

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ДЕРЖАВНИЙ ЗАКЛАД „ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА”**

Навчально - науковий інститут математики та інформаційних технологій

Кафедра математики та інформатики

**Шевчук Антон Олегович**

**РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ УПРАВЛІННЯ ПІДПРИЄМСТВОМ  
В СФЕРІ ЕЛЕКТРОННОЇ ТОРГІВЛІ**

кваліфікаційна робота  
здобувача вищої освіти першого (бакалаврського) рівня  
освітньої програми «Комп'ютерні науки та інформаційні технології»  
за спеціальністю 122 „Комп'ютерні науки ”

Особистий підпис

Антон ШЕВЧУК

Науковий керівник

Юрій КОЗУБ, д.т.н.,  
професор

В.о.завідувача кафедри

Ліна БОНДАРЕНКО,  
к.пед.н., доцент

Полтава – 2024

**Міністерство освіти і науки України**

**ДЗ „Луганський національний університет імені Тараса Шевченка”**

Навчально-науковий інститут математики та інформаційних технологій

Кафедра

математики та інформатики

Освітній рівень

Бакалавр

Спеціальність

122 - Комп'ютерні науки

Галузь знань

12 Інформаційні технології

ЗАТВЕРДЖУЮ

В.о.зав. кафедри\_

Ліна БОНДАРЕНКО

(підпис)

(ініціали, прізвище)

“ \_\_\_\_ ” \_\_\_\_\_ 202 \_\_\_\_ р.

**ЗАВДАННЯ  
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ**

Шевчуку Антону Олеговичу

(прізвище, ім'я, по батькові)

1. Тема роботи

Розробка веб-додатку для управління  
підприємством в сфері електронної торгівлі

Керівник кваліфікаційної роботи

Ю.Г. Козуб, д.т.н., проф.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджена наказом по університету

2. Строк подання студентом роботи

03.06.2024р.

3. Вихідні дані до проекту

Провести аналіз методів створення систем он-лайн маркетингу

(визначаються кількісні або (та) якісні показники, яким повинен відповідати об'єкт розробки)

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

Провести теоретичне дослідження систем управління підприємством торгівлі

Провести аналіз методів розробки веб-додатків

Розробити інтернет-магазин

(визначаються назви розділів або (та) перелік питань, які повинні увійти до тексту ПЗ)

## 6. Консультанти розділів роботи

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

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

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи )	Примітка
1.	Вибір теми роботи, вивчення наукової літератури, затвердження теми та керівника.	до 1 лютого	
2.	Аналіз літературних джерел за темою роботи. Розробка та апробація методики дослідно-експериментальної роботи. Подання структури теоретичної частини роботи та плану експериментальних досліджень.	другий тиждень лютого	
3.	Робота над теоретичною частиною. Подання теоретичної частини роботи для першого читання науковим керівником.	до 1 квітня	
4.	Усунення зауважень, урахування рекомендацій наукового керівника. Подання теоретичної частини роботи на друге читання.	до 15 квітня	
5.	Проведення експериментальної роботи. Поетапний аналіз та обговорення її результатів. Перевірка стану виконання роботи.	перший тиждень квітня	
6.	Урахування рекомендацій наукового керівника, усунення недоліків, підготовка варіанта роботи до передзахисту. Розробка презентації.	до 20 травня	
7.	Попередній захист роботи на кафедрі	Травень	
8.	Доопрацювання роботи з урахуванням рекомендацій після передзахисту. Подання роботи науковому керівникові та рецензентові на підготовку відгуку та рецензії	За 10 днів до державної атестації	
9.	Подання на кафедру остаточного варіанта роботи, переплетеного та підписаного автором, науковим керівником і рецензентом.	За 5 днів до державної атестації	

Студент

Керівник роботи

\_\_\_\_\_

підпис

\_\_\_\_\_

підпис

Антон Шевчук

\_\_\_\_\_

(ініціали, прізвище)

Юрій КОЗУБ

\_\_\_\_\_

(ініціали, прізвище)

## АНОТАЦІЯ

Шевчук А.О. Розробка веб-додатку для управління підприємством в сфері електронної торгівлі. Кваліфікаційна робота бакалавра. ЛНУ імені Тараса Шевченка, 2024. 76с.

Кваліфікаційна робота складається із вступу, 3-х розділів та загальних висновків, містить 22 рисунки, додатки, 34 бібліографічне джерело, загальний обсяг – 76 сторінок.

Розглянуто основні потреби створення комерційного веб – сайту, проектування інтернет – магазинів, їх переваги і недоліки. Було вивчено класифікація інтернет - магазинів та мови веб – програмування HTML, JavaScript, PHP. Також було розглянуто методологію проектування інтернет – магазину: його принципи роботи, вимоги до систем управління магазином, функціональну структуру. На основі цього було обране відповідне програмне забезпечення для проектування та реалізації основних функцій продажу.

Клієнтська частина містить систему формування і підтвердження замовлень (кошик покупця), пошуковий сервіс, каталог товару, інформацію про магазин і контакти, форму авторизації клієнтів. Адміністраторська частина містить інструменти управління та роботи з базою даних.

Розроблений інтернет – магазин з продажу автомобілів реалізує основні функції інтернет – магазину. Він простий і зручний у використанні. Модульна система php - скриптів дозволяє швидко покращувати його, з мінімальною втратою часу, і так само застосовувати до нього будь-які дизайнерські рішення.

**Ключові слова:** інформаційна система, система управління, база даних, веб-сторінка, дизайн, інтернет-магазин.

## ABSTRACT

**Shevchuk A.O. Development of a web application for enterprise management in the field of e-commerce.** Bachelor's qualifying work. LNU named after Taras Shevchenko, 2024. 76.

The qualification work consists of an introduction, 3 chapters and general conclusions, contains 22 figures, appendices, 34 bibliographic sources, the total volume is 76 pages.

The main needs of creating a commercial website, designing online stores, their advantages and disadvantages are considered. The classification of online stores and the web programming languages HTML, JavaScript, PHP were studied. The methodology of designing an online store was also considered: its principles of operation, requirements for store management systems, functional structure. Based on this, the appropriate software was selected for the design and implementation of the main sales functions.

The client part contains a system for creating and confirming orders (buyer's basket), a search service, a product catalog, information about the store and contacts, and a customer authorization form. The administrative part contains tools for managing and working with the database.

The developed online car sales store implements the main functions of an online store. It is simple and convenient to use. The modular system of php - scripts allows you to quickly improve it, with minimal loss of time, and also apply any design solutions to it.

**Keywords:** information system, management system, database, web page, design, online store.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	7
ВСТУП .....	8
РОЗДІЛ 1 ТЕОРЕТИЧНЕ ДОСЛІДЖЕННЯ СИСТЕМ УПРАВЛІННЯ ПІРПРИЄМСТВОМ ТОРГІВЛІ .....	11
1.1. Обґрунтування потреби в комерційних веб-сайтах .....	11
1.2. Проектування інтернет - маркетів .....	13
1.3. Переваги та недоліки інтернет - магазинів.....	14
1.4. Мова розмітки гіпертекстових документів HTML .....	16
1.5. Мова програмування PHP .....	18
1.6. Висновки до розділу 1.....	20
РОЗДІЛ 2 МЕТОДОЛОГІЯ ПРОЕКТУВАННЯ ВЕБ-ДОДАТКУ ДЛЯ ЕЛЕКТРОННОЇ ТОРГІВЛІ .....	21
2.1. Принципи роботи інтернет - магазину.....	21
2.2. Вимоги до системи управління інтернет - магазином.....	23
2.3. Базові поняття методології проектування інтернет - магазину.....	24
2.4. Функціональна структура інтернет - магазину .....	27
2.5. Вибір програмного забезпечення для проектування.....	28
2.6. Висновки до розділу 2 .....	31
РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ З ПРОДАЖУ АВТОМОБІЛІВ .....	32
3.1. Загальна структура та етапи розробки сайту.....	32
3.2. Реалізація бази даних магазину .....	33
3.3. Реалізація модулів інтернет-магазину .....	37
3.4. Просування інтернет магазину .....	45
3.5. Висновки до розділу 3.....	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	52
ДОДАТКИ.....	55

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

AIC	- автоматизована інформаційна система;
CMS	- Content Management Site - система керування вмістом;
DHTML	- Dynamic Hyper Text Markup Language ;
GUI	- графічний інтерфейс користувача;
HTML	- Hyper Text Markup Language;
IDE	- інтегроване середовище розробки;
PHP	- Hypertext Preprocessor - препроцесор гіпертексту;
SEO	- Search Engine Optimization - пошукова оптимізація;
WYSIWYG	- властивість редакторів, в яких зміст відображується в процесі редагування і виглядає максимально близько схожим на кінцеву продукцію, яка може бути друкарським документом, веб-сторінкою або презентацією;
XML	- Extensible Markup Language;
АСУ	- автоматизована система управління;
ЕОМ	- електронна обчислювальна машина;
ІКТ	- інформаційно-комукаційні технології;
ІС	- інформаційна система;
ІТ	- інформаційні технології;
ПЗ	- програмне забезпечення;
СКБД	- система керування базами даних.

## ВСТУП

Швидкий розвиток інформаційних технологій, засобів зв'язку і інформатизація суспільства привело до розвитку нової сфери економіки - електронна комерція, потенціал якої активно використовують багато компаній у всьому світі, що дозволяє їм добитися поліпшення якості своєї діяльності за рахунок ефективнішого обміну інформацією. Революційні зміни і швидкі темпи науково-технічного прогресу в області інформаційних технологій, викликані у виробничій і трудовій діяльності, і у сфері споживання привели до появи електронної комерції.

Поява електронної комерції - найбільш революційна зміна у світовій економіці. З появою глобальних комп'ютерних мереж стала можливий інтерактивний обмін інформацією, при мінімальних фінансових витратах, незалежно від відстані і часу. Саме ці якості роблять Інтернет перспективним і зручним для ведення комерції. Електронна комерція через віртуальні магазини в процесі свого розвитку зайняла одну зі лідируючих позицій і стала найважливішим чинником розвитку сучасної економіки.

Інтернет комерція в Україні набуває великої популярності. За останні 10 років кількість інтернет-магазинів виросла приблизно в 120 разів, а оборот інтернет-торгівлі перевищив 1 млрд доларів. За даними Київського міжнародного інституту соціології 36% з опитаних респондентів купували товари і послуги через інтернет.

Електронна торгівля в інтернет-магазині ведеться за тими же принципами, що і традиційна торгівля, але інтернет - магазин має значні переваги перед традиційним магазином.

У зв'язку з розвитком інтернет комерції збільшився попит на створення інтернет-магазинів. За останній час було створено безліч річних програмних продуктів, що дозволяють створювати інтернет магазини без особливих знань програмування, таких як ShopCMS, Shop - Script та ін. Так само створено багато книг і відеокурсів присвячених створенню інтернет магазинів,



найпоширеніших з них це публікації Орлов Л. В., Дари К., Баланеску Ем. та ін., проводяться щорічні конференції, такі як "інтернет-магазини України ". Тому дуже актуальними є дослідження, спрямовані на вирішення проблем розробки інтернет-магазинів засобами мови програмування HTML, DHTML, PHP, XML, JavaScript, MySQL.

**Мета роботи** є розробка Інтернет-магазину і його системи управління.

**Об'єкт дослідження** є процес проектування інтернет – магазину за технологією HTML, DHTML, PHP, XML, JavaScript, MySQL.

**Предмет дослідження** – Інтернет-магазин, який займається продажем автомобілів. На даний момент він є найбільш «молодим» зразком організації інтернет-магазину, тому найбільшу увагу приділяє створенню простого зручного інтерфейсу інтернет-магазину.

Виходячи з поставленої мети, визначено наступні **завдання**:

- вивчення існуючої організації інтернет-магазинів;
- обґрунтування необхідності створення Інтернет-магазину;
- проведення аналізу та вибір Інтернет-технологій для розробки Інтернет-магазину;
- розробка динамічних web-сторінок Інтернет-магазину, керованих базою даних;
- оцінка ефективності виконаної розробки.

Дипломна робота складається зі вступу, чотирьох розділів, загальних висновків, списку використаної літератури та додатків.

У першому розділі висвітлено теоретичні та практичні аспекти системи управління Інтернет-магазином, обґрунтовано потреби у створенні комерційного веб-сайту.

У другому розділі розглянуто принципи роботи і методологію проектування Інтернет-магазину, розроблено структуру сайту.

У третьому розділі описано етапи розробки Інтернет-магазину; розписано усі кроки створення сайту, розглянуто умови просування

створеного Інтернет-магазину у мережі електронної торгівлі. Додатки містять коди модулів розробленого сайту Інтернет магазину.

## **РОЗДІЛ 1 ТЕОРЕТИЧНЕ ДОСЛІДЖЕННЯ СИСТЕМ УПРАВЛІННЯ ПІРПРИЄМСТВОМ ТОРГІВЛІ**

### **1.1. Обґрунтування потреби в комерційних веб-сайтах**

Комерційні веб-сайти є необхідним інструментом для бізнесу з кількох причин.

По-перше, вони є потужним засобом для привертання нових клієнтів та розширення аудиторії. Наявність веб-сайту дозволяє компанії бути доступною 24/7 для потенційних клієнтів з усього світу, надаючи їм можливість ознайомитися з продуктами або послугами компанії у будь-який зручний для них час. Це значно збільшує можливості для залучення нових покупців і збільшення продажів.

По-друге, комерційні веб-сайти є важливим каналом для рекламної діяльності. Вони дозволяють бізнесу просувати свої продукти або послуги через інтернет-маркетингові кампанії, такі як контекстна реклама, SEO-оптимізація і соціальні медіа стратегії. Це дає можливість збільшити усвідомленість бренду та залучити цільову аудиторію.

По-третє, наявність веб-сайту дозволяє підтримувати і зміцнювати імідж компанії. Через веб-сайт компанія може ефективно комунікувати свої цінності, місію, історію і досягнення, що сприяє побудові довіри серед клієнтів і партнерів.

Загалом, комерційний веб-сайт не лише розширює можливості бізнесу, але й є важливим інструментом для його успішного функціонування в умовах сучасного інтернет-середовища.

На основі технічного завдання при розробці сайту треба врахувати основні вимоги до такого продукту. Веб-сайт має включати кілька ключових аспектів, що відповідають сучасним вимогам та очікуванням користувачів. Ось деякі рекомендації щодо проектування і реалізації вашого веб-сайту:

1. Структурна схема сайту: Необхідно передбачити чітко визначену структуру сайту з урахуванням розділів, контенту та навігації. Це важливо для зручності користувачів під час навігації по сайту і для забезпечення логічного розташування інформації.

2. Веб-дизайн: Розробка графічних елементів макету сайту повинна відповідати сучасним дизайнерським тенденціям. Доречним є використання анімації і графічних вставок, які підсилюють емоційно-ціннісний аспект контенту. Важливо також забезпечити, щоб дизайн був адаптивним (responsive), тобто добре виглядав як на десктопах, так і на мобільних пристроях.

3. Управління контентом: Рекомендується використовувати гнучку систему управління контентом (CMS), яка дозволить адміністраторам зручно оновлювати і редагувати вміст сайту без спеціалізованих знань програмування.

4. Друк сторінок: Необхідно створити можливість для користувачів друкувати будь-яку сторінку веб-сайту. Це може бути реалізовано через відповідний інтерфейс на сторінці або через спеціальні стилі для друку.

5. Тестування і оптимізація: Перед розміщенням сайту в мережі Інтернет необхідно провести тестування, щоб переконатися, що він працює коректно на різних платформах і браузерах. Також важливо оптимізувати сайт для швидкої завантаженості, що позитивно впливає на користувацький досвід і SEO.

6. Безпека: Для захисту веб-сайту зазвичай використовуються SSL-сертифікати для захисту персональних даних користувачів і забезпечення безпечних транзакцій, якщо це необхідно для веб-магазину.

Загалом, кожен етап проектування і розробки веб-сайту, від затвердження технічного завдання до розміщення в мережі, має бути уважно продуманим і забезпечити відповідність потребам вашої цільової аудиторії [8].

## **1.2. Проектування інтернет - маркетів**

Проектування і розробка інтернет - магазину передбачає створення бази даних, ефективних інструментів управління контентом, зручного інтерфейсу.

Для створення ефективної та функціональної титульної сторінки веб-сайту, яка задовольняє потреби користувачів і адміністраторів, необхідно слід врахувати, що титульна сторінка має містити наступні обов'язкові елементи.

