

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
АВТОМАТИЗОВАНИХ СИСТЕМ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка та аналіз веб-сервісу для оптимізації процесів в
автомобільному бізнесі»

на здобуття освітнього ступеня бакалавра
зі спеціальності 126 Інформаційні системи та технології
(код, найменування спеціальності)
освітньо-професійної програми Інформаційні системи та технології
(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело*

(підпис)

Кіріл Верягін
Ім'я, ПРІЗВИЩЕ здобувача

Виконав: здобувач вищої освіти гр. ІСД- 42

Кіріл Верягін
Ім'я, ПРІЗВИЩЕ

Керівник: І.ш.о., Шахматов Іван
науковий ступінь,
вчене звання
Ім'я, ПРІЗВИЩЕ

Рецензент: _____
науковий ступінь,
вчене звання
Ім'я, ПРІЗВИЩЕ

Київ 2024

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення автоматизованих систем

Ступінь вищої освіти бакалавр

Спеціальність Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

ЗАТВЕРДЖУЮ

Завідувач кафедру ІПЗАС

_____ Каміла СТОРЧАК

« ____ » _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Верягіну Кірілу Олександровичу

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Розробка та аналіз веб-сервісу для оптимізації процесів в автомобільному бізнесі

керівник кваліфікаційної роботи Іван Шахматов Олександрович

(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «27» лютого 2024 р. № 36

2. Строк подання кваліфікаційної роботи «31» травня 2024 р.

3. Вихідні дані до кваліфікаційної роботи:

1. Науково-технічна література з теми бакалаврської роботи.
2. проектування бази даних та веб-сервісу.
3. розробку веб-сайту, включаючи створення користувацького інтерфейсу і серверної частини системи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. зосереджений на аналізі існуючих процесів у автомобільній індустрії та визначенні проблем, які можуть бути оптимізовані за допомогою веб-сервісу.

2. вибір технологій та архітектурних рішень, які забезпечують високу продуктивність, надійність і масштабованість проекту.

3. Детально розглянуто процеси створення інтуїтивно зрозумілого веб-інтерфейсу

5. Ілюстративний матеріал: *презентація*

6. Дата видачі завдання: «27» лютого 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	27.02-05.03.2024	
2	Обґрунтування актуальності роботи	06.03-11.03.2024	
3	Аналіз основних прийомів	12.03-27.03.2024	
4	створення структури бази даних	28.03-10.04.2024	
5	розробка серверних сценаріїв, які забезпечують функціональність сайту, і взаємодію з базою даних через SQL-запити.	11.04-15.05.2024	
7	Оформлення роботи: вступ, висновки, реферат	16.05-22.05.2024	
8	Розробка демонстраційних матеріалів	23.05-24.05.2024	

Здобувач(ка) вищої освіти

_____ (підпис)

Кіріл Верягін
(Ім'я, ПРІЗВИЩЕ)

Керівник
кваліфікаційної роботи

_____ (підпис)

Іван Шахматов
(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавр: 57 стор., 39 рис., 21 джерела.

Мета роботи – покращення дизайну застосунків для управління розумного будинку за рахунок впровадження в його функціональну архітектуру елементів гейміфікації.

Об'єкт дослідження- процеси управління та оптимізації діяльності в автомобільному бізнесі..

Предмет дослідження – система оптимізації процесів у автомобільному бізнесі за допомогою розробки веб-сервісу.

Короткий зміст роботи: Наукова новизна полягає у розробці веб-сервісу, який інтегрується з системами управління автомобільного бізнесу для оптимізації процесів управління запасами, логістики та взаємодії з клієнтами. Застосування сучасних веб-технологій для створення масштабованої та ефективної платформи є значним внеском у вдосконалення процесів у цій галузі.

Практична значущість дослідження виявляється в здатності веб-сервісу знижувати витрати та підвищувати ефективність роботи автомобільних компаній. Результати можуть бути впроваджені для реального використання в автомобільних компаніях, покращуючи їхню конкурентоспроможність і клієнтське обслуговування.

КЛЮЧОВІ СЛОВА: АВТОМОБІЛІ, БІЗНЕС, МОДУЛЬ АНАЛІТИКИ ТА ЗВІТНОСТІ, РОЗРОБКА ДОДАТКУ, УПРАВЛІННЯ ВЗАЄМОДІЯМИ З КЛІЄНТАМИ

ABSTRACT

The textual part of the qualifying work for obtaining a bachelor's degree: 57 pages, 39 figures, 21 sources.

The purpose of the work is to improve the design of smart home control applications by introducing gamification elements into its functional architecture.

The object of the study is the processes of management and optimization of activities in the automotive business.

The subject of the study is a process optimization system in the automotive business using the development of a web service.

Summary of the work: The scientific novelty consists in the development of a web service that integrates with automotive business management systems to optimize the processes of inventory management, logistics and interaction with customers. The application of modern web technologies to create a scalable and efficient platform is a significant contribution to the improvement of processes in this field.

The practical significance of the research is revealed in the ability of the web service to reduce costs and increase the efficiency of the work of automobile companies. The results can be implemented for real-world use in automotive companies, improving their competitiveness and customer service.

KEYWORDS: CARS, BUSINESS, AWARENESS ANALYTICS MODULE, APP DEVELOPMENT, CUSTOMER INTERACTION MANAGEMENT

РЕФЕРАТ

Кваліфікаційна робота на здобуття освітнього ступеня бакалавра: 82 сторінки, 36 рисунків, 7 таблиць, 21 джерело.

Мета роботи полягає у розробці веб-сервісу для автомобільного бізнесу, який включає проєктування бази даних, веб-інтерфейсу та серверної частини системи з метою оптимізації внутрішніх процесів, підвищення ефективності обслуговування клієнтів та вдосконалення управління запасами.

Об'єктом дослідження є процеси управління та оптимізації діяльності в автомобільному бізнесі.

Предметом дослідження є система оптимізації процесів у автомобільному бізнесі за допомогою розробки веб-сервісу.

Короткий зміст роботи: Дана робота охоплює процес розробки та впровадження веб-сервісу для автомобільного бізнесу. Вона поділена на три основні розділи, кожен з яких розглядає різні аспекти проекту, включаючи аналіз, проєктування, реалізацію і тестування системи.

Перший розділ зосереджений на аналізі існуючих процесів у автомобільній індустрії та визначенні проблем, які можуть бути оптимізовані за допомогою веб-сервісу.

Другий розділ включає проєктування бази даних та веб-сервісу. Тут розглядається створення структури бази даних, вибір технологій та архітектурних рішень, які забезпечують високу продуктивність, надійність і масштабованість проекту.

Третій розділ описує розробку веб-сайту, включаючи створення користувацького інтерфейсу і серверної частини системи. Детально розглянуто процеси створення інтуїтивно зрозумілого веб-інтерфейсу, що включає HTML і CSS, розробку серверних сценаріїв, які забезпечують функціональність сайту, і взаємодію з базою даних через SQL-запити.

КЛЮЧОВІ СЛОВА: ОПТИМІЗАЦІЯ ПРОЦЕСІВ, PHP, HTML, CSS, MYSQL, БАЗИ ДАНИХ, АВТОБІЗНЕС, АВТОСАЛОН

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	8
ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	12
1.1. Загальний огляд предметної області	12
1.2. Огляд схожих застосунків.....	17
1.3. Проблемні питання.....	26
1.4. Висновки до першого розділу	30
РОЗДІЛ 2. РОЗРОБКА БАЗИ ДАНИХ.....	33
2.1. Проєктування бази даних.....	33
2.2. Проєктування таблиць.....	38
2.3. Проєктування та реалізація алгоритмів роботи системи	41
2.4. Висновки до другого розділу	42
РОЗДІЛ 3. РОЗРОБКА ВЕБ-САЙТУ	44
3.1. Мета розробки веб-сайту	44
3.1.1 Написання SQL запитів до БД відповідно до мети створення веб-сайту.	44
3.2 Опис створення сторінок	46
3.3 Розробка інтерфейсу користувача.....	48
3.4 Розробка сценаріїв серверної частини ІС.....	51
3.4.1 Компонування сценаріїв	51
3.5 Інструкція користувача	54
3.6 Висновки до третього розділу	59
ПЕРЕЛІК ПОСИЛАНЬ	60
ВИСНОВКИ	63
ДОДАТОК А – ЛІСТИНГ БАЗИ ДАНИХ.....	66
ДОДАТОК Б – ЛІСТИНГ ВЕБ-САЙТУ	71

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – База даних

ІС – Інформаційна система

ПЗ – Програмне забезпечення

ER – Entity-Relationship model

CRM – Система управління відносинами з клієнтами

ERP – Система планування ресурсів підприємства

AWS - Amazon Web Services

MySQL – Система управління базами даних

PHP - Hypertext Preprocessor

HTML - Hypertext Markup Language

CSS - Cascading Style Sheets

ВСТУП

Огляд проблеми та актуальності розвитку веб-сервісів у сфері автомобільного бізнесу дозволяє розкрити ключові виклики, з якими зіштовхнулися підприємства цієї галузі та необхідність впровадження інноваційних технологій для їх вирішення. Основні проблеми включають управління інвентарем, оптимізацію процесів продажу та обслуговування авто, ефективне ведення CRM, а також конкуренцію у цифровому просторі. Вирішення цих проблем може стати ключовим фактором успіху в сучасному автомобільному бізнесі [1]. Отже, розвиток веб-сервісів, спрямованих на оптимізацію цих процесів, є надзвичайно актуальним і важливим для підтримки конкурентоспроможності та задоволення потреб споживачів [2].

У розробленому веб-сервісі будуть вирішуватися проблеми управління запасами автомобілів, тобто, розробка системи для ефективного контролю за наявністю автомобілів на складах, процес оптимізації закупівель та збуту автотранспорту для запобігання надлишків або недостачі товару.

Поглиблений аналіз сучасних тенденцій та проблем автомобільного бізнесу розкриває ряд факторів, які потребують оптимізації для підвищення ефективності та конкурентоспроможності. Деякі з найважливіших тенденцій і проблем включають таке:

- споживачі очікують зручних та інтерактивних способів взаємодії з автомобільними компаніями, таких як онлайн-покупка, онлайн-консультації та швидка відповідь на запити;

- з'явлення онлайн-платформ для купівлі та продажу автомобілів, які пропонують широкий вибір і зручний сервіс, створює тиск на традиційні автомобільні дилерські центри;

- управління запасами автомобілів у дилерських центрах може стати складним завданням, особливо у великих організаціях з багатьма локаціями;

- ефективне обслуговування клієнтів включає в себе швидкий доступ до інформації про стан замовлення, сервісні послуги та вирішення проблем шляхом онлайн-звернень;

- рекламні стратегії потребують оптимізації для привертання нових клієнтів та збереження існуючих, враховуючи конкуренцію та зміни у споживчому поведінці;

- ефективне ведення бази даних клієнтів та взаємодія з ними вимагає інтегрованих CRM-систем, які забезпечують персоналізований підхід та збільшують лояльність клієнтів [3].

Аналіз цих тенденцій та проблем дозволяє ідентифікувати області, де веб-сервіси можуть принести найбільшу користь, оптимізуючи процеси і покращуючи взаємодію з клієнтами в автомобільному бізнесі [4-5].

Об'єктом дослідження є процеси управління та оптимізації діяльності в автомобільному бізнесі. Ці процеси включають ряд оперативних та стратегічних дій, які направлені на покращення продуктивності, зниження витрат та підвищення задоволеності клієнтів у автомобільній індустрії. Основні характеристики цих процесів включають логістику, управління запасами, продажі та послуги після продажного обслуговування.

Предметом дослідження є система оптимізації процесів у автомобільному бізнесі за допомогою розробки веб-сервісу. Конкретний фокус роботи направлений на аналіз та розробку програмних рішень, що дозволяють автоматизувати ключові процеси в автомобільному бізнесі, такі як управління запасами, маршрутизація доставок, онлайн-бронювання та сервісні послуги. Розробка такого веб-сервісу має на меті впровадження новітніх технологій та інноваційних підходів для підвищення ефективності та зниження операційних витрат.

Методи дослідження:

- аналіз літератури для з'ясування сучасних тенденцій в автомобільному бізнесі та виявлення ключових проблем, які можна оптимізувати за допомогою веб-сервісів;

- методи системного аналізу для вивчення та опису процесів в автомобільному бізнесі, які можуть бути покращені через впровадження ІТ-рішень;

- розробка програмного забезпечення для створення прототипу веб-сервісу, що включає дизайн, кодування та тестування. Даний програмний продукт буде

розроблюватися за допомогою технологій HTML, CSS, PHP, MySQL, JavaScript. За платформу буде використано Open Server;

- емпіричне тестування, що дозволяє аналізувати ефективність розробленого веб-сервісу на реальних даних від автомобільних компаній.

Наукова новизна полягає у розробці веб-сервісу, який інтегрується з системами управління автомобільного бізнесу для оптимізації процесів управління запасами, логістики та взаємодії з клієнтами. Застосування сучасних веб-технологій для створення масштабованої та ефективною платформи є значним внеском у вдосконалення процесів у цій галузі.

Практична значущість дослідження виявляється в здатності веб-сервісу знижувати витрати та підвищувати ефективність роботи автомобільних компаній. Результати можуть бути впроваджені для реального використання в автомобільних компаніях, покращуючи їхню конкурентоспроможність і клієнтське обслуговування.

Результати дослідження мають як практичне, так і теоретико-методичне значення. Зокрема, були сформовані нові методичні підходи до аналізу та оптимізації процесів управління в автомобільній індустрії, що можуть бути застосовані не тільки в автомобільній галузі, а й у інших секторах для підвищення ефективності бізнесу.

Структура роботи. Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Загальний огляд предметної області

Автомобільний бізнес включає виробництво, продаж, та післяпродажне обслуговування автомобілів. Це широка галузь, що охоплює виробників автомобілів, дилерські центри, ремонтні майстерні, а також постачальників автомобільних компонентів.

До основних процесів в автомобільному бізнесі:

- виробництво – охоплює процеси планування, проєктування, виготовлення та збірки автомобілів;
- логістика та управління запасами – включає управління запасами автомобілів та запчастин, а також логістику поставок;
- продажі та маркетинг – стратегії просування продуктів і послуг, взаємодія з клієнтами, управління продажами в дилерських центрах;
- сервісне обслуговування - управління післяпродажними послугами, такими як технічне обслуговування та ремонт автомобілів.

Виклики у автомобільному бізнесі:

- ефективність логістики – забезпечення своєчасної доставки автомобілів та компонентів при мінімальних витратах;
- задоволення попиту клієнтів – адаптація до змінних запитів ринку та впровадження персоналізованих рішень.
- управління запасами - оптимізація рівня запасів для забезпечення ефективного виробництва та мінімізації зберігання.
- підвищення якості обслуговування – впровадження новітніх технологій для покращення якості послуг.

Технологічні рішення:

Розробка веб-сервісу має на меті вирішення цих викликів шляхом:

- автоматизація процесів. Використання програмного забезпечення для автоматизації рутинних процесів, таких як управління запасами та облік;

- більша інтеграція. Забезпечення легкої інтеграції з іншими системами, такими як CRM і ERP;

- аналітика даних. Використання великих даних і аналітики для прийняття обґрунтованих рішень на основі точних даних.

Технічні рішення розробки веб-сервісу для автомобільного бізнесу включають ряд важливих вирішень, які впливають на якість, продуктивність та безпеку системи. Наведемо деякі механізми розробки:

1) Вибір технологій:

- мова програмування. Найпоширеніші мови для веб-розробки включають JavaScript (зокрема з використанням фреймворків, таких як React, Angular або Vue.js), Python (з Django або Flask), а також Java або C#;

- база даних. Важливо вибрати базу даних, яка відповідає потребам проекту. Для автомобільного бізнесу можуть бути використані реляційні бази даних, такі як MySQL або PostgreSQL, або NoSQL бази даних, такі як MongoDB;

- хмарні послуги. Використання хмарних платформ, таких як Amazon Web Services (AWS), Microsoft Azure або Google Cloud Platform (GCP), може забезпечити масштабованість та надійність системи.

2) Архітектурні рішення:

- мікросервісна архітектура. Розбиття системи на невеликі незалежні мікросервіси дозволяє легко масштабувати та розвивати окремі компоненти;

- RESTful або GraphQL API. Використання веб-сервісів з архітектурою REST або GraphQL дозволяє забезпечити ефективну взаємодію з клієнтами та іншими системами.

3) Інструменти розробки:

- розробка фронтенду. Використання редакторів коду, таких як Visual Studio Code, або IDE, таких як WebStorm чи PyCharm. Для верстки можуть використовуватись CSS-препроцесори (наприклад, Sass або Less) та бібліотеки для інтерфейсу, такі як Bootstrap або Material Design;

- розробка бекенду. Вибір середовища розробки та налагодження коду. Для Python можна використовувати PyCharm або VS Code, для JavaScript - Visual Studio Code або WebStorm, а для Java - IntelliJ IDEA або Eclipse;

- контроль версій. Використання систем контролю версій, таких як Git, для керування розробкою та забезпечення спільної роботи команди.

Проектування та розробка функціональності веб-сервісу для оптимізації процесів в автомобільній галузі може включати різноманітні модулі та функції, спрямовані на полегшення та автоматизацію різних аспектів бізнесу. Деякі з них включають:

1) Модуль управління інвентарем:

- додавання нових автомобілів до інвентарю;
- відстеження статусу та характеристик кожного автомобіля;
- управління доступністю авто для продажу та резервування.

2) Модуль онлайн-продажів:

- можливість клієнтам переглядати доступні автомобілі та їх характеристики;
- онлайн-замовлення автомобіля з можливістю вибору опцій та конфігурацій;
- інтеграція з платіжними системами для оплати онлайн.

3) Модуль управління обслуговуванням:

- запис на сервісне обслуговування або тест-драйв;
- відстеження статусу замовлень на сервіс та їх виконання;
- сповіщення клієнтів про поточний стан їх автомобілів та рекомендації щодо обслуговування.

4) Модуль CRM:

- зберігання даних клієнтів та їх історії взаємодії з компанією;
- аналіз поведінки клієнтів для персоналізації обслуговування та маркетингових кампаній;
- система нагадувань та сповіщень для підтримки зв'язку з клієнтами.

5) Модуль аналітики та звітності:

- збір та аналіз даних про продажі, сервісне обслуговування та взаємодію з клієнтами;

- створення звітів та аналітичних звітів для оцінки ефективності роботи та прийняття управлінських рішень.

б) Модуль адміністрування:

- керування доступом та правами користувачів системи;
- моніторинг та управління даними та інфраструктурою системи.

Ці модулі можуть бути інтегровані в одну цілісну платформу, яка забезпечує повний цикл обслуговування клієнтів та управління бізнесом в автомобільній галузі. Кожен з них спрямований на полегшення роботи персоналу, покращення взаємодії з клієнтами та підвищення ефективності процесів.

Тестування та валідація розробленого веб-сервісу для ефективності його функціонування та відповідності вимогам є критичним етапом перед впровадженням системи в експлуатацію. Основні кроки тестування та валідації включають такі:

1) Функціональне тестування:

- перевірка кожного модуля та функції системи на відповідність вимогам та очікуваному результату;
- виконання різних сценаріїв взаємодії з системою, щоб переконатися, що вона працює правильно у всіх умовах.

2) Тестування продуктивності:

- вимірювання швидкодії системи під навантаженням, щоб переконатися, що вона здатна працювати ефективно при реальному використанні;
- проведення тестів на великому обсязі даних або користувачів, щоб визначити можливі межі масштабування.

3) Тестування безпеки:

- перевірка на вразливості та потенційні загрози безпеці;
- виконання тестів на проникнення для виявлення слабких місць у системі та їх подальше виправлення.

4) Валідація відповідності вимогам:

- перевірка, що розроблений веб-сервіс відповідає всім вимогам, визначеним на етапі аналізу та проектування;

- порівняння фактичного функціонування системи з очікуваними результатами та вчасне виправлення будь-яких відхилень.

5) Тестування відновлення та стійкості:

- перевірка системи на відновлення після збоїв або відмов;
- тестування резервного копіювання даних та механізмів відновлення системи після критичних ситуацій.

6) Тестування сумісності:

- перевірка сумісності веб-сервісу з різними веб-браузерами, операційними системами та пристроями.

- перевірка реакції системи на різні роздільні здатності екранів та типи мережі.

Після успішного завершення цих етапів тестування та валідації можна переходити до впровадження веб-сервісу в реальному середовищі, з впевненістю, що він відповідає всім вимогам та очікуванням користувачів.

Схематичне подання розробки та аналізу веб-сервісу для оптимізації процесів в автомобільному бізнесі наведений на рис. 1.1.

