

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Сумський державний університет**

Факультет електроніки та інформаційних технологій

Кафедра комп'ютерних наук

«До захисту допущено»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

\_\_\_\_\_  
(підпис)

червень 2024 р.

**КВАЛІФІКАЦІЙНА РОБОТА**

**НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВР**

зі спеціальності 122 - Комп'ютерних наук,  
освітньо-професійної програми «Інформатика»  
на тему: «Адаптивний сайт для салону електромобілів»  
здобувача групи ІН - 03 Родіна Романа Владиславовича

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

Роман РОДІН

\_\_\_\_\_  
(підпис)

Керівник,  
доцент кафедри комп'ютерних наук,  
кандидат фізико-математичних наук

Надія ТИРКУСОВА  
\_\_\_\_\_

**СУМИ – 2024**

**Сумський державний університет**  
Факультет електроніки та інформаційних технологій

Кафедра комп'ютерних наук

«Затверджую»

В.о. завідувача кафедри

Ігор ШЕЛЕХОВ

(підпис)

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ**

**на здобуття освітнього ступеня бакалавра**

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»  
здобувача групи ІН-03 Родіна Романа Владиславовича

1. Тема роботи: «Адаптивний сайт для салону електромобілів»  
затверджую наказом по СумДУ від від 22 квітня 2024р. № 0414-VI
2. Термін здачі здобувачем кваліфікаційної роботи до 01 червня 2024 року
3. Вхідні дані до кваліфікаційної роботи \_\_\_\_\_
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їй належить розробити)  
1) Аналіз проблеми предметної області, постановка й формування завдань дослідження.  
2) Огляд технологій, що використовуються для прогнозування курсу валют. 3) Розробка інтелектуальної системи з прогнозування курсу валют. 4) Аналіз результатів.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) \_\_\_\_\_
6. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання «\_\_\_» \_\_\_\_\_ 20\_\_ р.

Завдання прийняв до виконання \_\_\_\_\_  
(підпис)

Керівник \_\_\_\_\_  
(підпис)

**КАЛЕНДАРНИЙ ПЛАН**

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	<i>Аналіз проблеми предметної області, постановка й формування завдань дослідження</i>	06.05.24-9.05.24	
2	<i>Огляд технологій, що використовуються для розробки веб сайту</i>	10.05.24-15.05.24	
3	<i>Розробка сайту салону електромобілів</i>	16.05.24-21.05.24	
4	<i>Аналіз отриманих результатів</i>	22.05.24-27.05.24	
5	<i>Оформлення пояснювальної записки до кваліфікаційної роботи</i>	28.05.24-31.05.24	

Здобувач вищої освіти \_\_\_\_\_

Керівник \_\_\_\_\_

## АНОТАЦІЯ

**Записка:** 45 стр., 26 рис., 1 додаток, 8 використаних джерел.

**Обґрунтування актуальності теми роботи** – Тема кваліфікаційної роботи є актуальною, оскільки створення сайту для автосалону електромобілів різних брендів з використанням сучасних технологій сприяє просуванню ідеї екологічного транспорту в маси.

**Об'єкт дослідження** — Процес продажу електромобілів різних брендів.

**Мета роботи** — Розробка інтернет сайту салону електромобілів

**Методи дослідження** — алгоритми створення веб сайтів з використанням певного стеку технологій

**Результати** — Розроблено сайт автосалону який дозволяє клієнту ознайомитися з авто, подивитися на нього у 3д, та оформити заявку на тест драйв

ВЕБ САЙТ, АВТО, HTML, CSS, JAVASCRIPT,PHP,MYSQL.

## ЗМІСТ

Вступ.....	5
1 Інформаційний огляд .....	6
1.1 Веб-сайт для автосалону електромобілів.....	6
1.2 Адаптивність сайту .....	7
1.3 Аналіз конкурентів.....	8
1.4 Чому потрібен адаптивний веб-сайт .....	9
1.5 Що таке .htaccess і навіщо він потрібен .....	9
1.6 Аналіз предметної області .....	10
1.7 Плюси і мінуси адаптивного дизайну .....	11
1.8 Чому саме PHP .....	12
1.9 Постановка задачі.....	14
2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ .....	16
2.1 Вибір бази даних .....	16
2.2 Вибір місця для зберігання коду.....	17
3 Програмна реалізація .....	21
3.1 Розробка прототипу сторінок веб-сайту.....	21
3.2 Створення візуальної частини сайту .....	25
3.3 Проектування бази даних.....	30
3.4 Розробка бази даних.....	31
Висновки.....	37
Список використаних джерел .....	38
ДОДАТОК.....	39

## ВСТУП

**Актуальність.** Тема кваліфікаційної роботи є актуальною, оскільки створення сайту для автосалону електромобілів різних брендів з використанням сучасних технологій сприяє просуванню ідеї екологічного транспорту в маси.

**Об'єкт дослідження.** Процес продажу електромобілів різних брендів.

**Предмет дослідження.** Методологія створення сайту та керування ним для продажу авто.

**Гіпотеза.** Створення сайту та керування ним реалізоване за допомогою спеціального стеку технологій, що краще за все підходить саме для подібних завдань.

**Новизна.** Створення подібного сайту створить можливість пошири вплив електротранспорту, що в свою чергу допоможе навколишній середовищу.

**Структура.** Дана робота складається зі вступу, аналітичного огляду, постановки задачі, вибір методу розв'язання поставленої задачі, опису програмного забезпечення, висновків та списку використаних джерел.

# 1 ІНФОРМАЦІЙНИЙ ОГЛЯД

## 1.1 Веб-сайт для автосалону електромобілів

Веб-сайт для Автосалону Електромобілів відіграє важливу роль у популяризації та доступності екологічно чистих транспортних рішень. Він сприяє зеленій революції в автомобільній промисловості та надає користувачам можливість обирати серед інноваційних електричних автомобілів.

У сучасному світі, де екологічність та інновації стають ключовими пріоритетами, Веб-сайт для автосалону електромобілів є важливим ресурсом для тих, хто прагне зробити свій внесок у захист навколишнього середовища та перехід до чистішої енергетики.

Створення веб-сайту для автосалону - це важлива тема, оскільки це допомагає покращити його конкурентоспроможність на ринку та збільшити прибуток. Наявність власного сайту дає їм можливість:

- Збільшити кількість потенційних клієнтів;
- Підвищити репутацію та впізнаваність бренду;
- Збільшити обсяг продажів;
- Забезпечити зручний доступ клієнтам до інформації про товари та послуги;
- Зменшити розходи на рекламу

Веб-сайт надає інформацію про різні типи електричних автомобілів, їхні характеристики та переваги. Він також пропонує користувачам можливість порівняти ціни на електричні автомобілі та знайти найближчий дилерський центр.

Створення успішного веб-сайту складається з декількох етапів:

1. Збір інформації – перед тим як створити сайт, потрібно зібрати інформацію. Зрозуміти, які цілі повинен виконувати веб-сайт та цільову аудиторію;
2. Планування – після збору інформації потрібно чітко спланувати сайт, створити навігаційну схему веб-платформи, яка допоможе зрозуміти, як

користувач може переходити з однієї сторінки на іншу та створити прототипи сторінок для того, щоб уявити як буде розміщатися контент сайту. Це крок має вирішальне значення, щоб створити привабливий сайт, на якому легко орієнтуватися;

3. Дизайн – на цьому етапі створюється дизайн. Він має бути з огляду на зручність використання, наскільки простий та лаконічний дизайн, настільки приємно взаємодіяти з веб-сайтом. Дизайнер обов'язково повинен врахувати дизайн для пристроїв з різними розмірами екрану;
4. Верстка – після того як дизайн був створений, розробник починає підготовлювати проект, уточнює деталі з дизайнером та починає працювати. Бажано, щоб розробник працював за методологією pixel-perfect – кодування сайту відповідно до веб-дизайну піксель за пікселем.
5. Бази даних- після завершення верстки можна починати робити серверну частину нашого сайту починаючи з бд
6. Бекенд – написання серверної частини коду для коректної роботи усіх частин сайту разом

## **1.2 Адаптивність сайту**

Адаптивний підхід передбачає, що дизайн сайту має адекватно реагувати на різні пристрої та умови використання, такі як розмір екрану, платформа та орієнтація.

При зміні пристрою, наприклад, з ноутбука на планшет, сайт повинен автоматично адаптуватися до роздільної здатності пристрою. Це дозволяє уникнути необхідності в окремому проектуванні для кожного нового пристрою.

Однак, ключовою перевагою адаптивних веб-сайтів є можливість забезпечити зручний та оптимальний користувацький досвід на будь-якому пристрої. Це досягається завдяки адаптації контенту, функціональності та дизайну до потреб конкретного користувача. [2]

У даній роботі будуть розглянуті основні аспекти: важливість адаптивних веб-сайтів, їх позитивні та негативні сторони, принципи та технології розробки.

### 1.3 Аналіз конкурентів

В епоху технологій і цифрової інформації веб-сайт втілюється в справжню візитівку бізнесу, а в асортименті автосалонів електромобілів він відіграє ще більш вагомую роль. Аналіз веб-сайтів конкурентів є невід'ємною частиною розробки стратегії для власного автосалону. Давайте розглянемо деякі ключові аспекти цього аналізу. Для аналізу було обрано 2 сайти автосалонів:

BMW – компанія має чудовий цифровий портал, що надає відвідувачам доступ до широкого спектру інформації про автомобілі, послуги та інновації, які пропонує німецький автовиробник. На головній сторінці сайту зазвичай розміщені актуальні новини, анонси подій та найбільш важливі огляди моделей

Сайт має великий спектр особливостей, такі як- модельний ряд, технології та інновації, акції та спеціальні пропозиції, сервіс та підтримка, конфігуратор

Porsche – сайт компанії це вражаюча онлайн-платформа, що пропонує всебічний огляд інноваційних автомобілів Porsche та їх характеристик. За допомогою сучасного дизайну та інтуїтивного інтерфейсу, сайт дозволяє користувачам досліджувати величезний асортимент автомобілів, від елегантних спорткарів до практичних SUV.

На сайті можна знайти відомості про різні моделі Porsche, включаючи їх технічні характеристики, фотографії, ціни та можливі конфігурації. Крім того, він надає інформацію про різноманітні послуги, які надає компанія, такі як сервіс та обслуговування автомобілів, можливості налаштування та інші спеціальні програми для власників.

Технологічний прогрес Porsche підкреслюється на сайті через використання відеоматеріалів, інтерактивних додатків та віртуальних турів по автомобільних новинках. Завдяки цьому сайт надає можливість користувачам поглибитися у світ Porsche та відчувати себе часткою ексклюзивного досвіду виробника автомобілів вищого класу.

Загалом, сайт компанії Porsche є відмінним ресурсом для тих, хто цікавиться автомобільною індустрією, бажає ознайомитися з найсучаснішими



технологіями та дизайном, або планує здійснити придбання високоякісного та ексклюзивного автомобіля.

