

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.
_____ Ю. П. Кондратенко
«_____» _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА
ВЕБЗАСТОСУНОК ДЛЯ ВИБОРУ АВТО ЗА
КРИТЕРІЯМИ

Спеціальність 122 «Комп'ютерні науки»

122 – КРБ – 402.22010230

Виконав студент 4-го курсу, групи 402
_____ *Р. Ю. Шапар*
«20» червня 2024 р.

Керівник: д-р техн. наук, доцент
_____ *О. В. Козлов*
«20» червня 2024 р.

Миколаїв – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет ім. Петра Могили

Факультет комп'ютерних наук

Кафедра інтелектуальних інформаційних систем

Рівень вищої освіти **бакалавр**
Спеціальність **122 «Комп'ютерні науки»**
(шифр і назва)
Галузь знань **12 «Інформаційні технології»**
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.

_____ Ю. П. Кондратенко

« ____ » _____ 2024 р.

З А В Д А Н Н Я

на виконання кваліфікаційної роботи

Видано студенту групи 402 факультету комп'ютерних наук Шапар Руслану Юрійовичу.

1. Тема кваліфікаційної роботи «Вебзастосунок для вибору авто за критеріями».

Керівник роботи Козлов Олексій Валерійович, д-р техн. наук, доцент

Затв. наказом Ректора ЧНУ ім. Петра Могили від «28» грудня 2024 р. № 271

2. Строк представлення кваліфікаційної роботи студентом «20» червня 2024 р.

3. Вхідні (початкові) дані до роботи: розробка вебзастосунку з фронтенд і бекенд частинами для підбору автомобілів за визначеними критеріями; інтеграція ШІ бота для допомоги з вибором авто в залежності від побажань клієнта.

Очікуваний результат: Повноцінний вебзастосунок для підбору авто за критеріями, та ШІ ботом для допомоги з вибором.

4. Перелік питань, що підлягають розробці (зміст пояснювальної записки):

- аналіз сучасних вебзастосунків що пропонують послуги у виборі авто;
- розробка фронтенд і бекенд частин застосунку;
- підключення ШІ для допомоги в виборі авто.

5. Перелік графічного матеріалу: презентація.

6. Завдання до спеціальної частини: «Автоматизація систем керування транспортом»

7. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис
Спеціальна частина з охорони праці	Алексеева А. О., доцент кафедри екології	

Керівник роботи д-р техн. наук, доцент Козлов О. В.

(наук. ступінь, вчене звання, прізвище та ініціали)

(підпис)

Завдання прийнято до виконання Шапар Р. Ю.

(прізвище та ініціали)

(підпис)

Дата видачі завдання « 14 » _____ січня _____ 2024 р.

КАЛЕНДАРНИЙ ПЛАН виконання кваліфікаційної роботи

Тема: Вебзастосунок для вибору авто за критеріями

№	Найменування роботи	Початок	Закінчення	Примітки
1	Подання заяви на затвердження теми та керівників КРБ	10.11.2023	15.11.2023	Виконано
2	Отримання завдання на виконання КРБ	10.01.2024	15.01.2024	Виконано
3	Складання календарного плану роботи на весь період виконання КРБ	16.01.2024	30.01.2024	Виконано
4	Отримання завдання на переддипломну практику	15.04.2024	29.04.2024	Виконано
5	Проходження переддипломної практики, збір та аналіз матеріалів до КРБ	29.04.2024	11.05.2024	Виконано
6	Розробка звіту з переддипломної практики	12.05.2024	15.05.2024	Виконано
7	Виконання КРБ: проектування вебзастосунку для вибору авто за критеріями, огляд існуючих технологій, розробка ПЗ	13.05.2024	22.06.2024	Виконано
8	Перший попередній захист КРБ на засіданні комісії кафедри	27.05.2024	27.05.2024	Виконано
9	Доробка та остаточне оформлення КРБ	28.05.2024	09.06.2024	Виконано
10	Другий попередній захист КРБ на засіданні комісії кафедри	10.06.2024	10.06.2024	Виконано
11	Подання КРБ рецензенту	13.06.2024	13.06.2024	
11	Подання КРБ, її електронної копії та інших документів (відгуку, рецензії) до захисту	17.06.2024	21.06.2024	
12	Захист КРБ перед екзаменаційною комісією (ЕК)	27.06.2024	28.06.2024	

Розробив студент Шапар Р. Ю.

(прізвище, ім'я, по батькові студента)

(підпис)

Керівник роботи д-р техн. наук, доцент Козлов О. В.

(посада, прізвище, ім'я, по батькові)

(підпис)

« 29 » 01 2024 р.

АНОТАЦІЯ

кваліфікаційної роботи студента групи 402 ЧНУ ім. Петра Могили

Шапара Руслана Юрійовича

Тема: «Вебзастосунок для вибору авто за критеріями»

Актуальність. У сучасному світі електронна комерція, зокрема продаж автомобілів через інтернет, стає все більш популярною та важливою сферою бізнесу. Вебсайти, що пропонують продукцію автомобілів, стають важливими інструментами як для покупців, так і для продавців.

Об'єктом дослідження є процеси розробки вебзастосунків для продажу автомобілів в інтернеті.

Предметом дослідження є технології та методи, які сприяють впровадженню та функціонуванню вебзастосунку з продажу автомобілів.

Мета полягає у створенні вебзастосунку з продажу автомобілів, що відповідає потребам споживачів та сприяє розвитку електронної комерції в цій сфері.

Робота складається з фахового розділу і спеціальної частини з охорони праці. Пояснювальна записка складається зі вступу, трьох розділів та висновків.

У першому розділі було проведено аналіз ринку електронної комерції в галузі продажу автомобілів, визначення актуальності теми, перегляд застосунків-аналогів та технологій для досягнення поставленої мети.

У другому розділі виконано моделювання та реалізацію фронтенд частини вебзастосунку.

У третьому розділі продемонстровано моделювання та реалізацію бекенд частини та виправлення зустрічних помилок у ході розробки.

В результаті розроблено вебзастосунок мовами програмування Html, CSS, JavaScript, PHP з використанням gpt API.

Кваліфікаційна робота містить 53 сторінки, 18 рисунків, 0 таблиць, 12 використаних джерел та 0 додатків.

Ключові слова: Html, CSS, JavaScript, PHP, GPT API, Visual Studio Code.

ABSTRACT

**To the qualification work by student of group 402 of
Petro Mohyla Black Sea National University
Shapar Ruslan**

Theme: «Web site for selecting a car based on criteria»

Relevance. In the regular e-commerce sector, large car sales via the Internet remain increasingly popular and important for spherical businesses. Websites that offer automotive products contain important tools for both buyers and sellers. The object of the study is the process of developing a web application for online car sales.

The subject of the study is the technologies and methods that facilitate the implementation and operation of the web application for car sales.

The aim is to create a web application for car sales that meets consumer needs and promotes the development of e-commerce in this field.

The work consists of a professional section and a special part on occupational safety. The explanatory note consists of an introduction, three chapters, and conclusions.

In the first chapter, an analysis of the ecommerce car sales market was conducted, determining the topic's relevance, reviewing similar applications, and the technologies for achieving the goal.

In the second chapter, the modeling and implementation of the frontend part of the web application were carried out.

In the third chapter, the modeling and implementation of the backend were demonstrated, necessary functions were implemented, and development errors were corrected.

As a result, a web application was developed using HTML, CSS JavaScript, and PHP with the use of the GPT API.

The qualification work contains 53 pages, 18 figures, 0 tables, 12 references, and 0 appendices.

Keywords: HTML, CSS, JavaScript, PHP, GPT API, Visual Studio Code.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	3
ВСТУП.....	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Постановка задачі.....	7
1.2 Дослідження і аналіз об'єкту програмування	7
1.3 Огляд існуючих аналогів	8
Висновки до розділу 1	10
2 ПРОЄКТУВАННЯ FRONTEND ЧАСТИНИ ТА НАПИСАННЯ ОСНОВНОГО ІНТЕРФЕЙСУ ВЕБЗАСТОСУНКУ	12
2.1 Особливості застосунку	12
2.2 Опис вимог до застосунку	13
2.3 HTML/CSS верстка.....	15
2.4 Адаптивна верстка.....	19
2.5 JavaScript.....	20
2.6 Огляд реалізації та коду Frontend частини проєкту:	22
Висновки до розділу 2	41
3 ПРОЄКТУВАННЯ ТА НАПИСАННЯ BACKEND ЧАСТИНИ ВЕБЗАСТОСУНКУ ТА ВИКОРИСТАННЯ ШІ.....	42
3.1 Огляд використаних backend технологій	42
3.2 Розбір коду на PHP	45
Висновки до розділу 3	51
ВИСНОВКИ	52
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	53

ПЕРЕЛІК СКОРОЧЕНЬ

ПЗ	–	Програмне Забезпечення
HTML	–	Hypertext Markup Language
JS	–	JavaScript
NPM	–	Node Package Manager
PHP	–	Hypertext Preprocessor
SCSS	–	Syntactically Cascading Style Sheet

ВСТУП

Сьогодні сфера електронної комерції стрімко розвивається, охоплюючи нові ринки та залучаючи все більше користувачів. Щоб закріпитися на ринку та втримати позиції, компанії повинні знаходити нові шляхи розвитку та вдосконалення процесу обслуговування клієнтів. Особливо це стосується автомобільної галузі, де споживачі вимагають не тільки високої якості товарів, але й відмінного обслуговування та зручності у виборі автомобіля.

Автоматизація процесу підбору авто за критеріями значно підвищує ефективність роботи магазинів та задовольняє потреби покупців. Сучасні інформаційні технології дозволяють організувати роботу підприємства з максимальною ефективністю, знижуючи вплив людського фактору та мінімізуючи ризики. З розвитком інтернет-технологій, розробка веб-застосунків стає однією з найперспективніших галузей для впровадження нових рішень в торгівлю.

Високий рівень конкуренції вимагає від лідерів галузі впровадження найсучасніших та найкомфортніших технологій, зокрема системи автоматизованого підбору авто за критеріями. Запровадження такого веб-застосунку дозволить підвищити якість надання послуг споживачам, що забезпечить збільшення прибутку. Саме тому тема дипломної роботи спрямована на вирішення проблеми вдосконалення процесу підбору авто в інтернет-магазинах.

Задачі роботи

- 1) провести аналіз існуючих аналогів та визначити їхні переваги та недоліки;
- 2) визначити функціональні вимоги до веб-застосунку;
- 3) розробити архітектуру веб-застосунку та вибрати технології для його реалізації;
- 4) реалізувати фільтрацію авто за заданими критеріями;
- 5) розробити та інтегрувати ШІ бота для підбору автомобілів.

