

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

До захисту допускається

Завідувач кафедри

_____ Скарга-Бандурова І.С.

«___» _____ 20__ р.

ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА

ПОЯСНЮВАЛЬНА ЗАПИСКА

НА ТЕМУ:

Інформаційно-комп'ютерна система керування інтернет-магазином з продажу
автомобілів

Освітньо-кваліфікаційний рівень “бакалавр”

Спеціальність 122 “комп'ютерні науки”

Керівник проекту:

_____ (підпис)

_____ (ініціали, прізвище)

Консультант з охорони праці:

_____ (підпис)

_____ (ініціали, прізвище)

Студент:

_____ (підпис)

_____ (ініціали, прізвище)

Група: _____

Севєродонецьк 2017

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки

Кафедра Комп'ютерної інженерії

Освітньо-кваліфікаційний рівень бакалавр

Напрямок підготовки

(шифр і назва)

Спеціальність

(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав.кафедрою КІ

О.І. Рязанцев

«___» _____ 20__ р.

ЗАВДАННЯ

НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТА

Стиров Сергій Едуардович

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи)

Інформаційно-комп'ютерна система керування інтернет-магазином з продажу автомобілів
затверджена наказом по інституту від « 22» вересня 2017г. № _____ 01-08/160с

2. Термін подання студентом закінченого проекту (роботи) 16.01.2017р.

3. Початкові дані до проекту (роботи)

4. Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробці)

1) Аналіз задачі і розробка технічного завдання

2) Вибір і обґрунтування програмних засобів для розробки системи

4) Розробка системи

5) Охорона праці

6) Висновки

5. Перелік графічного матеріалу (з точною вказівкою обов'язкових креслень)

6. Консультанти проекту (роботи) з визначенням розділів проекту, які відносяться до них

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

Керівник проекту _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Пошук літературних джерел	28.04.2017 – 02.05.2017	
2	Вивчення літературних джерел	03.05.2017 – 06.05.2017	
3	Розробка технічного завдання	07.05.2017 – 08.05.2017	
4	Проектування дизайну	09.05.2017 – 10.05.2017	
5	Написання системи	11.05.2017 - 28.06.2017	
6	Розробка заходів по безпечному веденню робіт	29.06.2017 - 02.06.2017	
7	Оформлення пояснювальної записки	03.06.2017 - 10.06.2017	

Студент-дипломник _____
(підпис)

Керівник проекту _____
(підпис)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра: 60с., 29рис., 3 табл., 27 бібліографічних джерел посилань , 2 додатки.

Об'єкт розробки: інформаційно-комп'ютерна система керування Інтернет-магазином з продажу автомобілів.

Мета роботи: проектування інформаційно-комп'ютерної системи керування Інтернет-магазином.

В проекті виконано:

1. Спроектований унікальний дизайн клієнтської частини системи з використанням технологій HTML, CSS, JavaScript.
2. Спроектовано серверну частину системи за допомогою технологій PHP, AJAX та моделі MVC.
3. Розроблено систему продажу автомобілів з подальшим розвитком.

Практичне значення, галузь застосування роботи: в комерційних цілях, для продажу автомобілів за допомогою мережі Інтернет.

ІНТЕРНЕТ, СИСТЕМА, СЕРВІС, ВЕБ-СТОРІНКА, РОЗМІТКА, ІНТЕРФЕЙС, БАЗА ДАНИХ, HTML, CSS, JS, PHP, MVC.

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр. Центральний 59-А,. м. Сєвєродонецьк, 93400.

Зміст

ВСТУП.....	7
1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.2 Система керування Інтернет-магазином продажу автомобілів	8
1.2 Веб-сайт.....	9
1.2.1 Різновиди сайтів.....	10
1.3 Передача даних в мережі Інтернет.....	11
1.4 Сервер.....	12
1.5 Search Engine Optimization (SEO)	13
1.4 Висновки і постановка завдання	13
2 ВИБІР ТЕХНОЛОГІЙ РОЗРОБКИ.....	15
2.1 Мова гіпертекстової розмітка HTML.....	15
2.2 Каскадні таблиці стилів CSS.....	16
2.3 Мова програмування JavaScript.....	18
2.4 Бібліотека JQuery	21
2.5 Мова програмування PHP	22
2.6 MVC - модель реалізації додатків	23
2.7 Веб-технологія AJAX	24
2.8 Бази даних.....	25
2.8.1 Порівняння MySQL та PostgreSQL	26
3 РОЗРОБКА СИСТЕМИ	28
3.1 Розробка структури системи.....	28
3.2 Система реєстрації користувачів.....	29
3.3 Панель адміністрування	33
3.4 Сторінка з автомобілями	36

3.5 Сторінка з запчастинами та корзина	39
4 ОХОРОНА ПРАЦІ	44
4.1 Загальні питання з охорони праці	44
4.1.1 Правові та організаційні основи охорони праці	47
4.1.2 Організаційно-технічні заходи з безпеки праці	48
4.2 Аналіз стану умов праці	50
4.2.1 Вимоги до приміщень.....	50
4.2.2 Вимоги до організації місця праці	51
4.2.3 Навантаження та напруженість процесу праці.....	52
4.3 Виробнича санітарія.....	54
4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу.....	54
4.3.2 Пожежна безпека	56
4.3.3 Електробезпека.....	57
4.4 Гігієнічні вимоги до параметрів виробничого середовища.....	58
4.4.1 Освітлення	58
4.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій	59
ВИСНОВОК.....	64
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	65
ДОДАТОК А.....	67
ДОДАТОК Б	87

ВСТУП

На сьогоднішній день в мережі Інтернет налічується більше 1 мільярда сайтів [27]. Це дуже велике досягнення, так як всього десять років назад цей показник становив близько ста мільйонів сайтів. 67% населення землі користується глобальною павутиною. Мережа Інтернет складається з мільйонів сайтів багатьох різновидів: блогів, новинних порталів, соціальних мереж, Інтернет-магазинів й ще багатьох менш розповсюджених типів сайтів.

В наш час у кожної поважаючої себе компанії повинен бути Веб-сайт. Це вже не відмінна частина нашого часу, частина іміджу компанії. Один сайт вирішує більшу частину проблем з маркетингом компанії. Більше не потрібно шукати покупців, щоб розширитися на всю країну. Досить зробити сайт з описом компанії та доброзичливим інтерфейсом, який буде підлаштовуватися під різні розміри екранів, а потім зробити невелику рекламу в мережі, щоб люди дізналися про компанію й почали користуватися її послугами.

Отримати свій веб-сайт - це спосіб розширення ринку на велику аудиторію з акцентом не тільки на свою країну, а й на інші країни. Завжди можливо домовитися з компанією іншої країни на взаємну рекламу і таким чином розширити свою клієнтську аудиторію.

В цьому дипломному проекті проектується інформаційно-комп'ютерна система управління Інтернет-магазином «AutoShop», який призначений для продажу автомобілів.

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.2 Система керування Інтернет-магазином продажу автомобілів

Однією із задач цього дипломного проекту є проектування системи керування Інтернет-магазином та її взаємодії з користувачем.

До цього часу існуюча система керування магазином полягала в наступному. Для того щоб продавати якийсь товар, потрібно було знайти будівлю під склад та офіс. При цьому продаж міг йти тільки у місці знаходження офісу, потрібно було заказувати якусь рекламу, трати кошти на візитки і т. д.

Щоб купити собі автомобіль або продати його, людині потрібно їхати в автосалон чи на авторинок, причому потрібних послуг може не бути на місці. Це дуже не зручно споживачеві, особливо якщо місто невелике і цих послуг немає.

Так само і з придбанням запчастин. Потрібних комплектуючих може не буде в місті і потрібно буде кудись їхати, а якщо запчастина коштовна, буде потрібно брати велику кількість грошей. При цьому товар може бути важким і споживачеві потрібно ще шукати способи, щоб доставити товар до дому.

З наведених даних стає зрозуміло, що потрібно спроектувати веб-систему покупки і продажу автомобілів та комплектуючих до них з такими можливостями:

- вітрина товарів;
- доставка товарів по всій країні;
- оплата товарів;
- повна інформація про товари і ресурси;
- швидкі відповіді, які надходять від користувачів.

Переваги такої системи в Інтернеті над реальними аналогами в житті:

- можливість почати бізнес без капіталу;
- доступність кожному без обмежень у віці;
- мінімум вкладень;
- велика кількість користувачів в мережі Інтернет;
- можливість розвивати бізнес в будь-якому куточку земної кулі;
- швидке поширення інформації;

- доступ до системи 24 години на добу;
- можливість аналізувати потреби користувачів.

1.2 Веб-сайт

Веб-сайт - це набір текстових сторінок на мові HTML (HyperText Markup Language), які пов'язані з іншими сторінками сайту і містяться на сервері зі своїм унікальним адресом і отримуються користувачем по глобальній мережі Інтернет за допомогою браузера. Сторінки також мають свою адресу, що складається з адреси сайту і імені файлу. Всі Інтернет-сторінки в сукупності утворюють велику Інтернет - “павутину”.

Виготовлення Інтернет-ресурсу, як великої працюючої Інтернет-системи - це складений процес, який включає в себе багато технологій і який можна розділити на такі частини:

- розробка макету сайту;
- верстання шаблону з отриманого макету;
- програмування логіки сайту з нуля або з використанням готових CMS або фреймворків;
- тестування системи;
- розміщення ресурсу на веб-сервері.

Можливості застосування веб-технологій дуже великі, але якщо говорити конкретно про сайт Інтернет-магазину, то є багато переваг над звичайним магазином:

1. Взаємодія з потенціальним споживачем. Сайт - це можливість для спілкування з клієнтом та подання товару, інформації, щоб споживач міг ознайомитися й замовити потрібний товар, не виходячи з дому. І якщо появиться якесь питання – то зможе задати його відразу на самому сайті і відразу отримати відповідь від адміністратора.
2. Організація товарів. Сайт буде як великий каталог з товаром, який можливо завжди доповнити, оновити.

3. Мінімальні затрати. Не потрібно мати великі будівлі з офісами. Потрібен тільки склад та невеликий штат співробітників.
4. Розширення ринку на велика аудиторія з акцентом на захід. А це вже новий рівень торгівлі і приріст прибутку.

1.2.1 Різновиди сайтів

В мережі Інтернет велика різноманітність веб-ресурсів. Як такої, класифікації сайтів немає, але є характеристики, за якими відрізняються системи:

1. Комерційний веб-сайт – це обов'язкова частина будь-якої компанії. Допомагає формувати імідж компанії, знаходження й поповнення клієнтської бази. Зберігання важливої інформації компанії, якою може скористуватися клієнт. Це найдешевший різновид реклами для вашої організації.
2. Візитка – набір інформації який буде допомагати клієнтам швидко знаходити інформацію про компанію. Не дорогий різновид реклами, яку можливо в любий момент змінити.
3. Інтернет-магазин - найкращий спосіб для розширення ринку збуту, отримання великої аудиторії. Можливість клієнтів покупати товар, не виходячи з дому, отримання повної інформації про товар. Можливість розміщення необмеженого асортименту товарів, завжди вчасно реагувати на потреби покупців.
4. Особисті сайти – це сайти де керування виконує одна людина. Спектр цих сайтів дуже великий, від маленьких сторінок з інформацією про користувача, до великих популярних ресурсів.
5. Інформаційний портал – надає відвідувачам різноманітну інформацію.

Також сайти поділяються за своєю складністю на статичні та динамічні:

1. Статичні сайти – це набір HTML-сторінок, розміщених на веб-сервері, зміни на цих сторінках можуть робитися тільки власником сайту. Такі сторінки швидко загрузаються, мінімально навантажують сервер, легко

переносяться на другий сервер, можливо подивиться код сторінки у браузері. Але у статичних сторінок є великий мінус - неможливо динамічно генерувати вміст й неможливо змінювати зміст сторінки з самого сайту.

2. Динамічні сайти – це сайт який складається з шаблону, скриптов, контенту в окремих файлах. З цих файлів на стороні сервера складається динамічна сторінка. З одного шаблону може буде сформовано дуже багато однотипних сторінок.

1.3 Передача даних в мережі Інтернет

Передача інформації в мережі Інтернет здійснюється засобами мережевих протоколів передачі даних. Сам протокол - це набір правил і дій за допомогою яких здійснюється передача даних.

Передача даних з одного комп'ютера на другий проводиться в декілька етапів:

1. Одержування даних від користувача.
2. Стиснення отриманих даних.
3. Шифрування даних.
4. Формування пакетів, на яке ділиться повідомлення, установлення процедури відношення між комп'ютером який передає данні й приймає їх.
5. Вибір результативного маршруту для передачі.
6. Формування вихідного документу з пакетів.

На кожному з цих етапів використовується різні протоколи передачі даних, які формують набір протоколів Інтернету TCP/IP.

TCP – це протокол, який формує зв'язок між двома комп'ютерами в Інтернеті.

IP – це протокол, який відповідає за те, щоб пакет був доставлено, куди потрібно.

Протоколи, які найчастіше використовують в Інтернеті:

1. HTTP – протокол який використовується для пересилання веб-сторінок.
2. FTP – використовується для обміну двійковими та текстовими файлами.
3. POP – використовується для опрацювання запитів на отримання пошти від поштових програм.
4. SMTP – протокол який потрібен для передавання пошти.

Для того щоб передавати дані в Інтернеті, для кожної машини надається унікальна IP-адреса. IP-адреса – це унікальна адреса, що ідентифікує комп'ютер в мережі Інтернет.

Кожний сайт в мережі Інтернет, розміщується на сервері і йому надається унікальна IP-адреса. Але щоб знайти цей сайт в Інтернеті, не потрібно вводити його IP, для цього є DNS. DNS (Domen Name System) – це система імен, яку записують як послідовність розділених крапкою слів.

1.4 Сервер

Для того щоб користувачі бачили систему й могли користуватися нею в мережі Інтернет, потрібно покласти веб-сайт на сервер.

Сервер - це той же самий комп'ютер, тільки в більшості випадків могутніше. Сервер, як і комп'ютер, складається з: процесора, материнської плати, оперативної пам'яті і жорсткого диска. Найчастіше для великих серверів використовують спеціально призначені для цього комплектуючі.

Раніше сервера були в звичайному системному блоці, але це незручно, так як в більшості організацій є необхідність в декількох серверах, і такі «ящики» були дуже непрактичними. З новими технологіями, починаючи з кінця 2000-х років, сервера стали мати дещо іншу форму. Тепер вони перебувають у спеціальному відсіку, який монтується в стійку.

Управління великою кількістю серверів, які знаходяться в стійках, проводиться віддалено. Тобто інженер, перебуваючи як в сусідньому кабінеті, так

і в іншому місті - може на відстані керувати серверами. Установка ж операційних систем на сервер виробляється на місці.

1.5 Search Engine Optimization (SEO)

SEO – це ряд заходів які допомагають підняти сайт в системі пошуку по ключовим запитам. Головною метою цієї процедури є збільшення кількості відвідувачів веб-ресурсу. Пошукова оптимізація являє собою цілу науку, метою якої являється зробити сайт популярним й попудити користувачів купити товар.

Процедура просування Інтернет ресурсу ,це досить складний і трудомісткий процес, потребує знань й вмінь. Тільки фахівець який має великий досвід в цій області, може вивести сайт на перші рядки і зробити його популярним в Інтернеті. Для кожного сайту SEO має своє унікальне, індивідуальне просування в мережі Інтернет. Однієї великої схеми яка підійшла би під усі види сайтів не має. Існують тільки інструменти, застосування яких допоможе вивести сайт на першу позицію, але спосіб реалізації цих інструментів для усіх систем різний. Якісно зроблене SEO здатне вивести бізнес в Інтернеті на дуже великий рівень.

1.4 Висновки і постановка завдання

На даний час Інтернет-технології досягли великого рівня можливостей, які надають нові можливості для розвитку системи продажу автомобілів й комплектуючих.

1. Проектована система повинна бути орієнтована на збільшення якості та кількості продажу автомобілів в Україні;
2. Користувач отримає доступ до пропонованих послуг, не виходячи з дому:
 - великий список авто;
 - отримання інформації по любому автомобілю;
 - каталог запчастин на автомобіль;
 - інформація про кожну запчастину;

- доступ до розділу з новинами;
- отримання інформації про компанію;
- зворотний зв'язок з користувачем.

3. Адміністратор отримає доступ до таких послуг:

- додавання нових автомобілів, новин, товарів до каталогів;
- редагування існуючих товарів, новин, автомобілів;
- розділ з прийманням заказів від користувачів та їх обробка.

Таким чином, система зможе забезпечувати необхідні потреби користувачів і виконуватиме це зручно, не займаючи багато часу.

2 ВИБІР ТЕХНОЛОГІЙ РОЗРОБКИ

2.1 Мова гіпертекстової розмітка HTML

HTML - це мова розмітки документів у мережі Інтернет. Вона складається з тегів, які формують зміст сторінки. На даний момент версія мови HTML вже досягла п'ятої, і це не кінець, мова продовжує доповнюватися новими можливостями.

Тег - це мітка, яка використовується для вказівки браузеру, що робити з тією інформацією, що знаходиться в контейнері тега. Теги бувають одинарні (наприклад, `
`) й парні теги (наприклад, `<div>...</div>`). Також у тега може бути атрибут, який буде вказувати, які параметри будуть у тега (наприклад, `<p text-align="center" class="text"></p>`). В мові HTML є таке поняття як семантика тексту, яка означає, що на сторінці повинен бути тільки HTML, а стилі - в окремому файлі стилів CSS, скрипти також повинні знаходитися в окремому файлі JS.

Основною особливістю HTML є принцип, за яким не тільки допускається вкладення одних елементів в інші, але і декларується необхідність такого вкладення. Кожен елемент HTML допускає безпосереднє вкладення тільки ряду елементів, які в свою чергу, допускають вкладення інших, дозволених для них, і т.д. Таким чином формується не тільки загальна структура гіпертексту, але і створюються різноманітні візуальні ефекти.

