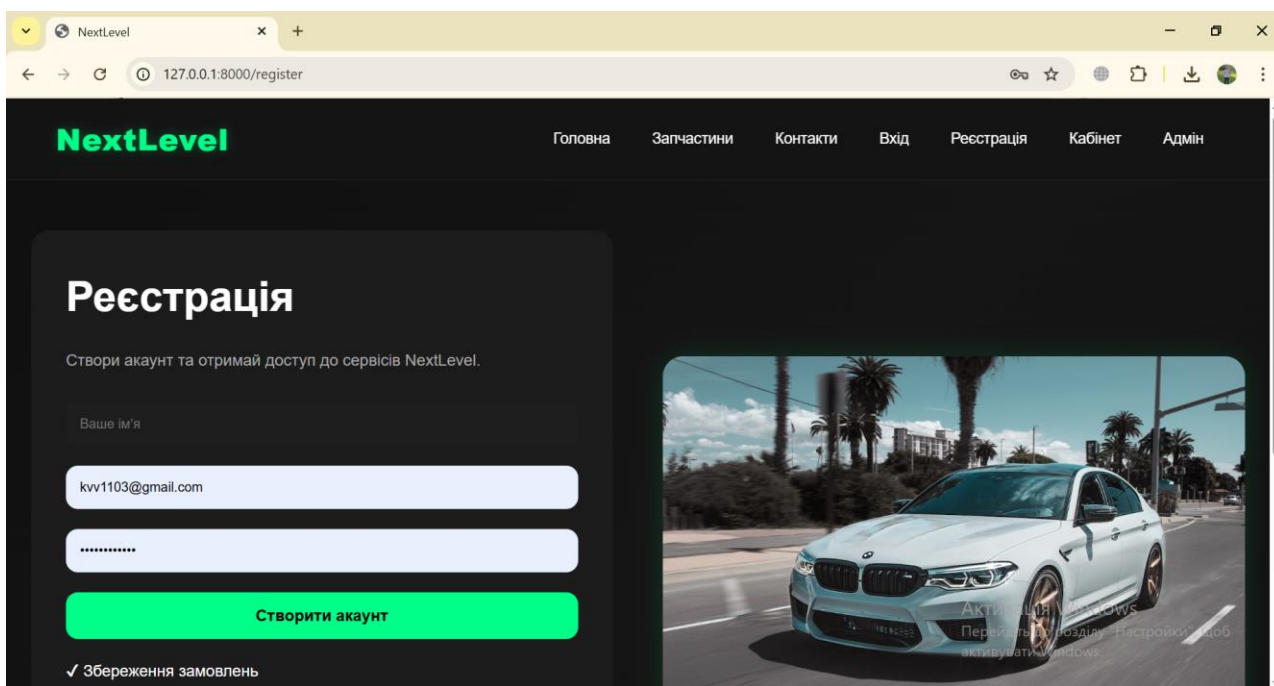


Рисунок 3.5 Сторінка реєстрації

Джерело: розроблено автором

На рисунку 3.6 зображено сторінку адміністратора. Адміністративна частина системи призначена для керування основними даними веб-застосунку, зокрема інформацією про автомобільні прошивки, товари для тюнінгу, замовлення та користувачів. Завдяки адміністративній панелі власник або адміністратор сервісу може оновлювати вміст системи без необхідності безпосереднього редагування програмного коду. Це підвищує зручність експлуатації застосунку та робить його більш придатним до практичного використання.