Об'єктом дослідження є процеси розробки вебзастосунків для продажу автомобілів в інтернеті

Предметом дослідження є технології та методи, які сприяють впровадженню та функціонуванню вебзастосунку з продажу автомобілів.

Для вирішення практичного завдання з веб-розробки було використано декілька мов програмування. Для створення структури і вигляду веб-сторінок використовувалися мови HTML і CSS, що дозволило створити коректне та естетичне відображення інформації на сайті. Для реалізації інтерактивності та динаміки на сторінках веб-додатка була використана мова JavaScript, яка дозволила створити різноманітні функціональні можливості для користувачів.

Для забезпечення роботи бекенду веб-додатка використовувався PHP. Ця мова програмування дозволила обробляти запити користувачів і взаємодіяти з базою даних для отримання та збереження інформації. PHP забезпечив стабільну та ефективну роботу всієї системи, забезпечуючи швидку обробку запитів і відповідей на них.

Для реалізації штучного інтелекту (ШІ) на веб-сайті було використано gpt API та PHP. Це дозволило створити бота, який може взаємодіяти з користувачем, розуміти його запити та надавати зрозумілі відповіді. Використання PHP для взаємодії з gpt API забезпечило швидку та ефективну роботу бота на веб-сайті.

1.1 Постановка задачі

Метою дипломної роботи є аналіз проблем автоматизації процесу обробки інформації, пов'язаної з обліком товарів, а також розробка пропозицій по вдосконаленню цієї діяльності. Для досягнення цієї мети були вирішені наступні завдання:

- вивчення основних напрямів впровадження нових технологій в торговельному бізнесі;
- дослідження основ організації ведення обліку товарів та надання послуг споживачам в магазинах;
- визначення сутності автоматизації та її ролі у підприємстві електронної комерції;
- розгляд видів програмного забезпечення на ринку торгівлі;
- розгляд процесу автоматизації введення та обробки інформації про товар;
- проведення дослідження існуючих проблем в галузі комерції;
- проведення критичного аналізу програмних продуктів магазинів;
- розгляд напрямків вирішення питань щодо вдосконалення процесу автоматизації роботи користувачів.

1.2 Дослідження і аналіз об'єкту програмування

Основною діяльністю магазинів є продаж товарів. Для якісного виконання цієї задачі, необхідно розуміти, скільки товару і за якою ціною знаходиться на вашому складі. Товарний облік існує для управління переміщеннями товарів у реалізації, тобто допомагає розуміти, що відбувається з товаром у магазині, оптимізувати процеси, сформувати необхідний асортимент, який буде привабливим для клієнтів та уникнути крадіжок.

Невеликі магазини ведуть облік "вручну", тобто без допомоги додаткових пристроїв. "Ручний режим" в бухгалтерії допускається чинним законодавством і не тягне за собою санкцій з боку податкової служби або правоохоронних органів. Однак на практиці цей спосіб не просто незручний і дорогий, але і тягне за собою не мало негативних наслідків, таких як погіршення якості розрахунків, низький товарообіг, складність ведення документації, відсутність систематизації, неможливість сортування або відбору за заданими критеріями, неефективна робота постачальників, помилки, викликані людським фактором, нестачі. Автоматизована система ведення товарообігу є зручним та ефективним способом для ведення бізнесу та дозволяє зосередитись на більш важливих процесах.

1.3 Огляд існуючих аналогів

Програми товарообліку спеціалізуються на різних напрямках виробництва. Кожна компанія обирає таку систему обліку товару, яка підходить їм та виконує всі необхідні функції ведення обліку для підприємства. Перед розробкою програмної реалізації системи, необхідно ознайомитись із існуючими аналогами та проаналізувати їх сильні та слабкі сторони, щоб уникнути повторення слабкостей та створити унікальний функціонал системи.

AutoTrader:

AutoTrader є одним з найпопулярніших веб-сайтів для продажу автомобілів. Цей сервіс дозволяє користувачам шукати та порівнювати автомобілі з різних дилерських центрів та приватних продавців. AutoTrader (рис1.1) надає детальну інформацію про кожен автомобіль, включаючи фотографії, технічні характеристики та ціни (рис 1.1).

Кафедра інтелектуальних інформаційних систем
Вебзастосунок для вибору авто за критеріями

Lease a brand new car





 <p>From £198 Per month</p> <p>£2,376 initial payment 48 month contract 6,000 miles p/a</p>	 <p>From £191 Per month</p> <p>£2,292 initial payment 48 month contract 6,000 miles p/a</p>	 <p>From £233 Per month</p> <p>£2,092 initial payment 24 month contract 6,000 miles p/a</p>	 <p>From £270 Per month</p> <p>£3,239 initial payment 36 month contract 6,000 miles p/a</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Рисунок 1.1 – Головна сторінка сайту AutoTrader

Cars.com:

Cars.com - це інший популярний веб-сайт який спеціалізується на продажу автомобілів. Сайт пропонує широкий вибір нових та вживаних автомобілів від різних брендів. Крім того, Cars.com надає користувачам можливість знаходити дилерські центри, читати відгуки про автомобілі та отримувати поради щодо покупки авто (рис 1.2).

Кафедра інтелектуальних інформаційних систем Вебзастосунок для вибору авто за критеріями

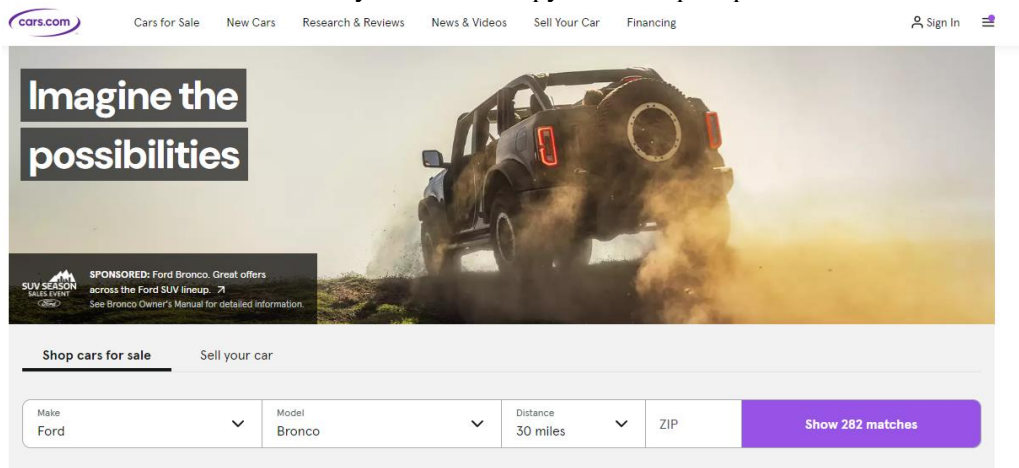


Рисунок 1.2 – Головна сторінка сайту Cars.com

TrueCar:

TrueCar - це онлайн-платформа, яка допомагає покупцям отримати найкращу ціну на новий автомобіль. Сервіс дозволяє користувачам порівнювати ціни від різних дилерських центрів та отримувати рекомендації щодо того, яка ціна є справедливою для конкретної марки та моделі автомобіля (рис 1.3).

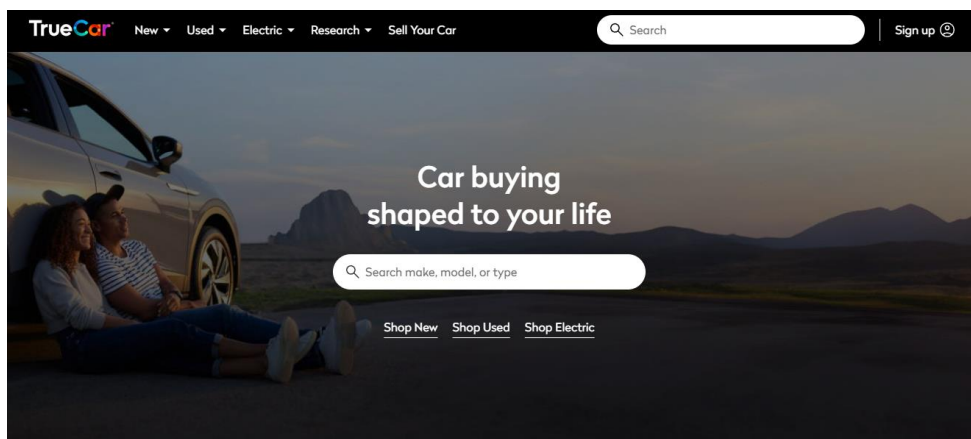


Рисунок 1.3 – Головна сторінка сайту TrueCar

Висновки до розділу 1

У першому розділі було детально розглянуто проблеми автоматизації процесу обробки інформації, пов'язаної з обліком товарів, а також запропоновано шляхи їх вирішення для підвищення ефективності діяльності торговельних підприємств.

Застосування таких систем не тільки знижує вплив людського фактора, а й дозволяє отримати доступ до більш детальної та своєчасної інформації про стан товарних запасів, що в свою чергу покращує прийняття управлінських рішень. Автоматизація процесів обліку також дозволяє оптимізувати логістику та зменшити витрати на зберігання і транспортування товарів.

На основі проведених досліджень та аналізу було виявлено, що ручний облік товарів є неефективним, що може призводити до низки проблем, таких як помилки в розрахунках, низький товарообіг, складність ведення документації та інші. Автоматизовані системи обліку товарообігу, з іншого боку, забезпечують зручність, ефективність та точність у веденні бізнесу, дозволяючи зосередитися на важливіших процесах.

Огляд існуючих аналогів, таких як AutoTrader, Cars.com і TrueCar, показав, що успішні платформи для продажу автомобілів використовують сучасні технології для надання детальної інформації про товари, що сприяє зручності та ефективності процесу покупки для споживачів.

Таким чином, результати даного розділу підтверджують важливість автоматизації процесів обліку товарів для підвищення ефективності роботи торговельних підприємств та задоволення потреб споживачів.

2 ПРОЄКТУВАННЯ FRONTEND ЧАСТИНИ ТА НАПИСАННЯ ОСНОВНОГО ІНТЕРФЕЙСУ ВЕБЗАСТОСУНКУ

2.1 Особливості застосунку

Інтуїтивний інтерфейс користувача: вебзастосунок для підбору авто за критеріями розроблено з акцентом на зручність та інтуїтивність користувацького інтерфейсу.

Форма підбору автомобілів: форма підбору автомобілів є ключовою функцією застосунку. Вона дозволяє користувачам задавати критерії пошуку, такі як: діапазон цін, тип кузова, колір.

Марка та модель автомобіля: після заповнення форми користувач може натиснути кнопку пошуку, і застосунок відобразить список автомобілів, які відповідають заданим критеріям.