В HTML багато тегів, але для форматування сторінки достатньо кількох.

Наприклад:

`<div>...</div>` - блочний тег, потрібний для стилів;

`<p>...</p>` - тег нового абзацу;

`<h1>...</h1>` - тег заголовка, яких буває п'яти рівнів важливості;

`...` - тег посилання на другу сторінку, а `href` - це атрибут який містить адресу другої сторінки;

`` - тег розміщення зображення.

Незважаючи на те, що специфікація HTML є стандартом, цю мову доповнюють новими елементами. Тому деякі сторінки зручніше переглядати за допомогою певних браузерів. Розширення створюються тільки відомими фірмами, які розробляють програмне забезпечення, а рядові користувачі можуть удосконалювати свої WEB-сторінки за допомогою програмування.

2.2 Каскадні таблиці стилів CSS

Одного HTML для побудови красивого й адаптивного сайту занадто мало. Для стилізації потрібна технологія каскадних таблиць CSS.

CSS – це мова опису зовнішнього виду текстових документів написаних на мові розмітки HTML. Основним правилом CSS є розділення мови розмітки від опису зовнішнього виду. Це дозволяє очистити документ HTML від зайвого коду й додати читабельності. Також CSS допомагає зменшити повторення коду на сторінці.

Правильно писати всі правила CSS в окремому документі не засмічуючи HTML й слідуючи правилам семантики.

CSS правило складається з двох частин: селектор й блок об'явлення. Де селектор це назва атрибуту к якому буде прийнято стилі ,або назва класу, ид. А об'явлення складається з одного чи кількох правил.

Також в CSS є пріоритет стилів. Прикладом цьому є, що якщо писати однакові стилі в документі CSS и на сторінці HTML, то прийнятий буде стиль зі сторінки HTML (рис.2.1).

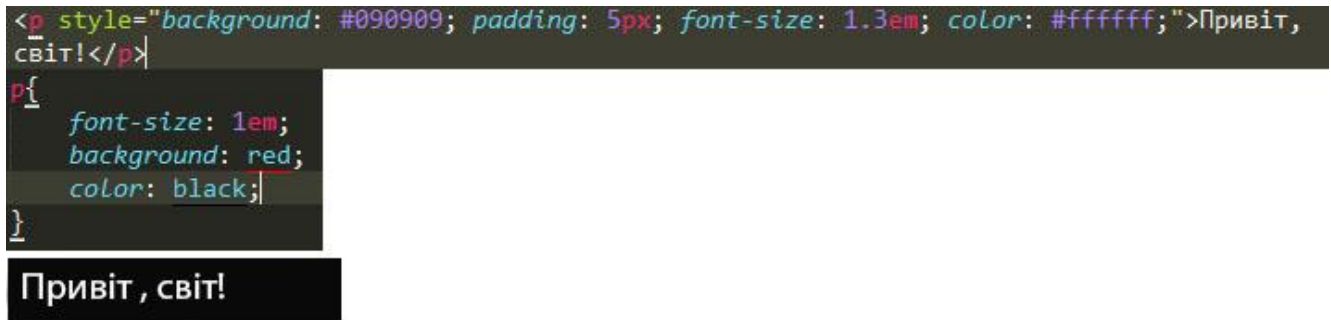


Рис. 2.1 - Пріоритет стилів.

Пріоритети стилів:

1. Самий низький пріоритет у стилів браузера;
2. Потім йдуть стилі, задані користувачем у настройках браузера;
3. Стилi, задані в зовнішніх таблицях (.css)
 - 3.1 Стилi задані через id;
 - 3.2 Стилi задані через class;
 - 3.3 Стилi задані тегам.
4. Стилi, задані в самому документі через <style></style>;
5. Стилi, задані в самому атрибуті;
6. Стилi, задані с доданням !important.

В 2017 році за статистикою 80% користувачів для перегляду глобальної мережі користується мобільними телефонами, з цього стає зрозуміло, що без адаптивності сайтів під розміри екрану вже не обійтись. Тому в версії CSS3 були введені медіа запити. Вони допомагають вводити нові стилі для елемента під певний розмір екрану.

Типи медіа запитів:

1. all - за замовчуванням, для усього;
2. handheld - для смартфонів й схожих апаратів;
3. print - для печаті;
4. projection - для проекторів;
5. screen - під екран монітора;
6. speech - для програм звукового відтворення тексту;
7. tty - пристрої з фіксованим розміром символів;

8. tv - під телевізори;

9. braille - пристрої для читання сліпим людям.

Мабуть одним з важливих нововведень в CSS3 є анімація transitions й анімації, які дозволяють виконувати анімацію на сторінці без використання JavaScript. Але для того, щоб досягти найкращого й плавного ефекту анімацій, все одно потрібно доповнювати каскадні таблиці за допомогою мови JS.

Колись технології CSS-анімацій зможуть відмовитися від JavaScript, але на даному етапі розвитку це неможливо.

2.3 Мова програмування JavaScript

Важливим засобом для розробки красивого, “живого” дизайну системи є мова програмування JavaScript (JS). Ця мова найчастіше використовується в парі з мовою гіпертекстової розмітки для забезпечення динамічності, придання інтерактивності Інтернет-сторінці. JavaScript використовує об’єктно-орієнтовану і імперативну парадигму програмування [5].

JavaScript - це мова програмування, програма на якій інтерпретується. Щоб запустити програму на виконання, не потрібно її компілювати, досить запустити її в браузері.

Основні можливості JS:

1. Включення в веб-сторінку нових тегів HTML, видалення вже включених, додавання стилів до елементів.
2. Реагування на дії користувача, відслідковування пересування курсору, оброблення кліків миші й ще багато іншого.
3. Посилання запитів на сервер й отримання відповідей без перезавантаження сторінки.

І це тільки основні можливості, мова JavaScript набагато ширше. Але й є деякі обмеження (рис. 2.2):

1. Головним обмеженням є те, що JS не може записувати щось на жорсткий диск, програма не має прямого доступу до операційної системи.

2. JavaScript, працюючий в одній вкладці, не може впливати на інші вкладки браузера.
3. Мова не може посилати запит на інші домени, так як є обмеження безпеки.

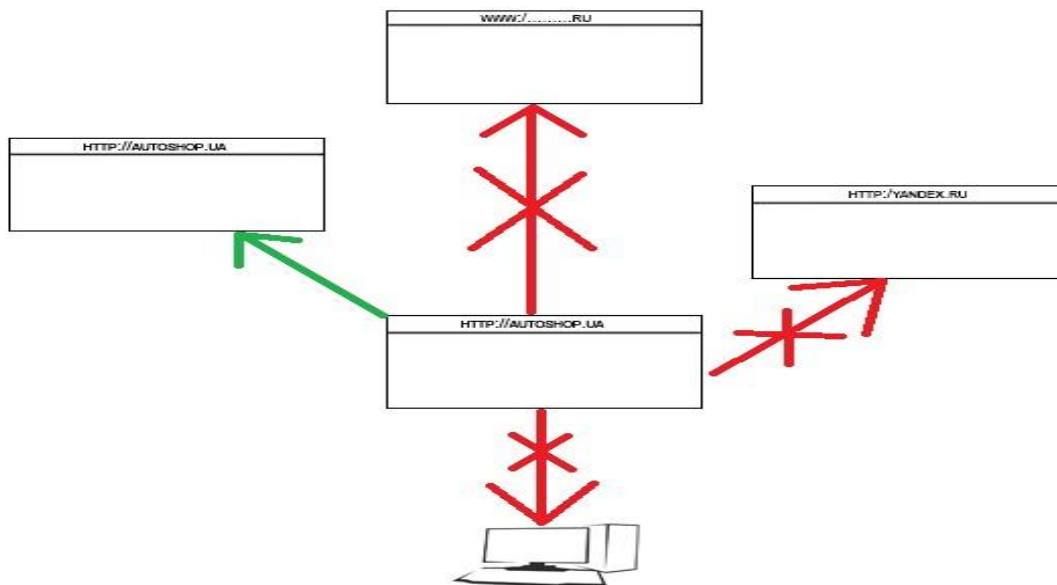


Рисунок 2.2 - Можливості JS обмежені однією сторінкою.

JavaScript працює з HTML за допомогою DOM-дерева, яке формує браузер при загрузці сторінки. DOM-дерево (рис. 2.3) – це інтерфейс, який дозволяє програмам JavaScript отримати доступ і взаємодіяти з елементами HTML.

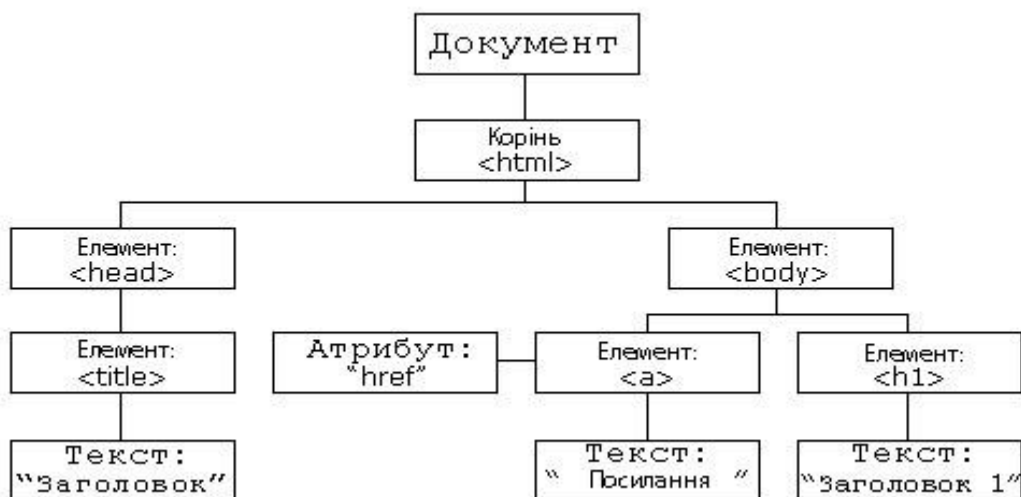


Рисунок 2.3 - приклад DOM дерева

Для того, щоб отримати якийсь об'єкт з DOM-дерева, потрібно звернутися до глобального об'єкту `document`. Найчастіше запити DOM-дереву реалізують через атрибути `id` або `class` елементів.

Об'єктно-орієнтований стиль написання програмного коду заснований на поданні програми в вигляді сукупності об'єктів, кожний з цих об'єктів є екземпляром класу, а класи можуть спадкуватись.

У об'єктно-орієнтованого програмування є три основні концепції:

Інкапсуляція - це коли програміст не дає другим об'єктам змінювати стан поточного об'єкта, тобто внутрішній стан об'єкта може змінюватися тільки самим об'єктом. Прикладом цього може бути модель автомобіля, в якій механік змінює якусь частину одного об'єкта, наприклад двигуна, інші об'єкти (кузов, салон) при цьому залишаються незмінними.

Поліморфізм - це спосіб звернення до об'єктів різних підкласів одного класу так, ніби вони усі одного класу. Тобто кілька різних класів можуть об'явити один и той же метод чи властивість.

Спадкування - особливість, що дозволяє зробити клас на базі вже існуючого класу, з частковим або ідентичним інтерфейсом. При цьому клас від якого успадковують має назву – батьківський, а клас спадкоємець – дочірній. Також клас-спадкоємець може мати свої методи в поряд з методами батьківського класу.

Сам клас - це опис властивостей і методів, які визначають характеристики об'єктів, які будуть створені на базі цього класу.

Архітектурні риси JavaScript наступні:

1. Динамічна типізація – змінна зв'язується зі своїм типом в момент присвоєння значення.
2. Слабка типізація – володіє слабкою системою типів.
3. Прибирання сміття - автоматичне звільнення пам'яті.
4. Прототипне програмування – розділ ООП, де відсутнє поняття класу, а спадкування відбувається клонуванням прототипу об'єкта.

Мова сама дуже компактна, але гнучка: розробники написали багато API-інтерфейсів для браузерів, які розкривають деякі здатності мови, такі як

динамічна будова сторінки HTML й додання для неї стилів CSS. Також є сторонні бібліотеки й CMS, які дозволяють прискорити розробку сайту. На даний час мова JS є самою перспективною мовою в сайто-будівництві, яка в сукупності з CSS і HTML переводить сайт на новий рівень креативності.

Усі скрипти, так само як й CSS-стили зазвичай містяться в різних файлах, щоб не забивати HTML-код зайвою інформацією.

Якщо це не так, то потрібно розміщати усі скрипти в кінці HTML сторінки, тоді не буде постійних зависань сторінки, але це не вирішує проблему повністю. При повільному Інтернеті дуже помітно, як браузер поступово завантажує сторінку, браузер не чекає поки завантажиться уся сторінка, а показує те, що вже встиг завантажити. Якщо на сторінці багато скриптів, це може дійсно згубно впливати на швидкість завантаження сторінки. Тому розробники ввели асинхронні запити за допомогою атрибутів `async` і `defer`; з цими атрибутами можна вказувати скрипти в шапці.

При виявленні атрибута `async`, браузер не закінчує роботу і не переходить в JavaScript, а продовжує роботу, а коли скрипт завантажиться повністю, браузер запустить його. Якщо таких скриптів з `async` буде кілька, то першим запуститься той, що завантажиться раніше.

`Defer` також виконується асинхронно, з різницею в тому, що кілька скриптів з `defer` будуть оброблятися по порядку, в якому вони записані в HTML; це зроблено для того, щоб скрипти запускалися послідовно.

2.4 Бібліотека JQuery

Якщо говорити про мову JavaScript, то не можна пропустити бібліотеку JQuery.

JQuery представляє із себе невелику, швидку й багатофункціональну бібліотеку, яка дозволяє писати набагато швидше код JS. JQuery допомагає реалізовувати анімацію, обробники подій, полегшує вибір елементів з DOM й створення Ajax-запитів.

Бібліотека функціонує з усіма браузерами, тому не потрібно переживати за кросбраузерність. Також для jQuery написано багато додатків, які в сукупності з JavaScript допомагають ще швидше розібратися з певною задачею при організації веб-сторінки.

Щоб приєднати бібліотеку до сторінки, потрібно загрузити з сайту бібліотеку, скинути її в папку з сайтом й потім підключити в кінці сторінки після усього HTML коду.

2.5 Мова програмування PHP

PHP – це мова з відкритим кодом, яка підтримує об'єктно-орієнтоване та процедурне програмування, використовується для написання серверних веб-додатків. Синтаксис мови є похідним від мов C, Java, Perl. Головним плюсом PHP є надання користувачу можливості швидкого створення генеруємих веб-сторінок [9].

Програма на мові PHP виконується на стороні сервера. Мова PHP дозволяє швидко робити якісні веб-додатки, які потім можна легко поліпшувати.

PHP має багато можливостей: оброблювати данні форми, породжувати динамічні сторінки, відправляти й отримувати cookies.

PHP може не тільки швидко генерувати сторінки, а й ще формувати зображення, файли PDF та Flash-відео; вона може автоматично генерувати ці файли й зберігати їх в файловій системі сервера.

Щоб переглядати PHP скрипти, потрібен веб-сервер або локальний сервер і браузер. Але є можливість для побудови скриптів, котрі будуть запускатися поза веб-сервером чи браузером; для цього потрібен парсер PHP. Також на мові можливо писати віконні додатки на стороні користувача.

PHP підтримує усі сучасні операційні системи Windows, Linux, RISC OS, Mac OS, також мова підтримує більшість сучасних веб-серверів таких як Microsoft IIS, Apache, Information Server, Personal Веб Server і ще багато інших. Також PHP підтримує багато баз даних: Adabas D, dbase, Ingres, Solid, MySQL.

Важливі характеристики мови:

1. Традиційність мови - багато взято з C, Perl тому вивчити її не так складно.
2. Простота мови - якщо код має простий синтаксис, то вона буде виконувати в точності те, що ви її доручили.
3. Ефективність - двигун мови дозволяє дуже ефективно оброблювати скрипти.
4. Безпека - в PHP реалізовані механізми безпеки, що знаходяться під контролем адміністратора, що дає велику свободу дій і максимальну безпеку.
5. Гнучкість - вона відрізняється винятковою пристосовністю до потреб розробника.

2.6 MVC - модель реалізації додатків

MVC (Model, View, Controller) – це схема поділення інтерфейсу й логіки додатку на окремі компоненти (рис.2.4):

Model – це компонент, який реагує на команди контролера й змінює свій стан. Надає данні та методи для роботи з ними. Не залежить від контролера, модель просто надає доступ до даних і управління ними.

View – відповідає за відображення даних моделі. Тобто отримує дані від моделі і відображає їх.

Controller – забезпечує користувача зв'язком з системою; контролює дані, які йдуть від користувача до системи й навпаки.

Модель MVC не має якоїсь усталеної реалізації. Деякі фреймворки дуже жорстко задають, де повинна знаходитися бізнес-логіка, а в MVC-моделі це не має значення. Вона може знаходитися як в контролері так і в моделі. Також не має жорстоких правил де повинна знаходитися перевірка введених даних користувачем, але найчастіше їх реалізують в контролері.