На етапі введення визначаються основні проблеми та потреби, що виникають у автомобільному бізнесі, які можуть бути вирішені за допомогою веб-сервісу. Проводиться аналіз існуючих практик та визначення ключових вимог до нового сервісу.

На етапі аналізу проблем глибше вивчається природа проблем, які були виявлені у попередньому етапі. Проводиться аналіз процесів управління автомобільним бізнесом, виявлення та уточнення проблемних аспектів.

Після визначення основних проблем формулюються технічні вимоги до веб-сервісу. Розглядаються можливі технології, архітектура системи, інтеграційні можливості та інші технічні аспекти розробки.

На етапі проектування та розробки проводиться проектування архітектури веб-сервісу, розробка інтерфейсу користувача, створення бази даних та реалізація функціональності веб-сервісу згідно з визначеними вимогами.

Після завершення розробки проводяться тестування, що включає модульне тестування окремих компонентів, інтеграційне тестування системи в цілому та приймальне тестування замовником. Валідація проводиться для перевірки відповідності розробленого сервісу вимогам та очікуванням.

У заключному етапі проводиться аналіз результатів розробки та тестування веб-сервісу. Формулюються висновки щодо його ефективності, робляться рекомендації щодо можливих покращень або додаткового розвитку.



Рис. 1.1 Схема розробки та аналізу веб-сервісу для оптимізації процесів в автомобільному бізнесі

1.2. Огляд схожих застосунків

До схожих застосунків можна віднести:

1) Salesforce Automotive Cloud [6] - це CRM-система, спеціалізована на автомобільній індустрії, яка дозволяє компаніям управляти відносинами з клієнтами, відстежувати продажі, сервіс та маркетингові кампанії. Система забезпечує інтеграцію з іншими платформами і аналітичні можливості.

2) Microsoft Dynamics 365 for Automotive [7] – цей продукт включає рішення для оптимізації виробничих ланцюгів, управління взаємодіями з клієнтами та покращення процесів обслуговування. З його допомогою можна автоматизувати

багато аспектів діяльності автомобільних компаній. Також можна значно підвищити ефективність управління всіма аспектами автомобільного бізнесу. Microsoft Dynamics 365 for Automotive надає широкий спектр інструментів для аналізу даних, управління ресурсами, а також для планування і виконання виробничих процесів, що дозволяє автовиробникам, дилерам та постачальникам комплектуючих краще взаємодіяти між собою і оптимізувати свою діяльність на кожному етапі виробничого ланцюга.

Розглянемо детальніше програму Salesforce Automotive Cloud.

Salesforce Automotive Cloud є інноваційним рішенням, спеціально розробленим для автомобільної індустрії, яке використовує можливості хмарної платформи Salesforce для вдосконалення взаємодій між автовиробниками, дилерами, постачальниками та кінцевими споживачами. Воно забезпечує компанії необхідними інструментами для аналізу поведінки клієнтів, їхніх переваг та потреб, дозволяючи створювати більш персоналізоване та ефективне сервісне обслуговування.

Головна сторінка Salesforce Automotive Cloud наведена на рис. 1.2.

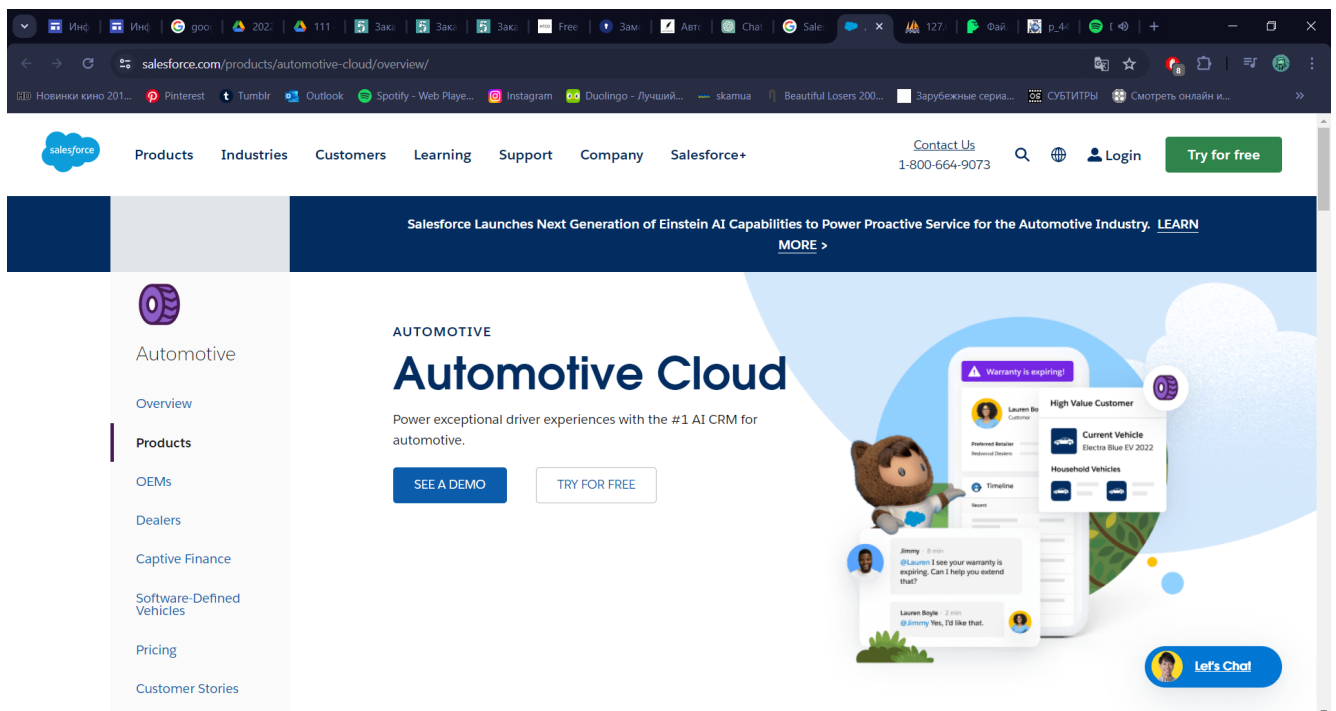


Рисунок 1.2 Головна сторінка Salesforce Automotive Cloud

До основних можливостей Salesforce Automotive Cloud:

- повне розуміння клієнта. Використання великих даних і штучного інтелекту для збору та аналізу інформації про клієнтів, що дозволяє створювати більш точні та ефективні маркетингові кампанії;

- персоналізація взаємодії. Надання можливості дилерам налаштовувати свої пропозиції і спілкування в залежності від індивідуальних потреб та переваг кожного клієнта;

- управління взаєминами з клієнтами (CRM). Інтеграція з іншими системами Salesforce для забезпечення єдиної точки доступу до всіх клієнтських даних;

- оптимізація ланцюжка постачань. Використання аналітики для прогнозування та оптимізації ланцюжків постачання, зниження витрат на логістику та покращення управління запасами;

- мобільний доступ і інтеграція. Підтримка мобільних пристроїв та інтеграція з іншими бізнес-додатками для забезпечення швидкого доступу до інформації в будь-який час і з будь-якого місця;

- інтеграція з Інтернетом речей (IoT). Використання даних з IoT-пристроїв для відстеження стану автомобілів у реальному часі, що дозволяє проводити превентивне обслуговування та забезпечувати кращий досвід користувачів;

- штучний інтелект (AI) для персоналізації. Salesforce Einstein AI допомагає в аналізі поведінки клієнтів та прогнозуванні їхніх потреб, що дозволяє компаніям налаштовувати маркетингові кампанії та рекомендації продуктів;

- управління життєвим циклом клієнтів. Інтеграція взаємодії з клієнтами у всіх точках дотику, від придбання до обслуговування та лояльності, що допомагає створити безперервний досвід для клієнтів;

- робочі процеси та автоматизація. Можливість автоматизувати складні бізнес-процеси з метою зниження витрат на ручну працю та помилок, пов'язаних з людським фактором.

До переваг Salesforce Automotive Cloud відносять:

- висока адаптивність. Рішення легко адаптується до конкретних потреб бізнесу, завдяки широким можливостям кастомізації;
- підвищення продуктивності. Автоматизація багатьох процесів дозволяє зменшити ручну роботу та покращити продуктивність;
- покращення задоволеності клієнтів. Більш глибоке розуміння потреб клієнтів і здатність надавати високоперсоналізовані послуги покращує загальну задоволеність клієнтів;
- масштабування. Рішення може масштабуватися зі зростанням вашого бізнесу, забезпечуючи підтримку від стартапів до великих міжнародних корпорацій;
- гнучкість та масштабованість. Платформа Salesforce дозволяє легко масштабувати рішення в залежності від змін у бізнесі, забезпечуючи також гнучкість для адаптації до нових викликів та можливостей;
- залучення клієнтів на всіх етапах. Оптимізація взаємодії з клієнтами на кожному етапі їхнього досвіду, від знайомства з брендом до післяпродажного обслуговування;
- операційна ефективність. Інтеграція та автоматизація бізнес-процесів сприяють зниженню операційних витрат і підвищенню ефективності операцій.

Недоліки:

- висока вартість. Як і багато інших рішень Salesforce, Automotive Cloud може бути дорогим у впровадженні та підтримці, особливо для малих та середніх компаній;
- складність інтеграції. Незважаючи на широкі можливості інтеграції, іноді з'єднання з існуючими системами може бути складним та потребувати додаткових ресурсів;
- комплексність. Велика кількість функцій та налаштувань може зробити систему складною для освоєння, особливо для користувачів, які не мають технічного досвіду;

- залежність від вендора. Залежність від одного постачальника програмного забезпечення може бути ризикованою, якщо з якихось причин потрібно буде змінити платформу;

- комплексність управління змінами. Впровадження Salesforce може вимагати значних зусиль для управління змінами в організації, особливо у великих компаніях з усталеними процесами;

- необхідність спеціалізованих знань. Для налаштування та управління системою потрібні висококваліфіковані спеціалісти, які розуміють як Salesforce, так і специфіку автомобільної індустрії;

- потреба у постійному оновленні. Система потребує регулярних оновлень для забезпечення ефективності та безпеки, що може призвести до додаткових витрат та необхідності періодичного перенавчання співробітників;

- захист даних та приватність. З огляду на великі обсяги збору та аналізу даних, компанії мають забезпечити дотримання нормативних вимог щодо захисту даних та приватності клієнтів.

2) Microsoft Dynamics 365 for Automotive – це галузеве рішення в межах ширшої платформи Microsoft Dynamics 365, спеціально адаптоване для потреб автомобільної індустрії. Це комплексне рішення допомагає автовиробникам, дистриб'юторам та постачальникам запчастин управляти виробничими процесами, взаємодіями з клієнтами та цілісністю постачальницьких ланцюгів, що значно покращує ефективність і оптимізує операції. Головна сторінка Microsoft Dynamics 365 for Automotive наведена на рис. 1.3 [8].

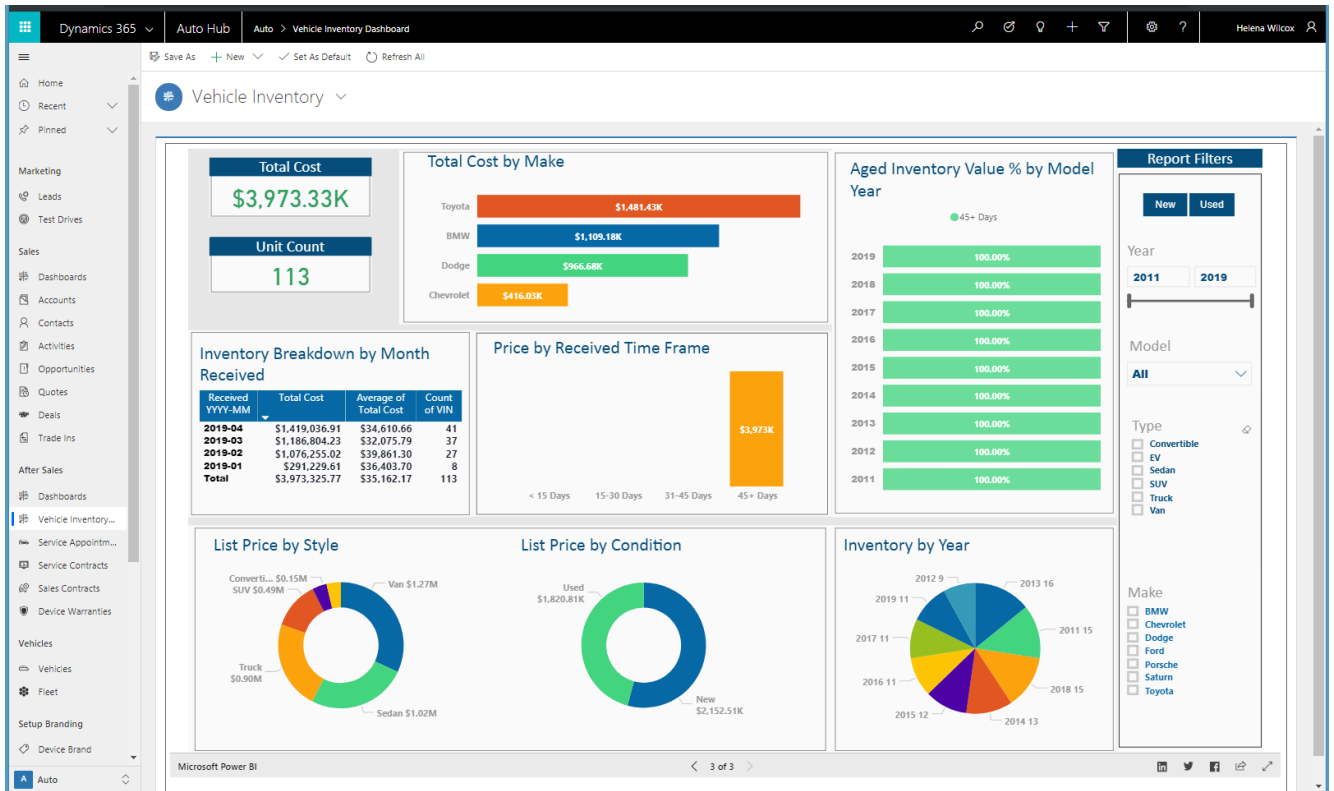


Рис. 1.3 Головна сторінка Microsoft Dynamics 365 for Automotive

До основних характеристик і можливостей Microsoft Dynamics 365 for Automotive відносять:

- управління взаємодіями з клієнтами (CRM). Покращення обслуговування клієнтів через розширені інструменти CRM, що дозволяють збирати, аналізувати та використовувати дані для налагодження більш ефективних відносин із клієнтами;
- інтеграція з виробництвом і ланцюгами поставок. Інструменти для управління виробничими процесами та ланцюгами поставок, що включають планування, прогнозування та оптимізацію ресурсів для забезпечення безперервності виробництва;
- аналітика та інсайти. Використання передових аналітичних інструментів для обробки великих даних, що дозволяє здійснювати глибокий аналіз ринкових тенденцій та поведінки споживачів, що покращує стратегічне планування та прийняття рішень;

- мобільність та доступність. Мобільні рішення дозволяють співробітникам доступатися до важливих даних і систем з будь-якого місця, що значно підвищує гнучкість робочих процесів і оперативність відповідей;

- безпека та відповідність стандартам. Сильні функції безпеки і здатність дотримуватися міжнародних і регіональних нормативів допомагають забезпечити захист даних та відповідність бізнесу до вимог законодавства;

- інтеграція з іншими платформами Microsoft. Тісна інтеграція з іншими продуктами Microsoft, як-от Office 365, Azure та інші додатки Dynamics 365, створює єдине інформаційне середовище, що покращує співпрацю та ефективність.

До переваг Microsoft Dynamics 365 for Automotive відносять:

- оптимізація виробничих процесів. Допомагає знизити витрати і збільшити продуктивність завдяки автоматизації та оптимізації робочих процесів;

- підвищення задоволеності клієнтів. Покращення якості обслуговування та індивідуалізація підходу до кожного клієнта збільшують лояльність і задоволеність;

- здатність швидко реагувати на зміни на ринку. Гнучкість та адаптивність системи дозволяють швидко адаптуватися до ринкових умов та змін у попиті;

- централізація даних. Єдина платформа для управління всіма бізнес-процесами дозволяє мати централізований доступ до інформації, забезпечуючи точність даних і ефективність у прийнятті рішень;

- автоматизація процесів. Рутинні завдання, такі як замовлення запчастин, управління інвентарем та ведення обліку, можуть бути повністю автоматизовані, що знижує ризик людської помилки і звільняє ресурси для більш стратегічних задач;

- гнучкість. Платформа легко адаптується під специфічні вимоги та процеси кожної окремої компанії, що дозволяє користувачам модифікувати функціонал відповідно до змін у бізнес-моделі або регулятивних вимог;

- інтегрованість з IoT (інтернет речей). Інтеграція з IoT технологіями дозволяє збирати дані в реальному часі з виробничих ліній або прямо з транспортних засобів, що підвищує можливості моніторингу і управління якістю;

- підтримка прийняття рішень. Розширені аналітичні інструменти і штучний інтелект надають глибокі аналітичні дані та прогнозування, що допомагає керівництву ефективніше ухвалювати рішення.

До недоліків Microsoft Dynamics 365 for Automotive відносять:

- висока вартість впровадження. Як і більшість корпоративних рішень, Microsoft Dynamics 365 може вимагати значних інвестицій на початковому етапі для впровадження та налаштування системи під конкретні потреби підприємства;

- комплексність. Велика функціональність системи може виявитись складною для засвоєння співробітниками, особливо без належного навчання та підтримки;

- залежність від одного постачальника. Залежність від одного постачальника програмного забезпечення може створювати ризики, пов'язані з обмеженнями на інновації та можливими змінами у політиці постачальника;

- потреба в постійному оновленні. Щоб залишатися ефективною і безпечною, система потребує регулярних оновлень, що може призводити до додаткових витрат і необхідності періодичного перенавчання співробітників;

- можливі проблеми з інтеграцією. Хоча Dynamics 365 інтегрується з багатьма іншими системами та платформами, іноді можуть виникати проблеми сумісності з усталеним програмним забезпеченням або існуючими бізнес-процесами.

Проаналізувавши дві системи Sales Automotive Cloud і Microsoft Dynamics 365 for Automotive можна зробити таке порівняння (табл. 1.1).

Таблиця 1.1

Порівняння Sales Automotive Cloud і Microsoft Dynamics 365 for Automotive

Параметр	Salesforce Automotive Cloud	Microsoft Dynamics 365 for Automotive
Сильні сторони	<p>1) Salesforce Automotive Cloud пропонує широкий спектр CRM-інструментів, спеціально призначених для автомобільної індустрії, включаючи управління клієнтами, продажами та маркетингом.</p> <p>2) Платформа Salesforce дозволяє легко доступатися до даних з будь-якого пристрою, що робить її зручною для користувачів у рухливій сфері автомобільного бізнесу.</p> <p>3) Salesforce Automotive Cloud може бути швидко налаштований та впроваджений з урахуванням потреб конкретного автомобільного підприємства, що забезпечує гнучкість у роботі з різними бізнес-завданнями.</p>	<p>1) Microsoft Dynamics 365 for Automotive легко інтегрується з іншими продуктами Microsoft, такими як Microsoft Office, що забезпечує єдиний екосистему для роботи з даними.</p> <p>2) Платформа надає потужні засоби аналітики та звітності, що дозволяє користувачам ефективно аналізувати дані та приймати обґрунтовані рішення.</p> <p>3) Microsoft Dynamics 365 дозволяє створювати персоналізовані відносини з клієнтами та ефективно управляти ними, що є ключовим для успіху в автомобільному бізнесі.</p>
Обмеження	<p>1) Salesforce Automotive Cloud може бути високою вартістю для невеликих автомобільних компаній або підприємств з обмеженим бюджетом.</p> <p>2) Для ефективного використання Salesforce Automotive Cloud може знадобитися певний час на навчання персоналу з використання платформи.</p>	<p>1) Деякі користувачі відзначають, що інтеграція Microsoft Dynamics 365 for Automotive з іншими системами може бути складною та вимагати додаткових зусиль.</p> <p>2) Використання платформи може вимагати наявності висококваліфікованих ІТ-спеціалістів для її ефективного налаштування та управління.</p>

Програмні системи, такі як Salesforce Automotive Cloud і Microsoft Dynamics 365 for Automotive, є ключовими гравцями в автомобільній індустрії, надаючи

комплексні рішення для управління клієнтами, продажами, сервісом та іншими аспектами бізнесу. У проведеному дослідженні ці системи використовуються як точки посилення для аналізу сучасних тенденцій та проблем в автомобільній галузі, а також для ідентифікації областей, які потребують оптимізації та вдосконалення через веб-сервіси.