Відображення результатів пошуку: результати пошуку відображаються у вигляді списку автомобілів з основною інформацією про кожне авто, включаючи модель, рік випуску та фото. Користувачі можуть переглянути деталі кожного автомобіля, натиснувши на відповідний елемент списку.

Форма зворотного зв'язку: форма зворотного зв'язку дозволяє користувачам відправляти свої відгуки та запити. Вона містить три поля: ім'я (Name), електронна пошта (E-mail), текст повідомлення (Your text).

Інтеграція ШІ бота: ШІ бот допомагає користувачам у виборі автомобіля, відповідаючи на їхні запитання та надаючи рекомендації. Бот може:

- відповідати на загальні питання про автомобілі;
- допомагати у виборі автомобіля за критеріями;
- надати інформацію про конкретні моделі.

Продуктивність та надійність: застосунок оптимізовано для швидкої роботи, забезпечуючи миттєву обробку запитів та відображення результатів пошуку. Також забезпечується висока надійність роботи застосунку, з мінімальними можливими простоями та безперебійною роботою.

Ці особливості забезпечують високий рівень зручності, функціональності та безпеки вебзастосунку для підбору автомобілів за критеріями, роблячи його ефективним інструментом для користувачів, які шукають автомобіль своєї мрії.

2.2 Опис вимог до застосунку

Функціональні вимоги

Користувацький інтерфейс:

- форма підбору авто: містить поля для введення критеріїв підбору (ціна, колір, тип кузова тощо);
- результати пошуку: відображення списку автомобілів, що відповідають заданим критеріям. Кожний результат повинен містити інформацію про авто (модель, рік, пробіг, ціна, фото тощо);
- деталі автомобіля: сторінка з детальною інформацією про вибране авто.
- форма зворотного зв'язку: містить поля "Name", "E-mail", "Your text" для відправлення відгуків та запитів;
- чат з ШІ ботом: інтерфейс для взаємодії з ботом, який допомагає користувачам у виборі автомобіля, відповідаючи на питання та надаючи рекомендації.

Форма підбору авто:

- поле ціна: користувач може вказати бажаний діапазон цін;
- поле колір: випадаючий список з доступними кольорами;
- поле тип кузова: випадаючий список з різними типами кузова (седан, позашляховик, хетчбек тощо);
- кнопка пошуку: запуск процесу пошуку автомобілів за заданими критеріями.

Обробка даних:

- пошук автомобілів: застосунок повинен швидко обробляти запити та надавати результати пошуку відповідно до заданих критеріїв;

Кафедра інтелектуальних інформаційних систем
Вебзастосунок для вибору авто за критеріями

– фільтрування результатів: можливість фільтрувати результати пошуку за додатковими параметрами (наприклад, рік випуску, пробіг).

III бот:

– консультації користувачів: бот повинен відповідати на питання користувачів та надавати рекомендації щодо вибору автомобіля.

– обробка запитів: бот повинен розуміти та обробляти текстові запити користувачів, надаючи релевантні відповіді.

Нефункціональні вимоги

Продуктивність:

– швидкодія: застосунок повинен забезпечувати швидкий відгук на дії користувача, особливо при пошуку та фільтруванні автомобілів;

– масштабованість: застосунок повинен бути здатним обробляти великий обсяг запитів без втрати продуктивності.

Надійність:

– безперебійність роботи: застосунок повинен працювати стабільно без частих збоїв;

– відновлення даних: забезпечення збереження даних та їх відновлення у разі виникнення неполадок.

Зручність використання:

– інтуїтивний інтерфейс: користувачі повинні легко розуміти, як використовувати функції застосунку;

– адаптивний дизайн: застосунок повинен коректно відображатися на різних пристроях, включаючи смартфони, планшети та настільні комп'ютери.

Технічні вимоги

Фронтенд:

– HTML/CSS: використання сучасних технологій HTML5 і CSS3 для розмітки та стилізації інтерфейсу;

– SASS: використання препроцесора SASS для ефективного управління стилями;

– адаптивна верстка: забезпечення адаптивності інтерфейсу за допомогою медіа-запитів та гнучких макетів.

JavaScript:

– функціональність: використання JavaScript для реалізації динамічної поведінки на стороні клієнта, включаючи валідацію форм та взаємодію з API.

2.3 HTML/CSS верстка

HTML (Hyper Text Markup Language) була розроблена Тімоті Бернерс-Лі у 1991 році в ЦЕРН (Європейський центр ядерних досліджень) в якості спрощеної версії SGML (Standart Generalized Markup Language). Спочатку вона використовувалася для обміну документами між вченими, але потім стала основою для створення веб-сторінок. HTML використовується для створення структури сторінок, визначення їх змісту і розміщення різних елементів, таких як текст, зображення, відео і посилання

Основними принципами HTML є:

– розмітка – HTML використовує теги для визначення різних типів контенту і його структури. Теги зазвичай мають відкриваючий ‘<tag>’ і закриваючий ‘</tag>’ теги, наприклад ‘<p>’ для параграфа;

– структура – HTML документ складається з декількох основних частин, таких як ‘<html>’, ‘<head>’, ‘<title>’, ‘<body>’, які визначають структуру і зміст сторінки;

– елементи та атрибути – елементи визначають різні типи контенту (текст, зображення, посилання, тощо), а атрибути надають додаткову інформацію про елемент, наприклад ‘href’, для посилань;

– семантика – HTML постійно розвивалася і вдосконалювалася. Починаючи з оригінальної версії HTML, вийшло кілька версій: HTML 2.0, HTML 3.2 HTML 4.01, XHTML (HTML який відповідає стандартам XML), і остання на даний час версія – HTML5. HTML5 вводить нові елементи, атрибути та можливості, спрощуючи створення веб-сторінок і покращуючи їх функціональність.

CSS (Cascading Style Sheets) (рис 2.2) є мовою стилів, яка використовується для визначення вигляду і форматування веб-сторінок, написаних мовою HTML. CSS дозволяє розділити зміст сторінки від її представлення, що дозволяє змінювати вигляд сторінки без зміни її змісту. Розглянемо деталі CSS та історію її створення:

– історія створення CSS: CSS була запропонована Хаконом Віума Лібеком та Бертом Босом у 1994 році. Основною метою CSS було розділення структури сторінки (HTML) від її представлення (стили CSS), щоб полегшити додавання стилів до веб-сторінок і покращити їх зовнішній вигляд.

Основні концепції CSS:

- селектори: CSS використовує селектори для вибору елементів, до яких застосовуються стилі. Наприклад 'p' вибирає всі параграфи на сторінці;
- властивості: властивості CSS визначають вигляд елементів, такі як колір тексту, розмір шрифту, відступи тощо;
- значення: значення властивостей вказуються, як саме має відобразитися елемент. Наприклад, 'color: blue;' встановлює колір тексту на синій;
- каскадність і спадковість: CSS використовує принцип каскадності, коли стилі можуть бути успадковані від батьківських елементів. Це дозволяє застосувати стилі до груп елементів одночасно;
- боксова модель: елементи в CSS моделізуються як прямокутні «бокси», які мають внутрішній вміст, відступи, рамки і поля.

CSS постійно розвивається, і вийшло кілька версій, починаючи з CSS1 в 1996 році і закінчуючи CSS3, який є поточною версією. CSS3 вводить багато нових можливостей, таких як анімація, тіні, градієнти та інші ефекти, що полегшують створення стильних і сучасних веб-сторінок.

HTML і CSS відіграють ключову роль у веб-розробці, визначаючи структуру, вигляд і поведінку веб-сторінок. Давайте розглянемо їхню роль детальніше.

Роль HTML у веб-розробці:

- структура сторінки: HTML визначає структуру сторінки за допомогою різних елементів, таких як заголовки ('<h1>' '<h1>', абзаци ('<p>'), списки ('', ''), таблиці ('<table>'), форми ('<form>') та інші;
- зміст сторінки: HTML визначає зміст сторінки, включаючи текст, зображення, відео, посилання та інші медіа-елементи;
- семантика: HTML надає семантичні теги, які допомагають пошуковим системам краще розуміти зміст сторінки. Наприклад тег '<article>' вказує на незалежну статтю, а тег '<nav>' вказує на навігаційний блок;
- базовий шаблон: HTML використовується для створення базового шаблону сторінки, який включає в себе заголовок, навігацію, вміст і підвал.

Роль CSS у веб-розробці:

- вигляд сторінки: CSS відповідає за вигляд сторінки, включаючи колір тексту, розмір шрифту, відступи, рамки, фонові кольори та інші візуальні аспекти;
- модульність і каскадність: CSS дозволяє створювати стилі, які можна застосувати до різних елементів або груп елементів одночасно. Крім того, він має каскадність, що дозволяє визначити пріоритетність стилів у випадку конфлікту;
- адаптивний дизайн: CSS дозволяє створювати адаптивні стилі, які змінюються в залежності від розміру екрану пристрою, що дозволяє сторінці виглядати добре на різних пристроях;
- анімація і переходи: CSS дозволяє створювати анімацію та переходи для покращення візуального ефекту сторінки без використання JavaScript;
- друкування сторінки: CSS дозволяє визначити вигляд сторінки при друку, що дозволяє створювати дружні до друку сторінки.

Усі ці аспекти HTML і CSS роблять їх ключовими технологіями для створення веб-сторінок, які є платформонезалежними, доступними і добре структурованими. (рис 2.1, 2.2)



Рисунок 2.1 - Логотип HTML



Рисунок 2.2 - Логотип CSS

2.4 Адаптивна верстка

Адаптивна верстка - це методологія розробки веб-сайтів, яка дозволяє сторінкам адекватно змінювати свій вигляд та розміщення елементів в залежності від розміру екрану, на якому вони відображаються. Головна мета адаптивної верстки - забезпечити користувачам зручне і ефективне відображення контенту незалежно від пристрою, на якому вони переглядають сайт (наприклад, комп'ютер, планшет або смартфон).

Основні принципи адаптивної верстки включають:

- гнучкі сітки (Flexbox та Grid): використання гнучких сіток дозволяє створювати динамічні макети, які адаптуються до різних розмірів екрану;
- медіа-запити (Media Queries): це CSS-правила, які дозволяють застосовувати стилі в залежності від характеристик пристрою, таких як ширина екрану, висота екрану, роздільна здатність тощо.;
- резинові картинки та медіа-вміст: використання відносних одиниць для визначення розмірів зображень та відео дозволяє їм пристосовуватися до розміру екрану;
- прихований вміст: використання CSS для приховування або показу певних елементів в залежності від розміру екрану;
- адаптивні типографія та шрифти: використання відносних одиниць для визначення розмірів шрифтів, що дозволяє їм адаптуватися до розміру екрану;
- мобільне перше (Mobile-first): підхід, при якому стилі для мобільних пристроїв задаються першими, а потім, за допомогою медіа-запитів, додаються стилі для більших екранів;
- тестування на різних пристроях: перевірка вигляду та функціональності сайту на різних пристроях, щоб впевнитися, що він коректно відображається на всіх пристроях.

