

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ

к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "бакалавр"

на тему:

«Розробка інформаційної підсистеми каршерінгу автомобілей»

Виконала: студентка 2147ст групи

Стецюк В.В.

(підпис)

Керівник роботи:

к.т.н., доцент

(посада, науковий ступінь вчене звання)

Гайдаєнко О.В.

(підпис)

м. Миколаїв – 2024 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій
Спеціальність 122 "Комп'ютерні науки"
Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми

к.т.н, доц. Гайда А.Ю.

" ____ " ____ 2024 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття ступеня вищої освіти "бакалавр"

Студенту Стецюк Валерія Володимирівна

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка інформаційної підсистеми каршерінгу автомобілей

Керівник роботи Гайдаєнко Оксана Володимирівна

Затверджені наказом ректора № 243-уч від "18" березня 2024 року.

2. Термін подання роботи: 20 червня 2024 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем,

4. Перелік питань, що належать до розробки (найменування розділів) 1. Розробка концепції ІС каршерінгу автомобілей. 2. Розробка проекту ІС каршерінгу автомобілей. 3. Реалізація проекту ІС. 4. Охорона праці. Висновки. Додатки.

5. Перелік презентаційних матеріалів Мета та об'єкт дослідження. Архітектура та етапи розробки інформаційної підсистеми. Структура вимог до системи. Діаграма варіантів використання. Діаграма варіантів використання. Логічна та фізична модель. Результат роботи. Технології, які використовувались для розробки сайта.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	К.т.н., доц. Гайдаєнко О.В.		
2	К.т.н., доц. Гайдаєнко О.В.		
3	К.т.н., доц. Гайдаєнко О.В.		
4	Ст. викл. Гурець Н.В.		

7. Дата видачі завдання 05 лютого 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Затвердження теми роботи	До 15.02.24	
2	Планування робіт	До 20.02.24	
3	Проведення аналізу предметної області	До 01.03.24	
4	Розробка концепції та аналіз вимог	До 10.03.24	
5	Розробка технічного завдання	До 15.03.24	
6	Вирішення питань охорони праці	До 29.03.24	
7	Розробка додатку	До 20.04.24	
8	Тестування додатку	До 25.05.24	
9	Оформлення та здача пояснювальної записки та файлів розробленого проекту	До 07.06.24	
10	Захист	18.06.24 – 20.06.24	

Студент

_____ (підпис)

_____ (ПІБ)

Керівник роботи

_____ (підпис)

_____ (ПІБ)

АНОТАЦІЯ

Об'єктом дослідження є фірма. Ціль роботи – підвищення ефективності роботи цієї фірми.

Метою роботи є розробка та впровадження інформаційної підсистеми каршерінгу автомобілів, яка дозволить удосконалити процеси управління автопарком, забезпечити зручність та швидкість обслуговування клієнтів, а також зменшити витрати та ризики.

У процесі виконання випускної кваліфікаційної роботи здійснена постановка задачі, визначені основні способи вирішення поставленого завдання, виконаний аналіз вимог на розробку.

Кваліфікаційна робота складається зі вступу, чотирьох розділів, висновку та списку використаних джерел та трьох додатків.

ABSTRACT

The object of work research is the firm. The goal of the work is to improve the efficiency of this company.

The purpose of the work is the development and implementation of a car sharing information system that will improve fleet management processes, ensure convenience and speed of customer service, and reduce costs and risks.

In the process of completing the final qualification work, the task was set, the main methods of solving the task were determined, and the development requirements were analyzed.

The qualification work consists of an introduction, four chapters, a conclusion, a list of used sources and three appendices.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1. Розробка концепції ІС каршерінгу автомобілей	9
1.1 Опис об'єкта дослідження та аналіз предметної області.....	9
1.2 Формування вимог до системи.....	14
1.3 Розробка концепції ІС	15
РОЗДІЛ 2. Проектування інформаційної підсистеми каршерінгу автомобілей.....	21
2.1 Рішення щодо інформаційного забезпечення.....	21
2.2 Рішення щодо програмного забезпечення	26
РОЗДІЛ 3. Розробка інформаційної підсистеми.....	30
РОЗДІЛ 4. Охорона праці	41
4.1 Умови роботи.....	41
4.2 Нормативно-правова база	43
4.3 небезпечні та шкідливі фактори робочого місця	44
ВИСНОВКИ	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49
ДОДАТКИ	
ДОДАТОК А.....	51
ДОДАТОК Б	57
ДОДАТОК В.....	60

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ІС – інформаційна система

БД – база даних

СУРБД – система управління реляційними базами

ВСТУП

Класифікована робота виконана по темі «Розробка інформаційної підсистеми каршерінгу автомобілей». Актуальність роботи складається у зростанню популярності каршерінгу як зручного та швидкого способу переміщення у великих містах клієнтів, для вирішення своїх поставлених задач.

Об'єктом дослідження є фірма. Ціль роботи – підвищення ефективності роботи цієї фірми каршерінгу.

Задача – розробка та впровадження інформаційної системи каршерінгу автомобілів, яка дозволить удосконалити процеси управління автопарком, забезпечити зручність та швидкість обслуговування клієнтів, а також зменшити витрати та ризики.

В Розділі 1 описаний об'єкт дослідження, розглянуті функціональні обов'язки робітників. Проведений аналіз предметної області в обробці інформації по системі управління, де знайдено недоліки на основі яких визначено завдання. Для розробки формулюємо вимоги до системи, на основі яких розробляється концепція, яка визначає основні положення складу функціональних задач для досягнення поставлених цілей, програмні ресурси використані для реалізації проекту і технічні засоби необхідні для його експлуатації.

В Розділі 2 виконана розробка проектної систем. Спроектowana БД, розроблена концептуальна, логічна та фізична моделі інформаційної системи. В математичному забезпеченні представлено алгоритм рішення задач, виконана реалізації програмних ресурсів, які дозволять досягнути поставлених цілей і описані технічні ресурси, які будуть використані для вирішення цих задач.

В Розділі 3 розроблені інструктивно-методичні матеріали по експлуатації системи і приведені результати обробки.

Реалізація проекту вирішила поставлену ціль – підвищення ефективності діяльності фірми у сфері каршерінгу та може служити основою для подальшого розвитку та впровадження інформаційних систем у цьому сегменті ринку.

РОЗДІЛ 1 Розробка концепції ІС каршерінгу автомобілей

1.1 Опис об'єкта дослідження та аналіз предметної області

У розвинених країнах спостерігається зростаюча тенденція до зменшення використання особистого автотранспорту. Хоча не всі водії поділяють цю ідею і продовжують користуватися власними автомобілями, у більшості таких країн спостерігається підвищений інтерес до альтернативних форм пересування, зокрема каршерінгу. Україна також не залишається осторонь цього тренду, і у великих містах спостерігається активний розвиток каршерінгових послуг, які знаходять популярність серед місцевого населення.

Каршерінг - це короткострокова оренда авто. Замовивши цю послугу, ви оплачуєте тільки той час, який використовували транспортний засіб. Оплата стягується щохвилини, при цьому машиною можна користуватися тільки в межах певного регіону або навіть міського району. Для користування послугою в смартфон необхідно встановити спеціальний додаток, створити в ньому власний аккаунт для бронювання авто, потім скористатися автомобілем і оплатити каршерінг [1].

Навіть за короткий період його існування, виникли різноманітні його форми й варіанти:

Однорангові – передбачають надання прокату машин приватними особами або компаніями, в яких авто оренда не відноситься до пріоритетних видів бізнесу. Наприклад, транспортна фірма, може здавати вільні автомобілі в прокат на умовах каршерінга, через спеціальні онлайн-сервіси, де кооперуються як солідні компанії, так і приватні власники автомашин. У цьому випадку, роль каршерінг-компанії, як раз і грає той самий онлайн-сервіс.

Часткові – характеризуються спільним використанням авто, жителями одного району чи територіального об'єднання. Люди самостійно створюють парки з певного числа автомобілів для мінімізації витрат та шкідливих викидів від машин [2].

Фірма каршерінгу автомобілей – система каршерінгу автомобілів, що надає послуги користувачам для короткотермінової оренди автомобілів за годину або за день. Напрями діяльності цього об'єкта включають:

- Забезпечення доступу до автомобілів для оренди через мобільний додаток або веб-платформу;
- Управління флотом автомобілів, включаючи технічне обслуговування, технічний огляд, та підтримку безпеки;
- Визначення тарифів та умов оренди;
- Підтримка клієнтського сервісу та вирішення конфліктних ситуацій.

У фірми існує власна організаційна структура для управління об'єктом, який представлений на рисунку 1.1.

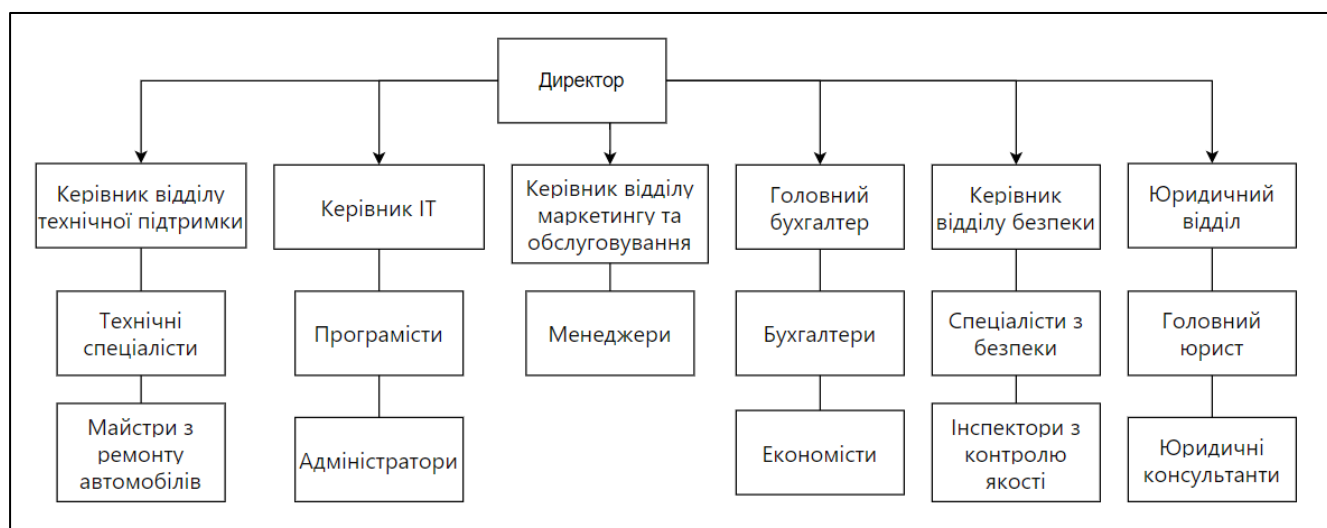


Рисунок 1.1 – Схема організаційної структури каршерінгу автомобілей

Директор у такій організаційній структурі зазвичай займається загальним керівництвом компанією. Його функції включають:

- Визначення стратегічні напрямки розвитку компанії і встановлення мету для всіх відділів та підрозділів;
- Участь у важливих рішеннях, що стосуються розвитку бізнесу, наприклад, щодо відкриття нових ринків, розширення послуг, реорганізації підрозділів;

- Створення стимулів для співробітників та забезпечення комунікації та співпрацю між відділами;
- Втручання у вирішення складних ситуацій та конфліктів в компанії.

Отже, директор відповідає за загальне керівництво компанією, встановлення стратегічних напрямків та забезпечення їх виконання.

Є бухгалтерський відділ, який очолює головний бухгалтер, займаючись керуванням і плануванням. В складі цього відділу є бухгалтера, які відповідають за реєстрацію та облік фінансових операцій компанії, таких як операції з оборотним та основним капіталом, операції з оплатою праці, операції з реалізації послуг. Також є економіст, який займається аналізом фінансових показників компанії, розробкою заходів для покращення фінансової стабільності та ефективності, контролем за рухом фінансових коштів в компанії, розрахунком податків, веденням розрахунків з постачальниками та клієнтами, а також вивченням та впровадженням ефективних фінансових стратегій.

Отже, бухгалтерський відділ забезпечує правильне ведення бухгалтерського обліку, складання фінансової звітності, аналіз фінансових показників та управління фінансовими ресурсами компанії з метою забезпечення її фінансової стабільності та успішного функціонування.

Функціональні обов'язки представленої структури управління об'єктом є:

Відділ технічної підтримки:

- Забезпечення технічної готовності автомобілів для користувачів;
- Виконання ремонтних робіт та технічних оглядів;
- Вирішення нагальних технічних проблем та аварій.

Відділ IT:

- Розробка та підтримка мобільного додатку та веб-платформи для замовлення автомобілів;
- Вдосконалення функціоналу системи.

Відділ маркетингу та клієнтського обслуговування:

- Продаж та просування послуг каршерінгу;

- Обробка запитів та консультації клієнтів;
- Вивчення ринку та конкурентної ситуації.

Відділ безпеки та контролю якості:

- Забезпечення безпеки користувачів та автомобілів;
- Контроль якості надання послуг та технічного стану автомобілів;
- Розробка та впровадження стандартів безпеки та якості.

Юридичний відділ:

- Укладання та аналіз договорів з користувачами та партнерами;
- Надання юридичної підтримки у вирішенні конфліктів та спорів;
- Розробка та оновлення правил користування та політик конфіденційності;
- Моніторинг законодавства щодо автомобільного транспорту, договірних відносин, конкуренції тощо;
- Проведення навчань та інструктажів з правових аспектів для співробітників і керівництва компанії;
- Підготовка правових документів та рекомендацій для захисту інтересів компанії.

Предметна область – це множина всіх об’єктів, властивості яких і відношення між якими розглядаються в межах деякого дослідження або у процесі діяльності [3].

Аналіз предметної області [4] процесів обробки інформації дозволяє виявити проблеми, які не можуть підвищити ефективність. Ось кілька недоліків, які потрібно виправити:

Бронювання та доступність автомобілів: Недостатня точність або застаріла інформація про наявність машини може спричинити неприємності клієнтів.

Технічне обслуговування та безпека: Проблеми можуть виникнути у процесі передачі інформації про несправності або у процесі оновлення даних про технічне обслуговування.

Управління замовленнями та платежами: Через помилку в системі або недостатній захист від шахрайства може бути зчитування особистих даних.

Управління парком автомобілів: Можливі проблеми у виникненні моніторингу автомобілей, які користуються попитом у місті та поповнення ними парку.

В результаті проведеного аналізу виявляємо конкретні проблемні області та пропонуємо рішення для підвищення ефективності каршерінгу. Це впровадження нових технологій, покращення систем управління та обробки даних, а також оновлення програмного забезпечення та інтерфейсів для користувачів. На основі цього аналізу було сформульовано об'єкт автоматизації процесів управління, а саме: здача в оренду, формування документообігу по оренді автомобілей і бухгалтерського звіту та оплати праці.

Компанія включає в себе документообіг. Документообіг — це рух документів в установі від моменту створення або від одержання зі сторони до моменту передачі на зберігання до архіву [5]. Він охоплює всі види документів, що стосуються діяльності компанії, включаючи договори, звіти, листування, заявки та інші документи, які потрібні для ведення бізнесу.

Документообіг у каршерінгу зазвичай включає наступні основні елементи:

Вхідні документи:

- Заявки на оренду: Це документи, які заповнюють клієнти, які бажають орендувати автомобіль. Зазвичай в них вказується інформація про час і дату оренди, місцезнаходження, а також дані про клієнта.
- Договір оренди: Після того, як клієнт підтвердив заявку, йому пропонується укласти договір оренди, який містить умови оренди, права і обов'язки клієнта та компанії каршерінгу.
- Документи, що підтверджують особу та водійські права: Для завершення процесу оренди клієнт повинен надати свої документи, щоб підтвердити особу та право керування автомобілем.

Вихідні документи:

- Рахунки та квитанції: Після завершення оренди компанія каршерінгу виставляє рахунок клієнту за послуги, а також видавати квитанцію про оплату.
- Звіти по оренді: Клієнтам можуть надаватися звіти про використання автомобіля, включаючи інформацію про маршрути, час оренди, витрату пального та інші дані.

Це основні форми документів, що використовуються в документообігу каршерінгових компаній. Залежно від внутрішніх процесів та політики компанії, також можуть використовуватися інші форми документів.

Ціль проекту полягає у необхідності створення системи, яка підвищить ефективність. Аналіз показав маркетинг сьогодні працює з обслуговуванням, вони мають техніку, яка фіксує оброблену інформацію, веде документацію – це являється стримуючим фактором ефективності роботи фірми. Система по бухгалтерському звіту і створений рекламний сайт немає зв'язку між сайтом заявки каршерінгу автомобіля з бухгалтерією, не існує комплексу з розрахунком формування квитанцій на оплату.

На базі проведеного аналізу предметної області об'єкта розробляються вимоги до інформаційної системи каршерінгу.

1.2 Формування вимог до системи

На основі аналізу вказаних недоліків формуємо вимоги до системи [6]:

Вимоги користувачів:

- ✓ Інтуїтивний і зручний інтерфейс користувача для бронювання автомобілів і проведення оплати;
- ✓ Можливість швидкої та точної інформації про доступність автомобілів;
- ✓ Забезпечення конфіденційності особистої інформації та безпеки платежів.