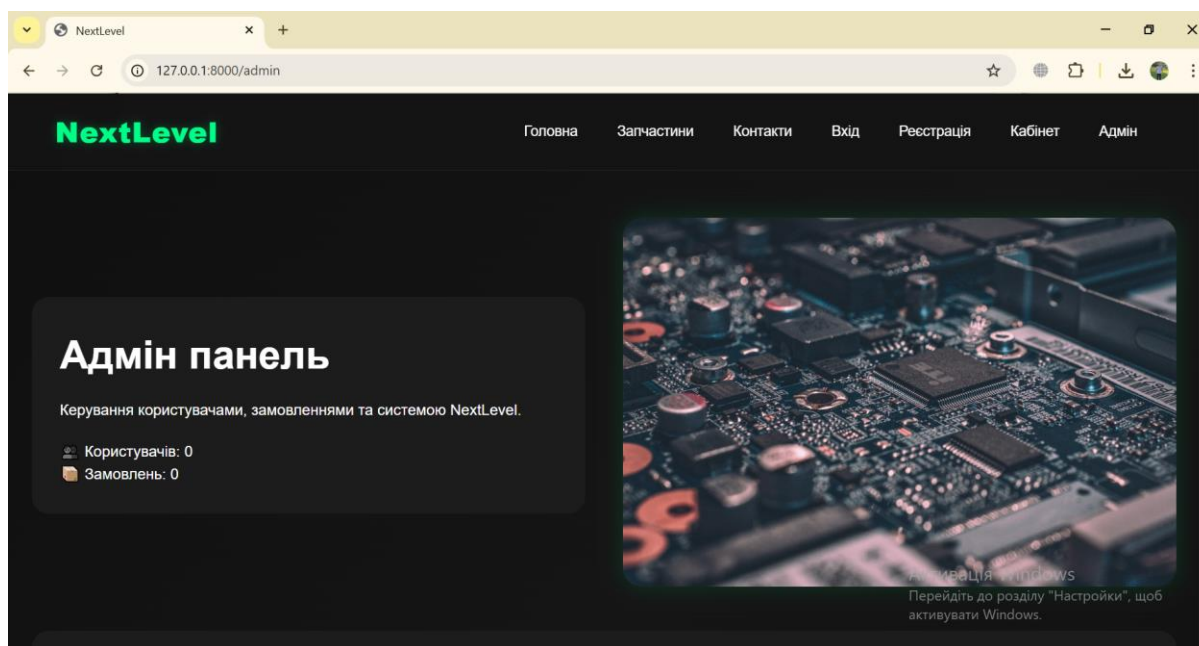


Рисунок 3.6 Сторінка адміністратора

Джерело: розроблено автором

Процес пошуку автомобільних прошивок відображено на рисунку 3.7. На цій сторінці користувач може ввести характеристики автомобіля, зокрема марку, модель, рік випуску та інші параметри, після чого система формує перелік рекомендованих прошивок. Такий підхід дозволяє автоматизувати процес підбору та зменшити ризик помилкового вибору. Замість ручного пошуку користувач отримує варіанти, які відповідають заданим параметрам, що скорочує час прийняття рішення та підвищує зручність роботи із сервісом.

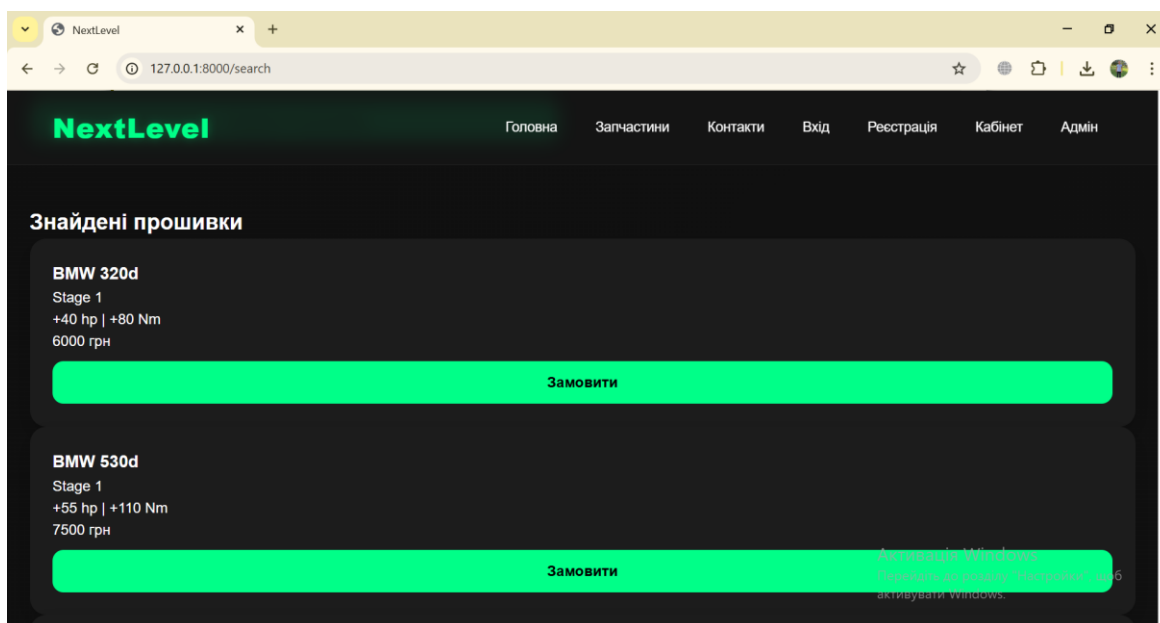


Рисунок 3.7 Процес пошуку прошивок

Джерело: розроблено автором

На рисунку 3.8 наведено сторінку запчастин для тюнінгу. Вона доповнює основну функціональність системи, оскільки користувач може не лише підібрати автомобільну прошивку, а й переглянути комплектуючі, які можуть бути використані для покращення технічних характеристик автомобіля. До таких товарів можуть належати турбіни, інтеркулери, вихлопні системи та інші елементи, пов'язані з модернізацією транспортного засобу. Це розширює можливості веб-застосунку та робить його більш комплексним рішенням для користувачів, зацікавлених у тюнінгу.

