

# КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка вебсайту для прокату автомобілів засобами HTML5, CSS3  
Bootstrap5, PHP7.3 та MySQL8.0

Виконав: студент IV курсу, групи СНс-41

спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

(підпис)

Вирста В.М.

(прізвище та ініціали)

Керівник

(підпис)

Млинко Б.Б.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Литвиненко Я.В.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Боднарчук І.О.

(прізвище та ініціали)

Рецензент

(підпис)

Шенгера Н.Я.

(прізвище та ініціали)



використаних технологій. 7. Вибір середовища розробки вебсайту. 8. Перелік інформаційних сутностей. 9. Опис атрибутів інформаційних сутностей. 10. Розробка моделей БД. 11. Розробка графічного інтерфейсу користувача та адміністратора. 12. Розробка серверної частини. 13. Експлуатація вебсайту для сервісу прокату автомобілів. 14. Висновки

#### 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці			

7. Дата видачі завдання \_\_\_\_\_ 23 січня 2023 р.

#### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	23.01.2023	Виконано
2.	Підбір джерел для написання кваліфікаційної роботи	24.01.2023-26.01.2023	Виконано
3.	Опрацювання джерел по темі кваліфікаційної роботи	27.01.2023-31.01.2023	Виконано
4.	Виконання дослідження щодо розробки вебсайту для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 та MySQL8.0.	01.02.2023-07.02.2023	Виконано
	Розроблення вебсайту		
5.	Оформлення розділу «Аналіз предметної області та постановка задачі на розробку»	08.02.2023-09.02.2023	Виконано
6.	Оформлення розділу «Розробка вебсайту для сервісу прокату автомобілів»	10.02.2023-12.02.2023	Виконано
7.	Виконання завдання до підрозділу «Безпека життєдіяльності, основи охорони праці»	05.06.2023-08.06.2023	Виконано
8.	Оформлення кваліфікаційної роботи	09.06.2023-11.06.2023	Виконано
9.	Нормоконтроль	12.06.2023-13.06.2023	Виконано
10.	Перевірка на плагіат		Виконано
11.	Попередній захист кваліфікаційної роботи		Виконано
12.	Захист кваліфікаційної роботи		

Студент

\_\_\_\_\_ (підпис)

Вирста В.М.

\_\_\_\_\_ (прізвище та ініціали)

Керівник роботи

\_\_\_\_\_ (підпис)

Млинко Б.Б.

\_\_\_\_\_ (прізвище та ініціали)

## АНОТАЦІЯ

Розробка вебсайту для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 та MySQL8.0. // Кваліфікаційна робота освітнього рівня «Бакалавр» // Вирста Владислав Михайлович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СНс-41 // Тернопіль, 2023 // С. 41, рис. – 15, табл. – 7, додат. – 6, бібліогр. – 16.

**Ключові слова:** бази даних, вебсайт, HTML5, CSS3, Bootstrap5, PHP7.3, MySQL8.0.

Кваліфікаційна робота присвячена дослідженню та розробці вебсайту для прокату автомобілів.

В першому розділі кваліфікаційної роботи проведено аналіз предметної області. Сформовано вимоги до вебсайту сервісу прокату автомобілів. Виконано пошук актантів. Сформовано таблицю з реєстром варіантів використання вебсайту сервісу прокатів автомобілів. Досліджено та обґрунтовано вибір технологій та середовища розробки для вебсайту прокату автомобілів.

В другому розділі кваліфікаційної роботи спроектовано та розроблено модель даних для сервісу прокатів автомобілів. Розроблено вебсайт для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 та MySQL8.0. Наведено та протестовано інструкцію для експлуатації вебсайту для сервісу прокату автомобілів.

В третьому розділі кваліфікаційної роботи описано тематику критичних станів програмістів. Наведено приклади критичних станів, описано ряд причин, які призводять до критичних станів програмістів. Висвітлено та проаналізовано критичні стани та наслідки, а також, поради для їх усунення. Проаналізовано тематику впливу кольорів робочого середовища на покращення умов праці та підвищення продуктивності програмістів.

## ANNOTATION

Website Development for Car Rent Service by Means of HTML5, CSS3, Bootstrap5, PHP 7.3 and MySQL 8.0 // Qualification work of the educational level "Bachelor" // Vladyslav Mykhailovych Vyrsta // Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SNs-41 // Ternopil, 2023 // P. 41, fig. – 15, tabl. – 7, annexes. – 6, references – 16.

**Keywords:** database, website, HTML5, CSS3, Bootstrap5, PHP7.3, MySQL8.0.

The qualification work is dedicated to the research and development of a car rental website.

The first chapter of the qualification work analyzes the subject area. The requirements for a car rental service website are formed. A search for actants was performed. A table with a register of options for using a car rental service website is formed. The choice of technologies and development environment for a car rental website has been researched and substantiated.

In the second chapter of the qualification work, a data model for a car rental service was designed and developed. A car rental website was developed using HTML5, CSS3, Bootstrap5, PHP7.3, and MySQL8.0. The instruction manual for the operation of the website for car rental service is presented and tested.

The third chapter of the qualification work describes the topic of critical states of programmers. Examples of critical states are given, a number of reasons that lead to critical states of programmers are described. The critical states and their consequences are highlighted and analyzed, as well as tips for their elimination. The influence of room colors on improving working conditions and increasing programmer productivity is analyzed.

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ТЕРМІНІВ

ACID (англ. Atomicity, Consistency, Isolation, Durability) – набір властивостей транзакцій бази даних, які гарантують дійсність даних, незважаючи на помилки, збої живлення та інші неприємності.

CSS (англ. Cascading Style Sheets) – каскадні таблиці стилів.

HTML (англ. HyperText Markup Language) – мова розмітки гіпертексту.

JavaScript – динамічна, об'єктно-орієнтована прототипна мова програмування.

MySQL – вільна система керування реляційними базами даних.

PHP (англ. PHP: Hypertext Preprocessor) – препроцесор гіпертексту.

Visual Studio Code – редактор вихідного коду, створений компанією Microsoft.

БД – база даних.

## ЗМІСТ

ВСТУП .....	9
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ВЕБСАЙТУ ДЛЯ СЕРВІСУ ПРОКАТУ АВТОМОБІЛІВ ....	10
1.1 Аналітичний огляд наявних рішень .....	10
1.1.1 Вебсайт прокату автомобілів Grand Garage.....	10
1.1.2 Вебсайт прокату автомобілів BLS.....	11
1.1.3 Вебсайт прокату автомобілів Race .....	11
1.1.4 Порівняння наявних рішень .....	12
1.2 Формування вимог для сервісу прокату автомобілів .....	13
1.3 Пошук актантів для сервісу прокату автомобілів.....	14
1.4 Варіанти використання сервісу прокату автомобілів.....	15
1.5 Вибір та обґрунтування використовуваних технологій.....	15
1.6 Вибір середовища розробки вебсайту.....	19
1.7 Висновок до першого розділу.....	20
РОЗДІЛ 2. РОЗРОБКА ВЕБСАЙТУ ДЛЯ СЕРВІСУ ПРОКАТУ АВТОМОБІЛІВ .....	21
2.1 Розробка моделей даних сервісу прокату автомобілів .....	21
2.1.1 Перелік інформаційних сутностей та їх опис .....	21
2.1.2 Розробка моделі БД.....	22
2.2 Створення файлової структури вебсайту для сервісу прокату автомобілів	23
2.3 Розробка вебсайту для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 ТА MYSQL8.0 .....	24
2.3.1 Розробка графічного інтерфейсу користувача та адміністратора .....	24
2.3.2 Розробка серверної частини.....	26
2.4 Експлуатація вебсайту для сервісу прокату автомобілів.....	28

2.4.1	Експлуатація користувацької частини вебсайту прокату автомобілів.....	28
2.4.2	Експлуатація адміністративної частини вебсайту прокату автомобілів.....	30
2.5	Висновок до другого розділу .....	32
РОЗДІЛ 3. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ		33
3.1	Критичні стани програміста.....	33
3.2	Вплив кольору на покращення умов праці та підвищення продуктивності програмістів .....	36
3.3	Висновок до третього розділу.....	38
ВИСНОВКИ.....		39
ПЕРЕЛІК ДЖЕРЕЛ.....		40
ДОДАТКИ		



## ВСТУП

**Актуальність теми.** Актуальність оренди та прокату автомобілів, зумовлена кількома факторами. Перший з них – це те, що не всі люди можуть собі дозволити авто, другий – не у всіх є необхідність використовувати авто часто, а на барбекю з сім'єю чи в поїздку хочеться. Ще однією причиною є те, що таким чином, можна регулярно міняти автомобілі, щоб не бути одноманітним.