Системні вимоги:

- ✓ Надійна та швидка робота серверів для забезпечення доступності системи;
- ✓ Запобігання витоку даних та забезпечення захисту від кібератак.

Функціональні вимоги:

- ✓ Оновлення інформації про доступність автомобілів в режимі реального часу;
- ✓ Розроблення АРМ по маркетингу.
- ✓ Оновлення бази даних про стан автомобілів та їх доступність в реальному часі;
- ✓ Забезпечення зв'язку між веб-сайтом для бронювання автомобілів і системою бухгалтерії для автоматичної генерації квитанцій та звітів.

Вимоги до технічних засобів:

- ✓ Забезпечення надійної мережі, яка дозволяє швидко та безперебійно обмінюватися даними між користувачами та серверами;
- ✓ Використання сучасних технологій для забезпечення безпеки та захисту від кібератак.

Атрибути якості:

- ✓ Надійність та стабільність системи для уникнення непередбачуваних збоїв;
- ✓ Висока швидкість роботи системи для забезпечення швидкого доступу користувачів;
- ✓ Підтримка та обслуговування системи для вирішення будь-яких проблем, що виникають.

Забезпечення виконання цих вимог допоможе у підвищенні ефективності каршерінгу та вирішенні виявлених проблемних областей.

На основі формувань вимог забезпечити інформаційну підтримку ухвалення рішень, спрямованих на розробку концепції ІС.

1.3 Розробка концепції ІС

На основі сформованих вимог, розроблена модель концепції ІС, яка представлена на рисунку 1.2 [7].

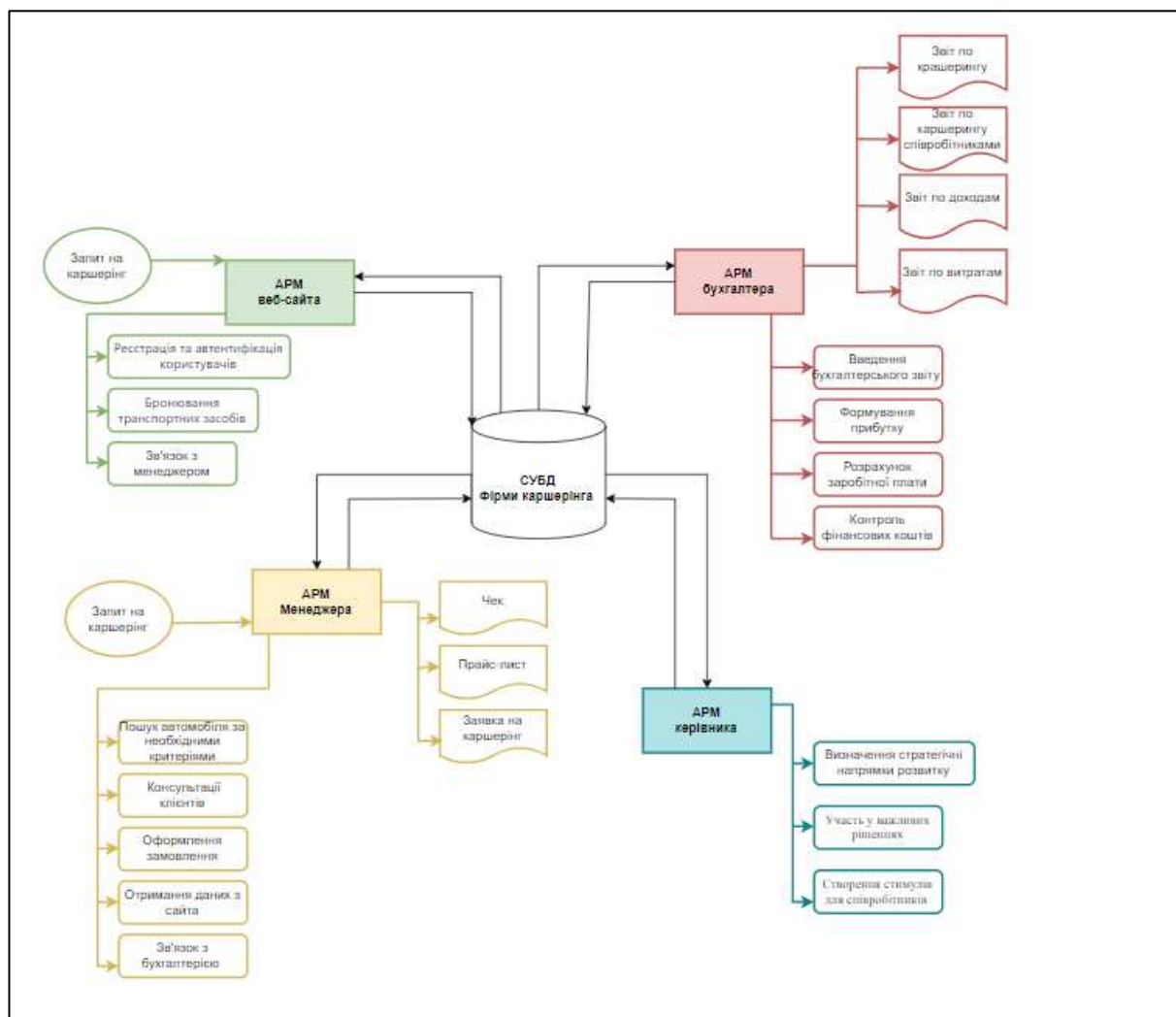


Рисунок 1.2 – Модель концепції ІС

Ціль проекту полягає у вдосконаленні бізнес-процесів управління об'єктом за допомогою впровадження сучасних інформаційних ресурсів та інформаційної системи. Це може включати наступні аспекти:

- 1) Підвищення ефективності управління: Використання інформаційної системи дозволить прискорити виконання завдань, зменшити кількість помилок та покращити контроль над виконанням завдань;
- 2) Покращення якості послуг або продукції: Система може допомогти в відстеженні якості, контролі за процесами виробництва або надання послуг, а також в оперативному реагуванні на можливі проблеми;

- 3) Підвищення рівня клієнтського сервісу: Допомога в автоматизації процесів обробки запитів клієнтів, наданні актуальної інформації про продукцію або послуги, а також в покращенні комунікації з клієнтами;
- 4) Зниження витрат та оптимізація ресурсів: Скорочення витрат та оптимізація процесу закупівель, управління складом, планування виробництва та інші аспекти управління ресурсами.

Інформаційна система каршерінгу автомобілей складається із армів:

АРМ веб-сайта вирішує наступні задачі:

Реєстрація та автентифікація користувачів: Користувачі можуть створювати облікові записи на веб-сайті, вказуючи особисті дані та контактну інформацію. Система повинна забезпечити безпечну автентифікацію користувачів шляхом введення логіна та пароля;

Бронювання транспортних засобів: Користувачі можуть переглядати доступні автомобілі, їх характеристики та доступність на певні дати та часи. Після вибору автомобіля користувач може здійснити бронювання, вказавши дату та час початку та завершення прокату. Система повинна забезпечити підтвердження бронювання та відображення інформації про неї в особистому кабінеті користувача;

Зв'язок з менеджером: Користувачі можуть звертатися до менеджера через спеціальну форму зворотнього зв'язку або чату. Звернення може стосуватися запитів щодо бронювання, допомоги в розв'язанні проблем або отримання додаткової інформації.

АРМ Менеджера реалізує наступні функції:

Пошук автомобіля за необхідними критеріями: Менеджер може використовувати пошукові фільтри для знаходження автомобілів за різними параметрами, такими як модель, тип кузова, доступність на певну дату та час, місцезнаходження тощо. Система повинна надавати результати пошуку в зручному форматі для подальшого оформлення замовлення;

Консультації клієнтів: Менеджер може проводити консультації для клієнтів з питань, пов'язаних з послугами каршерінгу, процесом бронювання, умовами

оренди, тарифами тощо. Зручність у взаємодії з клієнтами може бути забезпечена за допомогою чату, телефонного дзвінка або електронної пошти;

Оформлення замовлення: Після знаходження відповідного автомобіля менеджер може оформити замовлення в системі від імені клієнта, вказавши дату та час початку та завершення оренди, а також інші необхідні деталі;

Отримання даних з сайту: Менеджер отримує дані про бронювання, реєстрації користувачів та іншу інформацію з веб-сайту каршерінгу через спеціальний інтерфейс або звіти;

Зв'язок з бухгалтерією: Для вирішення фінансових питань, таких як виставлення рахунків, контроль оплати та інші, менеджер може мати можливість зв'язатися з бухгалтерією через спеціальні інструменти в системі або зв'язку на рівні електронної пошти або через внутрішній чат системи;

АРМ бухгалтера виконує такі функції:

Введення бухгалтерського звіту: Бухгалтер може вести різноманітні фінансові операції в систему, такі як витрати на обслуговування автомобілів, оплату пального, витрати на рекламу, оплату заробітної плати тощо;

Формування прибутку: Система може автоматично обробляти дані про витрати та доходи, щоб розрахувати чистий прибуток. Бухгалтер може переглядати звіти про прибуток за різними періодами часу, наприклад, за день, місяць або квартал;

Розрахунок заробітної плати: Бухгалтер може ввести дані про робочий час співробітників, їхню заробітну плату та інші відомості, необхідні для розрахунку зарплати;

Контроль фінансових коштів: Бухгалтер веде контроль за фінансовими коштами компанії, перевіряє рахунки, виставлені рахунки, операції з оплатою та отриманням коштів;

АРМ керівника вирішує такі функції:

Визначення стратегічних напрямків розвитку: Керівник може використовувати систему для аналізу даних про ринок, конкурентну ситуацію, тенденції розвитку галузі тощо. На основі цих даних можна визначити стратегічні

напрямки розвитку компанії, наприклад, розширення мережі пунктів прокату, впровадження нових технологій, підвищення якості обслуговування тощо;

Створення стимулів для співробітників: Керівник може використовувати систему для моніторингу роботи співробітників, їхнього внеску у досягнення цілей компанії. На основі аналізу даних можна визначити ефективні стимули для мотивації співробітників, такі як бонусні системи, премії, програми навчання та розвитку тощо;

Система бухгалтерського звіту вирішена на основі типових проектних рішень. Візьмемо готові програмні засоби і будемо інтегрувати їх. ІС розробляється на основі СУБД фірми каршерінгу.

Проект написано на мові програмування PHP, реалізований за допомогою таких засобів: JavaScript (jQuery), MySQL, css (bootstrap), Laravel 11.

Для адміністрування бази даних використано Dbeaver, а для створення таблиць механізм міграцій вбудований в фреймворк.

Для експлантації використовується операційна система Windows, яка надає широкий вибір інтегрованих розробницьких інструментів, таких як PhpStorm: Інтегроване середовище розробки (IDE), яке використовується для написання коду програми.

Для створення макету проекту використовується Figma.

Для ефективної роботи системи необхідно мати доступ до пошти та Інтернету. Ось як ці сервіси можуть бути використані:

- 1) Пошта: для сповіщень користувачів про підтвердження бронювання, оновлення статусу бронювання, нагадування про наближення часу оренди.
- 2) Інтернет: для реалізації бронювання автомобілів і взаємодії з клієнтами важливо мати доступ до Інтернету. Клієнти повинні мати змогу переглядати доступні автомобілі, робити бронювання та здійснювати оплату онлайн через веб-сайт системи каршерінгу.

Для експлуатації проекту потрібна обчислена техніка, а саме наступний набір комп'ютерів:

- Сервер баз даних: Один або декілька серверів, залежно від очікуваної кількості користувачів та обсягу даних. Для початку можна розглянути один сервер, проте при збільшенні обсягу та навантаження може бути потрібно додаткові сервери або розподілені бази даних;
- Веб-сервер та додаткові сервіси: Один або декілька серверів, які відповідають за обробку запитів веб-клієнтів, керування бронюваннями, автентифікацію тощо.
- Клієнтські комп'ютери: комп'ютери, на яких розробляється системи каршерінгу, тестуються нові функції та вирішуються проблеми. Зазвичай це можуть бути звичайні настільні комп'ютери або ноутбуки.
- Комутатори та маршрутизатори: Необхідно мати мережеве обладнання для побудови локальної мережі та забезпечення зв'язку між комп'ютерами та іншими пристроями.

Розроблена концепція дозволить реалізувати проект системи відповідно до її цільового призначення.

РОЗДІЛ 2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ КАРШЕРІНГУ АВТОМОБІЛЕЙ

2.1 Рішення щодо інформаційного забезпечення

Для ефективної реалізації проекту інформаційної системи каршерінгу автомобілів важливо ретельно визначити всі види даних, які будуть використовуватися та оброблятися в системі, а саме:

дані про користувачів, які необхідні для ідентифікації та взаємодії з користувачами сервісу: особисті дані, контактна інформація, водійські права;

інформація про всі автомобілі, доступні для оренди через систему каршерінгу: основні та технічні характеристики, вартість, категорія;

інформація про бронювання та оренду: поточні оренди, умови та терміни оренди.

Ці дані об'єднані в процесі моделювання даних, а саме створення моделі для організації даних, які будуть зберігатися в базі даних. Існує три основні стилі моделювання даних: концептуальні, логічні і фізичні моделі даних, кожна з яких має своє призначення. Моделі даних використовуються для представлення даних і їх зберігання в БД, а також для встановлення зв'язків між елементами даних.

Концептуальна модель даних: основною метою є визначення того, що система містить. Це високорівнева модель, яка використовується для представлення основних бізнес ідей і правил, без детальної технічної реалізації. Вона включає сутності (об'єкти, які важливі для бізнесу), їх атрибути (характеристики або властивості сутностей) та взаємозв'язки між сутностями.

Логічна модель даних: визначає структуру елементів даних і встановлює зв'язки між ними, незалежно від конкретної системи управління базами даних (СУБД). Вона є більш детальною порівняно з концептуальною моделлю і зазвичай створюється архітекторами даних та бізнес-аналітиками. Логічна модель включає конкретні типи даних, зв'язки між таблицями, цілісні правила та інші елементи, які допомагають у визначенні точного способу організації даних.

Фізична модель даних: описує конкретну реалізацію моделі даних для певної СУБД. Вона створюється адміністраторами баз даних та розробниками і включає фізичні характеристики БД, такі як типи даних, індекси, обмеження та інші технічні аспекти. Фізична модель враховує специфічні особливості вибраної СУБД, щоб забезпечити ефективне зберігання та доступ до даних [9,10].

На основі вище вказаної інформації було розроблено концептуальну модель, яка представлена на рисунку 2.1.

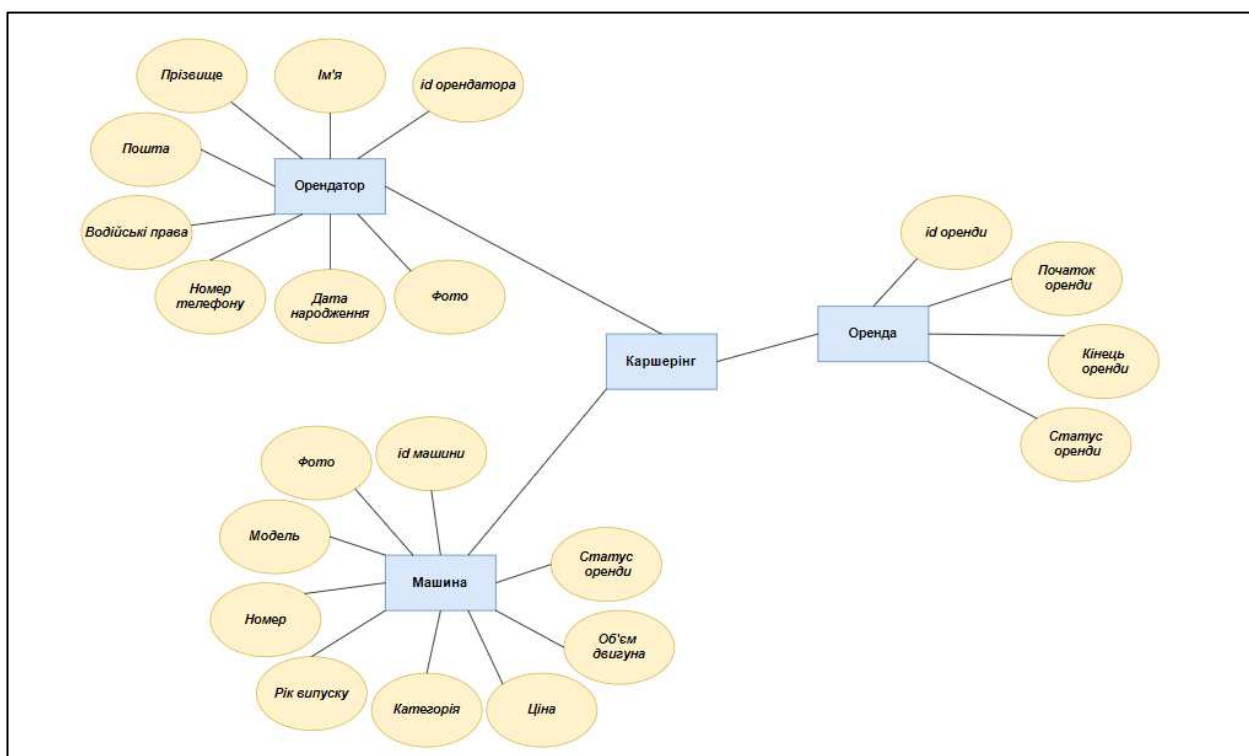


Рисунок 2.1 – Концептуальна модель в нотації Чена

Всі сутності атрибути та зв'язки представлені у таблиці 2.1.

