

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

До захисту допускається  
Завідувач кафедри  
\_\_\_\_\_ Скарга-Бандурова І.С.  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА**

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

НА ТЕМУ:

**Веб-додаток з використанням мов JavaScript та PHP**

---

Освітньо-кваліфікаційний рівень “бакалавр”  
Спеціальність 123 “Комп’ютерна інженерія”

Керівник проекту:

\_\_\_\_\_

(підпис)

**Щербаков Є.В.**

\_\_\_\_\_

(ініціали, прізвище)

Консультант з охорони праці:

\_\_\_\_\_

(підпис)

**Критська Я.О.**

\_\_\_\_\_

(ініціали, прізвище)

Студент:

\_\_\_\_\_

(підпис)

**Кротких В.Д.**

\_\_\_\_\_

(ініціали, прізвище)

Група:

**КІ-15з**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки  
Кафедра Комп'ютерних наук та інженерії  
Освітньо-кваліфікаційний  
рівень бакалавр  
Напрямок підготовки 6.050102 "Комп'ютерна інженерія"  
(шифр і назва)  
Спеціальність \_\_\_\_\_  
(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Завідувач кафедри \_\_\_\_\_  
\_\_\_\_\_ І.С. Скарга-Бандурова

**З А В Д А Н Н Я  
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА**

Кротких Владиславу Дмитровичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Веб-додаток з використанням мов JavaScript та PHP

керівник проекту (роботи) доц. Щербаков Є.В.  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від " " 201\_р. № \_\_\_\_\_

2. Термін подання студентом роботи \_\_\_\_\_

3. Вихідні дані до роботи матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз та постановка задачі; особливості розроблення веб-систем, огляд існуючих засобів розробки, проектування додатку.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

## 6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада Консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	ст. викл. Критська Я. О.		

7. Дата видачі завдання \_\_\_\_\_

Керівник

\_\_\_\_\_

(підпис)

Завдання прийняв до виконання

\_\_\_\_\_

(підпис)

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту ( роботи )	Примітка
1	Постановка завдання	07.05.19-09.05.19	
2	Аналіз предметної області	10.05.19-15.05.19	
3	Вибір засобів для розробки підсистеми	16.05.19-20.05.19	
4	Розробка структури підсистеми	21.05.19-27.05.19	
5	Розробка складових частин підсистеми	28.05.19-03.06.19	
6	Оформлення пояснювальної записки	04.06.19-09.06.19	

Студент

\_\_\_\_\_

( підпис )

Кротких В.Д.

\_\_\_\_\_

(прізвище та ініціали)

Керівник

\_\_\_\_\_

( підпис )

Щербаков Є.В.

\_\_\_\_\_

(прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра:  
77 с., 17 рис., 1 табл., 16 бібліографічних джерел, 1 додаток.

Об'єкт розробки: веб-додаток з використанням мов JavaScript та PHP.

Мета роботи: розробка веб-додатку для служби оренди автомобілів з використанням JavaScript та PHP.

У проекті виконано:

1. У розділі “Аналіз предметної області та постановка задачі” був виконаний огляд галузі, обґрунтування актуальності роботи, порівняння аналогічних систем, були поставлені задачі щодо розробки системи.

2. У розділі “Обґрунтування технологій для створення веб-додатку” були розглянуті засоби, за допомогою яких буде розроблятися система.

3. У розділі “Створення веб-додатку для служби оренди автомобілів” описано проектування підсистеми.

4. У розділі “Охорона праці” був проведений аналіз шкідливих виробничих факторів. На основі цього аналізу запропоновані заходи усунення цих факторів.

Отримано наступні результати: Розроблений інтерактивний веб-додаток, який може використовуватися в комерційній установі для надання послуг у мережі Інтернет.

Практичне значення, галузь застосування роботи: електронна комерція.

ВЕБ-ДОДАТОК, БАЗА ДАНИХ, СЕРВІС, ІНТЕРНЕТ, ІНТЕРФЕЙС,  
СИСТЕМА КЕРУВАННЯ

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

## ЗМІСТ

ВСТУП	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ	7
1.1 Огляд предметної області	7
1.2 Мережа Інтернет	8
1.3 Веб-сайт	9
1.4 Веб-додаток	11
1.5 Актуальність використання онлайн-послуг	13
1.6 Порівняння з аналогічними системами	14
1.7 Формулювання завдання	15
2 ОБҐРУНТУВАННЯ ТЕХНОЛОГІЙ ДЛЯ СТВОРЕННЯ ВЕБ-ДОДАТКУ	16
2.1 Веб-програмування	16
2.2 Мова гіпертекстової розмітки HTML	18
2.3 Каскадні таблиці стилів CSS	19
2.4 Мова програмування JavaScript	20
2.5 Бібліотека JQuery	22
2.6 Мова програмування серверної частини веб-сайтів PHP	22
2.7 Система керування базами даних MySQL	23
3 РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ СЕРВІСУ ОРЕНДИ АВТОМОБІЛІВ	25
3.1 Розробка структури системи	25
3.2 Головний екран	26
3.3 Система реєстрації користувачів	27
3.4 Адміністрування додатку	29

	5
3.5 Сторінка з каталогом автомобілів	32
3.6 Сторінка з водіями	34
3.7 Картка замовлення	35
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	37
4.1 Загальні питання з охорони праці	37
4.1.1 Правові та організаційні основи охорони праці	38
4.1.2 Організаційно-технічні заходи з безпеки праці	39
4.2 Аналіз стану умов праці	40
4.2.1 Вимоги до приміщень	40
4.2.2 Вимоги до організації місця праці	40
4.2.3 Навантаження та напруженість процесу праці	41
4.3 Виробнича санітарія	42
4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу	42
4.3.2 Пожежна безпека	45
4.3.3 Електробезпека	45
4.4 Гігієнічні вимоги до параметрів виробничого середовища	46
4.4.1 Мікроклімат	46
4.4.2 Освітлення	47
4.4.3 Шум та вібрація, електромагнітне випромінювання	49
4.4.4 Вентилювання	49
4.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій	50
ВИСНОВКИ	56
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	57
ДОДАТОК А	58

## ВСТУП

Інтернет та використовувани в його функціюванні новітні технології й засоби комунікації швидко розвиваються. Рішення на основі технологій Інтернету є особливо успішними в комерційній діяльності, що визначають появу свіжого явища в цій галузі – електронної комерції. Це потужна економіка, яка охоплює всі торговельні та фінансові операції, що проводяться через Інтернет, комерційну діяльність, пов'язану з виконанням цих операцій, а також електронну комерцію, маркетингові та страхові послуги. Наразі більшість установ, незалежно від того, чи є це мережа супермаркетів або невеликий спеціалізований дилер, беруть участь у реалізації свого онлайн-представництва. Є також торгові компанії, які працюють виключно в мережі Інтернет.

Велика кількість технологій дозволяє розробляти комерційні веб-програми, що є надійними, ефективними та практичними й зрозумілими для власників цих програм, а також для клієнтів, зацікавлених у придбанні запропонованих послуг.

Головною метою даної дипломної роботи є розробка веб-додатку для онлайн-сервісу оренди автомобілів. За допомогою цієї системи продавець, який володіє базовими навичками роботи з комп'ютером, може розміщувати свої послуги в Інтернеті та обробляти замовлення клієнтів. А клієнти можуть переглядати список послуг, ознайомлюватися з детальним описом моделей та осіб, що пропонують послуги, і, відповідно, оформлювати замовлення через електронну форму.

У майбутньому електронна комерція, веб-сайти та інші види веб-ресурсів будуть усувати конкурентів, якщо вони не придбають власної платформи для просування своєї діяльності. Тому тема цього дипломного проекту є актуальною.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

## 1.1 Огляд предметної області

Розвиток сучасних інтернет-технологій дає можливість створювати веб-сервіси, які дозволяють комерційним підприємствам, зокрема тим, що пропонують послуги оренди авто, знаходити нові можливості для взаємодії з клієнтами.

Перші друковані згадки про прокат автомобілів з'явилися на початку ХХ століття. Ще на зорі розвитку автомобілебудування оренда автомобіля була запропонована як альтернатива таксі. Першим, хто запропонував і організував послуги машини на прокат, був американський власник таксомоторної компанії. На сьогоднішній день прокат автомобілів є практично в кожній країні світу, а завдяки мережі Інтернет будь-який користувач цілодобово може оформити замовлення прокату автомобіля.

В Україні оренда автомобілів почала активно розвиватися на початку ХХІ століття. Сьогодні прокат авто став звичайною послугою, яку можуть дозволити собі навіть ті клієнти, що мають невисокий рівень економічного прибутку.

Цільовою аудиторією даного сервісу є:

- компанії, яким необхідно обслужити клієнтів у повному обсязі;
- працівники підприємств, які здійснюють ділові поїздки;
- особи, у яких тимчасово немає доступу до власного авто;
- люди, які бажають протестувати можливості конкретної моделі;
- туристи.

Переваги системи оренди автомобілів онлайн пов'язані з рядом важливих аспектів, серед яких:

- зручна організація каталогу послуг із вибором за категорією і можливістю швидкого перегляду усіх пропозицій.
- швидкий обмін інформацією між клієнтом та установою.



- можливість ефективно відслідковувати інтереси та потреби користувачів.
- цілодобовий доступ до системи.
- перспектива розширення діяльності установи за межі населеного пункту.

## 1.2 Мережа Інтернет

На сьогоднішній день Інтернет став однією з найважливіших складових життя людини. Його відсутність призведе до інформаційного колапсу по всій планеті і фактично відкине людство на століття назад.

Інтернет – це об'єднання всіх малих мереж в єдину глобальну мережу за допомогою спеціальних кабелів.

Винахідником Інтернету є не одна людина, а ціла група військових з Америки. Причиною послужила Холодна війна з СРСР, і мережі повинні були якимось чином протидіяти загрозі. Проект був розроблений в кінці жовтня 1969, але широке поширення знайшов тільки в 1991 році.

Як працюють провайдери Інтернету? Мережа містить дуже багато інформації і її потрібно десь зберігати. Якщо невеликі веб-сайти можуть бути розміщені на локальних обчислювальних машинах, то такі, як Вікіпедія, Google, AliExpress та інші великомасштабні платформи зберігаються на суперпотужних ПК з величезною кількістю жорстких дисків. Архітектура доступу наступна:

1. Від серверів проводяться виділені високошвидкісні магістралі.
2. До магістралей підключаються шлюзи інтернет-провайдерів.
3. Від постачальника мереж ведуться шлюзи до локальних мереж або приватним ПК.

Після підключення до компанії-постачальника у приміщення заводиться коаксіальний або оптоволоконний кабель, який підключається до

роутера або безпосередньо до ПК. Таким чином встановлюється з'єднання з сервером провайдера.

Для подальшої роботи користувачу необхідно встановити браузер. За допомогою цих програм буде доступний медіаконтент, спілкування, читання книг, огляд новин та інше.

При введенні конкретного запиту користувача в пошуковий рядок браузера програма відправляє сигнал на сервер провайдера. Останній, в свою чергу, передає його на всі доступні світові сервера для пошуку і видачі найбільш релевантної інформації клієнту.

Після підбору даних сигнал повертається до сервера провайдера і знову передається клієнту, але тепер у браузері відображається не пусте поле, а всі необхідні для користувача дані.

Трапляється і так, що користувач не може отримати відповідь на конкретний запит. Це відбувається з кількох причин: недоступний сервер, на якому зберігається інформація, недоступний конкретний файл з даними або запитуваний контент, який заблокований у вашому регіоні. У будь-якому випадку з'являється відповідне повідомлення.

### **1.3 Веб-сайт**

Веб-сайт-це група веб-сторінок, які мають логічне з'єднання одна з одною. Також сайт можна назвати місцем на сервері, яке виділяється для розміщення контенту. Веб-сайт зазвичай являє собою набір тісно пов'язаних між собою програмних файлів і таблиць даних, і сайт також має унікальну адресу (доменне ім'я), все це сприймається в цілому. Доступ до веб-сайту здійснюється через НТТР, тому вони мають таку назву.

НТТР означає "протокол передачі гіпертексту" і є протоколом передачі даних програми. НТТР заснований на технології клієнт-сервер – таким чином, передбачається існування клієнтів, які підключаються і надсилають запити, і є серверна частина, тобто постачальники даних, які чекають на

з'єднання, щоб почати і отримати запит, після чого вони виконують певні дії і повертають повідомлення з результатом у відповідь клієнту.

Разом всі сайти разом утворюють «Всесвітню павутину», де зв'язки об'єднують частини інформації світової спільноти в єдине ціле – комунікацію планетарного масштабу, найбільшу в базі даних планети.

Веб-сайти не завжди мали вигляд, звичний для сучасних користувачів. На початку існування вони представляли собою колекцію статичних сторінок і документів. З розвитком комунікацій збільшується кількість внутрішніх, зовнішніх сторінок і посилань. Із часом сайти стали використовуватися не тільки як сторінки для рефератів, довідки, але і як повнофункціональний офіс, або як медіа або інформаційний центр. На сьогоднішній день практично всі сайти не мають статичної оболонки, їм стали притаманні інтерактивність і динамізм.