Логотип веб-сайту. Розміщення логотипу, який є візуальним ідентифікатором бренду, має бути добре видимим і посылати на основну тематику сайту.

Основне меню сайту. Сторінка має містити навігаційне меню, яке містить посылання на основні розділи і сторінки сайту. Це дозволить користувачам легко переміщатися по сайту і знаходити необхідну інформацію.

Форма аутентифікації. Необхідно створити форму для входу зареєстрованих користувачів, яка може включати поля для введення логіну і пароля, а також кнопку входу.

Реєстраційне посылання. Для нових користувачів необхідно додати посылання на сторінку реєстрації. Це може бути кнопка або посылання, яке направляє користувача на сторінку реєстрації або форму для створення облікового запису.

Особливу увагу необхідно приділити розробці адміністративної частини.

Адміністративна частина повинна містити інформацію про нових користувачів. Адміністраторам має бути дотупною інформація про нових користувачів, які зареєструвалися на сайті. Це може включати основні дані про користувачів і їх дії.

Розділ налаштувань сайту має бути доступним для адміністраторів, що дозволить їм змінювати параметри сайту, такі як загальні налаштування, налаштування безпеки і інші параметри.

За необхідності забезпечуються можливості для адміністрування товарів, замовлень, платежів і доставки.

Обов'язковою вимогою є забезпечення можливості адміністраторам керувати обліковими записами користувачів, включаючи редагування, блокування або видалення облікових записів за необхідності.

Для управління контентом зазвичай використовують CMS. Необхідно обрати таку систему, яка найкраще підходить для конкретної задачі і має необхідний набір функцій.

### **1.3. Переваги та недоліки інтернет - магазинів**

Переваги інтернет-магазину перед звичайним магазином дуже очевидні:

- збільшення обсягу продажів: при створенні додаткової торговельної точки обсяг проданих товарів або послуг збільшується;
- розширення ринок збуту: сьогодні мережею Інтернет користуються величезна кількість людей, які можуть, познайомляться з компанією, продуктами чи послугами;
- зменшення виробничих витрат: витрати на просування і торгівлю товарами або послугами через інтернет-магазин менше, ніж через звичайний, тому виробничі витрати тут знижуються;
- торгувати з іншими країнами: можливість реалізовувати свою продукцію по всьому світу;
- працювати в будь-якій точці Землі: так як ваш магазин - електронний та всі операції здійснюються в режимі онлайн, ви зможете працювати в будь-якій точці планети;
- цілодобова робота інтернет магазину;
- можливість мати декілька не пов'язаних між собою проектів;
- рівна конкуренція великих і маленьких компаній: так як всі електронні магазини будуються за загальним принципом, маленька компанія

або навіть приватний підприємець може створити магазин, який буде виглядати і працювати так само, як великі магазини;

- поліпшення зв'язків з громадськістю: магазин постійно відкритий для широкої публіки, тому в ньому можна виставляти останні прес-релізи, повідомляти про розробку нових продуктів, проводити опитування та інші заходи, що принесе вашій фірмі популярність і зміцнить її положення в галузі;

- забезпечення єдиного банку даних для свого персоналу: для відсутніх в офісі співробітників (вони можуть знаходитися на семінарі або бути у клієнта) електронний магазин стане зручним засобом отримання інформації, навіть якщо офіс закритий [11].

Інтернет-магазин має і свої недоліки, насамперед це атаки хакерів. Це обумовлено низькою захищеністю. Наслідок атак можуть бути різними: від перебоїв роботи програми і до втрати бази даних клієнтських банківських карт.

Найпоширеніші види атак:

- SQL-ін'єкцій - введення в веб-форму спеціальних команд, з допомогою яких хакер змушує сервер виконувати потрібні йому дії;
- виклик виняткових ситуацій - штучний виклик помилок у роботі програми.
- підміна ціни та інших даних;
- атаки на процес аутентифікації і міжсайтового скриптинг - отримання доступу до даних та пароллю користувача.

Виникнення збоїв з-за наявності «багів» - помилок у програмі, допущені розробниками. До мінусів відноситься довга доставка (особливо міжнародна), не можна побачити реальний товар.

Також виникає необхідність мати досить гарне програмне і апаратне забезпечення і персонал, що обслуговує його. Робочі місця персоналу повинні бути обладнані відповідно. Сервер компанії повинен бути досить

продуктивним, щоб забезпечувати роботу інтернет-магазину. Додаткові витрати при використанні послуг хостингу.

#### **1.4. Мова розмітки гіпертекстових документів HTML**

HTML є основою для створення інтерактивних сторінок у WWW, де кожен елемент (текст, зображення, посилання тощо) може бути зв'язаним з іншими документами або ресурсами через гіпертекстові посилання [16].

Текст вашого запиту виглядає так, ніби він описує основні концепції HTML і його використання. Щоб зрозуміти HTML і його структуру, ось деякі основні поняття:

1. HTML (Hyper Text Markup Language): Це мова розмітки, яка використовується для створення веб-сторінок. Вона включає в себе різні теги, які визначають структуру і вигляд вмісту.

2. Структура HTML-документа:

- Документ завжди починається з тегу `<html>`, який вказує, що це HTML-документ.

- Заголовок документа задається тегами `<head>`, в якому зазвичай міститься інформація, яка не відображається безпосередньо на сторінці (наприклад, `<title>` для назви сторінки).

Основна частина документа, яка відображається у браузері, знаходиться в тегах `<body>`.

3. Теги для форматування:

`<p>`: Визначає абзац тексту. Може мати параметри, такі як `align`, для вирівнювання тексту (`align="right"`).

`<br>`: Вставляє перенесення рядка всередині поточного абзацу.

`<u>`: Виділяє текст підкресленням.

4. Посилання на інші документи:



Для встановлення посилань використовується тег `<a>`. Атрибут `href` вказує URL, на який веде посилання. Наприклад: `<a href="https://www.example.com">Посилання на інший документ</a>`.

#### 5. Пробільні символи:

- Пробіли, табуляції і переклади рядків враховуються у HTML, але кількість підрядних пробільних символів зменшується до одного пробілу під час відображення.

HTML є основою для створення веб-сторінок і дозволяє визначати структуру та вигляд контенту, який відображається у браузері.

Динамічний HTML (DHTML) використовується для позначення HTML-сторінок з динамічно змінюваним вмістом, що може змінюватися без перезавантаження сторінки. Реалізація DHTML ґрунтується на трьох основних складових: безпосередньо HTML, каскадні таблиці стилів (CSS) і мови сценаріїв, таких як JavaScript.

Основою для реалізації динамічного змінювання вмісту є об'єктна модель документа (DOM). DOM є інтерфейсом програмування, який зв'язує HTML-документ зі скриптами, дозволяючи їм отримувати доступ до кожного елемента сторінки як окремого об'єкта. Це означає, що скрипти можуть динамічно змінювати властивості і зміст будь-якого елемента на сторінці в залежності від дій користувача або інших умов.

Наприклад, за допомогою DOM і JavaScript можна змінювати текст, кольори, розміщення елементів, анімувати їх переміщення чи зміну розміру, реагувати на події миші або клавіатури, обмінюватися даними з сервером без перезавантаження сторінки тощо.

Щодо ваших зазначень про стандарти передачі даних по Інтернету, справді, HTML зорієнтований на відображення контенту веб-сторінок, а не на передачу даних або їх управління. Для цих цілей використовуються інші

технології, такі як XML, JSON, а також веб-сервіси і AJAX для взаємодії з сервером і обміну даними між веб-сторінками та додатками.

Отже, DHTML разом з DOM і сценаріями використовується для створення більш інтерактивних і динамічних веб-додатків, які можуть змінювати свій вміст та вигляд у відповідь на дії користувача, що значно підвищує користувацький досвід [18].

### **1.5. Мова програмування PHP**

Історія PHP починається осінню 1994 року. Так, історія про Расмуса Лердорфа і початки PHP дійсно має такий аспект. Коли Лердорф створював ранні версії PHP, його первісна мета полягала в тому, щоб мати зручний спосіб для керування своїм особистим веб-сайтом та відстеження того, хто його відвідує.

Спочатку PHP (Personal Home Page Tools) було створено як набір Perl скриптів для управління його власним веб-сайтом і збору статистики відвідувань. Він використовував CGI скрипт на Perl для збору інформації про відвідувачів, який вставлявся у HTML сторінки його резюме. Це дозволяло йому відслідковувати, хто і коли відвідував його сайт.

Такий підхід також міг зацікавити потенційних роботодавців, показуючи їм не лише резюме, а й технічні здібності Лердорфа у веб-розробці та аналізі даних про відвідувачів [6].

В мову PHP вбудований ряд сервісів, які полегшують роботу по створенню веб-додатків:

- автоматичний витяг POST і GET-параметрів;
- взаємодію з великою кількістю СКБД.
- обробка файлів, що завантажуються на сервер;
- робота з cookies.

Використання мови PHP дає ряд переваг, насамперед це практичність і простота використання, багатифункціональність, кросплатформенність

(робота різними апаратними платформами і операційними системами), ефективність обробки сценаріїв.

Ще однією перевагою мови PHP є взаємодія з багатьма веб-технологіями: HTML, DHTML, JavaScript, MySQL, та іншими, що дозволяє ефективно поєднувати їх у створенні програми.

Таким чином, виходячи з вищесказаного, скриптова мова PHP є оптимальним вибором.

Спільно з мовою програмування PHP були використані мови розмітки сторінок HTML 4 і XML 1.0.

XML - мова розмітки, що дозволяє стандартизувати вид файлів-даних, використовуваних комп'ютерними програмами, у вигляді тексту, зрозумілого людині [3].

Більшість веб-додатків для зберігання і роботи з інформацією використовують бази даних, що дозволяє збільшити ефективність роботи і полегшити процес створення додатку. У базах даних можна зберігати будь-яку структуровану інформацію: тексти, зображення, посилання, скрипти.

Використання баз даних в першу чергу забезпечує безпеку зберігання інформації, збільшення продуктивності роботи і цілісність даних.

Тому для зручності роботи з даними в своїй роботі я використовувала для зберігання інформації базу даних.

У процесі створення інтернет-магазину мною була використана технологія cookies.

Cookies - невеликий фрагмент даних, створений веб-сервером або веб-сторінкою та зберігається на комп'ютері користувача у вигляді файлу, який веб-клієнт (звичайно веб-браузер) кожного разу пересилає веб-серверу в HTTP-запиті при спробі відкрити сторінку відповідного сайту.

Застосовується для збереження даних на стороні користувача, на практиці зазвичай використовується для:

- аутентифікації користувача;

- зберігання персональних переваг і налаштувань користувача;
- відстеження стану сесії доступу користувача;
- ведення статистики про користувачів [3].

Використання технології cookies дозволив мені полегшити процес аутентифікації користувача як для самого користувача, так і для азіроботчіка. Детальніше технологія процесу аутентифікації користувача описана у Вікіпедії:

Користувач вводить ім'я користувача і пароль у текстових полях сторінки входу і відправляє їх на сервер.

1. Сервер отримує ім'я користувача і пароль, перевіряє їх і, при їх правильності, відправляє сторінку успішного входу, прикріпивши cookies з якимсь ідентифікатором сесії. Ця cookies може бути дійсна тільки для поточної сесії браузеру, але може бути налаштована і на тривале зберігання.

2. Кожного разу, коли користувач запитує сторінку з сервера, браузер автоматично відправляє cookies з ідентифікатором сесії сервера. Сервер перевіряє ідентифікатор по своїй базі ідентифікаторів і, за наявності в базі такого ідентифікатора, «дізнається» користувача [3].

## **1.6. Висновки до розділу 1**

У першому розділі кваліфікаційної роботи наведений огляд підходів до створення сайтів для комерційної діяльності, а також найбільш ефектиіних інструментів розробки.

Розглянуто основи мови веб - програмування HTML та PHP, як вибір до проектування.

## РОЗДІЛ 2 МЕТОДОЛОГІЯ ПРОЕКТУВАННЯ ВЕБ-ДОДАТКУ ДЛЯ ЕЛЕКТРОННОЇ ТОГРІВЛІ

### 2.1. Принципи роботи інтернет - магазину

Вибравши необхідний товар або послуги, користувач зазвичай має можливість тут же на сайті вибрати метод оплати і доставки. Сукупність відібраних товарів, спосіб оплати і доставки являють собою закінчений замовлення, яке оформляється на сайті шляхом повідомлення мінімально необхідної інформації про покупця. Інформація про покупця може зберігатися в базі даних магазину якщо бізнес-модель магазину розрахована на повторні покупки, або ж відправлятися разово. За даними опитування порталу Mail.ru необхідність реєстрації при замовленні товару або послуги часто заважає користувачам здійснювати покупки в інтернет-магазинах. В інтернет-магазинах, розрахованих на повторні покупки, також ведеться відстеження повернень відвідувача і історія покупок.

Часто при оформленні замовлення передбачається можливість повідомити деякі додаткові побажання від покупця продавцю. Крім того, існують сайти, в яких замовлення приймаються за телефоном, електронною поштою, Jabber або ICQ.

Використовуються такі варіанти способів оплати:

- банківська карта – безготівковий вид розрахунку, який часто піддається нарікань в плані безпеки;
- банківський переказ – оплата за замовлення проводиться банківським платіжним дорученням на розрахунковий рахунок магазину, часто застосовується в B2B-системах. У роздрібних інтернет магазинах банківський переказ здійснюється шляхом друку платіжки форми ПД-4 для оплати в Ощадбанку або інших банках, які приймають платежі населення;
- готівковий розрахунок – товар оплачується кур'єру готівкою при отриманні покупцем товару;

- електронні гроші – безготівковий вид розрахунку; термінали моментальної оплати – оплата проводиться в вуличних платіжних терміналах;
- SMS-платежі – це не платежі типу «відправте sms на короткий номер і отримаєте мелодію». Ні, sms в інтернет-магазинах використовується зовсім по-іншому. Інтернет-магазин формує запит коштів і передає його в одну зі спеціалізованих систем електронних платежів (аналогічно іншим способам), і відправляє до цієї ж системи користувача. Користувач вже у взаємодії з платіжною системою відправляє sms із зазначеними системою реквізитами і підтверджує платіж. Платіжна система після підтвердження відправляє магазину повідомлення про успішність або неуспішність переведення;
- накладений платіж – товар оплачується на пошті при отриманні. Даний спосіб, як правило, обмежений державними кордонами, оскільки відправкою товарів післяплатою займаються в основному державні поштові служби.

Описана схема є загальною, вона працює і для кредитних карт, і для електронних грошей, і для sms-платіжних систем. Часто навіть і для готівкового розрахунку – іноді магазини виділяють кур'єрську службу в окрему підсистему.

Слід звернути увагу на те, що існують способи оплати, підвладні шахрайським операціям і внаслідок цього не використовувані чесними інтернет-магазинами:

Переклад по Western Union – даний вид перекладу застосовується тільки між приватними особами, немає можливості відкликання або опротестування платежу.

«Звичайні» SMS-платежі на короткий номер з незмінною сумою – немає можливості відкликання або опротестування платежу. Маленькі суми, відсутність квитанцій. Через свою простоту часто використовується в різних шахрайських схемах, і зазвичай не застосовується серйозними інтернет-магазинами, які працюють, як говорилося вище, через спеціалізовані платіжні

системи. Вибір того чи іншого способу оплати залежить від безлічі факторів, серед яких можна відзначити статус покупця (фізична або юридична особа), особливості товару (матеріальний або електронний), ціна, способи доставки (неможливо прийняти готівку при відправленні товару за кордон).

Після відправки замовлення з покупцем зв'язується продавець і уточнює місце і час, у який слід доставити замовлення. Доставка здійснюється або власною кур'єрською службою, або компанією, що надає послуги доставки, або поштою – посилкою або бандероллю. Електронні товари, такі як програмне забезпечення або ключі до них, тексти, статті, фотографії, коди доступу та поповнення рахунків, можуть доставлятися електронними каналами – електронною поштою, доступом до файлу по FTP, доступом в захищену область сайту і так далі. Проте в цьому випадку слід бути обережним, оскільки довести неотримання товару електронним способом істотно складніше, ніж у випадку фізичної доставки. Сучасні інтернет-магазини часто вивантажують свої товарні пропозиції в системи порівняння та підбору товарів (зазвичай при великих порталах і пошукових системах), що дозволяє залучити додатковий потік покупців.