Таблиця 2.1 – Концептуальна таблиця каршерінгу

Назва сутності або зв'язку	Назва атрибуту
----------------------------	----------------

Орендатор	Id орендатора Ім'я Прізвище Пошта Водійські права Номер телефону Дата народження Фото
Машина	Id машини Фото Модель Номер Рік випуску Категорія Ціна Об'єм двигуна
Оренда	Id оренди Id орендатора Id машини Початок оренди Кінець оренди Статус оренди

Логічна модель-це більш детальне представлення даних, яке представлено в таблицях 2.2, 2.3, 2.4.

Таблиця 2.2 – Атрибути сутності «Орендатор»

Атрибут	Опис даних
Id орендатора	Унікальний ідентифікатор орендатора
Ім'я	Ім'я орендатора
Прізвище	Прізвище орендатора
Пошта	Електронна пошта
Водійські права	Номер водійських прав орендатора

Номер телефону	Контактний номер телефону
Дата народження	День, місяць, рік народження
Фото	Фото орендатора

Таблиця 2.3 – Атрибути сутності «Машина»

Атрибут	Опис даних
Id машини	Унікальний ідентифікатор машини
Фото	Фото машини
Модель	Модель машини
Номер	Номерний знак машини
Рік випуску	Рік випуску машини
Категорія	Категорія машини (економ, стандарт, люкс)
Ціна	Вартість оренди машини за годину
Об'єм двигуна	Об'єм двигуна машини

Таблиця 2.4 – Атрибути сутності «Оренда»

Атрибут	Опис даних
Id оренди	Унікальний ідентифікатор оренди
Id орендатора	Унікальний ідентифікатор орендатора
Id машини	Унікальний ідентифікатор машини
Початок оренди	Дата та час початку оренди
Кінець оренди	Дата та час закінчення оренди
Статус оренди	Статус оренди (активна, завершена)

Створена логічна модель представлена на рисунку 2.2

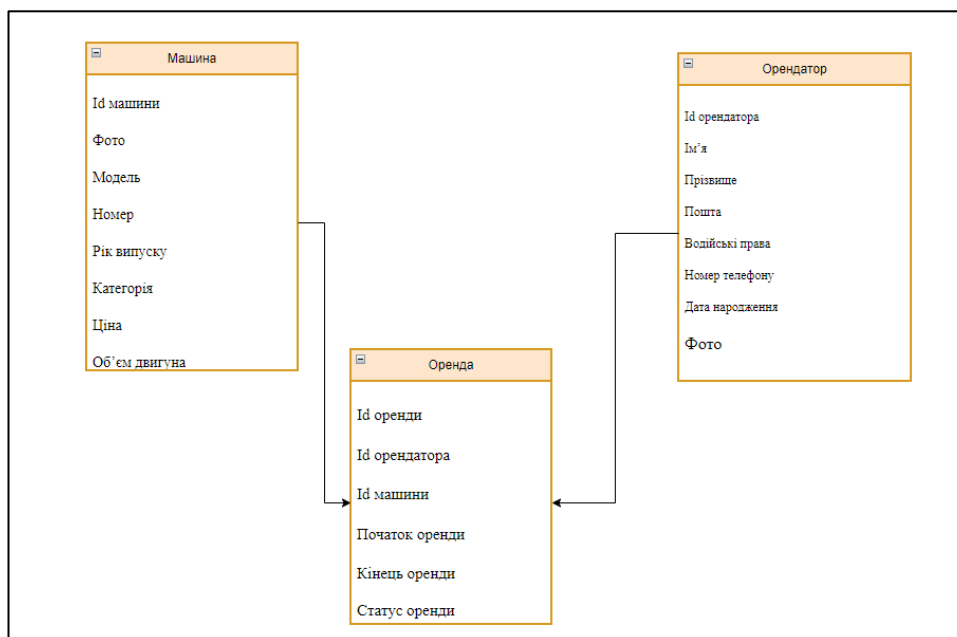


Рисунок 2.2 – Логічна модель

Фізична модель включає фізичні характеристики БД, такі як типи даних, індекси, обмеження та інші технічні аспекти, які представлені в таблиці 2.5.

Таблиця 2.5 – Фізична модель даних

Ім'я об'єкта фізичної мод.	Найменування поля	Тип даних	Розмір поля (байт)	Індекс	Ключ
Орендатор	Id орендатора	bigint	8	ІК	Первинний
	Ім'я	varchar	255		
	Прізвище	varchar	255		
	Пошта	varchar	255		
	Водійські права	varchar	30		
	Номер телефону	varchar	255		
	Дата народження	datetime	255		
	Фото	varchar	255		
	Id машини	bigint	8	ІК	Первинний
	Фото	varchar	255		

Машина	Модель	varchar	50		
	Номер	varchar	30		
	Рік випуску	varchar	4		
	Категорія	varchar	50		
	Ціна	tinyint	8		
	Об'єм двигуна	tinyint	8		
Оренда	Id оренди	bigint	8	ІК	Первинний
	Початок оренди	datetime	8		
	Кінець оренди	datetime	8		
	Статус оренди	tinyint	8		
	Id орендатора	bigint	8		Зовнішній
	Id машини	bigint	2585		Зовнішній

2.2 Рішення щодо програмного забезпечення

В процесі розробки було використано PHPStorm, завдяки його зручному середовищу, потужним інструментам і широкому набору функцій, значно спростили процес програмування.

Як середовище розробки клієнт-серверного додатку, було обрано СУБД MySQL та PHP.

MySQL – це система управління реляційними базами даних (СУРБД) з відкритим вихідним кодом. Вона використовує модель «клієнт-сервер». Комплекс програм або ціла служба, яка використовується для роботи з базами даних на основі реляційної моделі, називається СУРБД. Базова структура системи «клієнт-сервер» дуже проста. Клієнт підключається до сервера через мережу і робить запит через інтерфейс користувача. Якщо сервер розуміє інструкції, він надає необхідну інформацію.

У середовищі MySQL відбуваються такі процеси:

- створення база даних MySQL, визначення структури таблиці;

- клієнти роблять запити, використовуючи команди SQL [11].

Для адміністрування бази даних використано Dbeaver, а для створення таблиць механізм міграцій вбудований в фреймворк. DBeaver Community Edition – це графічний інструмент для роботи з різноманітними базами даних, такими як MySQL. Щодо переваг, DBeaver є зручним засобом для роботи з готовими базами даних. Він дозволяє переглядати ER-діаграми баз даних, писати скрипти та виконувати SQL-запити з переглядом їх результатів, редагувати дані в таблицях, а також отримувати деякі статистичні показники по стовпчиках, такі як середнє значення, мода, сума [12].

Проект написано на мові програмування PHP, реалізований за допомогою таких засобів: JavaScript (jQuery), MySQL, css (bootstrap), Laravel 11.

PHP - це мова програмування з відкритим вихідним кодом, яка розроблена для веб-розробки і може бути впроваджена безпосередньо в HTML код.

Основними областями застосування PHP є:

створення скриптів та програм, які працюють на серверній частині;
розробка графічних інтерфейсів.

PHP незамінний для створення динамічних сайтів, таких як соціальні мережі, інтернет-магазини, новинні портали та браузерні ігри.

jQuery – це бібліотека JavaScript, вона лише працює в парі з JavaScript. Однією з ключових особливостей jQuery є можливість легкого пошуку та зміни будь-якого елементу DOM (об'єктної моделі документа). Бібліотека пропонує безліч функцій, які дозволяють змінювати інформацію, що відображається перед відвідувачами, наприклад, показувати або приховувати елементи, які не видно під час завантаження сторінки [13, 14].

CSS — це мова стилів, що використовується для визначення вигляду веб-документів, таких як розташування та стиль елементів сторінки. Зазвичай CSS використовується для стилізації документів, написаних мовами розмітки, такими як HTML, XHTML та XML.

Однією з ключових переваг використання CSS є можливість розділення змісту сторінки від її візуального оформлення. Це розділення сприяє поліпшенню

доступності та сприйняття контенту, забезпечує більшу гнучкість і контроль над відображенням на різних пристроях, структурує зміст і усуває дублювання. Саме ціль відокремлення змісту від представлення була основною метою розробки цієї технології.

Переваги використання CSS включають:

- можливість відображення одного документу в різних стилях;
- адаптивний дизайн для різних пристроїв, що дозволяє оптимізувати вигляд сторінки для різних умов;
- зменшення часу завантаження сторінок завдяки кешуванню CSS-правил;
- легкість зміни дизайну без редагування кожної окремої сторінки;
- додаткові можливості оформлення, такі як фіксація меню при прокручуванні сторінки або зміна стилів посилань;
- можливість створення складних і пророблених дизайнів завдяки розширеній CSS-розмітці.

Bootstrap - це безкоштовна платформа для інтерфейсу, яка сприяє швидкій та простій веб-розробці. Вона містить готові шаблони дизайну на основі HTML і CSS для різноманітних елементів, таких як типографіка, форми, кнопки, таблиці, навігація, модальні вікна, каруселі зображень і багато іншого. Крім того, вона має додаткові модулі JavaScript. Bootstrap також дозволяє легко створювати адаптивний дизайн, що адаптується до різних розмірів екранів пристроїв.

Переваги Bootstrap:

- простота використання: навіть користувачі з базовими знаннями HTML і CSS можуть легко почати працювати з Bootstrap;
- адаптивність: CSS Bootstrap адаптується до різних пристроїв, таких як телефони, планшети і комп'ютери, забезпечуючи єдиний вигляд на різних пристроях;
- мобільна орієнтація: В Bootstrap 3 стилі, спрямовані на мобільні пристрої, є складовою частиною основної структури, що полегшує розробку для мобільних платформ;

- сумісність з веб-браузерами: Bootstrap сумісний із всіма сучасними веб-браузерами, такими як Chrome, Firefox, Internet Explorer, Edge, Safari та Opera, що забезпечує однаковий вигляд в різних середовищах.

Для створення макету проекту використовується Figma – зручний графічний редактор. Він пропонує різноманітні можливості, серед яких: створення прототипів веб-сайтів і додатків. Розробка окремих елементів інтерфейсу, таких як іконки, кнопки, форми та інші. Редагування та створення векторних зображень і ілюстраційх [15, 16,17].

У цілому, використання таких інструментів спростило та полегшило розробку проекту, забезпечуючи ефективність та якість результату.

РОЗДІЛ 3 РОЗРОБКА ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ

На початку розроблення проекту було виконано макет сайту в онлайн-сервісу для розробки інтерфейсів Figma.

Розробка інтерфейсу проводиться за допомогою інструментів та можливостей Figma, які представлені на рисунку 3.1.

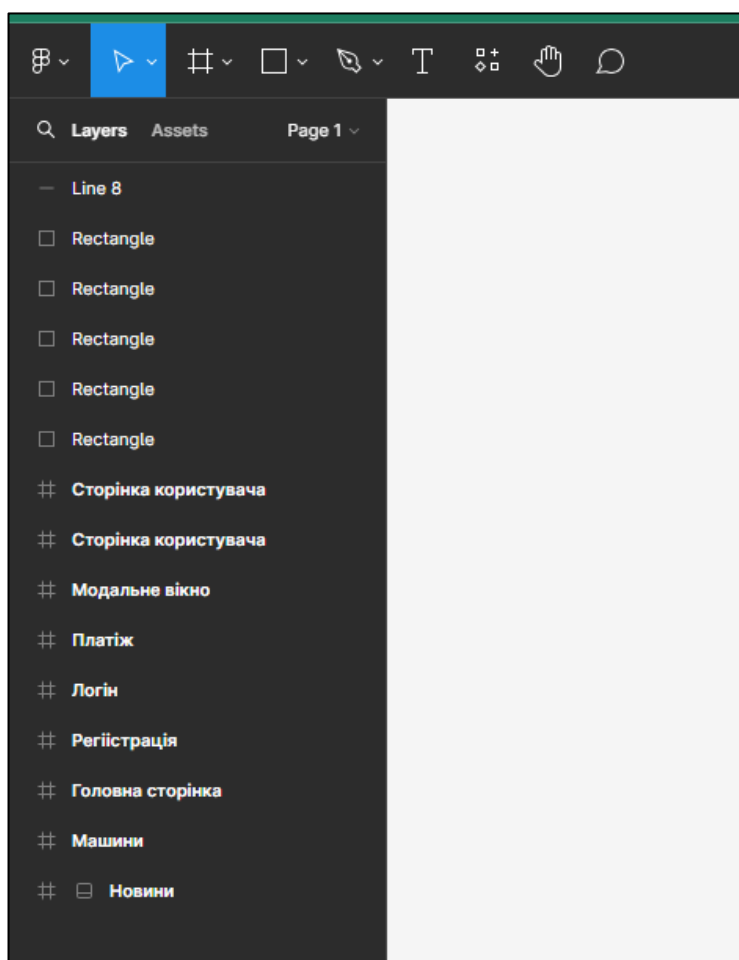


Рисунок 3.1 – Панель інструментів Figma

Далі буде представлено дизайн деяких ескізів сторінок для сайту каршерінгу, що дають можливість реалізувати сайт, представлені на рисунках 3.2, 3.3, 3.4.