#### **1.4 Чому потрібен адаптивний веб-сайт**

З кожним днем кількість мобільних користувачів Інтернету зростає. Відсоток користувачів мобільних пристроїв перевищив кількість користувачів настільних ПК у всьому світі в Інтернеті. У 2019 році Google підрахувала, що 61% усіх пошукових запитів надходили від людей, які шукали на своїх телефонах.

Справа в тому, що Google дивиться на мобільну версію вашого веб-сайту, щоб визначити, яке місце він повинен мати в результатах пошукової системи. Підсумовуючи: поганий мобільний веб-сайт = низький рейтинг = менший трафік.

Коли мова йде про веб-дизайн, простота справді найкраща. Багато веб-сайтів страждають від надскладних елементів, таких як навігація. Потрібно думати про те, як веб-сайт буде сприйматися користувачем, а коли мова йде про користувача, є просте правило - якщо він не зрозуміє, він піде. Ось чому зараз, як ніколи, важливо привернути увагу користувача, використовуючи простий і зрозумілий макет сайту.

#### **1.5 Що таке .htaccess і навіщо він потрібен**

.htaccess і його значення: Для розміщення сайтів в Інтернеті потрібен веб-сервер, який працює із запитамі користувачів і вирішує, як реагувати на запитуваний ресурс - відобразити його, зробити перенаправлення або повернути відповідь про відмову. Найпоширенішим є Apache. Він налаштовується за допомогою так званих конфігураційних текстових файлів з директивами і правилами обробки запитів. Центральний конфігураційний файл Apache називається `httpd.conf`, або `apache.conf`, все залежить від операційної системи.

Теоретично, цей файл повинен містити команди, але він не завжди доступний, наприклад, на віртуальному хостингу, де на одному сервері працюють сотні сайтів. На таких серверах небезпечно надавати всім користувачам доступ до центрального конфігураційного файлу, оскільки це

дозволяє будь-кому змінювати налаштування інших користувачів. Тут також вступає в гру файл `.htaccess`. Директиви в `.htaccess` застосовуються тільки до папки, в якій він розміщений, і всіх її підпапок. Іншими словами, цей файл дозволяє вам створити власні налаштування для Apache для цільових папок за вашим вибором.

У цьому випадку хостинг-провайдер має доступ до основного файлу, тоді як клієнт має можливість змінювати налаштування сервера для свого акаунта через `.htaccess`. Значення `.htaccess` полягає в тому, що це додатковий конфігураційний файл для Apache. Це дозволяє встановлювати параметри сервера для певних папок, не втручаючись в основний конфігураційний файл.

## 1.6 Аналіз предметної області

Бази даних дійсно є важливими для багатьох аспектів сучасного життя, включаючи роботу веб-додатків. Давайте розглянемо деякі ключові поняття та їх взаємозв'язок:

### 1. Файл:

- Файл можна розглядати як логічний запис фактів або ідей. Він може містити будь-яку інформацію, від тексту до зображень.
- Файли зберігаються на диску і можуть бути доступні для читання або запису.

### 2. База даних:

- База даних містить інформацію, подану таким чином, щоб забезпечити швидкий і гнучкий доступ до неї.
- Бази даних можуть містити структуровані дані (наприклад, таблиці зі стовпцями та рядками) або некеровані дані (наприклад, документи JSON).
- Вони використовуються для зберігання інформації, такої як реєстраційні дані користувачів, товари, замовлення тощо.

### 3. Реляційні бази даних (RDBMS):

- Це тип баз даних, в яких дані організовані у вигляді таблиць зі стовпцями та рядками.
- Вони використовують мову SQL для запитів та зв'язків між таблицями.

#### 4. Об'єктно-реляційні бази даних (ORDBMS):

- Це розширення реляційних баз даних, яке дозволяє зберігати об'єкти (наприклад, класи) разом зі звичайними даними.

#### 5. NoSQL бази даних:

- Це альтернатива реляційним базам даних, яка дозволяє зберігати некеровані дані.
- NoSQL бази даних можуть бути документ-орієнтованими, ключ-значеннями, колоночними або графовими.

Загалом, бази даних є невід'ємною частиною розробки програмного забезпечення, і вони допомагають зберігати, організовувати та отримувати доступ до даних.

### 1.7 Плюси і мінуси адаптивного дизайну

Завдяки якісній адаптації, сторінка сайту завжди відображається коректно, незалежно від розміру екрана. Це сприяє покращенню позицій ресурсу в пошукових системах, оскільки враховуються як динамічний показ, так і мобільна версія. Крім того, мобільна видача можлива лише за умови наявності адаптивного дизайну, що позитивно впливає на поведінкові показники користувачів. Проте є й недоліки.

Адаптивний дизайн забезпечує правильне відображення контенту, але потреби мобільних користувачів не завжди враховуються повною мірою. Ця група користувачів має інші вимоги, оскільки переглядає контент швидше. Неправильна адаптація може негативно вплинути на сприйняття десктопної версії сайту. Наприклад, великі яскраві кнопки, які добре виглядають на смартфонах, можуть виглядати недоречно на екрані ноутбука. Аналогічно,

великий шрифт, який гармонійно виглядає на планшеті, може бути незрозумілим на десктопі. У десктопній версії краще сприймається текст, ніж іконки з зображеннями, а повноекранні зображення можуть бути зайвими.

Адаптивний дизайн не впливає на швидкість завантаження сторінки, оскільки код для всіх версій завантажується однаковий, а різниця полягає лише в його відображенні. Надлишкове навантаження на процесор смартфона виникає через великий обсяг завантажуваної інформації та повнорозмірні зображення, які стискаються лише після завантаження.

Сайти з адаптивною версткою не надають користувачам вибору: вони не можуть перейти з мобільної версії на стандартну. [4]

## **1.8 Чому саме PHP**

Чому PHP ідеально підходить для створення серверної частини вашого веб-сайту[5]:

### **1. Широка популярність та підтримка:**

- PHP - одна з найпоширеніших мов програмування для веб-розробки, яку використовують мільйони розробників по всьому світу.
- Це мова, що лежить в основі CMS WordPress, яка використовується для мільйонів веб-сайтів.
- Завдяки такій популярності PHP має величезну та активну спільноту розробників, що гарантує доступність безлічі ресурсів для навчання та підтримки.

### **2. Простота вивчення та швидкість освоєння:**

- Синтаксис PHP простий та зрозумілий, що робить його доступним навіть для початківців.
- Завдяки невеликій кількості ключових концепцій, вивчення PHP відбувається значно швидше, ніж інших мов.

### **3. Безпроблемна інтеграція з веб-серверами:**

- PHP тісно інтегрується з популярними веб-серверами, такими як Apache та Nginx, що спрощує розгортання веб-додатків.
- Більшість хостинг-провайдерів підтримують PHP за замовчуванням, роблячи його зручним вибором для розробки.

#### **4. Багатофункціональність та універсальність:**

- PHP володіє широким набором вбудованих функцій та бібліотек, які охоплюють широкий спектр завдань веб-розробки.
- Від роботи з базами даних до обробки форм та файлів - PHP надає все необхідне для створення функціональних веб-додатків.

#### **5. Розширюваність та гнучкість:**

- Завдяки безлічі фреймворків та розширень, PHP можна легко розширювати та адаптувати до потреб будь-якого проекту.
- Популярні фреймворки, такі як Laravel, Symfony та CodeIgniter, значно спрощують та пришвидшують процес розробки.

#### **6. Висока продуктивність та масштабованість:**

- PHP відомий своєю високою продуктивністю, особливо при правильному налаштуванні та використанні з кешуванням OPcache.
- Це робить його чудовим вибором для веб-сайтів з високим трафіком та складними веб-додатками.

#### **7. Безкоштовність та відкритість:**

- PHP - це безкоштовна мова з відкритим кодом, що робить її доступною для розробників будь-якого рівня.
- Відкритий код гарантує прозорість та постійне вдосконалення мови завдяки внеску спільноти.

#### **8. Велика кількість ресурсів та навчальних матеріалів:**

- Завдяки своїй популярності, PHP має безліч доступних онлайн-ресурсів, документації та навчальних матеріалів.
- Це робить процес навчання та пошуку допомоги легким та швидким.

Зважаючи на всі ці переваги, PHP безсумнівно є чудовим вибором для розробки серверної частини вашого веб-сайту. Його простота, гнучкість, потужність та широка підтримка роблять його ідеальним інструментом як для досвідчених розробників, так і для початківців.

## **1.9 Постановка задачі**

**Мета:** розробити веб сайт для автосалону для збільшення обсягів продажу електромобілів

**Задачі:**

### **1. Дослідження ринку:**

- Вивчати ринок автомобілів і веб-сайти автосалонів
- Проаналізувати конкурентну ситуацію

### **2. Визначення функціональних вимог:**

- Розробити зручний інтерфейс для того, щоб шукати й купувати автомобілі.
- Впровадити систему обробки заявок

### **3. Дизайн:**

- Створити привабливий дизайн
- Покращити користувацький досвід за допомогою принципів UX-дизайну.

### **4. Розробка та інтеграція:**

- Реалізувати каталог автомобілів
- Розробити систему обліку замовлень
- Впровадити пошукову систему

### **5. Тестування та вдосконалення:**

- Провести тестування функціоналу

- Виявити та виправити помилки
- Скоригувати потрібно на основі результатів тестування.

#### **6. Розгортання та підтримка:**

- Розгорнути онлайн-платформу на хостингу
- Забезпечити підтримку та обслуговування в галузі техніки.

#### **7. Оцінка результатів:**

- Проаналізувати ефективність роботи платформи
- Внести корективи для подальшого розвитку

#### **8. Переваги:**

- Просування екологічних авто
- Позитивний вплив на економіку

Виконання цих завдань сприятиме успішному створенню сайту для продажу електромобілів, який залучатиме нових клієнтів та допомагатиме навколишнього середовища

## 2 ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ

### 2.1 Вибір бази даних

Сучасний цифровий світ керується даними. Кожен клік, пошуковий запит або покупка - це інформація, яку потрібно зберігати та обробляти. І саме база даних є тим місцем, де ця інформація розміщується. Без них не могли б працювати наші улюблені веб-сайти, додатки для смарт-пристроїв та комп'ютерні програми. Але що таке база даних? Це впорядкована колекція даних. У великих масштабах бази даних можуть містити терабайти інформації та обробляти мільйони запитів за секунду. Вони дійсно є ядром сучасних додатків і веб-сайтів. Однією з найважливіших частин роботи з базами даних є використання систем керування базами даних (СКБД). Більшість баз даних використовуються для зберігання, управління, оновлення та пошуку даних. Існує багато різновидів СУБД, але тут ми познайомимося з трьома найвідомішими, а саме: MySQL, PostgreSQL та MongoDB. Ці системи були обрані, оскільки вони широко популярні в індустрії та мають різні характеристики, які допоможуть вам зрозуміти широкий спектр можливостей, коли справа доходить до управління базами даних. Розглянемо докладніше кожен з них.