## **2.2. Вимоги до системи управління інтернет - магазином**

Інтернет-магазини створюються із застосуванням систем управління контентом сайтів, оснащених необхідними модулями. Великі інтернет-магазини працюють на спеціально для них розроблених або адаптованих типових системах управління. Середні і малі магазини зазвичай використовують типові комерційне та вільне ПО. Наприклад, широко відомий вільний движок osCommerce. Система управління контентом сайту інтернет-магазину може бути коробочним продуктом, самостійно встановлюються на хостинг-майданчик, може бути приватною розробкою веб-студії, їй же обслуговується, або може бути програмним сервісом, наданих з помісячною оплатою або безкоштовно.

Потреби адміністраторів інтернет-магазину в складському, торговому, бухгалтерському та податковому обліку повинні підтримуватися невидимою відвідувачам частиною інтернет-магазину – бек-офісом.

Безліч магазинів сьогодні крім торгових точок мають свої онлайн-представництва. Основний закон, що регулює відносини між покупцем і продавцем – ФЗ "Про захист прав споживачів" від 25.10.2007р.

Відповідно до закону "Про захист прав споживачів" окремою постановою введені "Правила продажу товарів дистанційним способом", які більш детально регламентують вимоги до продавців, провідним продажу через інтернет або по каталогах, і які покликані забезпечують безпеку придбання товару через інтернет-магазин. Зокрема правилами регламентується повнота інформації, наданої про пропонований товар і про сам продавця, а також забезпечуються додаткові права покупця з повернення товару.

### **2.3. Базові поняття методології проектування інтернет - магазину**

Офіс магазину може бути розміщений в одному невеликому приміщенні, де і буде перебувати персонал магазину. Персонал магазину складає 4 людини:

- технік – займається електронною частиною інтернет-магазину. В його обов'язки входить технічне обслуговування, супроводження та оновлення сайту магазину;
- клієнт-менеджер – розглядає і сортує заявки на товар, замовляє потрібні товари на склад, стежить за своєчасним надходженням платежів за товар, віддає кур'єру оплачені товари для їх доставки. Клієнт-менеджер за сумісництвом є адміністратором підприємства і управляє роботою магазину за відсутності директора;
- кур'єр – відповідає за доставку товару, відвозить клієнту товар сам або відправляє його клієнту поштою або службою доставки, залежно від бажання і платоспроможності клієнта;



– директор магазину – стежить за коректністю роботи персоналу, видає заробітну плату, сплачує податки та оренду приміщення, вирішує питання з незадоволеними клієнтами особисто.

Життєвий цикл програмного забезпечення (ЖЦ ПЗ) є ключовою концепцією методології проектування інформаційних систем. Він охоплює весь процес створення, впровадження, експлуатації та вилучення ПЗ з експлуатації. Міжнародний стандарт ISO/IEC 12207 визначає структуру ЖЦ ПЗ, яка включає три основні групи процесів: основні, допоміжні та організаційні.

#### 1 Основні процеси ЖЦ ПЗ:

Придбання: Включає всі дії, пов'язані з закупівлею програмного забезпечення або його компонентів.

Поставка: Охоплює процеси передачі створеного ПЗ замовнику.

Розробка: Містить усі роботи зі створення ПЗ відповідно до вимог, включаючи проектування, програмування, тестування та документування.

Експлуатація: Включає впровадження ПЗ, забезпечення його роботи, конфігурування баз даних і робочих місць користувачів, навчання персоналу та підтримку системи.

Супровід: Охоплює процеси підтримки, оновлення, модифікації та покращення ПЗ.

#### 2 Допоміжні процеси:

Документування: Підготовка та ведення документації, що супроводжує всі етапи ЖЦ ПЗ.

Управління конфігурацією: Організація і контроль змін у ПЗ, забезпечення уніфікованої структури системи.

Забезпечення якості: Включає верифікацію, атестацію, оцінку якості та аудит ПЗ.

Верифікація: Перевірка відповідності поточного стану розробки вимогам.

Атестація та оцінка: Процеси, що забезпечують відповідність ПЗ встановленим стандартам і вимогам.

Рішення проблем: Виявлення та усунення проблем, що виникають під час розробки і експлуатації ПЗ.

### 3 Організаційні процеси:

Управління проектами: Планування, організація, контроль термінів і якості виконання робіт, створення команд розробників.

Створення інфраструктури проекту: Вибір методів та інструментів для реалізації проекту, розробка методів і засобів тестування ПЗ.

Навчання: Підготовка і навчання персоналу, що бере участь у проекті, а також користувачів ПЗ.

#### Основні етапи розробки ПЗ:

1. Аналіз: Визначення вимог до ПЗ, аналіз потреб користувачів і визначення функціональних та нефункціональних вимог.

2. Проектування: Розробка архітектури ПЗ, створення детальних проектів компонентів.

3. Реалізація (Програмування): Написання коду, створення програмних модулів, інтеграція компонентів.

4. Тестування: Виявлення та усунення помилок, перевірка відповідності функціональним вимогам.

#### Етапи експлуатації ПЗ:

1. Впровадження: Інсталяція ПЗ, налаштування баз даних, конфігурація системи.

2. Навчання персоналу: Проведення навчання для користувачів і технічного персоналу.

3. Підтримка: Локалізація та усунення проблем, модифікація ПЗ, підготовка пропозицій щодо вдосконалення системи.

#### 4 Управління конфігурацією:

Управління конфігурацією є критичним процесом для складних проектів, що включають багато компонентів і версій. Воно забезпечує організацію і контроль змін, підтримку уніфікованої структури та розвиток системи, відповідно до проекту стандарту ISO 12207-2.

Ці процеси та принципи, визначені стандартом ISO/IEC 12207, сприяють ефективній організації розробки та підтримки ПЗ, забезпечуючи його високу якість та відповідність вимогам користувачів.

## **2.4. Функціональна структура інтернет - магазину**

Щоб бути зручним і зрозумілим покупцям, і бути ефективним і зручним в обслуговуванні, інтернет-магазин повинен мати чітку функціональну структуру сервісів клієнтської та адміністративної частини.

Розглянувши готові скрипти інтернет-магазинів були виділені основні функціональні сервіси інтернет магазину:

### **1. Сервіси клієнтської частини:**

- Головна сторінка - на головній сторінці мають бути розміщені логотип магазину і його назва, інформація про знижки, нові товари та оновленнях, посилання на сторінки партнерів і їх реклама. На головній сторінці повинна бути реалізована функція пошуку. Основною функцією головної станиці є інформативність і забезпечення зручності навігації.

- Товари - каталог товарів інтернет магазину повинен бути систематизований так, щоб забезпечити більш зручну навігацію користувача по списку товару. Зазвичай в цих цілях використовують деревоподібну структуру каталогу - систему каталогів і підкаталогів.

- Система реєстрації і авторизації - основна необхідність даної системи полягає в тому, щоб створити клієнтську базу, в якій повинна бути зазначена коротка інформація про клієнта, необхідна для оформлення доставки і зв'язку з клієнтом. Система реєстрації і авторизації дозволяє відстежувати адміністрації історію замовлень користувачів, розсилати

новини, а клієнтові дозволяє полегшити процес оформлення замовлення. Кожен зареєстрований клієнт має унікальний логін і пароль.

- Система формування та підтвердження замовлень - реалізація віртуальної кошику покупця, що дозволяє клієнтові змінювати свій вибір, до підтвердження замовлення. Після формування замовлення користувач може вибрати спосіб плати та доставки.

## 2. Сервіси адміністративної частини:

- Система підтримки клієнтів - дана система дозволяє в разі непередбачених обставин підтримувати зв'язок з клієнтом для вирішення цієї ситуації.

- Система правління даними - дана система призначена для редагування списку товарів і категорій, оновлення інформації та ціни товару, створення резервної копії бази даних і відновлення бази даних із копії.

- Система обробки замовлень - дана система обробляє замовлення, відстежує їх проходження, оплату замовлення, визначає вартість доставки замовлення і розраховує податок.

- Система статистики магазину - статистика замовлень і відвідувань клієнтів, статистика продажів переглядів продуктів, аналіз отриманої статистики. Спираючись на отримані статистичні дані можна спрогнозувати подальші тенденції розвитку магазину.

## 2.5. Вибір програмного забезпечення для проектування

Для створення веб-програм потрібні спеціальні програмне забезпечення, які повинні бути встановлені на комп'ютер.

Програмне забезпечення, що виконує роль веб-сервер - є основним.

Веб-сервер - це сервер, що приймає HTTP-запити від клієнтів, зазвичай веб-браузерів, і видає їм HTTP-відповіді, зазвичай разом з HTML-сторінкою, зображенням, файлом, медіа-потоків або іншими даними [3]. Він дозволяє

зберігати, відправляти, отримувати і обробляти інформацію. Зазвичай для тестування погромного продукту використовують локальний сервер.

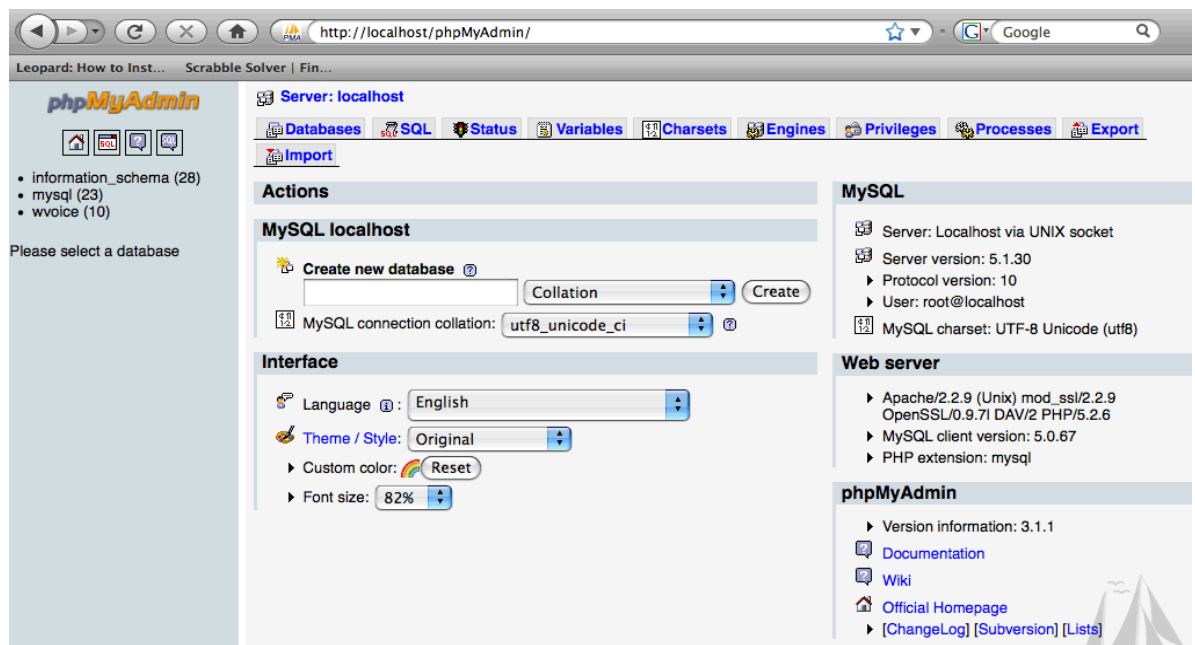
Для створінь веб-додатків, які використовують бази даних, потрібно система управління базами даних (СКБД), яка забезпечує управління створенням і використанням баз даних. Основними функціями, баз даних є: управління даними, підтримка мов баз даних, журналізациа змін, резервне копіювання і відновлення бази даних. Бази даних зберігають у собі структуровану інформацію, дозволяють автоматично оновлювати дані на сайті, дублювати дані, запобігаючи потею інформації.

Одним з поширених інструментом є вебдодаток PhpMyAdmin – вебінтерфейс для адміністрування СКБД MySQL.(рис. 2.1).

#### Можливості phpMyAdmin

- інтуїтивно зрозумілий веб-інтерфейс;
- підтримка більшості функцій MySQL;
- перегляд і видалення баз даних, таблиць, вьюшек, полів і індексів;
- створення, копіювання, видалення, перейменування і зміна баз даних, таблиць, полів і індексів;
- управління сервером, базами даних і таблицями, з порадами по налаштуванню сервера;
- виконання, редакція і збереження будь-якого SQL-вирази, включаючи пакетні запити;
- управління користувачами MySQL та їх привілеями;
- робота з збереженими процедурами і тригерами;
- підтримка імпорту даних з CSV і SQL;
- підтримка експорту в різні формати CSV, SQL, XML, PDF, ISO / IEC 26300 - OpenDocument текст і таблиці, Word, Excel, та інші;
- адміністрування декількох серверів;
- генерування наочних схем баз даних у вигляді PDF;

- створення комплексних запитів за допомогою функції Запит по шаблону;
- глобальний або частковий пошук у базі даних;
- трансформація даних в будь-який формат, використовуючи набір призначених функцій на зразок відображення BLOB-даних у вигляді картинки або посилання для завантажування.



**Рис 2.1.** Інтерфейс додатку PhpMyAdmin

Веб-сервер Apache є одним з найпопулярніших, завдяки гнучкості конфігурацій, надійності, і можливості підключати додаткові модулі.

СКБД MySQL безкоштовне програмне забезпечення з відкритим кодом, використовує для запитів стандартна мова SQL. Є багатокористувацьким СКБД, з чітко регламентованими рівнями доступу. MySQL автоматично підтримується PHP.

Для розробки програмного забезпечення використовують інтегровані середовища розробки, які зазвичай включають в себе: редактор, засоби

автоматизації роботи, компілятор (інтерпретатор), дебаггер призначений для пошуку помилок.

Для роботи мною було обрано редактор коду Visual Studio Code, легковісний редактор коду для кросплатформної розробки веб- і хмарних додатків з вбудованим відладчиком, підсвічуванням синтаксису та засобами рефакторінгу.

Браузер Mozilla Firefox версії 64, безкоштовно поширюваний, багатофункціональний, один з найпопулярніших браузерів у світі використовувався для тестування та в ролі клієнта додатку. Браузер підтримує HTML5 та CSS3, роботу з JavaScript .

## **2.6. Висновки до розділу 2**

У другому розділі магістерської роботи було розглянуто принципи роботи Інтернет-магазину, варіанти способів оплати за товари, а саме - банківська карта, банківський переказ, готівковий розрахунок , електронні гроші, SMS-платежі, накладений платіж.

Обґрунтовано вимоги до системи управління Інтернет-магазином. Описано базові поняття методології проектування інформаційних систем: технік, клієнт-менеджер, кур'єр, директор магазину та поняття життєвого циклу ІС програмного забезпечення.

Представлено функціональну структуру інтернет – магазину та виділено функціональні сервіси. Визначено програмне забезпечення для реалізації поставленої мети.

## РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ З ПРОДАЖУ АВТОМОБІЛІВ

### 3.1. Загальна структура та етапи розробки сайту

Основним завданням кваліфікаційної роботи було розробити сайт інтернет – магазину. Було обрано магазин з продажу автомобілів. При проектуванні сайту проведено аналіз існуючих інтернет-магазинів та визначено, що структура сайту повинна мати наступні веб- сторінки: Головна, Контакти, Авторизація та Регістрація, Категорії автомобілів, Пошук, Кошик. Дизайн сайту повинен відповідати тематиці ресурсу, бути не громіздким в світлих тонах (див. рис. 3.1).

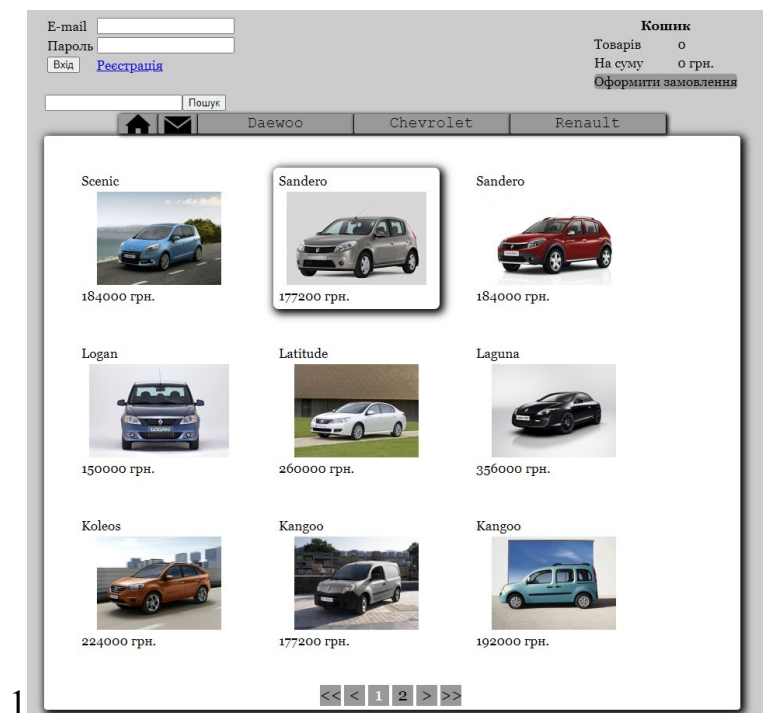


Рис. 3.1. Дизайн сайту Інтернет-магазину з продажу автомобілів

Магазин, що розробляється, повинен мати клієнтську і адміністративну частину.