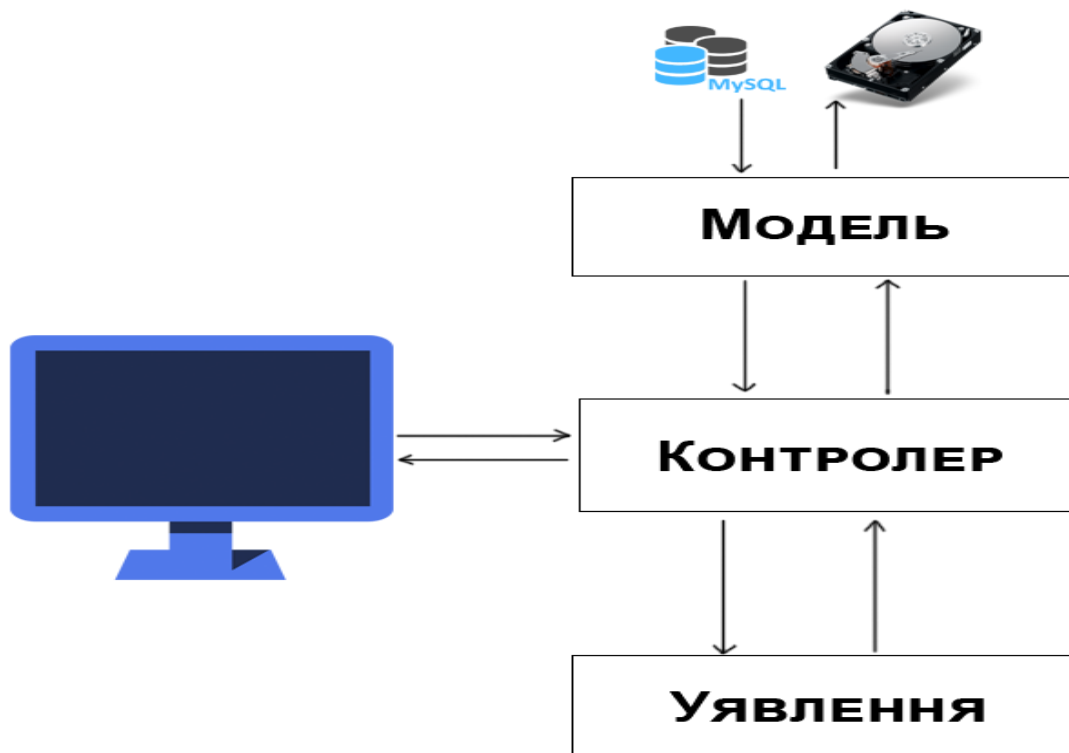


Рисунок 2.4 – Приклад MVC - моделі

Крім відділення логіки, модель MVC зменшує складність великих додатків, код виходить набагато більш структурованим і тим самим полегшується підтримка, тестування і повторне використання рішень.

2.7 Веб-технологія AJAX

Для якісної роботи браузера з сервером одного PHP занадто мало; для того щоб посилати запити на сервер і не перезавантажувати постійно сторінку, потрібна веб-технологія Asynchronous Javascript and XML, простіше говорячи AJAX.

AJAX – це побудова інтерфейсів веб-додатків які акцентуються на фоновому обміні даними браузером й веб-сервером. Говорячи простіше, AJAX дозволяє відправляти запит на сервер й отримувати його не перезавантажувати сторінку.

Особливості AJAX:

1. Формування зручного веб-інтерфейсу.

2. Взаємодія зі споживачем.
3. Часткове виконання сторінки.
4. Зручність використання.

AJAX на залежна методика, вона вживає кілька технологій. AJAX застосовує два принципи роботи зі сторінкою, зміна сторінки без перезавантаження її, і швидке звернення до сервера.

Переваги технології:

1. Економія трафіку Інтернет користувачами сторінкою.
2. Зменшення навантаження на сервер. Якщо правильно реалізувати AJAX, можливо в декілька раз зменшити навантаження на сервер.
3. Прискорення заміни частин сторінки, без перезавантаження сторінки цілком.

Недоліки:

1. Відсутність підтримки інструментами браузера.
2. Потрібен включений JavaScript в браузері, але це все мабуть й не зовсім недолік: пройшли ті часи коли багато людей виключали JS у своєму браузері.
3. Погана поведінка на слабких зв'язках з Інтернетом.
4. Старі відстеження й ведення обліку статистики відвідуваності не бачать вміст завантаженої AJAX, вони ведуть облік тільки нових сторінок.

2.8 Бази даних

Важливою технологією побудови системи в Інтернеті є бази даних (БД). Бази даних потрібні для зберігання, обробки даних й отримання доступу за допомогою мови запитів SQL.

Завданням баз даних є збереження даних, та надання доступу до них користувачеві або програмі. Таким чином БД складається з двох частин: збереженої інформації та системи управління нею.

В даний час є багато різних реалізацій баз даних:

- MySQL;
- PostgreSQL;
- Firebird;
- Oracle;
- SQLite та інші.

2.8.1 Порівняння MySQL та PostgreSQL

Для реалізації сайтів найчастіше використовують MySQL та PostgreSQL. Гнучкість MySQL та PostgreSQL дозволяє використовувати велику кількість різновидів таблиць. MySQL та PostgreSQL портировані на усі відомі платформи ОС.

MySQL - це найпоширеніша повноцінна серверна СУБД. MySQL дуже функціональна, безкоштовна, яка успішно працює з різними сайтами і веб додатками. Навчитися використанню цієї СУБД досить просто. MySQL пропонує досить багато інструментів для розробки додатків.

Переваги MySQL:

- простота в роботі;
- великий функціонал;
- велика кількість функцій які забезпечують безпеку;
- MySQL легко працює з великими об'ємами даних;
- швидкість.

Недоліки:

- деякі обмеження на функціонал;
- проблеми з надійністю;
- повільна розробка.

PostgreSQL – є найбільш професійною СУБД. На відміну від інших СУБД, PostgreSQL підтримує об'єктно-орієнтовані і реляційні підходи до баз даних. PostgreSQL легко розширюється за допомогою своїх збережених процедур. Збереження процедур – допомагає використовувати часто повторювані операції.

Переваги PostgreSQL:

- велика кількість доповнень;
- відкрите ПО;
- розширення функціоналу;
- об'єктно-орієнтована підтримка.

Недоліки PostgreSQL:

- якщо проект не великий, PostgreSQL сильно може загальмувати сервер і працювати на багато повільніше ніж MySQL;
- складно знайти хостинг з цією СУБД.

Із-за того, що в дипломному проекті не потрібні потужні технології для проектування системи, буде використана MySQL. Функціоналу цієї СУБД досить, щоб реалізувати базу даних веб-системи продажу автомобілів .

3 РОЗРОБКА СИСТЕМИ

3.1 Розробка структури системи

Система призначена для продажу-покупки вживаних автомобілів, та продажу авто-запчастин.

Система буде забезпечувати:

1. Зручний вибір автомобілів з повною інформацією про кожне авто.
2. Зручний вибір запчастин.
3. Спрощення покупки запчастин користувачу.
4. Сторінка з усіма актуальними новинами.

Програмне забезпечення на клієнтській частині спроектоване на HTML, CSS, JavaScript, а серверна частина - на мові PHP, використовуючи MVC-модель для об'єднання серверної й клієнтської частин. Система працюватиме з базою даних Microsoft MySQL.

Структура бази даних системи представлена на рис. 3.1. Структура системи зображена на рис. 3.2.

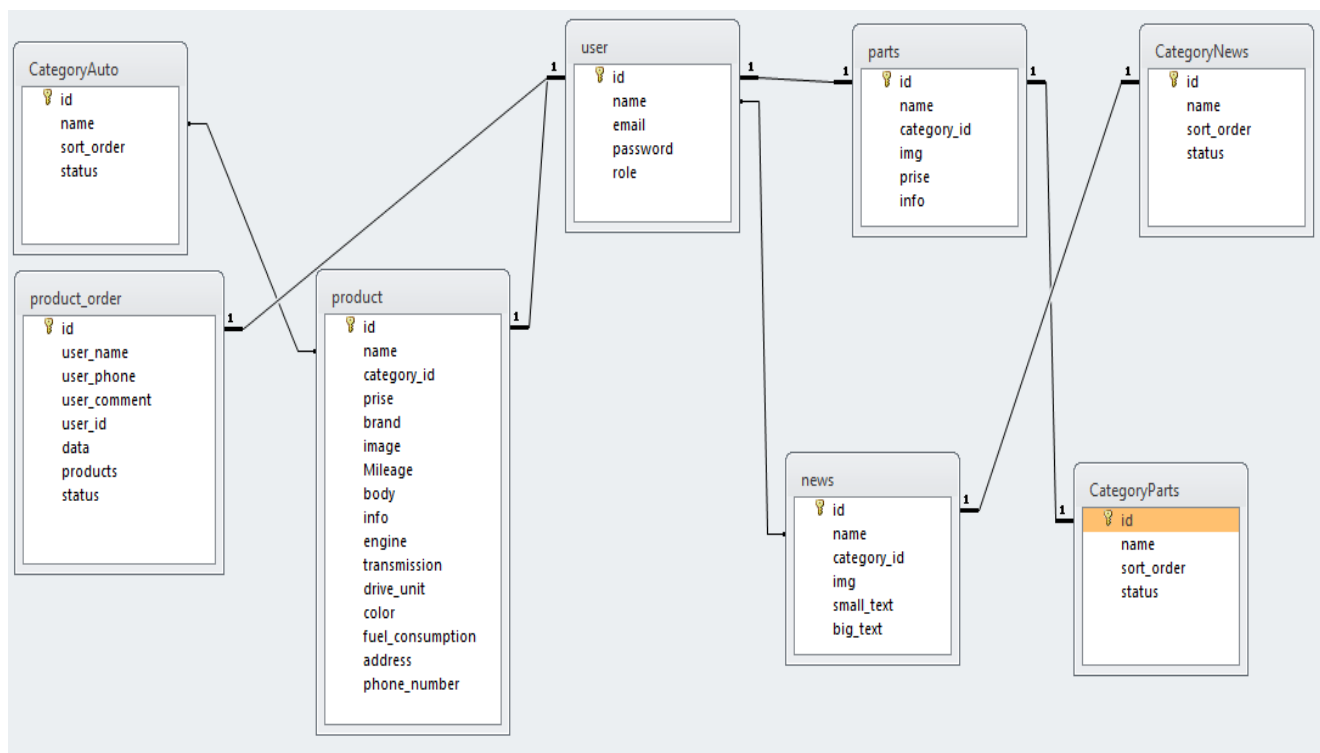


Рисунок 3.1 – Структура бази даних

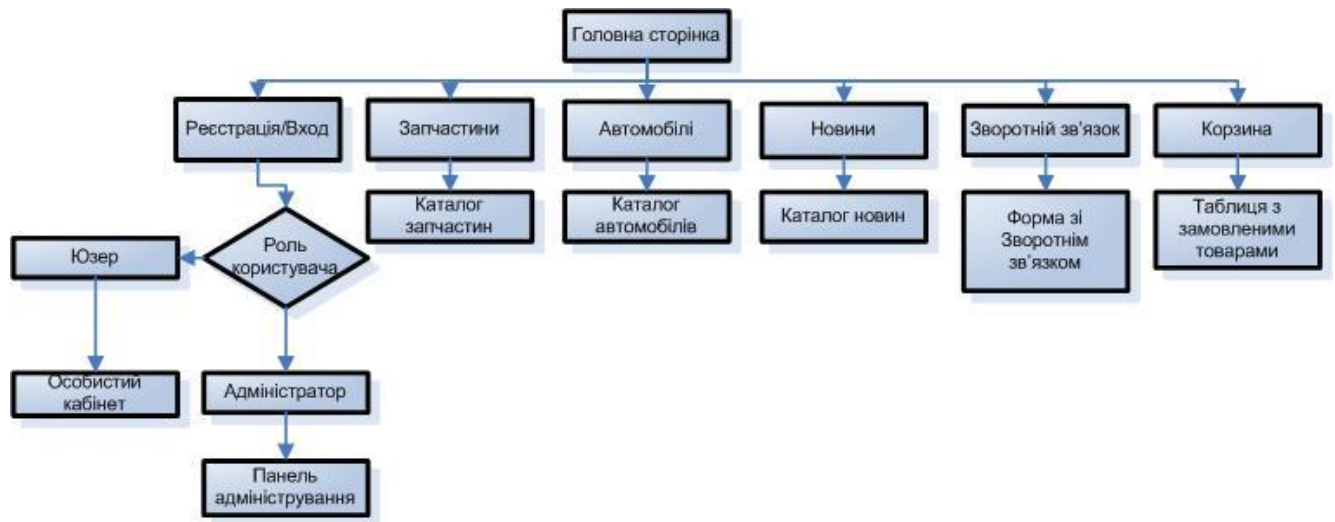


Рисунок 3.2 – Структура системи.

3.2 Система реєстрації користувачів

Обов'язковою частиною системи, є реєстрація користувачів. Реєстрація спроектована таким чином, щоб не заважати користувачу використовувати систему. В системі немає якогось обов'язкового реєстрування, основні сторінки будуть доступні і без реєстрації, але виконувати управління може тільки авторизований користувач з роллю адміністратора.

Схема алгоритму роботи реєстрації та авторизації зображені на рис. 3.3.

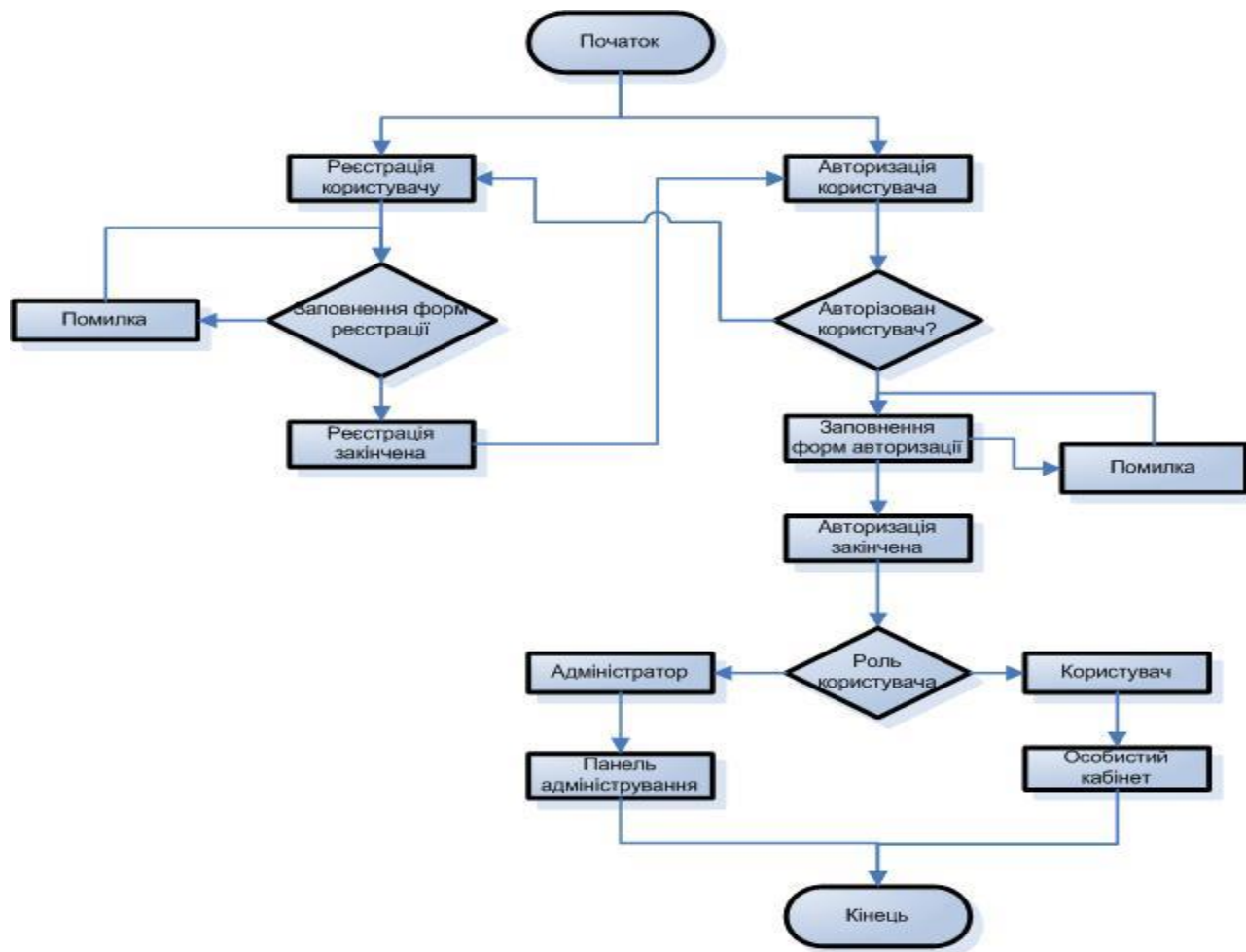


Рисунок 3.3 - Схема алгоритму роботи реєстрації та авторизації.

Реєстрація побудована таким чином, що користувачу потрібно увійти в вікно реєстрації, натиснувши на кнопку в верхньому меню, яке зображено на рис. 3.4, або якщо з мобільного пристрою, то в розгорнутому меню, й вибрати пункт реєстрації (рис. 3.5).

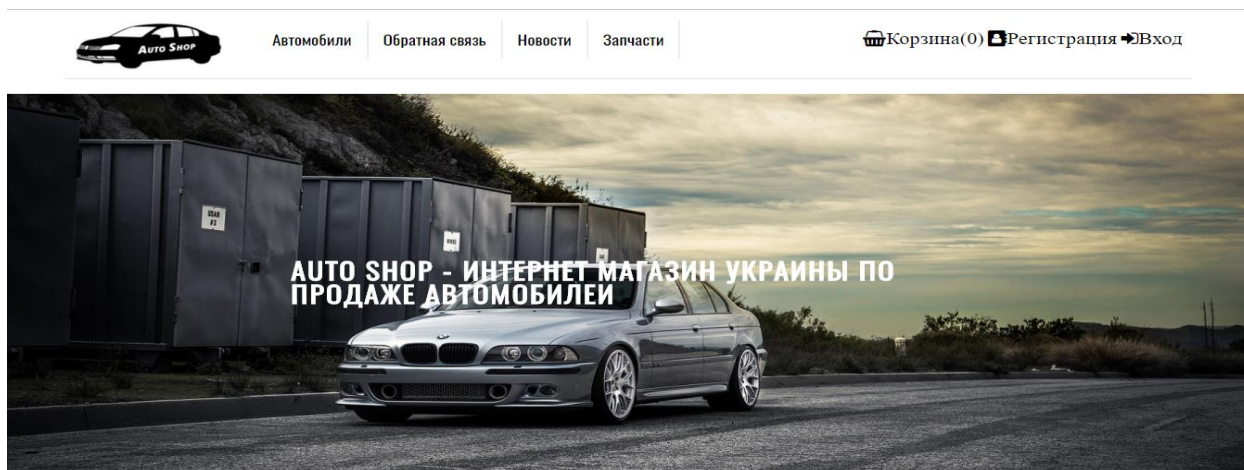


Рисунок 3.4 – Головна сторінка.