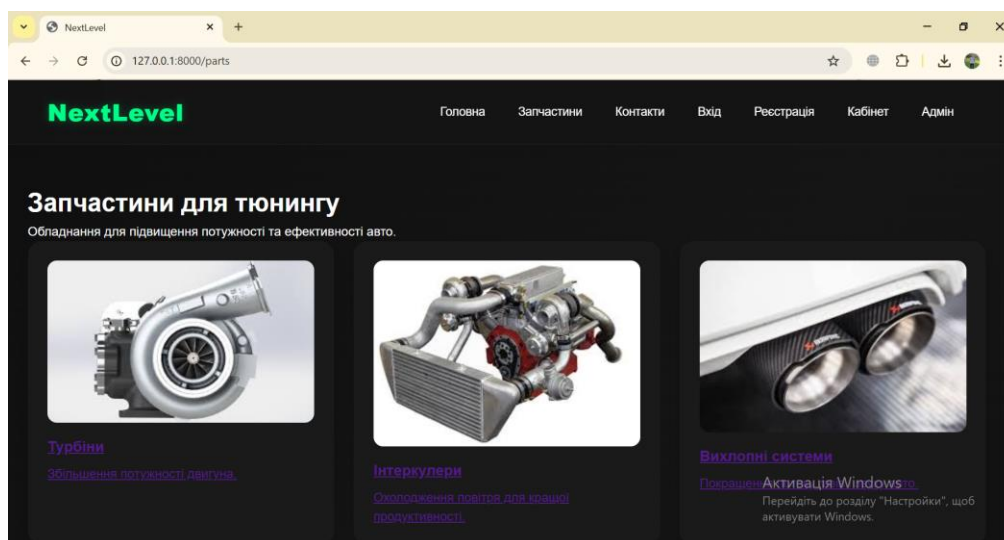


Рисунок 3.8 Сторінка запчастин для тюнингу

Джерело: розроблено автором

На рисунку 3.9 показано сторінку покупки запчастин. Вона забезпечує перехід від перегляду товарів до безпосереднього оформлення замовлення. Користувач може ознайомитися з доступними позиціями, обрати потрібний товар і перейти до подальших дій щодо його придбання. Така сторінка є важливою складовою веб-застосунку, оскільки реалізує комерційну функцію системи та дозволяє користувачеві завершити процес взаємодії з обраним товаром.

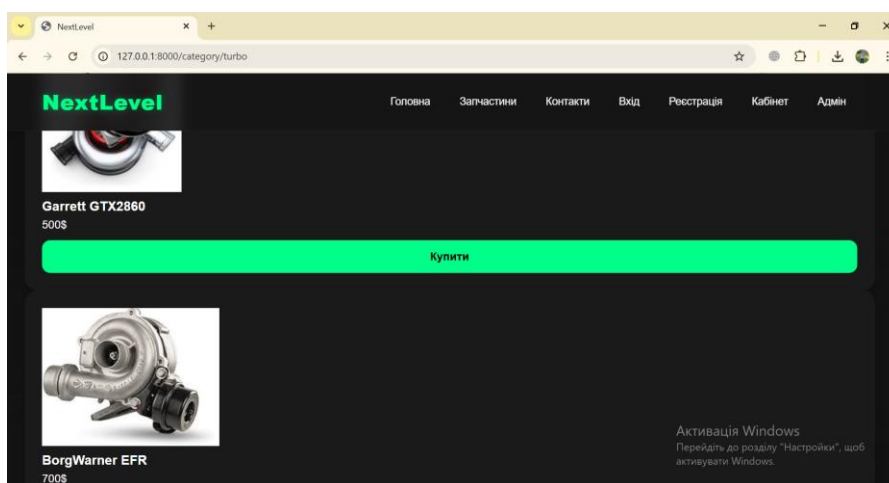


Рисунок 3.9 Сторінка покупки запчастин

Джерело: розроблено автором.