Зокрема, функції, які вирішують проблеми управління клієнтами (CRM), управління запасами та логістичні процеси, онлайн-продажі та обслуговування післяпродажного сервісу, які реалізовані в цих системах, можуть бути використані як основа для розробки веб-сервісу для оптимізації автомобільного бізнесу.

Аналіз систем, таких як Salesforce Automotive Cloud і Microsoft Dynamics 365 for Automotive, допомагає зрозуміти потреби та вимоги ринку, а також визначити успішні практики та інноваційні підходи до управління автомобільним бізнесом. Це дозволяє врахувати найбільш актуальні та ефективні рішення при розробці нового веб-сервісу, а також виявити можливості для вдосконалення та унікальні функції, які можуть відрізнити мій проєкт від існуючих систем.

1.3. Проблемні питання

В розробці веб-сервісів для автомобільного бізнесу важливе постановка проблемних питань, яка визначає напрямок дослідження та розробки. Правильне визначення проблем допомагає розробникам зосередитися на ключових аспектах оптимізації та покращенні процесів у цій галузі. Ретельний аналіз проблем дозволяє ідентифікувати потреби бізнесу та кінцевих користувачів, а також визначити можливості для впровадження інноваційних технологій. Тому у цьому розділі ми розглянемо ключові проблеми, які можуть бути вирішені за допомогою веб-сервісів у сфері автомобільного бізнесу.

Проблемні питання складається від постановки задач до розробки архітектури і тестування рішення. Етапи проблемних питань поділяються на наступні:

1. Визначення проблемних питань і потреб включає в себе детальне дослідження існуючих процесів в автомобільному бізнесі, виявлення неефективностей і больових точок, які можна вирішити за допомогою веб-сервісу.

До типових проблем відносять:

- недостатнє управління інвентарем. Неефективне управління запасами може призводити до надлишків або дефіциту запчастин;
- слабка координація між відділами. Необхідність у покращенні комунікації між виробництвом, логістикою та продажами;
- моніторинг стану виробничих потужностей. Важливість відстеження використання виробничих ліній та планування обслуговування;
- клієнтське обслуговування. Потреба в підвищенні якості обслуговування клієнтів і персоналізації пропозицій.

Інтеграція з системами управління ресурсами підприємства (ERP) та системами управління відносинами з клієнтами (CRM) відіграє ключову роль у вирішенні проблем управління запасами в автомобільному бізнесі. Це обґрунтовується наступним:

- інтеграція з системою ERP дозволяє автоматизувати процес управління запасами, враховуючи різноманітні фактори, такі як попит клієнтів, прогнозування попиту, історія продажів тощо. Це допомагає уникнути надлишкових запасів, зменшує витрати на утримання запасів та оптимізує поповнення запасів відповідно до реальних потреб;
- інтеграція з системою CRM дозволяє автоматизувати процес обслуговування клієнтів, забезпечуючи швидкий доступ до інформації про замовлення, історію взаємодії з клієнтом, його уподобання та інші важливі дані. Це допомагає створити персоналізований підхід до кожного клієнта, підвищує задоволеність клієнтів та сприяє побудові довгострокових відносин;
- інтеграція з ERP і CRM забезпечує доступ до об'єктивних та достовірних даних про запаси, продажі, клієнтів та інші аспекти бізнесу. Це допомагає керівництву приймати обґрунтовані рішення щодо розвитку бізнесу, стратегій продажу та маркетингу, а також планування виробництва і закупівель.

Інтеграція з цими системами дозволяє вирішувати проблеми, пов'язані з неефективним управлінням запасами, недостатньою увагою до потреб клієнтів та недостатньою обґрунтованістю прийнятих рішень.

2. Розробка вимог до веб-сервісу. На основі виявлених потреб формулюються функціональні та нефункціональні вимоги до веб-сервісу.

До функціональних відносять:

- інтеграцію з існуючими ERP, CRM;
- модулі для управління запасами, розклад виробництва, відстеження замовлення, дашборди для моніторингу продажів, тощо.

До нефункціональних вимог відносять:

- вимоги до надійності;
- масштабованості;
- безпеки;
- зручності інтерфейсу користувача;
- швидкості реакції системи.

3. Технологічне проектування та розробка включає вибір технологій, проектування архітектури веб-сервісу, розробку інтерфейсу користувача та серверної частини.

- технологічний стек. Вибір мов програмування (наприклад, Java, Python), фреймворків (наприклад, Spring Boot, Django), баз даних (MySQL, MongoDB), фронтенд-технологій (React, Angular);

- архітектура. Мікросервісна архітектура для забезпечення масштабованості і гнучкості, контейнеризація (Docker, Kubernetes) для полегшення розгортання та управління.

4. Тестування і деплоймент. Важливий етап забезпечення якості продукту і його підготовки до експлуатації. Включає в себе:

- тестування. Автоматизація тестування (unit tests, integration tests), використання інструментів для навантажувального тестування (JMeter, LoadRunner);

- деплоймент. Використання CI/CD пайплайнів для автоматизації випуску нових версій, моніторинг стану веб-сервісу в реальному часі.

5. Обслуговування і масштабування. Після запуску веб-сервісу необхідно забезпечити його стабільну роботу і готовність до масштабування, включивши в це:

- підтримку. Налаштування процесів технічної підтримки, оновлення програмного забезпечення, виправлення помилок;

- масштабування. Аналіз завантаження системи і за необхідності додавання ресурсів або оптимізація існуючих для підтримки збільшення кількості користувачів.

Обґрунтуємо вибір технологій для розробки

- HTML використовується для створення структури та основного контенту веб-сторінок. В контексті автомобільного бізнесу, HTML дозволяє створювати ефективний та доступний інтерфейс для користувачів, де можна зручно відображати інформацію про товари, послуги та інші аспекти автомобільного бізнесу;

- CSS використовується для стилізації і дизайну веб-сторінок, що дозволяє створювати привабливий та користувацьки зручний інтерфейс. У контексті автомобільного бізнесу, CSS допомагає покращити візуальний вигляд сайту, зробити його більш привабливим для користувачів, а також підвищити його конверсію;

- PHP є однією з найпопулярніших мов програмування для розробки веб-серверних додатків. У контексті автомобільного бізнесу, PHP може використовуватися для реалізації бізнес-логіки, обробки форм, взаємодії з базою даних та інших серверних операцій, що дозволяє створювати функціональні та динамічні веб-сервіси;

- JavaScript є мовою програмування, яка використовується для створення інтерактивних елементів та функціональності на веб-сторінках. У контексті автомобільного бізнесу, JavaScript може використовуватися для реалізації клієнтської валідації форм, анімації, взаємодії з користувачем та інших функцій, що підвищують зручність використання веб-сервісу;

- MySQL є однією з найпоширеніших відкритих реляційних баз даних. У контексті автомобільного бізнесу, MySQL може використовуватися для зберігання та управління даними про товари, клієнтів, замовлення, операції та іншу інформацію, що дозволяє забезпечити ефективне управління даними та їх швидкий доступ.

Обрані технології забезпечують широкий функціональний спектр та дозволяють вирішити конкретні проблеми, пов'язані з розробкою веб-сервісу для автомобільного бізнесу, забезпечуючи ефективну роботу, зручний інтерфейс та надійність системи.

1.4. Висновки до першого розділу

Підсумовуючи перший розділ, який був присвячений аналізу та розробці веб-сервісу для оптимізації процесів в автомобільному бізнесі, можна зробити наступні важливі висновки:

1. Актуальність та необхідність оптимізації

Аналіз існуючих процесів в автомобільному бізнесі виявив кілька ключових проблем, таких як неефективне управління запасами, слабка координація між відділами, і неоптимальне обслуговування клієнтів. Ці проблеми підкреслюють необхідність розробки веб-сервісу, який би оптимізував ці процеси, забезпечуючи кращу інтеграцію даних та сприяючи більш ефективному прийняттю рішень.

2. Формулювання вимог

Для розробки ефективного веб-сервісу було важливо точно формулювати як функціональні, так і нефункціональні вимоги. Функціональні вимоги забезпечують необхідний функціонал, в той час як нефункціональні вимоги, такі як безпека, надійність, та зручність користування, гарантують, що веб-сервіс буде стабільним і придатним для користувача.

3. Вибір технологій та розробка

Вибір відповідних технологій та архітектурних рішень є критично важливим для успішної реалізації проекту. Мікросервісна архітектура, контейнеризація, і

використання сучасних інструментів розробки дозволяють створити масштабований та легко підтримуваний продукт.

4. Тестування та впровадження

Ретельне тестування є ключем до забезпечення якості веб-сервісу. Автоматизоване тестування, навантажувальне тестування, та регулярні оновлення допомагають уникнути помилок у роботі сервісу та забезпечують його стабільну експлуатацію.

5. Операційне обслуговування та масштабування

Останнім кроком є забезпечення надійності та гнучкості веб-сервісу після його запуску, що включає технічну підтримку, моніторинг системи, та готовність до масштабування при зростанні навантаження.

6. Залучення зацікавлених сторін

Важливість врахування думок і потреб усіх зацікавлених сторін, включаючи менеджмент, співробітників, які безпосередньо використовуватимуть систему, та клієнтів, є ключовим. Включення їхніх побажань і вимог може значно покращити прийняття та ефективність системи.

7. Стійкість до змін

Розробка веб-сервісу повинна передбачати можливість легкого адаптування до змін у бізнес-процесах та зовнішньому середовищі. Це включає можливість швидко додавати нові функції, масштабувати існуючі та оптимізувати ресурси відповідно до змінюваних потреб бізнесу.

8. Безпека даних

Оскільки веб-сервіс буде обробляти великі обсяги чутливої інформації, забезпечення безпеки даних є абсолютно критичним. Це включає розробку суворих протоколів захисту даних, регулярні перевірки безпеки та впровадження сучасних методів шифрування та аутентифікації.

9. Впровадження та навчання

Для успішного впровадження веб-сервісу необхідно не тільки технічно вдосконалити систему, а й забезпечити якісне навчання співробітників. Організація семінарів та тренінгів, підготовка навчальних матеріалів і постійна підтримка

користувачів сприятимуть гладкій інтеграції системи в повсякденні операції компанії.

10. Моніторинг та оптимізація

Після запуску веб-сервісу необхідно проводити постійний моніторинг його ефективності, аналізувати отримані дані та оптимізувати роботу системи. Це дозволить виявити непередбачені проблеми, швидко реагувати на зміни та забезпечити тривалу цінність від інвестицій у технології.

Підсумовуючи, розробка веб-сервісу для оптимізації процесів у автомобільному бізнесі є складним, але дуже важливим завданням, яке може суттєво підвищити ефективність роботи підприємства, підвищити задоволеність клієнтів і в кінцевому підсумку збільшити прибуток.

Загалом, розробка веб-сервісу для автомобільного бізнесу повинна базуватися на глибокому розумінні потреб галузі та бути спрямована на створення інструменту, який не тільки вирішує поточні проблеми, а й має потенціал для майбутньої адаптації та розвитку. Успіх такого проекту вимагає злагодженої роботи всієї команди, від розробників до кінцевих користувачів, і відповідної підтримки на всіх рівнях організації.

РОЗДІЛ 2. РОЗРОБКА БАЗИ ДАНИХ

2.1. Проектування бази даних

Таблиці у базі даних називаються сутностями, а їхні поля – атрибутами. В результаті зв'язування цих таблиць будується модель «сутність-зв'язок», яку ще називають ER-діаграмою [9].

Модель «сутність-зв'язок» представляє собою універсальну схему для моделювання даних, що дозволяє детально описати концептуальні структури за допомогою стандартизованих блоків. Ця модель виступає як мета-модель, яка слугує для опису інших моделей даних [10]. Серед різноманітних підходів до представлення інформації, ER-модель вирізняється своєю універсальністю та незалежністю від конкретних технологій реалізації, роблячи її одним із найбільш гнучких інструментів для моделювання даних [11]. Ця модель має важливу властивість: з неї можна генерувати всі основні типи даних, такі як ієрархічні, мережеві, реляційні та об'єктні моделі, що підтверджує її універсальність як основу для будь-яких систем баз даних.

Сама ER-модель представляє процес моделювання як спосіб визначення об'єктів у певній предметній області та їх взаємодії [12]. Вона не описує процес безпосередньо, а скоріше забезпечує візуальне представлення структури даних. Дані в моделі організовані у формі сутностей, які пов'язані між собою через зв'язки. Ці зв'язки висловлюють логічні залежності, наприклад, розподіл будівлі на квартири або відношення між працівником і відділом. Кожна сутність характеризується атрибутами, які деталізують її особливості [13].

ER-діаграми є основним інструментом для візуалізації цих сутностей, їх атрибутів, та зв'язків між ними. Ці діаграми дають можливість легко оглянути та аналізувати складні структури даних, що значно спрощує процес проектування баз даних та систем обробки інформації [14].

Коли ми обговорюємо поняття «сутність» в контексті моделювання даних, ми маємо на увазі абстрактний елемент або компонент, який представляє певний об'єкт,

процес чи явище з реального світу, виділений для аналізу і управління в інформаційних системах [15]. Сутність може бути розглянута як узагальнене поняття, що об'єднує групу подібних об'єктів або явищ під одним заголовком. Розрізнення між «типом сутності» та «екземпляром сутності» дозволяє чітко визначити загальні класифікації та конкретні випадки [16]. Наприклад, якщо «МІСТО» є типом сутності, то конкретні міста, такі як Київ чи Львів, є її екземплярами. Приклад з'єднання сутностей наведено на рис.2.1 [17].



Рис. 2.1 Дві сутності з'єднані зв'язком

Сутності класифікують на кілька видів залежно від їх ролі та залежностей у моделі:

- стрижневі (сильні) сутності – це незалежні сутності, які існують самостійно без необхідності опори на інші сутності. Вони формують основу інформаційної моделі і не є ні асоціаціями, ні характеристиками;

- асоціативні сутності – втілюють в собі відносини «багато до багатьох» між двома або більше сутностями. Вони є повністю самостійними та служать для з'єднання інших сутностей, наприклад, асоціативна сутність «ШЛЮБ» з'єднує «ЧОЛОВІКА» та «ЖІНКУ»;

- характеристичні (слабкі) сутності – вони залежні від інших, більш міцних сутностей та не можуть існувати окремо. Ці сутності використовуються для деталізації або уточнення сильніших сутностей. Наприклад, «Зарплата» є характеристикою «Працівника» і не може існувати без нього [18];

Позначення (ідентифікатори) – ці сутності служать для встановлення зв'язків «багато до одного» або «один до одного». Наприклад, сутність «Факультет» може

ідентифікувати приналежність студента до певного факультету університету і водночас існувати як окремий об'єкт;

У моделюванні баз даних, кожна сутність та зв'язок між сутностями може мати атрибути (рис.2.2 – 2.3), які деталізують їх характеристики. Наприклад, сутність «громадянин» може мати атрибут «номер паспорту», а зв'язок між «гравцем» та «акаунтом» може включати атрибут «останній вхід». Важливо, що кожна сутність, яка не є слабкою, повинна мати первинний ключ – унікальний набір атрибутів, що дозволяє однозначно ідентифікувати кожен її екземпляр у базі даних.

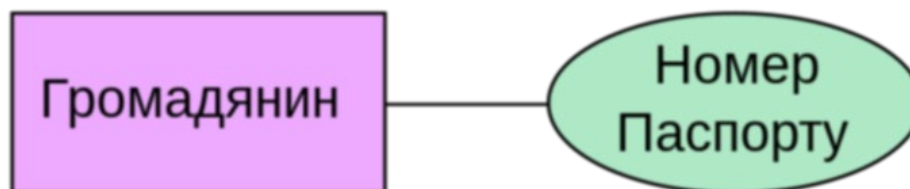


Рис. 2.2 Сутність та її атрибут

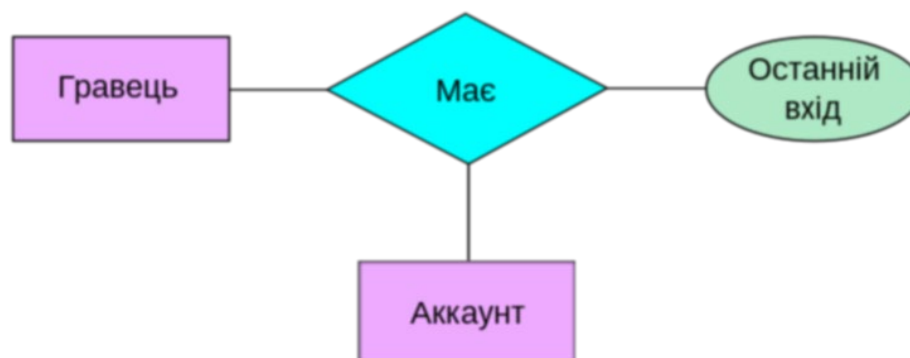


Рис. 2.3 Зв'язок та його атрибут

Фізична структура бази даних та ER-діаграма (Entity-Relationship diagram) є ключовими інструментами в процесі проектування баз даних, але вони відіграють різні ролі у цьому процесі.

Фізична структура бази даних описує, як дані фактично зберігаються у базі даних. Це включає відомості про:

- розташування даних на фізичних носіях, таких як жорсткі диски;
- методи індексації для ефективного доступу до даних;
- партитування (розподіл даних на різні частини для оптимізації продуктивності);
- структуру файлів і блоків даних.

Ця структура зосереджена на технічних можливостях зберігання даних та їх оптимізації для забезпечення швидкості виконання запитів, безпеки даних та ефективного управління ресурсами [19].

ER-діаграма — це графічне представлення структури бази даних, яке показує сутності (таблиці в реляційній базі даних) і зв'язки між ними. Основні компоненти ER-діаграми включають:

- сутності (Entities), які представляють групи об'єктів з однаковими властивостями (наприклад, таблиці в базі даних);
- зв'язки (Relationships), які показують, як сутності пов'язані одна з одною;
- атрибути кожної сутності, які представляють дані, що зберігаються в базі;
- первинні ключі (Primary keys) та вторинні ключі (Foreign keys), які використовуються для ідентифікації записів та забезпечення цілісності даних.

ER-діаграма є важливим інструментом для візуалізації та планування структури бази даних, що допомагає розробникам та аналітикам розуміти зв'язки між різними частинами даних і оптимізувати схему бази для запитів.

Фізична структура бази даних і ER-діаграма описують різні компоненти бази даних, вони тісно пов'язані. ER-діаграма допомагає визначити, які дані будуть зберігатися і як вони будуть пов'язані, тоді як фізична структура зосереджується на тому, як оптимізувати зберігання та доступ до цих даних. У разі розробки нової системи, ER-діаграма часто слугує початковим планом для створення фізичної структури бази даних [20].

Для даної предметної області ER-діаграма наведена на рис. 2.4.

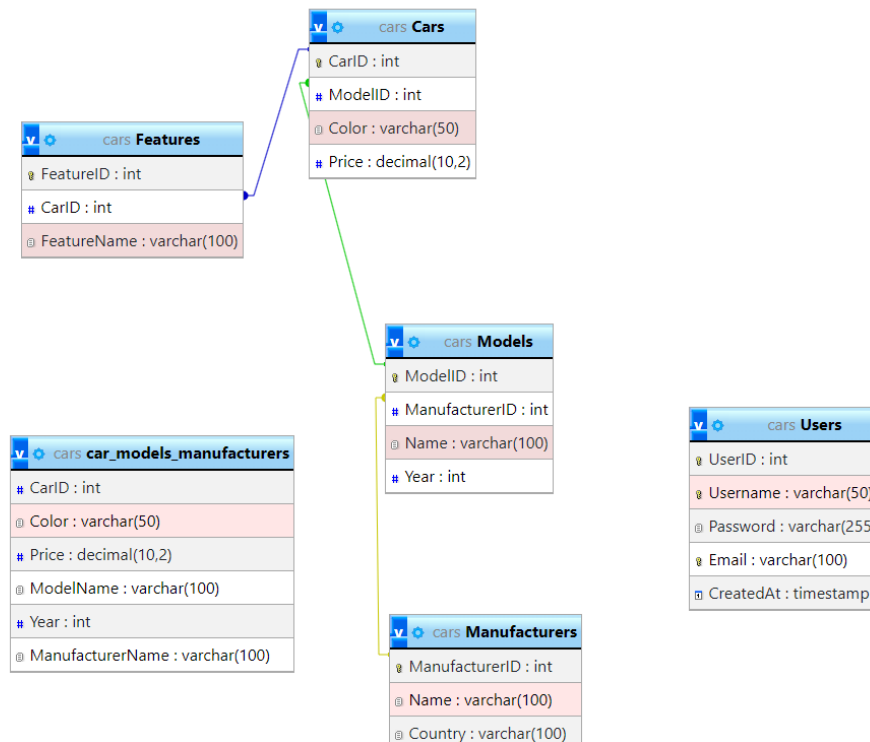


Рис. 2.4 ER-діаграма

Типи зв'язків між таблицями в базах даних можуть бути класифіковані на основі кількості сутностей, які можуть бути асоційовані одна з одною. До основних типів зв'язків відносять:

1. Один до одного (One-to-One). У цьому типі зв'язку кожен рядок в одній таблиці відповідає рівно одному рядку в іншій таблиці. Це рідкісний тип зв'язку і зазвичай використовується для розподілу даних однієї таблиці на дві для покращення організації або забезпечення конфіденційності.