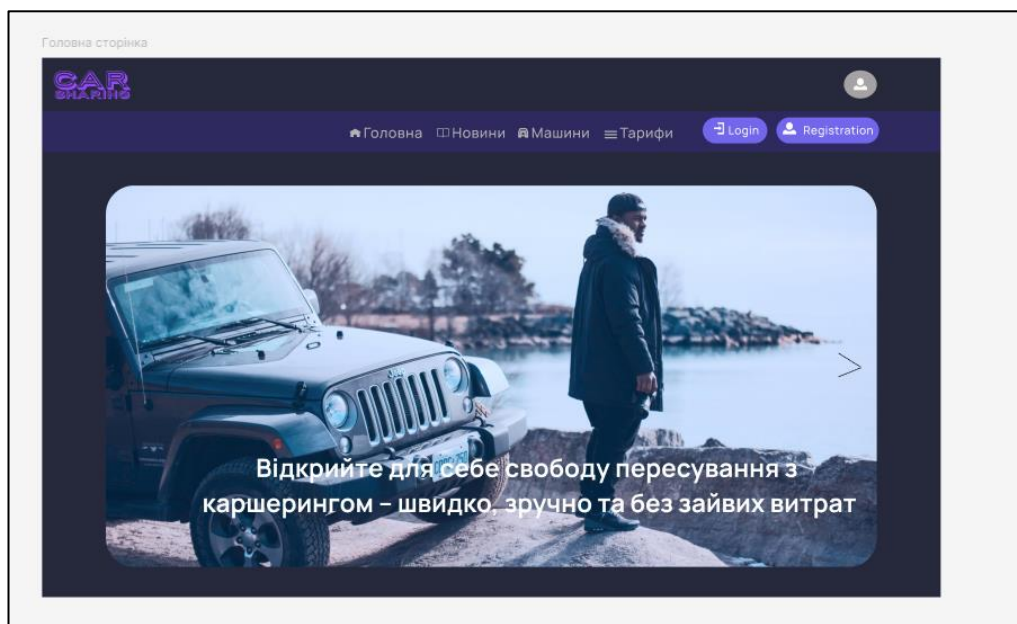


Рисунок 3.2 – Ескіз головної сторінки

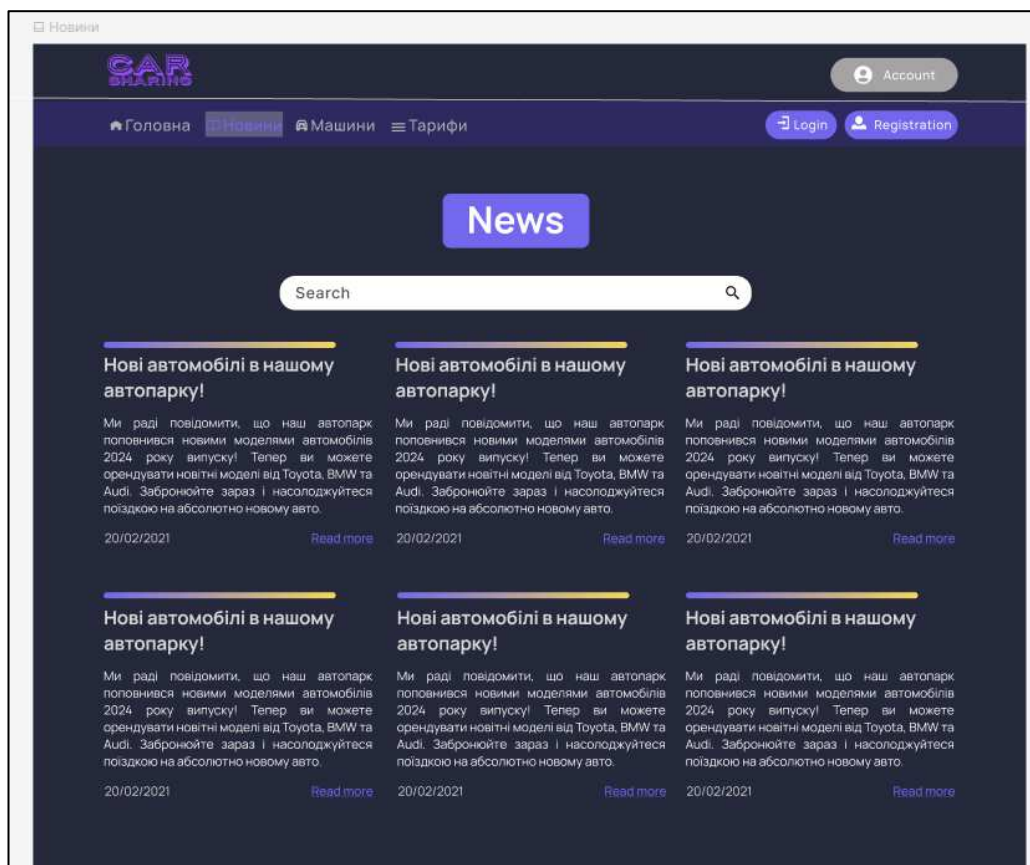


Рисунок 3.3 – Ескіз сторінки новин

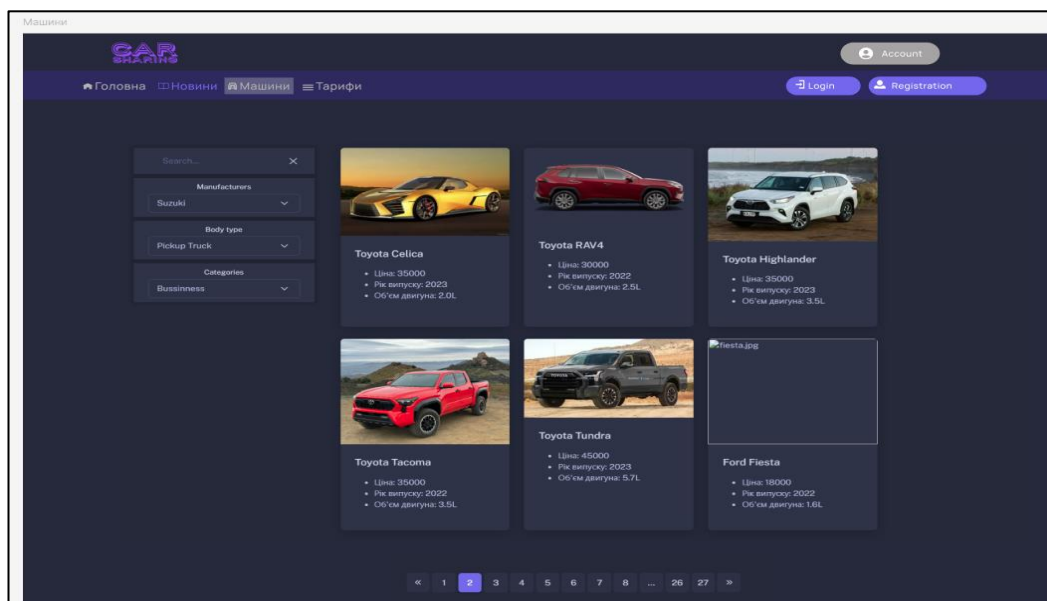


Рисунок 3.4 – Ескіз сторінки машин

В процесі розробки сайту, дизайн змінювався, але за основу було взято макет з програми Figma. Далі можна побачити вже готовий сайт, який слугуватиме для користувачів.

Головна сторінка має простий і зрозумілий інтерфейс, що забезпечує легкість у використанні та швидке освоєння на початковому етапі впровадження. Використання слайдера дозволяє зручно переглядати більше фотографій, забезпечуючи легкий доступ до всіх зображень, яке представлено на рисунку 3.5.

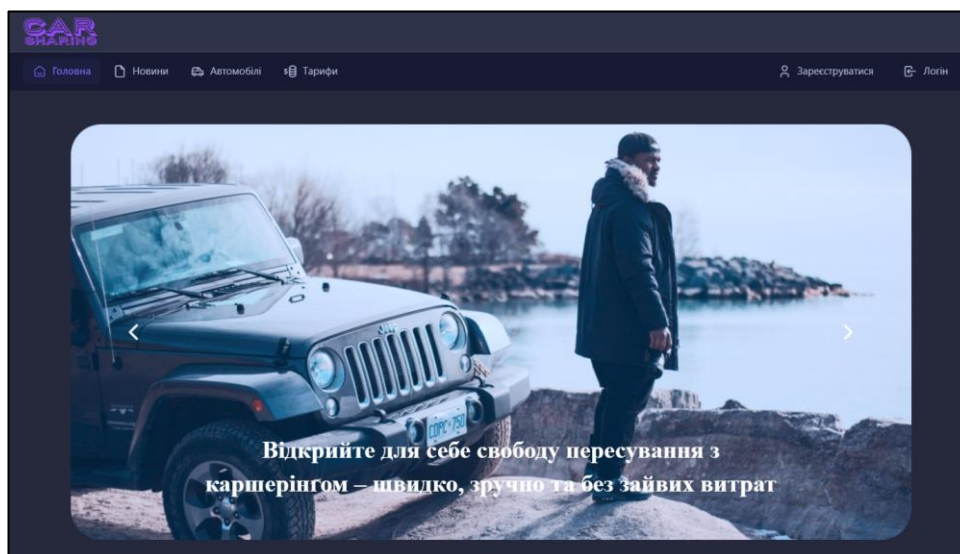


Рисунок 3.5 – Головна сторінка

Головна сторінка надає вичерпну інформацію про каршерінг. За допомогою інформаційних блоків, користувачі можуть ознайомитися з перевагами цього сервісу, його принципом роботи. Такий підхід сприяє кращому розумінню концепції каршерінгу та забезпечує більшу залученість користувачів до використання цієї послуги, що представлено на рисунку 3.6, 3.7.

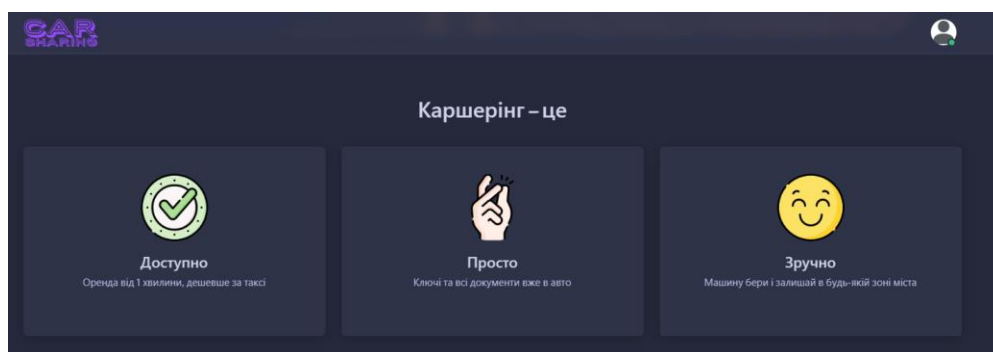


Рисунок 3.6 – Інформація про каршерінг

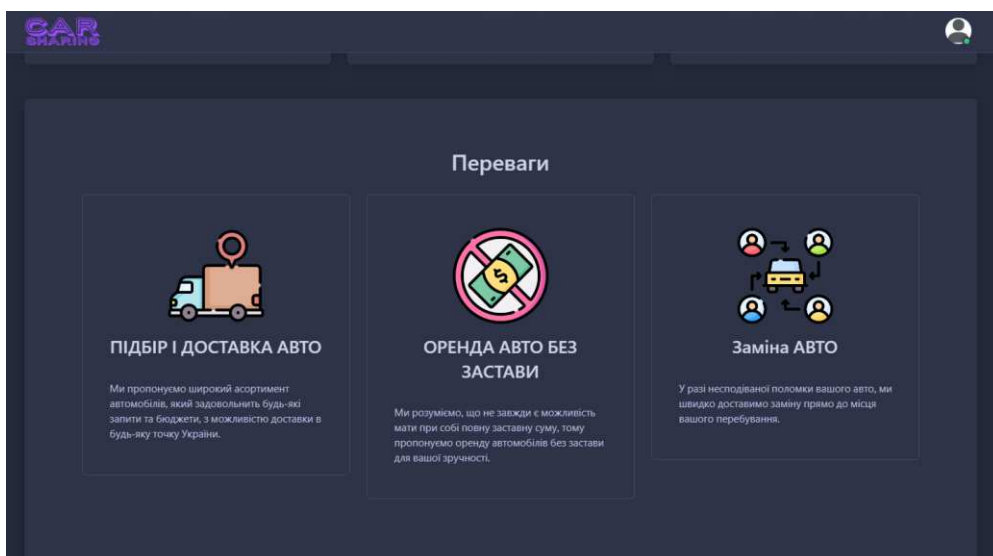


Рисунок 3.7 – Переваги каршерінгу

Сторінка новин структурована таким чином, щоб користувач міг ознайомитися з цікавою інформацією про автомобілі, дізнатися про новини та події у світі автомобільної індустрії. Також там присутня пагінація, яка полегшує перехід між сторінками та забезпечує зручний перегляд контенту, представлений на рисунку 3.8.

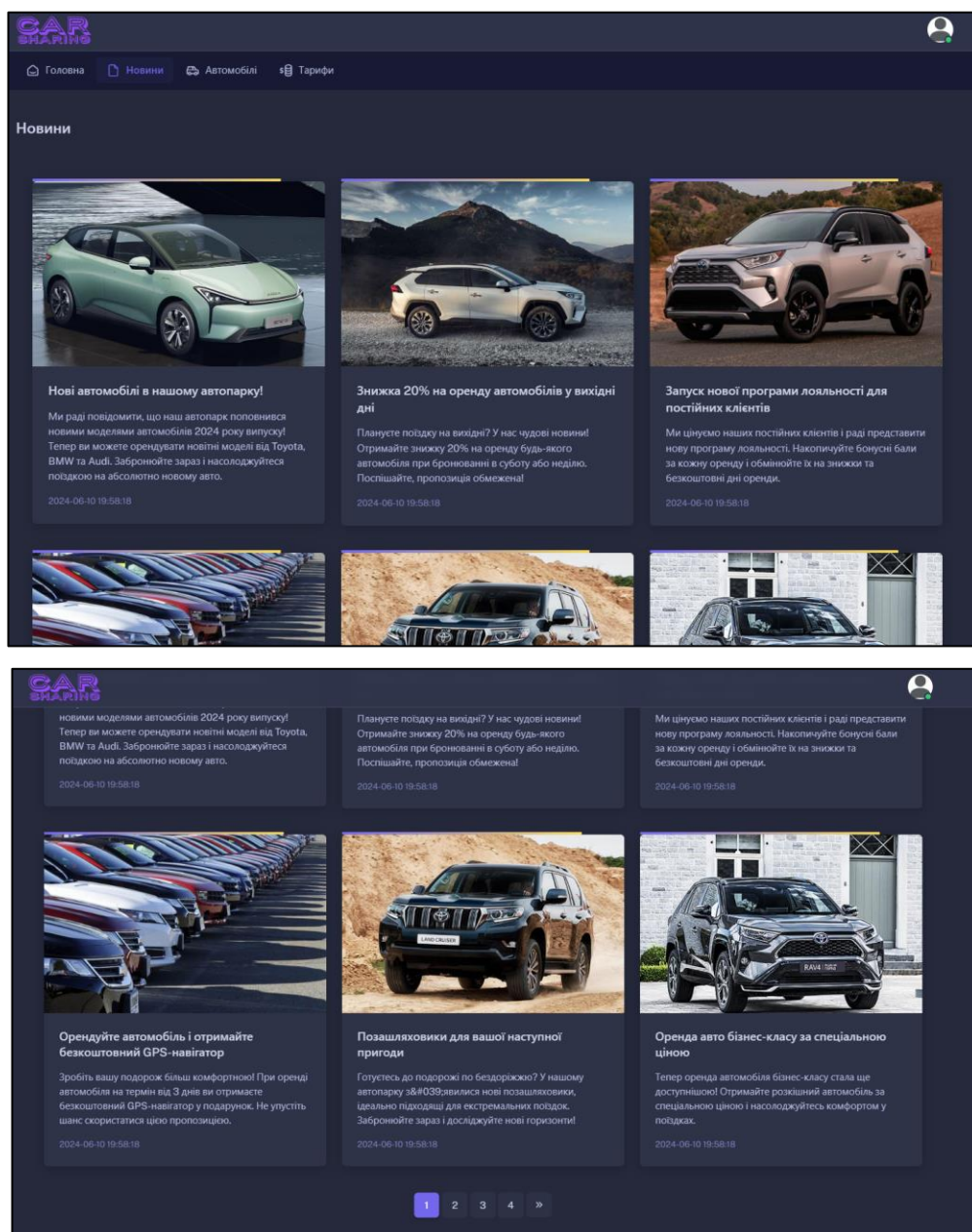


Рисунок 3.8 – Сторінка "Новини"

На сторінці "Машини" ви побачите перелік доступних автомобілів, їх категорію, вартість оренди та об'єм двигуна. Також є пагінація для перемикавання сторінок, представлено на рисунку 3.9.

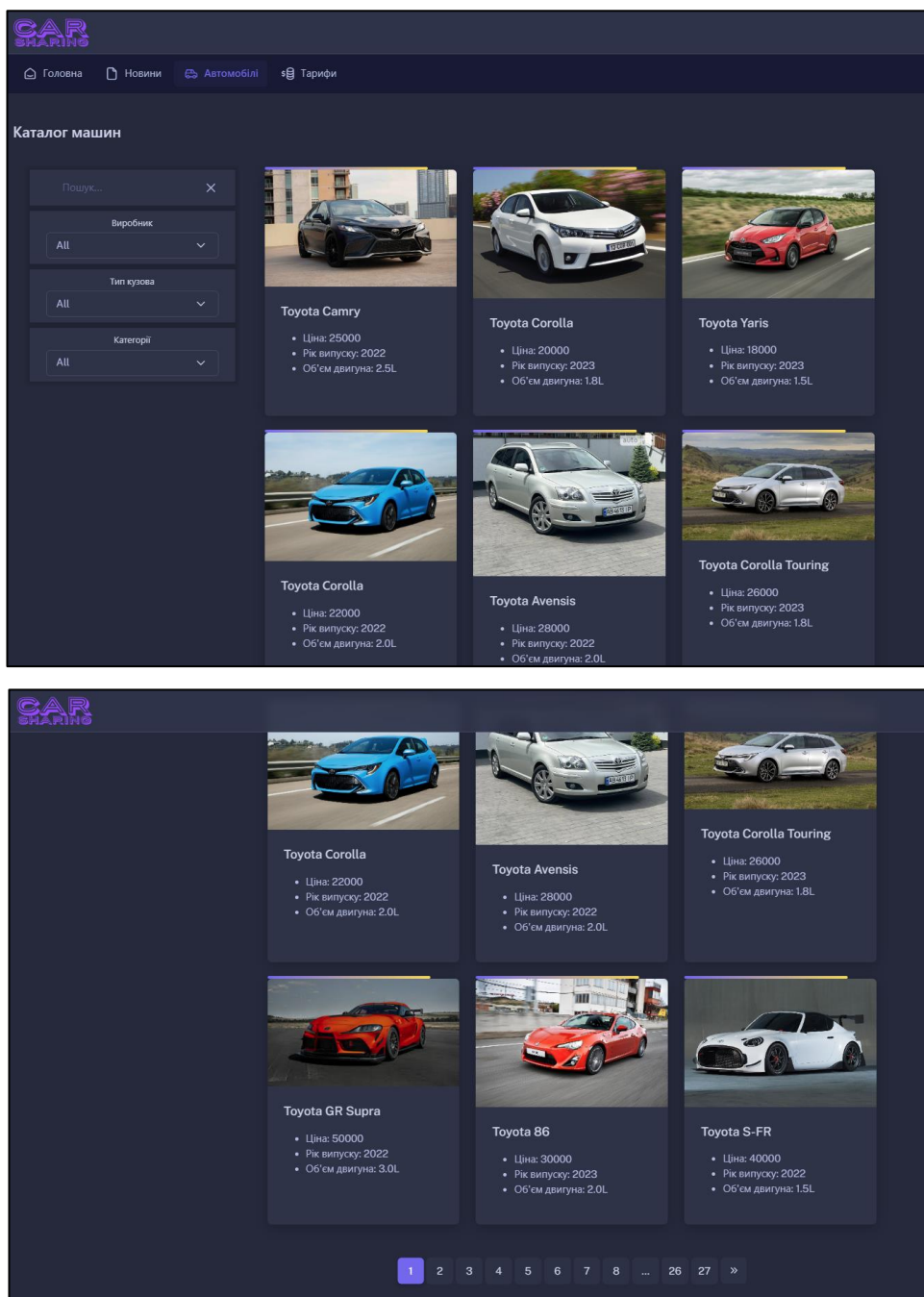


Рисунок 3.9 – Сторінка "Машини"

На сторінці також доступний пошук автомобілів за назвою, а також можна вибрати виробника, тип кузова та категорію. Такий функціонал дозволяє користувачам знайти авто за їхніми уподобаннями та потребами, представлено на рисунку 3.10.