**Мета і задачі дослідження.** Метою даної кваліфікаційної роботи освітнього рівня «Бакалавр» є популяризація та підвищення якості послуг сервісів прокату автомобілів завдяки створенню сучасного вебсайту. Для досягнення поставленої мети потрібно виконати ряд завдань:

- провести аналіз ринку вебсайтів для прокату автомобілів.
- сформулювати вимоги для сервісу прокату автомобілів.
- обґрунтувати вибір технологій для реалізації вебсайту для прокату автомобілів;
- обґрунтувати вибір середовища розробки для вебсайту прокату автомобілів;
- виконати проєктування та реалізувати БД для сервісу прокату автомобілів.
- розробити графічний інтерфейс для взаємодії з БД для сервісу прокату автомобілів.
- розробити план заходів щодо перенесення на хостинг та розгортання сайту для прокату автомобілів.

**Практичне значення одержаних результатів.** В ході виконання поставлених задач вебсайт для прокату автомобілів допоможе швидко та зручно опрацьовувати запити для потреб клієнтів. Вебсайт для прокату автомобілів буде забезпечувати зручність як і клієнтів, так і адміністраторів.

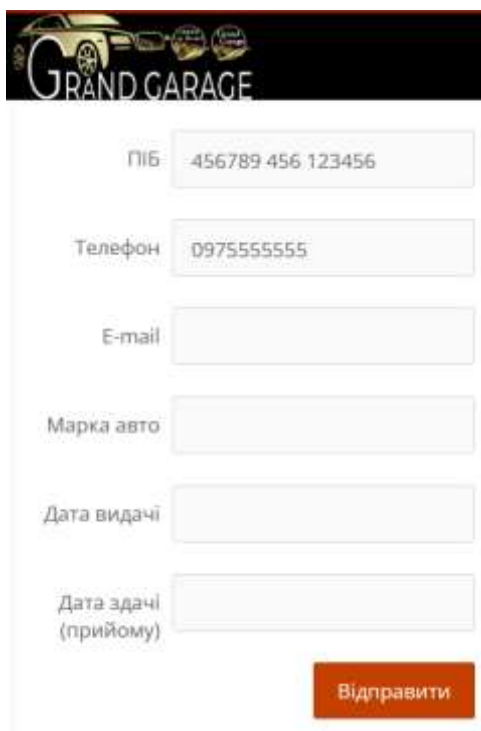
## РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ВЕБСАЙТУ ДЛЯ СЕРВІСУ ПРОКАТУ АВТОМОБІЛІВ

### 1.1 Аналітичний огляд наявних рішень

Ринок вебсайтів для прокату автомобілів є великим і різноманітним. Виділимо кілька існуючих рішень для аналізу постановки завдання.

#### 1.1.1 Вебсайт прокату автомобілів Grand Garage

Серед переваг вебсайту прокату автомобілів Grand Garage можна навести сучасний дизайн, велику палітру різноманітних інструкцій та хорошу оптимізацію під смартфони. Але є очевидний недолік – форми на сайті присутні без різноманітних валідацій, даний мінус наведений на рисунку 1.1 [1].



The image shows a web form for car rental on the Grand Garage website. At the top, there is a header with the 'GRAND GARAGE' logo and a car image. Below the header, the form consists of several input fields: 'ПІБ' (PIB) with the value '-456789-456 123456', 'Телефон' (Telephone) with '0975555555', 'E-mail', 'Марка авто' (Car brand), 'Дата видачі' (Rental date), and 'Дата здачі (прийому)' (Return date). At the bottom right of the form is a red button labeled 'Відправити' (Send).

Рисунок 1.1 – Форма для відправки користувачем замовлення на вебсайті прокату автомобілів Grand Garage

Іще одним мінусом є те, що сайт має низькоякісні зображення та іконки, що відразу кидаються в очі [1].

### 1.1.2 Вебсайт прокату автомобілів BLS

Серед переваг вебсайту прокату автомобілів BLS можна виділити зручну систему пошуку автомобілів. Крім пошуку по днях доступних машин, також даний сайт дозволяє обрати категорію автомобіля. Вікно вибору категорії автомобіля вебсайту прокату автомобілів BLS наведено на рисунку 1.2 [2].



Рисунок 1.2 – Вікно вибору категорії автомобіля вебсайту прокату автомобілів BLS

Серед мінусів можна виділити те, що елементи вебсайту непропорційні між собою, мобільна версія сайту незручна для використання.

### 1.1.3 Вебсайт прокату автомобілів Race

Серед переваг вебсайту прокату автомобілів Race можна навести доволі зручний пошук потрібного автомобіля за наступними категоріями: дати оренди, клас авто, тип палива, трансмісія. Пропрацьована мобільна версія вебсайту, проте не всі основні деталі. Головне меню вебсайту прокату автомобілів Race (мобільна версія) наведено на рисунку 1.3. Воно є доволі незручним під час використання мобільної версії [3].

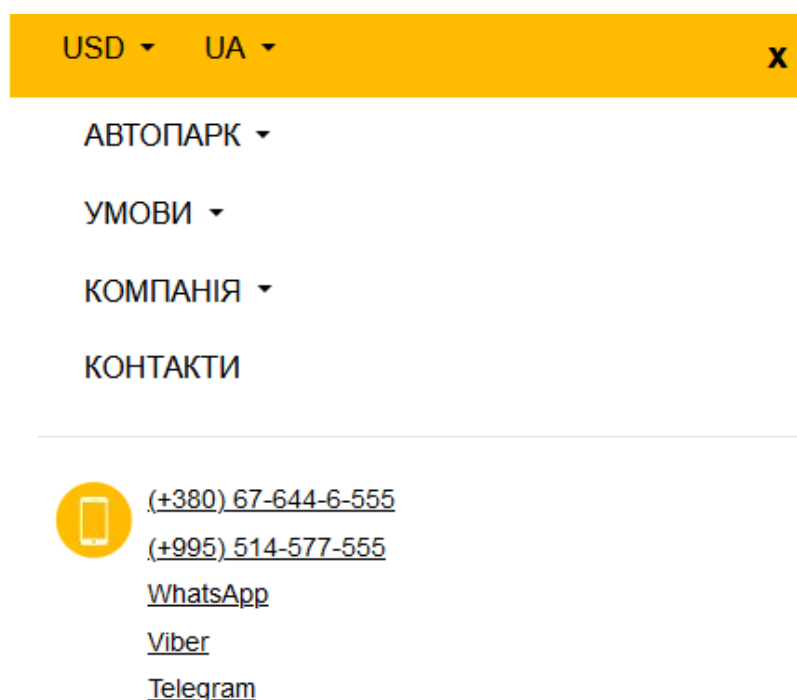


Рисунок 1.3 – Головне меню сайту прокату автомобілів Race (мобільна версія)

Також, доволі добре на сайті прописані умови прокату автомобілів.