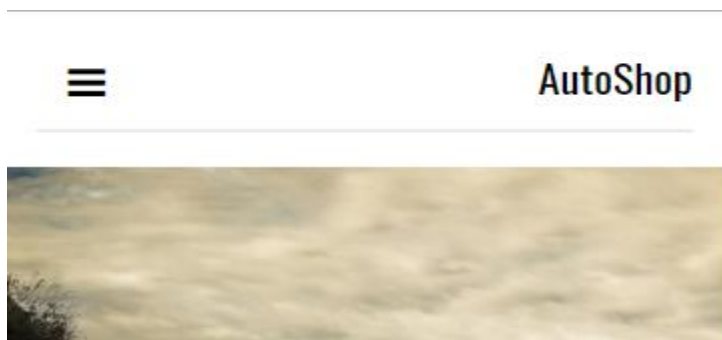


Рисунок 3.5 – Головна сторінка на мобільному пристрої.

Після вибору пункту меню реєстрації, користувачу буде відкрито форму реєстрації (рис.3.6) куди потрібно ввести “Имя”, ”Email адресс” та “Уникальный пароль” і якщо усі поля заповнені правильно данні користувача потраплять в базу даних системи.

Регистрация

Рисунок 3.6 – Форма Реєстрації на сайті.

Після реєстрації потрібно увійти в систему, натиснувши на кнопку “Вход” у верхньому меню сторінки. Після чого з’явиться форма авторизації (рис.3.7), в яку потрібно ввести “Email”, “Пароль” які були задані при реєстрації.

Авторизуйтесь

Рисунок 3.7 – Форма Авторизації на сайті.

Якщо під час авторизації буде зроблена помилка, то появиться ще раз вікно авторизації та повідомлення про помилку (рис.3.8).

- Неправильные данные для входа на сайт

Авторизуйтесь

Логин

Рисунок 3.8 – Помилка при авторизації.

Коли користувач увійде в свій акаунт, він потрапить в особистий кабінет, де зможе, якщо захоче, змінити свої данні (рис.3.9), або якщо це адміністратор, то зможе увійти в адмін-панель й здійснювати управління системою.

Редактирование данных!

Замена

Рисунок 3.9 – Редагування даних користувача.

Серверна частина для реєстрації та авторизації складається з одного контролера UserController та моделі User. В контролері розташовані дії (actions), які отримують данні з форм реєстрації та авторизації, потім звертаються до моделі User, яка перевіряє поля на валідність. Якщо помилок немає, то модель User звертається до бази даних й передає їй дані, якщо це реєстрація нового користувача, а якщо це аутентифікація, то отримує данні користувача з бази даних й запам'ятовує id користувача в сесії.

3.3 Панель адміністрування

Управління системою спроектоване таким чином, що усі основні функції управління системою виконувались через панель адміністрування. Панель адміністрування допомагає користувачу додавати, видаляти й редагувати автомобілі, запчастини і новини в системі, не відкриваючи код, а використовуючи спеціальний інтерфейс, який полегшує в декілька разів роботу з системою. Блок-схема панелі адміністрування зображена на рис. 3.10.

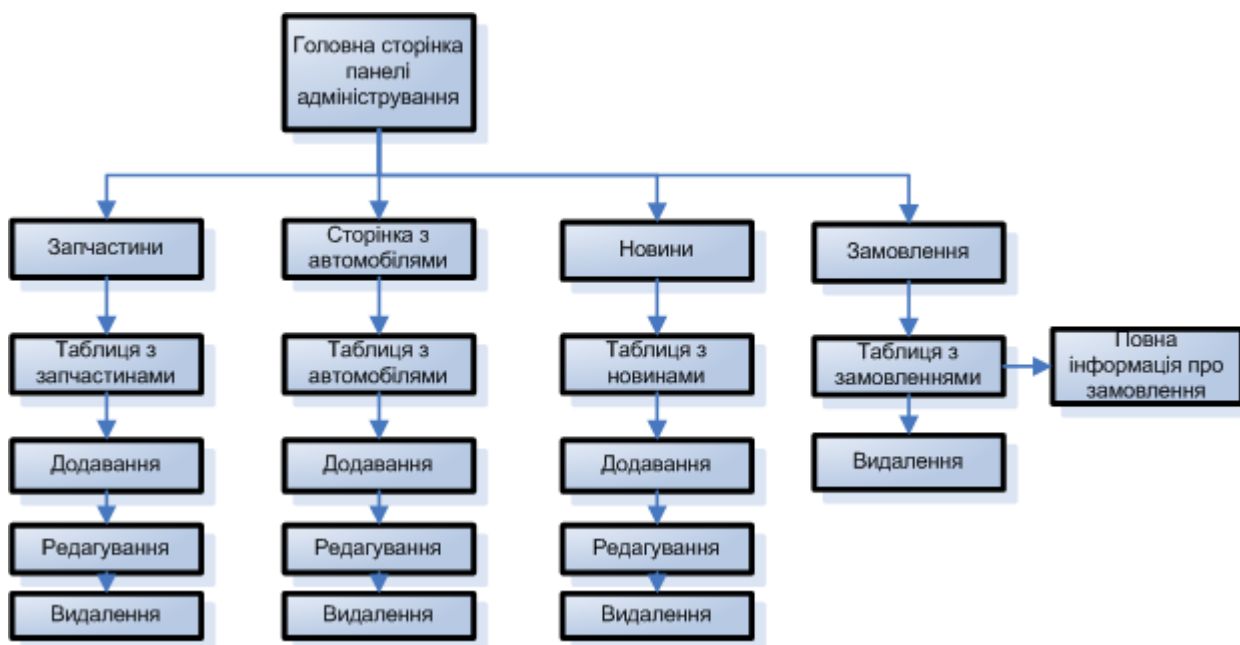


Рисунок 3.10 – Блок-схема панелі-адміністрування

Усі пункти меню в панелі адмініструванні представлені списком для зручного користування й управління системою (рис.3.11).

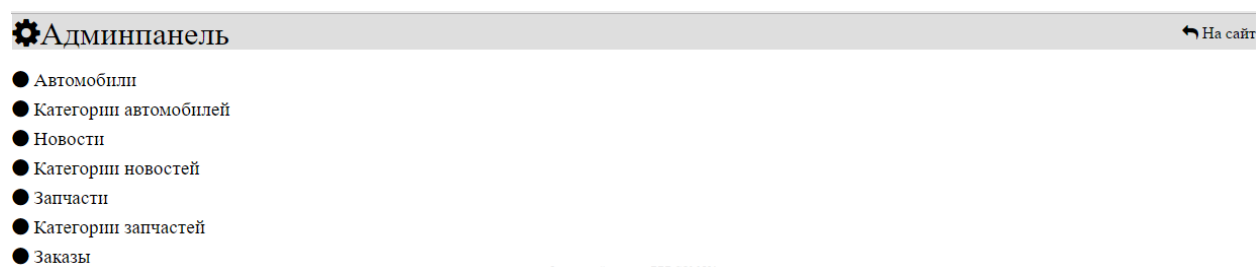
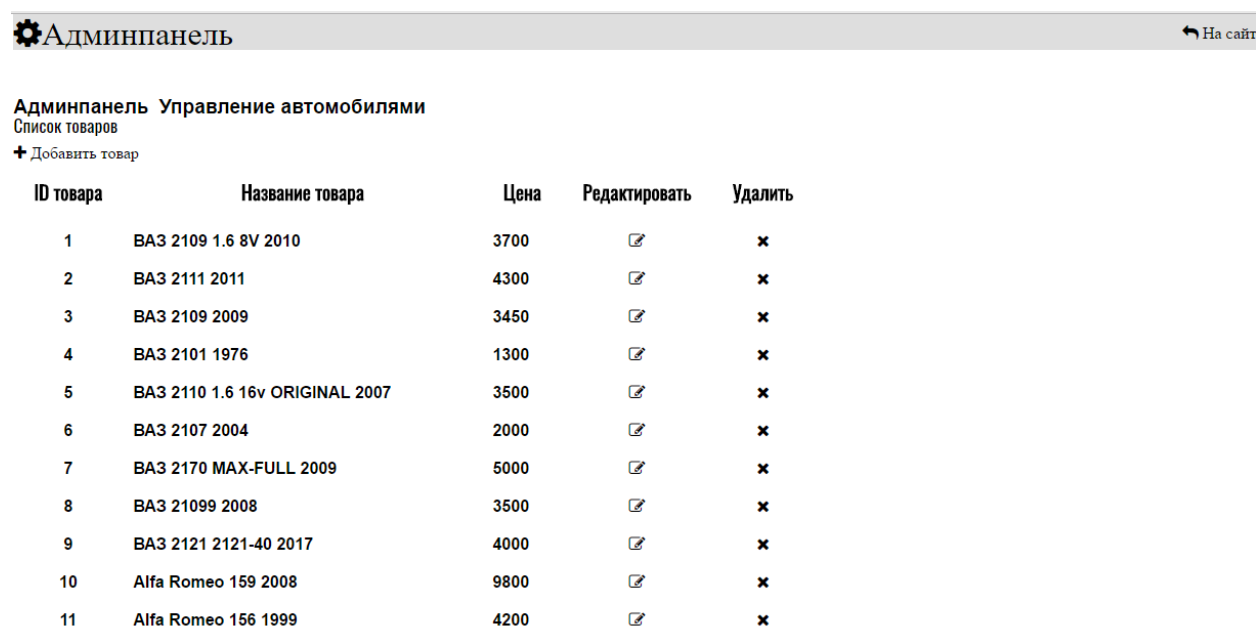


Рисунок 3.11 – Головна сторінка панелі-адміністрування.

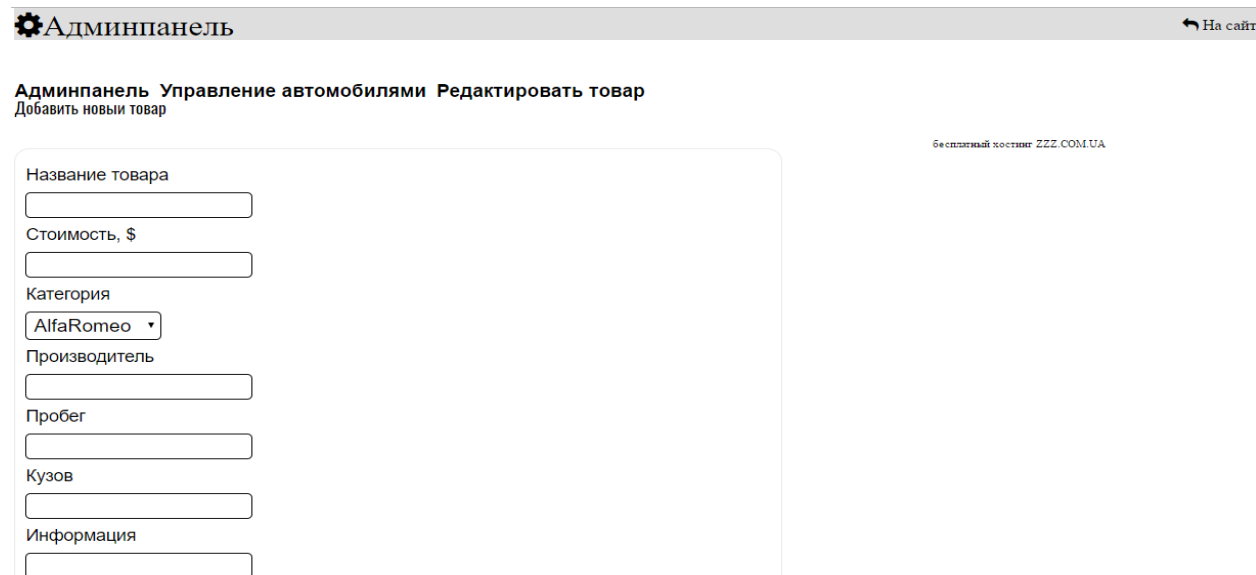
При натисненні на пункт меню “Автомобіли”, відкривається таблиця з управлінням автомобілями (рис.3.12), де можливо видалити, додати або змінити опис вже існуючого авто.



ID товара	Название товара	Цена	Редактировать	Удалить
1	ВАЗ 2109 1.6 8V 2010	3700		
2	ВАЗ 2111 2011	4300		
3	ВАЗ 2109 2009	3450		
4	ВАЗ 2101 1976	1300		
5	ВАЗ 2110 1.6 16v ORIGINAL 2007	3500		
6	ВАЗ 2107 2004	2000		
7	ВАЗ 2170 MAX-FULL 2009	5000		
8	ВАЗ 21099 2008	3500		
9	ВАЗ 2121 2121-40 2017	4000		
10	Alfa Romeo 159 2008	9800		
11	Alfa Romeo 156 1999	4200		

Рисунок 3.12 – Таблица управління автомобілями

При натисканні на пункт “Добавить товар” з’являється нова сторінка (рис.3.13), де адміністратор після заповнення усіх пунктів зможе додати новий автомобіль на продаж, який потім відобразиться на сторінці “Автомобіли” (рис.3.14).



Админпанель Управление автомобилями Редактировать товар
Добавить новый товар

Бесплатный хостинг ZZZ.COM.UA

Название товара

Стоимость, \$

Категория
AlfaRomeo ▾

Производитель

Пробег

Кузов

Информация

Рисунок 3.13 – Додавання нового автомобіля

Рисунок 3.14 – Сторінка з автомобілями

Для того, щоб видалити вже існуючі в системі автомобілі, потрібно натиснути в таблиці на кнопку видалення авто, після чого відкриється нова сторінка з видаленням (рис.3.15), або якщо потрібно відредагувати автомобіль, потрібно натиснути на кнопку редагування в таблиці адміністрування, після чого відкриється нова сторінка з редагуванням автомобілів (рис.3.16).

Рисунок 3.15 – Видалення існуючого автомобілю

Админпанель ↩ На сайт

Админпанель Управление товарами Редактировать товар
 Редактировать товар #1

бесплатный хостинг ZZZ.COM.UA

Название товара

Стоимость, \$

Категория

Производитель

Пробег

Кузов

Двигатель

Рисунок 3.16 – Зміна товару

Адміністративна панель, це невід’ємна частина системи, яка допомагає адміністратору зручно й швидко додавати нові товари, новини, автомобілі й обробляти замовлення.

Серверна частина адміністративної панелі, як і уся система написана за допомогою MVC-моделі. Для кожного пункту меню адміністративної панелі є окремий контролер зі своїми методами, які визивають відповідну до запиту модель й виконують запит від адміністратора системи. Це дуже зручно, якщо потрібно щось змінити в системі: не потрібно шукати щось у великому коді, усе поділено на окремі файли з відповідними їм назвами.

3.4 Сторінка з автомобілями

Головною метою системи є продаж автомобілів. Для цього було спроектовано сторінку з автомобілями, де можна вибрати марку, модель автомобіля, подивитися на характеристики і отримати інформацію про продавця.


Сторінка з автомобілями (рис.3.17) поділена на два блоки. В першому блоці знаходиться меню зі списком автомобілів, у другому блоці самі автомобілі з короткою інформацією і посиланням на повну інформацію про автомобіль.

ВЫБЕРИТЕ МАРКУ АВТОМОБИЛЯ

- AlfaRomeo
- Audi
- BMW
- Chevrolet
- Ford
- Honda
- BA3
- Mercedes
- Mitsubishi
- Nissan
- Opel
- Subaru
- Toyota
- Volkswagen
- Renault



ALFA ROMEO 147 2008
7900\$
 Хэтчбек
 Машинка своя на полном фарше,
 эксплуатация с 2010 года




ALFA ROMEO 156 1999
4200\$
 Седан
 Усилитель руля, Эп. стеклоподъемники,
 Кондиционер, Климат контроль, Подогрев
 зеркал, Электропакет



Рисунок 3.17 – Сторінка з автомобілями


Якщо користувач захоче переглянути автомобілі однієї марки, він завжди зможе вибрати потрібну йому марку в панелі автомобілів, після чого сторінка оновиться і в блоці з автомобілями будуть відображатися автомобілі однієї марки.

Для того щоб відкрити основну інформацію про автомобіль, потрібно натиснути на фотографію або назву автомобіля й тоді відкриється нова сторінка (рис.3.18), де буде представлено основну інформацію авто й дані о продавцях даного авто.



[Автомобили](#)
[Обратная связь](#)
[Новости](#)
[Запчасти](#)

[Корзина\(0\)](#)
[Выход](#)
[Аккаунт](#)



Toyota Supra 1997

Цена: 20000 \$

Тип транспорта: Купе

Пробег: 10000

Двигатель: 3 л., Бензин

Коробка передач:

Привод: Задний

Цвет: Хромированная пленка

Расход топлива: 8.3

Адрес продавца: Украина, Херсон

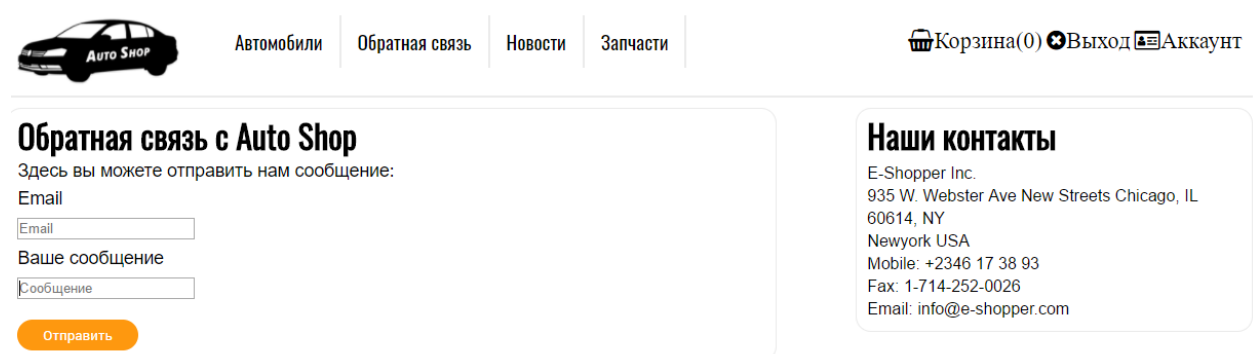
Номер телефона: +38(095) 421-12-41

Рисунок 3.18 – Основна інформація про автомобіль

Купити автомобіль через систему неможливо, так як покупцю все одно потрібно його оглянути й оформити на себе, але система надає можливість вибрати потрібне авто, прочитати характеристики й знайти адресу продавця, а це вже полегшує покупку автомобіля.