Адаптивна верстка дозволяє забезпечити зручний та ефективний досвід користувачам незалежно від пристрою, що вони використовують, і є важливою частиною розробки сучасних веб-сайтів.

2.5 JavaScript

JavaScript (JS) - це високорівнева, інтерпретована мова програмування, що використовується для розробки веб-додатків. Вона була створена для надання можливості веб-розробникам додавати динаміку до веб-сторінок. В JS використовується вбудований об'єктний підхід, що робить його дуже потужним для роботи з об'єктами та подіями.

Основні характеристики JS:

- динамічна типізація: змінні в JS не мають статичного типу даних. Тип змінної може змінюватися під час виконання програми;
- об'єктно-орієнтований підхід: JS використовує об'єктно-орієнтований підхід, де об'єкти можуть мати властивості та методи;
- прототипне наслідування: в JS використовується прототипне наслідування, де об'єкти можуть успадковувати властивості та методи від інших об'єктів;
- функціональне програмування: JS підтримує функціональне програмування, де функції можуть бути передані як аргументи іншим функціям та повертати інші функції;
- асинхронність: JS підтримує асинхронне програмування, що дозволяє виконувати деякі операції паралельно і не блокувати виконання інших операцій;
- DOM (Document Object Model): JS використовує DOM для взаємодії з веб-сторінкою. DOM - це інтерфейс, який представляє структуру документа веб-сторінки та дозволяє JS змінювати цю структуру;
- події: JS може реагувати на події, які відбуваються на сторінці, такі як клік миші, натискання клавіші тощо;

- модульність: JS підтримує модульність, що дозволяє розділяти код на логічні модулі, що полегшує його обслуговування та розширення;
- стандартизація: JS має стандарт ECMAScript, який визначає базову структуру мови та її функціональність.

JS є важливою складовою веб-розробки, оскільки вона дозволяє створювати інтерактивні та динамічні веб-сайти. Ця мова має багато бібліотек та фреймворків, які полегшують розробку веб-додатків (рис 2.3).

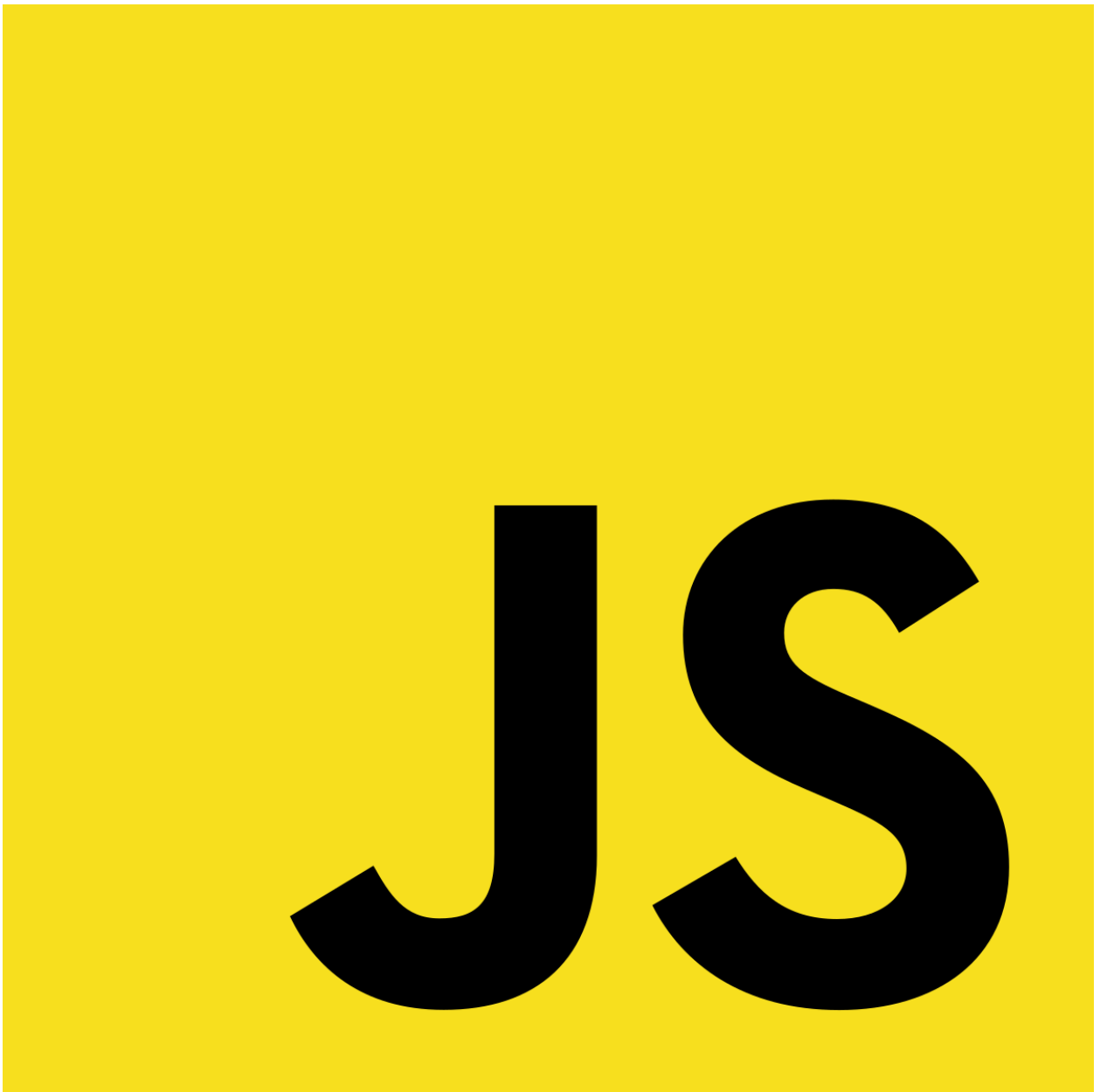


Рисунок 2.3 - Логотип JS

2.6 Огляд реалізації та коду Frontend частини проєкту:

Верстка головної сторінки:

Мета-дані та підключення стилів:

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="css/reset.css">
<link rel="stylesheet" href="css/style.css">
<title>DC</title>
```

- кодування: встановлюється на UTF-8, що забезпечує підтримку більшості мов;
- адаптивність: мета-тег viewport налаштовує масштабування сторінки на мобільних пристроях;
- стилі: підключення файлів стилів для скидання стандартних стилів (reset.css) та стилізація елементів (style.css).

Верхній блок навігації та заголовок

```
<section class="header">
  <div class="header-container">
    <div class="header-nav">
      <a href="index.html" class="llogo"></a>
      <ul class="header-nav-items">
        <li><a href="catalog.html">NEW CARS</a></li>
        <li><a href="#">PRE-OWNED CARS</a></li>
        <li><a href="#">FINANCE</a></li>
        <li><a href="#">LEASE VS BUY</a></li>
        <li><a href="contact.html">CONTACT US</a></li>
      </ul>
    </div>
    <div class="header-main">
      <h1>Leasing From D&M</h1>
      <div class="button">
        <a href="#">FIND A CAR</a>
```

```

    </div>
  </div>
</div>
</section>

```

- навігація: містить логотип та основні пункти меню для переходу по сайту;
- заголовок: основний заголовок сторінки з кнопкою для пошуку автомобіля (рис 2.4).

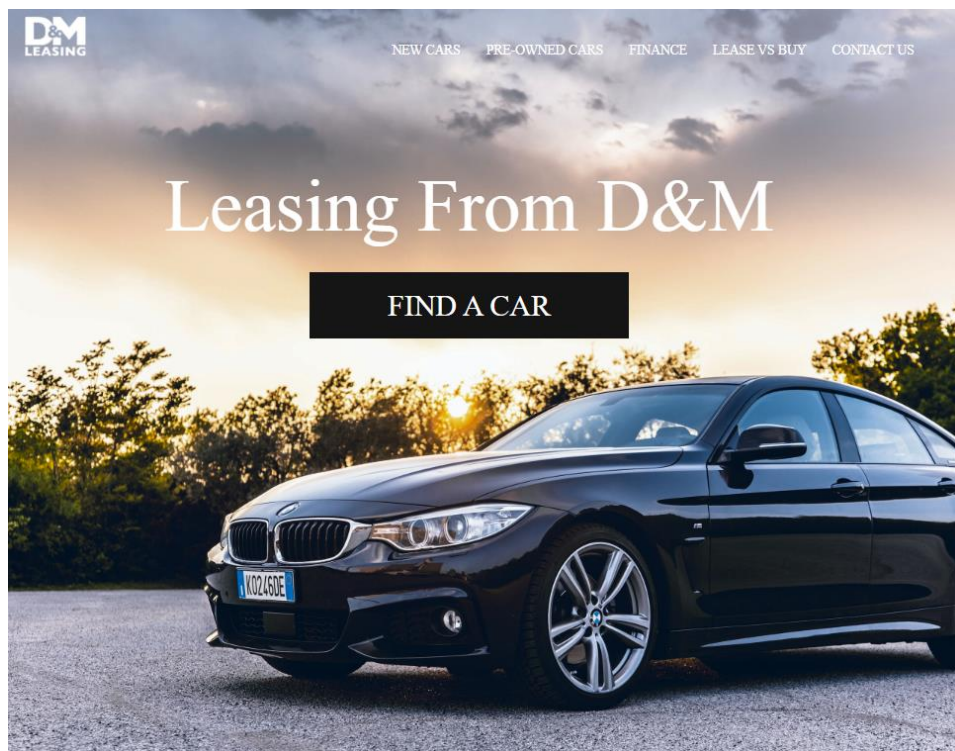


Рисунок 2.4 - Навігація сайту

Переваги лізингу з D&M

```

<section class="main">
  <div class="main-container">
    <div class="main__advantages">
      <h1>Why Lease with D&M?</h1>
      <div class="advantages__boxes">
        <!-- Пункти з перевагами -->
        <div class="advantages__item">

```

Кафедра інтелектуальних інформаційних систем

Вебзастосунок для вибору авто за критеріями

```

```

```
<h4>Top Dealers</h4>
```

```
<p>TrueCar Certified Dealers compete for your business by providing discounted pricing on in-stock inventory.</p>
```