#### 1.1.4 Порівняння наявних рішень

Підсумуємо та порівняємо три наявні рішення на ринку вебсайтів прокату автомобілів: Grand Garage, BLS та Race. Порівняння наявних рішень на ринку вебсайтів прокату автомобілів підсумовано і наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння наявних рішень на ринку вебсайтів прокату автомобілів

Параметр для порівняння	Grand Garage	BLC	Race
1	2	3	4
Мобільна оптимізація	Оптимізований	Не до кінця оптимізовано	Не до кінця оптимізовано

## Продовження таблиці 1.1

1	2	3	4
Функціонал	Замовлення авто, перегляд атрибутів авто, відправка форми замовлення, зворотній зв'язок	Замовлення авто, перегляд атрибутів авто, відправка форми замовлення, система пошуку потрібного авто, розрахунок ціни, зворотній зв'язок	Замовлення авто, перегляд атрибутів авто, відправка форми замовлення, система пошуку потрібного авто, розрахунок ціни, зворотній зв'язок, зміна валюти
Дизайн	Низька графіка зображень	Непропорційність елементів сайтів	Без мінусів
Контент	Відео, зображення і іконки	Відео, зображення і іконки	Зображення та іконки
Валідація форм відправлення	Непропрацьована	Без мінусів	Без мінусів
Зручність інтерфейсу	Зручний	Незручна мобільна версія	Незручна мобільна версія

На основі отриманих даних з порівняльної таблиці наявних рішень на ринку вебсайтів прокату автомобілів (див. таблицю 1.1) можна сформулювати список мінімальних вимог для майбутнього сервісу прокату автомобілів

## 1.2 Формування вимог для сервісу прокату автомобілів

В проєктованому вебсайті для сервісу прокату автомобілів повинні бути наступні функціональні можливості:

- реєстрація нових користувачів;

- авторизація користувачів;
- можливість попереднього оформлення оренди;
- можливість користувачу переглянути його попередні замовлення;
- можливість адміністратору добавляти нові позиції автомобілів для оренди;
- можливість адміністратора оновлювати атрибути позицій автомобілів для оренди;
- можливість адміністратора видаляти позиції автомобілів для оренди.
- можливість адміністратора переглядати замовлення;
- можливість адміністратора завершувати замовлення.

Враховуючи функціональні можливості можна сформувати наступні вимоги до захисту:

- введені користувачем дані повинні проходити процес валідації;
- неpubлічна інформація повинна бути закритою для користувачів.

Формулювання вимог – ітеративний процес, який може змінюватися на протязі розробки проєкту [4].

### **1.3 Пошук актантів для сервісу прокату автомобілів**

У вебсайті для сервісу прокату автомобілів будуть наявні такі актори:

- незареєстрований користувач;
- зареєстрований користувач;
- адміністратор;
- користувач.

Кожен з акторів виконує свою роль, і певним чином взаємодіє із іншими акторами. Вони є основними учасниками або сутностями, які взаємодіють в рамках системи [5].

Діаграма акторів системи для сервісу прокату автомобілів наведена на рисунку 1.4.

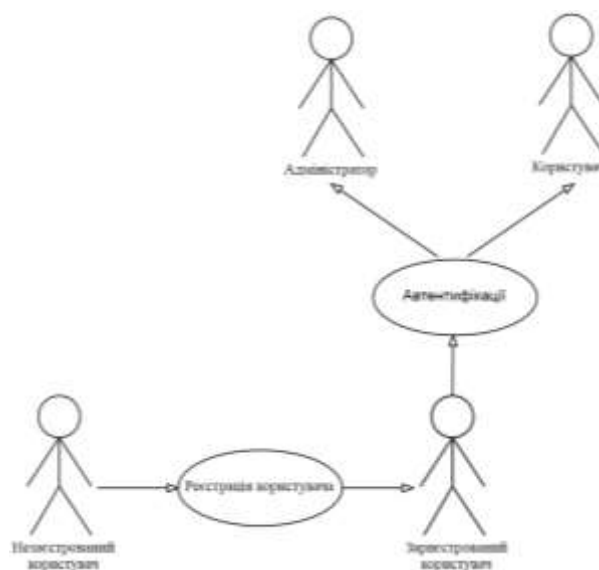


Рисунок 1.4 – Діаграма акторів для вебсайту сервісу прокату автомобілів

В процесі автентифікації користувача буде визначатися, які права матиме користувач вебсайту.

#### 1.4 Варіанти використання сервісу прокату автомобілів

Виходячи із списків вимог до вебсайту прокату автомобілів, сформуємо ключові варіанти використання системи. Їх опис подано у таблиці з реєстром варіантів використання вебсайту сервісу прокатів автомобілів у Додатку А.

Дана таблиця характеризує акторів та їх основний функціонал, найменування та опис [4].

#### 1.5 Вибір та обґрунтування використовуваних технологій

Будуємо порівняльну таблицю для вибору фреймворків для розробки вебінтерфейсів, яка наведена у таблиці 1.2.

Таблиця 1.2 – Порівняльна таблиця фреймворків для розробки вебінтерфейсів

<b>Фреймворк</b>	<b>Bootstrap</b>	<b>Tailwind CSS</b>	<b>Material-UI</b>
Мови програмування	HTML, CSS, JS	CSS	React, JavaScript
Вартість	Безкоштовний	Безкоштовний	Безкоштовний
Резинова сітка	Так	Так	Так
Компоненти	Так	Так	Так
Активне співтовариство	Так	Так	Так
Документація	Так	Так	Так

Дивлячись на порівняльну таблицю фреймворків для розробки вебінтерфейсів (див. таблицю 1.2) можемо побачити, що найпопулярніші фреймворки для розробки вебінтерфейсів схожі за основними характеристиками, відрізняються лише мовами програмування.

Для вибору мови програмування серверної частини вебсайту було створено порівняльну табличку. Вона наведена у таблиці 1.3.

Таблиця 1.3 – Порівняльна таблиця мов програмування серверної частини

<b>Мова програмування</b>	<b>Переваги</b>	<b>Недоліки</b>
<b>1</b>	<b>2</b>	<b>3</b>
JavaScript	Єдина мова для клієнтської та серверної частини	Асинхронна модель може вимагати особливого підходу до програмування, менша швидкодія порівняно зі скомпільованими мовами
Ruby	Зрозумілий синтаксис	Потребує інсталяції Ruby-платформи, порівняно повільніший з аналогами



## Продовження таблиці 1.3

1	2	3
Python	Простий синтаксис, велика кількість сторонніх бібліотек	Невисока швидкодія порівняно зі скомпільованими мовами, гранулярність потребує додаткових конфігурацій
Java	Велика спільнота розробників, висока швидкодія та масштабованість	Висока складність та багато формальностей, потребує великої кількості пам'яті
PHP	Широко використовується у веброботці, велика спільнота	Менша швидкодія порівняно зі скомпільованими мовами, застаріла синтаксична модель

В порівняльній таблиці (див. таблицю 1.3) було наведено кілька мов програмування для написання серверної частини сайту. Кожен із варіантів має свою специфіку.

Для зберігання даних у БД, було створено їх порівняльну таблицю. Вона наведена у таблиці 1.4.

Таблиця 1.4 – Порівняльна таблиця БД

Атрибут порівняння	MySQL	MongoDB	Microsoft SQL Server	SQLite
1	2	3	4	5
Тип	Реляційна	NoSQL	Реляційна	Реляційна
Вартість	Безкоштовна	Безкоштовна	Платна	Безкоштовна

Продовження таблиці 1.4

1	2	3	4	5
Мова запитів	SQL	JSON/Query	SQL	SQL
Підтримка ACID	Так	Ні	Так	Так
Скорочений опис	Популярна база даних з великою спільнотою розробників та великою кількістю доступних інструментів.	Гнучка документ-орієнтована база даних, яка дозволяє зберігати та опрацьовувати дані у вигляді JSON.	Масштабована та високопродуктивна реляційна база даних від Microsoft.	Легка та вбудована база даних, яка не вимагає окремого сервера і підходить для малих проєктів.

З порівняльної таблиці БД (див. таблицю 1.4) можна побачити, що підтримку ACID мають тільки реляційні БД (MySQL, Microsoft SQL Server, SQLite). При виборі основної БД для побудови вебсайту, даний атрибут матиме найбільше значення.

Для вебсайту сервісу прокату автомобілів було обрано наступні технології:

- Bootstrap – це набір відкритих інструментів, який дозволяє легко створювати сучасні компоненти інтерфейсу веб-сторінок, такі як форми, кнопки, навігаційні панелі та інші елементи. Вибраний через сумісність із HTML, CSS, JS [6].
- PHP – це мова програмування, яка призначена для створення скриптів, що генерують HTML-сторінки на сервері. Причиною вибору є те, що дана мова широко використовується у веброзробці та має велике ком'юніті [7].

- MySQL – це система керування базами даних з відкритим кодом, яка відрізняється високою продуктивністю, надійністю і простотою використання. Причиною вибору є те, що дана БД є сумісною із PHP і має підтримку ACID [8].

Дані технології між собою є сумісними.