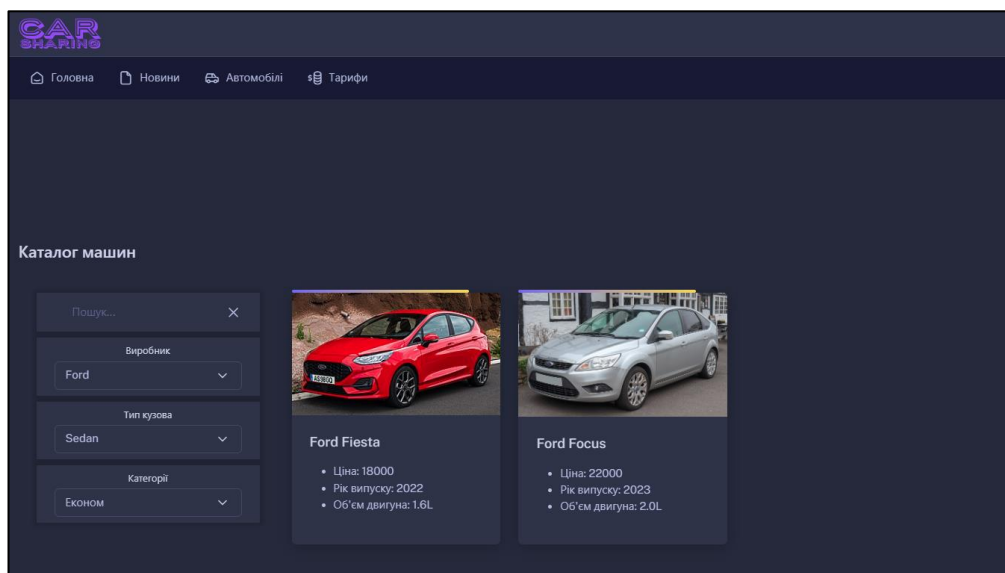


Рисунок 3.10 – Функціонал

Якщо користувач не зможе знайти автомобіль, він побачить повідомлення, яке представлено на рисунку 3.11.

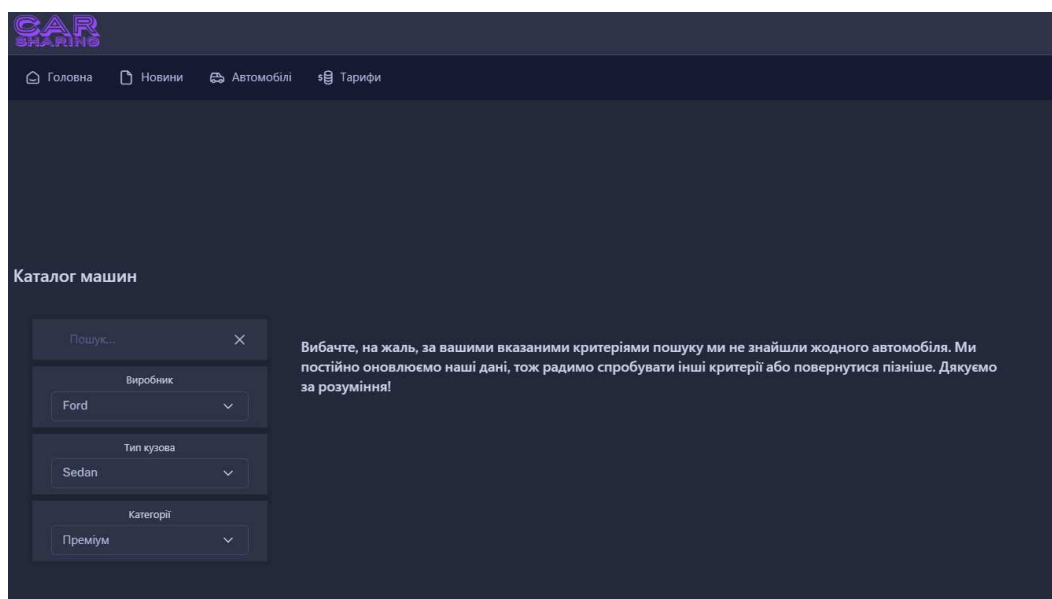


Рисунок 3.11 – Повідомлення про відсутність результатів пошуку автомобілів

На сторінці "Тарифи" ви зможете переглянути вартість оренди автомобіля, проте ціни будуть відрізнятися в залежності від категорії автомобіля, що представлено на рисунку 3.12.

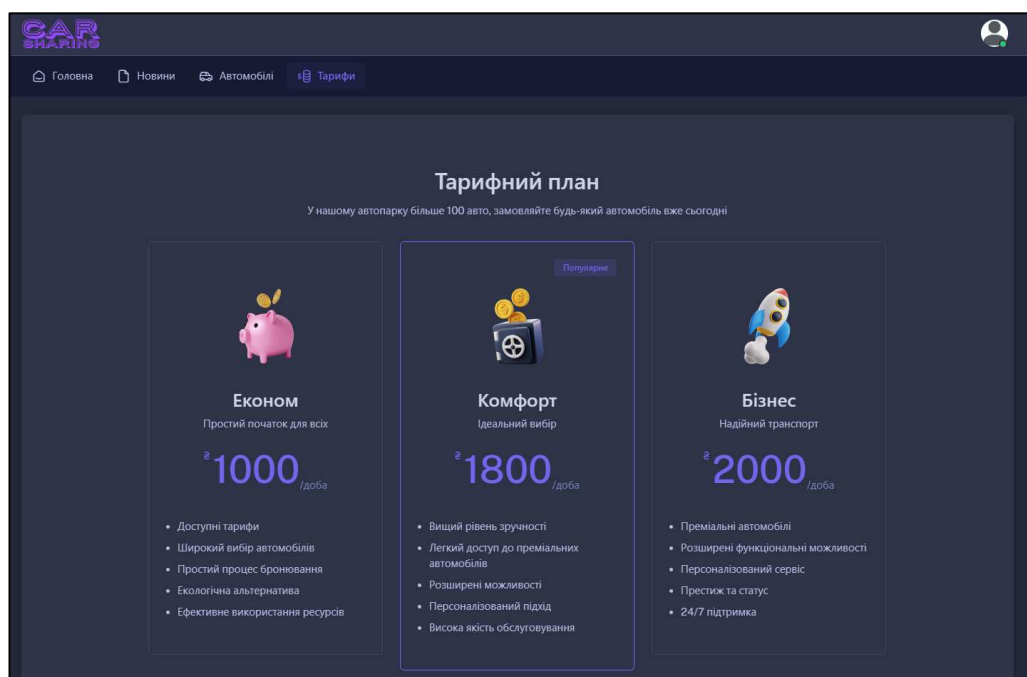


Рисунок 3.12 – Сторінка "Тарифи"

Реєстрація дозволяє користувачам побачити новий функціонал - оформлення замовлення на автомобіль у вигляді кнопки. При наявності облікового запису можна просто увійти та перейти до профілю користувача, що представлено на рисунках 3.13 та 3.14.

Створення облікового запису

Email:

Ім'я:

Прізвище:

Телефон:

Пароль:

Повторення пароля:

☒ Я згоден з політикою та умовами конфіденційності

Увійти

[Вже є акаунт? Увійти](#)

[Повернутися на головну](#)

Рисунок 3.13 – Створення облікового запису (реєстрація)

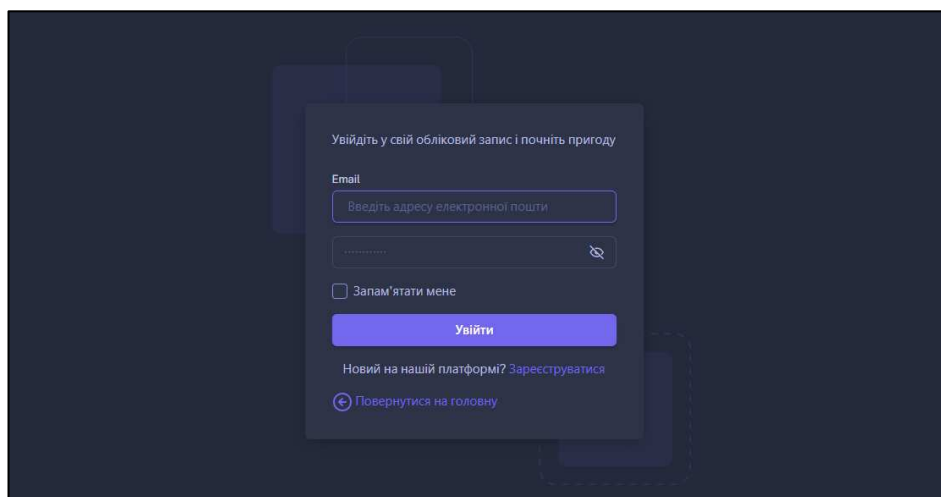


Рисунок 3.14 – Вхід до облікового запису

Після входу в обліковий запис користувач отримує доступ до різноманітних функцій, включаючи можливість здійснення бронювання автомобіля. У профілі користувача відображається вся необхідна інформація про нього, а також його остання активність. Користувач може додати свою платіжну картку для зручного здійснення платежів. Крім цього, система надає такі функції: завантаження фото користувача, редагування інформації, зміна пароля, а також видалення аккаунту, що представлено на рисунках 3.15, 3.16, 3.17, 3.18, 3.19.

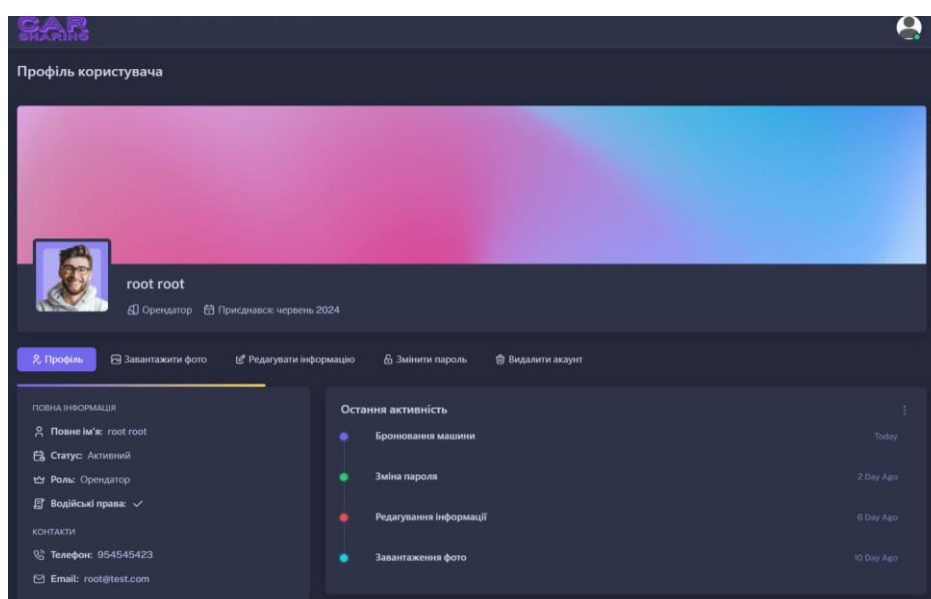


Рисунок 3.15 – Профіль користувача

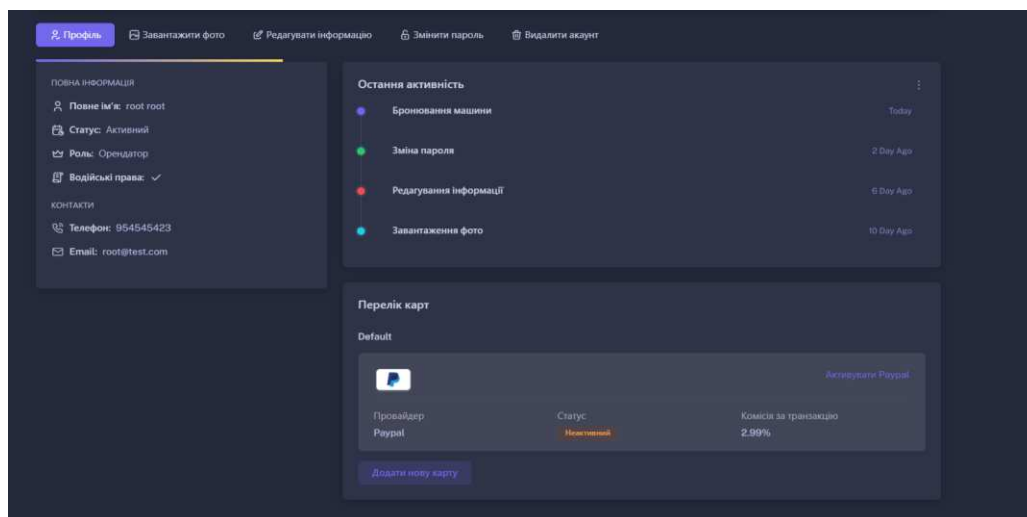


Рисунок 3.16 – Перелік карт

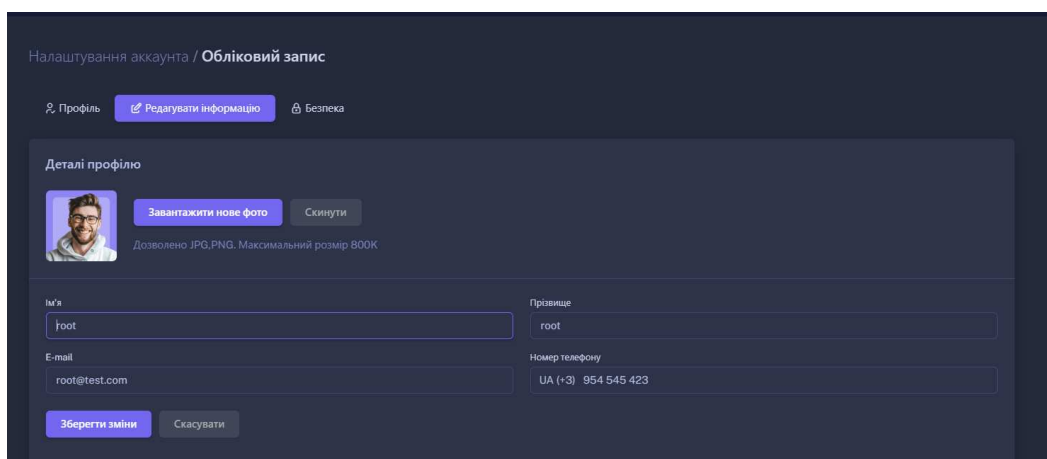


Рисунок 3.17 – Редагування інформації та фото користувача

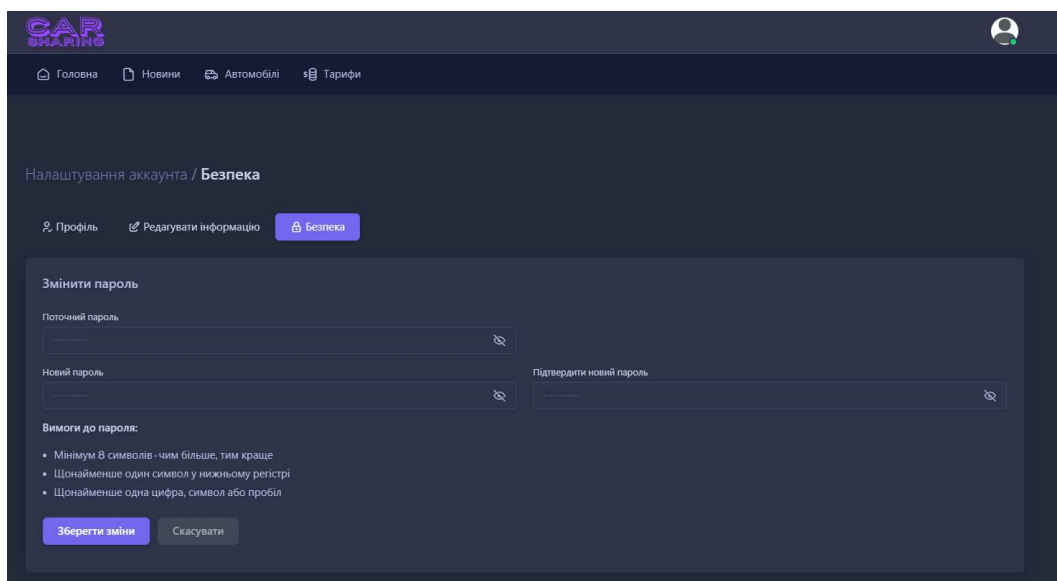


Рисунок 3.18 – Змінення паролю

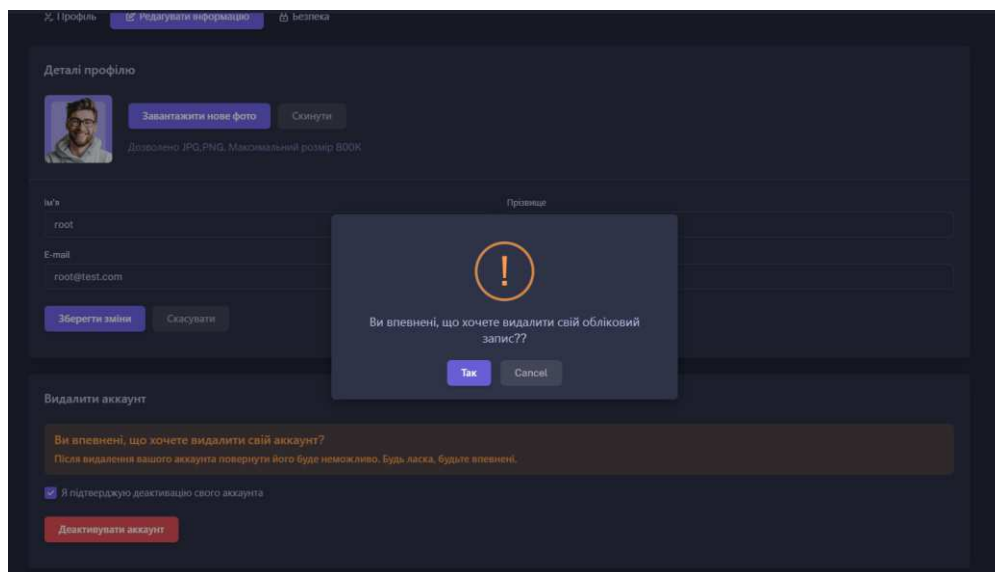


Рисунок 3.19 – Видалення акаунту

Після входу в акаунт користувач отримує можливість замовити автомобіль всього одним кліком, це представлено на рисунку 3.20.