```
</div>
```

```
<!-- Інші пункти -->
```

```
</div>
```

```
</div>
```

- заголовок та опис: описують переваги лізингу з D&M;
- картки переваг: кожна картка містить зображення, заголовок та опис (рис 2.5).

Why Lease with D&M?





			
Top Dealers	Discounted Pricing	Lower Payments	Easy Upgrade
TrueCar Certified Dealers compete for your business by providing discounted pricing on in-stock inventory.	View local inventory from the comfort of home and get access to guaranteed savings on the car of your choice.	Get discounted pricing when you shop for a car or truck using TrueCar and save on your lease payment.	Easy to upgrade to the newest vehicle model every few years if you would like to be modern and comfortable

Рисунок 2.5 - Переваги застосунку

Як працює лізинг:

```
<div class="howworks">
  <h1>How Does Leasing Work?</h1>
  <p>Leasing a vehicle is essentially entering into a long term rental
...</p>
  <h2>Similar to a short term car rental, there are rules that define:</h2>
  <div class="howworks_boxes">
    <div class="div">
      
      <p>How much you can drive, stated as annual miles ...</p>
    </div>
  </div>
```

```
<!-- Інші пункти -->
```

```
</div>
```

```
</div>
```

- заголовок та опис: пояснюють, як працює лізинг;
- картки правил лізингу: містять зображення та опис основних правил (рис 2.6).

How Does Leasing Work?

Leasing a vehicle is essentially entering into a long term rental agreement for that vehicle. Unlike a traditional car purchase, you don't actually own the vehicle. Instead, a leasing company purchases the vehicle from the dealer on your behalf and then you make monthly payments to the leasing company for the duration of your lease. Some leases however, do provide the option to purchase the vehicle at the end of the lease.

Similar to a short term car rental, there are rules that define:

- 1 How much you can drive, stated as annual miles (excess miles are charged at a specified rate)
- 2 Types of damage that you may be liable for at the end of your lease
- 3 The procedures and costs when you return the vehicle at the end of your lease

Lease contracts are a commitment and ending them early can be difficult and expensive so it is important to consider your circumstances carefully before you decide.

Рисунок 2.6 - Як працює лізинг

Відео:

```

<div class="video">
  <h1>WELCOME TO D&M LEASING</h1>
  <h2>Find out how easy it is to lease from D&M</h2>
  <iframe
    width="1000"
    height="500"
    src="https://www.youtube.com/embed/kBTYyLw7jQM?si=0MRyMMWrH_MMuMow"
    allowfullscreen
  </iframe>
</div>

```

- заголовок та підзаголовок: привітання та пояснення щодо простоти лізингу з D&M;
- відео: вставка відео з YouTube для додаткової інформації (рис 2.7).

WELCOME TO D&M LEASING

Find out how easy it is to lease from D&M



Рисунок 2.7 - Додавання відео на сайт

Важливі речі для розгляду:

```

<div class="important">
  <h1>Important Things to Consider</h1>

```

```
<h2>Depending on your circumstances and what is important to you, ...</h2>
<div class="important_boxes">
  <div class="div">
    
    <p>Monthly payments on leases are generally 20% - 30% cheaper
than loans</p>
  </div>
  <!-- Інші пункти -->
</div>
</div>
```

- заголовок та підзаголовок: пояснюють важливі аспекти лізингу;
- картки важливих аспектів: містять зображення та опис основних факторів для розгляду (рис 2.8).

Important Things to Consider

Depending on your circumstances and what is important to you, a lease could be a great option for your next vehicle. Below are some important factors that may influence your decision.



Monthly payments on leases are generally 20% - 30% cheaper than loans



You may be eligible for a tax break if you use the leased vehicle for business purposes



Easy to upgrade to the newest vehicle model every few years



There are often fees charged if you drive more than your lease's mileage allowance

Рисунок 2.8 - Вигляд блоку “Important Things to Consider”

Нижній блок (футер):

```
<section class="footer">
  <div class="footer-container">
    <div class="footer_nav">
      <div class="footer_products">
        <h1>Products</h1>
        <ul>
          <li><a href="#">Used</a></li>
          <li><a href="#">New</a></li>
          <li><a href="#">Sell your car</a></li>
```

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

</div>

<div class="footer__resources">

<h1>Resources</h1>

Blog

FAQ

Contact us

</div>

<div class="footer__workWithTrueCar">

<h1>Work With TrueCar</h1>

Dealers

Partners

</div>

<div class="footer__about">

<h1>About</h1>

About us

Team

Careers

</div>

</div>

<div class="footer__stores">

</div>

<p>For questions about the TrueCar Auto Buying Service please call 1-

888-878-3227.

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

Certified Dealers are contractually obligated by TrueCar to meet certain customer service requirements and complete the TrueCar Dealer Certification Program.

TrueCar does not broker, sell, or lease motor vehicles. Unless otherwise noted, all vehicles shown on this website are offered for sale by licensed motor vehicle dealers. All vehicles are subject to prior sale. By accessing this website, you agree to the TrueCar Terms of Service and Privacy Policy.</p>

```
<div class="footer__end">
  <ul>
    <li><a href="#">Terms of Service</a></li>
    <li><a href="#">Privacy Policy</a></li>
    <li><a href="#">Do Not Sell My Personal Information</a></li>
  </ul>
</div>
</div>
</section>
```

- навігація у футері: містить різні секції з посиланнями на сторінки про продукти, ресурси, роботу з TrueCar та інформацію про компанію;
- посилання на магазини додатків: іконки з посиланнями на App Store та Google Play;
- текстова інформація: описує умови використання та політику конфіденційності;
- кінцевий блок: містить посилання на умови використання, політику конфіденційності та інше (рис 2.9).

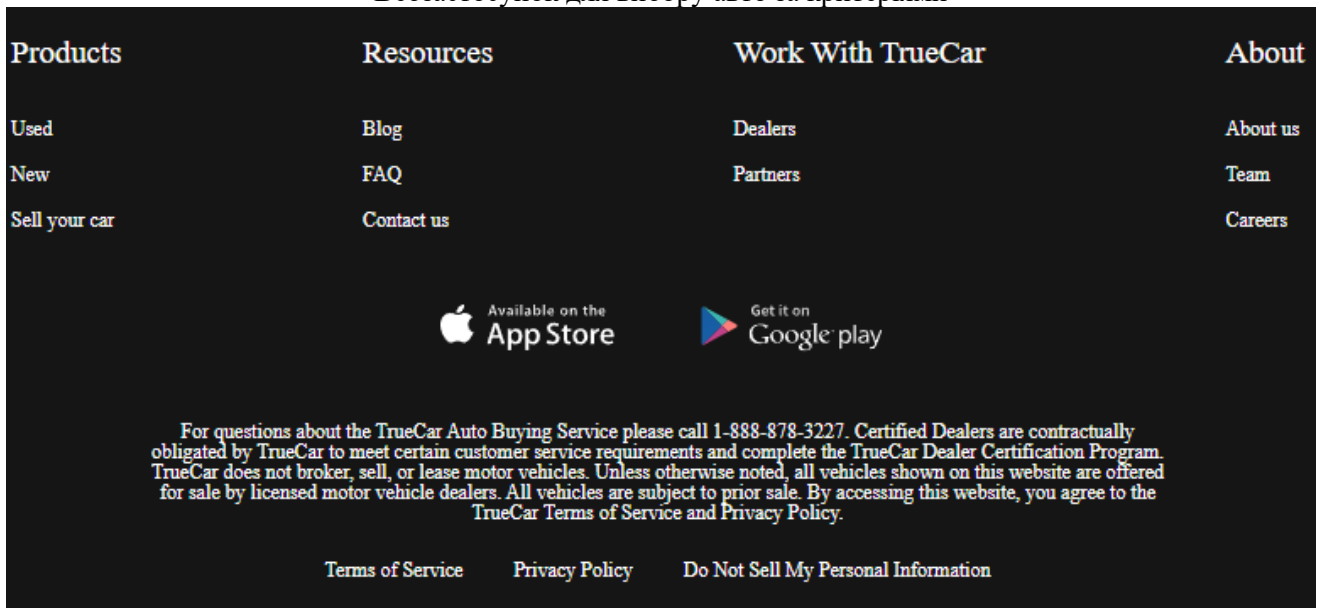


Рисунок 2.9 - Вигляд блоку футер

Огляд коду для сторінки "Contact"

Форма зворотного зв'язку:

```
<section class="mainContact">
  <div class="mainContact-container">
    <div class="contact__us">
      <h1>Contact Us</h1>
      <h2>Feel free to contact us with questions, potential partnerships
or media inquiries</h2>
      <form>
        <label for="name">Name:</label><br>
        <input type="text" id="name" name="name"><br><br>
        <label for="email">E-mail:</label><br>
        <input type="email" id="email" name="email"><br><br>
        <label for="message">Your text:</label><br>
        <textarea id="message" name="message" rows="4"
cols="50"></textarea><br><br>
        <input type="submit" value="Submit" class="formbutton">

```

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

```
</form>
</div>
```

- заголовок та підзаголовок: пояснюють мету форми зворотного зв'язку;
- форма: містить поля для імені, електронної пошти та повідомлення, а також кнопку відправки (рис 2.10).

Contact Us

Feel free to contact us with questions, potencial partnerships or media inquiries

Name:

E-mail:

Your text:

Submit

Рисунок 2.10 - форма зворотнього зв'язку

Останні новини:

```
<div class="mainCatalog_news">
  <div class="news">
    
    <h1>An easier way to buy your next Toyota</h1>
    <div class="button">
      <a href="#">more</a>
    </div>
  </div>
</div>
```

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

```

<div class="news">
  
  <h1>A Guide to Road-Tripping With Pets</h1>
  <div class="button">
    <a href="#">more</a>
  </div>
</div>
</div>
</div>
</section>

```

– новини: містять зображення, заголовки та кнопку для переходу до детальнішої інформації (рис 2.11).



An easier way to buy your next Toyota

more



A Guide to Road-Tripping With Pets

more

Рисунок 2.11 - Вид блоку новин

Підключення JavaScript файлу:

```
<script src="js/contact.js"></script>
```

– JavaScript: підключення файлу для обробки подій та функціональності на сторінці.

Сторінка Catalog (рис 2.12)

Основний каталог (Main Catalog):

```

<section class="mainCatalog">
  <div class="mainCatalogContainer">
    <div class="mainCatalog__auto">

```

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

```
<h1>Choose your car</h1>
```

```
<div class="filters">
```

```
  <label for="colorFilter">Color:</label>
```

```
  <select id="colorFilter">
```

```
    <option value="all">All</option>
```

```
    <option value="white">White</option>
```

```
    <option value="black">Black</option>
```

```
    <option value="blue">Blue</option>
```

```
</select>
```

```
  <label for="bodyTypeFilter">Body Type:</label>
```

```
  <select id="bodyTypeFilter">
```

```
    <option value="all">All</option>
```

```
    <option value="sedan">Sedan</option>
```

```
    <option value="suv">SUV</option>
```

```
    <option value="coupe">Coupe</option>
```

```
</select>
```

```
  <label for="priceFilter">Max Price:</label>
```

```
  <input type="number" id="priceFilter" placeholder="Enter max price">
```

```
  <button onclick="filterCars()">Filter</button>
```

```
</div>
```

```
<div id="carList">
```

```
</div>
```

```
<div class="button">
```

```
  <a href="#">SHOW MORE</a>
```

```
</div>
```

```
</div>
```

```
<div class="mainCatalog__news">
```

```
  <div class="news">
```

```
    
```

```
    <h1>An easier way to buy your next Toyota</h1>
```

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