## 1.6 Вибір середовища розробки вебсайту

Відповідно до вибраних технологій для розробки вебсайту прокату автомобілів, виберемо серед кількох одне найбільш підходяще середовище розробки. Для оптимального вибору середовища для розробки вебсайту прокату автомобілів створимо порівняльну таблицю. Вона наведена у таблиці 1.5.

Таблиця 1.5 – Порівняння наявних середовищ розробки вебсайтів

Характеристики	Visual Studio Code	Sublime Text	Atom	Brackets
Вартість	Безкоштовна	Платна ( є пробна версія)	Безкоштовна	Безкоштовна
Підтримка мов	Багатомовна	Багатомовна	Багатомовна	HTML, CSS, JS
Автодоповнення коду	Так	Так	Так	Так
Вбудоване налагодження	Так	Ні	Ні	Ні
Розширення/плагіни	Так	Так	Так	Так

Для розробки вебсайту для сервісу прокатів автомобілів було обрано середовище Visual Studio Code.

Це безкоштовний засіб для створення, редагування сучасних вебзастосунків і програм [9].

Можливість добавлення розширень, функція автодоповнення коду, а також, досвід написання вебсайтів в даному середовищі розробки відіграли ключову роль при виборі даного середовища розробки.

Одне з розширень, яке буде встановлено, називається PHP Intelephense. Це розширення має такі основні функції, як докладний статичний опис вбудованих функцій, методів і конструкторів, точне згортання коду та інтелектуальне виділення ключових слів та посилань [9].

### **1.7 Висновок до першого розділу**

В першому розділі кваліфікаційної роботи було проаналізовано наявні рішення на ринку вебсайтів для прокату автомобілів за допомогою порівняльної таблиці. Аргументовано доцільність створення вебсайту. Сформовано перелік вимог для сервісу прокату автомобілів. Проведений пошук актантів. Сформований перелік варіантів використання кожного актора.

Вибрано та обґрунтовано технології для написання вебсайту для сервісу прокату автомобілів. Обрано середовище розробки вебсайту.

## РОЗДІЛ 2. РОЗРОБКА ВЕБСАЙТУ ДЛЯ СЕРВІСУ ПРОКАТУ АВТОМОБІЛІВ

### 2.1 Розробка моделей даних сервісу прокату автомобілів

#### 2.1.1 Перелік інформаційних сутностей та їх опис

Формуємо перелік інформаційних сутностей вебсайту сервісу прокату автомобілів. Перелік інформаційних сутностей вебсайту сервісу прокату автомобілів наведений у таблиці 2.1.

Таблиця 2.1 – Перелік інформаційних сутностей вебсайту сервісу прокату автомобілів

Найменування сутності	Призначення
Car	Зберігає в собі інформацію про авто
Customer	Зберігає інформацію про облікові записи
Rent	Зберігає інформацію про замовлення
ReturnCar	Зберігає інформацію про повернення авто

В результаті, вийшло чотири інформаційні сутності: Car, Customer, Rent, ReturnCar.

Відповідно до предметної області, проектуємо таблицю Car, колонки якої і їх характеристики представлені у Додатку Б.1 у форматі таблиці. Таблиця Car описує атрибути автомобіля для прокату. Значення ідентифікатора таблиці CarID є автоінкрементним.

Проектуємо таблицю Customer для вебсайту сервісу прокату автомобілів, атрибути якої наведені у форматі таблиці в Додатку Б.2.

Таблиця Customer описує атрибути користувача вебсайту. Значення ідентифікатора таблиці CustID є автоінкрементним.

Проектуємо таблицю Rent. Колонки таблиці та їх характеристики наведені у Додатку Б.3.

Дана таблиця описує атрибути замовлення. Значення ідентифікатора таблиці RentID є автоінкрементним.

Проектуємо таблицю ReturnCar для вебсайту сервісу прокату автомобілів, яка подана у таблиці Додатку Б.4.

Таблиця описує атрибути повернення авто замовником прокату авто.

### 2.1.2 Розробка моделі БД

Зв'язки між таблицями БД сервісу прокату автомобілів наведені у таблиці 2.2.

Таблиця 2.2 – Опис зв'язків сутностей вебсайту сервісу прокату автомобілів

Зв'язки між таблицями	Тип зв'язку	Опис зв'язку
Car – Rent	1:n	Один автомобіль можуть замовляти n разів
Customer – Rent	1:n	Один користувач може оформити n замовлень
Rent – ReturnCar	1:1	Одне замовлення відповідає одному рядку інформації про дійсне повернення автомобіля

Структурна модель БД вебсайту для сервісу прокату автомобілів наведена у Додатку В.

Структурна модель БД описує організацію даних всередині БД. Вона визначає сутність, їх атрибути і зв'язок між ними [10].

## 2.2 Створення файлової структури вебсайту для сервісу прокату автомобілів

Вебзастосунок для сервісу прокатів автомобілів включає в себе наступні файли:

- index.php – вхідна точка вебсайту;
- connect.php – файл, який реалізує підключення до БД;
- login.php – файл, що містить форму з авторизацією користувача;
- registr.php – файл, що містить форму з реєстрацією користувача;
- process.php – файл, який містить функції, які забезпечують реєстрацію користувача, авторизацію, валідацію та хешування даних.
- adminCar.php – файл, що дозволяє переглянути адміністратору позиції авто, їх характеристики, а також, додати позицію;
- adminRent.php – файл, що дозволяє переглянути адміністратору інформацію про замовлення;
- adminUser.php – файл, що дозволяє переглянути адміністратору інформацію про зареєстрованих користувачів у БД вебсайту;
- carUpdate.php – файл, що містить форму для оновлення позиції автомобіля;
- rentEnd.php – файл, який містить скрипт, який дозволяє адміністратору завершити замовлення;
- userCarPark.php – файл, що дозволяє переглянути користувачу позиції автомобілів;
- userRent.php – файл, що дозволяє переглянути свою історію замовлень користувачу;
- rentCar.php – файл, який містить форму для створення замовлення користувачу;
- delete.php – файл, який містить в собі скрипти, для видалення заданого попередньо рядка;

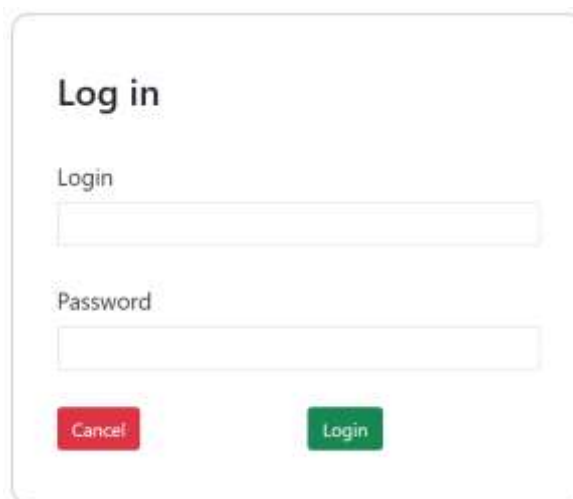
- `logout.php` – файл, що завершує сесію користувача та перенаправляє його на початкову сторінку (`index.php`).

Картинки зображень автомобілів зберігаються в директорії `uploads`.

## 2.3 Розробка вебсайту для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 та MySQL8.0

### 2.3.1 Розробка графічного інтерфейсу користувача та адміністратора

Реалізація HTML розмітки форми авторизації користувача для вебсайту прокату автомобілів наведено у Додатку Г. Зовнішній вигляд форми авторизації користувача для вебсайту прокату автомобілів наведено на рисунку 2.1.



The image shows a login form with the following elements:

- Title: **Log in**
- Label: Login
- Input field for the login name.
- Label: Password
- Input field for the password.
- Buttons: **Cancel** (red) and **Login** (green).

Рисунок 2.1 – Зовнішній вигляд форми авторизації для вебсайту прокату автомобілів

Реалізація HTML розмітки форми реєстрації користувача для вебсайту прокату автомобілів наведено у Додатку Д.

Зовнішній вигляд форми реєстрації користувача для вебсайту прокату автомобілів наведено на рисунку 2.2.