В Інтернеті в більшості випадків одному домену присвоюється одне доменне ім'я. Сайти в Інтернеті ідентифікуються іменами доменів, які користувач вводить у відповідне поле програми браузера, що створюється для перегляду веб-сторінок. Звичайно, є й інші випадки: коли один сайт доступний на декількох доменах або на декількох сайтах під одним доменом. Кілька доменів, як правило, використовуються великими сайтами, веб-службами для розділення типів послуг, які вони надають, наприклад, онлайн-ової програми, встановленої Google. Крім того, основні веб-сайти часто розрізняють окремі домени для різних локалізацій – маються на увазі різні мови, країни. Приклад – доменні імена google.com і google.com.ua. Вони ведуть на один сайт, пошукову систему Google, але ті два символи в кінці домену і впливають на мову веб-сторінки.

Комп'ютери, призначені для зберігання сайтів, називаються веб-серверами. Служба оренди сайтів на веб-сервері називається хостинг. Раніше можна було зберігати тільки один сайт на єдиному сервері, але розвиток технологій вплинув і на розробку серверів. Після цього можна було одночасно зберігати декілька сайтів на одному сервері. Сьогодні, якщо

потрібно зберігати тільки один сайт, на одному сервері, то потрібно використовувати таку послугу, як виділений сервер.

Доступ до одного сайту можна здійснювати по різних адресах, а також сайт може зберігатися на кількох серверах. Якщо скопіювати оригінальний сайт на ще один сервер, інший домен будет називатися дзеркальним. Існування й функціонування веб-сайту можливо без застосування серверного програмного забезпечення та підключення до Інтернету, тоді як його можна переглядати та редагувати на будь-якому комп'ютері. Автономна версія сайту використовується в основному в розробці сайту і його тестуванні, і це робиться для того, щоб не показати всім користувачам Інтернету всі помилки і недоліки, які не були негайно помічені і виправлені. Тестерам пропонується проаналізувати сайт на наявність помилок в автономній версії, доступ до яких здійснюється через локальну мережу або Інтернет, але в цьому випадку доступ до сайту обмежується паролем. Це дозволяє підвищити продуктивність великих проектів і встановити їх для великої кількості користувачів.

#### **1.4 Веб-додаток**

Веб-додаток являє собою веб-сайт, на якому розміщені сторінки з частково або повністю несформованим вмістом. Остаточний вміст формується тільки після того, як відвідувач сайту запросить сторінку з веб-сервера. У зв'язку з тим що остаточний вміст сторінки залежить від запиту, створеного на основі дій користувача, така сторінка називається динамічною.

Спектр використання веб-додатків дуже широкий, їх використання приносить певну користь як відвідувачам веб-сайтів, так і їх розробникам.

Веб-додатки дозволяють відвідувачам швидко і легко знаходити необхідну інформацію на веб-сайтах з великим об'ємом інформації.

Даний вид веб-додатків дозволяє здійснювати пошук у вмісті, упорядковувати вміст і переміщатися по ньому зручним для відвідувачів

способом. Прикладами таких додатків можуть служити внутрішні мережі компаній – Microsoft MSDN ([www.msdn.microsoft.com](http://www.msdn.microsoft.com)) і Amazon.com ([www.amazon.com](http://www.amazon.com)).

Довгий час використовувався метод, при якому дані, введені в HTML-форми, відсилалися для обробки CGI-додатків або спеціально призначеним працівникам у вигляді повідомлень електронної пошти. Веб-додаток дозволяє зберігати дані безпосередньо в базі даних, а також отримувати дані і формувати звіти на основі отриманих даних для аналізу. Як приклад можна привести інтерактивні сторінки банків, сторінки для контролю товарних запасів, соціологічні дослідження та опитування, а також форми для зворотного зв'язку з користувачами.

Веб-додаток звільняє веб-дизайнера від рутинної роботи постійного оновлення HTML-сторінок сайту. Постачальники вмісту, наприклад редактори новин, відповідають за наявність свіжого матеріалу, а веб-додаток стежить за автоматичним оновленням сайту. Як приклад, можна привести веб-версію журналу «The Economist» ([www.economist.com](http://www.economist.com)) і служби новин CNN ([www.cnn.com](http://www.cnn.com)).

Будь-який веб-додаток являє собою набір статичних і динамічних веб-сторінок. Статична веб-сторінка – це сторінка, яка завжди відображається перед користувачем в незмінному вигляді. Веб-сервер відправляє сторінку за запитом веб-браузера без будь-яких змін. На противагу цьому, сервер вносить зміни в динамічну веб-сторінку перед відправкою її браузеру. У зв'язку з тим що сторінка змінюється, вона називається динамічною.

Наприклад, можна створити сторінку, на якій будуть відображені результати програми оздоровлення. При цьому деяка інформація (наприклад, ім'я співробітника і його результати) буде визначатися в момент запиту сторінки співробітником.

### 1.5 Актуальність використання онлайн-послуг

З кожним роком все більше установ отримують представництво в мережі Інтернет. Особливу нішу займають заклади, що спеціалізуються на торгівлі та наданні послуг. Фірми із традиційною моделлю існують, але з розвитком веб-технологій їхній перехід у інтернет-простір неминучий. Однозначно в цій сфері не обійтися без новітніх програмних засобів і сучасної техніки, як то комп'ютер чи смартфон, які допомагають підприємствам працювати ефективно й покращувати економічні показники.

У табл. 1.1 вказані характерні ознаки електронної і традиційної комерції.

Таблиця 1.1 – Порівняльний аналіз традиційної та цифрової моделей сучасної комерційної діяльності.

Назва характеристики	Традиційний магазин	Веб-платформа
Розташування	Торговий зал	Веб-сервер, доступний будь-де
Переміщення	Клієнт ходить по торговому залу й оглядає товар.	Покупець відвідує веб-сторінки й переходить по посиланням.
Взаємодія з продавцем	Спілкування віч-на-віч.	Консультація через e-mail, онлайн-чат, відеозв'язок, месенджер.
Замовлення послуги	Безпосередньо у продавця	Оформлення електронної форми/картки через Інтернет.
Видача рахунку покупцеві	Виписка рахунку і вручення його покупцеві на оплату.	Пересилання рахунку через Інтернет або користування послугами доставки.
Оплата товару/послуги	У касі за допомогою банківської карти чи готівки.	Оплата через електронний гаманець, веб-банкінг або замовлення кур'єрських послуг.

## 1.6 Порівняння з аналогічними системами

Перший досвід розробки системи електронної комерції був здійснений авіакомпанією – спільно з ІВМ була розроблена програма для резервування місць на авіарейси, але це було зроблено в 1960 році. Але лише у 1992 був даний старт комерційним платформам, більш схожим на сучасні інтернет-магазини.

Першими інтернет-магазинами були сайти для продажу книг. Оскільки ця галузь тоді була не дуже розвинута, деякі фактори вплинули на недостатню конверсію. Головною проблемою був поганий розвиток Інтернету та слабка довіра консервативних клієнтів до таких незвичайних на той момент магазинів. Але з технологічним прогресом довіра покупців зростає. Число замовлень зростало в геометричній прогресії.

Сучасні комерційні веб-платформи пропонують приблизно однакову функціональність, їх головні відмінності полягають у дизайні й кількості вбудованих функцій. Виходячи з вищевикладеного, всі інтернет-системи переслідують єдину ціль – підвищення конкурентоспроможності, за якої саме і трапляється підвищення якості системи та максимальної адаптації сервісу до широкої аудиторії.

Типові вимоги, які ставлять перед веб-платформою більшість клієнтів – це простий інтерфейс, який буде легко зрозуміти і використовувати, швидке пересування по розділах і сторінках, щоб краще сприймати інформацію про продукти, зручність і безпека покупки, а також велика кількість способів оплати і прийнятний час доставки. Таким чином, чим швидше продукт потрапляє в руки клієнта, тим до більш позитивних висновків він приходять.

У даній роботі розроблена система з чітким і легким у використанні інтерфейсом, легкою для сприйняття інформацією про товар, а також можливістю замовлення, яке здійснюється через інтегровані засоби взаємодії між клієнтом і продавцем.

## 1.7 Формулювання завдання

Призначенням розробки є створення веб-додатку для фірми, що пропонує оренду авто й водійські послуги. Система призначена для користувачів інтернету, які бажають подивитися інформацію про можливі транспортні засоби або водіїв, а також зробити замовлення.

Доступ до системи має враховувати наступні категорії користувачів:

1. Адміністратор сайту. Цей користувач має доступ не тільки до всієї інформації (введення і редагування даних), але також може супроводжувати програмну частину і здійснювати її доопрацювання (якщо є необхідність).

2. Гість. Цей користувач може переглядати інформацію і залишати повідомлення у формі зворотнього зв'язку.

3. Зареєстрований користувач. Створивши акаунт на сайті і підтвердивши реєстрацію, користувачі можуть замовляти послуги на сайті, проте вони не мають доступу в адміністраторський розділ.

При розробці підсистеми замовлення транспортних послуг потрібно вирішити наступні завдання:

- дослідити задачі веб-сервісу послуг автопрокату;
- виконати порівняння з аналогічними сервісами;
- ознайомитись з наявним інструментарієм веб-розробки;
- розробити дизайн додатку;
- забезпечити зберігання й обробку інформації про користувачів додатку;
- розробити керування змістом сторінок;
- розробити набір сторінок з матеріалом.

Головним пріоритетом розробки додатку повинно бути полегшення користування сервісом транспортних послуг. Система повинна бути зручною для користувачів, простою в керуванні, інтуїтивно зрозумілою.



## 2 ОБҐРУНТУВАННЯ ТЕХНОЛОГІЙ СТВОРЕННЯ ВЕБ-ДОДАТКІВ

### 2.1 Веб-програмування

Веб-програмування – це розділ програмування, який робить можливим створення веб-додатків, що забезпечують функціональність для динамічних веб-сайтів у мережі Інтернет.

Абсолютно всі технології програмування, які використовують програмісти для розробки веб-сайтів, можна розділити на дві групи: клієнтські і серверні.

Веб-додаток – програмний додаток, в якому клієнтом виступає браузер, а сервером – веб-сервер. Браузер може бути реалізацією так званих тонких клієнтів. Браузер здатний відображати веб-сторінки і, як правило, входить до складу операційної системи, а функції його поновлення та супроводу лежать на постачальнику операційної системи. Логіка додатка зосереджується на сервері, а функція браузера полягає в основному в відображенні інформації, завантаженої з мережі з сервера, і передачі назад даних користувача. Одним з переваг такого підходу є той факт, що клієнти не залежать від конкретної операційної системи користувача, і веб-додатки, таким чином, є межплатформенними сервісами. Внаслідок цієї універсальності і відносно простоту розробки веб-додатки стали широко популярними в кінці 1990-х – початку 2000-х років.

Істотною перевагою побудови веб-додатків для підтримки стандартних функцій браузера полягає в тому, що функції повинні виконуватися незалежно від операційної системи даного клієнта. Замість того, щоб писати різні версії для Microsoft Windows, Linux, Unix та інших операційних систем, додаток створюється один раз і розгортається на будь-якій платформі. Однак різна реалізація HTML, CSS, DOM і інших специфікацій в браузерах може викликати проблеми при розробці веб-додатків і подальшої підтримки. Крім того, можливість користувача налаштовувати багато параметрів браузера

(наприклад, розмір шрифту, кольору, відключення підтримки сценаріїв) може перешкоджати коректної роботи програми.

Веб-додаток отримує запит від клієнта і виконує обчислення, після цього формує веб-сторінку і відправляє її клієнту через мережу з використанням протоколу HTTP. Саме веб-додаток може виступати в якості клієнта інших служб, наприклад, бази даних або іншого веб-додатку, розташованого на іншому сервері. Яскравим прикладом веб-додатку є система керування вмістом статей Вікіпедії: безліч її учасників можуть брати участь у створенні мережевої енциклопедії, використовуючи для цього браузери своїх операційних систем (чи то Microsoft Windows, GNU / Linux або будь-яка інша операційна система) і не завантажуючи додаткових виконуваних модулів для роботи з базою даних статей [1].

В даний час набирає популярність новий підхід до розробки веб-додатків, званий Ajax. При використанні Ajax сторінки веб-додатку не перезавантажуються цілком, а лише довантажують необхідні дані з сервера, що робить їх більш інтерактивними і продуктивними.

Програми, написані з використанням мов клієнтського програмування, обробляються на стороні клієнтського комп'ютера. У більшості випадків процеси, такі як браузер, виконують всю обробку. Звідси існують деякі недоліки мови клієнта, а саме, що кожна обробка залежить безпосередньо від браузера, що використовується користувачем. Користувач може добровільно або заборонити введення в експлуатацію клієнтських програм у своєму браузері, а потім їх виконання та реалізація не включає програму, як якщо б розробники не хотіли.