```

<div class="button">
    <a href="#">more</a>
</div>
</div>
<div class="news">
    
    <h1>A Guide to Road-Tripping With Pets</h1>
    <div class="button">
        <a href="#">more</a>
    </div>
</div>
</div>
<div class="button2">
    <a href="#">SHOW MORE</a>
</div>
</div>
</section>

```

- фільтри: дозволяють користувачам фільтрувати автомобілі за кольором, типом кузова та ціною;
- список автомобілів: динамічний список, який буде наповнюватися за допомогою JavaScript;
- новини: два блоки з новинами та кнопками для додаткової інформації.

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями
Choose your car

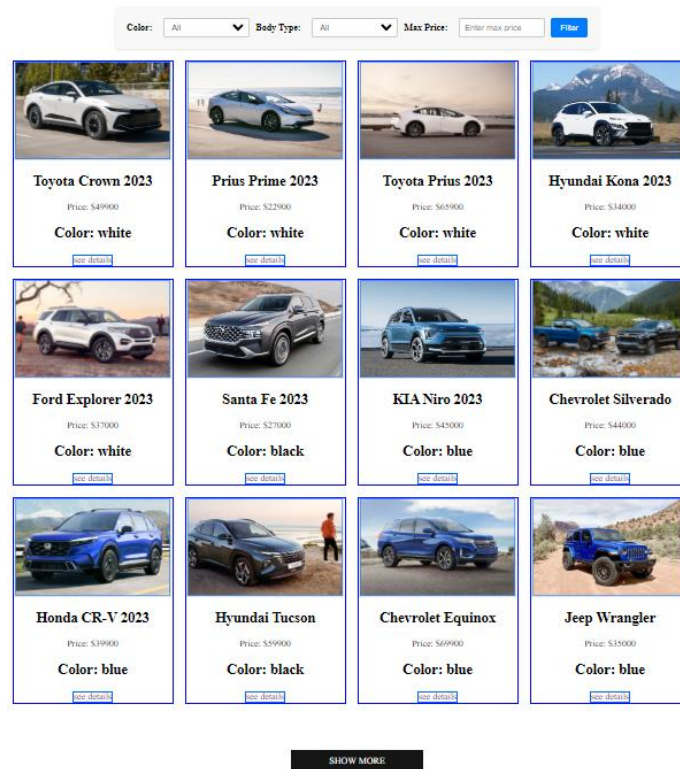


Рисунок 2.12 - Вид блоку каталог

Огляд frontend частини бота:

Заголовок (Head):

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/css/stylebot.css" type="text/css"
media="all" />
  <title>Chatbot</title>
  <script>
    function sendMessage() {
      var message = document.getElementById("message").value;

      var xhr = new XMLHttpRequest();
      xhr.open("POST", "chatbot.php", true);
```

```

xhr.setRequestHeader("Content-Type", "application/x-www-form-
urlencoded");

xhr.onreadystatechange = function () {
    if (xhr.readyState == 4 && xhr.status == 200) {
        var chatBox = document.getElementById("chat-box");
        chatBox.innerHTML += "<p><strong>You:</strong> " +
message + "</p>";
        chatBox.innerHTML += "<p><strong>Bot:</strong> " +
xhr.responseText + "</p>";
        document.getElementById("message").value = "";
    }
};

xhr.send("message=" + encodeURIComponent(message));
}
</script>
</head>
<body>

```

- DOCTYPE: Використовується HTML5, що є сучасним і стандартним;
- мета-теги: charset та viewport забезпечують правильне відображення та кодування сторінки (рис 2.13);
- стилі: підключення CSS файлу для стилізації сторінки;
- JavaScript: Функція sendMessage для обробки відправки повідомлень користувача.

You:
Порадь позашляховик ціною до 10000\$

Bot:
Одним з найпопулярніших позашляховиків у ціновому діапазоні до 10000\$ є Jeep Cherokee. Він має надійну конструкцію, високий простір підлоги і може бути оснащений повноприводною системою. Іншим варіантом може бути Toyota RAV4, яка також відома своєю надійністю та здатністю на бездоріжжя. Для тих, хто шукає більш компактний позашляховик, можна розглянути Honda CR-V або Nissan Rogue. Вони також пропонують добру маневреність та комфортну кабіну. Варто відзначити, що ціни на вживані автомобілі можуть варіюватися в залежності від року випуску, пробігу та стану транспортного засобу.

Send

Рисунок 2.13 - Frontend частина бота

Тіло документа (Body):

```

<body>
  <div class="chat-container">
    <div id="chat-box" class="chat-box"></div>
    <form onsubmit="event.preventDefault(); sendMessage();">
      <input type="text" id="message" class="chat-input"
placeholder="Enter your message...">
      <button type="submit" class="chat-button">Send</button>
    </form>
  </div>
</body>
</html>

```

- контейнер чату: основний контейнер для чату, який містить область для відображення повідомлень та форму для відправки нових повідомлень;
- область повідомлень: div з id="chat-box" для відображення історії повідомлень;
- форма відправки повідомлень: поле вводу для повідомлень та кнопка відправки. Використовується onsubmit атрибут для обробки подій.

Огляд файлу JS:

Цей JavaScript-код реалізує функцію фільтрації для масиву автомобілів на основі кольору, типу кузова та максимальної ціни. Також він відповідає за відображення відфільтрованих автомобілів на сторінці. Ось детальний аналіз та рекомендації:

Масив автомобілів:

```
const cars = [  
  { id: 1, name: 'Toyota Crown 2023', price: 49900, color: 'white',  
    bodyType: 'sedan', image: 'img/autoc1.svg'},  
  { id: 2, name: 'Prius Prime 2023', price: 22900, color: 'white', bodyType:  
    'coupe', image: 'img/autoc2.svg'},  
  { id: 3, name: 'Toyota Prius 2023', price: 65900, color: 'white',  
    bodyType: 'sedan', image: 'img/autoc3.svg'},  
  { id: 4, name: 'Hyundai Kona 2023', price: 34000, color: 'white',  
    bodyType: 'suv', image: 'img/autoc4.svg'},  
  { id: 5, name: 'Ford Explorer 2023', price: 37000, color: 'white',  
    bodyType: 'suv', image: 'img/autoc5.svg'},  
  { id: 6, name: 'Santa Fe 2023', price: 27000, color: 'black', bodyType:  
    'suv', image: 'img/autoc6.svg'},  
  { id: 7, name: 'KIA Niro 2023', price: 45000, color: 'blue', bodyType:  
    'suv', image: 'img/autoc7.svg'},  
  { id: 8, name: 'Chevrolet Silverado', price: 44000, color: 'blue',  
    bodyType: 'suv', image: 'img/autoc8.svg'},  
  { id: 9, name: 'Honda CR-V 2023', price: 39900, color: 'blue', bodyType:  
    'suv', image: 'img/autoc9.svg'},  
  { id: 10, name: 'Hyundai Tucson', price: 59900, color: 'black', bodyType:  
    'suv', image: 'img/autoc10.svg'},  
  { id: 11, name: 'Chevrolet Equinox', price: 69900, color: 'blue',  
    bodyType: 'suv', image: 'img/autoc11.svg'},  
]
```

```
{ id: 12, name: 'Jeep Wrangler', price: 35000, color: 'blue', bodyType:
'suv', image: 'img/autoc12.svg'},
];
```

– масив cars містить об'єкти з даними про автомобілі, включаючи id, name, price, color, bodyType, та image.

Функція фільтрації:

```
function filterCars() {
  const selectedColor = document.getElementById('colorFilter').value;
  const selectedBodyType =
document.getElementById('bodyTypeFilter').value;
  const maxPrice = document.getElementById('priceFilter').value;
  const filteredCars = cars.filter(car => {
    const colorMatch = selectedColor === 'all' || car.color ===
selectedColor;
    const bodyTypeMatch = selectedBodyType === 'all' || car.bodyType ===
selectedBodyType;
    const priceMatch = !maxPrice || car.price <= maxPrice;
    return colorMatch && bodyTypeMatch && priceMatch;
  });
  renderCars(filteredCars);
}
```

– функція filterCars зчитує значення фільтрів для кольору, типу кузова та максимальної ціни;

– використовує метод filter для створення нового масиву filteredCars, який містить лише ті автомобілі, що відповідають критеріям фільтрації;

– викликає функцію renderCars для відображення відфільтрованих автомобілів.

Функція відображення:

```
function renderCars(cars) {
  const carList = document.getElementById('carList');
  carList.innerHTML = '';
  cars.forEach(car => {
```

Кафедра інтелектуальних інформаційних систем

Вебзастосунок для вибору авто за критеріями

```
const carItem = document.createElement('div');

carItem.className = 'car';

carItem.innerHTML = `<h1>${car.name}</h1><h2>Price:
${car.price}</h2><h3>Color:
${car.color}</h3><a href="#">see details</a>`;

carList.appendChild(carItem);

});

}

// Display all cars on initial load
document.addEventListener("DOMContentLoaded", () => {
  renderCars(cars);
});
```

- функція `renderCars` приймає масив автомобілів і відображає їх на сторінці;
- очищує вміст контейнера `carList` перед додаванням нових елементів;
- використовує метод `forEach` для створення HTML-елементів для кожного автомобіля та додає їх до `carList`;
- під час завантаження сторінки (`DOMContentLoaded`) відображає всі автомобілі.

Висновки до розділу 2

Розробка фронтенд частини та написання основного інтерфейсу вебзастосунку для підбору авто за критеріями є ключовим етапом, який визначає зручність користування і ефективність взаємодії користувачів із системою. Зосередження уваги на інтуїтивному інтерфейсі користувача гарантує, що застосунок буде простим у використанні навіть для новачків, забезпечуючи зручність і швидкість доступу до необхідної інформації.

Особливу увагу приділено формі підбору автомобілів, яка є основним інструментом для користувачів. Ця форма дозволяє користувачам задавати різноманітні критерії пошуку, такі як діапазон цін, тип кузова та колір автомобіля. Після заповнення форми та натискання кнопки пошуку, система надає список автомобілів, які відповідають заданим параметрам. Цей список включає основну інформацію про кожне авто, зокрема модель, рік випуску та фото, що дозволяє користувачам отримати повну картину про доступні варіанти.