2. Один до багатьох (One-to-Many) або Багато до одного (Many-to-One). Це найбільш поширений тип зв'язку. У зв'язку один до багатьох, один рядок в одній таблиці може бути асоційований з одним або багатьма рядками в іншій таблиці.

3. Багато до багатьох (Many-to-Many). У цьому типі зв'язку багато рядків в одній таблиці можуть бути асоційовані з багатьма рядками в іншій таблиці. Цей тип зв'язку зазвичай реалізується за допомогою третьої таблиці, яка відома як таблиця зв'язку або асоціативна таблиця, де зберігаються ідентифікатори обох пов'язаних таблиць.

Кожен з цих типів зв'язків вибирається в залежності від логіки доменної області та вимог до структури даних. Правильний вибір зв'язків дозволяє оптимізувати запити до бази даних і забезпечити її ефективне функціонування та масштабування [21].

Таблицю зв'язків даної предметної області наведено у табл. 2.1.

Таблиця 2.1

Типи зв'язків таблиць БД

Таблиця 1	Тип зв'язку	Таблиця 2
Автомобілі	1:N	Моделі
Особливості	1:N	Автомобілі
Моделі	1:N	Виробники

2.2. Проектування таблиць

Під час виконання кваліфікаційної роботи було створено 5 таблиць необхідних для відображення предметної області. Створені таблиці: «Автомобілі», «Особливості», «Виробники», «Моделі», «Користувачі». Вміст розроблюваної бази даних наведено на рис. 2.5.

Таблиця	Дія	Рядки	Тип	Зіставлення	Розмір	Фрагментовані
<input type="checkbox"/> Cars	☆ 🗃 🔍 🛠 🗑	6	InnoDB	utf8mb4_0900_ai_ci	32.0 КБ	-
<input type="checkbox"/> car_models_manufacturers	☆ 🗃 🔍 🛠 🗑	~0	Подання	---	-	-
<input type="checkbox"/> Features	☆ 🗃 🔍 🛠 🗑	7	InnoDB	utf8mb4_0900_ai_ci	32.0 КБ	-
<input type="checkbox"/> Manufacturers	☆ 🗃 🔍 🛠 🗑	3	InnoDB	utf8mb4_0900_ai_ci	16.0 КБ	-
<input type="checkbox"/> Models	☆ 🗃 🔍 🛠 🗑	6	InnoDB	utf8mb4_0900_ai_ci	32.0 КБ	-
<input type="checkbox"/> Users	☆ 🗃 🔍 🛠 🗑	1	InnoDB	utf8mb4_0900_ai_ci	48.0 КБ	-
6 таблиць	Всього	~23	InnoDB	utf8mb4_0900_ai_ci	160.0 КБ	0 Б

↑ Позначити все Вибрані:

Рис. 2.5 – Вміст бази даних

Проаналізуємо кожну з таблиць.

Таблиця «Автомобілі». Її атрибути:

- код автомобіля;
- код моделі;

- колір;

- ціна.

Таблиця «Особливості». Її атрибути:

- код особливості;

- код автомобіля;

- назва особливості.

Таблиця «Виробники». Її атрибути:

- код виробника;

- назва;

- країна.

Таблиця «Моделі». Її атрибути:

- код моделі;

- код виробника;

- назва;

- рік.

Таблиця «Користувачі». Її атрибути:

- код користувача;

- логін;

- пароль;

- електронна пошта;

- дата створення акаунту.

Кожна з таблиць має свою логічну структуру, яка наведена у табл. 2.2-2.6.

Таблиця 2.2

Логічна структура таблиці «Автомобілі»

Назва атрибута	Формат	Обов'язковий (необов'язковий)	Обмеження на право звертання	Первинний (вторинний) ключ	Виводимість значень	Дублювання значень, %	Умови на допустимі значення	Індексне поле
CarID	int	Обов'язковий	Залежить від ролі	Первинний ключ	Так	0%	≥ 1	Так

Продовження табл. 2.2

ModelID	int	Необов'язковий	Залежить від ролі	Вторинний ключ	Так	Можливе	Відсутнє	Так
Color	varchar(50)	Необов'язковий	Залежить від ролі	-	Так	Високе	Відсутнє	Ні
Price	decimal(10,2)	Необов'язковий	Залежить від ролі	-	Так	Можливе	≥ 0	Ні

Таблиця 2.3

Логічна структура таблиці «Особливості»

Назва атрибута	Формат	Обов'язковий (необов'язковий)	Обмеження на право звертання	Первинний (вторинний) ключ	Виводимість значень	Дублювання значень, %	Умови на допустимі значення	Індексне поле
FeatureID	int	Обов'язковий	Залежить від ролі	Первинний ключ	Так	0%	≥ 1	Так
CarID	int	Необов'язковий	Залежить від ролі	Вторинний ключ	Так	Можливе	Відсутнє	Так
FeatureName	varchar(100)	Необов'язковий	Залежить від ролі	-	Так	Високе	Відсутнє	Ні

Таблиця 2.4

Логічна структура таблиці «Виробники»

Назва атрибута	Формат	Обов'язковий (необов'язковий)	Обмеження на право звертання	Первинний (вторинний) ключ	Виводимість значень	Дублювання значень, %	Умови на допустимі значення	Індексне поле
ManufacturerID	int	Обов'язковий	Залежить від ролі	Первинний ключ	Так	0%	≥ 1	Так
Name	varchar(100)	Обов'язковий	Залежить від ролі	-	Так	Можливе	Відсутнє	Ні
Country	varchar(100)	Необов'язковий	Залежить від ролі	-	Так	Високе	Відсутнє	Ні

Таблиця 2.5

Логічна структура таблиці «Моделі»

Назва атрибута	Формат	Обов'язковий (необов'язковий)	Обмеження на право звертання	Первинний (вторинний) ключ	Виводимість значень	Дублювання значень, %	Умови на допустимі значення	Індексне поле
ModelID	int	Обов'язковий	Залежить від ролі	Первинний ключ	Так	0%	≥ 1	Так
ManufacturerID	int	Необов'язковий	Залежить від ролі	Вторинний ключ	Так	Можливе	Відсутнє	Так
Name	varchar(100)	Обов'язковий	Залежить від ролі	-	Так	Низьке	Відсутнє	Ні
Year	int	Необов'язковий	Залежить від ролі	-	Так	Високе	Відсутнє	Ні

Таблиця 2.6

Логічна структура таблиці «Користувачі»

Назва атрибута	Формат	Обов'язковий (необов'язковий)	Обмеження на право звертання	Первинний (вторинний) ключ	Виводимість значень	Дублювання значень, %	Умови на допустимі значення	Індексне поле
UserID	int	Обов'язковий	Залежить від ролі	Первинний ключ	Так	0%	≥ 1	Так
Username	varchar(50)	Обов'язковий	Залежить від ролі	Вторинний ключ	Так	Можливе	Єдине в базі	Так
Password	varchar(255)	Обов'язковий	Залежить від ролі	Вторинний ключ	Ні	Можливе	Хешований; не має бути пустим	Ні
Email	varchar(100)	Обов'язковий	Залежить від ролі	Вторинний ключ	Так	Можливе	Має відповідати формату електронної пошти	Так

2.3. Проектування та реалізація алгоритмів роботи системи

Робота веб-сервісу для оптимізації процесів в автомобільному бізнесі залежить від системних вимог: вимоги до сервера, вимоги до підключеного пристрою.

Вимоги до сервера:

- операційні системи: Windows 7 SP1 x64 / Windows Server 2008 R2 SP1 або новіші;
- вільні апаратні ресурси: від 500 МБ RAM та від 10 ГБ місця на диску;
- системне ПЗ: MSVC++ 2005-2022 Redistributable Packages (входить у комплект);
- файлова система: NTFS (мережеві диски не підтримуються);
- надійне підключення до інтернету.
- Вимоги до підключеного пристрою:

Будь-який пристрій, що має намір запустити локальний хост, повинен відповідати наступним вимогам:

- встановлений Open Server Panel, що має такі модулі:
- HTTP: Apache_2.4-PHP_7.2-7.4;
- PHP: PHP_7.2;
- MySQL/MariaDB: MySQL – 8.0- Win10.
- наявність стабільного з'єднання з інтернетом.

Функціональні вимоги розмежовують обов'язки системи, окреслюючи операції та дії, які передбачено виконати. Опис дає уявлення про різні завдання, які повинна виконувати система, включаючи обробку даних, взаємодію з користувачами та коригування поведінки системи за різних умов.

Нефункціональні вимоги акцентують увагу на характеристиках продуктивності системи. Нefункціональні вимоги фокусуються на аналізі аспектів якості, таких як надійність, швидкість, пропускну здатність, доступність та здатність до взаємодії з іншими системами. Дані атрибути суттєво впливають на загальну продуктивність та задоволеність користувачів, а отже, відіграють важливу роль у формуванні користувацького досвіду в системі.

2.4. Висновки до другого розділу

Другий розділ кваліфікаційної роботи, присвячений проектуванню бази даних, таблиць, а також проектуванню та реалізації алгоритмів роботи системи, є

ключовим елементом для створення ефективної та надійної інформаційної системи. На основі аналізу потреб бізнесу та визначення основних вимог до системи, ми розробили детальну структуру бази даних, яка включає в себе комплексний набір таблиць та зв'язків між ними. Ретельно продумане проєктування таблиць дозволяє забезпечити цілісність, безпеку та доступність даних.

Основні теми, розглянуті у цьому розділі:

1. Проєктування бази даних. Розробка структури бази даних відбувалася з урахуванням необхідності високої продуктивності та здатності масштабування системи. Оптимальна нормалізація та використання зовнішніх ключів сприяють зниженню дублювання даних і підвищенню їхньої консистентності.

2. Проєктування таблиць. Кожна таблиця була детально спроектована з визначенням типів даних для полів, встановленням первинних та вторинних ключів, що забезпечує ефективне зберігання і маніпулювання даними. Проєктування з урахуванням можливих майбутніх змін дозволяє легше адаптувати базу даних до нових вимог.

3. Проєктування та реалізація алгоритмів роботи системи. Алгоритми, реалізовані в системі, були спрямовані на оптимізацію процесів введення, обробки, зберігання і виведення даних.

Розроблена база даних та алгоритми її обробки створюють міцну основу для подальшої реалізації функціональності системи. Ефективне проєктування на даному етапі забезпечує високу швидкість роботи системи та її надійність, що є вирішальним для забезпечення задоволення бізнес-потреб замовника. На основі цього проєкту можна розробляти додаткові модулі та функції, розширюючи можливості системи відповідно до зростаючих вимог користувачів та динаміки ринку.

РОЗДІЛ 3. РОЗРОБКА ВЕБ-САЙТУ

3.1. Мета розробки веб-сайту

Метою розробки веб-сайту є створення веб-сайту, що оптимізує процеси в автомобільному бізнесі (було прийнято рішення створити веб-сайт автосалону), що містить певну інформацію про автомобілі та їхні особливості.

Сайт містить 10 сторінок: «Головна», «Автомобілі», «Особливості», «Адмін», «Особистий кабінет», «Редагування», «Редагування автомобіля», «Редагування виробника», «Редагування особливості», «Редагування моделі», «Видалення».

На головній сторінці розміщена вся важлива інформація про сайт і актуальний каталог автомобілей.

3.1.1 Написання SQL запитів до БД відповідно до мети створення веб-сайту

Скрипт, що виводить представлення на головній сторінці по автомобілям, моделям та їхнім маркам (рис. 3.1).

```
$sql = "SELECT Manufacturers.Name AS ManufacturerName, Models.Name AS ModelName, Models.Year
      FROM Models
      INNER JOIN Manufacturers ON Models.ManufacturerID = Manufacturers.ManufacturerID
      WHERE Models.Name = ?";
```

Рис. 3.1 Запит, що здійснює пошук по назві моделі

Скрипт, що виводить представлення на головній сторінці по представленню «Автомобілі-моделі-виробники» (рис. 3.2).

```
$sql = "SELECT Cars.CarID, Cars.Color, Cars.Price, Models.Name AS ModelName, Models.Year, Manufacturers.Name AS ManufacturerName
      FROM Cars
      JOIN Models ON Cars.ModelID = Models.ModelID
      JOIN Manufacturers ON Models.ManufacturerID = Manufacturers.ManufacturerID";
```

Рис. 3.2 Запит, що виводить всі записи з представлення «Автомобілі-моделі-виробники»

Скрипт, що виводить всю інформацію з таблиці «Особливості» (рис. 3.3).

```
$$sql = "SELECT Cars.CarID, Cars.Color, Cars.Price, Models.Name AS ModelName, Models.Year, Manufacturers.Name AS ManufacturerName, Features.FeatureName
FROM Cars
JOIN Models ON Cars.ModelID = Models.ModelID
JOIN Manufacturers ON Models.ManufacturerID = Manufacturers.ManufacturerID
JOIN Features ON Cars.CarID = Features.CarID";
```

Рис. 3.3 Запит, що виводить всі записи з таблиці «Features»

Скрипт, що виводить всю інформацію з таблиці «Особливості» (рис. 3.4).

```
$query = "SELECT * FROM users WHERE username = ? AND password = ?";
```

Рис. 3.4 Запит, що здійснює підключення до бази даних у таблицю «Users» при авторизації

Скрипт, що оновлює інформацію з таблиці «Моделі» (рис. 3.5).

```
$$sql = "UPDATE Models SET ManufacturerID='$manufacturerID', Name='$name', Year='$year' WHERE ModelID='$modelID'";
if ($conn->query($sql) === TRUE) {
```

Рис. 3.5 Запит, що оновлює інформацію по моделям

Скрипт, що оновлює інформацію з таблиці «Виробники» (рис. 3.6).

```
$$sql = "UPDATE Manufacturers SET Name='$name', Country='$country' WHERE ManufacturerID='$manufacturerID'";
```

Рис. 3.6 Запит, що оновлює інформацію по виробникам

Скрипт, що оновлює інформацію з таблиці «Особливості» (рис. 3.7).

```
$$sql = "UPDATE Features SET CarID='$carID', FeatureName='$featureName' WHERE FeatureID='$featureID'";
```

Рис. 3.7 Запит, що оновлює інформація по особливостям машини

Скрипт, що оновлює інформацію з таблиці «Машини» (рис. 3.8).

```
$sql = "UPDATE Cars SET ModelID='$modelID', Color='$color', Price='$price' WHERE CarID='$carID'";
if ($conn->query($sql) === TRUE) {
```

Рис. 3.8 Запит, що оновлює інформацію по машинам

3.2 Опис створення сторінок

Для оформлення стилів (фон та блоки) веб-сайту використано наступну стильову специфікацію (рис.3.9):

```
<style>
/* Стили для контейнера особистого кабінету */
.dashboard-container {
    max-width: 600px;
    margin: 50px auto;
    padding: 20px;
    background-color: #f8f9fa;
    border: 1px solid #ced4da;
    border-radius: 5px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

/* Стили для заголовка особистого кабінету */
.dashboard-container h2 {
    font-size: 24px;
    margin-bottom: 20px;
    color: #333;
}

/* Стили для тексту в особистому кабінеті */
.dashboard-container p {
    font-size: 16px;
    line-height: 1.6;
    color: #666;
}
```

Рис. 3.9 Код стильової специфікації CSS для створення фону та блоків

Для побудови меню сайту навігаційні елементи (рис.3.10) у комбінації з

посиланнями укладеними у блочні конструкції.

Для створення навігаційного меню використовувався елемент `<nav>`, який дозволяє групувати посилання та інші елементи для навігації по сайту.

Для організації структури та розміщення контенту використовувалися різні блочні елементи, такі як `<div>`. Ці елементи дозволяють групувати та розташовувати інші елементи на сторінці відповідно до потреб дизайну та логіки веб-сервісу.

```

71
72 <!-- top-area Start -->
73 <div class="top-area">
74 <div class="header-area">
75 <!-- Start Navigation -->
76 <nav class="navbar navbar-default bootsnav navbar-sticky navbar-scrollspy" data-minus-value-desktop="70" data-minus-value-mobile="55" data-speed="10">
77
78 <div class="container">
79
80 <!-- Start Header Navigation -->
81 <div class="navbar-header">
82 <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#navbar-menu">
83 <i class="fa fa-bars"></i>
84 </button>
85 <a class="navbar-brand" href="index.php">cars<span></span></a>
86
87 </div><!-- /.navbar-header -->
88 <!-- End Header Navigation -->
89
90 <!-- Collect the nav links, forms, and other content for toggling -->
91 <div class="collapse navbar-collapse menu-ui-design" id="navbar-menu">
92 <ul class="nav navbar-nav navbar-right" data-in="fadeInDown" data-out="fadeOutUp">
93 <li class="scroll active"><a href="#home">Головна</a></li>
94 <li class="scroll"><a href="#service">Автомобілі</a></li>
95 <li class="scroll"><a href="#featured-cars">Особливості</a></li>
96
97 <li><a href="login.php">Адмін</a></li>

```

Рис. 3.10 Код написаний мовою HTML для створення меню веб-сайту

Вміст сторінки контролюється за допомогою наступного коду (рис.3.11):

```

<div class="container">
  <div class="welcome-hero-txt">
    <h2>Каталог вашої мрії</h2>
  </div>
</div>

<div class="container">
<div class="row">
<div class="col-md-12">
  <div class="model-search-content">
    <div class="row">
      <form method="post" action="" class="form-inline">
        <div class="col-md-offset-1 col-md-6 col-sm-12">
<div class="single-model-search">
<h2>ВВЕДІТЬ НАЗВУ МОДЕЛІ АВТО</h2>
<input type="text" class="form-control" name="model">

<div class="text-center">
  <button class="welcome-btn model-search-btn" type="submit" name="submit">
    Пошук
  </button>
</div>
</div>
</div>

```

Рис. 3.11 Код написаний мовою HTML для контролю над основним вмістом сторінки

Важливим параметром розробки веб-сервісу є забезпечення адаптивності сайту для різних типів пристроїв. Це досягається за допомогою CSS-медіа-запитів, які дозволяють налаштовувати стилі в залежності від характеристик пристрою, таких як розмір екрану, орієнтація тощо.

3.3 Розробка інтерфейсу користувача

Створення інтерфейсу користувача на базі Web-технологій вимагає знання мови розмітки сторінок (HTML), а також взаємодією з каскадними листами стилів (CSS).

Класичний інтерфейс Web-Додатків складається із трьох частин:

- Заголовок сайту – по можливості повинен давати користувачеві інформацію про призначення сайту.
- Меню першого рівня – також надає користувачеві додаткову інформацію про функціональність і можливості сайту.
- Вміст сайту – основний (найбільший) блок у якому безпосередньо відображається наповнення сайту.

Сайт ІС «Каталог автомобілей» буде мати наступну структуру:

- Заголовок: напис «Cars» - назва інтернет-каталогу, буде зроблено великим шрифтом та розташовано на видному місці.
- Меню сайту: «Головна», «Автомобілі», «Особливості автомобілей», «Адмін». Меню сайту продемонстровано на рисунку 3.12. Головна сторінка продемонстрована на рисунках 3.13-3.17.
- Вміст сайту: буде відображати вміст поточної сторінки згідно запиту користувача. На сторінках «Головна» відображається вся доступна інформація по автомобілям, тобто, їхня марка, назва, картинка, країна виробник. На сторінці «Особливості» відображається особливості автомобіля, наприклад, кондиціонер, або автопідсилювач керма. На сторінці «Адмін» відображається форма, куди може входити тільки адміністратор, який знає логін та пароль.