У клієнтській частині необхідно реалізувати наступне:

- розподіл товару по категоріях, при виборі категорії повинен виводиться список товару того, що належить цій категорії;



- передбачено виведення зображення товару;
  - додавання товару в кошик;
  - кошик покупця - в якій вказується список вибраного товару, загальна сума товару, кількість товару.
  - дані про товар, аутентифікації користувача, категоріях товару - зберігаються у базі даних;
  - при оформленні замовлення користувач вказує ім'я одержувача, адресу одержувача, так само вказана кількість товару і вартість;
  - при підтвердженні замовлення дані про замовлення вирушають на електронну поштову скриньку інтернет-магазину;
  - пошук товару;
- У адміністративній частині повинно бути реалізовано:
- редагування категорій;
  - редагування списку товарів.

### **3.2. Реалізація бази даних магазину**

У створеному застосуванні усі дані про товари і користувачів зберігаються у базі даних, що дозволило понизити об'єм написаного коду, тим самим полегшить процес створення проекту.

База даних складається з 10 таблиць. Кожна з таблиць має первинний ключ - поле таблиці, що однозначно ідентифікують кожен рядок.(рис.3.1).

Таблиця product - таблиця містить назви і індекси автомобілів та їх характеристики: ціну, рік випуску, тип, коробка передач, привід, двері, двигун, колір. (рис.3.3.).

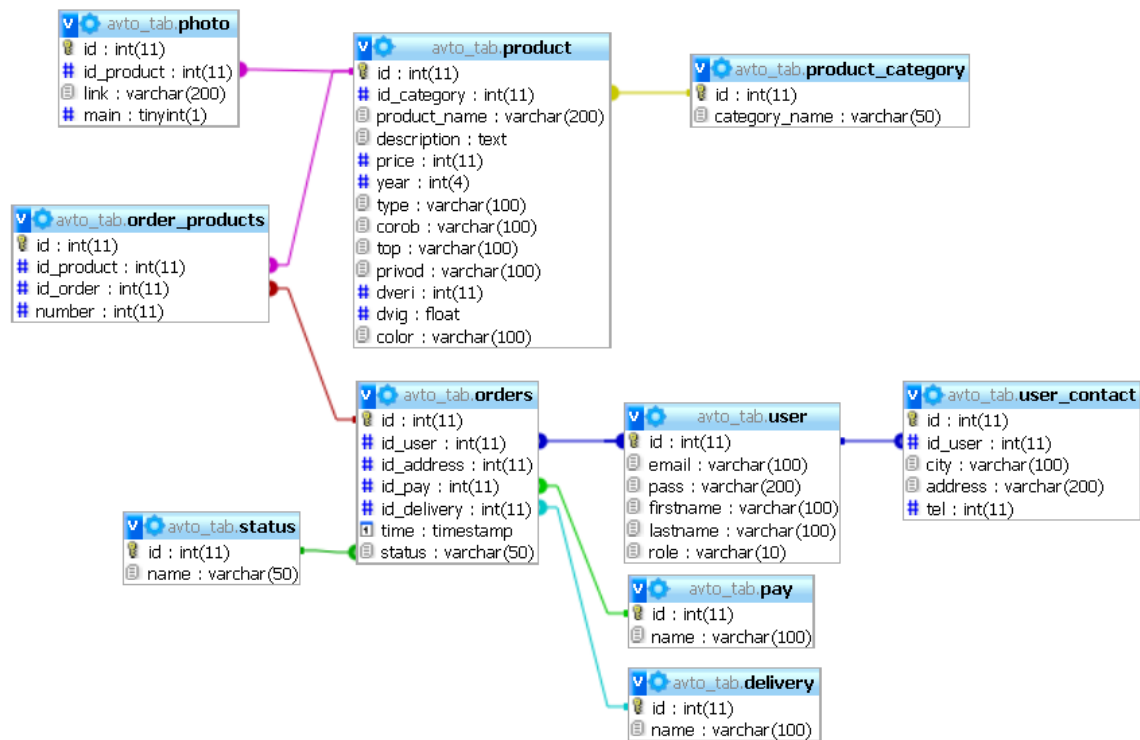


Рис 3.2. Логічна структура бази даних інтернет - магазину

	Переглянути	Структура	SQL	Пошук	Вставити	Експорт	Імпорт	Привілеї	Опел
#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/>	1 id	int(11)			Hi	Немає		AUTO_INCREMENT	
<input type="checkbox"/>	2 id_category	int(11)			Hi	Немає			
<input type="checkbox"/>	3 product_name	varchar(200)	utf8_unicode_ci		Hi	Немає			
<input type="checkbox"/>	4 description	text	utf8_unicode_ci		Hi	Немає			
<input type="checkbox"/>	5 price	int(11)			Hi	Немає			
<input type="checkbox"/>	6 year	int(4)			Hi	Немає			
<input type="checkbox"/>	7 type	varchar(100)	utf8_unicode_ci		Hi	Немає			
<input type="checkbox"/>	8 corob	varchar(100)	utf8_unicode_ci		Hi	Немає			
<input type="checkbox"/>	9 top	varchar(100)	utf8_unicode_ci		Hi	Немає			
<input type="checkbox"/>	10 privod	varchar(100)	utf8_unicode_ci		Hi	Немає			
<input type="checkbox"/>	11 dveri	int(11)			Hi	Немає			
<input type="checkbox"/>	12 dvig	float			Hi	Немає			
<input type="checkbox"/>	13 color	varchar(100)	utf8_unicode_ci		Hi	Немає			
↑ <input type="checkbox"/> Перевірити все Вибрані:									

Рис 3.3. Структура таблиці product

У таблиці product category - зберігається інформація про посилання на категорію, до якої відноситься товар, складається з 2 полів (рис. 3.4.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/> 1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
<input type="checkbox"/> 2	category_name	varchar(50) utf8_unicode_ci			Ні	Немає			

Рис. 3.4. Структура таблиці product category

Фото товарів – посилання на фото розміщені в таблиці photo (рис. 3.5.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/> 1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
<input type="checkbox"/> 2	id_product	int(11)			Ні	Немає			
<input type="checkbox"/> 3	link	varchar(200) utf8_unicode_ci			Ні	Немає			
<input type="checkbox"/> 4	main	tinyint(1)			Ні	Немає			

Рис. 3.5. Структура таблиці photo

Таблиця order\_products містить дані про замовлення - Кошик, при оформленні замовлення дані зберігаються у базі для збору журналу історії та статичних даних, та складається з 4 полів (рис. 3.6.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/> 1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
<input type="checkbox"/> 2	id_product	int(11)			Ні	Немає			
<input type="checkbox"/> 3	id_order	int(11)			Ні	Немає			
<input type="checkbox"/> 4	number	int(11)			Ні	Немає			

Рис. 3.6. Структура таблиці order\_products

Таблиця order містить дані про покупку товару, а саме: id користувача, адресу доставки, спосіб оплати, вид доставки, час та статус замовлення. Містить 7 полів (рис. 3.7.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/> 1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
<input type="checkbox"/> 2	id_user	int(11)			Ні	Немає			
<input type="checkbox"/> 3	id_address	int(11)			Ні	Немає			
<input type="checkbox"/> 4	id_pay	int(11)			Ні	Немає			
<input type="checkbox"/> 5	id_delivery	int(11)			Ні	Немає			
<input type="checkbox"/> 6	time	timestamp		on update CURRENT_TIMESTAMP	Ні	CURRENT_TIMESTAMP		ON UPDATE CURRENT_TIMESTAMP	
<input type="checkbox"/> 7	status	varchar(50) utf8_unicode_ci			Ні	Немає			

☐ Перевірити все    Вибрані:

Рис. 3.7. Структура таблиці orders

Таблиця status містить дані про статус замовлення та має всього 2 поля (рис. 3.8.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/> 1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
<input type="checkbox"/> 2	name	varchar(50) utf8_unicode_ci			Ні	Немає			

☐ Перевірити все    Вибрані:

Рис. 3.8. Структура таблиці status

Таблиця user містить дані користувача: ім'я та прізвище, електронну скриньку та пароль для авторизації, його роль (покупець, адміністратор). Містить 6 полів (рис. 3.9.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/> 1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
<input type="checkbox"/> 2	email	varchar(100) utf8_unicode_ci			Ні	Немає			
<input type="checkbox"/> 3	pass	varchar(200) utf8_unicode_ci			Ні	Немає			
<input type="checkbox"/> 4	firstname	varchar(100) utf8_unicode_ci			Ні	Немає			
<input type="checkbox"/> 5	lastname	varchar(100) utf8_unicode_ci			Ні	Немає			
<input type="checkbox"/> 6	role	varchar(10) utf8_unicode_ci			Ні	Немає			

☐ Перевірити все    Вибрані:

Рис. 3.9. Структура таблиці user

Таблиця user\_contact містить дані користувача для доставки – місто, адресу та телефон. Містить 5 полів (рис. 3.10.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
2	id_user	int(11)			Ні	Немає			
3	city	varchar(100)	utf8_unicode_ci		Ні	Немає			
4	address	varchar(200)	utf8_unicode_ci		Ні	Немає			
5	tel	int(11)			Ні	Немає			

Рис. 3.10. Структура таблиці user\_contact

Таблиці pay та delivery містять інформацію про спосіб оплати товару та доставки товару. Містять по 2 поля (рис. 3.11., рис. 3.12.).

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
2	name	varchar(100)	utf8_unicode_ci		Ні	Немає			

Рис. 3.11. Структура таблиці pay

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
1	id	int(11)			Ні	Немає		AUTO_INCREMENT	
2	name	varchar(100)	utf8_unicode_ci		Ні	Немає			

Рис. 3.12. Структура таблиці delivery

### 3.3. Реалізація модулів інтернет-магазину

В ході реалізації проекту були використані функції, які є загальними для усіх скриптів інтернет-магазину. Ці функції винесені в окремі модулі, які підключаються при запуску додатка, щоб не переписувати постійно код.

Вхід при запуску з локального сервера здійснюється за адресою <http://localhost/avto/index.php>.

Модуль connect.php є загальними для призначеної для користувача і адміністративної частини інтернет-магазину. Він відповідає за підключення до бази даних інтернет – магазину.

```
?php
mysql_connect('localhost', 'root', '');
mysql_select_db('avto_tab');
mysql_query("SET NAMES UTF8");
?>
```

Модуль index.php формує основну сторінку інтернет – магазину: відкриває сесію, підключає header та footer модулі та в залежності від ролі користувача відкриває сторінку адміністрування чи головну сторінку інтернет - магазину.

```
<?php
session_start();
error_reporting('e-error');
include('header.php');
if (($_SESSION['id_user']=='1' or
$_SESSION['id_user']=='11') and !isset($_GET['page']) and
$_GET['page']=='home.php')
{
include('admin.php');
}
else
{
if ($_GET['page'])
{
$page=$_GET['page'];
include($page);
}
else
include('home.php');
}
include('footer.php');
?>
```

Модуль admin.php – сторінка адміністрування. Надає можливість додавання нового товару та редагування статусу замовлення з відстеженням.

У адміністративній частині була реалізована основна функція - функція редагування списку товару, ціни, описи, і зображення товару, а також редагування списку категорій.

Фрагмент лістингу модуля, що відповідає за статус замовлення:

```

while($order=mysql_fetch_row($order_query))
{
    print"<tr><td class='order_list'>$order[0]
$order[1]</td><td class='order_list'>$order[2]</td><td
class='order_list'>";

    print"<select id='status_'. $order[4].'"
onchange='getValues'. "('admin', '', 'update', '',
'status_ $order[4]', '$order[4]')'. ' ">';

    $stat_query=mysql_query("SELECT id, name FROM
status");
    while ($stat=mysql_fetch_row($stat_query))
    {
        print"<option>$stat[1]</option>";
    }
    print"</select></td></tr>";
    $order_detail_query=mysql_query("SELECT
product.product_name, order_products.number,
order_products.id_product
FROM product,
order_products
WHERE
order_products.id_order='$order[4]' AND
order_products.id_product=product.id") or die(mysql_error());

    print"<tr><th>Название</th><th>Количество</th><th>Артикул</
th></tr>";
    while($order_detail=mysql_fetch_row($order_detail_query))
    {
        print"<tr><td>$order_detail[0]</td><td>$order_detail[1]</td>
><td>$order_detail[2]</td></tr>";
    }
    print"<tr><td colspan='3' height='20px'></td></tr>";
    print"
    <script>
    document.getElementById('status_ $order[4]').selectedIndex=$
order[3]-1;
    </script>
    ";
}

```

Повний результат дії модуля представлений на рис. 3.13.

Ласкаво просимо, Name Lastname

Мої замовлення Вихід

Кошик

Товарів 0

На суму 0 грн.

Оформити замовлення

Пошук

DaewooChevroletRenault

Назва

Категорія

Ціна

Рік випуску

Тип кузова

Седан

Коробка передач

Автомат

Паливо

Бензин

Привід

Передній

Кількість дверей

Об'єм двигуна

Цвет

Зображення

Вибрати файл

Файл не вибрано

Вибрати файл

Файл не вибрано

Вибрати файл

Файл не вибрано

Опис

Додати

Список замовлень

Жека Калина

2024-05-01 19:04:15

Доставлений

Назва

Кількість

Артикул

Жека Калина

2024-04-30 00:00:00

Доставлений

Назва

Кількість

Артикул

Nubira

1

8

Рис. 3.13. Сторінка адміністрування

Модуль product.php відповідає за відображення форми додавання нового автомобіля до бази даних інтернет – магазин та його попередній перегляд. Лістинг модуля представлено у Додатку А. На рис. 3.14. показано результат дії модулю.

Ласкаво просимо, Name Lastname

Мої замовлення Вихід

Кошик

Товарів 0

На суму 0 грн.

Оформити замовлення

Пошук

DaewooChevroletRenault

Файл успішно завантажено на сервер


Image resized OK

Файл успішно завантажено на сервер

Image resized OK

Файл успішно завантажено на сервер

Image resized OK



Renault KOLEOS

Рік випуску: 2023

Тип кузова: Позашляховик

Коробка передач: Автомат

Паливо: Бензин

Привід: Повний

Кількість дверей: 4

Двигун: 2.5

Колір: Червоний

700000 грн.

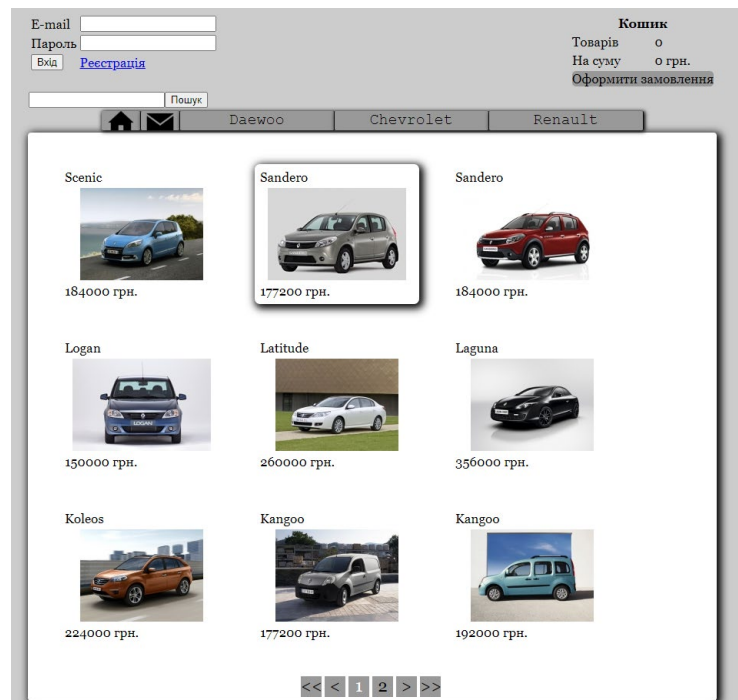
В КОШИК

Рис. 3.14. Результат дії модуля product.php додавання нового автомобіля до бази даних

Модуль home.php виводить на екран головну сторінку інтернет – магазину. На ній розташований головний банер та відображаються останні



добавлені моделі автомобілів. Головна сторінка сайту представлено на рис. 3.15.