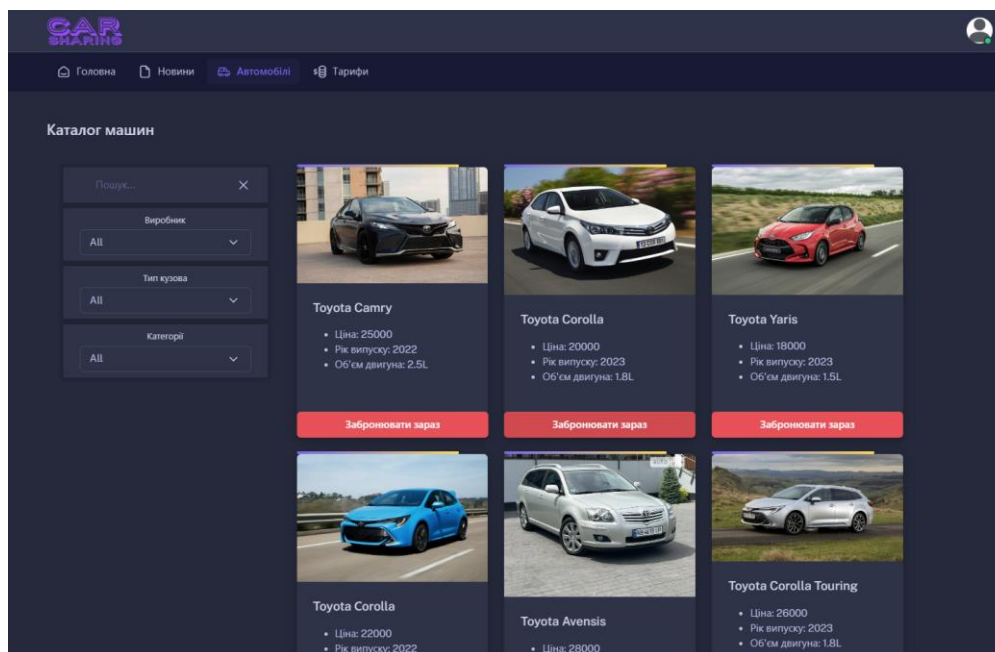


Рисунок 3.15 – Замовлення машини

РОЗДІЛ 4 ОХОРОНА ПРАЦІ

4.1 Умови роботи

Охорона праці є невід'ємною частиною діяльності будь-якої організації, яка забезпечує насамперед безпеку та здоров'я працівників та клієнтів.

Охорона праці – це область роботи в галузі охорони здоров'я, спрямована на сприяння та підтримку рівня фізичного, психічного та соціального благополуччя працівників усіх професій.

Її цілі включають в себе:

збереження та зміцнення здоров'я та працездатності працівників;

покращення умов праці та робочого середовища, щоб вони сприяли безпеці та здоров'ю;

розвиток організації праці та робочої культури, які повинні відображати основні системи цінностей, прийняті відповідним підприємством [18].

Для забезпечення нормальних умов роботи в компанії, що надає послуги каршерінгу автомобілів важливо організувати робочий час і режим роботи, щоб бізнес не втрачав ефективність та зберігав безпеку та здоров'я працівників. На фірмі повинен бути змінний графік по вісім годин, щоб забезпечити відпочинок для співробітників з обов'язковими перервами на харчування. Згідно з встановленим графіком працівники мають щотижневі вихідні.

Для забезпечення нормальних умов праці з технікою визначають санітарно-гігієнічні вимоги.

Санітарні норми — це документи, які встановлюють обов'язкові вимоги до безпеки та нешкідливості факторів навколишнього середовища для людини, а також стандарти для забезпечення оптимальних або припустимих умов для її життєдіяльності. Ці нормативи мають на меті забезпечити охорону здоров'я та безпеку громадян шляхом регулювання умов праці, виробничого процесу, життєвого середовища тощо.

Вимоги:

на робочих місцях в приміщеннях слід забезпечувати оптимальні параметри мікроклімату, такі як температура повітря (22–25°C), відносна вологість (40–60%), та обмежену швидкість руху повітря (не більше 0,1 м/с);

площа приміщень, де розташовані персональні комп'ютери, має відповідати встановленим нормативам. Згідно з ДСанПіН 3.3.2.007-98, на кожне робоче місце з ПК припадає площа не менше 6,0 кв. м та об'єм не менше 20,0 куб. м.;

заземлені конструкції в приміщеннях (такі як батареї опалення, водопровідні труби, заземлені кабелі) повинні бути захищені діелектричними щитками або сітками, щоб уникнути випадкового дотику;

робоче місце користувача ПК має бути спроектоване з урахуванням ергономічних вимог, що забезпечать зручну робочу позицію. Робочий стіл повинен відповідати сучасним стандартам та дозволяти зручне розташування обладнання та документів;

також у цих приміщеннях мають бути медичні аптечки першої допомоги, системи автоматичної пожежної сигналізації та переносні вуглекислотні вогнегасники. Доступ до засобів пожежогасіння повинен бути вільним;

у приміщеннях, де працюють з персональними комп'ютерами, застосовується система загального рівномірного освітлення, зокрема, за ДСанПіН 3.3.2.007-98. У випадках, коли робота в основному пов'язана з документами, можна використовувати комбіновану систему, де світильники місцевого освітлення доповнюють загальне освітлення;

для забезпечення оптимального освітлення приміщень рекомендується регулярно мити вікна та світильники і своєчасно замінювати перегорілі лампи;

підлога має бути рівною, неслизькою;

безпека під час роботи з електроприладами також має важливе значення. Комп'ютери та периферійні пристрої слід підключати до електричної мережі за допомогою справних штепсельних з'єднань, а контакти повинні бути заземлені. У разі аварійної ситуації необхідно негайно відключити електроприлади від мережі;

приміщення для роботи з персональними комп'ютерами повинні бути обладнані системами опалення, кондиціонування повітря або припливно-витяжною

вентиляцією. Це необхідно для забезпечення комфортних умов праці протягом усього року, з урахуванням зміни погодних умов та температурного режиму [19].

4.2 Нормативно-правова база

Основними нормативними актами, що регулюють питання охорони праці в Україні є:

Закон України "Про охорону праці" є основним нормативним законом і визначає основні принципи, завдання та механізми реалізації політики у сфері охорони праці в Україні. Основні положення закону включають:

обов'язки роботодавця забезпечити безпечні умови праці, провести інструктаж та навчання працівників;

права працівників на: безпечні умови праці, відмову від роботи, якщо є загроза життю чи здоров'ю, участь у розслідуванні нещасних випадків;

державний нагляд і управління: організація перевірки додержання вимог охорони праці.

Закон України «Про дорожній рух» регулює безпеку дорожнього руху та визначає права та обов'язки учасників дорожнього руху. Важливі положення закону для каршерінгових компаній включають:

вимоги до водіїв: обов'язковий медичний огляд, дотримання правил дорожнього руху, відповідальність за порушення;

вимоги до транспортних засобів: технічне обслуговування та техогляд транспортних засобів, відповідність технічним стандартам і правилам безпеки;

організація дорожнього руху: безпека дорожнього руху, у тому числі встановлення дорожніх знаків, світлофорів та інших засобів регулювання дорожнього руху [20, 21].

Кодекс законів про працю України визначає основні положення трудових відносин, у тому числі з питань охорони праці. Важливі статті про цей код включають:

трудоий договір: умови укладення та розірвання трудового договору, обов'язки роботодавця щодо забезпечення безпечних умов праці;

робочий час і перерви: положення про робочий час, вихідні та святкові дні, перерви та свята.

Норми і правила охорони праці затверджуються відповідними державними органами і висувають особливі вимоги до організацій охорони праці підприємства. Ці характеристики можуть включати:

правила безпеки під час виконання окремих видів робіт: особливі вимоги безпеки при обслуговуванні автомобілів, роботі з хімічними речовинами, виконанні ремонтних робіт тощо;

вимоги до гігієнічних і санітарних умов праці, включаючи вентиляцію, освітлення, температуру і чистоту робочого місця.

Забезпечення виконання цих нормативних норм є основною умовою безпечної діяльності каршерінгових компаній, оскільки вони становлять правову основу організації охорони праці та захисту здоров'я та життя працівників і користувачів послуг [22, 23].

4.3 Небезпечні та шкідливі фактори робочого місця

Під час роботи каршерінгового сервісу можуть виникати різноманітні небезпечні та шкідливі ситуації, які мають вплив на працівників та орендаторів. Аналіз цих факторів є важливим для забезпечення безпеки й створення відповідних умов для всіх.

Шкідливі виробничі фактори – це фактори середовища і трудового процесу, що можуть спричинити професійну патологію, тимчасове або стійке зниження працездатності, підвищити частоту захворювань, призвести до порушення здоров'я потомства.

Шкідливі фактори можна поділити на групи, а саме:

Фізичні фактори – ризик травм при проведенні технічного обслуговування та ремонту автомобілів (порізи, удари, опіки). Можливість отримання травм через

падіння важких предметів або роботу з несправним обладнанням. Тривале перебування в умовах підвищеного шуму та вібрації може призвести до зниження слуху, стомлення та інших негативних наслідків для здоров'я.

Хімічні фактори – використання хімічних речовин для обслуговування автомобілів (мастила, антифризи, очисники) може призвести до негативного впливу на здоров'я працівників через вдихання парів або контакт з шкірою. Можливість отруєння при неправильному використанні або зберіганні хімічних речовин, а також алергічні реакції.

Робота з електричним обладнанням та інструментами може призвести до ураження електричним струмом у разі порушення правил безпеки.

Ергономічні фактори – невідповідність робочих місць вимогам може призвести до фізичних проблем, таких як болі в спині, шийі, кінцівках через тривале перебування в незручних позах. Підйом та перенесення важких предметів можуть призвести до травм та перенапруження м'язів та суглобів.

Мікрокліматичні фактори – робота при високих або низьких температурах може негативно вплинути на здоров'я працівників, викликаючи перегрівання або переохолодження. Недостатня вентиляція та високий рівень вологості можуть спричинити проблеми з дихальною системою та дискомфортом.

Вібрація є одним із шкідливих виробничих факторів, і її класифікують за декількома параметрами:

За способом передачі:

Загальна – вплив вібрації поширюється на весь організм через опорну поверхню, наприклад, при роботі на вібруючому обладнанні. Локальна вібрація: вплив вібрації обмежений конкретними частинами тіла, наприклад, долонями під час роботи з вібруючим інструментом.

По часу дії:

Тимчасова – вплив вібрації відбувається протягом короткого періоду, наприклад, під час короткочасної роботи з вібруючим інструментом. Постійна – вплив вібрації триває протягом тривалого періоду, наприклад, при роботі з постійно

вібруючим обладнанням. постійний вплив вібрації може мати серйозні наслідки для здоров'я працівників.

Цей фактор може негативно впливати не лише на нервову систему, а й на опорно-рухову систему. Нервова система може страждати від постійного впливу вібрації через розвиток різних неврологічних порушень, таких як головні болі, запаморочення, втома, роздратування та інші симптоми.

Опорно-рухова система також може постраждати через постійний стрес, що може призвести до розвитку захворювань опорно-рухової системи, таких як м'язові напруги, болі в спині, суглобах та м'язах, а також до зниження м'язової сили та витривалості.

Для мінімізації впливу небезпечних та шкідливих факторів потрібно виконати такі заходи:

Надати працівникам засоби індивідуального захисту, використання спецодягу, рукавичок, захисних окулярів під час технічного обслуговування автомобілів.

Дотримання безпеки при проведенні робітних робіт, під час заміни зношених частин та шин, регулярні огляди автомобілів.

Безпека на робочому місці вимагає систематичних заходів, а інструктаж є одним із ключових елементів. Це допомагає працівникам зрозуміти потенційні ризики та вивчити правила та процедури безпеки, спрямовані на запобігання нещасних випадків.

Існують кілька видів інструктажів, кожен з яких має свою мету та специфіку:

вступний інструктаж: його проводять з прийнятими на роботу особами. Це базовий інформаційний курс, який надається незалежно від віку, стажу або посади працівника;

первинний інструктаж: зазвичай проводиться на місці роботи керівником відділу або майстром. Його мета - ознайомити працівника з конкретними правилами та процедурами безпеки, що стосуються його робочого місця;

повторний інструктаж: проводиться регулярно, зазвичай через кожні півроку, для всіх працівників. Це допомагає оновити їхні знання та нагадати про важливі аспекти безпеки.

позаплановий інструктаж: проводиться у випадках змін у правилах, технологічних процесах, придбанні нового обладнання або виявленні порушень правил техніки безпеки. Його мета - оперативно ознайомити працівників з новими або зміненими вимогами безпеки [24, 25].

Всі небезпечні та шкідливі фактори зазначені вище дозволять розробити ефективні заходи та навчання, що сприятиме підвищенню рівню охорони праці та безпеки на робочих місцях.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи була реалізована розробка проекту інформаційної системи каршерінгу автомобілів для фірми.

Проведено детальний аналіз предметної області, виявлені недоліки в існуючій системі управління, що послужило основою для сформулювання вимог. На основі вимог до системи розроблена концепція інформаційної системи, яка дозволяє досягти поставлених цілей, а саме підвищення ефективності роботи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Каршерінг проти класичної оренди автомобілів. URL: <https://narscars.com.ua/ua/blog/interesno/chem-otlichaetsya-klassicheskij-prokat-avtomobilej-ot-karsheringa>
2. Каршерінг – що це таке, як працює і навіщо потрібен. URL: https://termin.in.ua/karsherin#Vidi_Karseringa
3. Моделі. Типи моделей. Етапи побудови інформаційної моделі. URL: <http://surl.li/tpqin>
4. ДСТУ 34.601-90 "Інформаційна технологія. Комплекс стандартів на автоматизовані системи".
5. Документообіг. URL: <http://surl.li/kgfz>
6. Ременяк, Л.В. "Проектування інформаційних систем: Конспект лекцій". Одеса: Одеський державний екологічний університет, 2016. 152 с.
7. ГОСТ 19.701-90 "Єдина система програмної документації. Схеми алгоритмів, програм, даних і систем: умови позначення і правила виконання".
8. Марченко А.В. "Проектування інформаційних систем". 2016. 90 с.
9. Що таке моделювання даних? Типи (концептуальний, логічний, фізичний). URL: <http://surl.li/udiwu>
10. Моделювання даних URL: <https://www.maxzosim.com/data-modelling/>
11. ЩО TAKE MYSQL. URL: <https://freehost.com.ua/ukr/faq/wiki/chto-takoe-mysql/>
12. Засіб для роботи з базами даних DBEaver. URL: <https://ukr-technologies.blogspot.com/2020/03/dbeaver.html>
13. Що таке PHP? URL: <https://hyperhost.ua/uk/wiki/chto-takoe-php>
14. ЩО TAKE JQUERY? URL: <https://avada-media.ua/ua/jquery/>
15. Що таке CSS. URL: https://css.in.ua/article/shcho-take-html_10
16. Що таке Bootstrap? URL: https://w3schoolsua.github.io/bootstrap/bootstrap_get_started.html#gsc.tab=0

17. Що таке Figma: функції, інструменти та переваги. URL: <https://wezom.academy/ua/chto-takoe-figma-funktsii-instrumenty-ipreimuschestva/>

18. Охорона праці. URL: <https://www.who.int/health-topics/occupational-health#:~:text=Occupational%20health%20is%20an%20area,of%20workers%20in%20all%20occupations.>

19. Гігієна праці в деталях: вимоги виробничої санітарії до робочого місця. URL: <http://surl.li/ujwjm>

20. ЗАКОН УКРАЇНИ Про охорону праці. URL: <https://zakon.rada.gov.ua/laws/show/2694-12#Text>

21. ЗАКОН УКРАЇНИ Про дорожній рух. URL: <https://zakon.rada.gov.ua/laws/show/3353-12#Text>

22. Кодекс законів про працю України. URL: <https://zakon.rada.gov.ua/laws/show/322-08#Text>

23. Норми та правила з охорони праці. URL: <https://ukc.gov.ua/knowledge/normy-ta-vymogy-shhodo-ohorony-pratsi/>

24. Гігієна праці в деталях: шкідливі та небезпечні виробничі фактори, обов'язки роботодавця. URL: <https://pd.dsp.gov.ua/news/hihiiena-pratsi-v-detaliakh-shkidlyvi-ta-nebezpechni-vyrobnychi-factory-obov-iazky-robotodavtsia/>

25. Класифікація небезпечних і шкідливих виробничих факторів. URL: <https://oppb.com.ua/articles/klasyfikaciya-nebezpechnyh-i-shkidlyvyh-vyrobnychyh-faktoriv>

ДОДАТОК А

Інструкція користувача

При вході на сайт ви потрапляєте на головну сторінку, де можете ознайомитися з детальною інформацією про каршерінг та з його перевагами, який представлено на рисунку А.1.