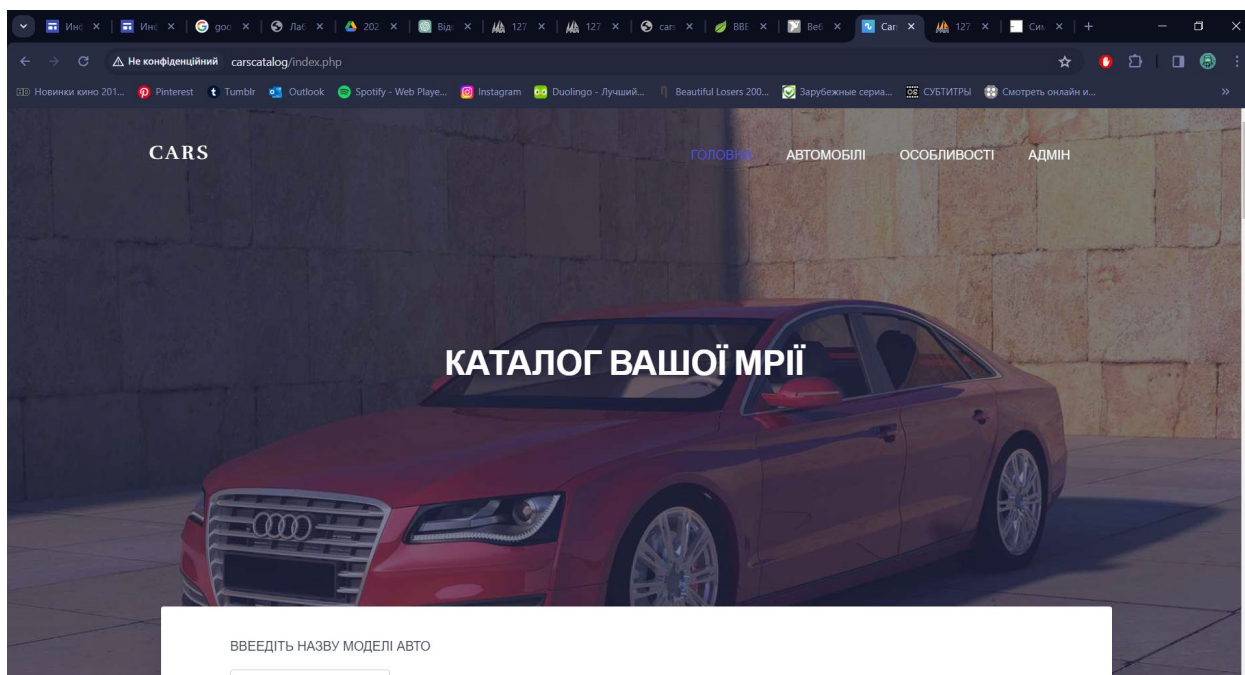


Рис. 3.12 Меню сайту

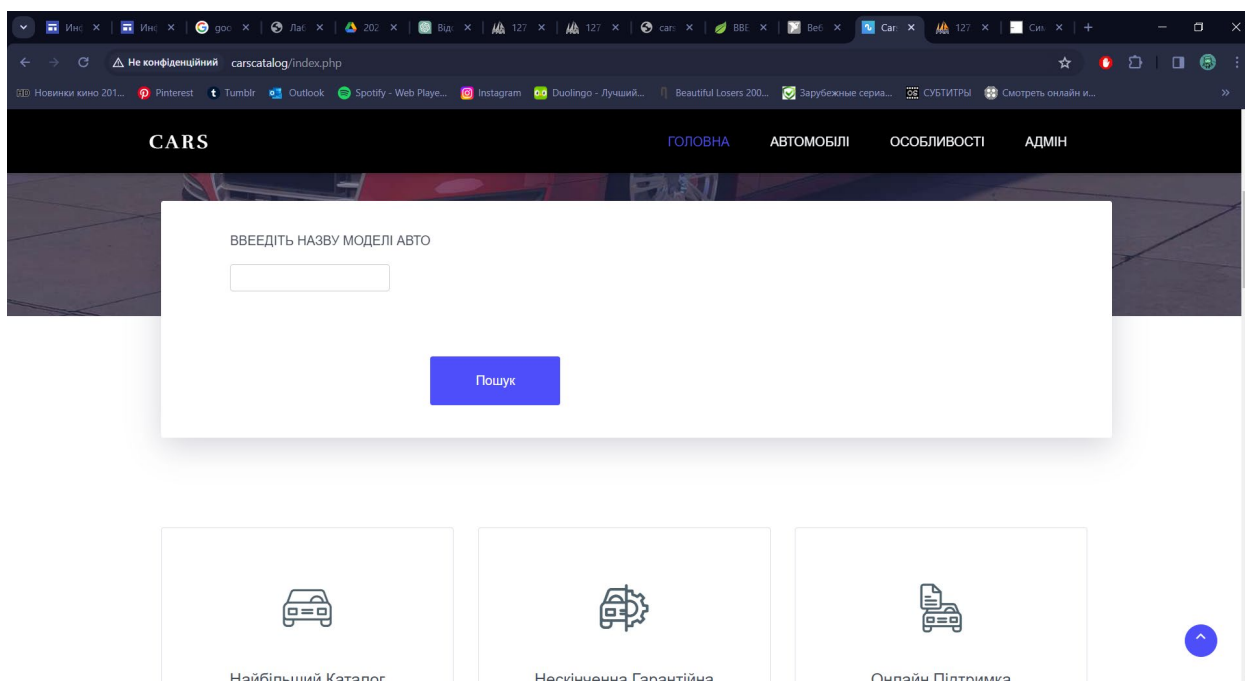


Рис. 3.13 Головна сторінка

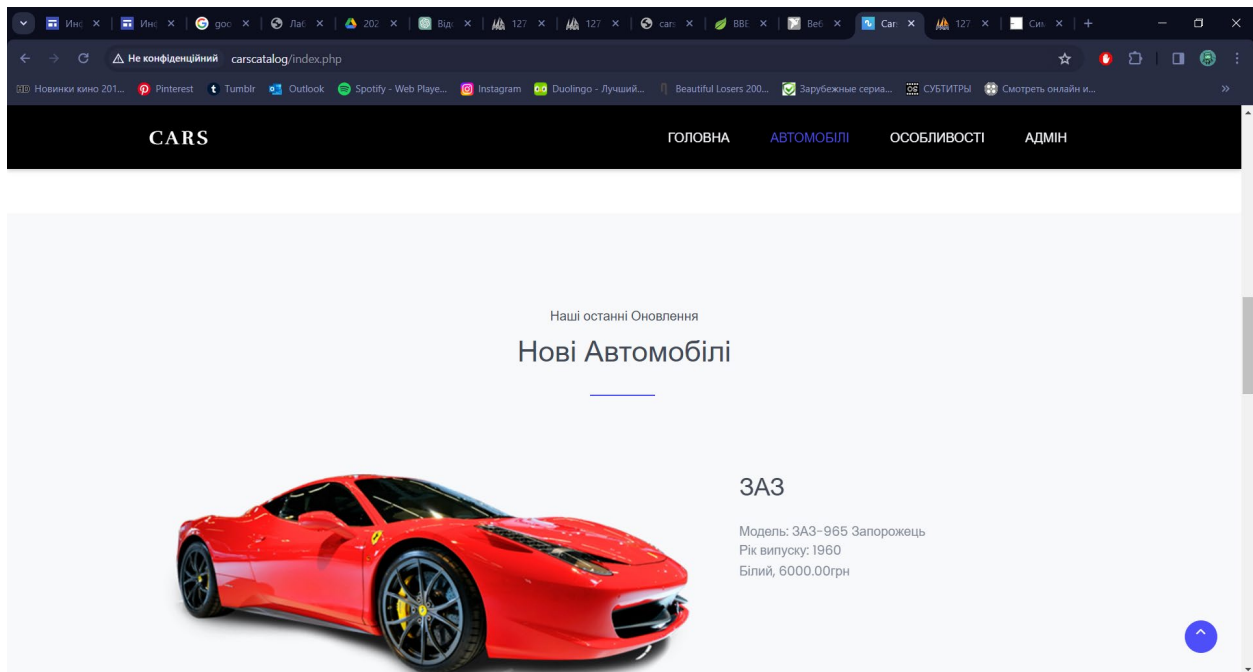


Рис. 3.14 Головна сторінка

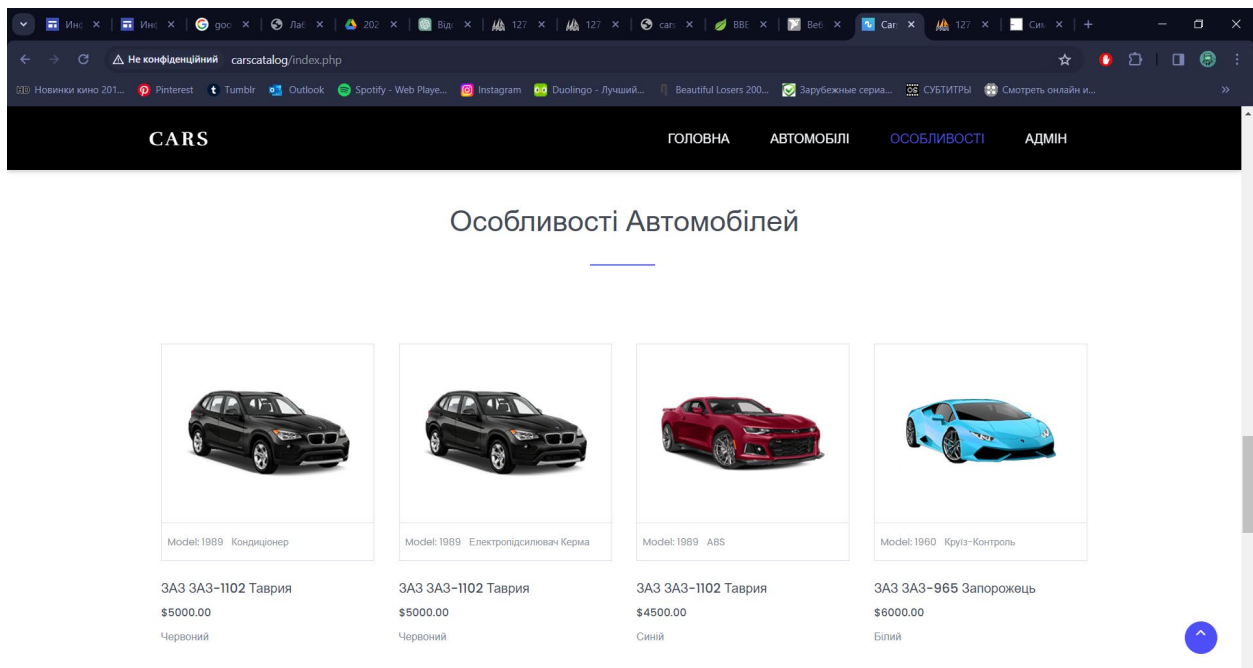


Рис. 3.15 Головна сторінка

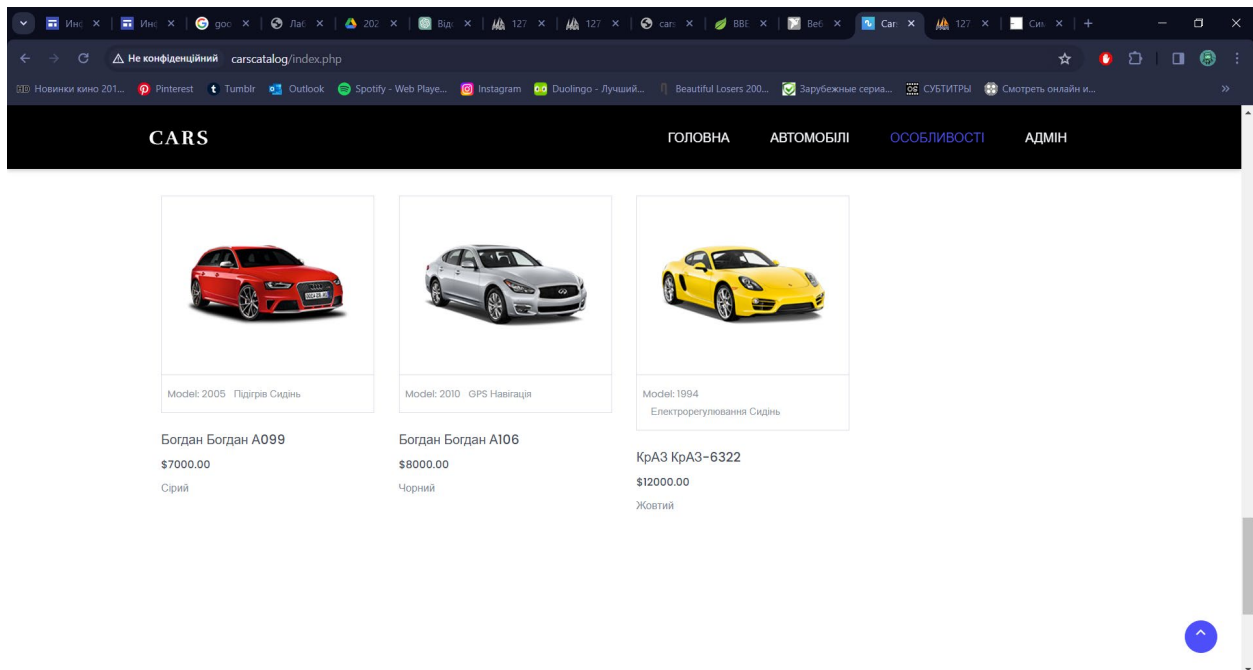


Рис.3.16 Головна сторінка

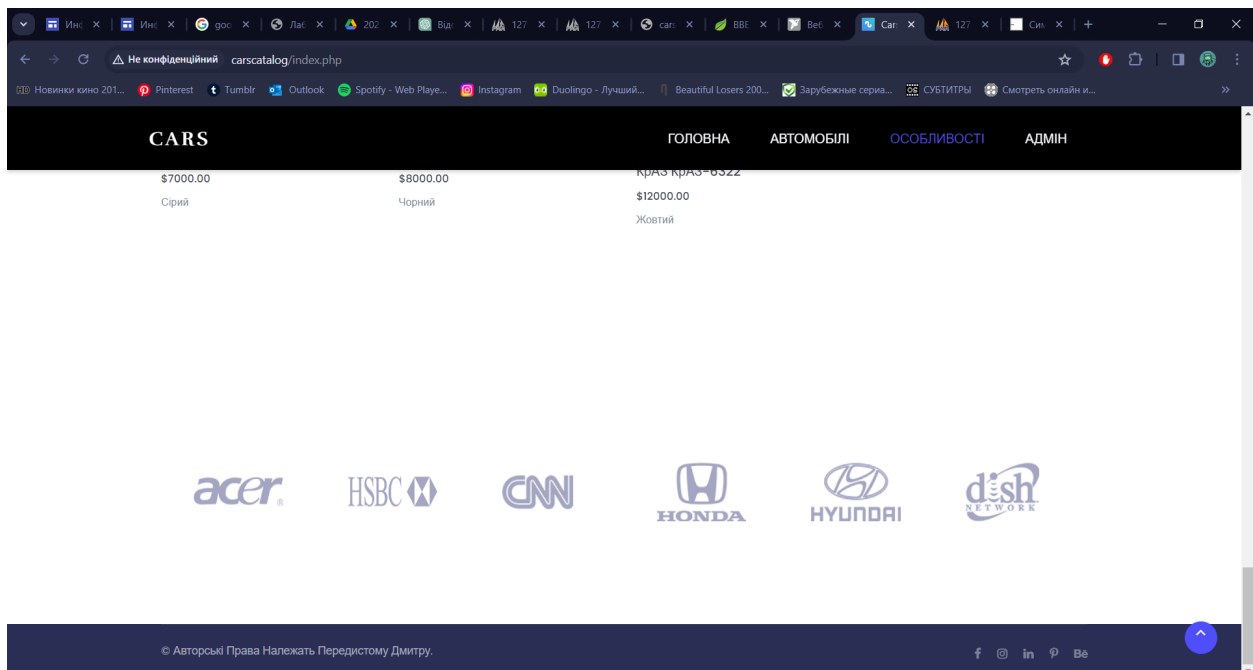


Рис.3.17 Головна сторінка

3.4 Розробка сценаріїв серверної частини ІС

3.4.1 Компонування сценаріїв

Існує два базових підходи компонування коду серверних сценаріїв:

1. Весь код фізично знаходиться у одному файлі, та в залежності від дій

користувача – генерує різний вміст (контент).

2. Код розподіляється по окремим файлам, в залежності від завдань.

3. Гібридний підхід, довільна комбінація перших двох.

Переваги та недоліки кожного з підходів:

1. Весь код у одному файлі:

Переваги:

- простота. Легше почати розробку, оскільки усі елементи знаходяться в одному місці;

- швидкість розробки. внесення змін швидше, оскільки немає необхідності шукати файли з кодом.

Недоліки:

- несприятливий для підтримки. великий файл може бути важко зберігати і вносити зміни в майбутньому;

- збільшення ризику помилок. складніше виявляти та виправляти помилки через об'єм коду.

2. Розподіл коду по окремим файлам:

Переваги:

- легкість підтримки. Код легше організувати та зберігати, оскільки він розділений на логічні частини;

- масштабованість. Зручно додавати новий функціонал та розширювати додаток.

Недоліки:

- збільшення складності. Може виникнути плутанина між файлами або потреба у більш складній системі управління версіями.

3. Гібридний підхід:

Переваги:

- гнучкість. Дозволяє використовувати кращі аспекти обох підходів залежно від конкретної ситуації;

- підвищення продуктивності. Можна використовувати простий підхід для швидкої розробки та перейти до розподілу коду, коли проект зростає у розмірах.

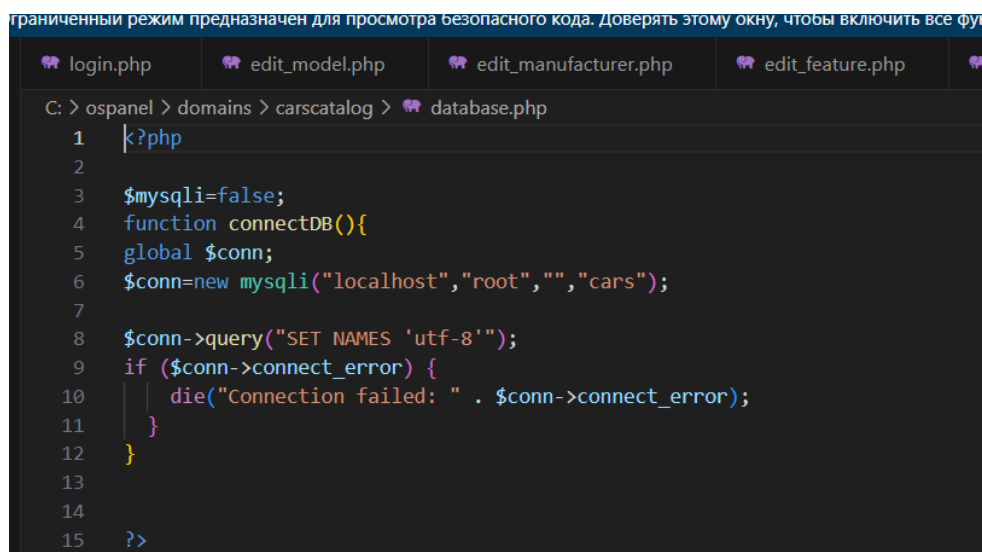
Недоліки:

- складність управління. Потребує уважного планування та комунікації між розробниками для забезпечення консистентності у роботі над проектом.

З метою вивчення процесу створення серверних сценаріїв, для початківців рекомендується використовувати другий підхід, і розподіляти логічно різні блоки коду по фізично різних файлах, що надалі полегшує налагодження й пошук помилок.

У випадку наявності ділянок коду, що дублюються в усіх (або багатьох) файлах, питання вирішується використанням оператора `require` і вилучення повторюваного коду в окремий файл.

У даній роботі використовується гібридний підхід, на рисунку 3.20 продемонстрований другий підхід, коли код розподіляється по окремим файлам, в залежності від завдань, як це продемонстровано на рисунку 3.21.



```

C: > ospanel > domains > carscatalog > database.php
1  |<code>?php</code>
2
3  |<code>$mysqli=false;</code>
4  |<code>function connectDB(){</code>
5  |<code>global $conn;</code>
6  |<code>$conn=new mysqli("localhost","root","","cars");</code>
7
8  |<code>$conn->query("SET NAMES 'utf-8'");</code>
9  |<code>if ($conn->connect_error) {</code>
10 |<code>    die("Connection failed: " . $conn->connect_error);</code>
11 |<code>}</code>
12 |<code>}</code>
13
14
15 |<code>?></code>

```

Рис. 3.20 Створення окремого файлу для підключення бази даних

На рис. 3.21 продемонстрований перший підхід, коли весь код знаходиться у одному файлі, і в залежності від дій користувача – генерує різний зміст.

```

38
39 <?php
40 require_once "database.php";
41 connectDB();
42
43 if(isset($_POST['submit'])) {
44     $modelName = $_POST['model'];
45
46     $sql = "SELECT Manufacturers.Name AS ManufacturerName, Models.Name AS ModelName, Models.Year
47            FROM Models
48            INNER JOIN Manufacturers ON Models.ManufacturerID = Manufacturers.ManufacturerID
49            WHERE Models.Name = ?";
50

```

Рис.3.21 Демонстрація першого підходу

3.5 Інструкція користувача

На сторінці «Головна» відображається інформація про новинку року, про інтернет магазин, його особливості, каталог продукції та знижки.