**Sign Up**

First Name  Last Name

Login Name  Add

Phone  e-Mail

Password

Рисунок 2.2 – Зовнішній вигляд форми реєстрації користувача для вебсайту прокату автомобілів

HTML розмітка таблиці позиції автомобілів адміністратором сайту наведена у Додатку Є.

Приклад таблиці позиції автомобілів адміністратором сайту наведений на рисунку 2.3.




Photo	Brand	Model	Price	Color	Status	Action
	Lexus	UX300	95	gray	free	<input type="button" value="Update"/> <input type="button" value="Delete"/>
	Peugeot	301	25	gray	reserve	<input type="button" value="Update"/> <input type="button" value="Delete"/>
	Toyota	Corolla	25	white	free	<input type="button" value="Update"/> <input type="button" value="Delete"/>

Рисунок 2.3 – Зовнішній вигляд сторінки перегляду автомобілів адміністратором сайту

Даний інтерфейс допоможе користувачам більш зручніше користуватися сайтом.

### 2.3.2 Розробка серверної частини

Для автентифікації користувача вебсайту для сервісу прокату автомобілів було написаний код, який наведений у лістингу 2.1.

Лістинг 2.1 – Реалізація автентифікації користувача вебсайту для сервісу прокату автомобілів

```
if($res['CustLoginName']==$loginName &&
$res['CustPassword']==$password){
    $_SESSION['CustID'] = $res['CustID'];
    $_SESSION['CustFirstName'] = $res['CustFirstName'];
}
else{
    echo "The username and password are incorrct";
}
```

Даний код (див. лістинг 2.1) встановлює дві сесії з даними користувача.

Для додавання нового користувача використовується функція вебсайту для сервісу прокату автомобілів, яка наведена у лістингу 2.2.

Лістинг 2.2 – Функція для додавання нового користувача вебсайту для сервісу прокату автомобілів

```
//функція для додавання користувача
function insertDetails($conn, $firstName, $lastName, $loginName,
    $add, $phone, $eMail, $password){
    $query = $conn->prepare("
        INSERT INTO `Customer`(`CustFirstName`, `CustLastName`,
`CustLoginName`, `CustAdd`, `CustPhone`, `CustEmail`,
`CustPassword`, `CustPrivileges`)
        VALUES
(:firstName,:lastName,:loginName,:add,:phone,:eMail,:password,'use
r')
    ");
    $query->bindParam(":firstName", $firstName);
    $query->bindParam(":lastName", $lastName);
    $query->bindParam(":loginName", $loginName);
    $query->bindParam(":add", $add);
    $query->bindParam(":phone", $phone);
    $query->bindParam(":eMail", $eMail);
    $query->bindParam(":password", $password);
    return $query->execute();
}
```

Функція для добавлення нової позиції автомобіля вебсайту для сервісу прокату автомобілів наведена у лістингу 2.3.

Лістинг 2.3 – Функція для добавлення нової позиції автомобіля вебсайту для сервісу прокату автомобілів

```
function insertCar($conn, $brand, $model, $price, $color, $status,
$image) {
    $query = $conn->prepare("
        INSERT INTO `Car` (`Brand`, `Model`, `Price`, `Color`,
`Status`, `Photo`)
        VALUES (:brand, :model, :price, :color, :status, :image)
    ");
    $query->bindParam(":brand", $brand);
    $query->bindParam(":model", $model);
    $query->bindParam(":price", $price);
    $query->bindParam(":color", $color);
    $query->bindParam(":status", $status);
    $query->bindParam(":image", $image);

    return $query->execute();
}
```

Для завершення сесії користувача використовуються наступні два вбудовані методи PHP:

```
session_unset();
session_destroy();
```

Для екранування даних використовується функція, яка представлена у лістингу 2.4.

Лістинг 2.4 – Функція для екранування даних

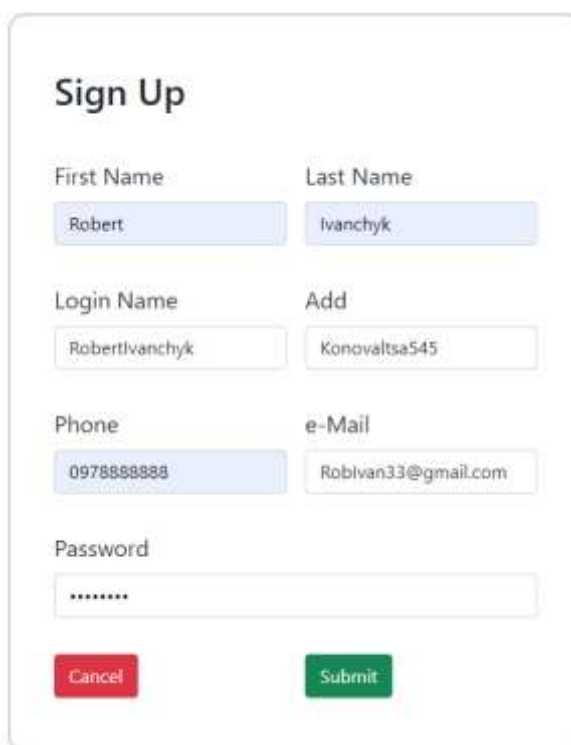
```
function sanitizeString($str) {
    $str = strip_tags($str);
    $str = str_replace("'", "", $str);
    return $str;
}
```

Екранування даних є важливим аспектом забезпечення безпеки і конфіденційності інформації.

## 2.4 Експлуатація вебсайту для сервісу прокату автомобілів

### 2.4.1 Експлуатація користувачької частини вебсайту прокату автомобілів

Для реєстрації на вебсайті прокату автомобілів, потрібно в головному меню вибрати кнопку Registration і заповнити форму. Приклад заповненої форми реєстрації користувача вебсайту прокату автомобілів наведено на рисунку 2.4.



The image shows a 'Sign Up' registration form. It contains the following fields and values:

Field	Value
First Name	Robert
Last Name	Ivanchyk
Login Name	RobertIvanchyk
Add	Konovaltsa545
Phone	0978888888
e-Mail	RobIvan33@gmail.com
Password	*****

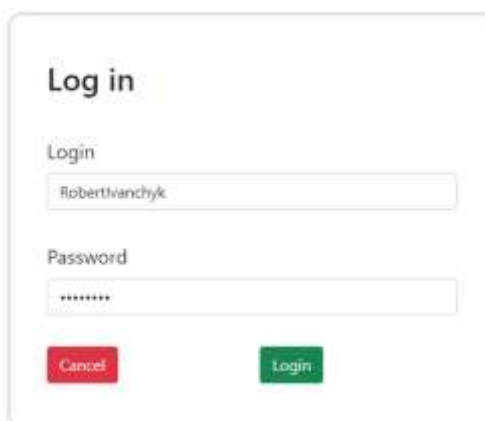
At the bottom of the form, there are two buttons: a red 'Cancel' button and a green 'Submit' button.

Рисунок 2.4 – Приклад заповненої форми реєстрації користувача вебсайту прокату автомобілів

Для відправлення запиту на реєстрацію потрібно натиснути на кнопку Submit. Якщо потрібно відхилити процес створення аккаунту, то необхідно натиснути на кнопку Cancel.

Для авторизації користувача на вебсайті прокату автомобілів в головному меню вебсайту прокату автомобілів потрібно натиснути на кнопку Log in та

ввійти, заповнити дані та натиснути на кнопку Login. Приклад заповненої форми авторизації користувача у вебсайт наведено на рисунку 2.5.



The image shows a 'Log in' form with the following elements:

- Title: Log in
- Field: Login (containing the text 'Robertvanchyk')
- Field: Password (containing seven asterisks '\*\*\*\*\*')
- Buttons: Cancel (red) and Login (green)

Рисунок 2.5 – Приклад заповненої форми авторизації користувача вебсайту прокату автомобілів

Для прокату авто потрібно вибрати машину серед запропонованих, натиснути на кнопку Rent. Далі потрібно заповнити форму та надіслати форму. Вікно вибору авто для його прокату наведено на рисунку 2.6.