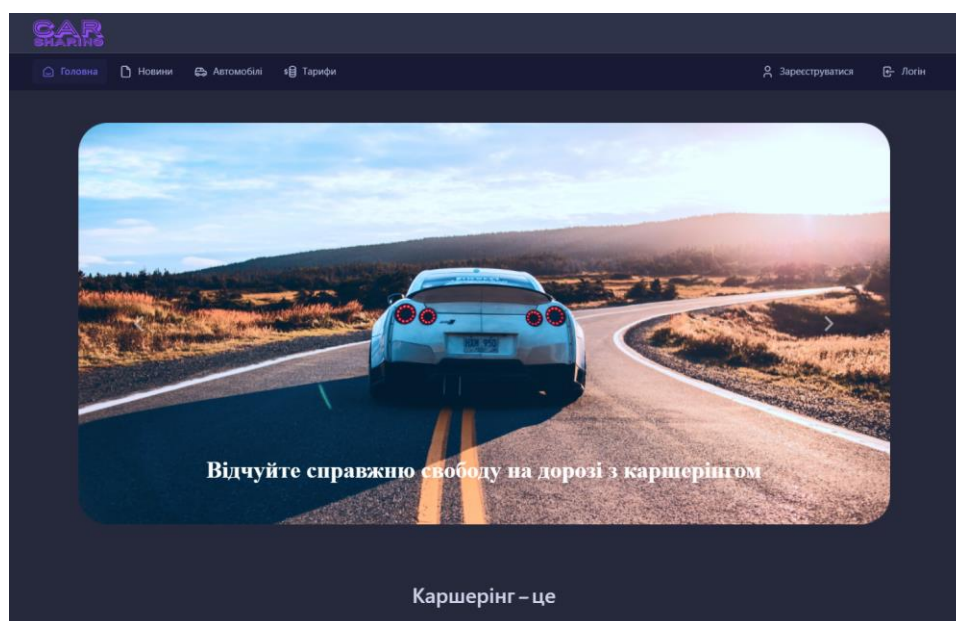


Рисунок А.1 – Головна сторінка

Вгорі сайту розташований навігаційний блок, який дозволяє швидко переходити на потрібну сторінку, що представлено на рисунку А.2.



Рисунок А.2 – Навігаційний блок

Далі є сторінка «Новини», де можна ознайомитися з цікавими статтями про автомобільну індустрію. Ви можете перемикати сторінки новин, як показано на рисунку А.3.

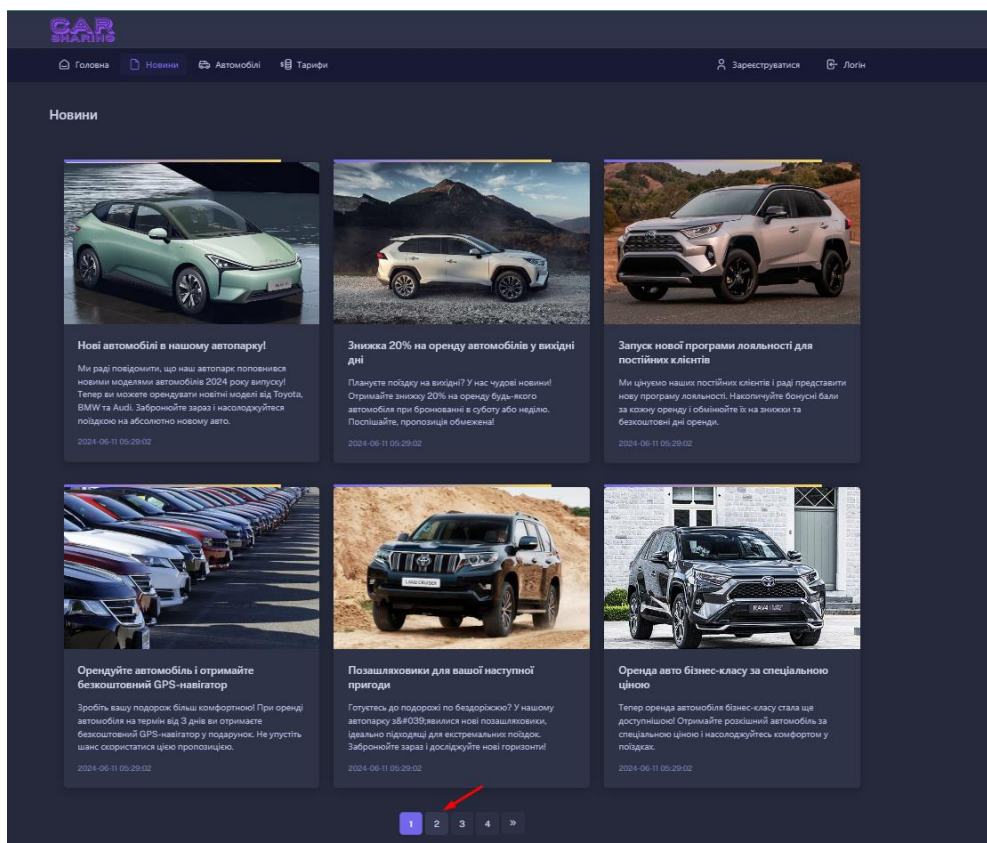


Рисунок А.3 – Сторінка «Новини»

Для ознайомлення з цінами на бронювання автомобілів потрібно перейти на сторінку «Тарифи» і обрати відповідний тарифний план, що представлено на рисунку А.4.

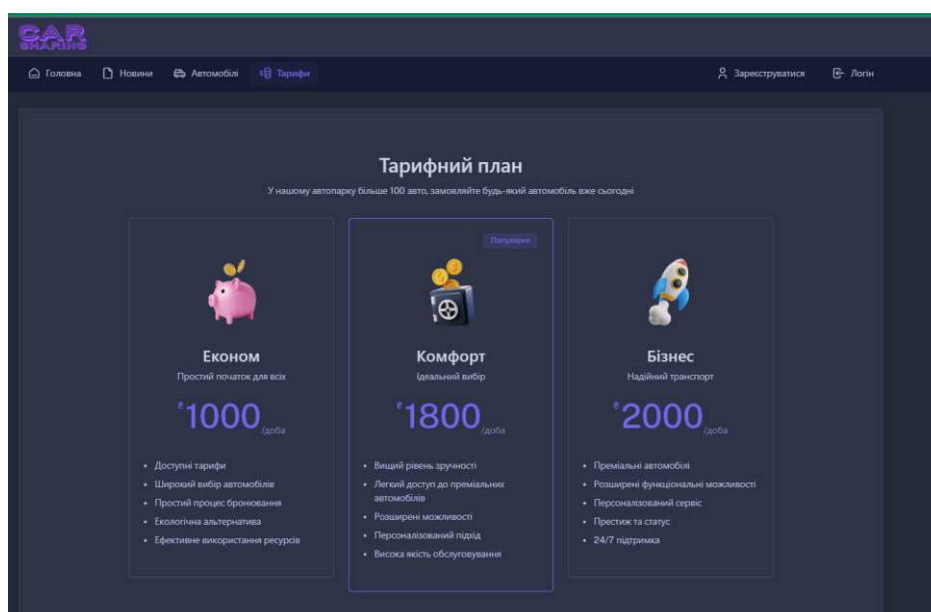


Рисунок А.4 – Сторінка «Новини»

Для перегляду каталогу машин перейдіть на сторінку «Автомобілі». Там є пошук по автомобілях з фільтром, де можна обрати виробника, тип кузова та категорію зі списку, який представлений на рисунку А.5.

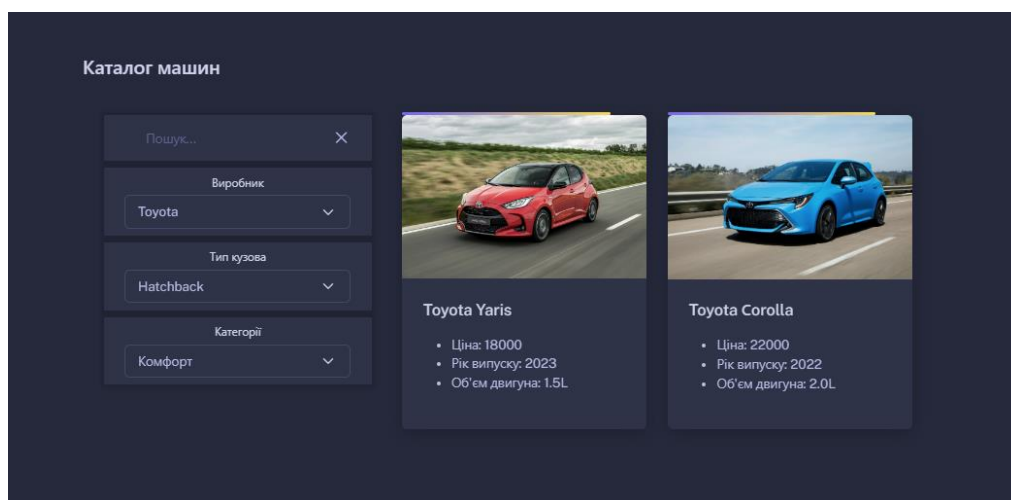
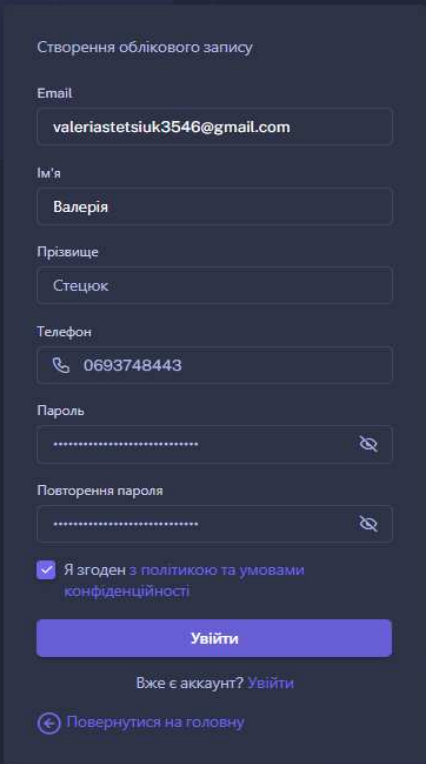


Рисунок А.5 – Сторінка «Автомобілі»

Для оформлення замовлення на автомобіль потрібно спочатку зареєструватися, вказавши електронну пошту, ім'я, прізвище, телефон, вигадати пароль і повторити його. Також потрібно підтвердити свою згоду з політикою та умовами конфіденційності. Якщо у вас вже є обліковий запис, ви можете написати пошту, а також пароль щоб увійти та перейти до вашого профілю користувача. Це все представлено на рисунках А.6, А.7.



Створення облікового запису

Email
valeriatetsiuk3546@gmail.com

Ім'я
Валерія

Прізвище
Стецюк

Телефон
0693748443

Пароль
.....

Повторення пароля
.....

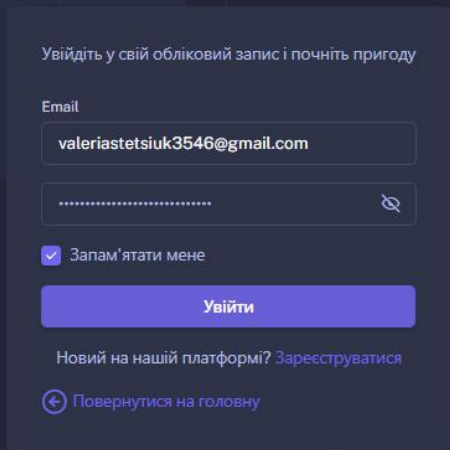
☒ Я згоден з політикою та умовами конфіденційності

Увійти

Вже є акаунт? Увійти

[← Повернутися на головну](#)

Рисунок А.6 – Створення облікового запису (реєстрація)



Увійдіть у свій обліковий запис і почніть пригоду

Email
valeriatetsiuk3546@gmail.com

.....

☒ Запам'ятати мене

Увійти

Новий на нашій платформі? Зареєструватися

[← Повернутися на головну](#)

Рисунок А.7 – Вхід до облікового запису

Після входження в обліковий запис на сторінці «Автомобілі» з'явиться під кожним автомобілем кнопка "Забронювати зараз". Після натискання на неї з'явиться модальне вікно для замовлення, яке представлено на рисунках А.8, А.9.

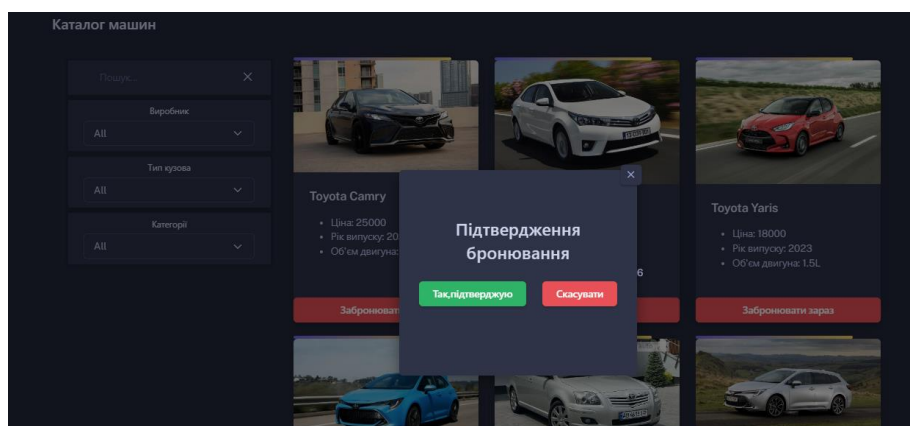


Рисунок А.8 – Підтвердження бронювання

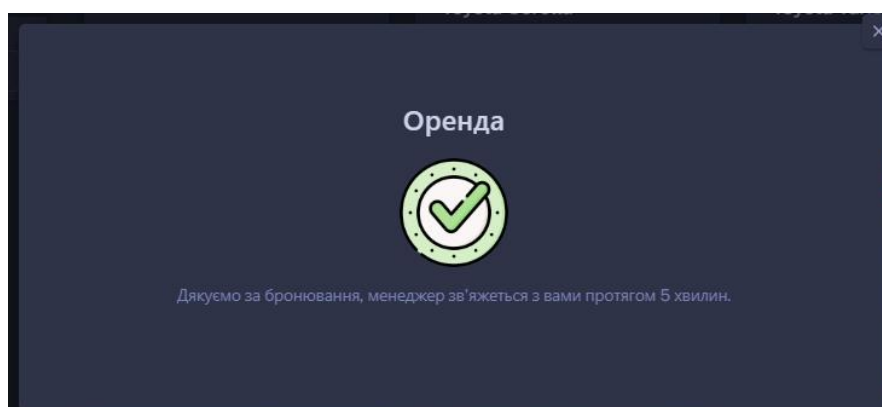


Рисунок А.9 – Успішне бронювання

На сторінці «Профіль користувача» ви зможете знайти всю вашу персональну інформацію, а також побачити останню активність. Якщо в процесі реєстрації ви допустили помилку або вказали неправильну інформацію, немає проблем – ви зможете легко її виправити. Крім того, тут є можливість завантажити своє фото для більш особистого вигляду профілю, що представлено на рисунку А.10.

Рисунок А.10 – Редагування профілю

Якщо ви раптом прагнете змінити свій пароль, переходьте до розділу "Безпека". Там вам потрібно буде вказати ваш поточний пароль, а потім придумати новий, повторивши його для підтвердження. Не забудьте зберегти внесені зміни, перш ніж залишити цю сторінку, це все представлено на рисунку А.11.

Рисунок А.11 – Зміна паролю

Також ви зможете додати свою карту до профілю, просто натиснувши на кнопку "Додати карту", це представлено на рисунку А.12.