Крім того, може статися, що один і той же скрипт програмного забезпечення обробляється абсолютно різними способами в різних версіях одного браузера або в різних браузерах. З іншого боку, розробник, який покладається на веб-додатки на стороні сервера, повинен спростити його роботу і зменшити навантаження на сервер через веб-додатків, запущених на

стороні браузера, тому що сторінку не потрібно перезавантажувати дуже часто. на основі коду, отриманого від сервера.

Мова сервера є протилежною від клієнта з точки зору обробки програми. Користувач, який відправив запит на веб-сторінку (тобто перейшов на неї через посилання або ввів адресу в адресному рядку вашого браузера), тобто веб-сторінку, що викликається, спочатку обробляється на сервері, тобто всі програми, пов'язані з сторінкою страчені. і тільки після повернення відвідувачу через мережу як файл.

Виконання такої програми повністю залежить від сервера, на якому знаходиться сайт, і від версії мови програмування, яку підтримує сервер. Існує досить багато мов серверного програмування, зокрема: Perl, PHP, Ruby, Python, такі технології, як ASP.NET, Groovy, Java. Сьогодні однією з найпоширеніших технологій є мова програмування PHP, яка більш детально описана в розділі 2.6.

Одна з важливих особливостей серверних мов програмування – підтримка безпосереднього контакту та використання системи керування базами даних (СУБД). Це сервер бази даних, який зберігає інформацію належним чином і може використовуватися в потрібний час.

## **2.2 Мова гіпертекстової розмітки HTML**

HTML, або універсальна мова для створення та розмітки документів в Інтернеті, допомагає створювати багато інтерактивних документів з мультимедійними та гіперпосиланнями, такими як веб-сайти, презентації та інтерфейси між різними додатками. Текстові документи в HTML-розмітці з розширенням .html або .htm містять і розробляють спеціальні програми, які відображають документ в редагованому вигляді на пристрої виводу, який може бути монітором комп'ютера або екраном мобільного пристрою. Див. Інші веб-сайти в Інтернет-розмітці в HTML. У рідкісних випадках це робиться в XHTML. XHTML – це вдосконалена мова розмітки гіпертексту,

що відноситься до сімейства мов розмітки XML. Програми, які стосуються мови розмітки та візуалізації, називаються "браузерами". У браузерах часто зручно і легко використовувати інтерфейс для браузера і виводити його на зовнішні пристрої, а при необхідності вхід користувача браузера повинен бути відправлений на сервер, оскільки сайт вже існує.

HTML не є мовою програмування. Тому необхідно виконувати дії на HTML-сторінці, наприклад, видалення тексту або різних візуалізацій у відповідь на дії користувача вимагає використання спеціалізованих інструментів, які розширюють функціональність HTML, мови програмування, такого як JavaScript.

HTML-код базується на тегах, або прихованих ключових словах, які надають інструкції для форматування документа. Теги HTML – це ключові слова, що визначають, яким чином веб-браузер форматує та відображає вміст даної веб-сторінки.

Більшість тегів повинні складатися з двох частин, які відкривають і закривають елемент стилю. Приклад: `<HTML>` День відкриття та `</ HTML>` День закриття. Закриваючий тег має той самий текст, що й початковий тег, але також символ косою лінії (/).

### **2.3 Каскадні таблиці стилів CSS**

Cascading Stile Sheets (CSS) – це мова стилів, яка змінює зовнішній вигляд документа в мові розмітки. Найбільш поширеним його застосуванням є стилізація веб-сторінок, написаних на HTML. CSS призначена в першу чергу для забезпечення можливості розділяти вміст документа (написаного на HTML або схожою мовою) і зовнішній вигляд документа, включаючи такі елементи, як зовнішній вигляд, кольори і шрифти.

Звичайно, для того, щоб надати сайту прийнятний вигляд, потрібно визначити багато різних стилів. Цілий набір стилів CSS включено до списку стилів. Існує два типи стилів: внутрішній і зовнішній. Залежно від того, де

визначається інформація про стиль: безпосередньо на самому сайті або в окремому файлі, який пов'язаний з веб-сайтом.

З моменту впровадження CSS зовнішні таблиці стилів є найкращим способом розробки веб-сторінок. Вони спрощують створення веб-сайтів, і оновлення відбувається швидше. Зовнішня таблиця стилів підсумовує всю інформацію про стиль у один файл, який потім доповнює сторінку, написавши весь рядок коду. Ви можете прикріпити один і той же зовнішній таблицю стилів для кожної сторінки сайту, щоб створити один дизайн. Оновлення зовнішнього вигляду всього сайту складається з редагування одного текстового файлу – зовнішніх таблиць стилів [2].

Коли переглядач відображає документ, він повинен поєднувати вміст документа з інформацією про стиль. Він керує документом у два етапи:

1. Перетворення HTML і CSS у DOM (Document Object Model) – документ у пам'яті комп'ютера, тобто поєднується вміст документа з його стилем.

2. У браузері відображається вміст DOM.

DOM має структуру дерева. Усі елементи, атрибути і текстові фрагменти в мові прапора стають вузлами DOM у дереві. Вузли визначаються їхнім відношенням до інших вузлів DOM.

Розуміння DOM допомагає розробляти, виправляти та керувати CSS, оскільки DOM є місцем, де збирається вміст CSS та документа.

## **2.4 Мова програмування JavaScript**

Важливим інструментом для розробки більш інтерактивного інтерфейсу системи є мова програмування JavaScript (JS). Найчастіше вона використовується в поєднанні з мовою гіпертекстової розмітки для забезпечення динаміки та респонсивності веб-сайту. JavaScript використовує об'єктно-орієнтоване й імперативне програмування.

JavaScript – це мова програмування, в якій інтерпретується програма. Щоб запустити програму, не потрібно її компілювати, достатньо запустити її у браузері.

Основні характеристики JS:

1. Додавання нових тегів HTML до веб-сторінки, видалення існуючих та додавання стилів до елементів.
2. Відповідь на дії користувача, відстеження курсору, керування курсором миші та багато іншого.
3. Посилання на запити сервера та отримання відповідей без перезавантаження сторінки.

І це лише головні особливості, адже мова Javascript набагато ширша. Однак існують певні обмеження:

1. Основним обмеженням є те, що JS не може нічого записувати на диск, і тому доступ програми безпосередньо до операційної системи не передбачений.
2. JavaScript на одній вкладці не може впливати на інші вкладки веб-переглядача.
3. Мова не може надіслати запит до інших доменів, оскільки існують обмеження безпеки.

Веб-браузер, який працює на комп'ютері-клієнті, забезпечує середовище, в якому JavaScript має доступ до об'єктів, які являють собою вікна, меню, діалоги, текстові області, фрейми, куки і введення-виведення в Веб-сторінку. Крім того, оглядач дозволяє приєднати сценарії на мові JavaScript до таких подій, як завантаження і вивантаження сторінок і графічних образів, натискання клавіш і рух миші, вибір тексту і пересилання форм. При цьому програмний код сценаріїв тільки реагує на події і тому не потребує головній програмі. Набір об'єктів, що надаються оглядачем, відомий під назвою Document Object Model (DOM). Веб-сервер забезпечує інше середовище, в якій об'єктами є запити до баз даних, клієнти, файли і механізми блокування та спільного використання даних. Спільне

використання сценаріїв і на стороні клієнта, і на стороні сервера дозволяє розподілити обчислення між ними і забезпечити необхідний користувацький інтерфейс для веб-додатку.

## **2.5 Бібліотека JQuery**

jQuery – це невелика, швидка і багатофункціональна бібліотека, яка дозволяє писати набагато швидший код JS. jQuery допомагає реалізувати анімацію, обробники подій, полегшує вибір елементів з DOM і створення запитів Ajax.

Бібліотека працює з усіма браузерами, тому вам не доведеться турбуватися про перехресний перегляд. Також написано багато додатків jQuery, які разом з JavaScript також допомагають вирішити певну проблему в організації веб-сайту.

Щоб підключитися до сторінки бібліотеки, потрібно завантажити бібліотеку з сайту, скинути її до папки з сайтом, а потім підключити до нижньої частини сторінки після всього HTML-коду [3].

## **2.6 Мова програмування серверної частини веб-сайтів PHP**

На сьогоднішній день PHP – це потужний мультиплатформовий набір засобів, який розташовується на сервері і призначається для обробки коду, вбудованого в html-документи. Мову PHP найкраще охарактеризувати як вбудовану мову веб-сценаріїв, що працює на стороні сервера та дозволяє розробникам швидко й ефективно створювати динамічні веб-додатки. PHP має спільні риси з мовою C, хоча розробники включили в неї деякі корисні засоби з інших мов програмування, в тому числі з Perl, Java і C++.

Завдяки PHP з'являється можливість створювати динамічні веб-сторінки. Тобто розробник створює їх як звичайно, використовуючи мову

HTML і вставляючи в документ, у міру необхідності, PHP-команди. Файли, створені таким чином, зберігаються на сервері. Коли відвідувач звертається до вашої сторінці, PHP обробляє вбудовані в неї команди і видає результат браузеру користувача – точно так само, як це робить CGI-програма, написана на C або Perl. Тільки на відміну від останньої, PHP має ряд переваг:

- створення та налагодження скриптів PHP значно простіше, ніж налагодження та створення скриптів на інших мовах;
- оскільки PHP-команди просто вставляються в текст html-документа, відпадає необхідність в різних IDE (інтегроване середовище розробки);
- для вирішення різних специфічних завдань не потрібно писати і налагоджувати численні маленькі CGI-програми, що зводить до мінімуму час доступу до ваших сторінок, а також тривалість розробки сторінок і сайту в цілому.

PHP є молодого і перспективною мовою, яка пропонує, щонайменше, наступні переваги: широка інтеграція з новітніми технологічними рішенням, сумісність з серверами й базами даних, простота і безкоштовність [4].

PHP має класи підключення до багатьох СУБД, таких як MySQL, MS ACCESS та інші. Вибір зберігання даних як в файлах (в окремих випадках)? так і в СУБД MySQL обґрунтований тим, що використовується зв'язка Apache, PHP, MySQL, типова для Unix-систем.

Оскільки додаток працюватиме з БД, яка буде зберігати і надавати дані для роботи, слід обрати СУБД.

## **2.7 Система керування базами даних MySQL**

MySQL є реляційною системою керування базами даних. У реляційній базі даних дані зберігаються не довільно, а в окремих таблицях, що збільшує швидкість і гнучкість. Таблиці пов'язані між собою відносинами, тому, коли розробник виконує запит, він може групувати дані з кількох таблиць. SQL як



складова системи MySQL є структурованою мовою запитів, а також найбільш поширеною мовою за замовчуванням для доступу до баз даних.

MySQL є відкритим програмним забезпеченням. Кожен може змінити і застосувати його. Це програмне забезпечення можна отримати в Інтернеті і використовувати безкоштовно. У цьому випадку кожен користувач може вивчати вихідний код і вносити до нього зміни, які він вважає необхідними.

Веб-програмісти обирають СУБД MySQL завдяки високій швидкості, надійності і простоті у використанні. MySQL також пропонує ряд практичних функцій, які були розроблені в тісному контакті з користувачами. Спочатку сервер MySQL був розроблений для керування великими базами даних для забезпечення більшої швидкості порівняно з існуючими аналогами на той час. І вже кілька років цей сервер успішно використовувався в промислових умовах з високими вимогами. Хоча MySQL постійно вдосконалюється, сьогодні він пропонує безліч корисних функцій. Завдяки своїй доступності, швидкості та безпеці, MySQL дуже добре підходить для доступу до баз даних в Інтернеті [5].

MySQL – це клієнт-серверна система, що включає багатопотоковий SQL-сервер, який підтримує різні комп'ютери з базами даних, а також різні клієнтські програми та бібліотеки, засоби керування та різноманітні програмні інтерфейси (API). Ми також надаємо серверу MySQL як багатопоточну бібліотеку, яка може бути підключена до спеціального додатка, щоб забезпечити більш компактний, швидший і простий у керуванні продукт. Також доступна велика кількість програм для MySQL, в основному безкоштовних.

## 3 РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ СЕРВІСУ ОРЕНДИ АВТОМОБІЛІВ

### 3.1 Розробка структури системи

Задачею розроблюваної системи є забезпечення онлайн-замовлень послуг оренди автомобілів або особистих водіїв.

Система повинна забезпечувати:

1. Вибір транспортних засобів з повною інформацією про кожну модель.
2. Вибір водіїв з інформацією про професійні якості кожної особи.
3. Спрощення оформлення замовлення.
4. Зворотній зв'язок.

Програмне забезпечення на клієнтській частині спроектоване за допомогою HTML, CSS, JavaScript, а серверна частина – на мові PHP, яка використовує MVC-модель, щоб об'єднувати серверну й клієнтську частини. База даних, з якою працюватиме система, – Microsoft MySQL.