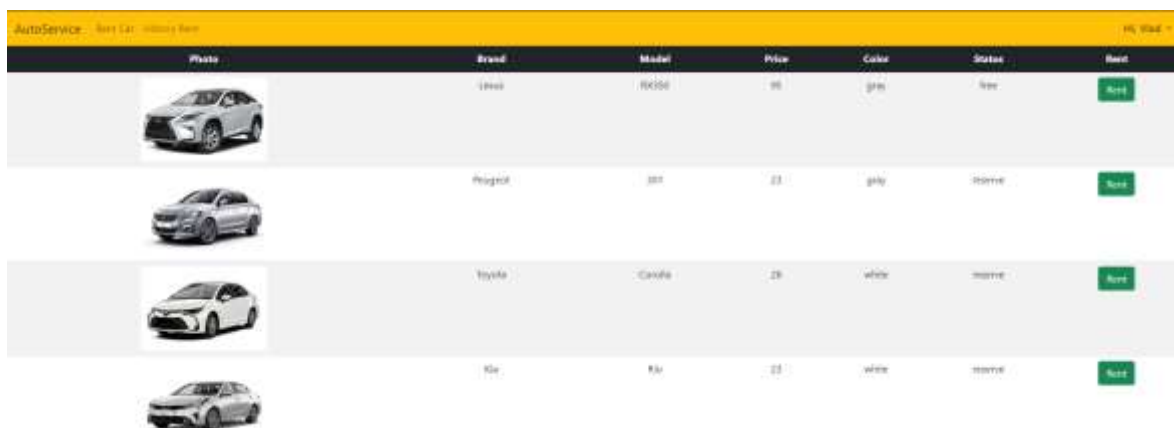






Photo	Brand	Model	Price	Color	Status	Rent
	Hyundai	ix35	26	grey	free	<a href="#">Rent</a>
	Hyundai	ix35	23	grey	none	<a href="#">Rent</a>
	Hyundai	Creta	26	white	none	<a href="#">Rent</a>
	Kia	Kia	23	white	none	<a href="#">Rent</a>

Рисунок 2.6 – Перегляд позицій автомобілів вебсайту прокату автомобілів

Для перегляду історії замовлень, їх статусу, ціни і дат прокату автомобілів потрібно в головному меню програми натиснути на кнопку History Rent. Вікно перегляду історії замовлень користувача вебсайту прокату автомобілів наведено на рисунку 2.7.



Car	RentDate	ReturnDate	Fees	Status
 Lexus RX350	2022-06-12	2022-06-22	231	end rent

Рисунок 2.7 – Перегляд історії замовлень користувача вебсайту прокату автомобілів

Для виходу із вебсайту потрібно натиснути на випадаюче меню, яке знаходиться з правого боку головного меню та натиснути на кнопку Logout.

#### 2.4.2 Експлуатація адміністративної частини вебсайту прокату автомобілів

Для авторизації адміністратора на вебсайті прокату автомобілів в головному меню потрібно натиснути на кнопку Log in та ввійти, заповнити дані та натиснути на кнопку Login.

Для перегляду позицій автомобілів потрібно в головному меню адміністративної частини натиснути на кнопку Car. Вікно перегляду автомобілів наведено на рисунку 2.8.



Photo	Brand	Model	Price	Color	Status	Action
	Lexus	RX350	85	grey	renting	<input type="button" value="Add"/> <input type="button" value="Delete"/>

Рисунок 2.8 – Перегляд позицій автомобілів на адміністративній частині вебсайту

Для додавання нової позиції, потрібно натиснути на кнопку Add car, заповнити форму та натиснути кнопку Insert. Приклад заповненої форми наведений на рисунку 2.9.



Insert car

Brand  
Subaru

Model  
Vitara

Price  
23

Color  
white

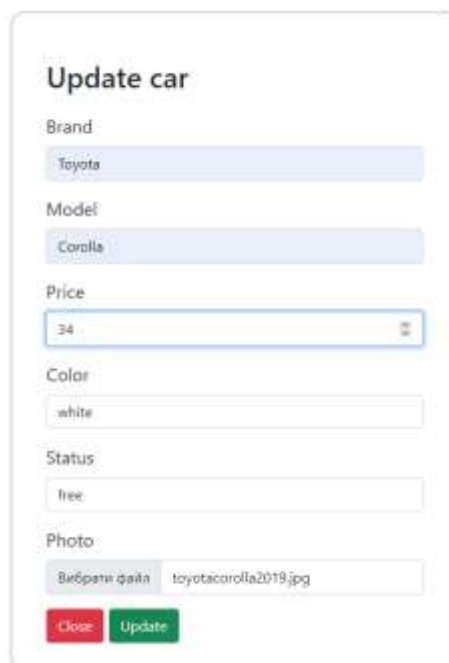
Status  
free

Photo  
Вибрати файл SUZUKI-VITARA.jpg

Close Insert

Рисунок 2.9 – Приклад заповненої форми для додавання нової позиції

Для оновлення даних позиції автомобіля потрібно натиснути на кнопку Update навпроти позиції. Після зміни характеристики автомобіля потрібно підтвердити свої дії, натиснувши на кнопку Update. Приклад форми оновлення позиції авто наведено на рисунку 2.10.



Update car

Brand  
Toyota

Model  
Corolla

Price  
34

Color  
white

Status  
free

Photo  
Вибрати файл toyotacorolla2019.jpg

Close Update

Рисунок 2.10 – Приклад заповненої форми для оновлення позиції

Для видалення автомобіля необхідно натиснути навпроти потрібної позиції кнопку Delete.

Для перегляду замовлень і їх статусу потрібно натиснути на кнопку Rent в головному меню сервісу прокатів автомобілів. Вікно перегляду замовлень на адміністративній частині сайту наведено на рисунку 2.11.

Order ID	Car	Start Date	End Date	Price	Status	Rent Date
1		2022-08-12	2022-08-12	200	rent date	<input type="button" value="Rent Date"/>
2		2022-08-12	2022-08-12	200	rent date	<input type="button" value="Rent Date"/>
3		2022-08-12	2022-08-12	200	rent date	<input type="button" value="Rent Date"/>
4		2022-08-12	2022-08-12	200	rent date	<input type="button" value="Rent Date"/>
5		2022-08-12	2022-08-12	200	rent date	<input type="button" value="Rent Date"/>
6		2022-08-12	2022-08-12	200	rent date	<input type="button" value="Rent Date"/>

Рисунок 2.11 – Вікно перегляду замовлень на адміністративній частині сайту

У вікні перегляду замовлень на адміністративній частині сайту (див. рис. 2.11) вручну адміністратор сайту може завершувати замовлення. Це зроблено через необхідність перевірки стану автомобіля після закінчення прокату та повернення його компанії-власника.

## 2.5 Висновок до другого розділу

В другому розділі кваліфікаційної роботи було розроблено модель даних для вебсайту прокату автомобілів. Описано перелік інформаційних сутностей, опис атрибутів інформаційних сутностей, розроблено модель БД.

Створено та описано файлову структуру вебсайту для сервісу прокату автомобілів. Було описані ключові моменти розробки графічного інтерфейсу користувача та адміністратора а також, розробки серверної частини вебсайту для сервісу прокату автомобілів. Представлено інструкцію для експлуатації користувачької та адміністративної частини вебсайту прокату автомобілів.



## РОЗДІЛ 3. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

### 3.1 Критичні стани програміста

Критичний стан – це медичний термін, що означає серйозний, загрозливий для життя стан здоров'я особи, який вимагає негайного медичного втручання. Критичний стан може бути спричинений різними причинами, включаючи травми, захворювання, отруєння, серцево-судинні проблеми, респіраторні порушення та інші небезпечні становища. Нижче наведено кілька прикладів критичних станів програміста [11]:

- **серцевий напад (інфаркт міокарда):** відбувається, коли кровопостачання до серця переривається, що може призвести до пошкодження серцевої м'язової тканини;
- **ураження голови:** серйозні травми голови, такі як черепно-мозкова травма, можуть викликати набряк мозку або крововилив, що призводить до порушення функції головного мозку;
- **хірургічні ускладнення:** після операцій можуть виникати критичні стани, такі як крововиливи, інфекції, розлади дихання або проблеми з кровообігом;
- **хронічна обструктивна легенева хвороба (ХОЛХ):** прогресуюче захворювання легень, таке як хронічний бронхіт або емфізем, може призводити до критичного стану, коли дихання стає недостатнім;
- **стан шоку:** шок – це серйозний стан, в якому органи та тканини не отримують достатньо крові та кисню, що може спричинити органу недостатність;
- **гостра ниркова недостатність:** відсутність або значне погіршення функції нирок, що призводить до порушення водно-електролітного балансу та накопичення токсичних речовин у організмі.