Самі авто в систему може додавати та редагувати тільки адміністратор, що зроблено для того, щоб не було обману в системі й усі авто, які виставляють на продаж, були заздалегідь перевірені на свій стан й легальність документів.

Саме для того, щоб користувачі могли додавати нові авто в систему, було зроблено сторінку з зворотним зв'язком (рис.3.19), де клієнт може написати повідомлення до адміністратора, або взяти номер телефона. Також на сторінці зі зворотним зв'язком можливо задати питання до адміністратора, пов'язане з покупкою авто, запчастин, а адміністратор може відповісти на поставлене питання.



Автомобили | Обратная связь | Новости | Запчасти

Корзина(0) | Выход | Аккаунт

Обратная связь с Auto Shop

Здесь вы можете отправить нам сообщение:

Email

Ваше сообщение

Отправить

Наши контакты

E-Shopper Inc.
935 W. Webster Ave New Streets Chicago, IL
60614, NY
Newyork USA
Mobile: +2346 17 38 93
Fax: 1-714-252-0026
Email: info@e-shopper.com

Рисунок 3.19 – Сторінка зі зворотним зв'язком

Серверна сторона сторінки зі зворотним зв'язком організована таким чином, що SiteController приймає інформацію з полів “Email” та “повідомлення”; якщо немає помилок в полі Email, то повідомлення буде відправлено на пошту адміністратора.

3.5 Сторінка з запчастинами та корзина

Для розширення магазину, було спроектовано сторінку з запчастинами (рис.3.20) на автомобілі. Сторінка з запчастинами являє собою два основні блоки, перший блок зі списком категорій запчастин, другий блок - це сам блок з запчастинами.

The screenshot shows the 'Auto Shop' website interface. At the top, there is a navigation menu with links for 'Автомобили', 'Обратная связь', 'Новости', and 'Запчасти'. On the right, there are links for 'Корзина(0)', 'Регистрация', and 'Вход'. On the left, there is a 'Категории' section with a list of categories: 'Двигатели', 'Тормозная система', 'Коробка', and 'Электроника'. The main content area displays four JVC car stereo models:

- JVC KD-G335**: Price: 130 \$, Information: Автомагнитола JVC KD-G335, Order button: Заказать.
- JVC KD-R48**: Price: 110 \$, Information: Автомагнитола JVC KD-R48, Order button: Заказать.
- JVC KD-R412**: Price: 130 \$, Information: Автомагнитола JVC KD- (partially visible), Order button: Заказать.
- JVC KD-DV6406**: Price: (not visible), Information: (not visible), Order button: (not visible).

Рисунок 3.20 – Сторінка з запчастинами

Якщо користувач захоче вибрати якийсь двигун, або коробку передач, йому потрібно вибрати потрібний пункт в меню “Категории”, після чого сторінка з товарами зміниться на ту, що йому потрібна.

Товари на сторінку запчастин може додавати, редагувати чи видаляти тільки адміністратор. Для цього в панелі адміністратора потрібно зайти на сторінку з запчастинами (рис.3.21).

Админпанель ↶ На сайт

Админпанель Управление запчастями
Список товаров
+ Добавить товар

ID товара	Название товара	Цена	Информация	Редактировать	Удалить
1	DICKASS system brembo	368	Суппорт для Mercedes-Benz AMG E 260 CGI 1.8 T		
2	Brembo	520	Суппорт Brembo для Porsche		
3	F-sport	300	тормозная система для Lexus		
4	Rinspeed	450	Rinspeed для Porsche 911		
5	1JZ-GTE	3500	1JZ-GTE первого поколения		
6	3UZ	2500	Двигатель Toyota UZ		
7	S50	4000	Двигатель BMW M3 E36		
8	GX100	3400	Коробка передач от Toyota Mark2		
9	AMT21-82	1500	Роботизированная коробка передач Лада Гранта		
10	GA70H	3400	Механическая коробка Toyota Supra		

Рисунок 3.21 – Сторінка з запчастинами в панелі адміністратора.

Якщо вибрати пункт “Добавить товар”, відкриється нова сторінка з додаванням товару (рис.3.22), де після заповнення усіх пунктів, запчастина буде додана на сторінку “Запчасти”.

Админпанель ↶ На сайт

Админпанель Управление запчастями Редактировать товар
Добавить новый товар

Бесплатный хостинг ZZZ.COM.UA

Название товара

Категория

Стоимость, \$

Информация

Изображение товара
 Файл не выбран

Рисунок 3.22 – Додавання товару в систему

Коли товар потрапить на “Інтернет-прилавок”, його можна відредагувати за допомогою адміністративної панелі, як зображено на рис.3.23, або зовсім видалити, як зображено на рис.3.24.

Админпанель [На сайт](#)

Админпанель Управление запчастями Редактировать товар
Редактировать запчасть #12

бесплатный хостинг ZZZ.COM.UA

Название товара
Pioneer DEH-X5500BT

Стоимость, \$
200

Категория
Электроника

Информация
Автомобильная магнитола Pioneer


Изображение товара

 |

Рисунок 3.23 – Редагування товару

Админпанель [На сайт](#)

Админпанель Управление запчастями Удалить запчасть
Удалить товар #12
Вы действительно хотите удалить этот товар?

бесплатный хостинг ZZZ.COM.UA

Рисунок 3.24 – Видалення товару

Після того як користувач вибере потрібний товар, йому потрібно натиснути на кнопку “Заказать”, після чого товар потрапить в корзину (рис.3.25), а користувач зможе далі вибирати собі запчастини, або перейти на саму корзину.

Корзина

Название	Стоимость, \$	Количество, шт	Удалить
Sony CDX-GT617UE	130	1	✘
Sony DSX-A30	180	1	✘
JVC KD-DV6406	130	1	✘
JVC KD-R412	130	1	✘
JVC KD-R48	110	1	✘
JVC KD-G335	130	1	✘
Общая стоимость:			810

бесплатный хостинг ZZZ.COM.UA

Рисунок 3.25 – Сторінка з корзиною

У самій корзині можливо подивитися на вибраний товар, загальну вартість товару і, якщо щось не влаштовує, видалити із списку, після чого натиснути на “Оформить заказ” й з'явиться форма для оформлення замовлення (рис.3.26).

Оформить заказ
 Выбрано товаров: 6, на сумму: 810, грн.

Ваше имя
 Сергей

Номер телефона
 Номер телефона

Комментарии к заказу
 Сообщение

Оформить

Рисунок 3.26 – Форма оформлення замовлення

Для того щоб адміністратору системи було зручно відповідати на замовлення покупців, була спроектована в панелі адміністрування сторінка з замовленнями (рис.3.27), яка представляє з себе таблицю зі списком замовлених товарів, з яким вже подалі може взаємодіяти.

Админпанель [На сайт](#)

Админпанель Управление заказами
 Список заказов

ID заказа	Имя покупателя	Телефон покупателя	Дата оформления	Статус	Просмотреть	Редактировать	Удалить
1	Сергей	0505229641	2017-06-06 21:28:45	Новый заказ			

Рисунок 3.27 – Сторінка замовлень

Щоб користувачу було зручно користуватися з замовленнями, були введені можливості редагування доданих замовлень, де адміністратор зможе, якщо це потрібно, змінити якусь інформацію про користувача, або змінити статус замовлення на “Новый заказ”, “В обработке”, “Доставляется”, “Закрыт”. Для того, щоб не засмічувати систему, є кнопка видалення оброблених замовлень.

Для зручного оброблення замовлень була спроектована сторінка з оглядом повної інформації про замовлення (рис.3.28).

Информация о заказе

Номер заказа	1
Имя клиента	Сергей
Телефон клиента	0505229641
Комментарий клиента	спасибо за предоставленные услуги
ID клиента	1
Статус заказа	Новый заказ
Дата заказа	2017-06-06 21:28:45

Товары в заказе

ID товара	Название	Цена	Количество
16	Sony CDX-GT617UE	\$130	1
17	Sony DSX-A30	\$180	1
18	JVC KD-DV6406	\$130	1
19	JVC KD-R412	\$130	1
21	JVC KD-G335	\$130	1

← Назад

Рисунок 3.28 – Сторінка з оглядом замовлення

На сторінці з оглядом замовлення можна подивитися повну інформацію, яку надає користувач при замовленні, а також подивитися товари, які замовив клієнт і, якщо не яких помилок не виявлено, то відправити замовлення на обробку.

4 ОХОРОНА ПРАЦІ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної роботи бакалавра було створення інформаційно-комп'ютерної системи керування інтернет магазином з продажу автомобілів, і як результат було створено інформаційно-комп'ютерна система "AutoShop".

Так як в процесі проектування використовувався комп'ютер, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблятися й використовуватися розроблена інформаційно-комп'ютерна система.

4.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Неправильна організація робочого місця сприяє

загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним. Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» законодавство про охорону праці складається з Закону, Кодексу законів про працю України, Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору.

На законодавчому рівні визначено такі пріоритетні напрямки з безпеки праці:

- кожен працівник несе безпосередню відповідальність за порушення зазначених Законом, нормами і правилами вимог;
- напрямки реалізації конституційного права громадян на їх життя і здоров'я в процесі трудової діяльності:
- пріоритет життя і здоров'я працівників по відношенню до результатів виробничої діяльності підприємства;
- повна відповідальність роботодавця за створення належних – безпечних і здорових умов праці;
- соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- комплексне розв'язання завдань охорони праці;
- підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;
- соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- використання економічних методів управління охороною праці, участь держави у фінансуванні заходів щодо охорони праці;
- використання світового досвіду організації роботи щодо поліпшення умов і підвищення безпеки праці на основі міжнародної співпраці.

Користувачі персональних комп'ютерів, для яких ця робота є головною, підлягають медичним оглядам: попереднім — під час влаштування на роботу і періодичним — протягом професійної діяльності раз на два роки.

Наявні трудові відносини між працівниками і роботодавцями в Україні за темою дипломного проекту регулюються Кодексом законів про працю (КЗпП) України, відповідно до якого права працюючої людини на охорону праці

охороняються всебічно та норми охорони праці неухильно інтегровані до правил внутрішнього розпорядку організації/підприємства.

4.1.1 Правові та організаційні основи охорони праці

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України, Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

На законодавчому рівні визначено такі пріоритетні напрямки з безпеки праці:

- кожен працівник несе безпосередню відповідальність за порушення зазначених Законом, нормами і правилами вимог;
- напрямки реалізації конституційного права громадян на їх життя і здоров'я в процесі трудової діяльності;
- пріоритет життя і здоров'я працівників по відношенню до результатів виробничої діяльності підприємства;
- повна відповідальність роботодавця за створення належних – безпечних і здорових умов праці;

- соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- комплексне розв'язання завдань охорони праці;
- підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;

Користувачі персональних комп'ютерів, для яких ця робота є головною, підлягають медичним оглядам: попереднім — під час влаштування на роботу і періодичним — протягом професійної діяльності раз на два роки. Жінок з часу встановлення вагітності та в період годування дитини грудьми до роботи з ПК не допускають.

Наявні трудові відносини між працівниками і роботодавцями в Україні за темою дипломного проекту регулюються Кодексом законів про працю (КЗпП) України, відповідно до якого права працюючої людини на охорону праці охороняються всебічно та норми охорони праці неухильно інтегровані до правил внутрішнього розпорядку організації/підприємства.

4.1.2 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці НПАОП 0.00-4.12 [26].

Обов'язковими вимогами враховане наступне:

- не слід допускати до роботи осіб, що в установленому порядку не пройшли навчання, інструктаж та перевірку знань з охорони праці, пожежної безпеки та цих Правил.
- на підприємстві/організації, де експлуатуються ЕОМ з відео дисплейними терміналами (ВДТ) і периферійними пристроями (ПП), розробляється

інструкція з охорони праці відповідно до Положення про розробку інструкцій з охорони праці, затвердженого наказом Держнаглядохоронпраці від 29.01.98 N 9 НПАОП 0.00-4.15 [27].

- ознайомлення з правилами безпеки праці, одержання відповідних інструктажів засвідчується у журналі інструктажів.
- перед допуском до самостійної роботи кожен працівник має право на навчання з питань охорони праці і роботодавець зобов'язаний, і проводить таке навчання у вигляді двох інструктажів з питань охорони праці:

1) вступного, який проводять працівники служби охорони праці об'єкта господарювання з усіма працівниками, яких приймають на роботу незалежно від їхньої освіти та стажу роботи за програмою, в якій подають загальні питання охорони праці із врахуванням її особливостей на об'єкті господарювання;

2) первинного, який проводять керівники структурних підрозділів на місці праці з кожним працівником до початку їхньої роботи на цьому робочому місці.

Проходження працівником цих інструктажів з питань охорони праці підтверджується записами у відповідних журналах обліку інструктажів і скріплюється підписами осіб, які проводили інструктажі та осіб, які отримали інструктажі.

3) Повторний (не рідше одного разу в 6 місяців);

4) Позаплановий (при зміні правил охорони праці);

5) Поточний (проводять з працівниками перед виконанням робіт, на яких оформляється наряд-допуск)

– обов'язкові організаційні заходи перед початком, під час і після завершення роботи повинні включати перевірку (візуально) наявності і справності електрообладнання та його заземлення, а під час виконання роботи вимогу «не залишати без нагляду обладнання, яке працює». Після закінчення роботи - вимагається прибирання робочого місця, відключення всіх електроприладів від електромережі.

Не допускається:

- виконувати обслуговування, ремонт та налагодження ЕОМ з ВДТ і ПП безпосередньо на робочому місці оператора;
- зберігати біля ЕОМ з ВДТ і ПП папір, дискети, інші носії інформації, запасні блоки, деталі тощо, якщо вони не використовуються для поточної роботи;
- відключати захисні пристрої, самочинно проводити зміни у конструкції та складі ЕОМ з ВДТ і ПП або їх технічне налагодження;
- працювати з ВДТ, у яких під час роботи з'являються нехарактерні сигнали, нестабільне зображення на екрані тощо;
- працювати з матричним принтером за відсутності вібраційного килимка та зі знятою (піднятою) верхньою кришкою.

4.2 Аналіз стану умов праці

Робота над створенням інформаційно-комп'ютерної системи проходитиме в приміщенні відповідної установи. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером

4.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в табл. А.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	6
Ширина, м	5
Висота, м	3
Площа, м ²	30
Об'єм, м ³	90

Згідно з ДСН 3.3.6.042 [16] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

4.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за ДСанПіН 3.3.2.007 [15] (табл. 4.2) і відповідними фактичними значеннями для робочого місця, контактуємо повну відповідність.

Таблиця 4.2 – Характеристика робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	700	680 ÷ 800
Висота простору для ніг, мм	680	не менше 600
Ширина простору для ніг, мм	620	не менше 500
Глибина простору для ніг, мм	680	не менше 650
Висота поверхні сидіння, мм	420	400 ÷ 500
Ширина сидіння, мм	410	не менше 400
Глибина сидіння, мм	420	не менше 400
Висота поверхні спинки, мм	330	не менше 300
Ширина опорної поверхні спинки, мм	470	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	720	700 ÷ 800

Робочий стіл на досліджуваному місці також містить достатньо простору для ніг. Крісло, що використовується в якості робочого сидіння, є підйомно

поворотним, має підлокітники і можливість регулювання за висотою і кутом нахилу спинки, також воно м'яке і виконане з екологічної шкіри, що дає можливість працювати у комфорті. Екран монітору знаходиться на відстані 0.8 м, клавіатура має можливість регулювання кута нахилу 5-15°. Отже, за всіма параметрами робоче місце відповідає нормативним вимогам. Приміщення кабінету знаходиться на другому поверсі трьох поверхової будівлі і має об'єм 78 м³, площу – 18 м².

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість - близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум в лабораторії знаходиться на рівні 50 дБА. Система вентилявання приміщення — повітряно-охолоджувальна, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою трьох ламп, забезпечує рівень освітленості не менше 200 Лк.

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет оснащений переносним вуглекислотним вогнегасником ВВК-5 .

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.

4.2.3 Навантаження та напруженість процесу праці

Як приклад наведено опис процесу праці оформлення дипломного проекту під час виконання випускної роботи бакалавра:

за фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування

виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму. Такі перекоси у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Тобто неявне психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- статичного;
- динамічного;

б) нервово-психічного перевантаження:

- розумового перенапруження;
- монотонності праці;
- перенапруження аналізаторів;
- емоційних перевантажень.

Рекомендовано застосування екранних фільтрів, локальних світлофільтрів (засобів індивідуального захисту очей) та інших засобів захисту, а також інші профілактичні заходи на ведені в ДСанПіН 3.3.2.007 [15].

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви:

- для розробників програм тривалістю 15 хв через кожну годину роботи;
- для операторів персональних комп'ютерів тривалістю 15 хв через дві години роботи;
- для операторів комп'ютерного набору тривалістю 10 хв через кожну годину роботи.

4.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3). Основними робочими характеристиками персонального комп'ютера є:

- робоча напруга $U=+220\text{В} \pm 5\%$;
- робочий струм $I=2\text{А}$;
- споживана потужність $P=350\text{ Вт}$.