На рисунку 3.10 представлено кабінет користувача. Цей розділ призначений для перегляду персональної інформації та історії замовлень. Наявність кабінету підвищує зручність користування системою, оскільки користувач може повернутися до попередніх замовлень, переглянути обрані конфігурації або перевірити інформацію про виконані дії. Таким чином, кабінет користувача забезпечує персоналізацію роботи із сервісом і робить взаємодію з веб-застосунком більш завершеною.

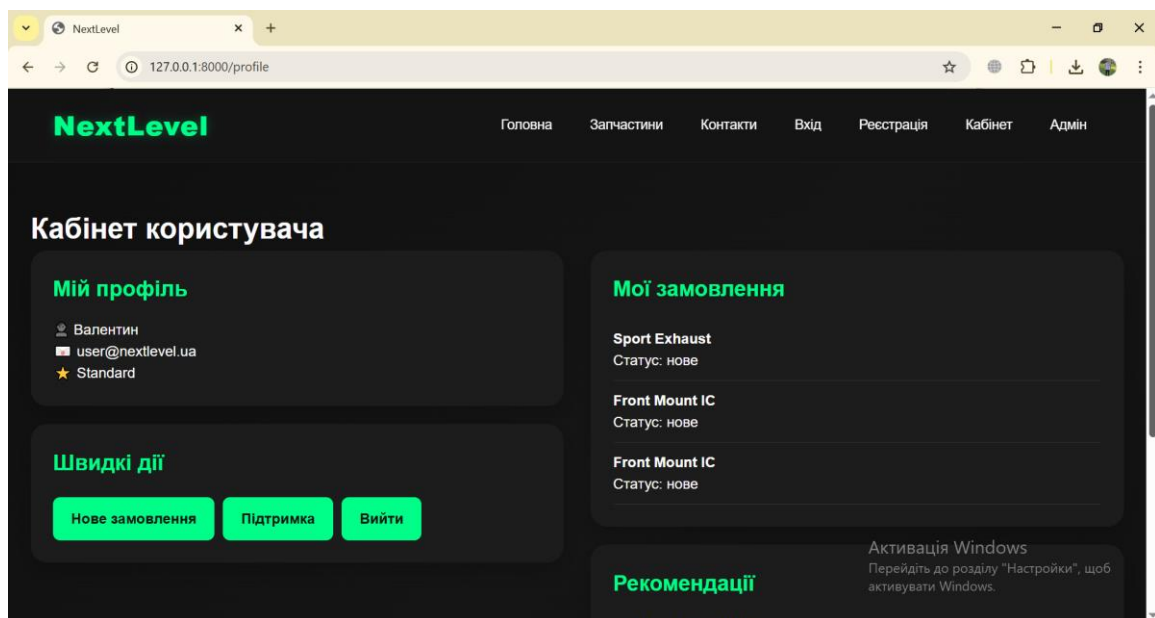


Рисунок 3.10 Кабінет користувача

Джерело: розроблено автором

Інтерфейс користувача виконаний у сучасному стилі з використанням адаптивного дизайну та візуальних елементів (зображення, картки товарів), що покращує зручність використання та загальне враження від роботи із системою.

Програмний продукт протестовано в локальному середовищі. Сервер успішно запускається за допомогою Uvicorn, а доступ до веб-застосунку здійснюється через браузер. Також реалізовано можливість демонстрації роботи

застосунку через зовнішній доступ (наприклад, із використанням тунелювання), що дозволяє переглядати сайт іншим користувачам.

Отриманий результат відповідає поставленим цілям і завданням даного проєкту. Розроблений веб-застосунок є працездатним, логічно структурованим та може бути використаний як основа для подальшого розвитку і масштабування.

Висновки до розділу 3

У третьому розділі було розглянуто процес реалізації веб-застосунку «NextLevel», який створено на основі сучасної клієнт-серверної архітектури.

Було реалізовано основні функціональні модулі системи: пошук прошивок, багатокритеріальну фільтрацію, рекомендаційний механізм, особистий кабінет користувача та систему замовлень.

Проведене тестування підтвердило коректність роботи програмного продукту, його стабільність та готовність до практичного використання.

ВИСНОВКИ

У кваліфікаційній роботі було вирішено актуальне завдання розробки веб-застосунку «NextLevel» для чіп-тюнінгу автомобілів із використанням багатокритеріальних фільтрів та рекомендаційної системи.

У результаті виконання проєкту було розроблено сучасний вебресурс для підбору та замовлення автомобільних прошивок і тюнінгових компонентів. Під час роботи було досліджено принципи створення вебдодатків на мові програмування Python із використанням фреймворку FastAPI, системи шаблонізації Jinja2 та бази даних SQLite.