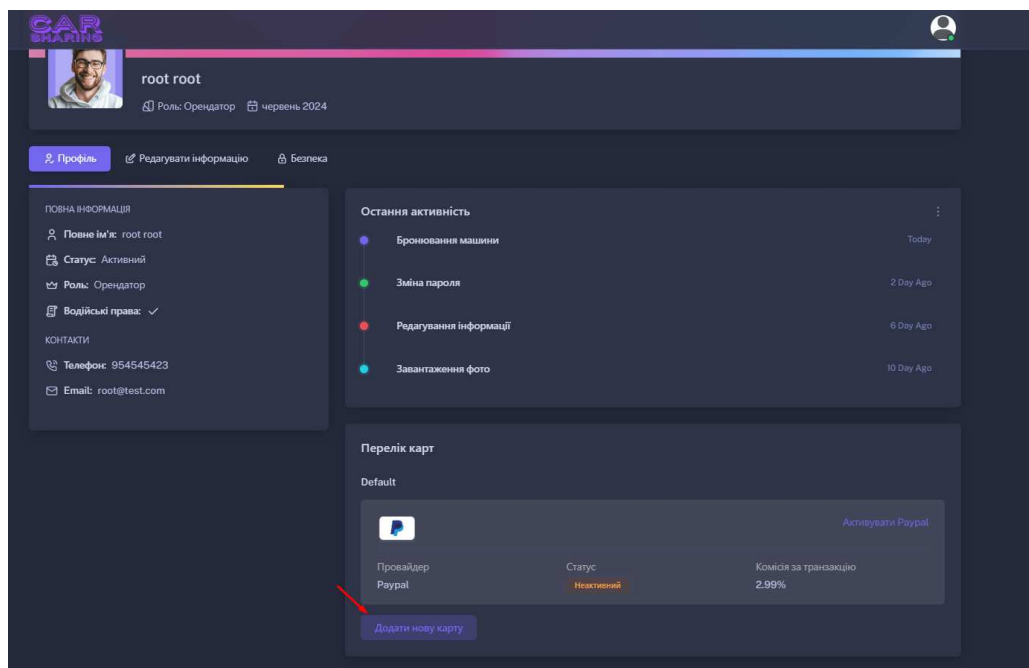


Рисунок А.12 – Додавання нової карти

ДОДАТОК Б

Технічне завдання на розробку

1. Призначення та мета створення сайту каршерінгу автомобілів фірми

Сайт каршерінгу автомобілів фірми призначений для надання зручного та швидкого доступу до послуг спільного користування автомобілями.

Метою роботи є розробка та впровадження інформаційної підсистеми каршерінгу автомобілів, яка дозволить удосконалити процеси управління автопарком, забезпечити зручність та швидкість обслуговування клієнтів, а також зменшити витрати та ризики.

2. Вимоги до сайту

Веб-додаток повинен бути інформаційною структурою і складатися з взаємопов'язаних розділів із чітко розмежованими функціями. Та має включати такі сторінки:

- головна;
- новини;
- автомобілі;
- тарифи;
- реєстрація;
- логін;
- профіль користувача.

Проект буде розроблений на основі дизайнерського макету, де продумана архітектура майбутнього функціоналу, щоб забезпечити його простоту та інтуїтивність для користувача. Дизайн повинен відповідати стандартам HTML5 та CSS3 і використовувати фреймворк Bootstrap. Деякі елементи інтерфейсу будуть реалізовані за допомогою JavaScript. Сайт буде написаний мовою PHP.

Функціональні вимоги:

- перегляд інформації про каршерінг;
- перегляд новин;

- перегляд тарифів;
- перегляд каталогу автомобілей;
- можливість бронювання машини;
- пошук автомобіля через фільтри;
- перегляд контактних даних;
- редагування своїх даних;
- зміна пароллю;
- видалення аккаунту;
- додавання свого фото;
- додавання карти.

3. Нефункціональні вимоги

Безпека: дані користувачів повинні бути захищені.

Юзабіліті: інтерфейс повинен бути дружнім до користувача, з чіткою навігацією та легко зрозумілими інструкціями.

Сумісність: сайт повинен коректно працювати на різних браузерах.

4. Технології

Бекенд: PHP з використанням фреймворка Laravel для забезпечення швидкої розробки та безпеки.

Фронтенд: HTML5, CSS3, Bootstrap для адаптивного дизайну, JavaScript (включаючи бібліотеки JQuery).

База даних: MySQL для зберігання даних користувачів, автомобілів, бронювань.

Система управління версіями: Git для координації командної роботи та відстеження змін у коді.

ДОДАТОК В

Фрагмент програмного коду

```

// Отримання списку машин
<?php
use \App\Http\Controllers\pages\CarsController;
Route::group([
    'prefix' => 'cars',
], function () {
    Route::get('show-all', [CarsController::class, 'index']) -
>name('pages-cars.all');
});

// Контролер, куди приходять запит на тримання списку машин
<?php
namespace App\Http\Controllers\pages;
use App\DTO\CarsDTO;
use App\Http\Controllers\Controller;
use App\Services\CarsService as Service;

/**@property Service $service*/
class CarsController extends Controller
{
    public function setNameService(): string
    {
        return Service::class;
    }
    public function index(CarsDTO $request)
    {
        $data_request = $request;
        $data = $this->service->getCars($data_request);

        return
view('content.pages.users.cars.show_all', compact('data'));

```



```

    }
}

// Отримання даних від Get запиту
<?php
namespace App\DTO;
use App\Http\Filters\Frontend\CarsFilter;
use App\Http\Requests\GetCarsRequest;
use App\Interfaces\DTOInterface;

class CarsDTO implements DTOInterface
{
    private CarsFilter $filter;
    private string $name;
    private int $type;
    private int $manufacturer;
    private int $price;
    private int $page;

    private int $category;
    public function __construct(GetCarsRequest $request, CarsFilter
$filter)
    {
        $this->filter = $filter;
        $this->name = $request->input('name', '');
        $this->type = $request->input('type', 0);
        $this->manufacturer = $request->input('manufacturer', 0);
        $this->price = $request->input('price', 100000);
        $this->page = $request->input('page', 1);
        $this->category = $request->input('category', 0);

    }
    public function getName(): string
    {
        return $this->name;
    }
}

```

```
public function getType(): int
{
    return $this->type;
}
public function getManufacturer(): int
{
    return $this->manufacturer;
}
public function getPrice(): int
{
    return $this->price;
}
public function getPage(): int
{
    return $this->page;
}
public function getCategory(): int
{
    return $this->category;
}
public function getArray(): array
{
    return [
        'name' => $this->getName(),
        'type' => $this->getType(),
        'manufacturer' => $this->getManufacturer(),
        'price' => $this->getPrice(),
        'page' => $this->getPage(),
        'category' => $this->getCategory(),
    ];
}
public function getFilter(): CarsFilter
{
    return $this->filter;
}}
```

// Створення білдеру для фільтрації даних при створенні запиту до БД

```
<?php
namespace App\Http\Filters\Frontend;
use App\Http\Filters\QueryFilter;
class CarsFilter extends QueryFilter
{
    public function fields(): array
    {
        $this->builder->leftJoin(
            'manufacturers',
            'manufacturers.id',
            '=',
            'cars.manufacturer_id'
        );
        return parent::fields();
    }
    public function name(string $name = ''): void
    {
        if (!empty($name)) {
            $escapedName = preg_quote($name, '/');
            $this->builder->where(function ($query) use ($escapedName) {
                $query->where('cars.model_name', 'REGEXP', $escapedName)
                    ->orWhere('manufacturers.name', 'REGEXP', $escapedName);
            });
        }
    }
    public function type(int $car_body_type_id): void
    {
        $this->builder->where('cars.car_body_type_id',
$car_body_type_id);
    }
    public function manufacturer(int $id): void
    {
        $this->builder->where('cars.manufacturer_id', $id);
    }
}
```

```

public function price(int $range = 0): void
{
    $this->builder->whereBetween('cars.price', [0, $range]);
}
public function category(int $id): void
{
    $this->builder->where('cars.car_body_type_id', $id);
}
}

```

// Сервіс, який отримує дані з контролера, слугує для обробки даних, якщо потрібно звернутися до БД то він викликає необхідний метод у репозиторія

```

<?php
namespace App\Services;
use App\DTO\CarsDTO;
use App\Http\Filters\Frontend\CarsFilter;
use App\Repository\CarsRepository as Repository;
class CarsService extends CoreService
{
    private Repository $repository;
    private CarsFilter $filter;
    public function __construct()
    {
        $this->repository = new Repository;
    }

    public function getCars(CarsDTO $data): array
    {
        $data = [
            'cars' => $this->repository->getCars($data),
            'manufacturers' => $this->repository->getManufacturers(),
            'getCardBodyTypes' => $this->repository->getCarBodyTypes(),
            'manufacturer' => $data->getManufacturer(),
            'categories' => $this->repository->getCategories(),
            'name' => $data->getName(),

```

```

        'type'                => $data->getType(),
        'price'               => $data->getPrice(),
        'category'            => $data->getCategory(),
    ];
    return $data;
}
protected function errors(): array
{
    return [
        0 => ['code' => 200, 'message' => 'Undefined error'],
    ];
}
}

// Репозиторій, що слугує для створення запитів до БД, використовуючи модель до
підключення до таблиці БД.
<?php
namespace App\Repository;
use App\DTO\CarsDTO;
use App\Models\CarBodyTypes;
use App\Models\Cars as Model;
use App\Models\Manufacturer;
use Illuminate\Support\Facades\Cache;
class CarsRepository extends CoreRepository
{
    public function getModelClass(): string
    {
        return Model::class;
    }
    public function getCars(CarsDTO $data): array
    {
        $data_base = Model::query()
            ->select([
                'cars.id',
                'cars.manufacturer_id',
                'cars.car_body_type_id',
            ]);
    }
}

```

```

        'cars.model_name',
        'cars.engine_capacity',
        'cars.images',
        'cars.number',
        'cars.year_of_issue',
        'cars.price',
        'cars.status_reservation',
        'cars.category',
        'manufacturers.name',
    ])
    ->filter($data->getFilter())
    ->paginate(
        Model::COUNT_RECORDS_IN_THE_PAGE,
        ['*'],
        'page',
        $data->getPage()
    )
    ->onEachSide(2);
    return $data_base->toArray();
}

public function getCategories(): array
{
    return Model::CATEGORIES;
}

public function getManufacturers(): array
{
    return Cache::remember('Manufacturers', 3600, function () {
        return Manufacturer::query()->get()->toArray();
    });
}

public function getCarBodyTypes(): array
{
    return Cache::remember('CarBodyTypes', 3600, function () {
        return CarBodyTypes::query()->get()->toArray();
    }); }
}

```

// Модель, що слугує для створення запитів до таблиці БД. В ній описано ім'я таблиці, та також є поля, які необхідно заховати при запиті, ще є відношення до інших таблиць.

```
<?php
namespace App\Models;
use App\Traits\Filterable;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\SoftDeletes;
class Cars extends Model
{
    use HasFactory, SoftDeletes, Filterable;
    protected $table = 'cars';
    public const CATEGORIES = [
        [
            'id' => 1,
            'name' => 'Економ'
        ],
        [
            'id' => 2,
            'name' => 'Комфорт'
        ],
        [
            'id' => 3,
            'name' => 'Преміум'
        ]
    ];
    protected $fillable = [
        'id',
        'manufacturer_id',
        'car_body_type_id',
        'model_name',
        'engine_capacity',
        'images',
        'price',
        'number',
    ];
```

```

        'category',
        'year_of_issue',
        'status_reservation',
        'deleted_at',
        'created_at',
        'updated_at',
    ];
/**
 * The attributes that should be hidden for serialization.
 *
 * @var array<int, string>
 */
protected $hidden = [
    'created_at',
    'updated_at',
    'deleted_at',
];
public const COUNT_RECORDS_IN_THE_PAGE = 9;
public function manufacturer()
{
    return $this->hasOne(Manufacturer::class, 'id',
'manufacturer_id');
}

    // Шаблон, який використовується для формування html коду, тут відбувається
відображення сторінки машин.
@php
$configData = Helper::appClasses();
@endphp
@extends('layouts/horizontalLayout')
@section('title', __('Cars'))

@section('content')
    <h4 class="py-3 mb-4">
        <span class="text-muted fw-light"></span> @lang('Каталог машин')
    </h4>

```



```

<div class="container">
  <div class="row">
    <div class="col-md-3">
      <div class="left-block">
        <x-search-form>
          <x-ui.form action="{{route('pages-cars.all')}}"
method="get">
            <div class="d-flex h-100 flex-column">
              <div class="container-xxl container navbar navbar-
expand-xl navbar-detached align-items-center bg-navbar-theme">
                <span class="twitter-typeahead" style="position:
relative; display: inline-block;">
                  <input type="text" class="form-control
search_cars_input container-xxl border-0 tt-input"
                    placeholder="@lang('Search')..."aria-label="Search..."
autocomplete="off" spellcheck="false"
                    dir="auto" style="position: relative; vertical-align: top;"
name="name" value="{{ $data['name'] }}">
                  <pre aria-hidden="true" style="position: absolute; visibility:
hidden; white-space: pre; font-family: &quot;Public Sans&quot;; -
apple-system, BlinkMacSystemFont, &quot;Segoe UI&quot;;, Oxygen,
Ubuntu, Cantarell, &quot;Fira Sans&quot;;, &quot;Droid Sans&quot;;,
&quot;Helvetica Neue&quot;;, sans-serif; font-size: 15px; font-style:
normal; font-variant: normal; font-weight: 400; word-spacing: 0px;
letter-spacing: 0px; text-indent: 0px; text-rendering: auto; text-
transform: none;"></pre><div class="tt-menu navbar-search-suggestion
ps" style="position: absolute; top: 100%; left: 0px; z-index: 100;
display: none;"><div class="tt-dataset tt-dataset-pages"></div><div
class="tt-dataset tt-dataset-files"></div><div class="tt-dataset tt-
dataset-members"></div><div class="ps__rail-x" style="left: 0px;
bottom: 0px;"><div class="ps__thumb-x" tabindex="0" style="left:
0px; width: 0px;"></div></div><div class="ps__rail-y" style="top:
0px; right: 0px;"><div class="ps__thumb-y" tabindex="0" style="top:
0px; height: 0px;"></div></div></div></span>

```

```

        <i class="ti ti-x ti-sm search-toggler cursor-
pointer clear-data-search"></i>

        </div>                                <div class="d-flex flex-column
container-xxl navbar mt-2 navbar-expand-xl navbar-detached align-
items-center bg-navbar-theme">

                <label for="manufacturer_field" class="form-
label">@lang('Manufacturer')</label>

                <select class="form-select manufacturers__cars"
id="manufacturer_field" aria-label="Default select example">

                        <option value="0">@lang('All')</option>

                        @foreach($data['manufacturers'] as
$item_manufacturers)

                                <option {{ $data['manufacturer'] ==
$item_manufacturers['id'] ? 'selected' : '' }}
value="{{ $item_manufacturers['id'] }}">{{ $item_manufacturers['name'] }}
</option>

                        @endforeach </select></div>


        <div class="d-flex flex-column container-xxl navbar
mt-2 navbar-expand-xl navbar-detached align-items-center bg-navbar-
theme">

                <label for="card_body_types" class="form-label">@lang('Body
type')</label>

                <select class="form-select card_body_types"
id="card_body_types" aria-label="Default select example">

                        <option value="0">@lang('All')</option>

                        @foreach($data['getCardBodyTypes'] as $item_body_type)<option {{
$data['type'] == $item_body_type['id'] ? 'selected' : '' }}
value="{{ $item_body_type['id'] }}">{{ $item_body_type['name']
}}</option>

                        @endforeach

                </select>    </div>

        <div class="d-flex flex-column container-xxl navbar
mt-2 navbar-expand-xl navbar-detached align-items-center bg-navbar-
theme">

```

```

        <label for="categories" class="form-
label">@lang('Categories')</label>
        <select class="form-select categories_filter"
id="categories" aria-label="Default select example">
            <option value="0">@lang('All')</option>
@foreach($data['categories'] as $item_categories)
            <option {{ $data['category'] == $item_categories['id'] ?
'selected' : '' }} value="{{ $item_categories['id'] }}">{{
$item_categories['name'] }}</option>
            @endforeach
        </select> </div></div>
    </x-ui.form>
</x-search-form>
</div>
</div>
<div class="col-md-8">
    <div class="row row-cols-1 row-cols-md-3 g-4 mb-5">
        @foreach($data['cars']['data'] as $key)
            <x-cars-card
                id="{{ $key['id'] }}"
                model="{{ $key['model_name'] }}"
                engineCapacity="{{ $key['engine_capacity'] }}"
                images="{{ $key['images'] }}"
                number="{{ $key['number'] }}"
                yearOfIssue="{{ $key['year_of_issue'] }}"
                price="{{ $key['price'] }}"
                status_reservation="{{ $key['status_reservation'] }}"
                category="{{ $key['category'] }}"
                name="{{ $key['name'] }}">
                @if(!auth()->guest())<button data-id="{{ $key['id'] }}"
                    class="btn btn-danger btn-buy-now waves-effect waves-light rent-
cars pay_confirm_{{ $key['id'] }}">@lang('Book now')
                </button>
                <button
                    data-id="{{ $key['id'] }}"

```

```

        class="btn btn-buy-now waves-effect waves-light
rent-cars d-none pay_confirm_wait_{{$key['id']}}">@lang('Wait')
    </button>@endif
    </x-cars-card>
    @endforeach
</div>
@if(!empty($data['cars']['data']))
    <x-ui.pagination :links="$data['cars']['links']" />
@else
    <h5>@lang('Sorry, cars not found')</h5>
@endif
</div>
</div>
</div>
    @include('_partials/_modals/model-select-car-rent')
    @include('_partials/_modals/model-select-car-confirmation')
    @vite(['resources/assets/js/cars_page.js'])
@endsection

```