MySQL - одна з найпопулярніших систем управління базами даних. Вона забезпечує швидку та надійну роботу і має багато можливостей. У цій системі використовується структурована модель даних зі структурами таблиць, де кожна з них має рядки (записи) і стовпці (поля). У рядках зберігаються дані, а в стовпчиках описуються дані. У MySQL використовується мова SQL (Structured Query Language - мова структурованих запитів), яка дозволяє створювати, змінювати та отримувати дані з бази даних.

PostgreSQL - популярна СУБД, яка використовується великими корпораціями для реалізації масштабних проектів. Вона має високу масштабованість, гнучкість і підтримує широкий спектр даних. З одного боку, PostgreSQL також являє собою



систему структурованих даних і SQL, але вона також має об'єктно-реляційні можливості, що дозволяють працювати з даними як з об'єктами в програмуванні. PostgreSQL здатна виконувати транзакції, глибоко вкладені підзапити та математичні функції як складні операції.

MongoDB - це інструмент управління базами даних типу NoSQL, що дуже відрізняється від MySQL і PostgreSQL, які, в свою чергу, є сховищами на основі SQL. Модель, яку використовує MongoDB - це дані з'єднань. Замість класичних таблиць і стовпців тут застосовується підхід, схожий на структуру даних JSON, що дозволяє зберігати дані у вільній формі. В даному випадку дані зберігаються в форматі Binary JSON, а, отже, запити до бази даних можна давати в формі, близькій до JSON. Це дуже зручно для розробників, які звикли працювати з JavaScript. MySQL, PostgreSQL та MongoDB - потужні та гнучкі системи управління базами даних, кожна з яких має свої особливості та безліч переваг.

Звичайно, поняття найкращої з них залежить від особливостей вашого проекту: MySQL та PostgreSQL - це технології, які мають зберігати структуровані дані. У свою чергу, MongoDB буде кращою в роботі із завданнями, які вимагають гнучкої структури даних, великих обсягів даних або коли дані часто змінюються. Що стосується веб-додатків, то MySQL - це шлях, яким слід йти, коли швидкість повинна поєднуватися з надійністю. [\[6\]](#)

## 2.2 Вибір місця для зберігання коду

Репозиторій - це місце, де зберігаються та відстежуються зміни файлів та каталогів. `expand_more` Він використовується для різних цілей, включаючи:

Зберігання коду: програмісти використовують репозиторії для зберігання коду своїх проектів. `expand_more` Це дозволяє їм відстежувати зміни, зроблені в коді, співпрацювати з іншими програмістами та повертатися до попередніх версій коду, якщо це необхідно

Найпоширенішою системою управління версіями (VCS) для зберігання коду є Git.expand\_more Git використовується для таких популярних платформ, як GitHub та GitLab.

Зберігання даних: репозиторії також можна використовувати для зберігання інших типів даних, таких як документи, зображення та відео. Це може бути корисно для співпраці над проектами або для створення резервних копій важливих файлів.

Контроль версій: репозиторії дозволяють відстежувати зміни, внесені до файлів та каталогів, з часом. Це може бути корисно для того, щоб побачити, хто зробив зміни, коли вони були зроблені та які зміни були внесені.expand\_more. Історія версій також може бути корисною для повернення до попередніх версій файлів, якщо це необхідно.

Співпраця: репозиторії дозволяють декільком людям працювати над одним проектом одночасно.expand\_more Це може бути корисно для прискорення розробки та покращення якості проекту. Різні системи VCS, такі як Git, пропонують різні функції для полегшення співпраці, такі як розгалуження та об'єднання.

Контроль доступу: репозиторії дозволяють контролювати, хто може отримувати доступ до файлів та каталогів, а також які дії вони можуть виконувати. Це може бути корисно для захисту конфіденційних даних або для запобігання ненавмисному внесенню змін. Більшість систем VCS пропонують різні рівні дозволів, які можна використовувати для надання користувачам різних рівнів доступу.

Публікація проектів: деякі репозиторії, такі як GitHub, дозволяють публікувати проекти, щоб інші могли їх бачити та використовувати. Це може бути корисно для отримання відгуків про проект або для співпраці з іншими людьми над проектом.

Створення резервних копій: репозиторії можна використовувати для створення резервних копій важливих файлів та каталогів. Це може бути корисно для захисту даних від втрати у разі пошкодження комп'ютера або іншої несприятливої події.

Розгортання проектів: Деякі системи VCS, такі як Git, пропонують функції для розгортання проектів на сервери. Це може бути корисно для публікації веб-сайтів або програмного забезпечення.

Загалом, репозиторії - це цінні інструменти, які можна використовувати для різних цілей. `expand_more` Вони можуть допомогти вам організувати свої файли, відстежувати зміни, співпрацювати з іншими та захищати свої дані.

Обираючи репозиторій, багато розробників віддають перевагу GitHub через його популярність та легкість використання. GitHub є одним з найпопулярніших сервісів для збереження коду, має велику спільноту розробників та активну підтримку. Його інтерфейс є дружнім до користувача, а функціональність дозволяє ефективно керувати репозиторіями, відстежувати зміни, створювати гілки та злиття, а також проводити перегляди коду та робити коментарі до нього.

### **Ось деякі з ключових особливостей GitHub:**

- **Безкоштовні та платні плани:** GitHub пропонує як безкоштовні, так і платні плани з різними функціями.
- **Публічні та приватні репозиторії:** Ви можете створювати як публічні, так і приватні репозиторії. Публічні репозиторії доступні для всіх, а приватні репозиторії доступні лише тим, хто має до них доступ.
- **Контроль версій Git:** GitHub використовує Git для контролю версій, що дає вам можливість відстежувати історію змін у коді.
- **Співпраця:** GitHub полегшує співпрацю над проектами з іншими розробниками.

- **Відстеження проблем:** Ви можете використовувати GitHub для відстеження проблем, пов'язаних з вашим проектом. Це дає вам можливість описувати проблеми, призначати їх розробникам та відстежувати їх статус..
- **Перегляди коду:** GitHub дозволяє вам проводити перегляди коду, де інші розробники можуть переглядати ваш код та надавати відгуки.
- **Візуалізація даних:** GitHub пропонує різні інструменти для візуалізації даних, які допомагають вам краще зрозуміти ваш проект, наприклад, графіки активності, діаграми змін та карти залежностей.
- **Інтеграція з іншими інструментами:** GitHub інтегрується з багатьма іншими інструментами, такими як IDE, системи безперервної інтеграції та системи безперервної доставки. [\[7\]](#)

Використання репозиторію, такого як GitHub, може значно покращити процес розробки програмного забезпечення. Це робить код більш безпечним, доступним, керованим та прозорим, а також полегшує співпрацю з іншими розробниками.

Якщо ви ще не використовуєте репозиторій для своїх проектів, я рекомендую вам спробувати GitHub. Це безкоштовно, просто у використанні та може значно покращити вашу продуктивність.

### 3 ПРОГРАМНА РЕАЛІЗАЦІЯ

#### 3.1 Розробка прототипу сторінок веб-сайту

Прототип веб-сайту – це попередня модель веб-сайту, яка використовується для тестування різних аспектів веб-сайту перед його створенням. Створення прототипів дозволяє дизайнерам експериментувати з різними макетами та елементами дизайну, а також тестувати взаємодію з користувачем. Перед створенням прототипу потрібно чітко визначити який контент буде представлено на сторінках веб-сайту. На рисунку 3.1 зображено навігаційну схему сайту, яка буде являтися мапою майбутнього веб-сайту. Схема дозволить раціонально розподілити функціональне та інформативне навантаження сайту.

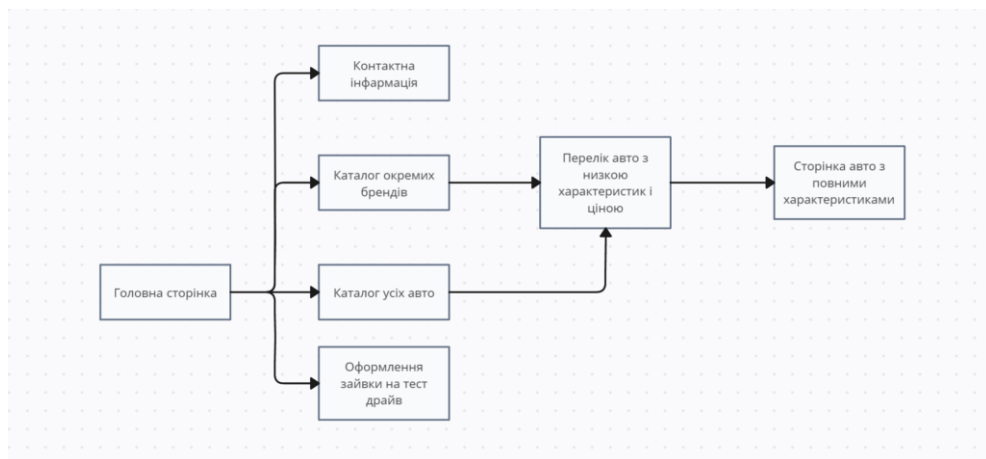


Рисунок 3 – Навігаційна схема веб-сайту

Майбутній веб-сайт матиме ієрархію сторінок. Дана схема являється оптимальною для структурованого наповнення інформаційної системи сайту. У користувача не повинно виникати складнощів при використанні веб-застосунку. Для повного представлення сайту який розробляється, було створено прототип головної сторінки який зображено на рисунку 3.1 блок-схем;

Головна сторінка – це домашня сторінка веб-сайту, яку відвідувачі бачать, коли переходять на сайт. Вона служить початковою точкою входу і часто відіграє вирішальну роль у формуванні першого враження користувача про веб-сайт і його бренд. Добре розроблена домашня сторінка є важливою з кількох причин:

- домашня сторінка – це перше, що бачать відвідувачі, тому дуже важливо справити позитивне враження. Візуально приваблива та добре організована домашня сторінка може привернути увагу відвідувачів, кожен користувач є потенційним клієнтом послуг які надає веб-сайт;
- домашня сторінка зазвичай містить навігаційне меню у якому є посилання, які спрямовують користувачів до різних розділів веб-сайту. Зручна структура навігації може допомогти відвідувачам швидко та легко знайти необхідне;
- головна сторінка має відображати ідентичність бренду включаючи логотип, колірну схему та загальний візуальний стиль. Це допоможе встановити впізнаваність бренду.