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні шкідливі виробничі фактори	Джерела факторів	Кількіс на оцінка	Нормативні документи
1	2	3	4
фізичні			
Підвищена температура поверхонь обладнання	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	2	[16]
Підвищений рівень шуму на робочому місці	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	2	[17]
Підвищений рівень вібрації	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	2	[18]
Підвищена або знижена вологість повітря	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	3	[16]
Підвищена або знижена рухливість повітря	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	4	[16]
Підвищений рівень іонізуючого випромінювання в робочій зоні	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	1	[16] [19]
Підвищений рівень електромагнітного випромінювання	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	2	[19]
Підвищений рівень напруги електричної мережі, замикання якої може відбуватися через тіло людини	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	2	[20] [21]
Підвищена напруженість електричного поля	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	1	[20]

Продовження таблиці 4.3

Підвищена напружність магнітного поля	Експлуатація ЕОМ, принтерів, сканерів або серверного обладнання для роботи	3	[19]
Недостатність природного світла	Порушення умов праці (вимоги до приміщень)	2	[19]
Недостатнє освітлення робочої зони	Порушення гігієнічних параметрів виробничого середовища	2	[22]
Підвищена яскравість світла	Порушення умов праці (організація місця)	2	[22]
Понижена контрастність	Порушення умов праці (організація місця)	1	[15]
Психофізіологічні:			
Нервово-психічна перевантаження	Пошук інформації для постановки теми, пошук та аналіз аналогів, виконання роботи за темою, оформлення роботи	1	[15] [23]
Фізичне(статичне-сидіння)	Порушення умов праці ,організація робочого місця	2	[23]

4.3.2 Пожежна безпека

Пожежна безпека при застосуванні ЕОМ забезпечується:

- 1) системою запобігання пожежі;
- 2) системою протипожежного захисту;
- 3) організаційно-технічними заходами.

Потенційними джерелами запалювання можуть бути:

- 1) іскри і дуги короткого замикання;
- 2) електрична іскра при замиканні і розмиканні ланцюгів;
- 3) перегріву від тривалого перевантаження;
- 4) відкритий вогонь і продукти горіння;
- 5) наявність речовин, нагрітих вище за температуру самозаймання;
- 6) розрядна статична електрика.

Причинами можливого загоряння і пожежі можуть бути:

- 1) несправність електроустановки;
- 2) конструктивні недоліки устаткування;
- 3) коротке замикання в електричних мережах;
- 4) запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки.

4.3.3 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три- провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

4.4 Гігієнічні вимоги до параметрів виробничого середовища

4.4.1 Освітлення

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

Розрахунок освітлення.

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше $1/8$, в побутових – $1/10$:

$$S_b = \left(\frac{1}{8} \div \frac{1}{10} \right) \times S_n, \quad (4.1)$$

де S_b – площа віконних прорізів, m^2 ;

S_n – площа підлоги, m^2 .

$$S_n = a \cdot b = 5 \cdot 5 = 25 \text{ м}^2,$$

$$S = 1/8 \cdot 25 = 3,125 \text{ м}^2.$$

Приймаємо 2 вікна площею $S=1,6 \text{ м}^2$ кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні.

Розрахунок кількості світильників n виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (4.2)$$

де N_c – кількість світильників;

E – нормована освітленість;

S – площа підлоги, кв.м ($S = 30 \text{ м}^2$);

Z – поправочний коефіцієнт світильника (для стандартних світильників $Z = 1.1 - 1.3$);

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації ($k = 1,1-1,3$);

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і тому подібне – 0.33;

M – число люмінесцентних ламп в світильнику;

F – світловий потік - 3000лм.

Згідно вимог ДБН В.2.5-28:2015 [22] зорова робота при експлуатації персонального комп'ютера відноситься до робіт середньої зорової точності (розряд IV, підрозряд «а»), освітленість робочого місця оператора обчислювальної техніки повинна бути 300 лк. В результаті підстановки числових значень у формулу (4.2) кількість світильників:

$$N_c = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} = \frac{300 \cdot 30 \cdot 1.1 \cdot 1.3}{3000 \cdot 0.33 \cdot 2} = 6.5$$

Таким чином, одержуємо, що для освітлення робочого місця необхідно 6 світильників, по 2 лампи марки ЛБ-40 у кожному.

4.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

1) Заходи безпеки під час експлуатації персонального комп'ютера та периферійних пристроїв передбачають:

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;
- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;
- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала 2/3 нормальної освітленості приміщення);

- облаштовуюючи приміщення для роботи з ПК, потрібно передбачити припливно-витяжну вентиляцію або кондиціонування повітря:

2) Заходи безпеки під час експлуатації інших електричних приладів передбачають дотримання таких правил:

- постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;

- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;

- не тягнути за мережевий кабель, щоб витягти вилку з розетки;

- не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;

- не підключати одночасно декілька потужних електропристроїв до однієї розетки, що може викликати надмірне нагрівання провідників, руйнування їхньої ізоляції, розплавлення і загоряння полімерних матеріалів;

- не залишати включені електроприлади без нагляду;

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом НПАОП 40.1-1.01-97 [24], приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контура заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача η – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних заземлювачів η_v в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4...0,99. Взаємну екрануючу

дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η_c .

Послідовність розрахунку.

1) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_3 \cdot R_{пр.з.}}{R_{пр.з.} - R_3} \quad (4.3)$$

де $R_{пр.з.}$ – опір природних заземлювачів; R_d – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Підставивши числові значення у формулу (4.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом}$$

2) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом·м. Приблизне значення питомого опору глини приймаємо $\rho = 40$ Ом·м (табличне значення).

3) Розрахунковий питомий опір ґрунту, $\rho_{розр.}$, Ом·м, визначається відповідно для вертикальних заземлювачів $\rho_{розр.в.}$, і горизонтальних $\rho_{розр.г.}$, Ом·м за формулою:

$$\rho_{розр.} = \psi \cdot \rho, \quad (4.4)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів і кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{розр.в.} = 1.7$ і горизонтальних $\rho_{розр.г.} = 5.5$ Ом·м.

$$\rho_{розр.в.} = 1.7 \cdot 40 = 68 \text{ Ом} \cdot \text{м}$$

$$\rho_{розр.г.} = 5.5 \cdot 40 = 220 \text{ Ом} \cdot \text{м}$$

4) Розраховується опір розтікання струму вертикального заземлювача R_B , Ом, за (4.5).

$$R_B = \left(\ln \frac{2 \cdot l_B}{d_{ст}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_B}{4 \cdot t - l_B} \right), \quad (4.5)$$

де l_B – довжина вертикального заземлювача (для труб - 2–3 м; $l_B = 3$ м);

$d_{ст}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{ст} = 0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф. (4.6);

$$t = h_B + \frac{1_B}{2}, \quad (4.6)$$

де h_B – глибина закладання вертикальних заземлювачів (0,8 м); тоді $t = 0.8 + \frac{3}{2} = 2.3\text{м}$

$$R_B = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0.05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2.3 + 3}{4 \cdot 2.3 - 3} \right) = 18.5 \text{ Ом}$$

5) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_B :

$$n = \frac{2 \cdot R_B}{R_d} = \frac{2 \cdot 18.5}{4} = 9.25 \quad (4.7)$$

η визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки $\eta_B = 0.57$ (табличне значення).

б) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання η_B , шт:

$$n_B = \frac{2 \cdot R}{R_d \cdot \eta_B} = \frac{2 \cdot 18.5}{4 \cdot 0.57} = 16.2 \approx 16 \quad (4.8)$$

7) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_c ,

$$l_c = 1.05 \cdot L_B \cdot (n_B - 1), \quad (4.9)$$

де L_B – відстань між вертикальними заземлювачами, (прийняти за $L_B = 3\text{м}$);
 n_B – необхідна кількість вертикальних заземлювачів.

8) Визначається опір розтіканню струму горизонтального заземлювача R_r , Ом:

$$R_r = \frac{\rho_{\text{розр.г}}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l^2}{d_{\text{см}} \cdot h}, \quad (4.10)$$

$d_{\text{см}}$ – еквівалентний діаметр смуги шириною b , $d_{\text{см}} = 0.95b$, $b = 0.15$ м;

h_r – глибина закладання горизонтальних заземлювачів (0.5 м);

l_c – довжина з'єднувальної стрічки горизонтального заземлювача l_c , м

$$R_{\Gamma} = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0.95 \cdot 0.15 \cdot 0.5} = 8.1 \text{ Ом}$$

9) Визначається коефіцієнт використання горизонтального заземлювача η . відповідно до необхідної кількості вертикальних заземлювачів пв.

Коефіцієнт використання з'єднувальної смуги $\eta=0,3$ (табличне значення).

10) Розраховується результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_{\text{в}} \cdot R_{\Gamma}}{R_{\text{в}} \cdot n_{\text{с}} \cdot R_{\Gamma} \cdot n_{\text{в}} \cdot n_{\text{в}}} \leq R_{\text{д}}, \quad (4.11)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{\text{заг}} < 4 \text{ Ом}$, а саме:

$$R_{\text{заг}} = \frac{18.5 \cdot 8.1}{18.5 \cdot 0.3 \cdot + 8.1 \cdot 16 \cdot 0.57} = 1.9 \leq R_{\text{д}}$$

3) При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявність перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;
- необережному поводженню з вогнем, а також вибухи газо-повітряних і паро-повітряних сумішей.

ВИСНОВОК

В дипломному проекті була спроектована інформаційно-комп'ютерна система керування Інтернет магазином з продажу автомобілів під назвою “AutoShop”.

Був спроектований унікальний дизайн системи з реєстрацією, авторизацією, сторінкою вибору автомобілів, сторінкою вибору запчастин, сторінкою зворотного зв'язку, корзиною, новинами. При розробці системи використовувались технології HTML, CSS, JavaScript, JQuery, PHP та MySQL.

Була спроектована панель адміністрування системи за допомогою мови програмування PHP та моделі MVC. Була розроблена база клієнтів з поділенням рівнів користування системою для адміністратора та користувача. Спроектована система продажу автомобілів, описи яких зберігаються в базі даних та виводиться на сторінку автомобілів, де користувач може отримати повну інформацію про автомобіль та продавця, що полегшує роботу продавцю й покупцю автомобіля.

В системі також була реалізована можливість користувачів купувати запчастини для автомобілів. Для цього була спроектована сторінка з запчастинами, а панель адміністрування була доповнена можливістю додавати, редагувати, видаляти запчастини, також було спроектовано управління запчастинами в системі за допомогою панелі адміністрування. Усі замовлення додаються до панелі адміністрування, де вони обробляються адміністратором.

Розроблена система повністю відповідає вимогам технічного завдання.

У розділі “Охорона праці” був проведений аналіз небезпечних та шкідливих виробничих факторів. На основі цього аналізу запропоновані заходи безпеки.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Бэн Фрейн HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств – Перевод на русский язык ООО Издательство «Питер», 2014 — 304 с.
2. Брайан Хоган. HTML5 и CSS3. Веб-разработка по стандартам нового поколения. 2-е издание Пер. с англ.- Евгений Матвеев. 2014.-318с.
3. Илья Контор. Язык JavaScript часть 1. 2015.-634с.
4. Дэвид Флэнаган. JavaScript: карманный справочник, 3-е изд. : "Пер. с англ. - М . . ООО И.Д. Вильяме , 2013. - 320 с .
5. Дэвид Флэнаган . JavaScript. Подробное руководство. – Пер. с англ. – СПб: Символ_Плюс,2008. – 992 с.
6. Херман Д. Сила JavaScript. 68 способов эффективного использования JS. СПб.: Питер, 2013. — 288 с.
7. Фаулер М. Рефакторинг: улучшение существующего кода. Пер. с англ. – СПб: Символ_Плюс,2003. – 432 с.
8. Мак-Дональд М. HTML5. Недостающее руководство: Пер. с англ. — СПб.: БХВ-Петербург, 2012. — 480 с.
9. Котеров Д.В. PHP 7 I ,n;. В. Corepos, 11. В. СНМ.!UHOB. - СПб.: БХВ-Петербург, 2016. -1088 С.
- 10.Клименко Р. А. Веб-мастеринг на 100%. — СПб.: Питер, 2013. — 512 с.
- 11.Макфарланд Д. Новая большая книга CSS. — СПб.: Питер, 2016. — 720 с.
- 12.Бенедетти Р., Крэнли Р.Изучаем работу с jQuery. — СПб.: Питер, 2012. — 528 с.
- 13.Фримен Э., Робсон Э.Изучаем программирование на JavaScript. — СПб.: Питер, 2015. — 640 с.
14. Антон Шевчук. jQuery учебник для начинающих. 2016 – 156с.
- 15.ДСанПіН 3.3.2.007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.
- 16.ДСН 3.3.6.042-99. Санітарні норми мікроклімату виробничих приміщень.

- 17.ДСН 3.3.6.037-99.Санітарні норми виробничого шуму, ультразвуку та інфразвуку.
- 18.ДСН 3.3.6.039-99. Державні санітарні норми загальної та локальної вібрації.
- 19.ГОСТ 12.1.006-84.Електромагнітні поля радіочастот, допустимі рівні на робочих місцях.
- 20.ГОСТ 12.1.030-81.Електробезпе́чність. Захисне заземлення.
- 21.ГОСТ 13109-97.Норми якості електроенергоснабження загального призначення.
- 22.ДБН В.2.5-28:2015.Природні і штучне освітлення.
- 23.НПАОП 0.00-1.28-10. Правил охорони праці під час експлуатації електронно-обчислювальних машин
24. НПАОП 40.1-1.01-97. Правила безпечної експлуатації електроустановок.
- 25.НПАОП 0.00-4.12-05 Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці.
- 26.НПАОП 0.00-4.15-98 Про розробку інструкцій з охорони праці.
27. Кількість сайтів в сеті інтернет: <http://techno.bigmir.net/technology/1577735-Kolichestvo-sajtov-v-internete-perevalilo-za-milliard>.

ДОДАТОК А

Лістинги програм

Файл сторінки з автомобілями index.php:

```

<?php include ROOT.'/views/layouts/header.php'; ?>
<div class="container">
<div class="OptionsAuto">
    <div class="AutoBrands">
        <h3>Выберите марку автомобиля</h3>
        <ul>
            <?php foreach ($categories as $categoryItem): ?>
                <li>
                    <a href="/category/<?php echo $categoryItem['id'];?>">
                        <?php echo $categoryItem['name'];?>
                    </a>
                </li>
            <?php endforeach; ?>
        </ul>
    </div>
</div>
<div class="autoList">
    <?php foreach ($latestProducts as $product): ?>
        <div class="autoBlock">
            <a href="/product/<?php echo $product['id'];?>">
                </a>
            <div class="autoInfoContant">
                <a href="/product/<?php echo $product['id'];?>">
                    <?php echo $product['name'];?>
                </a>
            </div>
        </div>
    </?php endforeach; ?>
</div>

```

```

        </a>
        <span><?php echo $product['prise'];?>$</span>
        <p><?php echo $product['body'];?></p>
        <p><?php echo $product['info'];?></p>
    </div>
</div>
<?php endforeach; ?>
</div>
</div>
<!-- Блок с контентом заканчивается-->
</div>
<?php include ROOT.'/views/layouts/footer.php'; ?>

```

Файл сторінки категорій автомобілів `category.php`:

```

<?php include ROOT.'/views/layouts/header.php'; ?>
<div class="container">
    <div class="OptionsAuto">
    <div class="AutoBrands">
        <h3>Выберите марку автомобиля</h3>
        <ul>
            <?php foreach ($categories as $categoryItem): ?>
                <li>
                    <a href="/category/<?php echo $categoryItem['id'];?>"
                    class="<?php if ($categoryId == $categoryItem['id']) echo 'active';?>">
                        <?php echo $categoryItem['name'];?>
                    </a>
                </li>
            <?php endforeach; ?>
        </ul>
    </div>
    </div>
</div>

```

```

        </div>
</div>
<div class="autoList">
    <?php foreach ($categoryProducts as $products): ?>
        <div class="autoBlock">
            <a href="/product/<?php echo $products['id'];?>">
                </a>
                <div class="autoInfoContant">
                    <a href="/product/<?php echo $products['id'];?>">
                        <?php echo $products['name']?>
                    </a>
                    <span><?php echo $products['prise'];?>$</span>
                    <p><?php echo $products['body'];?></p>
                    <p><?php echo $products['info'];?></p>
                </div>
            </div>
        <?php endforeach; ?>
        <?php echo $pagination->get(); ?>
    </div>
</div>
<!-- Блок с контентом заканчивается-->
</div>
<?php include ROOT.'views/layouts/footer.php'; ?>

```

Файл сторінки адміністративної-панелі index.php:

```

<?php include ROOT.'views/layouts/headerAdmin.php'; ?>
    <ul class="adminInfo">

```

```

<li><a href="/admin/product"><i class="fa fa-circle">
Автомобили</i></a></li>
<li><a href="/admin/categoryAuto"><i class="fa fa-circle"> Категории
автомобилей</i></a></li>
<li><a href="/admin/contant"><i class="fa fa-circle">
Новости</i></a></li>
<li><a href="/admin/categoryNews"><i class="fa fa-circle"> Категории
новостей</i></a></li>
<li><a href="/admin/spareparts"><i class="fa fa-circle">
Запчасти</i></a></li>
<li><a href="/admin/categoryParts"><i class="fa fa-circle"> Категории
запчастей</i></a></li>
<li><a href="/admin/order"><i class="fa fa-circle"> Заказы</i></a></li>
</ul>
</div>
</div>
<!-- Блок с контентом заканчивается-->
</div>
</body>

```

Файл сторінки додавання авто в систему create.php:

```

<?php include ROOT . '/views/layouts/headerAdmin.php'; ?>
<div class="adminProduct">
<br/>
<div class="breadcrumbs">
<ul>
<li><a href="/admin">Админпанель</a></li>
<li><a href="/admin/categoryAuto">Управление категориями</a></li>
<li class="active">Добавить категорию</li>