Це лише декілька прикладів критичних станів людини, існує багато інших медичних умов, які можуть бути надзвичайно небезпечними. У всіх випадках

важливо негайно звернутися до медичних фахівців для отримання допомоги та лікування [11].

У програмістів критичний стан може бути спричинений декількома факторами. Ось деякі з них [11]:

- Екстремальний стрес: програмування може бути вимогливою і складною професією, особливо коли треба виконувати проєкти з короткими термінами або розробляти критичні системи. Такий стрес може призвести до виснаження, зниження рівня концентрації та збільшення ризику помилок.

- Недостатній відпочинок: неправильне розподілення часу та недостатня кількість відпочинку можуть призвести до втоми та перевантаження. Постійне сидіння перед комп'ютером без достатньої паузи може спричинити фізичні та психологічні проблеми.

- Нераціональне харчування, погана харчова якість та нерегулярне харчування можуть призвести до недостатнього постачання організму необхідних поживних речовин, енергії та концентрації. Переважання шкідливої їжі може сприяти погіршенню загального здоров'я.

- Сидячий спосіб життя: довгі години проведені за робочим столом можуть призвести до недостатньої фізичної активності, що може вплинути на серцево-судинну систему, м'язи та загальну фізичну форму. Недостатня рухливість може сприяти розвитку проблем зі здоров'ям. Серед них – неправильна осанка, м'язова напруга та болі у спині.

- Відсутність підтримки та комунікації: відсутність ефективного комунікаційного процесу в команді або недостатня підтримка можуть створити негативну робочу атмосферу, що може вплинути на емоційний стан програміста та загальну продуктивність.

Ці фактори можуть взаємодіяти та провокувати критичний стан програміста, такий як виснаження, стрес та зниження якості роботи. Розуміння цих факторів допоможе програмістам приймати заходи для збереження свого здоров'я та підвищення продуктивності [11].

Хоча багато критичних станів, які загрожують програмістам, пов'язані зі здоров'ям, є також інші фактори, що можуть вплинути на їхню безпеку та ефективність. Ось кілька прикладів таких станів [12]:

- Карпальний тунельний синдром (КТС): це стан, при якому стискаються нерви в зап'ястях, що може виникати внаслідок тривалого неправильного положення рук або надмірного навантаження на зап'ястя. У програмістів, які багато часу проводять за клавіатурою, це може призвести до болю, поколювання або оніміння в руках, що впливає на їхню працездатність.

- Синдром загального виснаження: цей стан пов'язаний з надмірним фізичним та емоційним напруженням. У програмістів, які працюють над складними проектами з високим рівнем стресу, може виникнути виснаженість, погіршення фокусу та зосередженості, втома та погіршення роботи мозку.

- Візуальне напруження: довгі години роботи перед монітором можуть спричинити візуальне напруження, яке включає сухість, свербіж або почервоніння очей, розмитість зору, головні болі або неспокій. Це може суттєво погіршити комфорт і продуктивність програміста.

- Сидячий спосіб життя: багато програмістів проводять багато годин сидячи за комп'ютером, що може призвести до проблем зі здоров'ям, таких як: неправильна осанка, м'язові дисбаланси, знижена метаболічна активність та ризик розвитку серцево-судинних захворювань.

- Стрес та психологічне напруження: програмування може бути вимогливою та стресовою професією. Тривалий стрес може призвести до погіршення психічного стану, виснаженості, проблем зі сном, зниження концентрації та розумової працездатності.

Ці критичні стани можуть вплинути на здоров'я, добробут та продуктивність програмістів. Ось декілька порад, як запобігти критичним станам і зберегти своє здоров'я і ефективність [13]:

- Спланувати свій розпорядок дня: створити розумний графік роботи, який включає час для концентрованої роботи, перерви і відпочинок. Важливо дотримуватись цього розкладу і намагатись уникати надмірної роботи.

- **Вправи та фізична активність:** регулярні фізичні вправи можуть допомогти зняти напругу і покращити кровообіг. Це може бути, для прикладу, йога, прогулянка або пробіжка.
- **Збалансоване харчування:** правильне харчування має велике значення для збереження енергії та концентрації. Потрібно спробувати включити у свій раціон свіжі овочі та фрукти, здорові жири та збалансовані вуглеводи.
- **Сон і відпочинок:** забезпечити собі достатню кількість сну. Регулярний високоякісний сон допомагає відновлювати сили та підтримує когнітивну функцію.
- **Соціальна підтримка:** зберігайте здорові взаємини з колегами та близькими людьми. Обговорюйте свої труднощі і знайдіть підтримку у своєму оточенні. Іноді поділитися своїми переживаннями може зменшити стрес і допомогти знайти рішення.
- **Уникати постійного сидячого положення:** довгі години за комп'ютером можуть бути шкідливими для вашого здоров'я. Потрібно стежити за правильною посадкою за робочим столом, робити паузи для розтягування і рухатись впродовж дня.
- **Розвивати нові навички:** вкладати час у розвиток нових навичок або захоплень. Це може бути все, що зацікавлює, від читання книг до вивчення нових мов програмування. Розширення своїх інтересів може допомогти зменшити стрес і підтримувати позитивний настрій.

Збереження фізичного і психологічного здоров'я є найважливішим для тривалої та успішної кар'єри програміста.

### **3.2 Вплив кольору на покращення умов праці та підвищення продуктивності програмістів**

Оскільки робота програміста – це слово-синонім до виразу сидяча робота, можна сказати, що робоче середовище, в яких працює спеціаліст, прямолінійно

впливає на продуктивність. Одним із таких чинників є колір, який оточує працівника [14].

Кольорова гама робочого середовища має потенційний вплив на наше самопочуття, настрій та продуктивність. Особливо в робочому середовищі вибір кольорів може вплинути на умови праці та концентрацію програміста. Ось кілька кольорів, які присутні в робочому середовищі, та їх вплив на працездатність [14]:

- Синій: сприяє спокою, заспокоєнню та концентрації. Він може допомогти знизити стрес та покращити фокус програміста. Синій також асоціюється зі знаннями та розумом, що може бути корисним для програмістів.
- Зелений: сприяє відчуттю природи та гармонії. Він має розслаблюючий ефект та може допомогти зняти напруження. Зелений також пов'язується зі свіжістю та відновленням енергії, що може бути корисним для відновлення ресурсів програміста.
- Жовтий: стимулює мозок і може підвищити енергію та настрій. Він також асоціюється з творчістю та інтелектом. Проте, жовтий колір також може бути досить стимулюючим і викликати відволікання, тому слід використовувати його з обережністю.
- Червоний: є найбільш ефективним при виконання фізичної роботи, і його можна використовувати не тільки в офісних приміщеннях. Коли людина спостерігає червоний колір, то його відтінки спричиняють підвищення серцевого ритму та прискорення кровообігу.
- Нейтральні кольори (сірий, бежевий та білий) – це нейтральні кольори, які можуть створювати відчуття чистоти, простору та порядку. Вони можуть допомогти уникнути відволікань та сприяти концентрації.

Важливо враховувати індивідуальні вподобання та реакцію кожного програміста на кольори. Деякі люди можуть реагувати по-іншому на певні кольори через свої особливості. Рекомендується експериментувати з кольорами в особистому робочому просторі, щоб знайти оптимальний варіант, який сприяє продуктивності та комфорту [15].

При виборі кольорів робочого середовища програмістів потрібно відповідно до призначення робочого приміщення старатися обирати кольорову гаму. Один з варіантів кольорової гами для робочого середовища може бути комбінація блакитного та білого кольорів. Ця комбінація часто використовується в офісних приміщеннях, оскільки вона має відпочиваючий ефект і сприяє підвищенню концентрації та продуктивності [16].

Блакитний колір вважається успішним вибором для робочих приміщень, оскільки він асоціюється зі спокоєм, стабільністю та довірою. Він також може допомогти знизити рівень стресу і підвищити спокійну атмосферу [14].

Білий колір є універсальним вибором, оскільки він створює відчуття простору та гармонії. Він також відображає світло, що допомагає підтримувати яскравість та енергію приміщення [14].