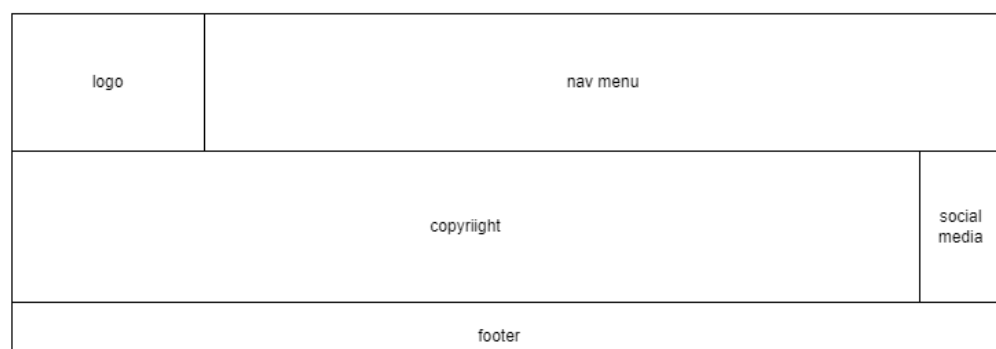
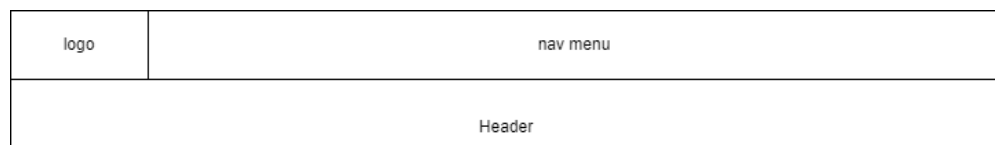


Рисунок 3.2 – Прототип головної сторінки веб-сайту

На рисунку 3.2 зображено прототип головної сторінки. В header розташовано логотип – найважливіший елемент іміджу компанії, навігаційне меню, яке дає можливість користувачу перейти на будь-які з доступних сторінок. У body розташовано блок з каталогом брендів авто. У footer дублюється навігаційне меню, логотип та розташовано блок з соціальними мережами та контактами.

Логотип, який зображений на рисунку 3.3, було створено у Illustrator.



Рисунок 3.3 – Логотип веб-сайту

За аналогічним алгоритмом роботи, розроблено прототипи інших сторінок сайту які зображені на рисунках 3.4, 3.5, 3.6:

- сторінка з каталогом;
- сторінка з відповідями на популярні питання;
- сторінка з контактами.

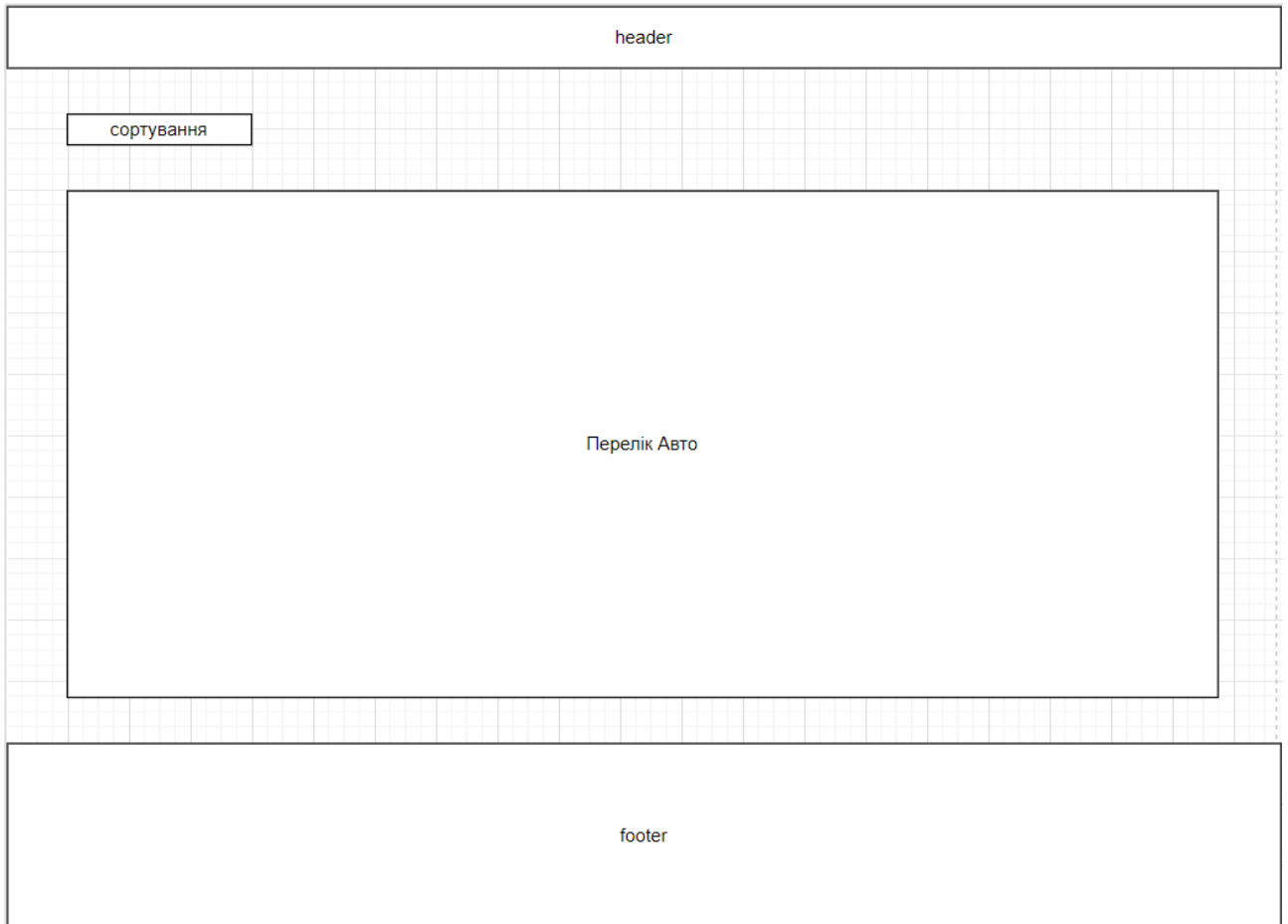


Рисунок 3.4 - сторінка з каталогом

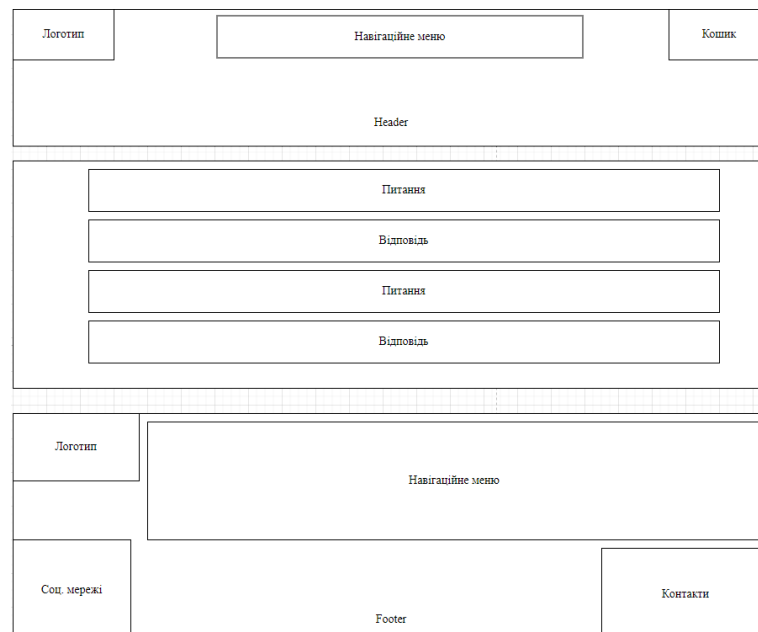


Рисунок 3.5 – Прототип сторінки з відповідями на популярні питання



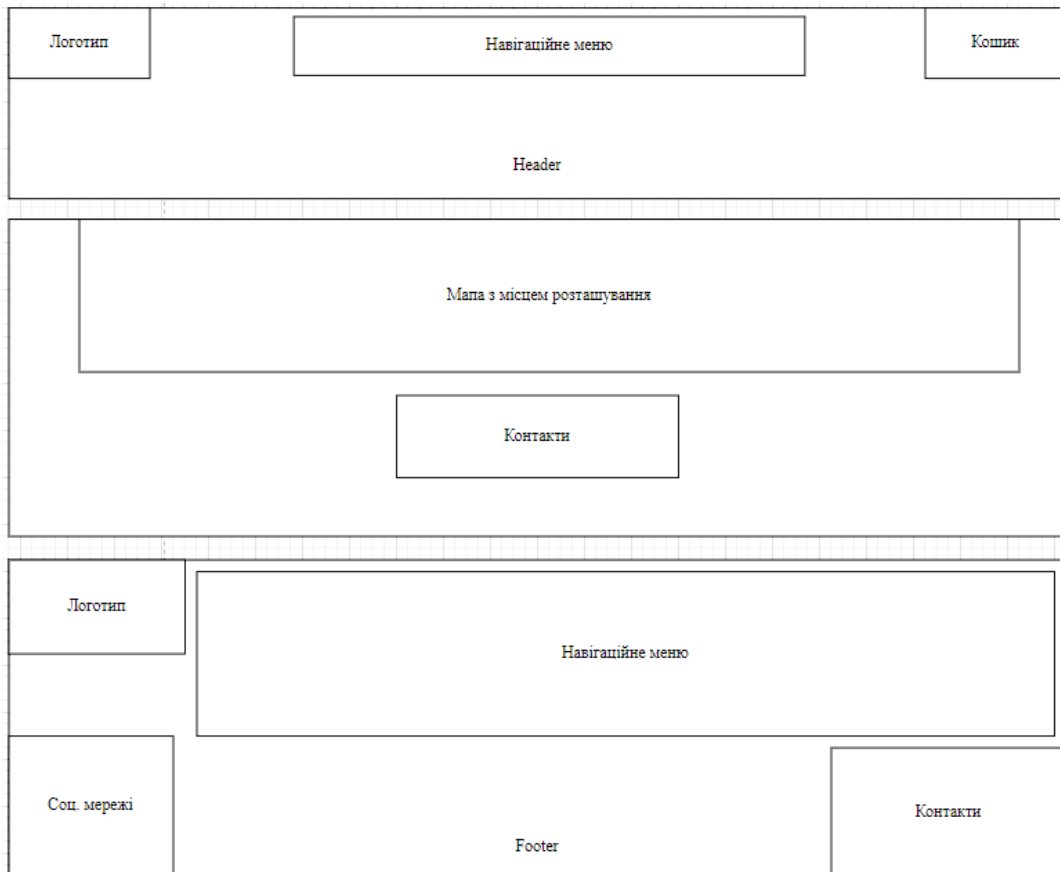


Рисунок 3.6 – Прототип сторінки з контактами

### 3.2 Створення візуальної частини сайту

Тепер, коли в нас є готові прототипи сторінок можна переходити до їх програмної реалізації.