```

```

    </ul>
</div>
<h3>Добавить новую категорию</h3>
<br/>
<?php if (isset($errors) && is_array($errors)): ?>
    <ul>
        <?php foreach ($errors as $error): ?>
            <li> - <?php echo $error; ?></li>
        <?php endforeach; ?>
    </ul>
<?php endif; ?>
<div class="adminForm">
    <form action="#" method="post">
        <p>Название</p>
        <input type="text" name="name" placeholder="" value="">
        <p>Порядковый номер</p>
        <input type="text" name="sort_order" placeholder="" value="">
        <p>Статус</p>
        <select name="status">
            <option value="1" selected="selected">Отображается</option>
            <option value="0">Скрыта</option>
        </select>
        <br><br>
        <input type="submit" name="submit" class="btn btn-default"
            value="Сохранить">
    </form>
</div>
</div>

```

Сторінка видалення автомобілів із системи delete.php:

```

<?php include ROOT . '/views/layouts/headerAdmin.php'; ?>
<div class="adminProduct">
  <div class="breadcrumbs">
    <ul>
      <li><a href="/admin">Админпанель</a></li>
      <li><a href="/admin/categoryAuto">Управление категориями</a></li>
      <li class="active">Удалить категорию</li>
    </ul>
  </div>
  <h4>Удалить категорию #<?php echo $id; ?></h4>
  <div class="adminForm">
    <p>Вы действительно хотите удалить эту категорию?</p>
    <form method="post">
      <input type="submit" name="submit" value="Удалить" />
    </form>
  </div>
</div>

```

Файл таблиці з автомобілями index.php:

```

<?php include ROOT . '/views/layouts/headerAdmin.php'; ?>
<div class="adminProduct">
  <div class="breadcrumbs">
    <ul>
      <li><a href="/admin">Админпанель</a></li>
      <li class="active">Управление категориями</li>
    </ul>
  </div>

```



```

<a href="/admin/categoryAuto/create" class="btn btn-default back"><i class="fa
fa-plus"></i> Добавить категорию</a>
<h4>Список категорий</h4>
<br/>
<table class="table-bordered table-striped table">
  <tr>
    <th>ID категории</th>
    <th>Название категории</th>
    <th>Порядковый номер</th>
    <th>Редактировать</th>
    <th>Удалить</th>
  </tr>
  <?php foreach ($categoriesList as $category): ?>
    <tr>
      <td><?php echo $category['id']; ?></td>
      <td><?php echo $category['name']; ?></td>
      <td><?php echo $category['sort_order']; ?></td>
      <td class="center"><a href="/admin/categoryAuto/update/<?php echo
$category['id']; ?>" title="Редактировать"><i class="fa fa-pencil-
square-o"></i></a></td>
      <td class="center"><a href="/admin/categoryAuto/delete/<?php echo
$category['id']; ?>" title="Удалить"><i class="fa fa-
times"></i></a></td>
    </tr>
  <?php endforeach; ?>
</table>
</div>

```

Файл контролера AdminProductController.php:

```
<?php
/**
 * Контроллер AdminProductController
 * Управление товарами в админпанели
 */
class AdminProductController extends AdminBase
{

/**
 * Action для страницы "Управление товарами"
 */
    public function actionIndex()
    {
        // Проверка доступа
        self::checkAdmin();
        // Получаю список товаров
        $productsList = Product::getProductsList();
        // Подключаю вид
        require_once(ROOT . '/views/admin_product/index.php');
        return true;
    }
/**
 * Action для страницы "Добавить товар"
 */
    public function actionCreate()
    {
        // Проверка доступа
```

```
self::checkAdmin());

// Получаю список категорий для выпадающего списка
$categoriesList = Category::getCategoriesListAdmin();

// Обработка формы
if (isset($_POST['submit'])) {
    // Если форма отправлена
    // Получаю данные из формы
    $options['name'] = $_POST['name'];
    $options['prise'] = $_POST['prise'];
    $options['category_id'] = $_POST['category_id'];
    $options['brand'] = $_POST['brand'];
    $options['Mileage'] = $_POST['Mileage'];
    $options['body'] = $_POST['body'];
    $options['info'] = $_POST['info'];
    $options['engine'] = $_POST['engine'];
    $options['drive_unit'] = $_POST['drive_unit'];
    $options['color'] = $_POST['color'];
    $options['fuel_consumption'] = $_POST['fuel_consumption'];
    $options['address'] = $_POST['address'];
    $options['phone_number'] = $_POST['phone_number'];

    $errors = false;
    // валидация полей
    if (!isset($options['name']) || empty($options['name'])) {
        $errors[] = 'Заполните поля';
    }
    if ($errors == false) {
```

```

// Если ошибок нет
// Добавляю новый товар
$хid = Product::createProduct($options);

// Если запись добавлена
if ($хid) {
    // загружалось ли через форму изображение
    if (is_uploaded_file($_FILES["image"]["tmp_name"])) {
        // Если загружалось, переместить его в нужную папку
        move_uploaded_file($_FILES["image"]["tmp_name"],
            $_SERVER['DOCUMENT_ROOT'] . "/upload/images/products/{$хid}.jpg");
    }
};
// Перенаправление пользователя на страницу управлениями товарами
header("Location: /admin/product");
}
}

// Подключение вида
require_once(ROOT . '/views/admin_product/create.php');
return true;
}

/**
 * Action для страницы "Редактировать товар"
 */
public function actionUpdate($хid)
{
    // Проверка доступа

```

```
self::checkAdmin());

// получение списка категорий для выпадающего списка
$categoriesList = Category::getCategoriesListAdmin();

// Получаю данные о конкретном заказе
$product = Product::getProductById($id);

// Обработка формы
if (isset($_POST['submit'])) {
    // Если форма отправлена
    // Получаю данные из формы редактирования. При необходимости можно
    валидировать значения
    $options['name'] = $_POST['name'];
    $options['prise'] = $_POST['prise'];
    $options['category_id'] = $_POST['category_id'];
    $options['brand'] = $_POST['brand'];
    $options['Mileage'] = $_POST['Mileage'];
    $options['body'] = $_POST['body'];
    $options['info'] = $_POST['info'];
    $options['engine'] = $_POST['engine'];
    $options['drive_unit'] = $_POST['drive_unit'];
    $options['color'] = $_POST['color'];
    $options['fuel_consumption'] = $_POST['fuel_consumption'];
    $options['address'] = $_POST['address'];
    $options['phone_number'] = $_POST['phone_number'];
    // сейв изменений
    if (Product::updateProductById($id, $options)) {

        // Если запись сохранена
```

```

// Проверка, загружалось ли через форму изображение
if (is_uploaded_file($_FILES["image"]["tmp_name"])) {

// Если загружалось, переместить его в нужную папку
move_uploaded_file($_FILES["image"]["tmp_name"],
$_SERVER['DOCUMENT_ROOT'] . "/upload/images/products/{$id}.jpg");
}
}

// Перенаправление пользователя на страницу управлениями товарами
header("Location: /admin/product");
}

// Подключение вида
require_once(ROOT . '/views/admin_product/update.php');
return true;
}

/**
 * Action для страницы "Удалить товар"
 */
public function actionDelete($id)
{
// Проверка доступа
self::checkAdmin();

// Обработка формы
if (isset($_POST['submit'])) {
// Если форма отправлена удаление товара по ид
Product::deleteProductById($id);
}
}

```

```

// Перенаправление пользователя на страницу управлениями товарами
header("Location: /admin/product");
}
// подключение вида
require_once(ROOT . '/views/admin_product/delete.php');
return true;
}
}

```

Файл с моделью Product.php:

```

<?php
class Product
{
    const SHOW_BY_DEFAULT = 6;

    /*получение массива с последними товарами*/
    public static function getLatestProducts($count = self::SHOW_BY_DEFAULT)
    {
        $count = intval($count);
        $db = Db::getConnection();
        $productsList = array();

        $result = $db->query('SELECT id, name, prise, image, body, info FROM product
ORDER BY id DESC LIMIT ' . $count);

        $i = 0;
        while ($row = $result->fetch()) {
            $productsList[$i]['id'] = $row['id'];

```

```

    $productsList[$i]['name'] = $row['name'];
    $productsList[$i]['prise'] = $row['prise'];
    $productsList[$i]['image'] = $row['image'];
    $productsList[$i]['body'] = $row['body'];
    $productsList[$i]['info'] = $row['info'];
    $i++;
}
return $productsList;
}

public static function getProductsListByCategory($categoryId = false, $page = 1)
{
    if ($categoryId) {
        $page = intval($page); // целое значение
        $offset = ($page - 1) * self::SHOW_BY_DEFAULT;

        $db = Db::getConnection();
        $products = array();
        $result = $db->query("SELECT id, name, prise, image, body, info FROM
product WHERE category_id = '$categoryId' ORDER BY id DESC LIMIT
".self::SHOW_BY_DEFAULT . " OFFSET " . $offset);
        $i = 0;
        while ($row = $result->fetch()) {
            $products[$i]['id'] = $row['id'];
            $products[$i]['name'] = $row['name'];
            $products[$i]['prise'] = $row['prise'];
            $products[$i]['image'] = $row['image'];
            $products[$i]['body'] = $row['body'];
            $products[$i]['info'] = $row['info'];
        }
        $i++;
    }
}

```



```

        return $products;
    }
}

public static function getProductById($id)
{
    $id = intval($id);

    if ($id) {
        $db = Db::getConnection();

        $result = $db->query('SELECT * FROM product WHERE id=' . $id);
        $result->setFetchMode(PDO::FETCH_ASSOC); /*setFetchMode — Задает
режим выборки по умолчанию для объекта запроса , FETCH_ASSOC извлекает
ассоциативный массив*/

        return $result->fetch();
    }
}

/**
 * Возвращает список товаров
 * @return array <p>Массив с товарами</p>
 */
public static function getProductsList()
{
    // Соединение с БД

    $db = Db::getConnection();

```

```

// Получение и возврат результатов
$result = $db->query('SELECT id, name, prise FROM product ORDER BY id
ASC');
$productsList = array();
$i = 0;
while ($row = $result->fetch()) {
    $productsList[$i]['id'] = $row['id'];
    $productsList[$i]['name'] = $row['name'];
    $productsList[$i]['prise'] = $row['prise'];
    $i++;
}
return $productsList;
}

```

```

    public static function getTotalProductsInCategory($categoryId)
    {
        $db = Db::getConnection();

        $result = $db->query('SELECT count(id) AS count FROM product WHERE
category_id = "'.$categoryId.'"');/*подсчитает количество товаров*/
        $result->setFetchMode(PDO::FETCH_ASSOC);
        $row = $result->fetch();

        return $row['count'];
    }
}

```

```
/**
```

```
* Удаляет товар с указанным id
```

```
* @param integer $id <p>id товара</p>
```

```

* @return boolean <p>Результат выполнения метода</p>
*/
public static function deleteProductById($id)
{
    // Соединение с БД
    $db = Db::getConnection();

    // Текст запроса к БД
    $sql = 'DELETE FROM product WHERE id = :id';

    // Получение и возврат результатов. Используется подготовленный запрос
    $result = $db->prepare($sql);
    $result->bindParam(':id', $id, PDO::PARAM_INT);
    return $result->execute();
}

/**
 * Добавляет новый товар
 * @param array $options <p>Массив с информацией о товаре</p>
 * @return integer <p>id добавленной в таблицу записи</p>
 */
public static function createProduct($options)
{
    // Соединение с БД
    $db = Db::getConnection();

    // Текст запроса к БД
    $sql = 'INSERT INTO product '
        . '(name, prise, category_id, brand, Mileage,'
        . 'body, info, engine, drive_unit, color, fuel_consumption, address,
phone_number)';

```

```

        . 'VALUES '
        . '(:name, :prise, :category_id, :brand, :Mileage, :body, '
        . ':info, :engine, :drive_unit, :color, :fuel_consumption, :address,
:phone_number)';

// Получение и возврат результатов. Используется подготовленный запрос
$result = $db->prepare($sql);
$result->bindParam(':name', $options['name'], PDO::PARAM_STR);
$result->bindParam(':prise', $options['prise'], PDO::PARAM_STR);
$result->bindParam(':category_id', $options['category_id'], PDO::PARAM_STR);
$result->bindParam(':brand', $options['brand'], PDO::PARAM_INT);
$result->bindParam(':Mileage', $options['Mileage'], PDO::PARAM_STR);
$result->bindParam(':body', $options['body'], PDO::PARAM_INT);
$result->bindParam(':info', $options['info'], PDO::PARAM_STR);
$result->bindParam(':engine', $options['engine'], PDO::PARAM_INT);
$result->bindParam(':drive_unit', $options['drive_unit'], PDO::PARAM_INT);
$result->bindParam(':color', $options['color'], PDO::PARAM_INT);
        $result->bindParam(':fuel_consumption',
$options['fuel_consumption'], PDO::PARAM_INT);
        $result->bindParam(':address', $options['address'],
PDO::PARAM_INT);
        $result->bindParam(':phone_number',
$options['phone_number'], PDO::PARAM_INT);
    if ($result->execute()) {

        // Если запрос выполнен успешно, возвращаем id добавленной записи
        return $db->lastInsertId();
    }
    // Иначе возвращаем 0
    return 0;

```

```
}

    /**
 * Редактирует товар с заданным id
 */
public static function updateProductById($id, $options)
{
    // Соединение с БД
    $db = Db::getConnection();

    // Текст запроса к БД
    $sql = "UPDATE product
        SET
            name = :name,
            prise = :prise,
            category_id = :category_id,
            brand = :brand,
            Mileage = :Mileage,
            body = :body,
            info = :info,
            engine = :engine,
            drive_unit = :drive_unit,
            color = :color,

            fuel_consumption = :fuel_consumption,
            address = :address,
            phone_number = :phone_number
        WHERE id = :id";

    // Получение и возврат результатов. Используется подготовленный запрос
```

```

$result = $db->prepare($sql);
$result->bindParam(':id', $id, PDO::PARAM_INT);
$result->bindParam(':name', $options['name'], PDO::PARAM_STR);
$result->bindParam(':prise', $options['prise'], PDO::PARAM_STR);
$result->bindParam(':category_id', $options['category_id'], PDO::PARAM_STR);
$result->bindParam(':brand', $options['brand'], PDO::PARAM_INT);
$result->bindParam(':Mileage', $options['Mileage'], PDO::PARAM_STR);
$result->bindParam(':body', $options['body'], PDO::PARAM_INT);
$result->bindParam(':info', $options['info'], PDO::PARAM_STR);
$result->bindParam(':engine', $options['engine'], PDO::PARAM_INT);
$result->bindParam(':drive_unit', $options['drive_unit'], PDO::PARAM_INT);
$result->bindParam(':color', $options['color'], PDO::PARAM_INT);
                $result->bindParam(':fuel_consumption',
Options['fuel_consumption'], PDO::PARAM_INT);
                $result->bindParam(':address', $options['address'],
PDO::PARAM_INT);
                $result->bindParam(':phone_number',
Options['phone_number'], PDO::PARAM_INT);
    return $result->execute();
}

/**
 * Возвращает путь к изображению
 *
 * @param integer $id
 * @return string <p>Путь к изображению</p>
 */
public static function getImage($id)
{
    // Название изображения-пустышки

```

```
$noImage = 'no-image.png';

// Путь к папке с товарами
$path = '/upload/images/products/';

// Путь к изображению товара
$pathToProductImage = $path . $id . '.jpg';

if (file_exists($_SERVER['DOCUMENT_ROOT'].$pathToProductImage)) {
    // Если изображение для товара существует
    // Возвращаем путь изображения товара
    return $pathToProductImage;
}

// Возвращаем путь изображения-пустышки
return $path . $noImage;
}
}
```