У процесі виконання роботи було реалізовано серверну частину сайту, яка забезпечує взаємодію користувача з базою даних, обробку HTTP-запитів, авторизацію та систему керування замовленнями. Також було створено клієнтську частину вебресурсу з адаптивним дизайном, сучасним оформленням та зручною навігацією.

У межах практичної реалізації проєкту було створено такі функціональні можливості:

- система реєстрації та авторизації користувачів;
- підтримка сесій користувачів;
- адміністрування вебресурсу через окрему адміністративну панель;
- пошук автомобільних прошивок за маркою, моделлю, роком випуску та типом двигуна;
- система замовлення та оплати прошивок;
- відображення історії замовлень у кабінеті користувача;
- каталог тюнінгових запчастин із окремими категоріями;
- можливість переходу до окремих сторінок товарів;
- система завантаження готових файлів прошивок;
- динамічне оновлення статусів замовлень.

Особливу увагу було приділено структурі програмного коду, організації маршрутів FastAPI та роботі з базою даних SQLite. Для збереження інформації були створені таблиці користувачів, прошивок та замовлень. Реалізовано механізм взаємодії між клієнтською та серверною частинами через HTML-форми та HTTP-запити.

У ході розробки також було вдосконалено інтерфейс вебресурсу. На сторінках було використано тематичні зображення автомобілів, турбін, інтеркулерів та інших елементів тюнінгу, що значно покращило зовнішній вигляд сайту та зробило його більш привабливим для користувачів. Було реалізовано єдиний стиль оформлення всіх сторінок, адаптивні блоки та навігаційне меню.

Під час тестування вебресурсу було перевірено працездатність основних функцій системи: реєстрації, входу в акаунт, пошуку прошивок, створення замовлень, зміни статусів та завантаження файлів. Усі реалізовані модулі працюють коректно та забезпечують стабільну роботу системи.

Практична цінність роботи полягає у можливості використання створеного вебресурсу як основи для реального сервісу автомобільного чип-тюнінгу. Розроблений проєкт може бути розширений шляхом підключення платіжних систем, онлайн-бази файлів прошивок, системи відгуків користувачів, інтеграції з хмарними сервісами та повноцінної CRM-системи.

Отже, поставлена мета кваліфікаційної роботи була повністю досягнута, а всі визначені завдання успішно виконані. Отримані результати підтверджують ефективність використання сучасних вебтехнологій для створення функціональних та зручних вебресурсів автомобільної тематики.

ПЕРЕЛІК ДЖЕРЕЛ ТА ПОСИЛАННЯ

1. ChipBox Україна: вебсайт. URL: <https://chipbox.com.ua> (дата звернення: 04.02.2026).
2. RaceChip: вебсайт. URL: <https://racechip.com> (дата звернення: 04.02.2026).
3. TuningFiles: вебсайт. URL: <https://tuningfiles.com> (дата звернення: 04.02.2026).
4. Sommerville I. Software Engineering. 10th ed. Boston: Pearson, 2016. 816 p.
5. Pressman R. S., Maxim B. R. Software Engineering: A Practitioner's Approach. 8th ed. New York: McGraw-Hill Education, 2014. 976 p.
6. Fowler M. UML Distilled: A Brief Guide to the Standard Object Modeling Language. 3rd ed. Boston: Addison-Wesley, 2004. 208 p.
7. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. Boston: Addison-Wesley, 1995. 395 p.
8. Date C. J. An Introduction to Database Systems. 8th ed. Boston: Pearson, 2003. 1024 p.
9. PostgreSQL Global Development Group. PostgreSQL Documentation: вебсайт. URL: <https://www.postgresql.org/docs/> (дата звернення: 27.04.2026).
10. Mozilla Developer Network. HTML, CSS, JavaScript Documentation: вебсайт. URL: <https://developer.mozilla.org/> (дата звернення: 27.04.2026).
11. Fielding R. T. Architectural Styles and the Design of Network-based Software Architectures: dissertation. Irvine: University of California, 2000. URL: <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>
12. Welling L., Thomson L. PHP and MySQL Web Development. 5th ed. Boston : Addison-Wesley, 2016. 688 p.
13. React Documentation: вебсайт. URL: <https://react.dev/learn> (дата звернення: 21.03.2026).

14. Node.js Documentation: вебсайт. URL: <https://nodejs.org/docs/latest/api/> (дата звернення: 14.04.2026).
15. Express Documentation: вебсайт. URL: <https://expressjs.com/> (дата звернення: 17.04.2026).
16. SQLAlchemy Documentation: вебсайт. URL: <https://docs.sqlalchemy.org/> (дата звернення: 17.04.2026).