Першою була реалізована головна сторінка. Як було сказано вище – головне щоб сторінка була інформативна але при цьому проста, щоб користувач не витрачав багато часу на пошук інформації що його цікавить

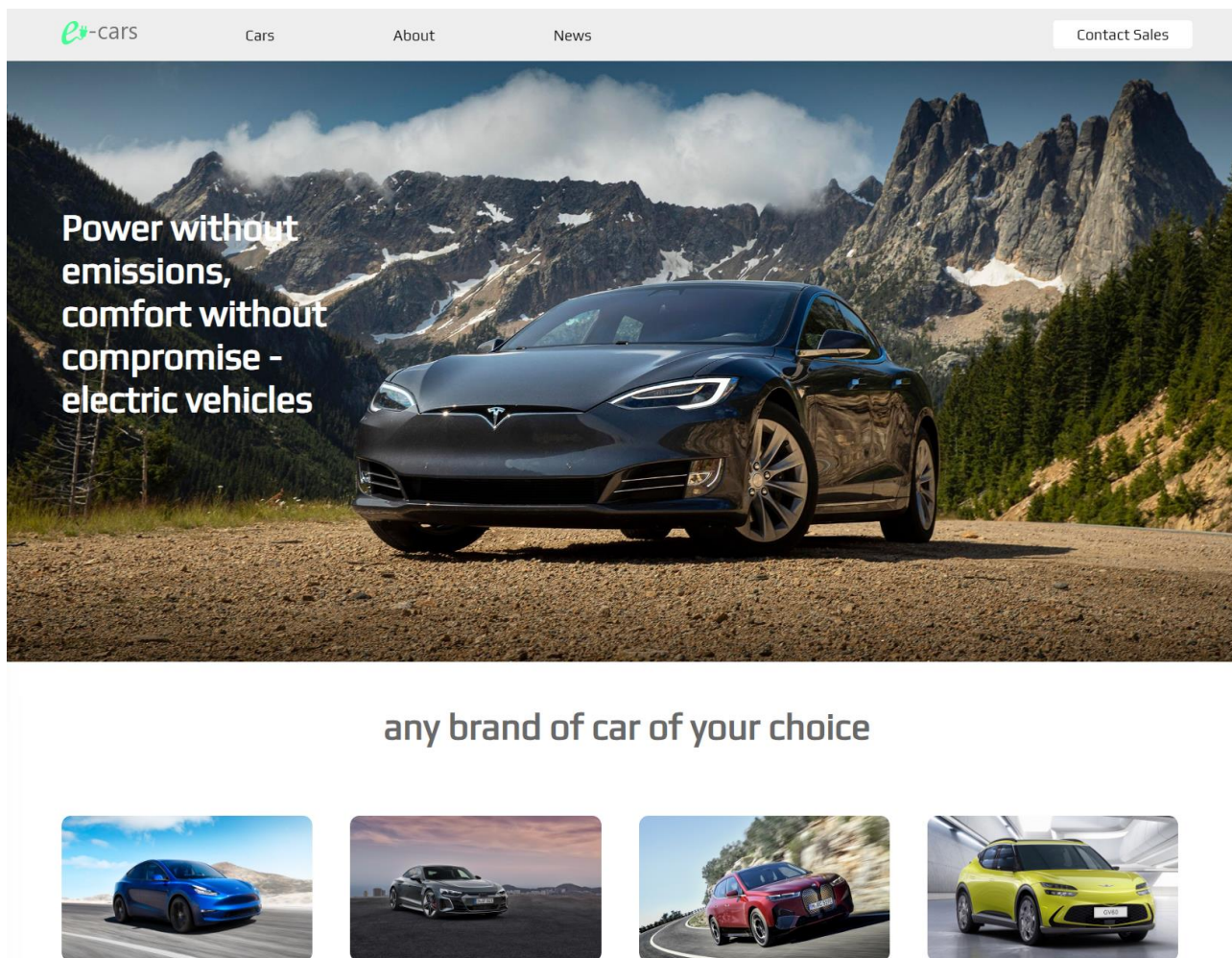


Рисунок 3.7 – Головна сторінка

Там де список усіх авто присутня проста анімація яка спрацьовує при прокручуванні сторінки. Це додає відчуття інтерактивності, що є дуже важливим моментом для утримання клієнта.

```

document.addEventListener('DOMContentLoaded', function() {
  var items = document.querySelectorAll('.cars_list');

  function checkVisibility() {
    for (var i = 0; i < items.length; i++) {
      var item = items[i];
      var itemTop = item.getBoundingClientRect().top;
      var windowHeight = window.innerHeight;

      if (itemTop < windowHeight) {
        item.classList.add('animate');
      }
    }
  }

  // Проверить видимость элементов при загрузке страницы
  checkVisibility();

  // Проверить видимость элементов при прокрутке страницы
  window.addEventListener('scroll', checkVisibility);
});

```

Рисунок 3.8 – Код анімацій

На сторінці з списком авто було вирішено додати деякі характеристики авто які цікавлять клієнтів частіше всього, і вже на їх основі створити сортування товарів див. рис. 3.9-3.10

The screenshot shows the Tesla website interface. At the top, there is a navigation bar with 'e-cars' logo, 'Cars', 'About', 'News', and 'Contact Sales' buttons. Below the navigation, the 'Tesla' brand name is displayed. A dropdown menu is set to 'In ascending order'. The main content area features a grid of eight Tesla car models, each with a photo, name, price, range, and connector type.

Model	Price (€)	Range (km)	Connector type
Tesla Model S Dual Motor	105490€	575 km	Type 2
Tesla Model S Plaid	130490€	560 km	Type 2
Tesla model Y	47567€	350 km	Type 2
Tesla Model Y Long Range Dual Motor	59017€	435 km	Type 2
Tesla Model Y Performance	63667€	415 km	Type 2
Tesla Model 3	44668€	405 km	Type 2
Tesla Model 3 Long Range Dual Motor	53668€	485 km	Type 2
Tesla Model 3 Performance	57668€	460 km	Type 2

Рисунок 3.9 – Каталог авто

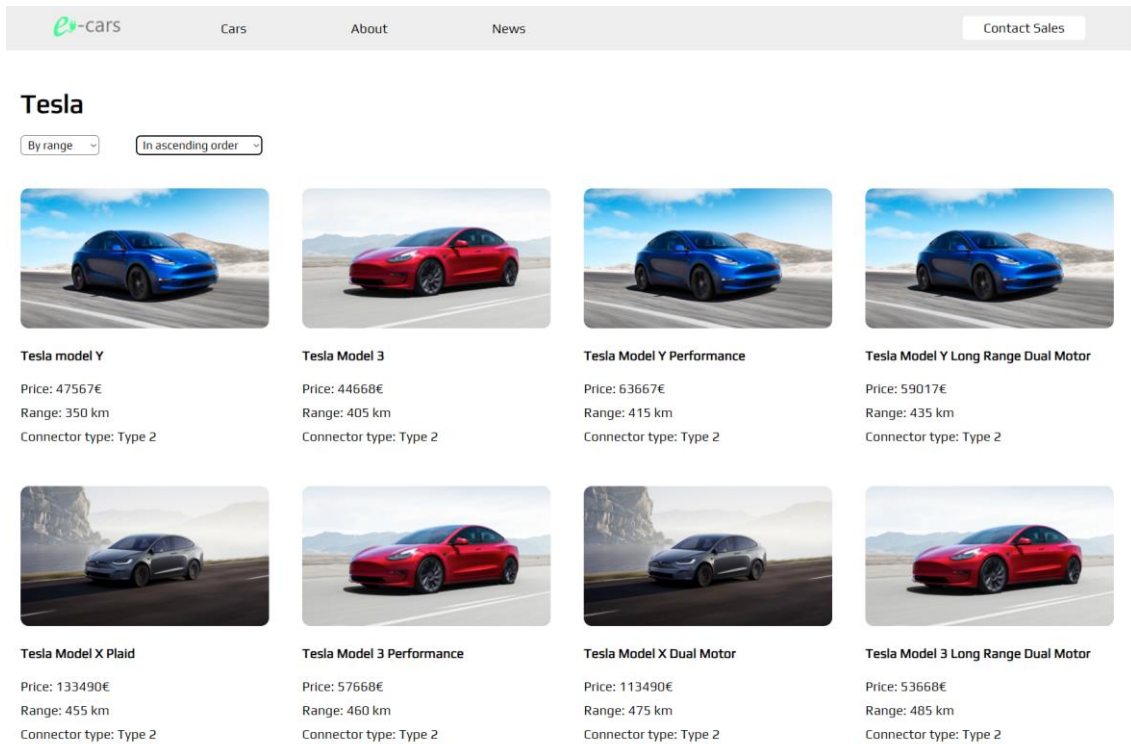


Рисунок 3.10 – Каталог авто

Самі товари та інформація беруться з js файлу.

```

{
  name: "Tesla Model X Dual Motor",
  price: 113490,
  range: "475 km",
  socketType: "Type 2",
  image: "../../../img/tesla/tesla model X/Tesla_Model_X_3.jpg",
},
{
  name: "Tesla Model X Plaid",
  price: 133490,
  range: "455 km",
  socketType: "Type 2",
  image: "../../../img/tesla/tesla model X/Tesla_Model_X_3.jpg",
},
];

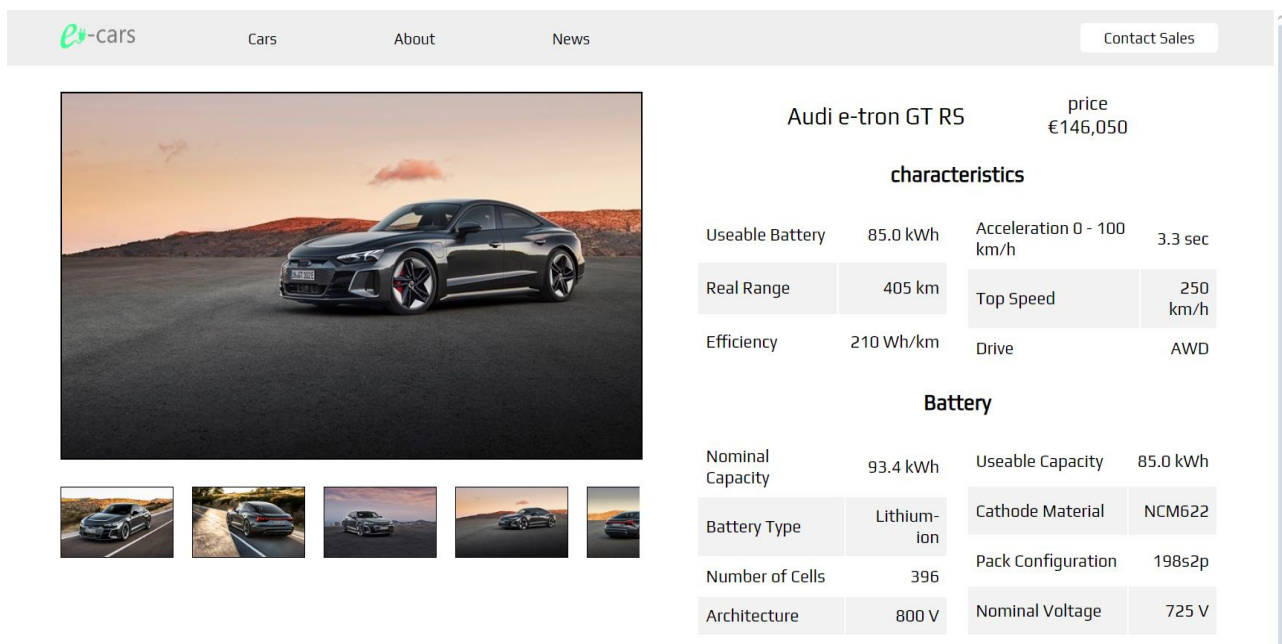
// Функция для отображения товаров
function displayProducts() {
  var productList = document.getElementById("productList");
  productList.innerHTML = "";

  products.forEach(function (product) {
    var productElement = document.createElement("div");
    productElement.className = "product";
    productElement.innerHTML = `
      <a href="${product.name}.html"></a>
      <h3>${product.name}</h3>
      <p>Price: ${product.price}€</p>
      <p>Range: ${product.range}</p>
      <p>Connector type: ${product.socketType}</p>
    `;
    productList.appendChild(productElement);
  });
}