**Рис. 3.15.** Головна сторінка інтернет – магазину

Фрагмент лістингу модуля, що відповідає за відображення останніх добавлених шести моделей автомобілів:

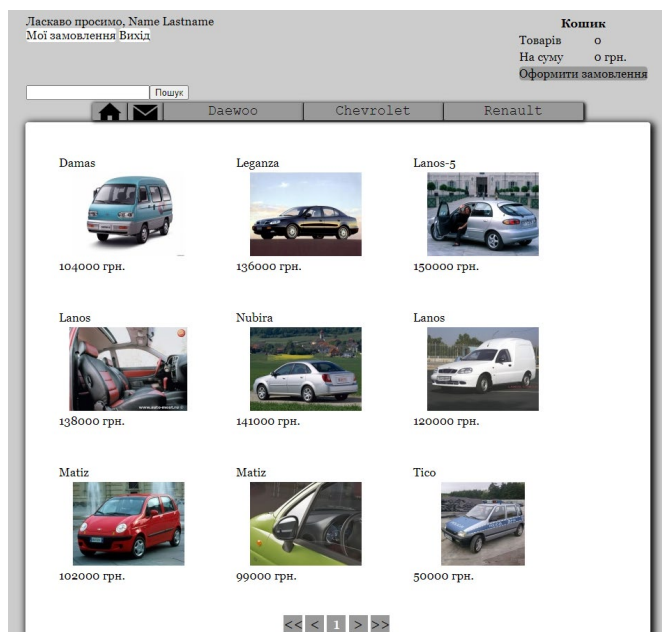
```
?php
print"<table align='center' cellpadding='40px' border=0>";
for($i=0; $i<6; $i+=3)
{
    $product_query=mysql_query("SELECT product.id,
product_name, price, photo.link
FROM product, photo
WHERE main='1' AND photo.id_product=product.id
ORDER BY product.id DESC
LIMIT 3 OFFSET $i") or die(mysql_error());
    print"<tr>";
    while($product=mysql_fetch_row($product_query))
    {
        print"<td class='product'
        valign='top'><table><tr><td>".$product[1]."</td></tr><tr><td
        height='100px'>
        <a href='index.php?page=product.php&id=".$product[0]."'><img
        src='".$product[3]."@s.jpg"></a></td></tr><tr><td>".$product[2].
        " грн.</td></tr></table></td>";
    }
    print"</tr>";
}
```

```
print"</table>"; ?>
```

Модуль `category.php` - містить інформацію про відображення автомобілів певної категорії. Інформація запрошується з бази даних та на екран виводиться увесь перелік автомобілів обраної марки. Фрагмент лістингу модуля, що відповідає за запит до бази даних:


```
$quantity=9;
if(!is_numeric($_GET['pg']))
    $page=1;
if ($_GET['pg']<1)
    $page=1;
$element_count_query=mysql_query("SELECT id
FROM product
WHERE
id_category='".$_GET[id]'" ) or die(mysql_error());
$num=mysql_num_rows($element_count_query);
$pages = $num/$quantity;
$pages = ceil($pages);
$pages++;
if($_GET['pg']>$pages)
    $page = 1;
if(!isset($list))
    $list=0;
$list=--$_GET['pg']*$quantity;
$p=$_GET['pg'];
$limit=++$p*$quantity;
```

Результат дії даного модулю представлено на рис. 3.16.



**Рис. 3.16.** Сторінка інтернет – магазину, що виводить товари за категорією.

Модуль login\_form.html – це форма авторизації користувача (рис. 3.17.).



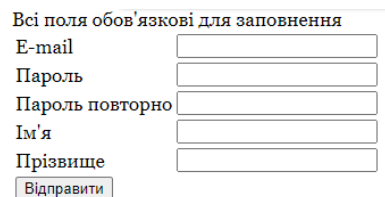
E-mail

Пароль

[Реєстрація](#)

**Рис.3.17.** Форма авторизації користувача

Модуль registration\_form.php – це форма реєстрації користувача (рис. 3.18)



Всі поля обов'язкові для заповнення

E-mail

Пароль

Пароль повторно

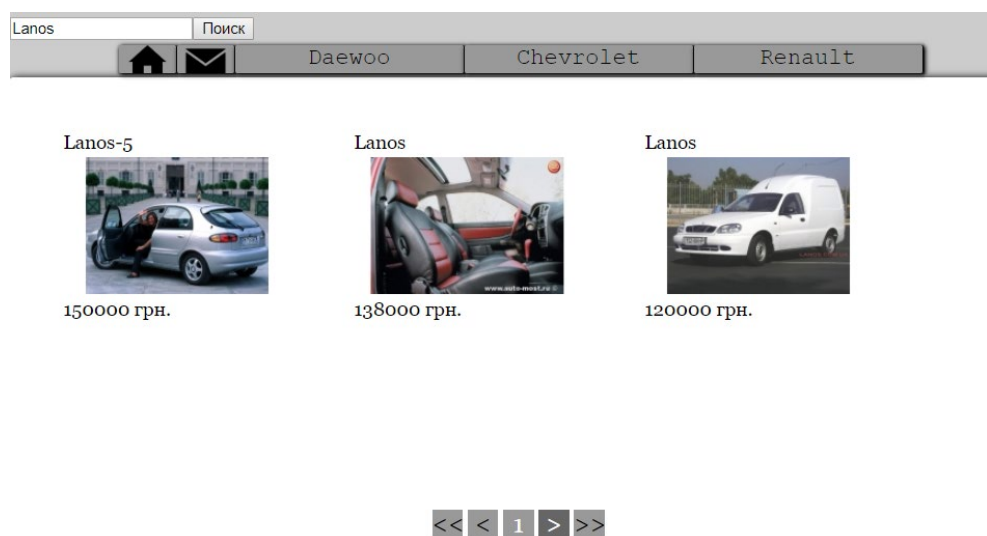
Ім'я

Прізвище

**Рис. 3.18.** Форма реєстрації користувача

Модуль registration.php – модуль адміністратора для перевірки реєстраційних даних та додавання нового користувача до бази даних. Лістинг модуля представлений у Додатку А.

Модуль search.php відповідає за пошук на сайті. Приклад пошуку на сайті представлений на рис. 3.19.



**Рис. 3.19.** Результат пошуку на сайті за запитом «Lanos»

Модуль order.php - сторінка оформлення замовлення. Користувач перевіряє товар, що надійшов до кошика, його ціну та обирає метод доставки та оплати (рис. 3.20.). Якщо користувач обирає доставку Кур'єром чи Службою доставки, то з'являється форма з вводом адреси доставки чи, для зареєстрованого користувача, виводиться остання його адреса з журналу історії бази даних.

Лістинг модуля представлено у Додатку Б.

Найменування	Кількість	Ціна
Leganza	1	136000 грн.
Сумма	Расчитать	136000 грн.

Доставка

☐ Самовивіз

☐ Кур'єром по Полтавській області

☐ Службою доставки по Україні

Оплата

☐ Готівкою

☐ Безготівково

Рис. 3.20. Оформлення замовлення у кошику

Модуль my\_order.php відображає історію покупок зареєстрованого користувача.

Модуль contact.php відображає сторінку контактів власників інтернет-магазину (рис. 3.21.).

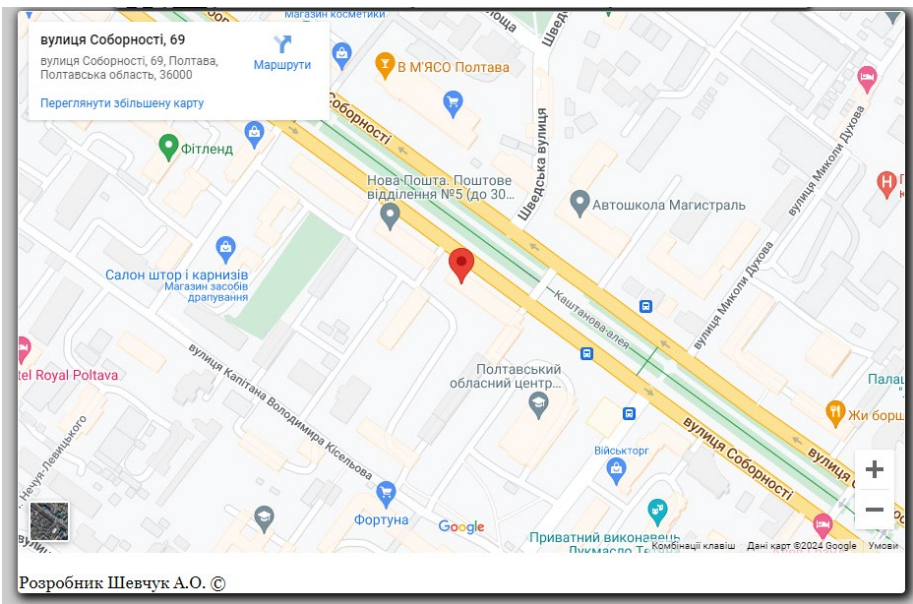


Рис. 3.21. Сторінка контактів

Модуль `img_resize.php` має функцію автоматичного стиснення зображення до вказаного розміру.

Модуль `header.php` – це верхня частина інтернет – магазину: відкриває необхідні теги, підключає `css`, має функцію перерахунку суми кошику, форму авторизації тощо.

Модуль `footer.php` – це нижня частина інтернет – магазину, що закриває усі теги.

Модуль `style.css` - містить параметри оформлення елементів веб-сторінок: зміна тексту і блоків тексту, заголовків, таблиць, посилань форм, графічних елементів, і так далі. Зміна параметрів цього модуля дозволяє значно міняти оформлення веб-сторінок. Лістинг модуля представлений у Додатку В.

### **3.4. Просування інтернет магазину**

Інтернет торгівля в Україні набирає обертів, все більш відповідаючи світовим тенденціям, залучаючи все нових і нових покупців звернути свою увагу не на традиційні, а на електронні полки. У 2017 році через мережу було реалізовано товарів на \$ 350 млн., що в 1,5 рази більше ніж у 2016 році. За результатами 2017 року учасники ринку прогнозують дворазове збільшення обсягів продаж через Інтернет.

На сьогоднішній день побутова техніка та електроніка є найбільш популярними Інтернет товарами. У забезпеченні пропозиції дуже важливу роль відіграє реклама. Зазвичай використовують контекстну рекламу або порівняльні прайс-листи.

Контекстна реклама є одним з найефективніших методів залучення цільової аудиторії до інтернет-магазину. Вона базується на показі оголошень, які відповідають змісту веб-сторінки або пошуковим запитам користувачів. Основні переваги контекстної реклами:

Точне націлювання: Оголошення показуються лише тим користувачам, які вже цікавляться певним товаром або послугою, що збільшує ймовірність покупки.

Гнучкість налаштувань: Можливість налаштовувати рекламу за географічними, демографічними та іншими параметрами.

Оплата за клік (CPC): Ви сплачуєте лише за тих користувачів, які перейшли на ваш сайт, що робить цей метод економічно вигідним.

Швидкі результати: Відразу після запуску кампанії ви можете побачити перші відвідування і замовлення.

Аналітика та оптимізація: Інструменти для аналітики дозволяють відстежувати ефективність кампанії та вносити необхідні корективи.

Сайти порівняння цін дозволяють користувачам порівнювати ціни на товари від різних продавців, що робить їх важливим каналом залучення покупців. Основні переваги використання сайтів порівняння цін:

Збільшення видимості: Ваші товари будуть представлені на популярних платформах, що значно збільшує шанси на їх купівлю.

Цільова аудиторія: Користувачі таких сайтів зазвичай готові до покупки, тому вони є цільовою аудиторією.

Простота використання: Легкість інтеграції з інтернет-магазином через прайс-листи та автоматичні завантаження даних.

Конкурентоспроможність: Можливість порівняння цін з конкурентами допомагає встановлювати оптимальні ціни та залучати більше покупців.

Довіра користувачів: Багато покупців довіряють таким сайтам, оскільки вони надають можливість об'єктивно оцінити різні пропозиції.

Кожний з підходів має свої переваги та недоліки.

До переваг контекстної реклами можна віднести швидку реакцію, точне націлювання, можливість оплати за клік. Недоліки: висока конкуренція, особливо у популярних нішах, що може збільшити вартість кліка.

Сайти порівняння цін мають переваги: велика кількість цільової аудиторії, збільшення видимості товарів, підвищення довіри користувачів. До недоліків можна віднести можливість конкурентів пропонувати більш низькі ціни, що може знизити вашу конкурентоспроможність.

Контекстна реклама і сайти порівняння цін є ефективними методами просування інтернет-магазину побутової техніки та електроніки. Контекстна реклама дозволяє швидко залучити цільову аудиторію і досягти результатів, сплачуючи тільки за кліки, тоді як сайти порівняння цін забезпечують довготривалу видимість товарів і залучення покупців, які готові до покупки. Використання обох методів у комплексі може значно підвищити ефективність вашого маркетингового бюджету і збільшити продажі.

Пошукова контекстна реклама – це текстові оголошення з'являється праворуч або вгорі над основними результатами пошуку у відповідь на пошуковий запит користувача, який містить відповідні оголошенню ключові слова. Тематична реклама – це те ж оголошення, що відображається не в пошуковій системі, а на сторінці, текст якої містить ключові слова, пов'язані з оголошенням.

Оплата контекстної реклами, як правило, здійснюється за моделлю СРС, тобто за клік. При цьому слід уважно підійти до вибору ключових слів і до складання тексту оголошення. Адже використовуючи слова, що не відповідають магазину, можна отримати нецільові кліки і відповідно витрачені гроші даремно.

У результаті дохід від продажів або ледь виправдовує витрати на рекламу або просто не досягає величини витрат на рекламу. У цьому й полягає головна проблема. У більшості випадків найефективнішими є безпосередньо назва моделі побутової техніки або електроніки (наприклад, «Compaq 6710b»). Так, їх запитують рідше, але головне, ціна кліка часом в десятки разів нижче загальних високочастотних запитів. Так як найменувань моделей товарів у

магазинів зазвичай величезна кількість, ефективність реклами збільшується у кілька разів.

Також необхідно, щоб оголошення про продаж конкретної моделі техніки вело саме на сторінку з цією моделлю. Відвідувач не шукатиме на сайті її сам і просто піде на інший сайт. Для цього в системах контекстної реклами є інструменти роботи з великим обсягом слів та оголошень, де легко можна вказати пошукові фрази, адреси потрібних сторінок і заголовки з описами для кожного товару. Очевидно-створити і налаштувати в ручну тисячі оголошень досить проблематично.

При раціональному підході сайти порівняння цін можуть стати найбільш ефективним каналом просування, особливо при правильній оптимізації заголовків і наявності опис товарів в прайс-рядках. Обов'язкова перевірка працездатності посилань, на які ведуть позиції. Оптимізуйте сторінки входу відвідувачів – вони повинні обов'язково відповідати товарам в прайс-рядках. На сайтах лідерів галузі використовується також аукціонна цінова модель розміщення прайсів, тому необхідно також використовувати оптимальну ціну за клік, знайшовши найбільш прийнятне співвідношення між обсягами продажів і ROI (повернення вкладень) на рекламу.

Пошукова оптимізація хоч і є найбільш дешевим каналом просування, назвати його основним можна, тому що даний спосіб має багато ризиків і приносить помітну віддачу лише в довгостроковому періоді. З урахуванням зростаючої конкуренції за позиції у видачі вартість пошукової оптимізації постійно збільшується.

Як же відстежити ефективність грошей витрачених на рекламу і правильно оцінити джерела просування, оптимізувати структуру рекламного бюджету? Для проведення ефективних рекламних кампаній важливий не тільки досвід, але й уміння по ходу рекламної кампанії робити коригування її стратегії для підвищення ефективності. При цьому з'являється задача аналізу рекламного трафіку для виявлення його кількісного та якісного показників.



Тільки на основі цих даних можна судити про ефективність реклами і складати вичерпні звіти для клієнтів. Для цього існують спеціальні системи обліку статистики відвідування і аналізу поведінки відвідувачів на сайті. Саме вони дозволяють точно розрахувати ефективність витрат на рекламу, а також встановити корисність кожного з Ваших джерел просування. Google Analytics ([analytics.google.com](http://analytics.google.com)) – найбільш популярна система, з широкими можливостями, яка до того ж є повністю безкоштовною.