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

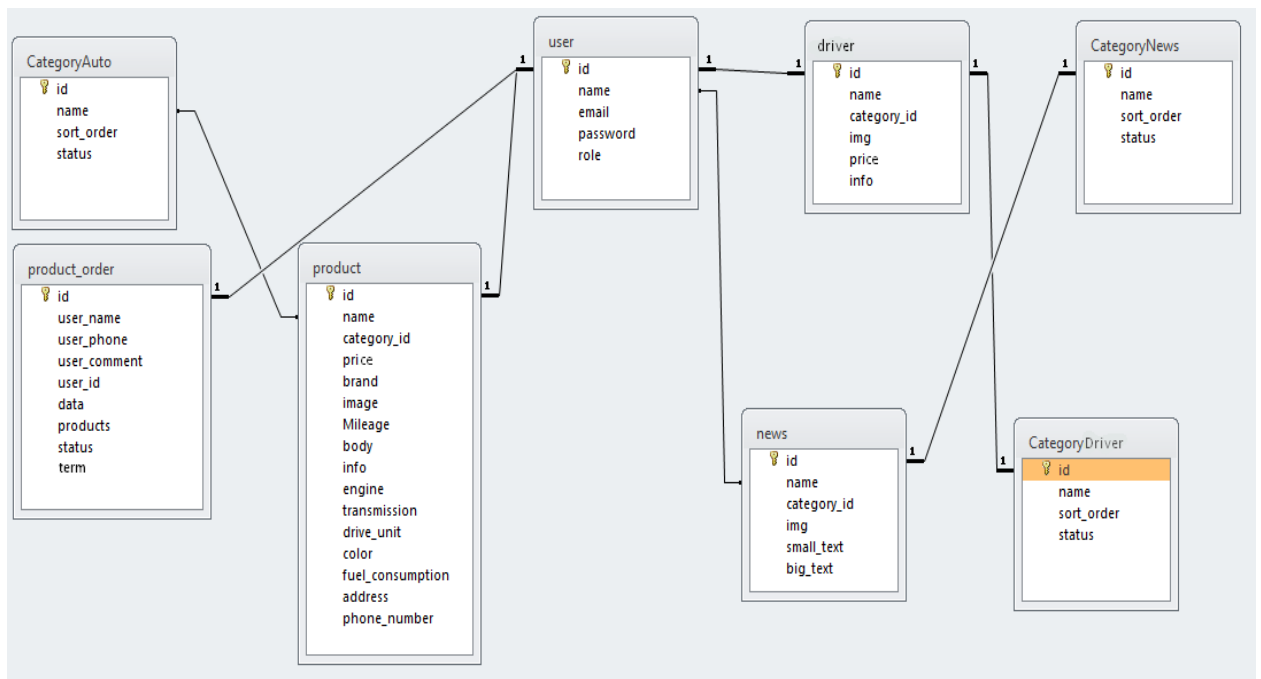


Рисунок 3.1 – Структура бази даних.

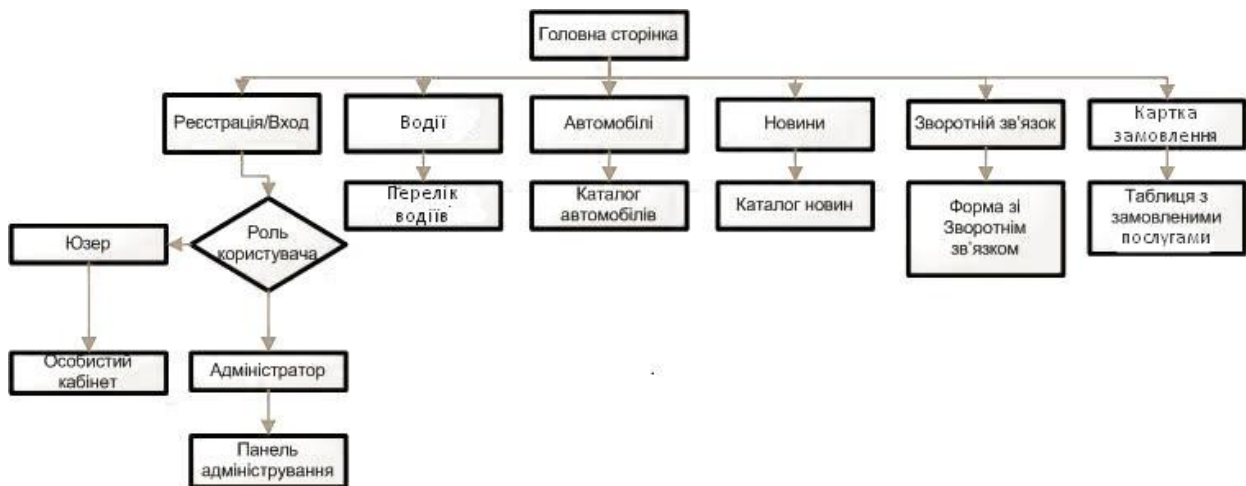


Рисунок 3.2 – Структура розроблюваної системи.

### 3.2 Головний екран

Головна сторінка служби містить в собі логотип, навігаційне меню і основну область для того, щоб користувач при першому відвідування міг отримати уявлення про сервіс. Таким чином, клієнт перед собою головний екран, як на рис. 3.3:

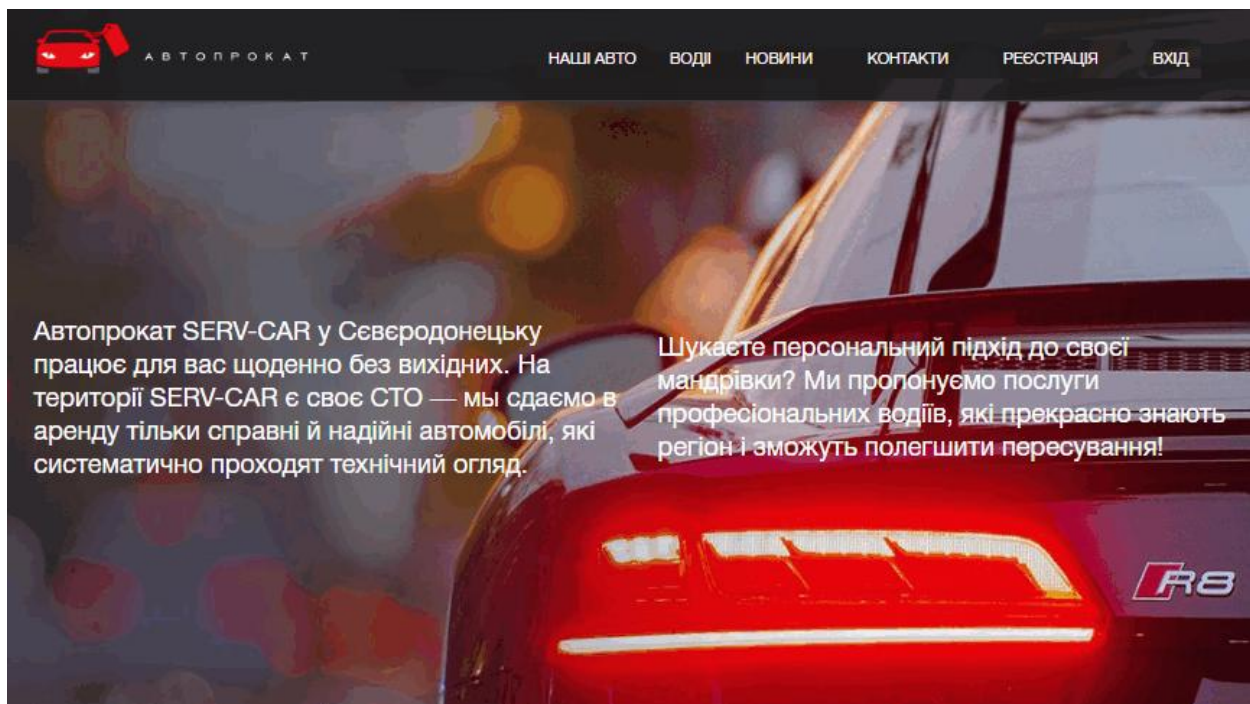


Рисунок 3.3. – Головний екран додатку

### 3.3 Система реєстрації користувачів

Впровадження системи реєстрації користувачів є обов'язковим. При її проектуванні особливим пріоритетом була зручність використання веб-додатку. В системі відсутня примусова реєстрація – тобто перегляд основних сторінок можливий без наявності акаунту. У той же час виконувати керування має право тільки авторизований користувач, який має роль адміністратора.

На рис. 3.4. показаний алгоритм роботи реєстрації й авторизації у вигляді схеми.

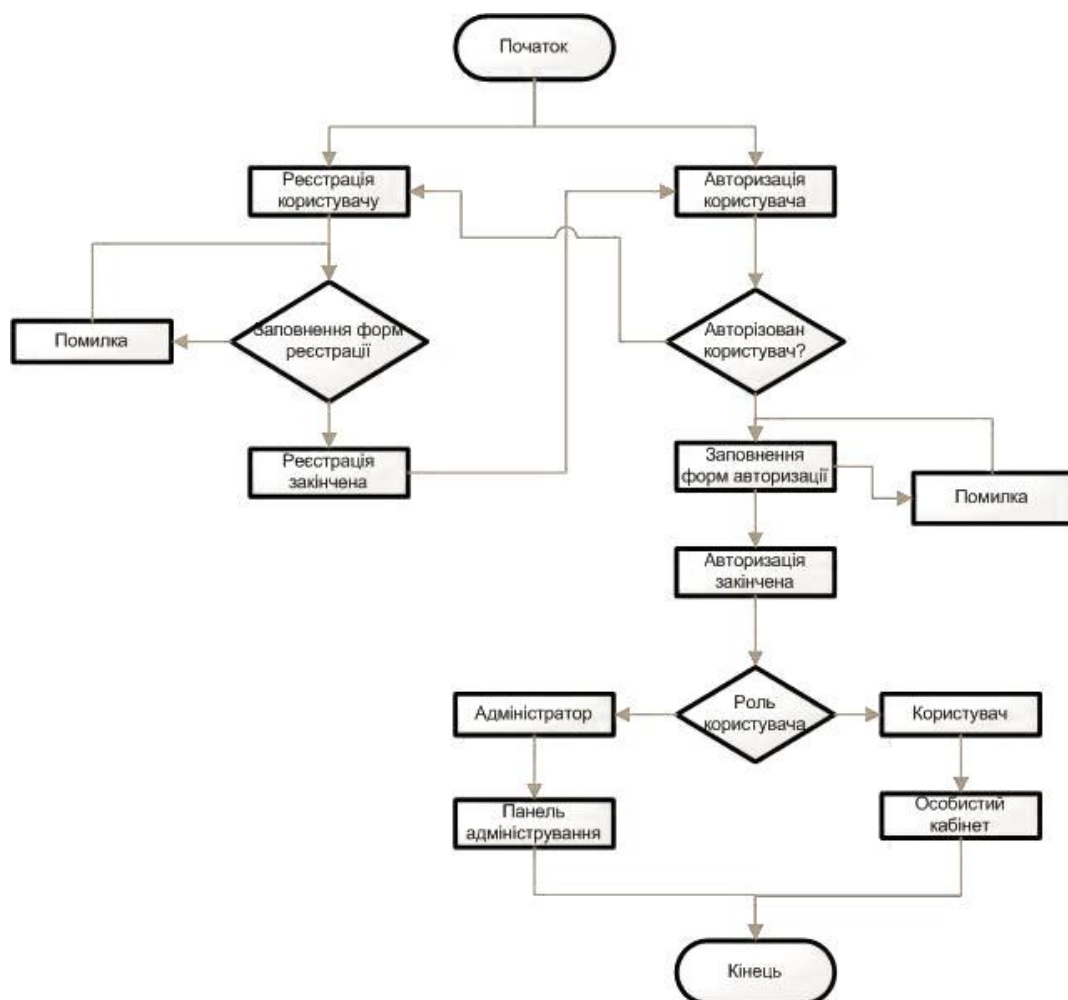


Рисунок 3.4 – Схема алгоритму здійснення реєстрації й авторизації.

Перш ніж створити обліковий запис, користувач має потрапити у відповідне вікно, натиснувши кнопку реєстрації у верхній частині додатку, як зображено на рис. 3.5.

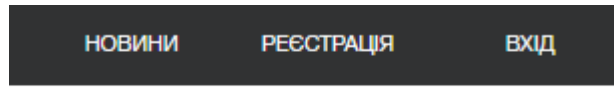


Рисунок 3.5 – Кнопки реєстрації та входу користувача в систему

Після виконання цієї дії користувач зможе заповнити реєстраційну форму, у якій присутні наступні поля: «Ім'я», «Електронна адреса», «Пароль».