```

Рисунок 3.11– Код файлу з товарами

Було вирішено що сторінка з повними характеристиками та фотографіями авто має бути максимально простою та не перенасиченою інформацією. Тому на ній присутні лише написаний слайдер та таблиця з головними характеристиками див. рис. 3.12-3.13



**e-cars**   Cars   About   News   [Contact Sales](#)

**Audi e-tron GT RS**   price €146,050

**characteristics**

Useable Battery	85.0 kWh	Acceleration 0 - 100 km/h	3.3 sec
Real Range	405 km	Top Speed	250 km/h
Efficiency	210 Wh/km	Drive	AWD

**Battery**

Nominal Capacity	93.4 kWh	Useable Capacity	85.0 kWh
Battery Type	Lithium-ion	Cathode Material	NCM622
Number of Cells	396	Pack Configuration	198s2p
Architecture	800 V	Nominal Voltage	725 V

Рисунок 3.12 – Сторінка товару

```
const imageColumn = document.querySelector('.image-column');
let isDown = false;
let startX;
let scrollLeft;

imageColumn.addEventListener('mousedown', (e) => {
  isDown = true;
  startX = e.pageX - imageColumn.offsetLeft;
  scrollLeft = imageColumn.scrollLeft;
});

imageColumn.addEventListener('mouseleave', () => {
  isDown = false;
});

imageColumn.addEventListener('mouseup', () => {
  isDown = false;
});

imageColumn.addEventListener('mousemove', (e) => {
  if(!isDown) return;
  e.preventDefault();
  const x = e.pageX - imageColumn.offsetLeft;
  const walk = (x - startX) * 1.5; //scroll-fast
  imageColumn.scrollLeft = scrollLeft - walk;
});
```

Рисунок 3.13 – Код слайдеру

### 3.3 Проектування бази даних

Перед тим, як розпочинати проектування бази даних для сайту автосалону, необхідно чітко визначити його ключові вимоги. Ці вимоги можна розділити на дві категорії: функціональні та нефункціональні. Функціональні вимоги до сайту автосалону включають:

Функціональні вимоги описують те, що сайт повинен робити та як він має це робити. До ключових функціональних вимог сайту автосалону належать:

- Реєстрація та авторизація: Користувачі повинні мати можливість створювати облікові записи та входити в систему.
- Перегляд салонів: Користувачі повинні мати можливість переглядати список доступних автосалонів.
- Відгуки: Користувачі повинні мати можливість залишати відгуки про автомобілі та автосалони.
- Заовлення: Користувачі повинні мати можливість створювати заовлення на купівлю автомобілів.
- Налаштування: Користувачі повинні мати можливість налаштовувати власний профіль, включаючи вибір мови та теми інтерфейсу.

Нефункціональні вимоги описують якості сайту, такі як його безпека, надійність та продуктивність. До ключових нефункціональних вимог сайту автосалону належать:

- Безпека: Сайт має гарантувати безпеку даних користувачів, захищаючи їх від несанкціонованого доступу, модифікації чи видалення.
- Надійність: Сайт має бути надійним та стійким до збоїв, щоб мінімізувати час простою та забезпечити безперебійну роботу.
- Продуктивність: Сайт має швидко завантажуватися та реагувати на дії користувачів, навіть при значному навантаженні.

- Масштабованість: Сайт має бути масштабованим, щоб його можна було легко розширювати та адаптувати до зростаючого трафіку та кількості користувачів.
- Користувальницький інтерфейс: Сайт має мати зручний та інтуїтивно зрозумілий інтерфейс, який буде простим у навігації та користуванні.
- Доступність: Сайт має бути доступним для людей з обмеженими можливостями.

Чітке визначення та розуміння вимог до сайту автосалону є критично важливим для успішного проектування та розробки бази даних. Ці вимоги слугують основою для створення сайту, який буде функціональним, надійним, зручним у користуванні та відповідатиме потребам як власників автосалонів, так і їхніх клієнтів.

### 3.4 Розробка бази даних

Почнемо зі створення бази даних. Щоб створити MySQL базу даних на локальному пристрої, вам зазвичай знадобиться встановлене програмне забезпечення для управління базами даних наприклад, MySQL Server або XAMPP, який містить MySQL сервер.

Ми обрали MySQL Server, ви можете завантажити та встановити його з офіційного веб-сайту MySQL див. рис. 3.14

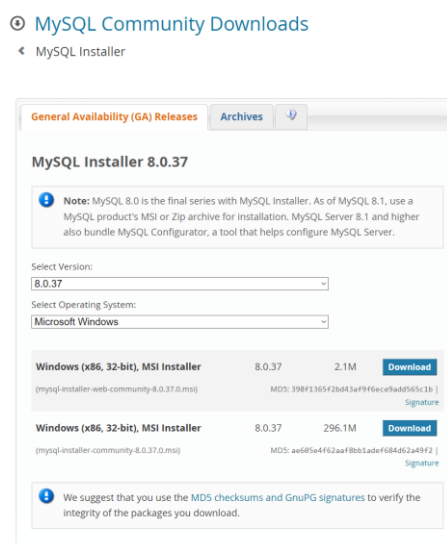


Рисунок 3.14 – Офіційний сайт MySQL для скачування ПО

Далі обираємо потрібну нам версію, скачуємо і встановлюємо її.

Також треба буде встановити phpMyAdmin щоб не робити усе через консоль. Його також можна встановити з офіційного сайту див. ри. 3.15

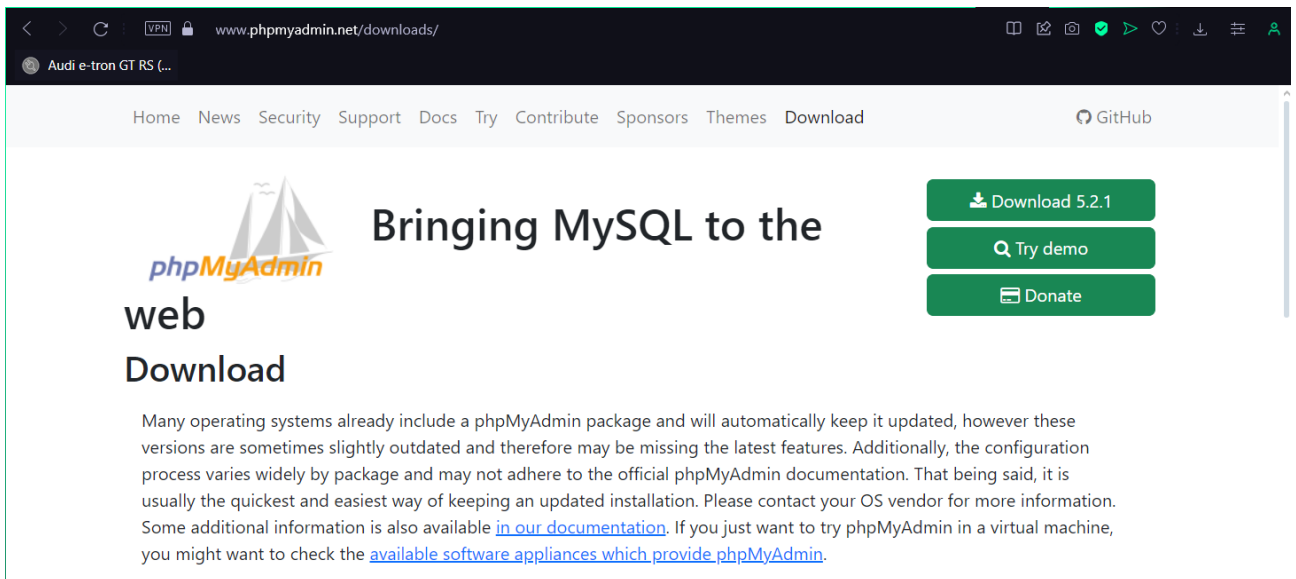


Рисунок 3.15 – Офіційний сайт phpMyAdmin

Після встановлення всього необхідного треба створити РНР файл для налаштування підключення до нашої бази даних див. рис. 3.16

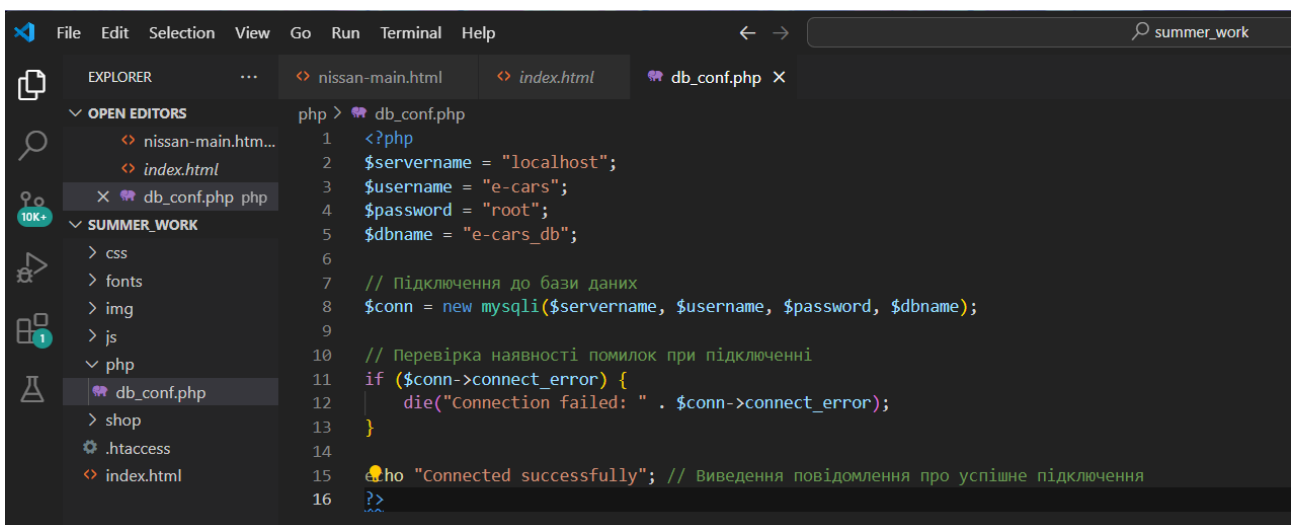


Рисунок 3.16 – Файл конфігурації



Для створення усіх потрібних таблиць нам доведеться зайти до phpMyAdmin, ввести наші данні від БД та написати код для їх створення див. рис. 3.17