Необхідно поставити спеціальний ява-скрипт на сторінки сайту і налаштувати відстеження дій в персональному акаунті. У підсумку ви зможете отримати інформативні звіти про поведінку відвідувачів: звідки вони вперше прийшли на ваш сайт, які сторінки відвідували, зробили якесь ключове дію (наприклад, покупка товару). Цілком можливо розробити і свою просту систему з урахуванням специфіки вашого бізнесу, використовуючи cookies і унікальні посилання для кожного з каналів просування. Крім того, можна проводити опитування на сайті при оформленні покупки, або по телефону, або в момент завершення доставки. Однак це не самий точний спосіб, тому що покупець не завжди захоче відповісти, або просто може забути це зробити. Варто відзначити, що наприклад, в каталозі порівняння цін Price.ua, крім інтернет - магазинів, присутня значна кількість офф-лайн магазинів (даний ресурс дає можливість вести бізнес в мережі не маючи інтернет-магазину взагалі). В даному випадку надійним способом відстеження ефективності може стати унікальний, спеціально для цих цілей створений телефонний номер і (або) e-mail.

### **3.5. Висновки до розділу 3**

Описана структура та етапи розробки інтернет - магазину з продажу автомобілів.

Розглянуто реалізацію бази даних інтернет – магазину, її логічну структуру та структури таблиць. Описані основні модулі реалізації роботи інтернет - магазину: адміністративна та клієнтська частини.

Проаналізовано можливість SEO-оптимізації.

## ЗАГАЛЬНІ ВИСНОВКИ

В ході виконання кваліфікаційної роботи було досягнуто основну мету роботи - розробка інтернет - магазину.

В процесі дослідження теоретичного матеріалу по цій темі було розглянуто основні потреби створення комерційного веб – сайту, проектування інтернет – магазинів, їх переваги і недоліки. Було вивчено класифікація інтернет - магазинів та мови веб – програмування HTML, JavaScript, PHP. Також було розглянуто методологію проектування інтернет – магазину: його принципи роботи, вимоги до систем управління магазином, функціональну структуру. На основі цього було обране відповідне програмне забезпечення для проектування та реалізації основних функцій продажу.

При розробці архітектури, інтернет-магазин було розділено на дві частини: клієнтську і адміністраторську:

1. Клієнтська частина містить систему формування і підтвердження замовлень (кошик покупця), пошуковий сервіс, каталог товару, інформацію про магазин і контакти, форму авторизації клієнтів.

2. Адміністраторська частина містить інструменти управління та роботи з базою даних і має можливість редагування списку товару і списку категорій та замовлень.

Розроблений інтернет – магазин з продажу автомобілів реалізує основні функції інтернет – магазину. Він простий і зручний у використанні. Модульна система php - скриптів дозволяє швидко покращувати його, з мінімальною втратою часу, і так само застосовувати до нього будь-які дизайнерські рішення.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ajax: Новый подход к web-приложениям [Электронный ресурс] - Режим доступа: <http://www.codenet.ru/webmast/js/ajax/AJAX-New.php> – Заголовок з екрану.
2. <http://www.cmslist.ru>.
3. CMS огляд: CMS, движок сайту, система управління сайтом, mambo, php nuke, netcat, phpbb, invision power board, vbulletin. <http://cmsobzor.ru/news.php>.
4. Content management system <http://www.brutto.ru/informacija/uznat-bolshe/content-management-system>.
5. Framework-системы [Электронный ресурс] - - Режим доступа: <http://wiki.agiledev.ru/doku.php?id=frameworks>– Заголовок з екрану.
6. Java Server Faces [Электронный ресурс].- Режим доступа: <http://java.sun.com/javaee/javaxserverfaces/> – Заголовок з екрану.
7. Microsoft ASP.NET [Электронный ресурс] - Режим доступа: <http://www.asp.net/>– Заголовок з екрану.
8. Microsoft SQL Server [Электронный ресурс]. - Режим доступа: <http://www.microsoft.com/sqlserver/>– Заголовок з екрану.
9. MySQL AB. Mysql [Электронный ресурс]. - Режим доступа: <http://www.mysql.com/>Гаевский, А. Ю. 100% – Заголовок з екрану.
10. PHP and MySQL Web Development (4th Edition)», Luke Welling, Laura Thomson 848 стр., с ил.; ISBN 978-5-8459-1574-0, 978-0-672-32916-6.
11. TNS Gallup Media [Электронный ресурс]. - Режим доступа: [www.tns-global.ru/rus/index.wbp](http://www.tns-global.ru/rus/index.wbp)– Заголовок з екрану.
12. Алексеев, Ю.М. Быстро и легко создаем, программируем, шлифуем и раскручиваем Web – сайт [Текст] / Ю.М. Алексеев - Москва, 2003. – 680 с.
13. Веб Database Application with PHP and MYSQL, 2nd Edition By David Lane, Hugh E. Williams. © O'Reilly, May 2004. ISBN: 0-596-00543-1.

14. Дари К., Баланеску Э. PHP и MySQL. Создание интернет-магазина / К. Дари, Э. Баланеску – СПб.: Вильямс, 2010. – 640 с.
15. Джерк, Н. Разработка приложений для электронной коммерции / Н. Джерк. – СПб. : Питер, 2001. – 512 с.
16. Дронов В. А. PHP, MySQL и Dreamweaver MX 2004. Разработка интерактивных Web-сайтов / В. А. Дронов. – СПб. : БХВ-Петербург, 2005. – 448 с : ил.
17. ДСТУ 3008-06. Державний стандарт України. Документація. Звіти у сфері науки і техніки. Структура і правила Оформлення. Держстандарт України, 2006.
18. Колисниченко, Д.Н. Самоучитель PHP5 / Д.Н. Колисниченко. – 3-е изд. – СПб. : Наука и Техника, 2006. – 576 с.
19. Кузнецов, М.В. PHP 5. Практика разработки Web-сайтов / М.В. Кузнецов, И.В. Симдянов, С.В. Голышев. – СПб.: БХВ-Петербург, 2005. – 960 с.
20. Ломакин П.А. Web – строительство [Текст] / П. А. Ломакин – М.: Майор, 2003. – 565 с.
21. Мазуркевич А. PHP: настольная книга программиста / А. Мазуркевич, Д. Еловой. – Минск: Новое знание, 2003. – 480 с.
22. Мак-Дональд М. HTML5. Недостающее руководство / М. Мак-Дональд. – СПб. : БХВ-Петербург, 2012. – 480 с.
23. Маккоу А. Веб-приложения на JavaScript / А. Маккоу. – СПб. : Питер, 2012. – 288 с.
24. Маклафлин Б. PHP и MySQL. Исчерпывающее руководство / Б. Маклафлин. – СПб. : Питер, 2013. – 512 с.
25. Орлов Л. В. Как создать электронный магазин в Интернет /Л.В. Орлов. – 2-е изд., испр. - М.: Бук-пресс, 2006. - 384 с.
26. Печников Создание Web сайтов без посторонней помощи. – СПб., 2005.
27. Рик Д. HTML 4. Энциклопедия пользователя : пер. с англ. / Дарнелл Рик и др.– К. : Изд-во «ДиаСофт», 2008. – 688 с.

28. Тиге Дж. К. DHTML и CSS для Internet / Джейсон Кренфорд Тиге. – М., 2005.
29. Томсон Л. Разработка Web-приложений на PHP и MySQL /Л. Томсон, Л. Веллинг. – 2-е изд., испр. – СПб: ООО «ДиаСофт», 2003. – 672 с.
30. Ульман Л. MySQL / Л. Ульман. – СПб. : Питер, 2004. – 352 с.
31. Ульман Л. Основы программирования на PHP / Ларри Ульман. – М. : НТ Пресс, 2008.
32. Уорнер Дж. Dreamweaver MX 2004 для "чайников" / Уорнер Джанни, Гарднер Сюзанна : пер. с англ. – М. : Изд. дом "Вильямс", 2004. – 352 с.
33. Фролов, А. В. Базы данных в Интернете: практическое руководство по созданию Web-приложений с базами данных / А.В. Фролов, Г.В. Фролов. – 2-ое изд., испр. – М. : Издательско-торговый дом «Русская Редакция», 2000. – 448 с.
34. Холмогоров, В. Интернет-маркетинг / В. Холмогоров. – СПб. : Питер, 2002. – 272 с.

## ДОДАТКИ

### Додаток А

#### Код розробленого сайту Інтернет магазину

#### Модуль admin.php

```
<?php
if ($_GET['instr']=='update')
{
    include('connect.php');
    $stat=$_GET['p4']+1;
    mysql_query("UPDATE orders SET status='$stat' WHERE id='$_GET[p5]'") or
die(mysql_error());
}
else
{
    print'<table border=0><form action="index.php?page=product.php" method="post"
enctype=multipart/form-data>
    <tr><td valign="top">Назва</td><td><input type="text"
name="product_name"></td>
    <td valign="top">Категорія</td><td><input type="text"
name="product_category" list="list_category">
    <datalist id="list_category">';
    $category_query=mysql_query("SELECT category_name
FROM product_category");
    while($category=mysql_fetch_row($category_query))
    {
        print"<option>$category[0]</option>";
    }
    print'</datalist></td>
    <td valign="top">Ціна<input type="text" name="price"></td></tr>
    <tr><td>Рік випуску</td><td><input type="text" name="year"></td>

    <td>Тип кузова</td><td><select
name="type"><option>Седан</option><option>Хетчбек</option><option>Універсал</option><option>Внедорожник</option></select></td>
    <td>Коробка передач<select
name="corob"><option>Автомат</option><option>Полуавтомат</option><option>Механика</option></select></td></tr>
    <tr><td>Топливо</td><td><select
name="top"><option>Бензин</option><option>Дизель</option></select></td>
    <td>Привід</td><td><select
name="priv"><option>Передній</option><option>Задній</option><option>Повний</option></select></td>

    <td>Кількість дверей<input type="text" name="dveri"></td></tr>
    <tr><td>Об'єм двигателя</td><td><input type="text"
name="dvig"></td><td>Колір</td><td><input type="text" name="color"></td></tr>
    <tr><td>Зображення</td><td><input type="file" name="uploadfile"
accept="image/jpeg,image/png,image/gif"><br>
    <input type="file" name="uploadfile1"
accept="image/jpeg,image/png,image/gif"><br>
    <input type="file" name="uploadfile2"
accept="image/jpeg,image/png,image/gif"><br>
    </td>
    <td valign="top">Опис</td><td><textarea
name="description"></textarea></td>
    <td align="right" valign="bottom"><input type="submit"
value="Додати"></td></tr></table>';

    $order_query=mysql_query("SELECT firstname, lastname, time, status, orders.id
FROM orders, user
WHERE user.id=orders.id_user") or
die(mysql_error());
    print"<strong>Список замовлень</strong>";
    print'<table width='100%'>';
```

```

while($order=mysql_fetch_row($order_query))
{
    print"<tr><td class='order_list'>$order[0] $order[1]</td><td
class='order_list'>$order[2]</td><td class='order_list'>";
    print"<select id='status_'. $order[4].'" onchange='getValues'." ('admin',
'', 'update', '', 'status_ $order[4]', '$order[4]')'.">";
    $stat_query=mysql_query("SELECT id, name FROM status");
    while ($stat=mysql_fetch_row($stat_query))
    {
        print"<option>$stat[1]</option>";
    }
    print"</select></td></tr>";
    $order_detail_query=mysql_query("SELECT product.product_name,
order_products.number, order_products.id_product
FROM product,
order_products
WHERE
order_products.id_order='$order[4]' AND order_products.id_product=product.id") or
die(mysql_error());

    print"<tr><th>Название</th><th>Количество</th><th>Артикул</th></tr>";
    while($order_detail=mysql_fetch_row($order_detail_query))
    {
        print"<tr><td>$order_detail[0]</td><td>$order_detail[1]</td><td>$order_detail[2]
</td></tr>";
    }
    print"<tr><td colspan='3' height='20px'></td></tr>";

    print"
<script>
document.getElementById('status_ $order[4]') .selectedIndex=$order[3]-1;
</script>
";
}
print"</table>";
}
?>

```

## Модуль contact. php

```

<?php
?>

<iframe width="900" height="550" frameborder="0" scrolling="no" marginheight="0" marginwidth="0"
src="https://maps.google.com.ua/maps?oe=utf-8&client=firefox-
a&q=%D0%BE%D0%B1%D0%BE%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F+2+%D0%BB%
D1%83%D0%B3%D0%B0%D0%BD%D1%81%D0%BA&ie=UTF8&hq=&hnear=%D0%9E%D0%B
1%D0%BE%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F+%D1%83%D0%BB.,+2,+%D0%9B%D1%8
3%D0%B3%D0%B0%D0%BD%D1%81%D0%BA,+%D0%9B%D1%83%D0%B3%D0%B0%D0%BD%D1%81%D
0%BA%D0%B0%D1%8F+%D0%BE%D0%B1%D0%BB%D0%B0%D1%81%D1%82%D1%8C&gl=ua&t
=m&z=14&ll=48.564829,39.316914&output=embed"></iframe><br /><small><a
href="https://maps.google.com.ua/maps?oe=utf-8&client=firefox-
a&q=%D0%BE%D0%B1%D0%BE%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F+2+%D0%BB%
D1%83%D0%B3%D0%B0%D0%BD%D1%81%D0%BA&ie=UTF8&hq=&hnear=%D0%9E%D0%B
1%D0%BE%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F+%D1%83%D0%BB.,+2,+%D0%9B%D1%8
3%D0%B3%D0%B0%D0%BD%D1%81%D0%BA,+%D0%9B%D1%83%D0%B3%D0%B0%D0%BD%D1%81%D
0%BA%D0%B0%D1%8F+%D0%BE%D0%B1%D0%BB%D0%B0%D1%81%D1%82%D1%8C&gl=ua&t
=m&z=14&ll=48.564829,39.316914&source=embed" style="color:#0000FF;text-
align:left">Просмотреть увеличенную карту</a></small>
<br><br>Разработка Д.Данилов ©

```

## Модуль header.php



```

!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Магазин</title>
<link href="css/flexslider.css" rel="stylesheet" type="text/css" />
<link href="css/style.css" rel="stylesheet" type="text/css">
<script src="js/jquery-1.9.1.js" language="javascript" type="text/jscript"></script>
<script src="js/function.js" language="javascript" type="text/javascript">
</script>
<script src="js/jquery.flexslider.js" type="text/javascript">
</script>

<script>
function sum(num, price, target, summ)
{
    var sum=document.getElementById(num).value*price;
    document.getElementById(target).innerHTML=sum+' грн.';
}
function summa(string)
{
    var ordersum=0;
    for (var i=0; i<summa.arguments.length; i++)
    {
        operands=summa.arguments[i].split(':');

        sumprice=parseFloat(document.getElementById(operands[0]).value)*parseFloat(operands[1]);
        var ordersum=ordersum+sumprice;
        document.getElementById('summa_td').innerHTML=ordersum+' грн.';
    }
}
</script>
</head>

<body>
<center>
<div style="width:900px; text-align:left">
<table border="0" width="100%">

<tr>
<td valign="top" width="500px">
<?php
include('connect.php');
if ($_GET['exit']=='true')
{
    session_unset();
}

if(isset($_POST['email']) and isset($_POST['pass']) and isset($_POST['firstname']) and isset($_POST['lastname']) and
isset($_POST['pass_rep']))
{
    include('registration.php');
}
$_SESSION['id_user']=-1;
if (isset($_POST['email_log']) and isset($_POST['pass_log']))
{