Для покращення взаємодії з користувачами реалізована форма зворотного зв'язку, яка дозволяє відправляти відгуки та запити. Ця форма містить поля для введення імені, електронної пошти та тексту повідомлення, забезпечуючи зручний спосіб зв'язку з адміністрацією сайту.

У сукупності, всі ці особливості забезпечують високий рівень зручності, функціональності та безпеки вебзастосунку, роблячи його ефективним інструментом для користувачів, які шукають автомобіль своєї мрії. Вимоги до функціональності та технічної реалізації були розглянуті і деталізовані, включаючи використання сучасних технологій HTML5, CSS3, та JavaScript.

Завдяки цим зусиллям, вебзастосунок здатний ефективно задовольнити потреби своїх користувачів, надаючи їм інтуїтивний, зручний і швидкий інструмент для підбору автомобілів за різними критеріями.

3 ПРОЄКТУВАННЯ BACKEND ЧАСТИНИ ВЕБЗАСТОСУНКУ НА ОСНОВІ GPT API

3.1 Огляд використаних backend технологій

Розміщення сайту на хостингу:

– хостинг: хостинг - це сервіс, який надається компанією для зберігання файлів вашого сайту та його доступу в Інтернеті. Хостинг може бути спільним (де ви ділите сервер з іншими користувачами) або присвяченим (де сервер призначений лише для вашого сайту);

– вибір хостингу: при виборі хостингу в Україні важливо враховувати такі критерії, як потужність серверів, доступність підтримки, ціна та можливості розширення. Деякі популярні хостинг-провайдери в Україні включають "Хостинг Україна", "Ukrnames" та "Хостинг.ua";

– параметри хостингу: придбавши хостинг, вам будуть надані параметри для доступу до вашого облікового запису, FTP-доступ для завантаження файлів сайту, а також можливість налаштування бази даних та інших налаштувань сервера;

– розміщення файлів: після отримання доступу до хостингу ви можете завантажити файли свого сайту на сервер за допомогою FTP-клієнта або інших доступних інструментів (рис 3.1).

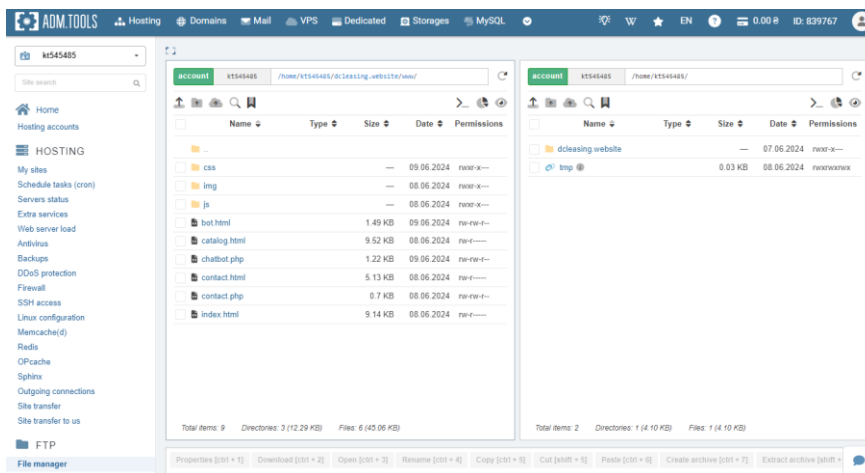


Рисунок 3.1 - Вигляд каталогу сайту на хостингу

Оренда домену:

- домен: домен - це унікальне ім'я, за допомогою якого користувачі знаходять ваш сайт в Інтернеті. Наприклад, "example.com" є доменом;
- реєстратор доменів: для оренди домену вам потрібно звернутися до реєстратора доменів, який є організацією, що управляє доменами у певній доменній зоні. Наприклад, в Україні реєстратором може бути "Ukrnames";
- вибір домену: при виборі домену важливо враховувати його унікальність, легкість запам'ятовування та відповідність вашому бренду або назві компанії;
- процес оренди: для оренди домену вам потрібно вибрати доступний домен, вказати свої контактні дані та здійснити оплату відповідно до тарифів реєстратора. Після цього домен буде зареєстровано на ваше ім'я на певний термін (зазвичай від 1 до 10 років);
- подовження домену: після закінчення терміну дії домену ви можете подовжити його оренду за допомогою реєстратора (рис 3.2).

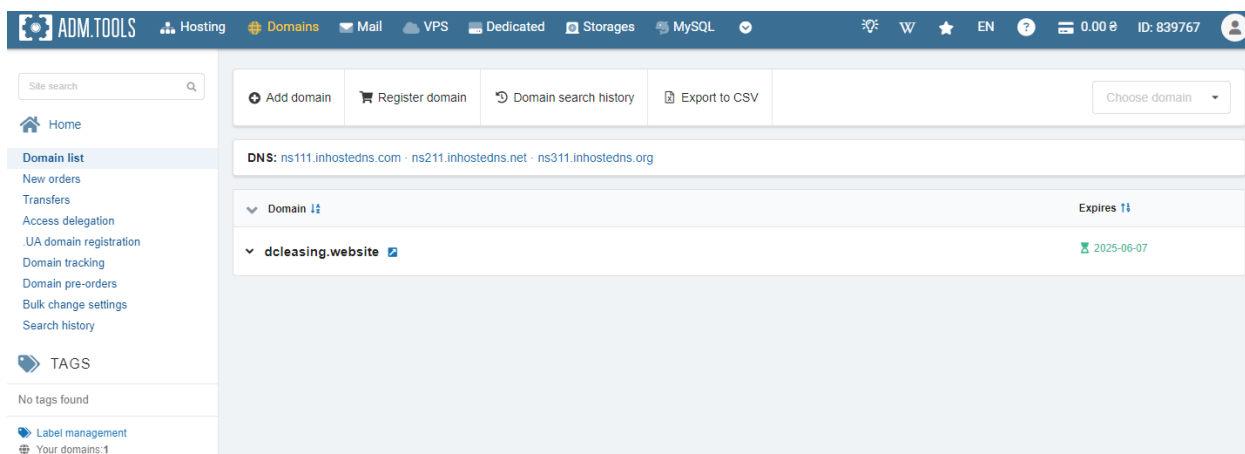


Рисунок 3.2 – Орендований домен сайту

Мова програмування PHP

PHP (Hypertext Preprocessor) - це широко використовувана серверна мова програмування, створена спеціально для веб-розробки. PHP дозволяє створювати динамічні веб-сторінки та взаємодіяти з базами даних.

Історія PHP:

- розробка та випуск: PHP був створений Расмусом Лердорфом у 1994 році як набір CGI-скриптів на мові С для відстеження відвідувань його онлайн-резюме. Оригінальна назва була Personal Home Page (PHP);
- еволюція: з часом мова розвивалася та перетворилася в повноцінний інструмент для веб-розробки. У 1997 році був випущений PHP/FI 2.0, і у 1998 році був випущений PHP 3.0, що привернув велику кількість розробників;
- сучасні версії: сьогодні PHP продовжує розвиватися, з останніми версіями (на момент написання) PHP 8.0 і PHP 8.1, що включають численні нові можливості та оптимізації.

Основні Можливості PHP:

- серверна мова програмування: PHP працює на сервері, генеруючи HTML, який потім відправляється клієнту;
- інтеграція з HTML: PHP можна вбудовувати безпосередньо в HTML-код, що робить його зручним для створення динамічних веб-сторінок.
- підтримка баз даних: PHP підтримує численні бази даних, включаючи MySQL, PostgreSQL, Oracle, SQLite та інші;
- широкий набір функцій: PHP має багатий набір вбудованих функцій для роботи з файлами, рядками, сесіями, куки та іншими важливими аспектами веб-розробки;
- розширюваність: PHP можна розширити за допомогою власних модулів та інтеграції з іншими мовами програмування.

Переваги PHP:

- простота та зручність у використанні: PHP легко навчитися і використовувати, особливо для початківців у веб-розробці;
- відкритий код: PHP є безкоштовним, з відкритим вихідним кодом, що робить його доступним для всіх;
- широка підтримка: завдяки великій спільноті розробників, існує багато ресурсів, документації, форумів та бібліотек для PHP;

- портативність: PHP можна використовувати на будь-якій платформі, яка підтримує веб-сервери, такі як Apache, Nginx та інші;
- висока продуктивність: завдяки оптимізації та кешуванню, PHP може ефективно обробляти великі обсяги запитів.

PHP залишається однією з найбільш популярних мов програмування для веб-розробки завдяки своїй простоті, гнучкості та широкій підтримці (рис 3.3). Його здатність інтегруватися з HTML, взаємодіяти з базами даних та підтримувати різноманітні функції робить його потужним інструментом для створення динамічних веб-додатків. З постійними оновленнями та поліпшеннями PHP продовжує бути важливим інструментом для веб-розробників у всьому світі.



Рисунок 3.3 - Логотип PHP

3.2 Розбір коду на PHP

Файл `contact.php`:

```
<?php
// Перевіряє, чи метод запиту є POST
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Отримує дані з POST-запиту та застосовує функцію htmlspecialchars для
    захисту від XSS-атак
```

Кафедра інтелектуальних інформаційних систем
 Вебзастосунок для вибору авто за критеріями

```

$name = htmlspecialchars($_POST['name']);
$email = htmlspecialchars($_POST['email']);
$message = htmlspecialchars($_POST['message']);

// Встановлює адресу одержувача
$to = "ruslanshapar2003@gmail.com"; // Змініть на вашу електронну адресу

// Встановлює тему листа
$subject = "DC Leasing";

// Створює тіло листа
$body = "Name: $name\nEmail: $email\nMessage: $message";

// Встановлює заголовок From для листа
$headers = "From: ruslanshapar2003@gmail.com"; // Змініть на вашу електронну
адресу

// Використовує функцію mail для відправки листа
if (mail($to, $subject, $body, $headers)) {
    // Якщо лист успішно відправлений, виводить повідомлення про успіх
    echo "Form submitted successfully";
} else {
    // Якщо відправка не вдалася, виводить повідомлення про невдачу
    echo "Form submission failed";
}
} else {
    // Якщо метод запиту не POST, виводить повідомлення про недійсний метод
запиту
    echo "Invalid request method";
}
?>

```

Пояснення коду:

<?php: Відкриває блок PHP-коду.

if (\$_SERVER["REQUEST_METHOD"] == "POST") {: Перевіряє, чи метод запити є POST. Це потрібно для того, щоб впевнитися, що форма була відправлена через POST, а не GET;

\$name = htmlspecialchars(\$_POST['name']); Отримує значення з поля name форми та застосовує функцію htmlspecialchars для запобігання XSS-атак. Те саме робиться для полів email та message;

\$to = "ruslanshapar2003@gmail.com"; Встановлює адресу одержувача електронного листа;

\$subject = "DC Leasing"; Встановлює тему електронного листа;

\$body = "Name: \$name\nEmail: \$email\nMessage: \$message"; Формує тіло електронного листа з даними, отриманими з форми;

\$headers = "From: ruslanshapar2003@gmail.com"; Встановлює заголовок From для електронного листа;

if (mail(\$to, \$subject, \$body, \$headers)) {: Використовує функцію mail для відправки електронного листа. Якщо лист успішно відправлений, виконується блок коду всередині if;

echo "Form submitted successfully"; Виводить повідомлення про успішне відправлення форми;

} else {: Якщо відправка листа не вдалася, виконується блок коду всередині else;

echo "Form submission failed"; Виводить повідомлення про невдачу відправлення форми;

} else {: Якщо метод запити не POST, виконується блок коду всередині else;

echo "Invalid request method"; Виводить повідомлення про недійсний метод запити;

}: Закриває блок коду if;

?>: Закриває блок PHP-коду.