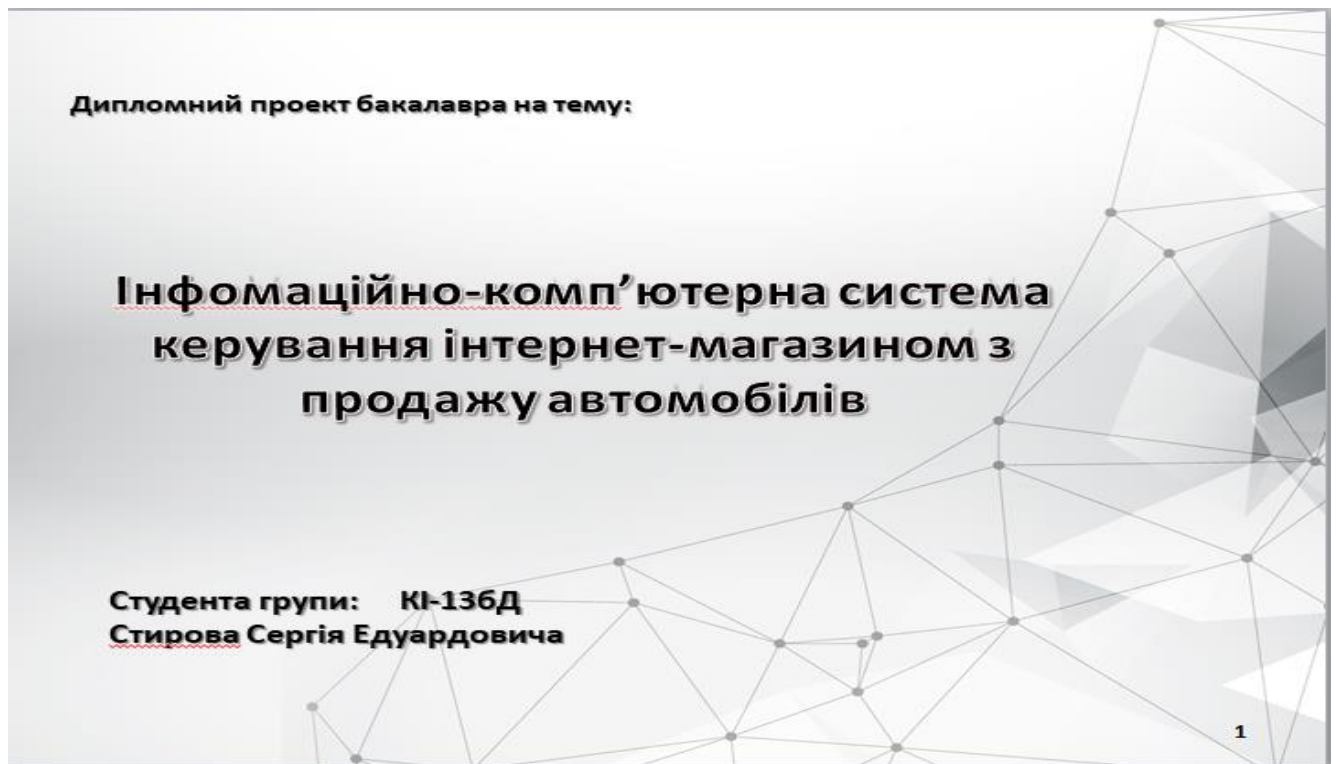


Рисунок Б.1 – Титульний лист

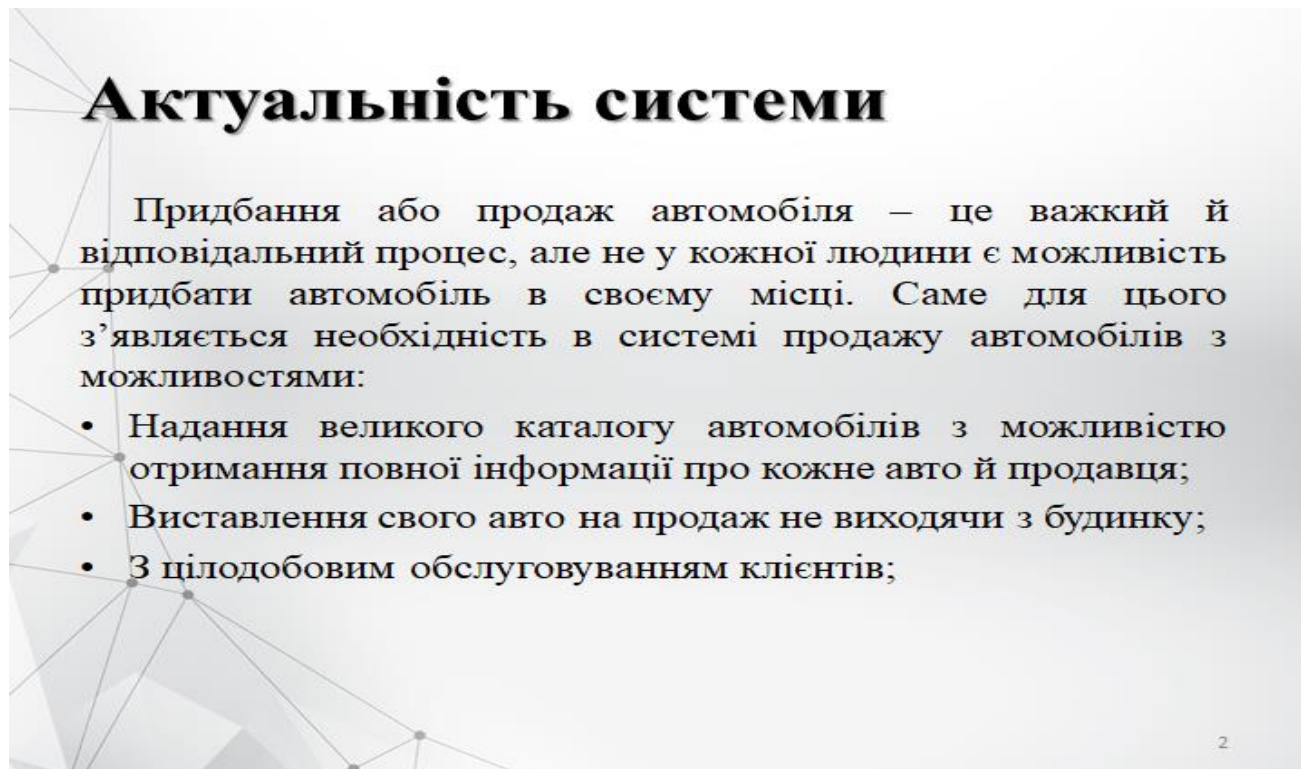


Рисунок Б.2 – Актуальність системи

Мета проекту

- Розробка адаптивного інтерфейсу системи під всі розміри екранів з використанням технологій HTML, CSS, JavaScript;
- Розробка адміністративної частини за допомогою технологій PHP, Ажах та з використанням баз даних MySQL;
- З'єднання інтерфейсу системи й серверної частини за допомогою моделі реалізації додатків (MVC).

3

Рисунок Б.3 – Мета проекту

Технології

Для розробки частини користувачів використовуються:

- HTML - це мова розмітки документів у мережі Інтернет.
- CSS – це мова опису зовнішнього виду текстових документів написаних на мові розмітки HTML.
- JavaScript - це мова програмування, програма на якій інтерпретується.

Для серверної частини використовується:

- PHP – це мова з відкритим кодом, яка підтримує об'єктно-орієнтоване та процедурне програмування, використовується для написання серверних веб-додатків.

4

Рисунок Б.4 - Технології

Model-View-Controller

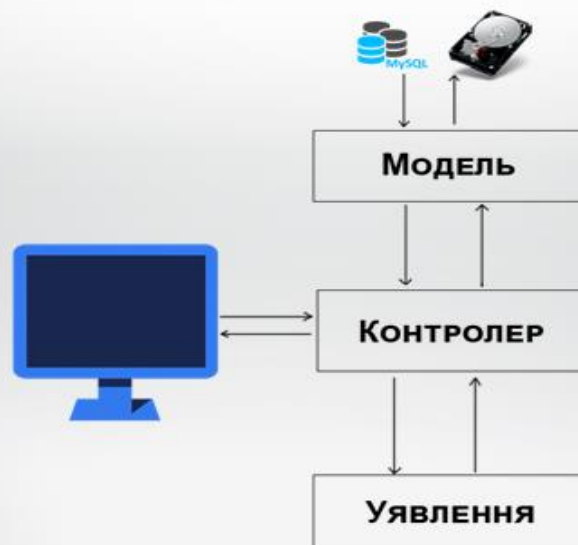
MVC (Model, View, Controller) – це схема поділення інтерфейсу й логіки додатку на окремі компоненти.

- Model – це компонент, який реагує на команди контролера й змінює свій стан.
- View – відповідає за відображення даних моделі. Тобто отримує дані від моделі і відображає їх.
- Controller – забезпечує користувача зв'язком з системою; контролює дані, які йдуть від користувача до системи й навпаки.

4

Рисунок Б.5 – MVC модель

Схема моделі MVC



5

Рисунок Б.6 – Схема MVC

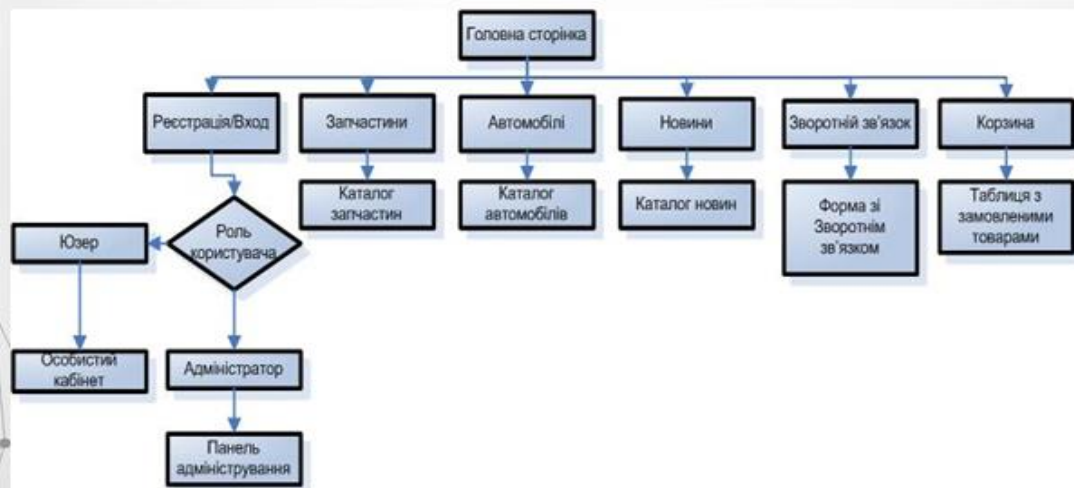
Функціонал системи

- Головна сторінка з короткою інформацією про компанію;
- Реєстрація та ідентифікація в системі;
- Сторінка зі списком автомобілів та інформацією про кожне авто;
- Сторінка зі списком запчастин;
- Корзина з можливістю замовити товар;
- Панель адміністрування, для зручного управління системою;

6

Рисунок Б.7 – Функціонал системи

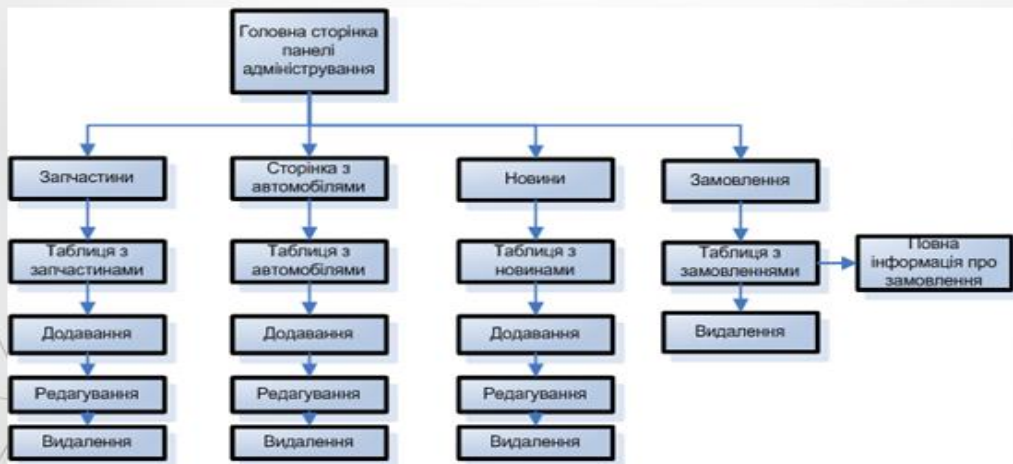
Структура системи



7

Рисунок Б.8 – Структура системи

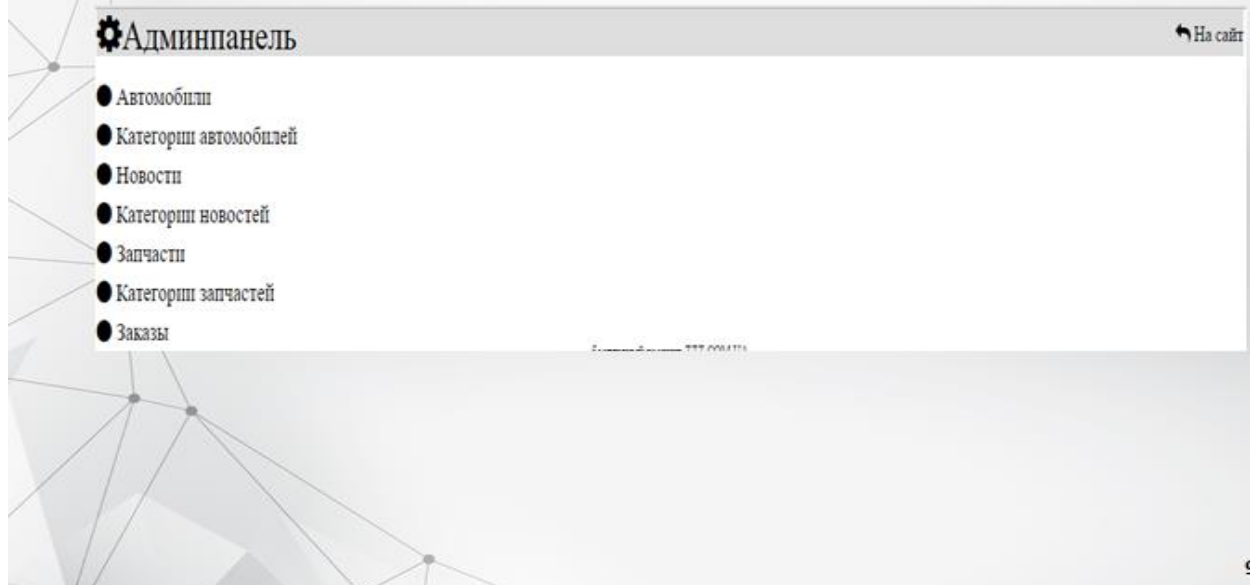
Структура панелі адміністрування



8

Рисунок Б.9 – Структура панелі адміністрування

Панель адміністрування



9

Рисунок Б.10 – Головна сторінка панелі адміністрування

Управління списком автомобілів

Адмінпанель На сайт

Адмінпанель Управління автомобілями
Список товарів
+ Додати товар

ID товара	Назва товара	Ціна	Редагувати	Видалити
1	ВАЗ 2109 1.6 8V 2010	3700	✓	✗
2	ВАЗ 2111 2011	4300	✓	✗
3	ВАЗ 2109 2009	3450	✓	✗
4	ВАЗ 2101 1976	1300	✓	✗
5	ВАЗ 2110 1.6 16v ORIGINAL 2007	3500	✓	✗
6	ВАЗ 2107 2004	2000	✓	✗
7	ВАЗ 2170 MAX-FULL 2009	5000	✓	✗
8	ВАЗ 21099 2008	3500	✓	✗
9	ВАЗ 2121 2121-40 2017	4000	✓	✗
10	Alfa Romeo 159 2008	9800	✓	✗
11	Alfa Romeo 156 1999	4200	✓	✗

10

Рисунок Б.11 – Таблиця управління автомобілями в системі

Додавання нових автомобілів

Адмінпанель На сайт

Адмінпанель Управління автомобілями Редагувати товар
Додати новий товар

Назва товару

Вартість, \$

Категорія

Виробник

Пробіг

Кузов

Інформація

11

Рисунок Б.12 – Сторінка додавання автомобілів в систему



Рисунок Б.13 – Сторінка з автомобілями

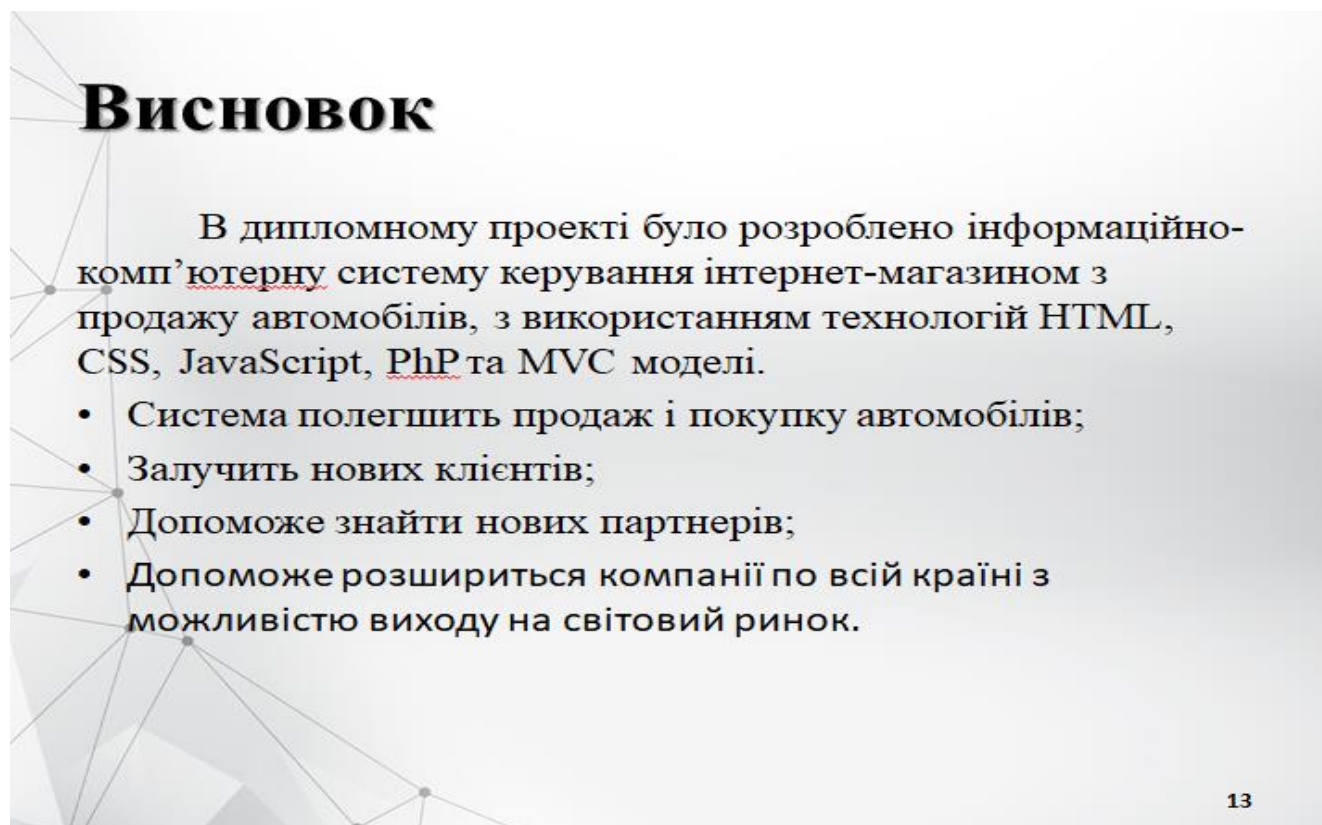


Рисунок Б.14 - Висновок