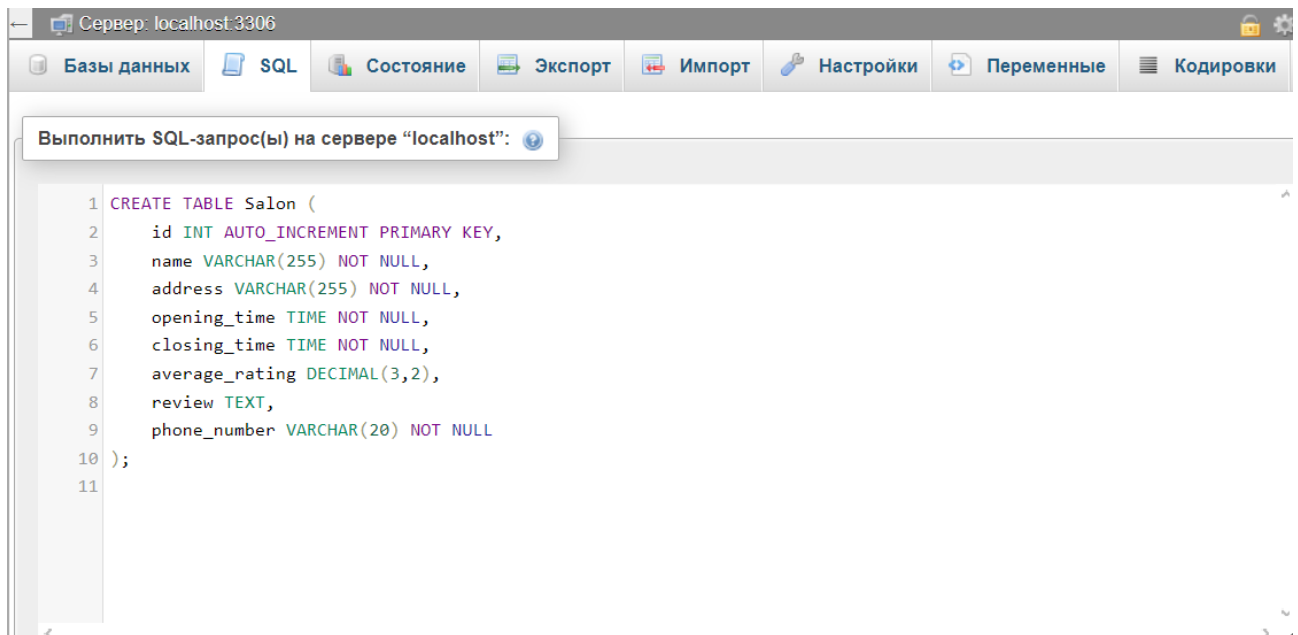


Рисунок 3.17 – Код створення таблиці

Теж саме повторюємо для усіх потрібних нам таблиць. Також нам треба написати код PHP код який заповнить даними ці таблиці див. рис. 3.18

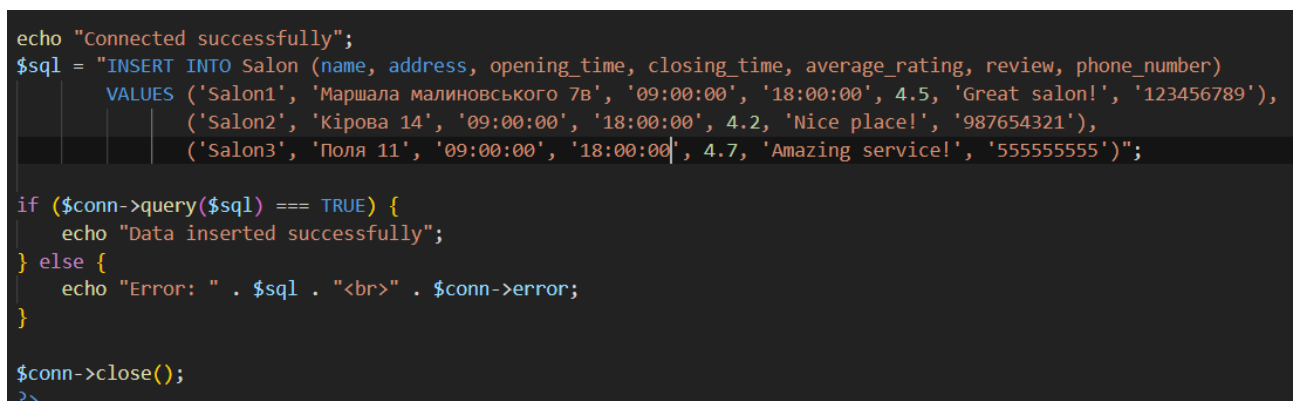


Рисунок 3.18 – Заповнення таблиці

Далі нам треба підготувати місце для виведення цих даних на нашу сторінку див. рис. 3.19

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Salon Data</title>
</head>
<body>

<?php
require_once "php/db_conf.php";

$sql = "SELECT * FROM Salon";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
  while($row = $result->fetch_assoc()) {
    echo "<a href='#'>";
    echo "Salon Name: " . $row["name"] . "<br>";
    echo "Address: " . $row["address"] . "<br>";
    echo "Opening Time: " . $row["opening_time"] . "<br>";
    echo "Closing Time: " . $row["closing_time"] . "<br>";
    echo "Average Rating: " . $row["average_rating"] . "<br>";
    echo "Review: " . $row["review"] . "<br>";
    echo "Phone Number: " . $row["phone_number"] . "<br>";
    echo "</a><br>";
  }
} else {
  echo "0 results";
}

$conn->close();
?>

</body>
</html>

```

Рисунок 3.19 –Код для відправлення листа

На цьому базові підготовки взаємодії з БД завершені.

Наступнім кроком буде створення сторінки для оформлення заявки для на тест драйв авто див.рис.3.20 – 3.21

```

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $to = "tgcharging@gmail.com"; // Замените на ваш адрес электронной почты
    $subject = "Новый заказ"; // Тема письма

    $name = $_POST["name"];
    $lastname = $_POST["second-name"];
    $phone = $_POST["phone "];
    $birthdate = $_POST["birthdate "];
    $preferred_time = $_POST["preferred_time"];

    $message .= "ім'я: $name\n\n";
    $message .= "Прізвище: $lastname \n\n";
    $message .= "Адреса електронної пошти: $email\n\n";
    $message .= "Номер телефону: $phone \n\n";
    $message .= "Номер телефону: $preferred_time \n\n";

    $headers = "MIME-Version: 1.0" . "\r\n";
    $headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";
    $headers .= "From: $email" . "\r\n";

    if (mail($to, $subject, $message, $headers)) {
        header("Location: ../success.html");
        exit();
    } else {
        $errorMessage = "Произошла ошибка при отправке заказа.";
    }
}
?>

<?php
if (isset($errorMessage)) {
    echo $errorMessage;
}
?>

```

Рисунок 3.20 – Код для відправлення повідомлення

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Test Drive Request Form</title>
</head>
<body>

<h2>Test Drive Request Form</h2>
<form action="process_form.php" method="post">
  <label for="name">First Name:</label><br>
  <input type="text" id="name" name="name" required><br>

  <label for="lastname">Last Name:</label><br>
  <input type="text" id="lastname" name="lastname" required><br>

  <label for="phone">Phone Number:</label><br>
  <input type="tel" id="phone" name="phone" required><br>

  <label for="birthdate">Date of Birth:</label><br>
  <input type="date" id="birthdate" name="birthdate" required><br>

  <label for="preferred_time">Preferred Time for Test Drive:</label><br>
  <input type="datetime-local" id="preferred_time" name="preferred_time" required><br><br>

  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Рисунок 3.21 – Форма для вводу даних користувача

## ВИСНОВКИ

Під час кваліфікаційної роботи було створено інтернет-сайт автосалону, що відповідає вимогам сучасного ринку та забезпечує зручне користування для клієнтів. Мета полягала в тому, щоб розробити інтуїтивно зрозумілий та функціональний інтерфейс, який би значно полегшував процес пошуку та вибору автомобілів для користувачів.

Обрані інструменти розробки, такі як HTML, CSS, JavaScript та PHP, дозволили оптимізувати процес роботи та зосередитися на важливих аспектах проекту. Зокрема, використання цих технологій забезпечило швидку завантажувальність сторінок, високу продуктивність та адаптивність дизайну для різних пристроїв.

Під час розробки було враховано всі ключові функції, необхідні для зручної навігації по сайту, зручного пошуку автомобілів та отримання додаткової інформації про них. Наприклад, було реалізовано функції фільтрації та сортування автомобілів за різними критеріями, таких як марка, модель, рік випуску та ціна. Також було додано можливість перегляду детальних описів та фото автомобілів, що значно підвищує інформативність сайту для потенційних покупців.

Результатом стала готова платформа, яка стала основою для подальшого розширення та вдосконалення. Сайт не лише відповідає поточним потребам автосалону, але й має потенціал для інтеграції нових технологій та функціоналу в майбутньому.

Серед можливих покращень – впровадження системи онлайн-бронювання автомобілів, інтеграція з соціальними мережами для зручного обміну інформацією та відгуками, а також розробка мобільного додатку для ще більшого спрощення доступу до сервісів автосалону.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Коды помилок HTTP та інші відповіді сервера : HOSTiQ Wiki. *HOSTiQ Wiki*. URL: <https://hostiq.ua/wiki/ukr/http-status-codes/>
2. Desktop vs Mobile vs Tablet Market Share Worldwide | Statcounter Global Stats. *StatCounter Global Stats*. URL: <https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet/worldwide>
3. Що таке .htaccess URL: <https://hostiq.ua/wiki/htaccess/#meaning>
4. Чому саме PHP URL: [Чому PHP: переваги PHP перед іншими мовами - блог ІТ-школи Hillel \(ithillel.ua\)](#)
5. Реляційні та нереляційні бази даних URL: <https://alternativescience.net/programming/242-sql-chy-nosql-os-v-chomu-pytannya/>
6. Чому був обраний MySQL URL: <https://dan-it.com.ua/uk/blog/vse-shho-potribno-znati-pro-bazi-danih-dlja-pochatkivciv-mysql-postgresql-mongodb/>
7. Чому варто обрати GitHub URL: <https://dou.ua/lenta/articles/github-vs-gitlab/>
8. Desktop vs Mobile vs Tablet Market Share Worldwide | Statcounter Global Stats. *StatCounter Global Stats*. URL: <https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet/worldwide>
9. Responsive Design Vs Adaptive Design (Pros and Cons). *CareerFoundry*. URL: <https://careerfoundry.com/en/blog/ui-design/responsive-vs-adaptive-design/>

## ДОДАТОК

## Animation.js