Розбір файлу bot.php

Цей код використовує API OpenAI для отримання відповіді від моделі GPT-3.5-turbo-16k. Він приймає повідомлення від користувача через POST-запит, надсилає це повідомлення до API OpenAI і повертає відповідь від бота:

```
<?php
// Задаємо ключ API для доступу до сервісу OpenAI
$apiKey = "sk-proj-hJLpvyqXBTYbvYR1i9NST3B1bkFJpZWFVvkzA5KGUGmPp75jP";
// URL для запиту до API OpenAI
$url = "https://api.openai.com/v1/chat/completions";
// Отримуємо повідомлення від користувача через POST-запит
$message = isset($_POST['message']) ? $_POST['message'] : '';
// Формуємо дані для запиту до API у форматі JSON
$data = json_encode(array(
    'model' => 'gpt-3.5-turbo-16k',
    'messages' => array(
        array('role' => 'user', 'content' => $message)
    ),
    'max_tokens' => 4000, // Збільшуємо кількість токенів
    'temperature' => 0.7
));
// Ініціалізуємо cURL сесію для запиту до API
$ch = curl_init($url);
// Налаштовуємо cURL для повернення відповіді у вигляді рядка
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
// Вказуємо, що запит буде POST
curl_setopt($ch, CURLOPT_POST, true);
// Вказуємо дані для відправлення у POST-запиті
curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
// Вказуємо заголовки, включаючи тип контенту та авторизаційний ключ
curl_setopt($ch, CURLOPT_HTTPHEADER, array(
    'Content-Type: application/json',
    'Authorization: Bearer ' . $apiKey
));
```

```
// Виконуємо cURL запит та зберігаємо відповідь
$response = curl_exec($ch);

// Перевіряємо на наявність помилок у cURL
if (curl_errno($ch)) {
    // Виводимо помилку, якщо така є
    echo 'Error:' . curl_error($ch);
} else {
    // Декодуємо JSON-відповідь від API
    $responseData = json_decode($response, true);
    // Перевіряємо наявність відповіді від бота
    if (isset($responseData['choices'][0]['message']['content'])) {
        // Отримуємо відповідь від бота
        $botResponse = $responseData['choices'][0]['message']['content'];
        // Виводимо відповідь
        echo $botResponse;
    } else {
        // Виводимо повідомлення про помилку, якщо відповідь неправильна
        echo 'Error: Invalid response from API';
        // Додатковий відладковий вивід для аналізу відповіді API
        echo '<pre>';
        print_r($responseData);
        echo '</pre>';
    }
}

// Закриваємо cURL сесію
curl_close($ch);

?>
```

Пояснення:

- `<?php:` Відкриває блок PHP-коду;
- `$apiKey` = `"sk-proj-hJLpvyqXBTYbvYR1i9NST3BlbkFJpZWFVkHzA5KGUGmPp75jP"`; Задає ключ API для доступу до сервісу OpenAI;

- `$url = "https://api.openai.com/v1/chat/completions";` Задає URL для запиту до API OpenAI;
- `$message = isset($_POST['message']) ? $_POST['message'] : "";` Отримує повідомлення від користувача через POST-запит, якщо таке повідомлення є;
- `$data = json_encode(array(...));` Формує дані для запиту до API у форматі JSON. Тут вказується модель (`gpt-3.5-turbo-16k`), повідомлення користувача, максимальна кількість токенів (`max_tokens`), та `temperature` для регулювання креативності відповіді;
- `$ch = curl_init($url);` Ініціалізує cURL сесію для запиту до API;
- `curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);` Налаштовує cURL для повернення відповіді у вигляді рядка;
- `curl_setopt($ch, CURLOPT_POST, true);` Вказує, що запит буде POST;
- `curl_setopt($ch, CURLOPT_POSTFIELDS, $data);` Вказує дані для відправлення у POST-запиті;
- `curl_setopt($ch, CURLOPT_HTTPHEADER, array(...));` Вказує заголовки, включаючи тип контенту та авторизаційний ключ;
- `$response = curl_exec($ch);` Виконує cURL запит та зберігає відповідь.
- `if (curl_errno($ch)) { ... } else { ... };` Перевіряє наявність помилок у cURL. Якщо є помилка, виводить її. Інакше декодує JSON-відповідь від API;
- `if (isset($responseData['choices'][0]['message']['content'])) { ... } else { ... };` Перевіряє наявність відповіді від бота. Якщо є відповідь, виводить її. Інакше виводить повідомлення про помилку і додатково відображає повну відповідь API для відладки;
- `curl_close($ch);` Закриває cURL сесію.

Висновки до розділу 3

Розділ 3 висвітлює процес проєктування та розробки backend частини вебзастосунку з використанням PHP та інтеграції ШІ, зокрема OpenAI GPT-3.5. Основна увага приділяється розміщенню сайту на хостингу, оренді домену, та написанню PHP-коду для обробки форми зворотного зв'язку та інтеграції чат-бота.

Обговорено вибір хостингу та процес оренди домену. Хостинг є критичним аспектом, оскільки він забезпечує зберігання файлів сайту та його доступність в Інтернеті. Основні критерії вибору хостингу включають потужність серверів, доступність технічної підтримки, ціну та можливості розширення. Розглянуто популярні хостинг-провайдери в Україні, такі як "Хостинг Україна" та "Ukrnames". Оренда домену є важливим кроком для забезпечення унікального імені сайту, яке користувачі зможуть легко знайти в Інтернеті. Реєстратори доменів, такі як "Ukrnames", допомагають у цьому процесі.

Розглянуто роботу PHP-коду, що використовується для обробки форми зворотного зв'язку та інтеграції чат-бота. PHP є потужною серверною мовою програмування, яка дозволяє створювати динамічні веб-сторінки та взаємодіяти з базами даних. Детально розглянуто код для обробки форми зворотного зв'язку, який включає валідацію введених даних та відправлення електронного листа з отриманими даними. Окрім цього, розглянуто код для інтеграції чат-бота, який використовує API OpenAI для обробки запитів користувачів та надання відповідей на основі моделі GPT-3.5.

Розробка backend частини вебзастосунку з використанням PHP та інтеграцією ШІ забезпечує можливість створення інтерактивних та динамічних веб-сторінок. Завдяки використанню сучасних технологій, таких як OpenAI GPT-3.5, можна значно покращити взаємодію з користувачами, забезпечуючи їм швидкі та точні відповіді на їхні запити.

ВИСНОВКИ

В результаті проведення дослідження було розроблено вебзастосунок для підбору автомобілів за критеріями, який вирішує задачі підбору автомобілів за заданими користувачем критеріями.

У першому розділі проведено аналіз предметної області, поставлено задачу розробки, досліджено загальні принципи функціонування вебзастосунків для підбору автомобілів, розглянуто існуючі аналоги та визначено функціональність власної системи, а також проаналізовано важливі критерії вибору автомобілів для користувачів.

У другому розділі описано особливості створюваного застосунку, включаючи опис вимог до системи, проектування функціональності та задач, які він повинен виконувати. Описано вибір технологій для frontend частини (HTML, CSS для адаптивного дизайну, JavaScript для динамічної взаємодії).

У третьому розділі описані етапи розробки вебзастосунка, включаючи проектування backend, інтеграцію ШІ бота через gpt API, процес розробки форми зворотного зв'язку з використанням PHP та функціонал для обробки запитів користувачів.

Завдяки правильному вибору технологій та методів розробки, створено систему, яка відповідає всім вимогам, автоматизує процес підбору автомобілів, забезпечує швидкий доступ до інформації та зручність її обробки, а також надає можливість зберігання введеної інформації для подальшого використання. Вебзастосунок є унікальним помічником для користувачів, які шукають автомобілі, значно полегшуючи процес вибору та порівняння.

Кафедра інтелектуальних інформаційних систем
Вебзастосунок для вибору авто за критеріями
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Personalization in E-commerce. URL: <https://www.shopify.com/enterprise/blog/ecommerce-personalization-examples> (дата звернення: 01.05.2024).
2. The Role of Social Media in E-commerce. URL: <https://www.locate2u.com/ecommerce/the-role-of-social-media-in-e-commerce-marketing/> (дата звернення: 01.05.2024).
3. Дакетт Д., HTML та CSS: Дизайн та розробка вебсайтів. Вілямс, 2018. 480 с. URL: http://vk.academy.lv/file/Dakett_J_HTML_i_CSS_Razrobotka_i_dizayn_web_stranic.pdf (дата звернення: 05.05.2024).
4. Фленаган Д., JavaScript: Докладний посібник. Шосте видання. Москва, 2020. 1080 с. URL: https://my-js.org/assets/files/definitive_guide-7eea8a7e5bab411a133ba9d0204c214a.pdf (дата звернення: 05.05.2024).
5. Скляр Д., Вивчаємо PHP 7. Посібник зі створення інтеративних вебсайтів. Київ, 2017. 382 с. URL: https://library-it.com/wp-content/uploads/2021/05/izuchaem_php_7_rukovodstvo.pdf (дата звернення: 06.05.2024).
6. Документація EmailJS. Вебсайт. URL: <https://www.emailjs.com/docs/> (дата звернення: 06.05.2024).
7. Документація NPM. Вебсайт. URL: <https://docs.npmjs.com/> (дата звернення: 06.05.2024).
8. Gulp для початківців. URL: <https://itproger.com/ua/course/gulp> (дата звернення: 06.05.2024).
9. Open Server Panel для початківців. Документація. URL: <https://ospanel.io/docs/> (дата звернення: 06.05.2024).
10. ASOS. URL: <https://www.asos.com/us/> (дата звернення: 08.05.2024).
11. Zalando. URL: <https://zalando.com/> (дата звернення: 08.05.2024).
12. Farfetch. URL: <https://www.farfetch.com/> (дата звернення: 08.05.2024).