```

```

        $user_query=mysql_query("SELECT id, firstname, lastname, role FROM user WHERE
email='".$_$_POST['email_log']."' AND pass='".$_$_POST['pass_log']."'") or die(mysql_error());
        $user_id=mysql_fetch_row($user_query);
        if (!isset($user_id[0]))
            print"Неверный e-mail или пароль";
        else
            $_SESSION['id_user']=$user_id[0];
    }

    if (isset($_GET['id']) and isset($_GET['price']))
    {
        $key=sizeof($_SESSION);
        $_SESSION['num']++;
        $_SESSION['price']+=$_GET['price'];
        $_SESSION['pr'.$key]=$_GET['id'];
    }
    if (!isset($_SESSION['price']))
    {
        $_SESSION['num']=0;
        $_SESSION['price']=0;
    }
    if (isset($_SESSION['id_user']) and $_SESSION['id_user']!=-1)
    {
        $user_query=mysql_query("SELECT id, firstname, lastname, role FROM user WHERE
id='".$_$_SESSION['id_user']."'");
        $user_id=mysql_fetch_row($user_query);

        print"Добро пожаловать, $user_id[1] $user_id[2] <br><a class='name'
href=index.php?page=my_order.php>Мои заказы</a> <a class='name' href=index.php?exit=true>Выход</a></td>";
    }
    else
    {
        include('login_form.html');
    }
?>
</td>

<?php
}
print"<td align='right'><table><tr><th
colspan='2'>Корзина</th></tr><tr><td>Товаров</td><td>$_SESSION[num]</td></tr><tr><td>На
сумму</td><td>$_SESSION[price] грн.</td></tr><tr>
<td colspan='2'><a class='cart' href='index.php?page=order.php'>Оформить
заказ</a></td></tr></table></td>";

?>
</tr><tr><td colspan="2" align="left"><form action="index.php?page=search.php" method="post"><input type="text"
name="search" /><input type="submit" value="Поиск" /></form></td></tr>
<tr>
<td colspan='2'>
<?php

$category_query=mysql_query("SELECT category_name, id
FROM product_category") or die(mysql_error());

print"<center><ul id='navigation'><li title='Главная'><a class='link' style='width:50px;' href='index.php'><div
class='home'></div></a></li><li title='Контакты'><a class='link' style='width:50px;'
href='index.php?page=contact.php'><div class='mail'></div></a></li>";
while($category=mysql_fetch_row($category_query))
{

```

```

        print"<li><a class='link' style='width:200px;'
href='index.php?id=$category[1]&page=category.php'>$category[0]</a></li>";
    }

```

```

print"</ul></center>";
?>

```

```

</td></tr><tr>    <td class='main' colspan='2'>

```

## Модуль footer.php</td>

```

</tr>

```

```

</table>

```

```

</div>

```

```

</center>

```

```

</body>

```

```

</html>

```

## Модуль home.php

```

<div class="flexslider" style="width:900px">
    <ul class="slides">
        <li>
            
        </li>
        <li>
            
        </li>
        <li>
            
        </li>
        <li>
            
        </li>
        <li>
            
        </li>
        <li>
            
        </li>
    </ul>
</div>
<!-- jQuery -->
<script
src="http://ajax.googleapis.com/ajax/libs/jquery/1/jquery.min.js"></script>
<script>window.jQuery || document.write('<script src="js/libs/jquery-
1.7.min.js">\x3C/script>')</script>
<script src="js/jquery.flexslider.js" type="text/javascript">
</script>
<script>
// Can also be used with $(document).ready()
$(window).load(function() {
    $('.flexslider').flexslider({
        animation: "slide"
    });
});
</script>
<?php
print"<table align='center' cellpadding='40px' border=0>";
for ($i=0; $i<6; $i+=3)
{
    $product_query=mysql_query("SELECT product.id, product_name, price,
photo.link

```

```

FROM product, photo
WHERE main='1' AND

photo.id_product=product.id

ORDER BY product.id DESC
LIMIT 3 OFFSET $i") or

die(mysql_error());
print"<tr>";
while($product=mysql_fetch_row($product_query))
{
    print"<td class='product'
valign='top'><table><tr><td>".$product[1]."</td></tr><tr><td height='100px'>
    <a href='index.php?page=product.php&id=".$product[0]."'><img
src='".$product[3]."@s.jpg'></a></td></tr><tr><td>".$product[2]."
прн.</td></tr></table></td>";
}
print"</tr>";
}
print"</table>";
?>

```

## Модуль img\_resize.php

```

<?php
/*****
****
Функция img_resize(): генерация thumbnails
Параметры:
    $src          - имя исходного файла
    $dest         - имя генерируемого файла
    $width, $height - ширина и высота генерируемого изображения, в пикселях
Необязательные параметры:
    $rgb         - цвет фона, по умолчанию - белый
    $quality     - качество генерируемого JPEG, по умолчанию - максимальное
(100)
****/
function img_resize($src, $dest, $width, $height, $rgb=0xFFFFFF, $quality=100)
{
    if (!file_exists($src))
    {
        return false;
    }
    $size = getimagesize($src);
    if ($size === false) return false;

    // Определяем исходный формат по MIME-информации, предоставленной
    // функцией getimagesize, и выбираем соответствующую форму
    // imagecreatefrom-функцию.
    $format = strtolower(substr($size['mime'], strpos($size['mime'], '/')+1));
    $icfunc = "imagecreatefrom" . $format;
    if (!function_exists($icfunc)) return false;

    $x_ratio = $width / $size[0];
    $y_ratio = $height / $size[1];

    $ratio      = min($x_ratio, $y_ratio);
    $use_x_ratio = ($x_ratio == $ratio);

    $new_width  = $use_x_ratio ? $width : floor($size[0] * $ratio);
    $new_height = !$use_x_ratio ? $height : floor($size[1] * $ratio);
    $new_left   = $use_x_ratio ? 0 : floor(($width - $new_width) / 2);

```

```

$new_top      = !$use_x_ratio ? 0 : floor(($height - $new_height) / 2);
$src = $icfunc($src);
$idest = imagecreatetruecolor($width, $height);

imagefill($idest, 0, 0, $rgb);
imagecopyresampled($idest, $src, $new_left, $new_top, 0, 0,
    $new_width, $new_height, $size[0], $size[1]);
imagejpeg($idest, $dest, $quality);
imagedestroy($src);
imagedestroy($idest);
return true;
}
?>

```

## Модуль index.php

```

<?php
session_start();
error_reporting('e-error');
include('header.php');
if (($_SESSION['id_user']=='1' or $_SESSION['id_user']=='11') and
!isset($_GET['page']) and $_GET['page']='home.php')
{
    include('admin.php');
}
else
{
    if ($_GET['page'])
    {
        $page=$_GET['page'];
        include($page);
    }
    else
        include('home.php');
}
include('footer.php');
?>

```

## Модуль login\_form.php

```

<table>
    <tr>
        <td>E-mail</td>
        <td><form action="index.php" method="post"><input type="text"
name="email_log" /></td>
    </tr>
    <tr>
        <td>Пароль</td>
        <td><input type="password" name="pass_log"></td>
    </tr>
    <tr>
        <td><input type="submit" value="Вход"/></form></td>
        <td><a
href='index.php?page=registration.php'>Регистрация</a></td>
    </tr>
</table>

```

## Модуль my\_order.php

```

<?php
$my_order_query=mysql_query("SELECT orders.id, time, status.name
FROM orders, status

```

```

WHERE
id_user='$_SESSION[id_user]' AND status.id=orders.status") or
die(mysql_error());
$n=mysql_num_rows($my_order_query);
if ($n==0)
print"У Вас пока нет заказов";
while($my_order=mysql_fetch_row($my_order_query))
{
    print"<strong>$my_order[1]</strong> <em>$my_order[2]</em>";
    $product_query=mysql_query("SELECT product_name, number
                                FROM order_products,
product
                                WHERE
order_products.id_product=product.id AND id_order='$my_order[0]') or
die(mysql_error());
    print"<table border='1' cellspacing='0' width='500px'>";
    while($product=mysql_fetch_row($product_query))
    {
        print"<tr><td
width='450px'>$product[0]</td><td>$product[1]</td></tr>";
    }
    print"</table><br>";
}
?>

```

## Модуль order.php

```

<?php
if ($_SESSION['id_user']==-1)
{
    include('login_form.html');
}
else
{
    if (isset($_POST['address']))
    {
        if ($_POST['address']=='new')
        {
            mysql_query("INSERT INTO user_contact (id_user, city,
address, tel) VALUES ('".$_SESSION['id_user']."', '".$_POST['city']."',
'".$_POST['addr']."',
                                '".$_POST['tel']."'") or
die(mysql_error());
            $id_addr=mysql_insert_id();
            mysql_query("INSERT INTO
                                orders (id_user, id_pay, id_delivery, id_address,
status)
                                VALUES ('".$_SESSION['id_user']."',
'".$_POST['pay']."', '".$_POST['delivery']."', '".$_$id_addr."', '1')") or
die(mysql_error());

            $products=explode(',', $_POST['prod_list']);
            $order_id=mysql_insert_id();
            for($i=0, $j=3; $i<sizeof($products); $i++, $j++)
            {
                $key="num_".$j;
                mysql_query("INSERT INTO order_products (id_product,
id_order, number) VALUES ('".$products[$i]."', '".$_$order_id."',
'".$_POST[$key]."')") or die(mysql_error());
            }
            print"Заказ успешно оформлен";

```

```

        $ses_key=array_keys($_SESSION);
        foreach($ses_key as $value)
        {
            if ($value!='id_user')
                unset($_SESSION[$value]);
        }
    }
    else
    {
        mysql_query("INSERT INTO
            orders (id_user, id_pay, id_delivery, id_address,
status)
                VALUES ('".$_SESSION['id_user']."',
'".$_POST['pay']."', '".$_POST['delivery']."', '".$_POST['address']."', '1')")
or die(mysql_error());
        $products=explode(',', $_POST['prod_list']);
        $order_id=mysql_insert_id();
        for($i=0, $j=3; $i<sizeof($products); $i++, $j++)
        {
            $key="num_".$j;
            mysql_query("INSERT INTO order_products (id_product,
id_order, number) VALUES ('".$products[$i]."', '".$_$order_id."',
'".$_POST[$key]."'") or die(mysql_error());
        }
        print"Заказ успешно оформлен";
        $ses_key=array_keys($_SESSION);
        foreach($ses_key as $value)
        {
            if ($value!='id_user')
                unset($_SESSION[$value]);
        }
    }
}

if (isset($_POST['pay']) and isset($_POST['delivery']) and
!isset($_POST['address']))
{
    if ($_POST['delivery']!='1')
    {
        $address_query=mysql_query("SELECT id, city, address, tel
FROM
user_contact
WHERE
id_user='".$_SESSION['id_user']."'");
        print"<form action='index.php?page=order.php'
method='post'><table>";
        while($address=mysql_fetch_row($address_query))
        {
            print"<tr><td rowspan='3' valign='top'><input
type='radio' name='address' value='$address[0]'></td><td>$address[1]<td></tr>
<tr><td colspan='2'>$address[2]</td></tr>
<tr><td colspan='2'>$address[3]</td></tr>";
        }
        print"</table>";

        print"<table border=0>
            <tr><td rowspan='3' valign='top'><input type='radio'
name='address' value='new'></td><td>Город</td><td><input type='text'
name='city'></td></tr>

```

```

        <tr><td>Адрес</td><td><input type='text'
name='addr'></td></tr>
        <tr><td>Телефон</td><td><input type='text'
name='tel'></td></tr></table>";

        $products=explode(',', $_POST['prod_list']);
        for($i=0, $j=3; $i<sizeof($products); $i++, $j++)
        {
            $key="num_". $j;
            print"<input type='hidden' name='$key'
value='$_POST[$key]'>";
        }
        print"<input type='hidden' name='delivery'
value='".$_POST['delivery']."'>
        <input type='hidden' name='pay'
value='".$_POST['pay']."'>
        <input type='hidden' name='prod_list'
value='".$_POST['prod_list']."'>
        <input type='submit' value='Далее'></form>";
    }
    else
    {

        mysql_query("INSERT INTO
                                orders (id_user, id_pay, id_delivery,
status)
                                VALUES ('".$_SESSION['id_user'].',
'".$_POST['pay'].', '".$_POST['delivery'].', '1')") or die(mysql_error());

        $products=explode(',', $_POST['prod_list']);
        $order_id=mysql_insert_id();
        for($i=0, $j=3; $i<sizeof($products); $i++, $j++)
        {
            $key="num_". $j;
            mysql_query("INSERT INTO order_products (id_product,
id_order, number) VALUES ('".$_products[$i].', '".$_order_id.',
'".$_POST[$key].')') or die(mysql_error());
        }
        print"Заказ успешно оформлен";
        $ses_key=array_keys($_SESSION);
        foreach($ses_key as $value)
        {
            if ($value!='id_user')
                unset($_SESSION[$value]);
        }
    }

    if (!isset($_POST['pay']) and !isset($_POST['delivery']) and
!isset($_POST['address']))
    {
        print"<form action='index.php?page=order.php' method='post'><table
border=1
cellspacing='0'><tr><th>Наименование</th><th>Количество</th><th>Цена</th></tr>"
;

        for ($i=3; $i<sizeof($_SESSION); $i++)
        {
            $keys=array_keys($_SESSION);
            $id_pr=$_SESSION[$keys[$i]];
            $product_query=mysql_query("SELECT product_name, price, id

```



```

FROM product
WHERE

id='$id_pr') or die(mysql_error());

        while($product=mysql_fetch_row($product_query))
        {
            print'<tr><td width="400px">'.$product[0].'/><td
align="center"><input name="num_'. $i.'" id="num_'. $i.'" type="text" size="7"
value="1"
                                onkeyup="'.sum('num_ $i', '$product[1]', 'res_ $i',
'$_SESSION[price]')'.'"
                                ></td><td width="100px" id="res_'. $i.'"
align="right">'.$product[1].' ррн.</td></tr>';
            if ($i==sizeof($_SESSION)-1)
            {
                $string.='num_ $i:$product[1]';
                $product_string.=$product[2];
            }
            else
            {
                $string.='num_ $i:$product[1]', " ";
                $product_string.=$product[2].', ' ;
            }
            $c=1;
        }
        print"<input type='hidden' name='prod_list'
value='$product_string'>";
        print'<tr><td>Сумма</td><td><input type="button" value="Рассчитать"
onclick="summa.'. ($string) .'"></td><td id="summa_td"
align="right">'.$_SESSION['price'].' ррн.</td></tr></table>';
        print"Доставка<br>";

        $delivery_query=mysql_query("SELECT id, name FROM delivery");

        while ($delivery=mysql_fetch_row($delivery_query))

            print"<input type='radio' id='delivery' name='delivery'
value=$delivery[0]>$delivery[1]<br>";
            print"Оплата<br>";

            $pay_query=mysql_query("SELECT id, name FROM pay");

            while ($pay=mysql_fetch_row($pay_query))

                print"<input type='radio' id='pay' name='pay'
value='$pay[0]'>$pay[1]<br>";
                print"<input type='submit' value='Далее'></form>";
            }
        }
    }
}
?>

```

## Модуль product.php

```

<?php
require ('img_resize.php');
if (isset($_POST['product_name']))
{
    $category_query=mysql_query("SELECT id FROM product_category WHERE
category_name='". $_POST['product_category']."'");
    $category_id=mysql_fetch_row($category_query);

```

```

        if(!isset($category_id[0]))
        {
            mysql_query("INSERT INTO product_category (category_name) VALUES
('".$_POST['product_category']."'");
            $category_id=mysql_insert_id();
        }
        else
        {
            $category_id=$category_id[0];
        }

        mysql_query("INSERT INTO product (id_category, product_name,
description, price, year, type, corob, top, privod, dveri, dvig, color) VALUES
('$category_id', '".$_POST['product_name']."', '".$_POST['description']."',
        '".$_POST['price']."', '".$_POST['year']."', '".$_POST['type']."',
        '".$_POST['corob']."', '".$_POST['top']."', '".$_POST['priv']."',
        '".$_POST['dveri']."', '".$_POST['dvig']."', '".$_POST['color']."'");
        $_GET['id']=mysql_insert_id();

        $uploadaddir = 'images/';
        $main=1;
        foreach($_FILES as $file)
        {
            $uploadfile = $uploadaddir.basename($file['name']);
            if (copy($file['tmp_name'], $uploadfile))
            {
                echo "<h3>Файл успешно загружен на сервер</h3>";
            }
            else
            {
                echo "<h3>Ошибка! Не удалось загрузить файл на
сервер!</h3>";
                exit;
            }
            mysql_query("INSERT INTO photo (id_product, link, main) VALUES
('".$_GET[id]', '$uploadfile', '$main')");
            $main=0;
            if (img_resize($uploadfile, $uploadfile.'@s.jpg', 200, 120))
                echo 'Image resized OK';
            else
                echo 'Resize failed!';
        }
    }

    print"<table width='100%' cellpadding='40px'>";

        $product_query=mysql_query("SELECT product.id, product_name, price,
description, year, type, corob, top, privod, dveri, dvig, color
                                FROM product
                                WHERE
product.id='".$_GET['id']."'") or die(mysql_error());
        while($product=mysql_fetch_row($product_query))
        {
            print"<tr><td>";
            $query="SELECT link
                                FROM photo
                                WHERE id_product='".$_GET['id']."'
                                ORDER BY main DESC";
            $img_query=mysql_query($query);
            $img_main=mysql_fetch_row($img_query);