По завершенню процесу реєстрації користувач отримає можливість авторизуватися, натиснувши кнопку «Вхід». Після цього одразу з'являється форма входу, як на рис. 3.6 – у відповідні поля необхідно ввести дійсний e-mail і пароль.

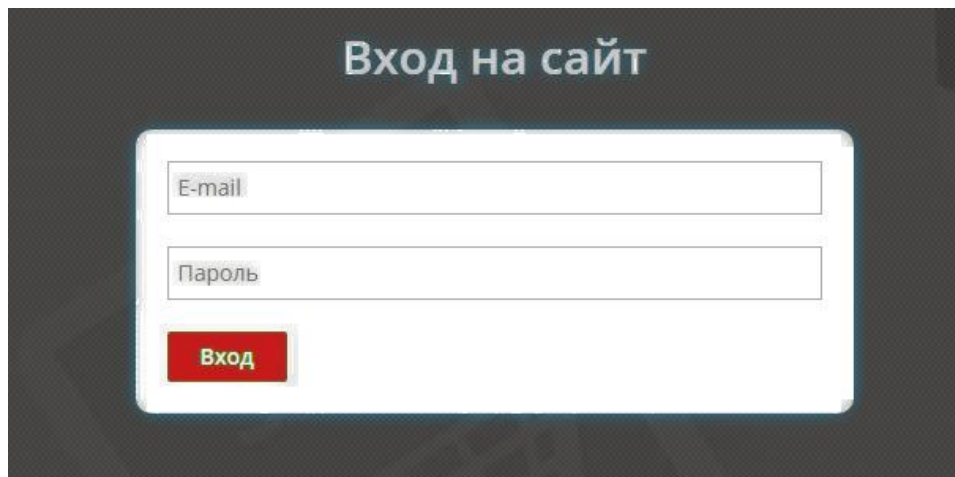


Рисунок 3.6 – Форма авторизації

Система фіксує помилки при авторизації і блокує вхід на сайт, поки не будуть уведені правильні дані. Інакше з'являтиметься попереджувальне повідомлення.

Після входу користувач потрапляє в особистий кабінет. В залежності від ролі («Адміністратор» чи «Юзер») визначаються функції, доступні в особистому кабінеті.

Серверна частина, що відповідає за реєстрацію й авторизацію, складається з контролера UserController і моделі User. У контролері присутні дії (actions), які отримують дані з форм реєстрації / авторизації, а потім звертаються до моделі User, щоб перевірити поля на валідність. У разі відсутності помилок модель User виконує звернення до бази даних і передає їй інформацію, якщо це реєстрація нового акаунту; у випадку аутентифікації вона, навпаки, отримує дані користувача з бази даних і запам'ятовує користувачський id в сесії.

### 3.4. Адміністрування додатку

Система спроектована таким чином, що всі основні опції керування нею виконуються за допомогою панелі адміністрування. Адміністраторська панель допомагає користувачу із роллю «Адміністратор» додавати, видаляти й редагувати види транспорту для оренду й додавати статті й новини, не працюючи з кодом, а через спеціальний інтерфейс, який значно спрощує роботу з додатком. Блок-схема панелі адміністрування зображена на рис. 3.7.

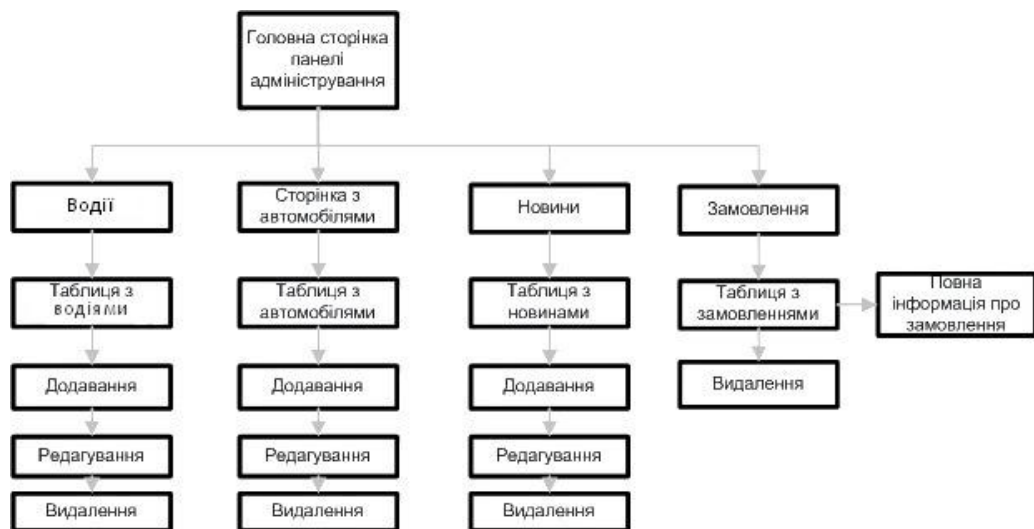


Рисунок 3.7 – Блок-схема адміністраторської панелі

Меню в панелі адміністрування має вигляд списку для комфортного користування й ефективного керування системою (рис.3.8).



Рисунок 3.8 – Панель адміністрування системи

Таким чином, адміністратор може змінювати перелік автомобілів, доступних для замовлення, а також редагувати їх і додавати / корегувати опис – ця дія виконується через пункт меню «Автомобілі», де після натискання відкривається відповідна таблиця, як на рис. 3.9.

↗ На сайт

**ПАНЕЛЬ АДМІНІСТРАТОРА -> АВТОМОБІЛІ**

+ Додати елемент

ID товару	Артикул	Назва автомобіля	Ціна за добу		
1	242542	MAZDA 3	850	✎	✖
2	432523	FORD FOCUS	850	✎	✖
3	532667	KIA RIO 2012	900	✎	✖
4	878430	HYUNDAI SONATA 2011	500	✎	✖
5	942134	VOLKSWAGEN JETTA 2015	1000	✎	✖
6	532341	MITSUBISHI OUTLANDER XL	900	✎	✖
7	325235	MAZDA 6 NEW 2016	850	✎	✖
8	553452	MAZDA CX-5 2014	700	✎	✖
9	675432	BMW X5 F15 2015	950	✎	✖
10	172349	SKODA OCTAVIA A5	800	✎	✖

Рисунок 3.9 – Таблиця для керування списком автомобілів

Другий пункт – «Додати елемент», він відкриває нову сторінку, де адміністратор може поповнити список новою моделлю, доступною для оренди. Усі внесені зміни буде видно на сторінці каталогу автомобілів.

Редагування здійснюється через таблицю адміністрування, де присутня позначка редагування навпроти кожного запису. Адміністратор в може змінювати дані про автівку, такі як вартість, категорія, властивості і так далі (рис. 3.10).

**ПАНЕЛЬ АДМІНІСТРАТОРА -> АВТОМОБІЛІ -> РЕДАГУВАННЯ ЕЛЕМЕНТУ**

Назва товару:

Ціна за добу:

Категорія

Колір:

Ємність баку:

Рисунок 3.10 – Редагування елементу таблиці

Позначка у вигляді хрестика в таблиці, відповідно, дозволяє видалити модель. Але ця дія вимагає підтвердження – для цього система запрошує згоду на видалення. Після підтвердження модель зникне із списку і буде недоступна для подальших замовлень.

Панель адміністрування гарантує зручність роботи із додатком для користувачів із роллю «Адміністратор» – завдяки спроектованій системи можна швидко додавати нові записи і працювати із замовленнями.

Згідно з принципом MVC-моделі, кожен пункт меню має свій окремий контроллер, який, в свою чергу, має методи, що йому належать, – а вони викликають модель відповідно до запиту і здійснюють запит, що поступив від адміністратора. З точки зору керування це зручно, оскільки робота з кодом не є необхідною – дані поділені на файли із окремими назвами.



### 3.5. Сторінка з каталогом автомобілів

Мета додатку – дозволити користувачеві зручно обирати автомобіль для оренди через інтуїтивно зрозумілий дизайн, який дозволить би розрізняти й обирати поміж різних виробників і моделей та детально ознайомлюватися із характеристиками. Вигляд даної сторінки демонструється на рис. 3.11.

#### МАШИНЫ СЕВЕРОДОНЕЦКА

BMW  
DAEWOO  
FORD  
HONDA  
HYUNDAI  
KIA  
MAZDA  
MITSUBISHI  
SKODA  
TOYOTA  
VOLKSWAGEN

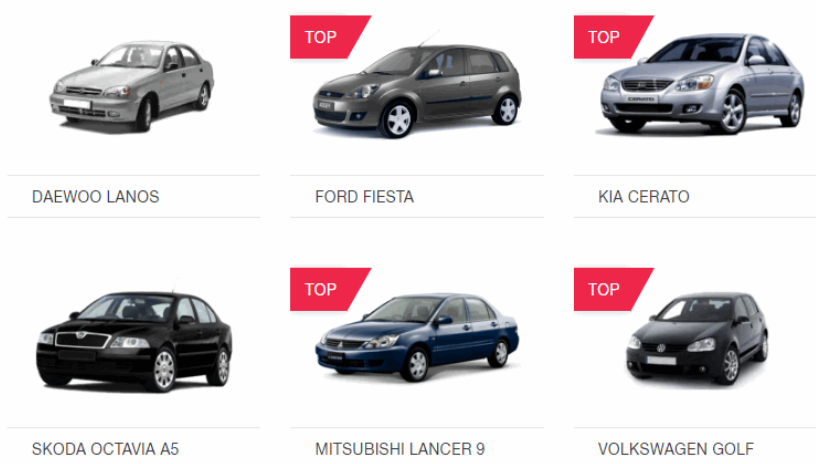
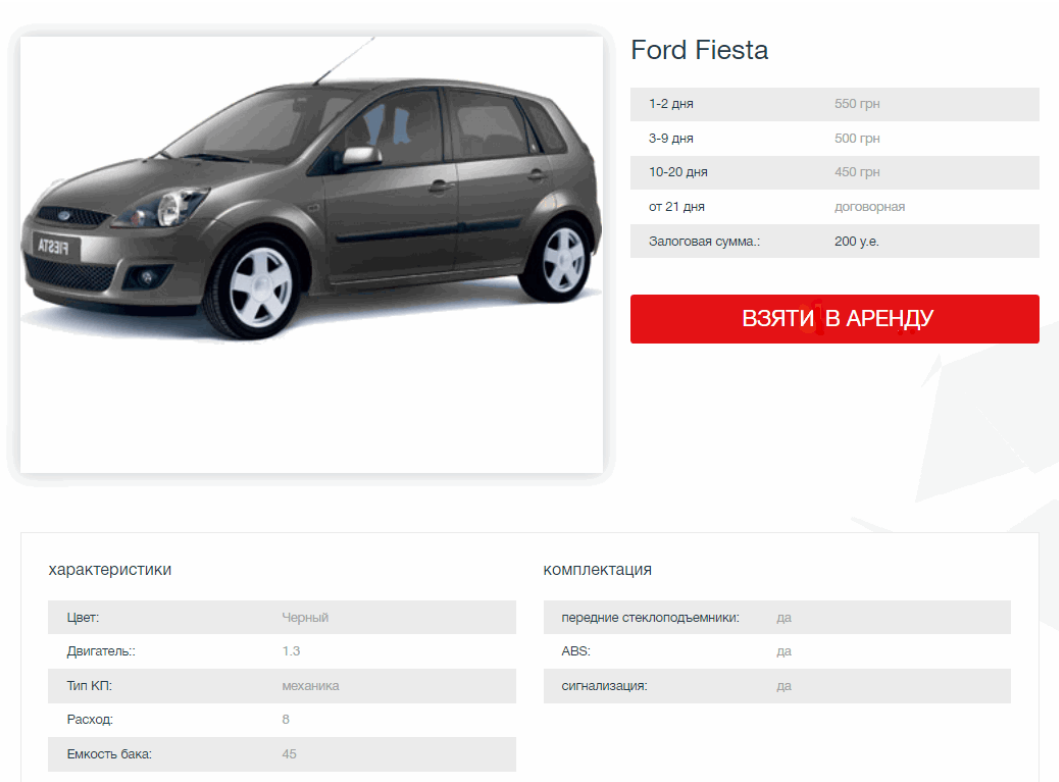


Рисунок 3.11 – Сторінка каталогу автомобілів

Щоб відкрити дані про авто, користувачу треба натиснути на зображення або назву автомобіля в каталозі. На новій сторінці будуть показані детальні дані про дану модель, як показано на рис. 3.12.



**Ford Fiesta**

1-2 дня	550 грн
3-9 дня	500 грн
10-20 дня	450 грн
от 21 дня	договорная
Залоговая сумма.:	200 у.е.

**ВЗЯТИ В АРЕНДУ**

характеристики		комплектация	
Цвет:	Черный	передние стеклоподъемники:	да
Двигатель.:	1.3	ABS:	да
Тип КП:	механика	сигнализация:	да
Расход:	8		
Емкость бака:	45		

Рисунок 3.12 – Розділ з інформацією про авто

Кнопка «Взяти в оренду» створює нове замовлення, але для його підтвердження клієнту потрібно буде особисто оформити замовлення у фірми, яка надає послуги, і вже на місці отримати в оренду автомобіль.