Сторінка «Про нас» містить всю актуальну інформацію про інтернет магазин. Демонстрацію наведено на рисунку 3.22.

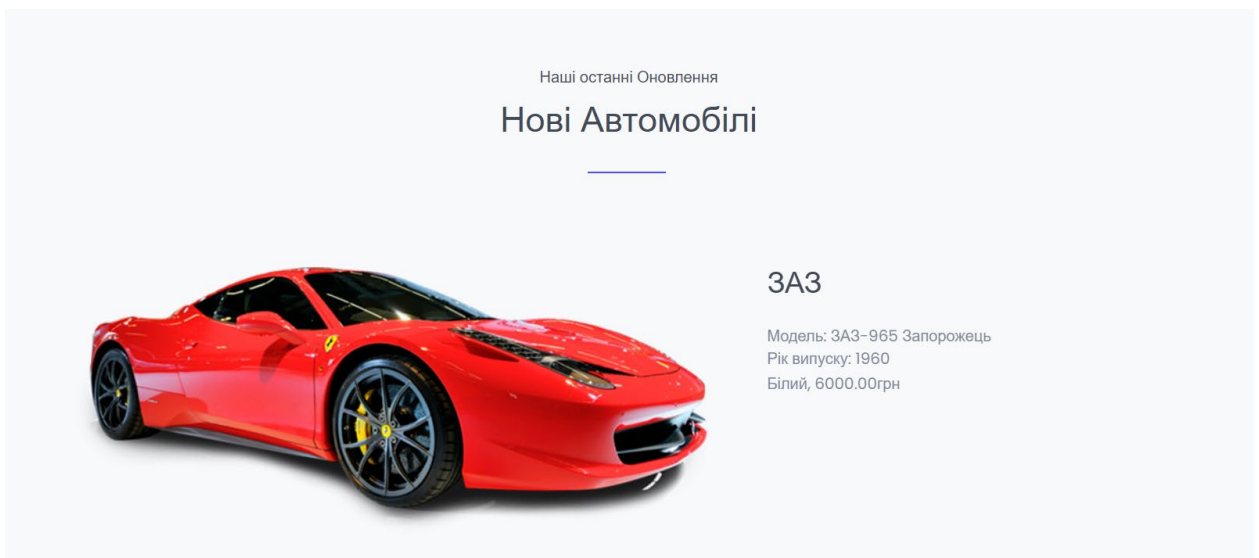


Рис. 3.22 Сторінка «Автомобілі»

Сторінка «Особливості» відображає всі особливості, що мають певні автомобілі. Наведена на рис. 3.23.

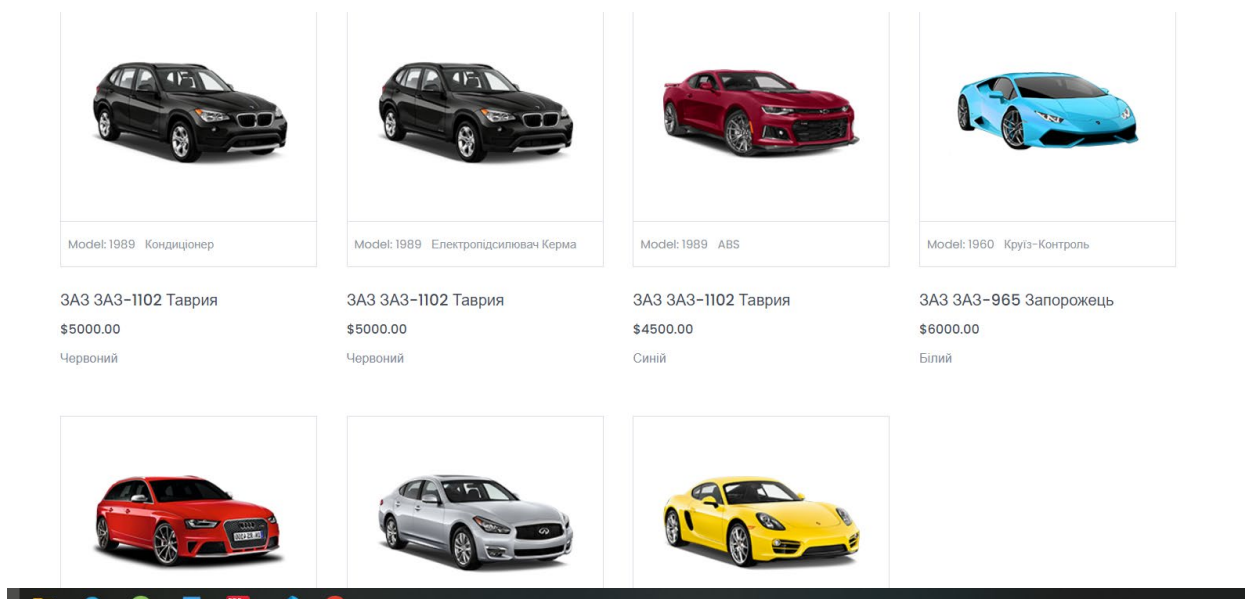


Рис. 3.23 Сторінки «Особливості»

Сторінка «Адмін» використовується для авторизації. Демонстрація сторінки наведена на рисунку 3.24.

При неправильній авторизації видається повідомлення, що наведено на 3.25.

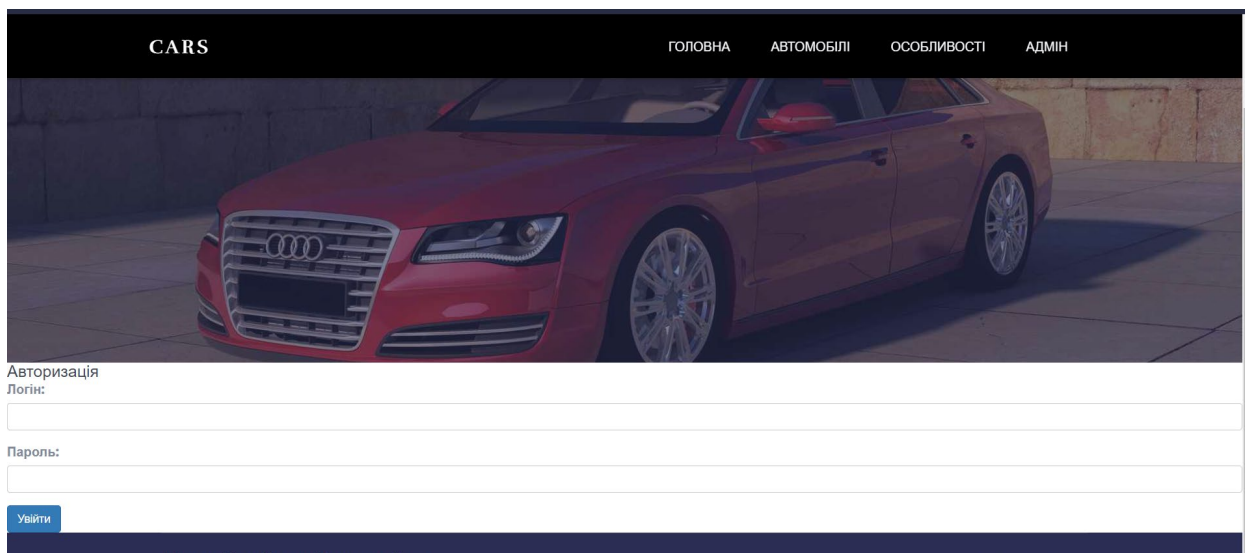


Рис. 3.24 Сторінка «Особистий кабінет»

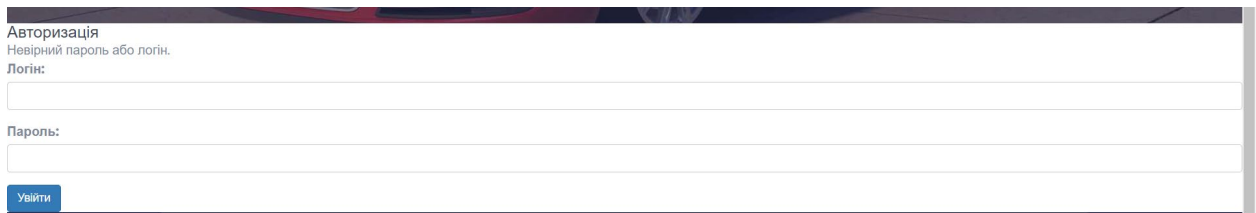


Рис. 3.25 Сторінка «Авторизація», якщо неправильно уведено логін чи пароль

Сторінка «Особистий кабінет» наведена на рис.3.26.

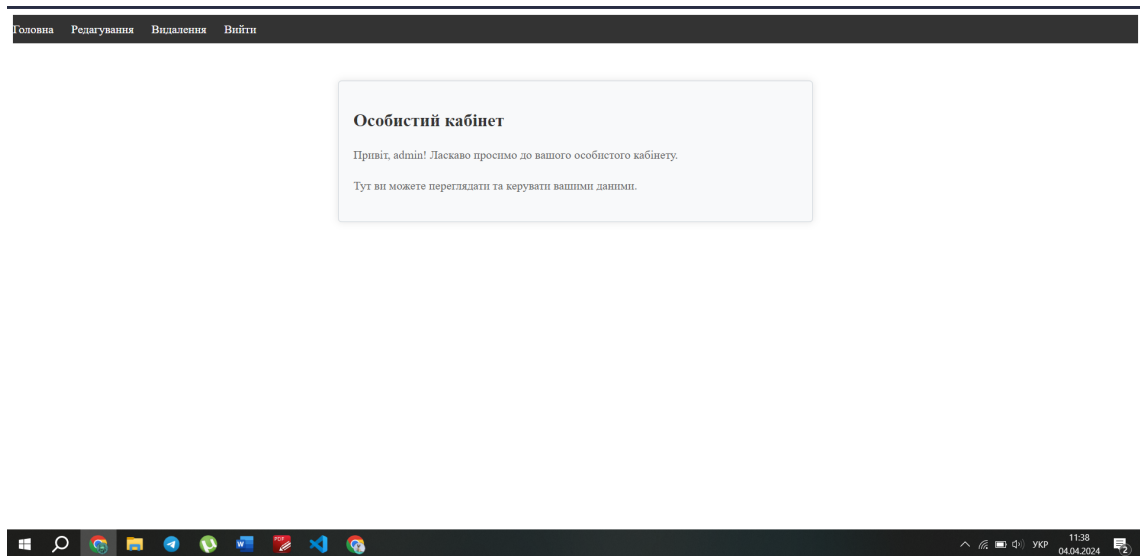


Рис. 3.26 Сторінка «Особистий кабінет»

У даній ІС передбачена можливість додавання або видалення ігор, це можна зробити з особистого кабінету у «Сторінці адміністратора».

Як зрозуміло з назви самої сторінки, це сторінка розроблена спеціально для адміністратора з табличним поданням при авторизації, щоб відредагувати (рис.3.27).

Сторінка видалення інформації наведена на рис.3.28.

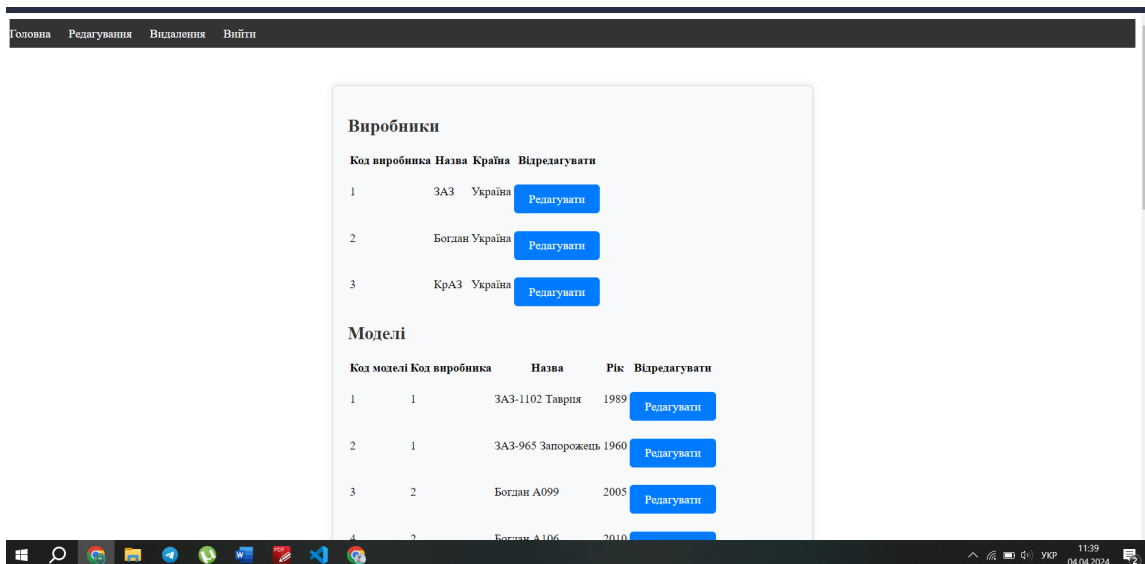


Рис 3.27 «Редагування»

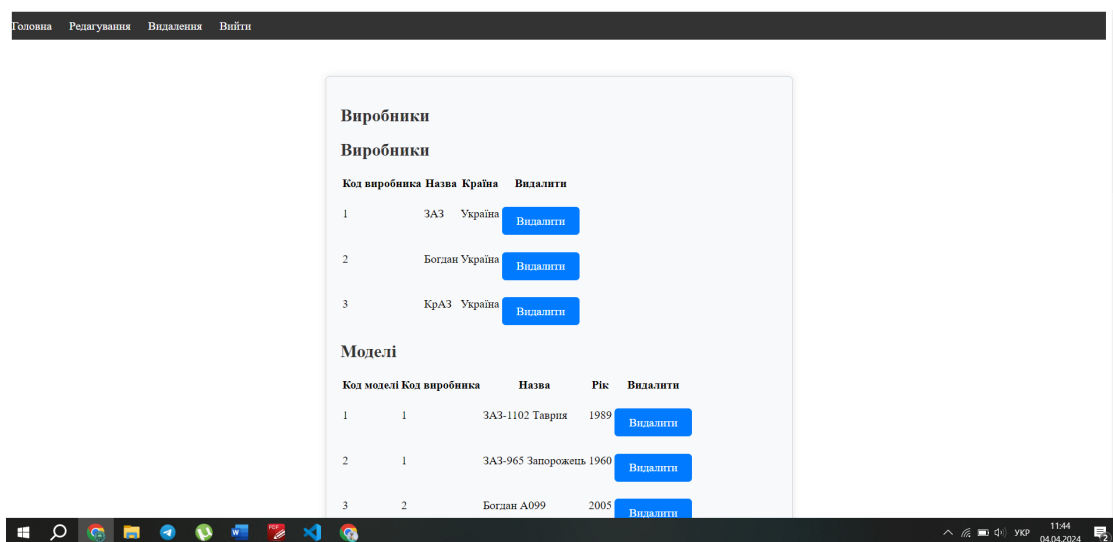


Рис. 3.28 Особистий кабінет адміністратора з можливостями на видалення

На рисунках 3.29-3.30 продемонстровано видалення інформації з особистого кабінета адміністратора.

Моделі

Код моделі	Код виробника	Назва	Рік	Видалити
1	1	ЗАЗ-1102 Таврія	1989	Видалити
2	1	ЗАЗ-965 Запорожець	1960	Видалити
3	2	Богдан А099	2005	Видалити
4	2	Богдан А106	2010	Видалити
5	3	КрАЗ-6322	1994	Видалити
6	3	КрАЗ-255	1963	Видалити

Рис. 3.29 Таблиця «Моделі» до видалення інформації

Моделі

Код моделі	Код виробника	Назва	Рік	Видалити
1	1	ЗАЗ-1102 Таврія	1989	Видалити
2	1	ЗАЗ-965 Запорожець	1960	Видалити
3	2	Богдан А099	2005	Видалити
4	2	Богдан А106	2010	Видалити
5	3	КрАЗ-6322	1994	Видалити

Рис.3.30 Таблиця «Моделі» після видалення інформації

3.6 Висновки до третього розділу

Даний розділ присвячений розробці веб-сайту для автосалону, зосереджується на детальному аналізі і впровадженні різноманітних аспектів створення веб-ресурсу, від проєктування бази даних до інтерфейсу користувача і серверних сценаріїв. Завдяки цьому вдається створити функціональний та естетично привабливий сайт, який ефективно відповідає на потреби користувачів і водночас забезпечує легке управління контентом.

Основні елементи, які були реалізовані в рамках проєкту:

- створення веб-сайту, який дозволяє користувачам легко ознайомитися з асортиментом автомобілів, їх характеристиками і доступними моделями. Ціль була успішно досягнута через структуровану розробку та імплементацію інформаційної системи з чітко визначеними категоріями та детальними описами;

- веб-інтерфейс був спроектований так, щоб бути інтуїтивно зрозумілим і зручним для користувачів. Використання HTML та CSS дозволило створити приємний візуальний дизайн, який сприяє кращому сприйняттю інформації та взаємодії з сайтом;

- використання різних підходів у написанні серверного коду сприяло створенню масштабованого і гнучкого рішення, що може адаптуватися до змінних вимог і розширення функціоналу веб-сайту;

- написання SQL-запитів, які ефективно взаємодіють з базою даних для відображення, оновлення, видалення та додавання даних, гарантувало, що інформація на сайті завжди актуальна і точна.

Цей проєкт демонструє здатність інтегрувати різні технології та підходи для створення комплексного веб-сайту, який служить важливим ресурсом для автосалону. Він не лише забезпечує ефективне управління даними та користувачами через веб-інтерфейс, а й покращує загальну клієнтську взаємодію з автосалоном, що сприяє підвищенню продажів і задоволення клієнтів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Що таке CRM система і як вона працює? URL: <https://www.creatio.com/ua/crm/what-is-crm> (дата звернення: 07.05.2024)
2. Система для автосалонів і автодилерів. URL: <https://wezom.com.ua/ua/blog/crm-sistema-dlja-avtosalonov-i-avtodilerov> (дата звернення: 07.05.2024)
3. Що таке CRM система і як її обрати? URL: <https://business.diia.gov.ua/handbook/prodazi/so-take-crm-sistema-ak-obrati-j-pracuvati-z-crm-so-vazlivo-zamiruvati> (дата звернення: 07.05.2024)
4. CRM система для автосервісу, автосалону, СТО. URL: <https://salesdrive.ua/by-industry/auto/> (дата звернення: 07.05.2024)
5. 10 систем для бізнесу в різних нішах. URL: <https://cases.media/article/10-crm-sistem-dlya-biznesu-v-riznikh-nishakh> (дата звернення: 07.05.2024)
6. Automotive Cloud. URL: <https://www.salesforce.com/products/automotive-cloud/overview/> (дата звернення: 07.05.2024)
7. Microsoft Dynamics 365 for Automotive. URL: <https://learn.microsoft.com/en-us/dynamics365/industry/accelerators/automotive> (дата звернення: 07.05.2024)
8. Microsoft Dynamics 365 for Automotive. URL: https://www.googleadservices.com/pagead/aclk?sa=L&ai=DChcSEwjq5cbVmP-FAxUeaJEFHTtLBsQYABACGgJscg&ase=2&gclid=Cj0KCQJwxeYxBhC7ARIsAC7dS3_zs6Tz77n1Muwg3pjxP1626t6toiLKZlcDwB7sZ61ZwwTFGleCGs8aAtT6EALw_wcB&ohost=www.google.com&cid=CAESVeD2hBekOLG2TQUP_ZvyQepHH0AvBpDNgbUCkzFow-3rlE0z84af8h64IEpto_9rpnG-_W89Ie0HuZosatLZUhO7EVy4hCqijb4X6QTr9K9AIMLdfJU&sig=AOD64_1D2UTFWQtpEaOBZt7QEox0DGQxnQ&q&nis=4&adurl&ved=2ahUKEwimycHVmP-FAxUWJhAIHWm8BksQ0Qx6BAgIEAM (дата звернення: 07.05.2024)
9. Пасічник В. В., Резніченко В. А. Організація баз даних та знань. – К.: Видавнича група BHV, 2006. – 384 с.: іл.

10. Харів Н. О. Х 20 Бази даних та інформаційні системи: навчальний посібник /

Н. О. Харів. – Рівне : НУВГП, 2018. – 127 с.

11. Лосєв М. Ю. Бази даних : навчально-практичний посібник для самостійної роботи студентів [Електронний ресурс] / М. Ю. Лосєв, В. В. Федько. – Харків : ХНЕУ ім. С. Кузнеця, 2018. – 233 с.