```

```

        print"<table><tr><td colspan='3'><img id='main'
src='\".$img_main[0].\"' width='300px'></td></tr><tr>";
        $img_query=mysql_query($query);
        while ($img=mysql_fetch_row($img_query))
        {
            ?><td></td><?php
        }
        print"</tr></table></td><td align='right'
valign='top'><h3>\".$product[1].\"</h3><br><div align='left'><strong>Год
выпуска</strong>: \".$product[4].\"<br><strong>Тип кузова</strong>:
\".$product[5].\"<br><strong>Коробка передач</strong>:
\".$product[6].\"<br><strong>Топливо</strong>:
\".$product[7].\"<br><strong>Привод</strong>:
\".$product[8].\"<br><strong>Количество дверей</strong>:
\".$product[9].\"<br><strong>Двигатель</strong>:
\".$product[10].\"<br><strong>Цвет</strong>: \".$product[11].\"</div></td></tr>
        <tr><td>\".$product[2].\" грн.</td><td><a class='cart'
href='index.php?id=\".$product[0].\"&price=\".$product[2].\"&page=product.php'>В
корзину</a></td></tr>";
    }

print"</table>";
?>

```

## Додаток. Б. Модуль registration.php

```
<?php
if(isset($_POST['email']) and isset($_POST['pass']) and
isset($_POST['firstname']) and isset($_POST['lastname']) and
isset($_POST['pass_rep']))
{
    $test=mysql_query("SELECT email
                                FROM user
                                WHERE email='".$_POST[email]."'");

    $n=mysql_num_rows($test);

    if($n>0)
    {
        print"Пользователь с таким email уже существует<br>";
        include('registration_form.php');
    }
    else if($_POST['pass']!= $_POST['pass_rep'])
    {
        print"Пароли не совпадают!<br>";
        include('registration_form.php');
    }
    else
    {
        mysql_query("INSERT INTO user (email, pass, firstname, lastname,
role)
                                VALUES ('".$_POST['email']."',
'".$_POST['pass']."', '".$_POST['firstname']."', '".$_POST['lastname']."',
'user')") or die(mysql_error());

        print"Регистрация прошла успешно";
        $_SESSION['id_user']=mysql_insert_id();
    }
}
else
{
    include('registration_form.php');
}
?>
```

## Модуль registration\_form.php

```
Все поля обязательны к заполнению<br>
<table>
    <form action="index.php" method="post">
        <tr><td>E-mail</td><td><input type="text" name="email"></td></tr>
        <tr><td>Пароль</td><td><input type="password" name="pass"></td></tr>
        <tr><td>Пароль повторно</td><td><input type="password"
name="pass_rep"></td></tr>
        <tr><td>Имя</td><td><input type="text" name="firstname"></td></tr>
        <tr><td>Фамилия</td><td><input type="text" name="lastname"></td></tr>
        <tr><td colspan="2"><input type="submit" value="Отправить"></td></tr>
    </form>
</table>
```

## Модуль search.php

```
<?php
$quantity=12;
if(!is_numeric($_GET['pg']))        $page=1;
if ($_GET['pg']<1) $page=1;
$selement_count_query=mysql_query("SELECT id
```

```

FROM product
WHERE

product_name LIKE '%$_POST[search]%'") or die(mysql_error());
$num=mysql_num_rows($element_count_query);
    $pages = $num/$quantity;
    $pages = ceil($pages);
    $pages++;
    if ($_GET['pg']>$pages)
        $page = 1;
    if (!isset($list))
        $list=0;
    $list=--$_GET['pg']*$quantity;
$p=$_GET['pg'];
$limit=++$p*$quantity;

print"<table cellpadding='40px'>";
for ($i=$list; $i<$limit; $i+=3)
{
    $product_query=mysql_query("SELECT product.id, product_name, price,
photo.link
                                FROM product, photo
                                WHERE product_name LIKE
                                '%$_POST[search]%' AND main='1' AND photo.id_product=product.id
                                ORDER BY id DESC
                                LIMIT 3 OFFSET $i") or
die(mysql_error());
    print"<tr>";
    while($product=mysql_fetch_row($product_query))
    {
        print"<td
class='product'><table><tr><td>".$product[1]."</td></tr><tr><td height='100px'>
        <a href='index.php?page=product.php&id=".$product[0]."'><img
src='".$product[3]."@s.jpg'></a></td></tr><tr><td>".$product[2]."
rph.</td></tr></table></td>";
    }
    print"</tr>";
}
print"</table>";
print"<center><ul id='page'>";
if ($_GET['pg']>=1)
{
    print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=1><<</a></li>&nbsp;";
    print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=".$_GET['pg']."><</a></
li>&nbsp;";
}
else
{
    print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=".$_GET['pg']+2."><<<
/a></li>&nbsp;";
    print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=".$_GET['pg']+2."><</
a></li>&nbsp;";
}
}

```

```

        for ($j = 1; $j<$pages; $j++)
        {
            if ($j==($_GET['pg']+1))
            {
                print"<li><a class='linked'
href=index.php?search=$_POST[search]&page=search.php&pg=".$j.">".$j."</a></li>&
nbsp;";
            }
            else
            {
                print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=".$j.">".$j."</a></li>&
nbsp;";
            }
        }
        if ($j>$_GET['pg'] && ($_GET['pg']+2)<$j)
        {
            print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=".(($_GET['pg']+2)).">></
a></li>&nbsp;";
            print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=".(($j-
1)).">></a></li>&nbsp;";
        }
        else
        {
            print"<li><a class='link'
href=index.php?search=$_POST[search]&page=search.php&pg=1>></a></li>&nbsp;";
            print"<li><a class='link'
href=index.php?search=$_POST[search]&age=search.php&pg=1>>></a></li>&nbsp;";
        }
        print"</ul></center>";

?>

```

## Модуль category.php

```

<?php
$quantity=9;

if(!is_numeric($_GET['pg']))
    $page=1;
if ($_GET['pg']<1)
    $page=1;
$element_count_query=mysql_query("SELECT id
FROM product
WHERE

id_category='".$_GET[id]'" ) or die(mysql_error());
$num=mysql_num_rows($element_count_query);
$pages = $num/$quantity;
$pages = ceil($pages);
$pages++;
if ($_GET['pg']>$pages)
    $page = 1;
if (!isset($list))
    $list=0;
$list=--$_GET['pg']*$quantity;
$p=$_GET['pg'];
$limit=++$p*$quantity;

```

```

print"<table cellpadding='40px'>";
for ($i=$list; $i<$limit; $i+=3)
{
    $product_query=mysql_query("SELECT product.id, product_name, price,
    photo.link
                                FROM product, photo
                                WHERE id_category='$_GET[id]'
    AND main='1' AND photo.id_product=product.id
                                ORDER BY id DESC
                                LIMIT 3 OFFSET $i") or

    die(mysql_error());
    print"<tr>";
    while($product=mysql_fetch_row($product_query))
    {
        print"<td
class='product'><table><tr><td>".$product[1]."</td></tr><tr><td height='100px'>
        <a href='index.php?page=product.php&id=".$product[0]."'><img
src='".$product[3]."@s.jpg"></a></td></tr><tr><td>".$product[2]."
rph.</td></tr></table></td>";
    }
    print"</tr>";
}
print"</table>";
print"<center><ul id='page'>";

    if ($_GET['pg']>=1)
    {
        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=1><<</a></li>&nbsp;";
        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=".$_GET['pg']. "><</a></li>&nbsp;
p;";
    }
    else
    {
        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=".$_GET['pg']+2). "><</a></li>
>&nbsp;";
        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=".$_GET['pg']+2). "><</a></li>
&nbsp;";
    }

    for ($j = 1; $j<$pages; $j++)
    {
        if ($j==($_GET['pg']+1))
        {
            print"<li><a class='linked'
href=index.php?id=$_GET[id]&page=category.php&pg=".$j.">".$j."</a></li>&nbsp;";
        }
        else
        {
            print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=".$j.">".$j."</a></li>&nbsp;";
        }
    }
    if ($j>$_GET['pg'] && ($_GET['pg']+2)<$j)
    {

```

```

        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=".( $_GET['pg']+2).">></a></li>
&nbsp;";
        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=".( $j-1).">></a></li>&nbsp;";
    }
    else
    {
        print"<li><a class='link'
href=index.php?id=$_GET[id]&page=category.php&pg=1>></a></li>&nbsp;";
        print"<li><a class='link'
href=index.php?id=$_GET[id]&age=category.php&pg=1>></a></li>&nbsp;";
    }

print"</ul></center>";
?>

```

## Модуль connect.php

```

?php
    mysql_connect('localhost', 'root', '');
    mysql_select_db('avto_tab');
    mysql_query("SET NAMES UTF8");
?>

```

## Модуль flexslider.css

```

.flex-pauseplay span {text-transform: capitalize;}

/* Clearfix for the .slides element */
.slides:after {content: "."; display: block; clear: both; visibility: hidden;
line-height: 0; height: 0;}
html[xmlns] .slides {display: block;}
* html .slides {height: 1%;}

/* No JavaScript Fallback */
/* If you are not using another script, such as Modernizr, make sure you
* include js that eliminates this class on page load */
.no-js .slides > li:first-child {display: block;}

/* FlexSlider Default Theme
******/
.flexslider {margin: 0 0 60px; background: #fff; border: 4px solid #fff;
position: relative; -webkit-border-radius: 4px; -moz-border-radius: 4px; -o-
border-radius: 4px; border-radius: 4px; box-shadow: 0 1px 4px rgba(0,0,0,.2); -
webkit-box-shadow: 0 1px 4px rgba(0,0,0,.2); -moz-box-shadow: 0 1px 4px
rgba(0,0,0,.2); -o-box-shadow: 0 1px 4px rgba(0,0,0,.2); zoom: 1;}
.flex-viewport {max-height: 2000px; -webkit-transition: all 1s ease; -moz-
transition: all 1s ease; transition: all 1s ease;}
.loading .flex-viewport {max-height: 300px;}
.flexslider .slides {zoom: 1;}

.carousel li {margin-right: 5px}

/* Direction Nav */
.flex-direction-nav {*height: 0;}
.flex-direction-nav a {width: 30px; height: 30px; margin: -20px 0 0; display:
block; background: url(../images/bg_direction_nav.png) no-repeat 0 0; position:
absolute; top: 50%; z-index: 10; cursor: pointer; text-indent: -9999px;
opacity: 0; -webkit-transition: all .3s ease;}

```



```

.flex-direction-nav .flex-next {background-position: 100% 0; right: -36px; }
.flex-direction-nav .flex-prev {left: -36px;}
.flexslider:hover .flex-next {opacity: 0.8; right: 5px;}
.flexslider:hover .flex-prev {opacity: 0.8; left: 5px;}
.flexslider:hover .flex-next:hover, .flexslider:hover .flex-prev:hover
{opacity: 1;}
.flex-direction-nav .flex-disabled {opacity: .3!important;
filter:alpha(opacity=30); cursor: default;}

/* Control Nav */
.flex-control-nav {width: 100%; position: absolute; bottom: -40px; text-align:
center;}
.flex-control-nav li {margin: 0 6px; display: inline-block; zoom: 1; *display:
inline;}
.flex-control-paging li a {width: 11px; height: 11px; display: block;
background: #666; background: rgba(0,0,0,0.5); cursor: pointer; text-indent: -
9999px; -webkit-border-radius: 20px; -moz-border-radius: 20px; -o-border-
radius: 20px; border-radius: 20px; box-shadow: inset 0 0 3px rgba(0,0,0,0.3);}
.flex-control-paging li a:hover { background: #333; background:
rgba(0,0,0,0.7); }
.flex-control-paging li a.flex-active { background: #000; background:
rgba(0,0,0,0.9); cursor: default; }

.flex-control-thumbs {margin: 5px 0 0; position: static; overflow: hidden;}
.flex-control-thumbs li {width: 25%; float: left; margin: 0;}
.flex-control-thumbs img {width: 100%; display: block; opacity: .7; cursor:
pointer;}
.flex-control-thumbs img:hover {opacity: 1;}
.flex-control-thumbs .flex-active {opacity: 1; cursor: default;}

@media screen and (max-width: 860px) {
    .flex-direction-nav .flex-prev {opacity: 1; left: 0;}
    .flex-direction-nav .flex-next {opacity: 1; right: 0;}
}

```

## Додаток В. Модуль style.css

```
@charset "utf-8";
/* CSS Document */

body{background:#CCC; font-family:Georgia, "Times New Roman", Times, serif;
font-size:17px}

.product{background-color:#FFF; border-radius:7px; border-style:solid; border-
color:#FFF}
.product:hover{box-shadow: 3px 3px 10px 3px #000;}
.main{background-color:#FFF; box-shadow: 3px 3px 10px 3px #000; border-
radius:4px}

.order_list{border-bottom:thin; border-bottom-style:solid}
#navigation {
    list-style-type:none;
    margin:0px; padding:0px
}

#navigation li{display:inline-block;
margin:0px; padding:0px}
#navigation a {

height:25px;
float:left;

font-family:"Courier New", Courier, monospace; font-style:inherit; font-
size:20px;

margin:0px; padding:0px;
border-right:thin; border-right-style:solid; border-radius:2px;
box-shadow: 1px 1px 5px 1px #000;
}

#page {
    list-style-type:none;
    margin:0px; padding:0px
}

#page li{display:inline-block;
margin:0px; padding:0px}
#page a {
width:27px;
height:27px;
float:left;
margin:0px; padding:0px; text-align:center;
font-size:22px
}
a.link { background-color:#999; text-decoration:none; color:#000;}
a.linked { background-color:#999; text-decoration:none; color:#FFF;}
a.link:hover { background-color:#666; text-decoration:none; color:#fff;}
a.linked:hover { background-color:#666; text-decoration:none;
color:#FFF;}
a.visited{color:#000}
```

```

a.cart{background-color:#999; color:#000; text-decoration:none; border-
radius:5px}
a.cart:hover{background-color:#666; color:#fff; text-decoration:none; border-
radius:5px}
a.name{background-color:#FFF; color:#000; text-decoration:none; border-
radius:5px}
a.name:hover{background-color:#999; color:#000; text-decoration:none; border-
radius:5px}

.home {
font-size: 5px;
height: 1em;
width: 0.5em;
margin-top: 2em;
margin-left: -1em;
border-bottom: none;
border-right: 1.5em solid #2C2C2C;
border-left: 1.5em solid #2C2C2C;
border-top: 1.4em solid #2C2C2C;
position: relative;
}

.home::before {
border-left: 2.4em solid transparent;
position: absolute;
content: "";
top: -2.8em;
right: -2.1em;
width: 0em;
height: 0em;
border-right: 2.4em solid transparent;
border-bottom: 1.5em solid #2C2C2C;
}

.mail{
margin-top: 0.5em;
font-size:8px;
display:block;
position:relative;
border-top:1em solid transparent;
border-left:1.5em solid #2c2c2c;
border-right:1.5em solid #2c2c2c;
border-bottom:1em solid #2c2c2c;
height:0;
width:0;
}
.mail:before{
content: "";
top: -1.23em;
left: -1.5em;
display: block;
position: absolute;
border-top: 1em solid #2c2c2c;
border-left: 1.5em solid transparent;
border-right: 1.5em solid transparent;
border-bottom: 1em solid transparent;
font-size: 0.8em;
}

```

```

.phone {
  display: block;
  position: relative;
  width: 1em;
  height: 0;
  border-bottom: 1.5em solid #2c2c2c;
  border-left: 0.8em solid transparent;
  border-right: 0.8em solid transparent;
  border-radius: 0.5em;
  margin-top: 0.5em;
}

.phone:before {
  content: "";
  position: absolute;
  width: 1.4em;
  height: 0em;
  border-top: 0.2em solid #2c2c2c;
  border-left: 0.4em solid #2c2c2c;
  border-right: 0.4em solid #2c2c2c;
  border-bottom: 0.5em solid transparent;
  display: block;
  top: 0;
  left: 0;
  margin: -0.4em 0 0 -0.6em;
  border-radius: 0.2em;
}

.phone:after {
  content: "";
  position: absolute;
  color: #2c2c2c;
  font: bold 0.64em/1.2em sans-serif;
  position: absolute;
  top: 0.3em;
  left: 0;
  background: #FFF;
  display: block;
  width: 1.2em;
  text-align: center;
  height: 1.2em;
  border-radius: 1.2em;
  border: 0.3em dotted #2c2c2c;
}

```