Сторінка із зворотнім зв'язком (рис. 3.13) дозволяє клієнту надсилати свої особисті пропозиції стосовно послуг або поставити питання в письмовому вигляді. Цей же розділ відображає контактні дані й адресу установи.

**КОНТАКТИ**





<ul style="list-style-type: none"> <li> м. Северодонецьк, вул. Федоренка, 37</li> <li> пн-нд 9.00 - 20.00 послуги в неробочий час за домовленістю</li> <li> office@serv-car.com.ua</li> <li> +380 (97) 899 00 87 +380 (95) 899 00 87 +380 (93) 899 00 87</li> </ul>	<p>Имя <input type="text"/></p> <p>Телефон <input type="text"/></p> <p>e-mail <input type="text"/></p> <p>Ваше Сообщение <input type="text"/></p> <p style="text-align: center;"><b>ОТПРАВИТЬ</b></p>
---	---

Рисунок 3.13 – Зворотній зв'язок і контактні дані фірми

Сторінка зворотнього зв'язку працює наступним чином: SiteController отримує дані з полів «E-mail» та «Ваше повідомлення». Якщо адреса набрана правильно, адміністратору надсилається електронний лист.

### 3.6 Сторінка з водіями

Фірма з автопрокату пропонує, окрім оренди автомобілів, замовлення послуг водія. В залежності від потреб клієнта, можна обрати водіїв різних категорій. Сторінка із списком водіїв, показана на рис. 3.14 подібна до каталогу автомобілів і представляє собою блок із вибором водійської категорії та безпосередньо зображення осіб, що пропонують водійські послуги.

#### ВОДІЇ СЄВЕРОДОНЕЦЬКА

Категорія А  
Категорія В  
Категорія С  
Категорія D



Ю. АСТАХОВ



В. БОЯРЧУК



К. ТОМЕНКО



А. ЖУРАВЛЬОВ



В. ЛЕСНОВ



Д. ТАТАРЧЕНКО

Рисунок 3.14 – Сторінка із переліком водіїв

У панелі адміністрування також доступний елемент для редагування списку водіїв. Перейшовши до нього, адміністратор зможе, як і в випадку з каталогом автомобілів, додавати нових осіб і змінювати їх дані.

### 3.7. Картка замовлення

Картка замовлення (рис. 3.15) є елементом додатку, що нагадує кошик у традиційних інтернет магазинах. Клієнт потрапляє на неї після того, як натиснув на кнопку «Взяти в оренду» (для автомобілів) або «Замовити послугу» (для водіїв).

КАРТКА ЗАМОВЛЕННЯ				
Ви зробили наступний вибір:				
Код товару	Назва	Вартість, грн	Термін	Видалити
942134	Аренда: KIA CERATO	900	11.05.19 - 12.05.19	✘
532341	Водій: Д. ТАТАРЧЕНКО	800	13.05.19 - 13.05.19	✘
Загальна вартість, грн				1700
<a href="#">Оформити</a>				

Рисунок 3.15 – Картка замовлення послуг

Всередині картки можливо передивитися перелік обраних послуг, загальну їх вартість і, якщо клієнт передумає, видалити їх зі списку. Після натискання на “Оформити” з'явиться форма для оформлення замовлення (рис.3.16).

**ОФОРМЛЕННЯ ЗАМОВЛЕННЯ**  
Обрано послуг: 2, на суму: 1700 грн.

Ваше ім'я:

Номер телефону:

Коментар:

[Підтвердити](#)

Рисунок 3.16 – Форма підтвердження замовлення

Наведена на рис. 3.17 схема демонструє процес замовлення послуг від початку до кінця.



Рисунок 3.17 – Блок-схема процесу замовлення послуг

## **4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ**

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

Завданням даної роботи бакалавра було спроектувати систему керування продажами комп'ютерних компонент в мережі Інтернет, і як результат було створено повноцінний проект, котрий можна використовувати у розробці такої системи. За цим проектом в подальшому розроблятиметься реальна система, яка значно полегшить процес продажу комп'ютерних компонент у мережі Інтернет. Так як в процесі проектування використовувалося спеціалізоване приміщення, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблятися розроблена система.

### **4.1 Загальні питання з охорони праці**

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

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика,

електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

#### **4.1.1 Правові та організаційні основи охорони праці**

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

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

#### **4.1.2 Організаційно-технічні заходи з безпеки праці**

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511 [8].

Також впроваджені організаційні заходи з пожежної безпеки – навчання і перевірку знань відповідно до вимог Типового положення про інструктажі, спеціальне навчання та перевірку знань з питань пожежної безпеки на підприємствах, в установах та організаціях України, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 29.09.2003 N 368, зареєстрованого в Міністерстві юстиції України 11.12.2003 за N 1148/8469 [9].



## 4.2 Аналіз стану умов праці

Робота над створенням системи керування продажами комп'ютерних компонент в мережі Інтернет проходитиме в приміщенні приватного будинку. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

### 4.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в табл. 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	5
Ширина, м	5
Висота, м	3
Площа, м <sup>2</sup>	25
Об'єм, м <sup>3</sup>	75

Згідно з [10] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм – не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

### 4.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за [11] (табл. 4.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 □ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 □ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 □ 800

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом.

#### 4.2.3 Навантаження та напруженість процесу праці

За фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

### **4.3 Виробнича санітарія**

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

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

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00.-1.28-10 «Правил охорони праці під час експлуатації електронно-обчислювальних машин», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп'ютери (ПК) та периферійні пристрої. Основними робочими характеристиками персонального комп'ютера є:

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

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
1	2	3	4
<b>фізичні</b>			
- підвищена температура поверхонь обладнання	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[10]
- підвищений рівень шуму на робочому місці	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[14]
- підвищений рівень вібрації	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[14] [15]
- підвищена або знижена вологість повітря	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[10]
- підвищена або знижена рухливість повітря	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	1	[10]
- підвищений рівень електромагнітного випромінення	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[16]
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	4	[18]
- підвищена напруженість електричного поля	експлуатація ЕОМ, принтерів, сканерів чи/або серверного обладнання для роботи	2	[16]
- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[13]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[13]
- підвищена яскравість світла	порушення умов праці (організації місця праці-налагодження моніторів)	1	[17]

Таблиця 4.3 – Лист 2

<b>хімічні:</b>			
- загазованість повітря робочої зони, яка впливає на організм людини через органи дихання та надає токсичну і канцерогенну дію	від експлуатації сканерів, принтерів для роботи – O <sub>3</sub> , оплавлення електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів, транзисторів й інше в ЕОМ та системах кондиціонування повітря - CO, CO <sub>2</sub> , SO <sub>2</sub> , P <sub>2</sub> O <sub>5</sub> , H <sub>2</sub> S, HCl, H, NH <sub>3</sub> , ClF <sub>3</sub> , F <sub>2</sub> O <sub>2</sub> , F <sub>2</sub> O <sub>3</sub> , SeO <sub>2</sub> , SeF <sub>6</sub> , TeF <sub>6</sub> , COCl <sub>2</sub> , SO <sub>2</sub> F <sub>2</sub> , інш.	3	[14] [12]
<b>психофізіологічні:</b>			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	[11]
- фізичні (статичне –	порушення умов праці робочого часу - безперервна робота)	2	[11]

Робочі місця мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 [11]. За умов роботи з ПК виникають наступні небезпечні та шкідливі чинники: несприятливі мікрокліматичні умови, освітлення, електромагнітні випромінювання, забруднення повітря шкідливими речовинами (джерелом, яких можуть бути: принтер, сканер та інші джерела виділення багатьох хімічних речовин - напр., озону, оксидів азоту та аерозолів високодисперсних частинок тонера), шум, вібрація, електричний струм, електростатичне поле, напруженість трудового процесу та інше.

### 4.3.2 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ЕОМ, вентиляції і кондиціонування. Небезпека загоряння пов'язана з особливістю комп'ютерів - із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. Надійна робота окремих елементів і мікросхем в цілому забезпечується тільки в певних інтервалах температури, вологості і при заданих електричних параметрах. При відхиленні реальних умов експлуатації від розрахункових можуть виникнути пожежонебезпечні ситуації.

Запобігти утворенню горючого середовища (замінити горючі речовини і матеріали на негорючі і важко горючі) не надається технічно можливим. Тому проектом передбачаються способи і засоби запобігання утворення (або внесення) в горюче середовище джерел запалювання, таких як: застосування електроустаткування, відповідної пожежонебезпечної і вибухонебезпечної зонам відповідно до ПУЕ;

Згідно [12] таке приміщення, площею 25 м<sup>2</sup>, відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння.

### 4.3.3 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і

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

#### **4.4 Гігієнічні вимоги до параметрів виробничого середовища**

##### **4.4.1 Мікроклімат**

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт Іа. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають [10].

#### 4.4.2 Освітлення

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

Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працівника на ПЕОМ.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла в світильниках загального освітлення, оскільки люмінесцентні лампи мають високу потужність (80 Вт), тривалий термін служби (до 10000 годин), спектральний складом випромінюваного світла, близький до сонячного. При експлуатації ЕОМ виконується зорова робота IVв розряду точності (середня точність). При цьому нормована освітленість на робочому місці (Ен) рівна 200 лк. Джерелом природного освітлення є сонячне світло.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає [13]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

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

$$S_b = \left( \frac{1}{5} \div \frac{1}{10} \right) \times S_n \quad (4.1)$$



де  $S_b$  – площа віконних прорізів,

$m^2$ ;  $S_n$  – площа підлоги,  $m^2$ .

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

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

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

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників  $n$  виробляється по формулі (4.2):

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

де  $E$  – нормована освітленість робочої поверхні, визначається нормами – 300 лк;  $S$  – освітлювана площа,  $m^2$ ;  $S = 25 \text{ м}^2$ ;

$Z$  – поправочний коефіцієнт світильника ( $Z = 1,15$  для ламп розжарювання та ДРЛ;  $Z = 1,1$  для люмінесцентних ламп) приймаємо рівним 1,1;

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

$U$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

$M$  – число люмінесцентних ламп в світильнику – 2;  $F$  – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 * 25 * 1,1 * 1,5}{5400 * 0,575 * 2} \approx 2,0$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

#### **4.4.3 Шум та вібрація, електромагнітне випромінювання**

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів (зумовлений як роботою системних блоків, клавіатури, так і друкуванням на принтерах, а також зовнішніми чинниками), коливається у межах 50–65 дБА [14]. Шум такої інтенсивності на тлі високого ступеня напруженості праці негативно впливає на функціональний стан користувачів. Тому на практиці рекомендують знижувати фактичний рівень шуму у приміщеннях, де створюють комп'ютерні програми, виконують теоретичні та творчі роботи, проводять навчання до 40 дБА, а в приміщеннях, де виконують роботу, що потребує зосередженості, — до 55 дБА. У залах опрацювання інформації та комп'ютерного набору рівні шуму не повинні перевищувати 65 дБА.

#### **4.4.4 Вентилювання**

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти), тобто при  $V$  приміщення  $> 40 \text{ м}^3$  на одного працюючого допускається природна вентиляція. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП.

Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

#### **4.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій**

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання наводимо приклади деяких заходів безпеки.

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

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;
- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;
- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала 2/3 нормальної освітленості приміщення);
- облаштовуючи приміщення для роботи з ПК, потрібно передбачити припливно-витяжну вентиляцію або кондиціонування повітря:
- зниження рівня шуму та вібрації:

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

- постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;
- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;
- не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;

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

- не залишати включені електроприлади без нагляду;

- не допускати потрапляння всередину електроприладів крізь вентиляційні отвори рідин або металевих предметів, а також не закривати їх та підтримувати в належній чистоті, щоб уникнути перегрівання та займання приладу;

- не ставити на електроприлади матеріали, які можуть під дією теплоти, що виділяється, загорітися (канцелярські товари, сувенірну продукцію тощо).

Вимоги безпеки при надзвичайних ситуаціях:

1) При раптовому припиненні подачі електричної енергії вимкнути всі пристрої ПК в такій послідовності: периферійні пристрої, ВДТ, системний блок, стабілізатор (або блок безперервного живлення). Витягнути вилки з розеток. При наявності ознак горіння (дим, запах горілого) необхідно вимкнути всі пристрої ПК, знайти місце загоряння і виконати всі можливі заходи для його ліквідації, попередивши терміново про це керівництво. У випадку виникнення пожежі негайно попередити про це пожежну частину та керівництво, виконати усі можливі заходи по евакуації людей з приміщення і розпочати гасіння пожежі первинними засобами пожежогашіння.