12. Тарасов О. В. Використання мови SQL для роботи з сучасними системами керування базами даних / О. В. Тарасов, В. В. Федько, М. Ю. Лосєв. – Харків : Вид. ХНЕУ, 2013. – 348 с

13. . Федько В. В. Лабораторний практикум з модуля "Основи баз даних та знань" навчальної дисципліни "Організація баз даних та знань" / В. В. Федько, О. В. Тарасов, М. Ю. Лосєв. – Харків : Вид. ХНЕУ, 2011. – 192 с.

14. Тарасов О. В. Проектування баз даних : навч. посіб. / О. В. Тарасов, В. В. Федько, М. Ю. Лосєв. – Харків : Вид. ХНЕУ, 2011. – 200 с.

15. Федько В. В. Організація баз даних та знань : навч.-прак. посіб. / В. В. Федько, О. В. Тарасов, М. Ю. Лосєв. – Харків : Вид. ХНЕУ, 2013. – 200 с.

16. Доценко С. І. Організація та системи керування базами даних: Навч. посібник. – Харків: УкрДУЗТ, 2023. – 117 с.

17. Гайна Г. А. Основи проектування баз даних: навч. посіб. Київ: КНУБА, 2005. 204 с.

18. Берко А.Ю., Верес О.М.Організація баз даних: практичний курс: Навч. посібник / За наук. ред. В.В. Пасічника. - Львів: Видавництво Національного університету "Львівська політехніка", 2003. - 152 с.

19. Берко А.Ю. Проектування та експлуатація баз даних реляційного гину: Методичні вказівки по виконанню лабораторних робіт з курсу “Організація баз даних” для студентів, що навчаються за базовим напрямком “Комп’ютерні науки”. - Львів, ДУ ЛГІ, 1996. - 56 с

20. Буй Д. Б., Сільвейструк Л. М. Формалізація моделі «сутність-зв’язок». Монографія. – К.: ВПЦ «Київський університет»., 2011.– 175 с.

21. Гайна Г. А. Основи проектування баз даних: Навчальний посібник. – К: КНУБА, 2005. – 204 с.

ВИСНОВКИ

У роботі розглянуто кілька важливих особливостей, пов'язаних з розробкою інформаційної системи для автомобільного бізнесу. Це включало аналіз, проектування, та реалізацію веб-сервісу, що оптимізує процеси в автомобільній індустрії, створення структурованої бази даних, а також детальне планування користувацького інтерфейсу та серверної частини системи.

Перший розділ розкрив важливість і необхідність оптимізації процесів через веб-сервіси, які можуть значно підвищити ефективність роботи автосалонів, покращити управління запасами та забезпечити кращу інтеграцію даних для прийняття обґрунтованих бізнес-рішень.

У автомобільному бізнесі веб-сервіси допомагають вирішувати різні проблеми, зокрема, управління запасами, обслуговування клієнтів та прийняття рішень. Наприклад, веб-сервіси для управління запасами дозволяють автосалонам вести облік автомобілів на складі, автоматично оновлювати інформацію про наявність та виробляти замовлення при необхідності. Це сприяє оптимізації запасів та уникненню надмірної або недостатньої наявності товарів. Обслуговування клієнтів через веб-сервіси забезпечує швидкий доступ до інформації про автомобілі, послуги та стан замовлення, що покращує комунікацію та задоволення клієнтів. Вирішення цих проблем сприяє покращенню управління автомобільним бізнесом та прийняттю обґрунтованих рішень.

Другий розділ продемонстрував методологію проектування бази даних та таблиць, яка вимагала детального підходу до вибору технологій і структурних рішень. Забезпечення цілісності даних, безпека, і гнучкість системи були визначені як ключові критерії успіху проекту.

У проектуванні бази даних враховувалися ключові аспекти, що забезпечили цілісність даних та гнучкість системи. Наприклад, структурні рішення, такі як нормалізація бази даних, допомагали уникнути дублювання інформації та забезпечити її консистентність. Використання індексів та оптимізованих запитів дозволяло підвищити швидкість доступу до даних. Такі підходи забезпечили

ефективне управління даними та забезпечили зручність у використанні бази даних для різних потреб системи.

Третій розділ описав практичні кроки створення веб-сайту, включно з розробкою інтерфейсу користувача і серверних сценаріїв. Це включало створення інтуїтивно зрозумілого дизайну, який дозволяє користувачам легко навігувати між різними розділами сайту, і забезпечення стабільної роботи системи через ефективні серверні технології.

Серверні сценарії та інтерфейс користувача мали значний вплив на функціональність та зручність веб-сайту. Шляхом впровадження ефективних серверних сценаріїв, таких як обробка форм та запитів, забезпечувалася стабільна та швидка робота веб-сайту. Крім того, розробка інтуїтивно зрозумілого інтерфейсу користувача дозволила забезпечити зручну навігацію та приємний досвід взаємодії для користувачів.

Можливі напрямки майбутнього розширення системи включають інтеграцію з додатковими сервісами та платформами, розширення функціональності для покращення аналітики даних та забезпечення автоматизації бізнес-процесів, а також розвиток мобільних додатків для покращення доступності та зручності взаємодії з системою. Ці рішення допоможуть автосалонам адаптуватися до змін на ринку та збільшити їхню конкурентоспроможність.

Проект підкреслив не тільки технічні аспекти розробки інформаційних систем, але й значення взаємодії з зацікавленими сторонами та кінцевими користувачами для забезпечення їх задоволеності та відповідності бізнес-вимогам. Управління проектом, тестування і налагодження є критично важливими для забезпечення якості та надійності системи.

Завершена робота закладає основу для подальшого розширення і масштабування системи, дозволяючи інтегрувати нові технологічні рішення та адаптувати систему до змінюваних умов ринку. Потенціал для інновацій і поліпшення бізнес-процесів за допомогою ІТ-інструментів є величезним, що дозволить компаніям утримувати лідерські позиції в автомобільному бізнесі.

Таким чином, можна підсумувати, що важливість інтеграції ІТ-рішень для оптимізації бізнес-процесів та забезпечення конкурентоспроможності в автомобільній індустрії підкреслює, що ІТ-інструменти є невід'ємною частиною сучасного бізнесу. Їх використання дозволяє підприємствам не лише ефективно управляти процесами, але й швидко реагувати на зміни в ринкових умовах та задовольняти потреби клієнтів.

ДОДАТОК А – ЛІСТИНГ БАЗИ ДАНИХ

```

-- phpMyAdmin SQL Dump
-- version 5.2.0
-- https://www.phpmyadmin.net/
--
-- Хост: 127.0.0.1:3306
-- Час створення: Квт 04 2024 р., 11:56
-- Версія сервера: 8.0.30
-- Версія PHP: 7.2.34

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- База даних: `cars`
--

-----

--
-- Структура таблиці `Cars`
--

CREATE TABLE `Cars` (
  `CarID` int NOT NULL,
  `ModelID` int DEFAULT NULL,
  `Color` varchar(50) DEFAULT NULL,
  `Price` decimal(10,2) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- Дамп даних таблиці `Cars`
--

INSERT INTO `Cars` (`CarID`, `ModelID`, `Color`, `Price`) VALUES
(1, 1, 'Червоний', '5000.00'),
(2, 1, 'Синій', '4500.00'),
(3, 2, 'Білий', '6000.00'),
(4, 3, 'Сірий', '7000.00'),
(5, 4, 'Чорний', '8000.00'),
(6, 5, 'Жовтий', '12000.00');

-----

--
-- Дублююча структура для представлення `car_models_manufacturers`
-- (Див. нижче для фактичного подання)
--

CREATE TABLE `car_models_manufacturers` (
  `CarID` int

```

```

, `Color` varchar(50)
, `Price` decimal(10,2)
, `ModelName` varchar(100)
, `Year` int
, `ManufacturerName` varchar(100)
);

-----

--
-- Структура таблиці `Features`
--

CREATE TABLE `Features` (
  `FeatureID` int NOT NULL,
  `CarID` int DEFAULT NULL,
  `FeatureName` varchar(100) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- Дамп даних таблиці `Features`
--

INSERT INTO `Features` (`FeatureID`, `CarID`, `FeatureName`) VALUES
(1, 1, 'Кондиціонер'),
(2, 1, 'Електропідсилювач керма'),
(3, 2, 'ABS'),
(4, 3, 'Круїз-контроль'),
(5, 4, 'Підігрів сидінь'),
(6, 5, 'GPS навігація'),
(7, 6, 'Електрорегулювання сидінь');

-----

--
-- Структура таблиці `Manufacturers`
--

CREATE TABLE `Manufacturers` (
  `ManufacturerID` int NOT NULL,
  `Name` varchar(100) NOT NULL,
  `Country` varchar(100) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

--
-- Дамп даних таблиці `Manufacturers`
--

INSERT INTO `Manufacturers` (`ManufacturerID`, `Name`, `Country`) VALUES
(1, 'ЗАЗ', 'Україна'),
(2, 'Богдан', 'Україна'),
(3, 'КрАЗ', 'Україна');

-----

--
-- Структура таблиці `Models`
--

CREATE TABLE `Models` (
  `ModelID` int NOT NULL,
  `ManufacturerID` int DEFAULT NULL,
  `Name` varchar(100) NOT NULL,
  `Year` int DEFAULT NULL

```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
--
-- Дамп даних таблиці `Models`
--
```

```
INSERT INTO `Models` (`ModelID`, `ManufacturerID`, `Name`, `Year`) VALUES
(1, 1, 'ЗА3-1102 Таврия', 1989),
(2, 1, 'ЗА3-965 Запорожець', 1960),
(3, 2, 'Богдан А099', 2005),
(4, 2, 'Богдан А106', 2010),
(5, 3, 'КрАЗ-6322', 1994);
```

```
-----
```

```
--
-- Структура таблиці `Users`
--
```

```
CREATE TABLE `Users` (
  `UserID` int NOT NULL,
  `Username` varchar(50) NOT NULL,
  `Password` varchar(255) NOT NULL,
  `Email` varchar(100) NOT NULL,
  `CreatedAt` timestamp NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
```

```
--
-- Дамп даних таблиці `Users`
--
```

```
INSERT INTO `Users` (`UserID`, `Username`, `Password`, `Email`, `CreatedAt`) VALUES
(1, 'admin', 'admin', 'admin@gmail.com', '2024-04-03 17:34:44');
```

```
-----
```

```
--
-- Структура для представлення `car_models_manufacturers`
--
```

```
DROP TABLE IF EXISTS `car_models_manufacturers`;
```

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`%` SQL SECURITY DEFINER VIEW
`car_models_manufacturers` AS SELECT `CarID` AS `CarID`, `Color` AS `Color`, `Price` AS
`Price`, `models`.`Name` AS `ModelName`, `models`.`Year` AS `Year`, `manufacturers`.`Name` AS
`ManufacturerName` FROM ((`cars` join `models` on((`ModelID` = `models`.`ModelID`))) join
`manufacturers` on((`models`.`ManufacturerID` = `manufacturers`.`ManufacturerID`))) ;
```

```
--
-- Індекси збережених таблиць
--
```

```
--
-- Індекси таблиці `Cars`
--
```

```
ALTER TABLE `Cars`
  ADD PRIMARY KEY (`CarID`),
  ADD KEY `ModelID` (`ModelID`);
```

```
--
-- Індекси таблиці `Features`
--
```

```
ALTER TABLE `Features`
  ADD PRIMARY KEY (`FeatureID`),
  ADD KEY `CarID` (`CarID`);
```

```

--
-- Індекси таблиці `Manufacturers`
--
ALTER TABLE `Manufacturers`
  ADD PRIMARY KEY (`ManufacturerID`);

--
-- Індекси таблиці `Models`
--
ALTER TABLE `Models`
  ADD PRIMARY KEY (`ModelID`),
  ADD KEY `ManufacturerID` (`ManufacturerID`);

--
-- Індекси таблиці `Users`
--
ALTER TABLE `Users`
  ADD PRIMARY KEY (`UserID`),
  ADD UNIQUE KEY `Username` (`Username`),
  ADD UNIQUE KEY `Email` (`Email`);

--
-- AUTO_INCREMENT для збережених таблиць
--

--
-- AUTO_INCREMENT для таблиці `Cars`
--
ALTER TABLE `Cars`
  MODIFY `CarID` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;

--
-- AUTO_INCREMENT для таблиці `Features`
--
ALTER TABLE `Features`
  MODIFY `FeatureID` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;

--
-- AUTO_INCREMENT для таблиці `Manufacturers`
--
ALTER TABLE `Manufacturers`
  MODIFY `ManufacturerID` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT для таблиці `Models`
--
ALTER TABLE `Models`
  MODIFY `ModelID` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;

--
-- AUTO_INCREMENT для таблиці `Users`
--
ALTER TABLE `Users`
  MODIFY `UserID` int NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;

--
-- Обмеження зовнішнього ключа збережених таблиць
--

--
-- Обмеження зовнішнього ключа таблиці `Cars`
--
ALTER TABLE `Cars`

```

```
ADD CONSTRAINT `cars_ibfk_1` FOREIGN KEY (`ModelID`) REFERENCES `Models` (`ModelID`);

--
-- Обмеження зовнішнього ключа таблиці `Features`
--
ALTER TABLE `Features`
  ADD CONSTRAINT `features_ibfk_1` FOREIGN KEY (`CarID`) REFERENCES `Cars` (`CarID`);

--
-- Обмеження зовнішнього ключа таблиці `Models`
--
ALTER TABLE `Models`
  ADD CONSTRAINT `models_ibfk_1` FOREIGN KEY (`ManufacturerID`) REFERENCES `Manufacturers`
  (`ManufacturerID`);
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

ДОДАТОК Б – ЛІСТИНГ ВЕБ-САЙТУ

```

<?php
session_start();
require_once "database.php";
connectDB();

if ($_SERVER["REQUEST_METHOD"] == "POST") {

    if (!empty($_POST["username"]) && !empty($_POST["password"])) {
        $username = $_POST["username"];
        $password = $_POST["password"];
        $query = "SELECT * FROM users WHERE username = ? AND password = ?";
        if ($stmt = $conn->prepare($query)) {
            $stmt->bind_param("ss", $username, $password);
            $stmt->execute();
            $result = $stmt->get_result();
            if ($result->num_rows == 1) {

                $_SESSION["username"] = $username;
                header("Location: dashboard.php");
                exit;
            } else {
                $error_message = "Неправильний логін чи пароль.";
            }
        } else {
            $error_message = "Помилка виконання запиту.";
        }
    } else {
        $error_message = "Будь ласка, заповніть всі поля.";
    }
}
?>

<!doctype html>
<html class="no-js" lang="en">

    <head>
        <!-- meta data -->
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">

        <!-- The above 3 meta tags *must* come first in the head; any other head
        content must come *after* these tags -->
        <link
href="https://fonts.googleapis.com/css?family=Poppins:100,100i,200,200i,300,300i,400,
400i,500,500i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">

```

```
<link href="https://fonts.googleapis.com/css?family=Rufina:400,700" rel="stylesheet">

<!-- title of site -->
<title>Cars Catalog</title>

<!-- For favicon png -->
<link rel="shortcut icon" type="image/icon" href="assets/logo/favicon.png"/>

<!--font-awesome.min.css-->
<link rel="stylesheet" href="assets/css/font-awesome.min.css">

<!--linear icon css-->
<link rel="stylesheet" href="assets/css/linearicons.css">

<!--flaticon.css-->
<link rel="stylesheet" href="assets/css/flaticon.css">

<!--animate.css-->
<link rel="stylesheet" href="assets/css/animate.css">

<!--owl.carousel.css-->
<link rel="stylesheet" href="assets/css/owl.carousel.min.css">
<link rel="stylesheet" href="assets/css/owl.theme.default.min.css">

<!--bootstrap.min.css-->
<link rel="stylesheet" href="assets/css/bootstrap.min.css">

<!-- bootsnav -->
<link rel="stylesheet" href="assets/css/bootsnav.css" >

<!--style.css-->
<link rel="stylesheet" href="assets/css/style.css">

<!--responsive.css-->
<link rel="stylesheet" href="assets/css/responsive.css">

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries --
>
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

<!--[if lt IE 9]>
    <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
    <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->

</head>
<body>

    <!--[if lte IE 9]>
```



```

        <p class="browserupgrade">You are using an <strong>outdated</strong>
browser. Please <a href="https://browsehappy.com/">upgrade your browser</a> to
improve your experience and security.</p>
    <![endif]-->

    <!--welcome-hero start -->
    <section id="home" class="welcome-hero">

        <!-- top-area Start -->
        <div class="top-area">
            <div class="header-area">
                <!-- Start Navigation -->
                <nav class="navbar navbar-default bootsnav navbar-sticky navbar-
scrollspy" data-minus-value-desktop="70" data-minus-value-mobile="55" data-
speed="1000">

                    <div class="container">

                        <!-- Start Header Navigation -->
                        <div class="navbar-header">
                            <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target="#navbar-menu">
                                <i class="fa fa-bars"></i>
                            </button>
                            <a class="navbar-brand"
href="index.php">cars<span></span></a>

                        </div><!--/.navbar-header-->
                        <!-- End Header Navigation -->

                        <!-- Collect the nav links, forms, and other content for
toggling -->
                        <div class="collapse navbar-collapse menu-ui-design"
id="navbar-menu">
                            <ul class="nav navbar-nav navbar-right" data-
in="fadeInDown" data-out="fadeOutUp">
                                <li ><a href="index.php">Головна</a></li>
                                <li ><a href="index.php">Автомобілі</a></li>
                                <li ><a href="index.php">Особливості</a></li>

                                <li> <a href="login.php">Адмін</a></li>
                            </ul><!--/.nav -->
                        </div><!-- /.navbar-collapse -->
                    </div><!--/.container-->
                </nav><!--/nav-->
                <!-- End Navigation -->
            </div><!--/.header-area-->
            <div class="clearfix"></div>

        </div><!-- /.top-area-->

```

```

        <!-- top-area End -->

        <br> <br> <br> <br> <br> <br> <br> <br> <br> <br>
        <br> <br> <br> <br> <br> <br> <br> <br> <br> <br>
        <br> <br> <br> <br> </section><!--/.service-->
    <!--service end-->
    <div class="login-container">
<h2>Авторизація</h2>
<?php if (isset($error_message)) { ?>
    <div class="error-message"><?php echo $error_message; ?></div>
<?php } ?>
    <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>"
method="post">
    <div class="form-group">
        <label for="username">Логін:</label>
        <input type="text" id="username" name="username" class="form-control"
required>
    </div>
    <div class="form-group">
        <label for="password">Пароль:</label>
        <input type="password" id="password" name="password" class="form-control"
required>
    </div>
    <button type="submit" class="btn btn-primary">Увійти</button>
    </form>
</div>

    <!--contact start-->
    <footer id="contact" class="contact">
        <div class="container">

            <div class="footer-copyright">
                <div class="row">
                    <div class="col-sm-6">
                        <p>
                            &copy; Авторські права належать Передистому
Дмитру</a>.
                        </p><!--/p-->
                    </div>
                    <div class="col-sm-6">
                        <div class="footer-social">
                            <a href="#"><i class="fa fa-facebook"></i></a>
                            <a href="#"><i class="fa fa-instagram"></i></a>
                            <a href="#"><i class="fa fa-linkedin"></i></a>
                            <a href="#"><i class="fa fa-pinterest-p"></i></a>
                            <a href="#"><i class="fa fa-behance"></i></a>
                        </div>
                    </div>
                </div>
            </div><!--/.footer-copyright-->
        </div><!--/.container-->

```

```

        <div id="scroll-Top">
            <div class="return-to-top">
                <i class="fa fa-angle-up " id="scroll-top" data-toggle="tooltip"
data-placement="top" title="" data-original-title="Back to Top" aria-
hidden="true"></i>
            </div>

        </div><!--/.scroll-Top-->

</footer><!--/.contact-->
<!--contact end-->
<!-- Include all js compiled plugins (below), or include individual files as
needed -->

<script src="assets/js/jquery.js"></script>

<!--modernizr.min.js-->
<script
src="https://cdnjs.cloudflare.com/ajax/libs/modernizr/2.8.3/modernizr.min.js"></scrip
t>

<!--bootstrap.min.js-->
<script src="assets/js/bootstrap.min.js"></script>

<!-- bootsnav js -->
<script src="assets/js/bootsnav.js"></script>

<!--owl.carousel.js-->
<script src="assets/js/owl.carousel.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-
easing/1.4.1/jquery.easing.min.js"></script>

<!--Custom JS-->
<script src="assets/js/custom.js"></script>
</body>
</html>

<!doctype html>
<html class="no-js" lang="en">

<head>
    <!-- meta data -->
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head
content must come *after* these tags -->
    <script>

```

```

function showMessage() {
    alert("Нічого не знайдено.");
}
</script>
<!--font-family-->
<link
href="https://fonts.googleapis.com/css?family=Poppins:100,100i,200,200i,300,300i,400,
400i,500,500i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Rufina:400,700"
rel="stylesheet">