```

document.addEventListener('DOMContentLoaded', function() {
  var items = document.querySelectorAll('.cars_list');

  function checkVisibility() {
    for (var i = 0; i < items.length; i++) {
      var item = items[i];
      var itemTop = item.getBoundingClientRect().top;
      var windowHeight = window.innerHeight;

      if (itemTop < windowHeight) {
        item.classList.add('animate');
      }
    }
  }

  // Проверить видимость элементов при загрузке страницы
  checkVisibility();

  // Проверить видимость элементов при прокрутке страницы
  window.addEventListener('scroll', checkVisibility);
});

```

## Footer.js

```

window.onload = function() {
  let productList = document.querySelector('#productList');
  let footer = document.querySelector('#footer');

  function adjustFooterPosition() {
    let productListHeight = window.getComputedStyle(productList).getPropertyValue('height');

    if (parseInt(productListHeight) > 360) {
      footer.style.position = 'static';
    } else {
      footer.style.position = 'fixed';
    }
  }

  adjustFooterPosition(); // Вызываем функцию один раз при загрузке страницы

  window.addEventListener('resize', adjustFooterPosition); // Вызываем функцию при изменении размера окна
};

function toggleMenu(event) {
  event.preventDefault();
  var submenu = document.getElementById("submenu");
  var arrow = document.getElementById("arrow");
  if (submenu.classList.contains("open")) {
    submenu.classList.remove("open");
    arrow.classList.remove("open");
  } else {
    submenu.classList.add("open");
    arrow.classList.add("open");
  }
}

```

## Header.js

```
const body = document.body;
const header = document.querySelector("header");
const scrollUp = "scroll-up";
const scrollDown = "scroll-down";
let lastScroll = 0;

window.addEventListener("scroll", () => {
  const currentScroll = window.pageYOffset;
  if (currentScroll == 0) {
    header.classList.remove(scrollUp);
    return;
  }

  if (currentScroll > lastScroll && !body.classList.contains(scrollDown)) {
    // down
    let elem = document.getElementById("menu");
    elem.classList.remove("active");
    header.classList.remove(scrollUp);
    header.classList.add(scrollDown);
  } else if (
    currentScroll < lastScroll &&
    header.classList.contains(scrollDown)
  ) {
    // up
    header.classList.remove(scrollDown);
    header.classList.add(scrollUp);
  }
  lastScroll = currentScroll;
});

function menu() {
  let elem = document.getElementById("menu");
  elem.classList.add("active");
}

function menuClose() {
  let elem = document.getElementById("menu");
  elem.classList.remove("active");
}
```



## Sortin.js

```
// Функция для сортировки товаров
function sortProducts() {
  var selectElement = document.getElementById('sortSelect');
  var selectedValue = selectElement.value;

  if (selectedValue === 'price') {
    products.sort(function(a, b) {
      return a.price - b.price;
    });
  } else if (selectedValue === 'range') {
    products.sort(function(a, b) {
      return a.range.localeCompare(b.range);
    });
  } else if (selectedValue === 'socketType') {
    products.sort(function(a, b) {
      return a.socketType.localeCompare(b.socketType);
    });
  }

  productList.appendChild(productElement);
}
```

```
// Функция для сортировки товаров
function sortProducts() {
  var sortFieldSelect = document.getElementById('sortFieldSelect');
  var sortDirectionSelect = document.getElementById('sortDirectionSelect');
  var sortField = sortFieldSelect.value;
  var sortDirection = sortDirectionSelect.value;

  if (sortField === 'price') {
    products.sort(function(a, b) {
      if (sortDirection === 'asc') {
        return a.price - b.price;
      } else {
        return b.price - a.price;
      }
    });
  } else if (sortField === 'range') {
    products.sort(function(a, b) {
      if (sortDirection === 'asc') {
        return a.range.localeCompare(b.range);
      } else {
        return b.range.localeCompare(a.range);
      }
    });
  } else if (sortField === 'socketType') {
    products.sort(function(a, b) {
      if (sortDirection === 'asc') {
        return a.socketType.localeCompare(b.socketType);
      } else {
        return b.socketType.localeCompare(a.socketType);
      }
    });
  }
}
```

```

        displayProducts();
    }

    // Функция вызывается при изменении способа сортировки
    function onSortChange() {
        sortProducts();
    }

```

## Products.js

```

// Массив с товарами
var products = [
    {
        name: "Audi e-tron GT RS",
        price: 146050,
        range: "405 km",
        socketType: "Type 2",
        image: "../../img/audi/Audi_e-tron_GT_RS/Audi_e-tron_GT_RS-01.jpg",
    },
    {
        name: "Audi e-tron GT quattro",
        price: 106050,
        range: "420 km",
        socketType: "Type 2",
        image: "../../img/audi/Audi_e-tron_GT_quattro/Audi_e-tron_GT_quattro-01.jpg",
    },
    {
        name: "Audi Q4 e-tron 40",
        price: 51900,
        range: "405 km",
        socketType: "Type 2",
        image: "../../img/audi/Audi_Q4_e-tron_40/Audi_Q4_e-tron_2021-01.jpg",
    },
    {
        name: "Audi Q4 e-tron 50 quattro",
        price: 57900,
        range: "385 km",
        socketType: "Type 2",
        image: "../../img/audi/Audi_Q4_e-tron_40/Audi_Q4_e-tron_2021-01.jpg",
    },
    {
        name: "Audi Q8 e-tron 55 quattro",
        price: 85300,
        range: "495 km",
        socketType: "Type 2",
        image: "../../img/audi/Audi_Q8_e-tron_55_quattro/Audi_Q8-etron-2023-01.jpg",
    },
    {
        name: "Audi Q8 e-tron Sportback 55 quattro",
        price: 87550,
        range: "525 km",
        socketType: "Type 2",
        image: "../../img/audi/Audi_Q8_e-tron_Sportback_55_quattro/Audi_Q8-etron-sportback-2023-01.jpg",
    },

```

```
{
  name: "Audi e-tron GT RS",
  price: 146050,
  range: "405 km",
  socketType: "Type 2",
  image: "../../img/audi/Audi_e-tron_GT_RS/Audi_e-tron_GT_RS-01.jpg",
},
{
  name: "Audi e-tron GT quattro",
  price: 106050,
  range: "420 km",
  socketType: "Type 2",
  image: "../../img/audi/Audi_e-tron_GT_quattro/Audi_e-tron_GT_quattro-01.jpg",
},

{
  name: "Audi Q4 e-tron 40",
  price: 51900,
  range: "405 km",
  socketType: "Type 2",
  image: "../../img/audi/Audi_Q4_e-tron_40/Audi_Q4_e-tron_2021-01.jpg",
},
{
  name: "Audi Q4 e-tron 50 quattro",
  price: 57900,
  range: "385 km",
  socketType: "Type 2",
  image: "../../img/audi/Audi_Q4_e-tron_40/Audi_Q4_e-tron_2021-01.jpg",
},

{
  name: "Audi Q8 e-tron 55 quattro",
  price: 85300,
  range: "495 km",
  socketType: "Type 2",
  image: "../../img/audi/Audi_Q8_e-tron_55_quattro/Audi_Q8-etron-2023-01.jpg",
},

{
  name: "Audi Q8 e-tron Sportback 55 quattro",
  price: 87550,
  range: "525 km",
  socketType: "Type 2",
  image: "../../img/audi/Audi_Q8_e-tron_Sportback_55_quattro/Audi_Q8-etron-sportback-2023-01.jpg",
},
},
```

```
{
  name: "BMW iX xDrive40",
  price: 77300,
  range: "360 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW iX xDrive40/BMW_iX_2022-01.jpg",
},
{
  name: "BMW iX1 xDrive30",
  price: 55000,
  range: "370 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW iX1 xDrive30/BMW_iX1_2022-01.jpg",
},
{
  name: "BMW i4 M50",
  price: 70800,
  range: "450 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW i4 M50/BMW_i4_M50-01.jpg",
},
{
  name: "BMW i4 eDrive40",
  price: 59200,
  range: "515 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW i4 eDrive40/BMW_i4_eDrive40-01.jpg",
},
{
  name: "BMW i5 M60 xDrive Sedan",
  price: 99500,
  range: "425 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW i5 M60 xDrive Sedan/BMW_i5_M60_xDrive_Sedan-01.jpg",
},
{
  name: "BMW i7 eDrive50",
  price: 115700,
  range: "520 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW i7 eDrive50/BMW_i7_2022-01.jpg",
},
{
  name: "BMW i7 M70 xDrive",
  price: 181800,
  range: "490 km",
  socketType: "Type 2",
  image: "../../img/bmw/BMW i7 M70 xDrive/BMW_i7_M70_2023-01.jpg",
},
}
```

```

{
  name: "Genesis GV60 Sport",
  price: 63200,
  range: "380 km",
  socketType: "Type 2",
  image: "../../img/genesis/Genesis GV60 Sport/Genesis_GV60-01.jpg",
},
{
  name: "Genesis GV70 Electrified Sport",
  price: 67300,
  range: "350 km",
  socketType: "Type 2",
  image: "../../img/genesis/Genesis GV70 Electrified Sport/Genesis_GV70-01.jpg",
},
{
  name: "Genesis G80 Electrified Luxury",
  price: 69200,
  range: "440 km",
  socketType: "Type 2",
  image: "../../img/genesis/Genesis G80 Electrified Luxury/Genesis_G80-01.jpg",
},

```

```

{
  name: "Hyundai Kona Electric 64 kWh",
  price: 42900,
  range: "395 km",
  socketType: "Type 2",
  image: "../../img/hyundai/Hyundai Kona Electric 64 kWh/Hyundai_Kona_Electric_2021-01.jpg",
},
{
  name: "Hyundai Kona Electric 65 kWh",
  price: 48000,
  range: "390 km",
  socketType: "Type 2",
  image: "../../img/hyundai/Hyundai Kona Electric 65 kWh/Hyundai_Kona_Electric_2024-01.jpg",
},
{
  name: "Hyundai IONIQ 5 Long Range AWD",
  price: 59200,
  range: "385 km",
  socketType: "Type 2",
  image: "../../img/hyundai/Hyundai IONIQ 5 Long Range AWD/Hyundai_IONIQ5_2021-01.jpg",
},
{
  name: "Hyundai IONIQ 6 Long Range AWD",
  price: 61100,
  range: "440 km",
  socketType: "Type 2",
  image: "../../img/hyundai/Hyundai IONIQ 6 Long Range AWD/Hyundai_IONIQ6_2022-01.jpg",
},

```