2) При замиканні, перевантаженні електричного струму на електричному обладнанні, внаслідок ураження грозової блискавки та ймовірної небезпеки ураженням електричним струмом, приймають наступне:

- попередження замикання здійснюється правильним вибором, монтажем експлуатації мереж;

- застосування захисту схем у вигляді швидкодіючих реле, а також вимикачів, плавких запобіжників, автоматичних вимикачів.

Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом [15], приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контура заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача  $\eta$  – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами.

Послідовність розрахунку.

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

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

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

Підставивши числові значення у формулу (4.3), отримуємо:

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

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

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

$$\rho_{розр.} = \Psi * \rho \quad (4.4)$$

де  $\psi$  – коефіцієнт сезонності для вертикальних заземлювачів і кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів  $\rho_{\text{розр.в}} = 1,7$  і горизонтальних  $\rho_{\text{розр.г}} = 5,5$  Ом · м.

$$\rho_{\text{розр.в}} = 1,7 * 40 = 680 \text{ м} \cdot \text{м}$$

$$\rho_{\text{розр.г}} = 5,5 * 40 = 2200 \text{ м} \cdot \text{м}$$

4) Розраховується опір розтікання струму вертикального заземлювача  $R_{\text{в}}$ , Ом, за (4.5).

$$R_{\text{в}} = \frac{\rho_{\text{розр.в}}}{2 * \pi * l_{\text{в}}} * \left( \ln \frac{2 * l_{\text{в}}}{d_{\text{ст}}} + \frac{1}{2} * \ln \frac{4 * t + l_{\text{в}}}{4 * t - l_{\text{в}}} \right) \quad (4.5)$$

де  $l_{\text{в}}$  – довжина вертикального заземлювача (для труб – 2–3 м;  $l_{\text{в}}=3$  м);  $d_{\text{ст}}$  – діаметр стержня (для труб – 0,03–0,05 м;  $d_{\text{ст}}=0,05$  м);

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

$$t = h_{\text{в}} + \frac{l_{\text{в}}}{2} \quad 4.6)$$

де  $h_{\text{в}}$  – глибина закладання вертикальних заземлювачів (0,8 м); тоді  $t$  0,8 м

$$R_{\text{в}} = \frac{68}{2 * \pi * 3} * \left( \ln \frac{2 * 3}{0,05} + \frac{1}{2} * \ln \frac{4 * 2,3 + 3}{4 * 2,3 - 3} \right) = 18,50 \text{ м}$$

5) Визначається теоретична кількість вертикальних заземлювачів  $n$  штук, без урахування коефіцієнта використання  $\eta_{\text{в}}$ :

$$n = \frac{2 \cdot R_B}{R_d} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (4.7)$$

І визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки  $\eta_B = 0,57$  (табличне значення).

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

$$n_B = \frac{2 \cdot R_B}{R_d \cdot \eta_B} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (4.8)$$

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

$$l_c = 1,05 \cdot L_B \cdot (n_B - 1) \quad (4.9)$$

де  $L_B$  – відстань між вертикальними заземлювачами, (прийняти за  $L_B = 3$  м);  
пв – необхідна кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ м}$$

8) Визначається опір розтіканню струму горизонтального заземлювача (з'єднувальної стрічки)  $R_\Gamma$ , Ом:

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

де  $d_{\text{см}}$  – еквівалентний діаметр смуги шириною  $b$ ,  $d_{\text{см}} = 0,95b$ ,  $b = 0,15$  м;

$h$  – глибина закладання горизонтальних заземлювачів (0,5 м);

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

$$R_\Gamma = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48}{0,95 \cdot 0,15 \cdot 0,5} = 8,10 \text{ Ом}$$

- 9) Визначається коефіцієнт використання горизонтального заземлювача  $\eta_c$  відповідно до необхідної кількості вертикальних заземлювачів  $n_B$ . Коефіцієнт використання з'єднувальної смуги  $\eta_c=0,3$  (табличне значення).
- 10) Розраховується результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_B * R_{\Gamma}}{R_B * \eta_c + R_{\Gamma} * n_B * \eta_B} \leq R_d \quad (4.11)$$

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

$$R_{\text{заг}} = \frac{18,5 * 8,1}{18,5 * 0,3 + 8,1 * 16 * 0,57} = 1,9 \leq R_d$$

Важливу увагу слід звернути на пожежну безпеку підприємства в цілому і окремих його приміщень. В приміщеннях не повинно накопичуватися сміття, непотрібний папір, мотлох та ін. речі, які не використовуються у виробничому процесі. Наявний вільний аварійний вихід за межі приміщення в разі пожежі, бути передбачені вогнегасники. Вони повинні бути в робочому стані і перевірятися згідно з нормами. У приміщеннях повинна бути пожежна сигналізація, вогнегасник.

У разі виникнення пожежі необхідно повідомити в найближчу пожежну частину, убезпечити інших працівників і по можливості прийняти кроки по запобіганню можливих наслідків та усуненню пожежі.



## ВИСНОВКИ

В дипломному проєкті з використанням мов JavaScript та PHP був розроблений веб-додаток для оренди автомобілів й замовлення водійських послуг через мережу Інтернет. Розроблена система задовольняє всім вимогам технічного завдання.

Під час роботи над дипломним проєктом було проведено порівняння платформ автоматизації функціонування звичайних закладів та інтернет-майданчиків з продажу послуг. За багатьма параметрами інтернет-платформи перевершують свої традиційні аналоги.

Був розроблений дизайн системи, що включає реєстрацію, авторизацію, сторінку вибору автомобілів, сторінку вибору водіїв, сторінку зворотнього зв'язку, картки замовлення. Створення системи підтримувалось обраними розробником технологіями та мовами програмування: HTML, CSS, JavaScript, PHP, MySQL.

Була розроблена загальна структура системи, на її основі сформований зміст кожної сторінки додатку. База зареєстрованих осіб поділена на два рівні – з правами користувача й адміністратора.

Розроблена адміністраторська панель дозволяє продавцеві зручно керувати процесами, що відбуваються в системі. Для цього була застосована мова програмування PHP та MVC-модель.

Була розроблена структура бази даних і сторінки в панелі адміністрування, які дозволяють продавцеві редагувати базу даних без необхідності працювати безпосередньо з кодом.

В розділі «Охорона праці та безпека в надзвичайних ситуаціях» був зроблений аналіз шкідливих виробничих факторів, запропоновані заходи щодо техніки безпеки, виробничої санітарії і гігієни праці.

## ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Web-дизайн. Справочник. - Дженнифер Нидерст Роббинс, 2008.
2. HTML & CSS: Проектирование и дизайн веб-сайтов - Джон Дакетт, 2011.
3. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера - Николай Прохоренко, Владимир Дронов, 2015.
4. Д. В. Котеров, А. Ф. Костарев. PHP 5, -СПб.: БХВ Петербург, 2008.
5. PHP и MySQL. Исчерпывающее руководство - Бретт Маклафлин, 2014.
6. НПАОП 0.00-4.12-05 Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці.
7. НПАОП Б.02.005-2003 Про інструктаж, з питань пожежної безпеки.
8. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих.
9. НАПБ Б.03.002-2007 Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою.
10. ДБН В.2.5-28:2015 Природне і штучне освітлення.
11. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку.
12. НПАОП 40.1-1.01-97 Правила безпечної експлуатації електроустановок.
13. ДСТУ ГОСТ 12.1.012-90 ССБТ. Вибрационная безопасность.
14. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот.
15. ДСанПіН 3.3.2.007-98 Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.
16. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление.

## ДОДАТОК А

### Лістинги програмних файлів

#### Файл Db.php

```
<?php
/**
 * Клас Db
 * Компонент для роботи з базою даних
 */
class Db
{
    /**
     * Встановлюємо з'єднання з базою даних
     * @return \PDO <p>Об'єкт класу PDO для роботи з базою даних</p>
     */
    public static function getConnection()
    {
        // Отримуємо параметри єднання з файла скорочення
        $paramsPath = ROOT . '/config/db_params.php';
        $params = include($paramsPath);
        // Встановлення з'єднання
        $dsn = "mysql:host={$params['host']};dbname={$params['dbname']}";
        $db = new PDO($dsn, $params['user'], $params['password']);
        // Задати кодування
        $db->exec("set names utf8");
        return $db;
    }
}
```

## Файл сторінки з переліком автомобілів index.php:

```

<?php include ROOT.'./views/layouts/header.php'; ?>
<div class="container">
<div class="OptionAuto">
    <div class="AutoBrand">
        <h3>Автомобілі м. Сєвєродонецьк</h3>
        <ul>
            <?php foreach ($categories as $categoryItem): ?>
                <li>
                    <a href="/category/<?php echo $categoryItem['id'];?>">
                        <?php echo $categoryItem['name'];?>
                    </a>
                </li>
            <?php endforeach; ?>
        </ul>
    </div>
</div>
<div class="autosList">
    <?php foreach ($lateProducts as $product): ?>
        <div class="autosBlock">
            <a href="/product/<?php echo $product['id'];?>">
                </a>
            <div class="autosInfoContant">
                <a href="/product/<?php echo $product['id'];?>">
                    <?php echo $product['name'];?>
                </a>
                <span><?php echo $product['price'];?>$</span>
                <p><?php echo $product['body'];?></p>
            </div>
        </div>
    </?php endforeach; ?>
</div>

```

```

        <p><?php echo $product['info'];?></p>
    </div>
</div>
<?php endforeach; ?>
</div>
</div>
<!-- Кінець контентного блоку-->
</div>
<?php include ROOT.'views/layouts/footer.php'; ?>

```

### **Файл сторінки категорій автомобілів category.php:**

```

<?php include ROOT.'views/layouts/header.php'; ?>
<div class="container">
    <div class="OptionAuto">
        <div class="AutoBrand">
            <h3>Оберіть марку автівки</h3>
            <ul>
                <?php foreach ($categories as $categoryElem): ?>
                    <li>
                        <a href="/category/<?php echo $categoryElem['id'];?>"
                            class="<?php if ($categoryId == $categoryItem['id']) echo
                            'active';?>">
                            <?php echo $categoryElem['nazva'];?>
                        </a>
                    </li>
                <?php endforeach; ?>
            </ul>
        </div>
    </div>
</div>

```

```

<div class="autosList">
    <?php foreach ($categoryProducts as $products): ?>
        <div class="autosBlock">
            <a href="/product/<?php echo $products['id'];?>">
                </a>
                <div class="autosInfoContant">
                    <a href="/product/<?php echo $products['id'];?>">
                        <?php echo $products['nazva']?>
                    </a>
                    <span><?php echo $products['price'];?>$</span>
                    <p><?php echo $products['body'];?></p>
                    <p><?php echo $products['info'];?></p>
                </div>
            </div>
        <?php endforeach; ?>
        <?php echo $pagination->get(); ?>
    </div>
</div>
<!-- Контентний блок закінується-->
</div>
<?php include ROOT.'./views/layouts/footer.php'; ?>

```

### Файл сторінки адмін-панелі index.php:

```

<?php include ROOT.'./views/layouts/headerAdmin.php'; ?>
<ul class="adminInfo">
    <li><a href="/admin/Auto"><i class="fa fa-circle">
        Автомобілі</i></a></li>

```

```

<li><a href="/admin/categoryAutos"><i class="fa fa-circle">
Категорії автомобилів</i></a></li>
<li><a href="/admin/Driver"><i class="fa fa-circle">
Водії</i></a></li>
<li><a href="/admin/categoryDriver"><i class="fa fa-circle">
Категорії водіїв</i></a></li>
<li><a href="/admin/articles"><i class="fa fa-circle">
Запчастини</i></a></li>
<li><a href="/admin/categoryArticles"><i class="fa fa-circle">
Категорії новин</i></a></li>
<li><a href="/admin/orders"><i class="fa fa-circle">
Замовлення</i></a></li>
</ul>
</div>
</div>
<!-- Блок з контентом закінчується-->
</div>
</body>

```

### Файл сторінки додавання автомобіля в систему create.php:

```

<?php include ROOT . '/views/layouts/headerAdmin.php'; ?>
<div class="adminProducts">
<br/>
<div class="breadcrumb">
<ul>
<li><a href="/admin">Адмінпанель</a></li>
<li><a href="/admin/categoryAutos">Керування
категоріями</a></li>
<li class="active">Додати категорію</li>

```

```

        </ul>
    </div>
    <h3>Додати нову категорію</h3>
    <br/>
    //Перевірка на помилки
    <?php if (isset($errors) && is_array($errors)): ?>
        <ul>
            <?php foreach ($errors as $error): ?>
                <li> - <?php echo $error; ?></li>
            <?php endforeach; ?>
        </ul>
    <?php endif; ?>
    <div class="admForm">
        <form action="#" method="post">
            <p>Назва</p>
            <input type="text" name="name" placeholder="" value="">
            <p>Порядковий номер</p>
            <input type="text" name="sort_order" placeholder="" value="">
            <p>Стан</p>
            <select name="state">
                <option value="1" selected="selected">Показується</option>
                <option value="0">Прихована</option>
            </select>
            <br><br>
            <input type="submit" name="submit" class="btn btn-def"
            value="Зберегти">
        </form>
    </div>
</div>