<!-- title of site -->
<title>Cars Catalog</title>

<!-- For favicon png -->
<link rel="shortcut icon" type="image/icon" href="assets/logo/favicon.png"/>

<!--font-awesome.min.css-->
<link rel="stylesheet" href="assets/css/font-awesome.min.css">

<!--linear icon css-->
<link rel="stylesheet" href="assets/css/linearicons.css">

<!--flaticon.css-->
<link rel="stylesheet" href="assets/css/flaticon.css">

<!--animate.css-->
<link rel="stylesheet" href="assets/css/animate.css">

<!--owl.carousel.css-->
<link rel="stylesheet" href="assets/css/owl.carousel.min.css">
<link rel="stylesheet" href="assets/css/owl.theme.default.min.css">

<!--bootstrap.min.css-->
<link rel="stylesheet" href="assets/css/bootstrap.min.css">

<!-- bootsnav -->
<link rel="stylesheet" href="assets/css/bootsnav.css" >

<!--style.css-->
<link rel="stylesheet" href="assets/css/style.css">

<!--responsive.css-->
<link rel="stylesheet" href="assets/css/responsive.css">

<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media
queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

```



```

// Подготовка и выполнение SQL-запроса
$sql = "SELECT Manufacturers.Name AS ManufacturerName, Models.Name AS
ModelName, Models.Year
        FROM Models
        INNER JOIN Manufacturers ON Models.ManufacturerID =
Manufacturers.ManufacturerID
        WHERE Models.Name = ?";

$stmt = $conn->prepare($sql);
$stmt->bind_param("s", $modelName);
$stmt->execute();
$result = $stmt->get_result();

if ($result->num_rows > 0) {
    echo '<div class="container">';
    echo '<div class="row">';
    echo '<div class="col-md-12">';
    echo '<div class="model-results">';
    echo '<h2>Результати пошуку:</h2>';
    echo '<table class="table">';
    echo '<thead>';
    echo '<tr>';
    echo '<th>Виробник</th>';
    echo '<th>Модель</th>';
    echo '<th>Рік</th>';
    echo '</tr>';
    echo '</thead>';
    echo '<tbody>';
    while($row = $result->fetch_assoc()) {
        echo '<tr>';
        echo '<td>' . $row['ManufacturerName'] . '</td>';
        echo '<td>' . $row['ModelName'] . '</td>';
        echo '<td>' . $row['Year'] . '</td>';
        echo '</tr>';
    }
    echo '</tbody>';
    echo '</table>';
    echo '</div>';
    echo '</div>';
    echo '</div>';
    echo '</div>';
} else {
    echo '<script>';
    echo 'showMessage();';
    echo '</script>';
}

$stmt->close();
}

```

?>

```

</section><!--/.welcome-hero-->
<!--welcome-hero end -->

<!--service start -->
<section id="service" class="service">
  <div class="container">
    <div class="service-content">
      <div class="row">
        <div class="col-md-4 col-sm-6">
          <div class="single-service-item">
            <div class="single-service-icon">
              <i class="flaticon-car"></i>
            </div>
            <h2><a href="#">Найбільший каталог <span>
автомобілів</span></a></h2>

              </div>
            </div>
          <div class="col-md-4 col-sm-6">
            <div class="single-service-item">
              <div class="single-service-icon">
                <i class="flaticon-car-repair"></i>
              </div>
              <h2><a href="#">Нескінченна гарантійна
підтримка</a></h2>

                </div>
              </div>
            <div class="col-md-4 col-sm-6">
              <div class="single-service-item">
                <div class="single-service-icon">
                  <i class="flaticon-car-1"></i>
                </div>
                <h2><a href="#">Онлайн підтримка</a></h2>

                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</section><!--/.service-->
<!--service end-->

<!--new-cars start -->
<section id="new-cars" class="new-cars">
  <div class="container">

```



```

        <div class="section-header">
            <p>Наші <span>останні</span> оновлення</p>
            <h2>Нові автомобілі</h2>
        </div><!--/.section-header-->
    <?php
require_once "database.php";
connectDB();

$sql = "SELECT Cars.CarID, Cars.Color, Cars.Price, Models.Name AS ModelName,
Models.Year, Manufacturers.Name AS ManufacturerName
FROM Cars
JOIN Models ON Cars.ModelID = Models.ModelID
JOIN Manufacturers ON Models.ManufacturerID = Manufacturers.ManufacturerID";

$result = $conn->query($sql);
?>

<div class="new-cars-content">
    <div class="owl-carousel owl-theme" id="new-cars-carousel">
        <?php
        if ($result->num_rows > 0) {
            while ($row = $result->fetch_assoc()) {
                echo '<div class="new-cars-item">';
                echo '<div class="single-new-cars-item">';
                echo '<div class="row">';
                echo '<div class="col-md-7 col-sm-12">';
                echo '<div class="new-cars-img">';
                $imagePath = 'assets/images/new-cars-model/' . $row["CarID"] .
'.png';

                echo '<img src="" . $imagePath . "" alt="img"/>';
                echo '</div>';
                echo '</div>';
                echo '<div class="col-md-5 col-sm-12">';
                echo '<div class="new-cars-txt">';
                echo '<h2><a href="#">' . $row["ManufacturerName"] . '</a></h2>';
                echo 'Модель: <span>' . $row["ModelName"] . '</span><br>';
                echo 'Рік випуску: <span>' . $row["Year"] . '</span><br>';
                echo '<p>' . $row["Color"] . ', ' . $row["Price"] . ' грн </p>';

                echo '</div>';
                echo '</div>';
                echo '</div>';
                echo '</div>';
                echo '</div>';
            }
        } else {
            echo '<p>Немає доступних автомобілів.</p>';
        }

        $result->free_result();

```

```

    ?>
  </div><!--/#new-cars-carousel-->
</div><!--/.new-cars-content-->
  </div><!--/.container-->

</section><!--/.new-cars-->
<!--new-cars end -->

<!--featured-cars start -->
<section id="featured-cars" class="featured-cars">
<div class="container">
  <div class="section-header">
    <h2>Особливості автомобілей</h2>
  </div><!--/.section-header-->
  <div class="featured-cars-content">
    <div class="row">
      <?php

        require_once "database.php";
        connectDB();

        $sql = "SELECT Cars.CarID, Cars.Color, Cars.Price, Models.Name AS
ModelName, Models.Year, Manufacturers.Name AS ManufacturerName, Features.FeatureName
FROM Cars
JOIN Models ON Cars.ModelID = Models.ModelID
JOIN Manufacturers ON Models.ManufacturerID =
Manufacturers.ManufacturerID
JOIN Features ON Cars.CarID = Features.CarID";

        $result = $conn->query($sql);
        if ($result->num_rows > 0) {
          while ($row = $result->fetch_assoc()) {
            echo '<div class="col-lg-3 col-md-4 col-sm-6">';
            echo '<div class="single-featured-cars">';
            echo '<div class="featured-img-box">';
            echo '<div class="featured-cars-img">';
            $imagePath = 'assets/images/featured-cars/' . $row["CarID"] .
'.png';

            echo '';
            echo '</div>'; // Закриваєм featured-cars-img
            echo '<div class="featured-model-info">';
            echo '<p>';
            echo 'model: ' . $row["Year"];
            echo '<span class="featured-mi-span"> ' . $row["FeatureName"]
. '</span>';

            echo '</p>';
            echo '</div>';
            echo '</div>';
            echo '<div class="featured-cars-txt">';

```

```

        echo '<h2><a href="#">' . $row["ManufacturerName"] . ' ' .
$row["ModelName"] . '</a></h2>';
        echo '<h3>$' . $row["Price"] . '</h3>';
        echo '<p>' . $row["Color"] . '</p>';
        echo '</div>';
        echo '</div>';
        echo '</div>';
    }
} else {
    echo '<div class="col-lg-12"><p>Немає доступних
автомобілів.</p></div>';
}

$result->free_result();
?>
</div><!--/.row-->
</div><!--/.featured-cars-content-->
</div><!--/.container-->
</section><!--/.featured-cars-->
<!--brand strat -->
<section id="brand" class="brand">
    <div class="container">
        <div class="brand-area">
            <div class="owl-carousel owl-theme brand-item">
                <div class="item">
                    <a href="#">
                        
                    </a>
                </div><!--/.item-->
                <div class="item">
                    <a href="#">
                        
                    </a>
                </div><!--/.item-->
                <div class="item">
                    <a href="#">
                        
                    </a>
                </div><!--/.item-->
                <div class="item">
                    <a href="#">
                        
                    </a>
                </div><!--/.item-->
                <div class="item">

```

```

        <a href="#">
            
        </a>
    </div><!--/.item-->

    <div class="item">
        <a href="#">
            
        </a>
    </div><!--/.item-->
</div><!--/.owl-carousel-->
</div><!--/.clients-area-->

</div><!--/.container-->

</section><!--/brand-->
<!--brand end -->

<!--blog start -->
<section id="blog" class="blog"></section><!--/.blog-->

<footer id="contact" class="contact">
    <div class="container">

        <div class="footer-copyright">
            <div class="row">
                <div class="col-sm-6">
                    <p>
&copy; Авторські права належать Передистому Дмитру</a>.
                    </p><!--/p-->
                </div>
                <div class="col-sm-6">
                    <div class="footer-social">
                        <a href="#"><i class="fa fa-facebook"></i></a>
                        <a href="#"><i class="fa fa-instagram"></i></a>
                        <a href="#"><i class="fa fa-linkedin"></i></a>
                        <a href="#"><i class="fa fa-pinterest-p"></i></a>
                        <a href="#"><i class="fa fa-behance"></i></a>
                    </div>
                </div>
            </div>
        </div><!--/.footer-copyright-->
    </div><!--/.container-->

    <div id="scroll-Top">
        <div class="return-to-top">

```

```

        <i class="fa fa-angle-up " id="scroll-top" data-toggle="tooltip"
data-placement="top" title="" data-original-title="Back to Top" aria-
hidden="true"></i>
        </div>

</div><!--/.scroll-Top-->

</footer><!--/.contact-->
<!--contact end-->

<!-- Include all js compiled plugins (below), or include individual files as
needed -->

<script src="assets/js/jquery.js"></script>

<!--modernizr.min.js-->
<script
src="https://cdnjs.cloudflare.com/ajax/libs/modernizr/2.8.3/modernizr.min.js"></scrip
t>

<!--bootstrap.min.js-->
<script src="assets/js/bootstrap.min.js"></script>

<!-- bootsnav js -->
<script src="assets/js/bootsnav.js"></script>

<!--owl.carousel.js-->
<script src="assets/js/owl.carousel.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-
easing/1.4.1/jquery.easing.min.js"></script>

<!--Custom JS-->
<script src="assets/js/custom.js"></script>

</body>

</html>

```

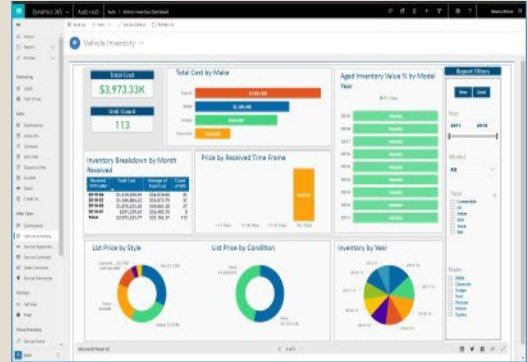
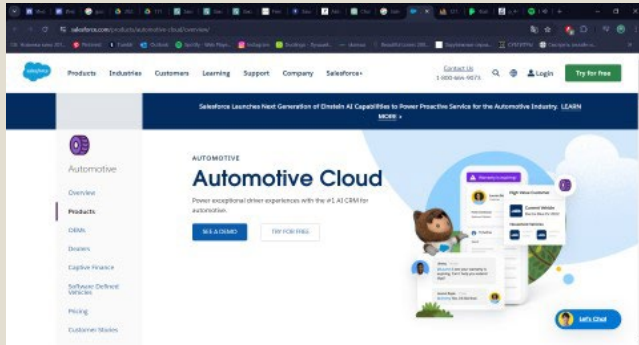
ТЕМА: РОЗРОБКА ТА АНАЛІЗ ВЕБ-СЕРВІСУ ДЛЯ ОПТИМІЗАЦІЇ ПРОЦЕСІВ В АВТОМОБІЛЬНОМУ БІЗНЕСІ

- Актуальність роботи: Огляд проблеми та актуальності розвитку веб-сервісів у сфері автомобільного бізнесу дозволяє розкрити ключові виклики, з якими зіштовхнулися підприємства цієї галузі та необхідність впровадження інноваційних технологій для їх вирішення. Основні проблеми включають управління інвентарем, оптимізацію процесів продажу та обслуговування авто, ефективне ведення CRM, а також конкуренцію у цифровому просторі.
- Основною метою є створення веб-сервісу, що сприятиме оптимізації процесів у автомобільному бізнесі через інтеграцію із системами управління.
- Предметом дослідження є система оптимізації процесів у автомобільному бізнесі за допомогою розробки веб-сервісу
- Завдання, які вирішувались для досягнення поставленої мети:
- Розробити веб-сервіс, що інтегрується з існуючими системами управління запасами, логістики та клієнтськими базами.
- Оптимізувати процеси продажу, обслуговування та ведення CRM через новий веб-сервіс.
- Забезпечити високий рівень масштабованості та ефективності веб-сервісу за допомогою сучасних веб-технологій.
- Наукова новизна полягає у розробці веб-сервісу, який інтегрується з системами управління автомобільного бізнесу для оптимізації процесів управління запасами, логістики та взаємодії з клієнтами. Застосування сучасних веб-технологій для створення масштабованої та ефективної платформи є значним внеском у вдосконалення процесів у цій галузі.

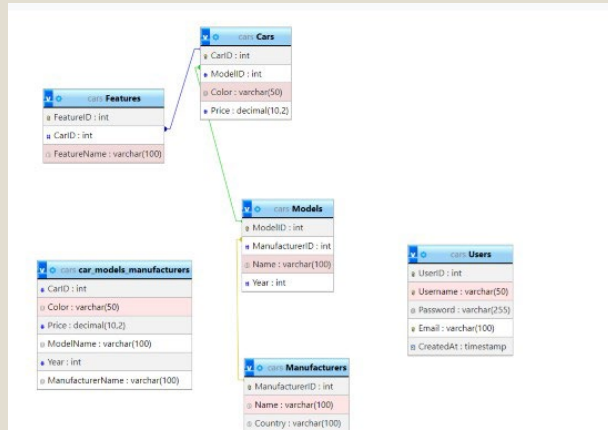
Огляд існуючих рішень

◦ Salesforce Automotive Cloud

◦ Microsoft Dynamics 365 for Automotive



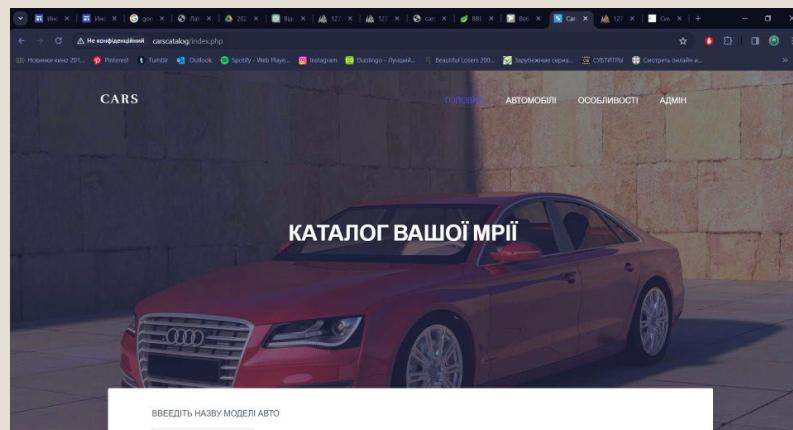
ER-діаграма



Типи зв'язків таблиць БД

Таблиця 1	Тип зв'язку	Таблиця 2
Автомобілі	1:N	Моделі
Особливості	1:N	Автомобілі
Моделі	1:N	Виробники


МЕНЮ САЙТУ



Сторінка «Автомобілі»

Наші останні Оновлення








Нові Автомобілі



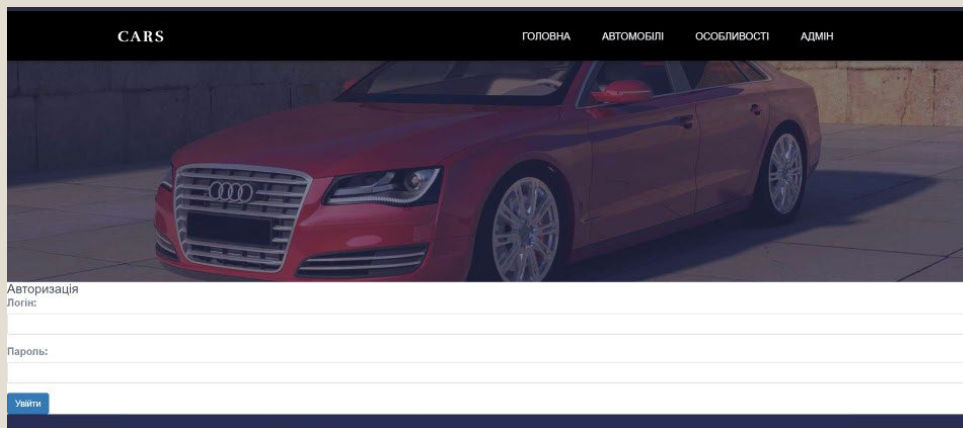
ЗАЗ

Модель: ЗАЗ-965 Запорожець
Рік випуску: 1960
Білий, 6000.00грн

Сторінка «Особливості»

			
Model: 1989 Кожаний	Model: 1989 Електрогидропідсил. Керма	Model: 1989 ABS	Model: 1960 Круїз-Контроль
ЗАЗ ЗАЗ-1102 Таврія \$5000.00 Червоний	ЗАЗ ЗАЗ-1102 Таврія \$5000.00 Червоний	ЗАЗ ЗАЗ-1102 Таврія \$4500.00 Синій	ЗАЗ ЗАЗ-965 Запорожець \$6000.00 Білий
			

Сторінка «Авторизація»



CARS ГОЛОВНА АВТОМОБІЛІ ОСОБЛИВОСТІ АДМІН

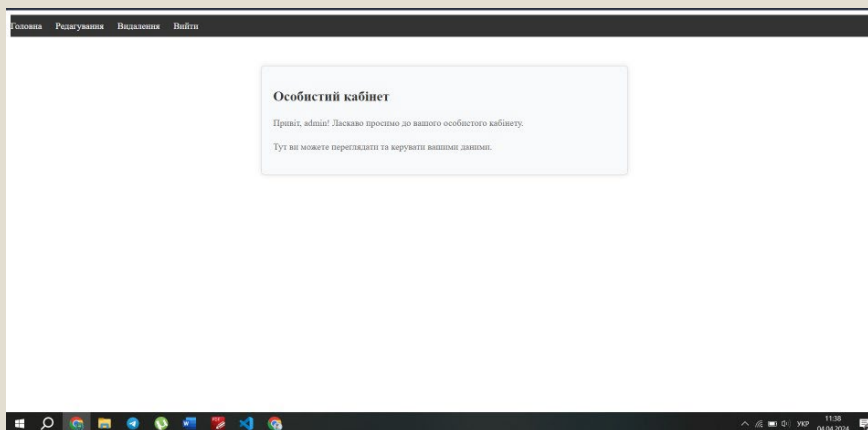
Авторизація

Логін:

Пароль:

Увійти

Сторінка «Особистий кабінет»



Головна Результати Відомості Вийти

Особистий кабінет

Привіт, адміні! Заскано просимо до вашого особистого кабінету.
Тут ви можете переглядати та керувати вашими даними.

11:38 04.04.2014

ВИСНОВКИ

- У цій роботі описано ключові аспекти розробки інформаційної системи для автомобільного бізнесу, включаючи аналіз, проектування і реалізацію веб-сервісу, створення бази даних і планування користувацького інтерфейсу та серверної частини. Важливість оптимізації процесів через веб-сервіси, підвищення ефективності роботи автосалонів і краще управління запасами обговорюється у першому розділі. Другий розділ демонструє методологію проектування бази даних, акцентуючи на цілісності даних, безпеці та гнучкості системи. Третій розділ описує створення веб-сайту з ефективним дизайном і серверними технологіями, які забезпечують стабільну роботу системи. Робота також підкреслює значення взаємодії з зацікавленими сторонами для задоволення бізнес-вимог, тестування і налагодження для якості та надійності системи. Завершальна частина роботи вказує на потенціал для подальшого масштабування та інтеграції нових технологічних рішень, що дозволяє адаптувати систему до ринкових умов та підтримувати лідерство в автомобільному бізнесі.