```



### Сторінка видалення елементів із системи delete.php:

```

<?php include ROOT . '/views/layouts/headerAdmin.php'; ?>
<div class="adminProducts">
  <div class="breadcrumb">
    <ul>
      <li><a href="/admin">Адмінпанель</a></li>
      <li><a href="/admin/categoryAuto">Керування
категоріями</a></li>
      <li class="active">Видалити категорію</li>
    </ul>
  </div>
  <h4>Видалити категорію #<?php echo $id; ?></h4>
  <div class="adminForm">
    <p>Ви дійсно хочете видалити цю категорію?</p>
    <form method="post">
      <input type="submit" name="submit" value="Видалити" />
    </form>
  </div>
</div>

```

### Файл таблиці автомобілів index.php:

```

<?php include ROOT . '/views/layouts/headerAdmin.php'; ?>
<div class="adminProduct">
  <div class="breadcrumbs">
    <ul>
      <li><a href="/admin">Панель адміністратора</a></li>
      <li class="active">Керування категоріями</li>

```

```

    </ul>
</div>

<a href="/admin/categoryAutos/create" class="btn btn-def back"><i
class="fa fa-pls"></i> Додати категорію</a>
<h4>Список категорій</h4>
<br/>
<table class="table-bordered table-striped table">
  <tr>
    <th>ID категорії</th>
    <th>Назва категорії</th>
    <th>Порядковий номер</th>
    <th>Редагувати</th>
    <th>Видалити</th>
  </tr>
  <?php foreach ($categoriesList as $category): ?>
    <tr>
      <td><?php echo $category['id']; ?></td>
      <td><?php echo $category['nazva']; ?></td>
      <td><?php echo $category['sort_order']; ?></td>
      <td class="center"><a href="/admin/categoryAuto/update/<?php
echo $category['id']; ?>" title="Редагування"><i class="fa fa-
pencil-square-o"></i></a></td>
      <td class="center"><a href="/admin/categoryAutos/delete/<?php
echo $category['id']; ?>" title="Видалити"><i class="fa fa-
times"></i></a></td>
    </tr> //Закінчення дії
  <?php endforeach; ?>
</table>
</div>

```

**Файл контролера AdminProductController.php:**

```
<? Php
/**
 * Контролер AdminProductController
 * Управління товарами в адмінпанелі
 * /
class AdminProductController extends AdminBase
{

    /**
     * Action для сторінки "Управління товарами"
     * /
    public function actionIndex ()
    {
        // Перевірка доступу
        self :: checkAdmin ();
        // Отримую список товарів
        $ ProductsList = Product :: getProductsList ();
        // Підключаю вид
        require_once (ROOT. '/views/admin_product/index.php');
        return true;
    }
    /**
     * Action для сторінки "Додати товар"
     * /
    public function actionCreate ()
    {
        // Перевірка доступу
```

```
self :: checkAdmin ();

// Отримую список категорій для списку
$ CategoriesList = Category :: getCategoriesListAdmin ();

// Обробка форми
if (isset ($ _ POST [ 'submit'])) {
    // Якщо форма відправлена
    // Отримую дані з форми
    $ Options [ 'name'] = $ _POST [ 'nazva'];
    $ Options [ 'price'] = $ _POST [ 'price'];
    $ Options [ 'category_id'] = $ _POST [ 'category_id'];
    $ Options [ 'brand'] = $ _POST [ 'brand'];
    $ Options [ 'info'] = $ _POST [ 'info'];
    $ Options [ 'engin_type'] = $ _POST [ 'engin_type'];
    $ Options [ 'drive_unit'] = $ _POST [ 'drive_unit'];
    $ Options [ 'colir'] = $ _POST [ 'colir'];
    $ Options [ 'fuel_consumpt'] = $ _POST [ 'fuel_consumpt'];
    $ Options [ 'adresa'] = $ _POST [ 'adresa'];
    $ Options [ 'phone'] = $ _POST [ 'phone'];
    $ Errors = false;
    // валідація полів
    if (! isset ($ options [ 'name']) || empty ($ options [ 'name'])) {
        $ Errors [] = 'Заповніть поля';
    }
    if ($ errors == false) {

// Якщо помилок немає
    // Додаю новий товар
```

```

    $ Id = Product :: createProduct ($ options);

    // Якщо запис додана
    if ($ id) {
        // завантажувалося чи через форму зображення
        if (is_uploaded_file ($ _FILES [ "image" ] [ "tmp_name" ])) {
            // Якщо завантажувалося, перемістити його в потрібну папку
            move_uploaded_file ($ _FILES [ "image" ] [ "tmp_name" ], $ _SERVER [
'DOCUMENT_ROOT']. "/Upload/images/products/{ $id }.jpg");
        }
    };
    // Перенаправлення користувача на сторінку управліннями
товарами
    header ( "Location: / admin / product");
}

// Підключення виду
require_once (ROOT. '/views/admin_product/create.php');
return true;
}

/ **
* Action для сторінки "Редагувати товар"
* /
public function actionUpdate ($ id)
{
    // Перевірка доступу

self :: checkAdmin ();

```

```

// отримання списку категорій для списку
$ CategoriesList = Category :: getCategoriesListAdmin ();

// Отримую дані про конкретність замовлення
$ Product = Product :: getProductById ($ id);

// Обробка форми
if (isset ($ _ POST [ 'submit'])) {
    // Якщо форма відправлена
    // Отримую дані з форми редагування. При необхідності можна
затверджувати значення
    $ Options [ 'nazva'] = $ _POST [ 'nazva'];
    $ Options [ 'price'] = $ _POST [ 'price'];
    $ Options [ 'category_id'] = $ _POST [ 'category_id'];
    $ Options [ 'brand'] = $ _POST [ 'brand'];
    $ Options [ 'info'] = $ _POST [ 'info'];
    $ Options [ 'engin-type'] = $ _POST [ 'engin-type'];
    $ Options [ 'drive_unit'] = $ _POST [ 'drive_unit'];
    $ Options [ 'color'] = $ _POST [ 'color'];
    $ Options [ 'fuel_consumption'] = $ _POST [ 'fuel_consumption'];
    $ Options [ 'adresa'] = $ _POST [ 'adresa'];
    $ Options [ 'phone'] = $ _POST [ 'phone'];
    // сейв змін
    if (Product :: updateProductById ($ id, $ options)) {

        // Якщо запис збережено
        // Перевірка, завантажувалося чи через форму зображення
        if (is_uploaded_file ($ _ FILES [ "image"] [ "tmp_name"])) {

```

```

        // Якщо завантажувалося, перемістити його в потрібну папку
        move_uploaded_file ($ _ FILES [ "image" ] [ "tmp_name" ], $ _ SERVER [
'DOCUMENT_ROOT']. "/upload/images/products/{ $id }.jpg");
    }
}

// Перенаправлення користувача на сторінку управліннями
товарами
    header ( "Location: / admin / product");
}

// Підключення виду
require_once (ROOT. '/views/admin_product/update.php');
return true;
}

/ **
* Action для сторінки "Видалити товар"
* /
public function actionDelete ($ id)
{
    // Перевірка доступу
    self :: checkAdmin ();

    // Обробка форми
    if (isset ($ _ POST [ 'submit' ])) {
// Якщо форма відправлена видалення товару по ід
        Product :: deleteProductById ($ id);
/// Перенаправлення користувача на сторінку управліннями товарами
        header ("Location: / admin / product");
    }
}

```

```

    }
    // підключення виду
    require_once (ROOT. '/views/admin_product/delete.php');
    return true;
}
}

```

### Файл з моделлю Product.php:

```

<?php
class Product
{
    const SHOW_BY_DEFAULT = 6;

    /**
     * Повертає список елементів каталогу
     * @return array <p>Масив з елементами</p>
     */
    public static function getProductList()
    {
        // З'єднання с БД

        $db = Db::getConnection();

        // Отримання і повернення результатів
        $result = $db->query('SELECT id, name, price FROM product ORDER BY
id ASC');
        $productList = array();
        $i = 0;
        while ($row = $result->fetch()) {

```



```

        $productList[$i]['id'] = $row['id'];
        $productList[$i]['nazva'] = $row['nazva'];
        $productList[$i]['price'] = $row['price'];
        $i++;
    }
    return $productList;
}

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

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

        return $row['count'];
    }

/**
 * Видаляє товар із вказаним id
 * @param integer $id <p>id товару</p>
 * @return boolean <p>Результат виконання методу</p>
 */
    public static function deleteProductById($id)
    {
        // З'єднання з БД
        $db = Db::getConnection();

```

```

// Текст запиту до БД
$sql = 'DELETE FROM product WHERE id = :id';

// Отримання і повернення результатів. Використовується підготовлений
запит
$result = $db->prepare($sql);
$result->bindParam(':id', $id, PDO::PARAM_INT);
return $result->execute();
}
/**
 * Додає новий товар
 * @param array $options <p>Масив з інформацією про товар</p>
 * @return integer <p>id доданого в таблицю запису</p>
 */
public static function createProduct($options)
{
    // З'єднання с БД
    $db = Db::getConnection();

    // Текст запиту до БД
    $sql = 'INSERT INTO product '
        . '(nazva, price, category_id, brand, '
        . 'info, engin-type, colir, fuel_consumpt, adresa, phone)'
        . 'VALUES '
        . '(:nazva, :price, :category_id, :brand, :Mileage, :, '
        . ':info, :engin-type, :colir, :fuel_consumpt, :adresa, :phone)';

    // Отримання і повернення результатів. Використовується підготовлений
запит

```

```

$result = $db->prepare($sql);
$result->bindParam(':nazva', $options['nazva'], PDO::PARAM_STR);
$result->bindParam(':price', $options['price'], PDO::PARAM_STR);
$result->bindParam(':category_id', $options['category_id'],
PDO::PARAM_STR);
$result->bindParam(':brand', $options['brand'], PDO::PARAM_INT);
    $result->bindParam(':info', $options['info'], PDO::PARAM_STR);
    $result->bindParam(':engin-type', $options['engine-type'],
PDO::PARAM_INT);
    $result->bindParam(':colir', $options['colir'], PDO::PARAM_INT);
        $result->bindParam(':fuel_consumpt',
Options['fuel_consumpt'], PDO::PARAM_INT);
        $result->bindParam(':adresa', $options['adresa'],
PDO::PARAM_INT);
        $result->bindParam(':phone', $options['phone'],
PDO::PARAM_INT);
    if ($result->execute()) {

        // Якщо запит виконаний успішно, повертаємо id доданої записи
        return $db->lastInsertId();
    }
    // В іншому випадку повертаємо 0
    return 0;
}

/**
 * Редагує товар із заданим id
 */
public static function updateProductById($id, $options)
{

```

```
// З'єднання з БД
$db = Db::getConnection();

// Текст запиту до БД
$sql = "UPDATE product
      SET
          nazva = :nazva,
          price = :price,
          category_id = :category_id,
          brand = :brand,
          info = :info,
          engin-type = :engin-type,
          colir = :colir,

          fuel_consumpt = :fuel_consumpt,
          adresa = :adresa,
          phone = :phone
      WHERE id = :id";

// Отримання і повернення результатів. Використовується підготовлений
запит
$result = $db->prepare($sql);
$result->bindParam(':id', $id, PDO::PARAM_INT);
$result->bindParam(':nazva', $options['nazva'], PDO::PARAM_STR);
$result->bindParam(':price', $options['price'], PDO::PARAM_STR);
$result->bindParam(':category_id', $options['category_id'],
PDO::PARAM_STR);
$result->bindParam(':brand', $options['brand'], PDO::PARAM_INT);
$result->bindParam(':info', $options['info'], PDO::PARAM_STR);
```

```

        $result->bindParam(':engin-type', $options['engin-type'],
PDO::PARAM_INT);

        $result->bindParam(':colir', $options['colir'], PDO::PARAM_INT);
            $result->bindParam(':fuel_consumpt',
$options['fuel_consumpt'], PDO::PARAM_INT);
            $result->bindParam(':adresa', $options['adresa'],
PDO::PARAM_INT);
            $result->bindParam(':phone', $options['phone'],
PDO::PARAM_INT);

        return $result->execute();
    }

    /**
     * Повертає шлях до зображення

     * @param integer $id
     * @return string <p>Шлях до зображення</p>
     */
    public static function getImage($id)
    {
        // Назва зображення-пустушки
        $noImage = 'no-image.png';

        // Шлях до папки з товарами
        $path = '/upload/images/products/';

        // Шлях до зображення товару
        $pathToProductImage = $path . $id . '.jpg';

        if (file_exists($_SERVER['DOCUMENT_ROOT'].$pathToProductImage)) {

```

```
// Якщо зображення для товару існує
// Повертаємо шлях зображення товару
return $pathToProductImage;
}

// Повертаємо шлях порожнього зображення
return $path . $noImage;
}
}
```