Комбінування блакитного та білого кольорів створює гармонійне поєднання, яке допомагає створити затишне та професійне робоче середовище. Окрім цього, можна використовувати акцентні деталі або аксесуари інших кольорів (комп'ютерна клавіатура, комп'ютерна мишка, ручка, блокнот тощо), таких як зелений або жовтий, щоб додати життєвості та емоційного вибуху [14].

### **3.3 Висновок до третього розділу**

В третьому розділі кваліфікаційної роботи описано тематику критичних станів програмістів. Наведено приклади критичних станів. Описаний ряд причин, які призводять до критичного стану програмістів. Проаналізовано критичні стани та наслідки, які можуть бути в програмістів, як можна їх уникнути.

Проаналізовано тематику впливу кольорів робочого середовища на покращення умов праці та підвищення продуктивності програмістів. Описано та зроблено аналіз впливу кольору на продуктивність програміста. Було створено один із можливих варіантів кольорової гами робочого середовища для програміста, для підвищення умов праці та його продуктивності.

## ВИСНОВКИ

В результаті виконання кваліфікаційної роботи освітнього рівня “бакалавр” розробки вебсайту для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 та MySQL8.0 виконано всі завдання, які були поставлені при дослідженні тематики.

В першому розділі кваліфікаційної роботи освітнього рівня «Бакалавр»:

- розглянуто та проаналізовано аналоги продуктів, відповідно до тематики;
- сформовано вимоги до вебсайту сервісу прокату автомобілів;
- виконано пошук актантів;
- сформовано таблицю з реєстром варіантів використання;
- досліджено та обґрунтовано вибір технологій та середовища розробки для вебсайту прокату автомобілів.

В другому розділі кваліфікаційної роботи:

- спроектовано, розроблено модель даних для сервісу прокатів автомобілів;
- спроектовано файлову структуру вебсайту;
- розроблено вебсайт для прокату автомобілів засобами HTML5, CSS3, Bootstrap5, PHP7.3 та MySQL8.0;
- наведено та протестовано інструкцію для експлуатації вебсайту для сервісу прокату автомобілів.

У розділі «Безпека життєдіяльності, основи охорони праці» описано тематику критичних станів програмістів. Наведено приклади критичних станів, описано ряд причин, які призводять до критичних станів програмістів. Висвітлено та проаналізовано критичні стани та наслідки, а також поради для їх усунення. Проаналізовано тематику впливу кольорів кімнати на покращення умов праці та підвищення продуктивності програмістів. Описано та зроблено аналіз впливу кольорової гами кімнати на продуктивність програміста.

## ПЕРЕЛІК ДЖЕРЕЛ

1. *Grand Garage*. URL: <https://grand-garage.com.ua/> (дата звернення: 08.02.2023).
2. *BLS*. URL: <https://bls.ua/ua> (дата звернення: 08.02.2023).
3. *Race*. URL: <https://race.net.ua/uk/arenda-avtomobilei-v-ternopole.html> (дата звернення: 08.02.2023).
4. Етапи створення веб сайтів. *WebTune*. URL: <https://webtune.com.ua/statti/web-rozrobka/etapy-stvorennya-veb-sajtiv/> (дата звернення: 08.02.2023).
5. What is Use Case Diagram?. *Visual Paradigm*. URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/> (date of access: 09.02.2023).
6. Build fast, responsive sites with Bootstrap. *Bootstrap*. URL: <https://getbootstrap.com/> (date of access: 09.02.2023).
7. PHP. URL: <https://www.php.net/> (дата звернення: 09.02.2023).
8. MySQL. URL: <https://www.mysql.com/> (дата звернення: 09.02.2023).
9. Code editing. Redefined. *Visual Studio Code*. URL: <https://code.visualstudio.com/> (date of access: 09.02.2023).
10. What is use case diagram?. *Visual Paradigm*. URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/> (date of access: 09.02.2023).
11. Starkman E. Critical, Stable, or Fair: Defining Patient Conditions. *WebMD*. URL: <https://www.webmd.com/a-to-z-guides/defining-patient-conditions> (date of access: 11.02.2023).
12. Griffin A. Critical and serious condition: What hospitals mean when they report a patient's state. *Independent*. URL: <https://www.independent.co.uk/life-style/health-and-families/health-news/critical-and-serious-condition-what-hospitals-mean-when-they-report-a-patient-s-state-a6696701.html> (date of access: 05.06.2023).



13. Левченко О. Г., Землянська О. В., Праховнік Н. А. безпека життєдіяльності та цивільний захист. Київ : КПІ ім. Ігоря Сікорського, 2019. 264 с.
14. Дубовик А. Як колір офісу впливає на продуктивність співробітників?. А-OFFICE. URL: <https://a-office.com.ua/uk/blog/kak-tsvet-ofisa-vliyaet-na-produktivnost-sotrudnikov/> (дата звернення: 07.06.2023).
15. Shapiro E., Augustin S., Calamia M. (2018). The Effects OF Feng Shui On Workplace Design. URL : World Wide Web: <https://www.workdesign.com/2018/12/the-effects-of-feng-shui-on-workplace-design/> (viewed on June 07, 2023). – Title from the screen.
16. Ткачук А. І., Богомаз-Назарова С. М. Основи охорони праці. Кропивницький : ПП "Центр опер. поліграфії "Авангард", 2017. 156 с.

# ДОДАТКИ

**Таблиця з реєстром варіантів використання вебсайту сервісу прокатів  
автомобілів**

<b>Актор</b>	<b>Найменування</b>	<b>Формулювання</b>
<b>1</b>	<b>2</b>	<b>3</b>
Незарєєстрований користувач	Реєстрація	Дозволяє зарєєструвати користувача.
Зарєєстрований користувач	Автентифікація	Дозволяє перевірити наявність певного користувача на основі логіна та пароля, починає сесію та надає певні привілеї.
Користувач	Перегляд автопарку	Дозволяє перевірити статус авто (вільне для оренди чи зайняте/заброньоване), додаткові характеристики.
	Замовлення авто	Екранує дані отримані із форми замовлення та дозволяє оформити заявку на оренду авто.
	Перегляд історії замовлень	Дозволяє користувачу переглянути історію замовлення, їх статус, часові рамки.

1	2	3
Користувач	Вихід з системи	Дозволяє користувачу вийти із облікового запису, завершує сесію.
Адміністратор	Добавити авто	Екранує дані отримані із форми добавлення позиції авто для оренди та дозволяє добавити позицію.
	Видалити авто	Видаляє відповідну позицію авто.
	Редагувати авто	Екранує дані отримані із форми редагування авто для оренди та дозволяє оновити атрибути позиції авто.
	Перегляд користувачів	Дозволяє переглянути інформацію про користувачів.
	Перегляд замовлення	Дозволяє переглянути інформацію замовлення.
	Завершення замовлення	Після підтвердження повернення та успішного технічного огляду завершити замовлення.

Продовження таблиці

1	2	3
Актор	Найменування	Формулювання
Адміністратор	Вихід з системи	Дозволяє адміністратору вийти із облікового запису, завершує сесію.

## Структура таблиць БД для сервісу прокату автомобілів

Таблиця Б.1 – Попередня структура таблиці Car

Ім'я поля	Тип даних	Довжина	Призначення
CarID	int	10	Ідентифікатор авто
Brand	varchar	255	Бренд авто
Model	varchar	255	Модель авто
Price	decimal	10	Ціна оренди (за день)
Color	varchar	255	Колір авто
Status	varchar	255	Статус автомобіля
Photo	varchar	255	Зберігає в собі шлях до фотографії авто

Таблиця Б.2 – Попередня структура таблиці Customer

Ім'я поля	Тип даних	Довжина	Призначення
CustID	int	10	Ідентифікатор користувача
CustFirstName	varchar	255	Ім'я користувача
CustLastName	varchar	255	Прізвище користувача
CustLoginName	varchar	255	Логін користувача
CustAdd	varchar	255	Адреса користувача
CustPhone	varchar	255	Телефонний номер користувача
CustEmail	varchar	255	Електронна адреса користувача
CustPassword	varchar	255	Пароль користувача
CustPrivileges	varchar	255	Привілеї користувача

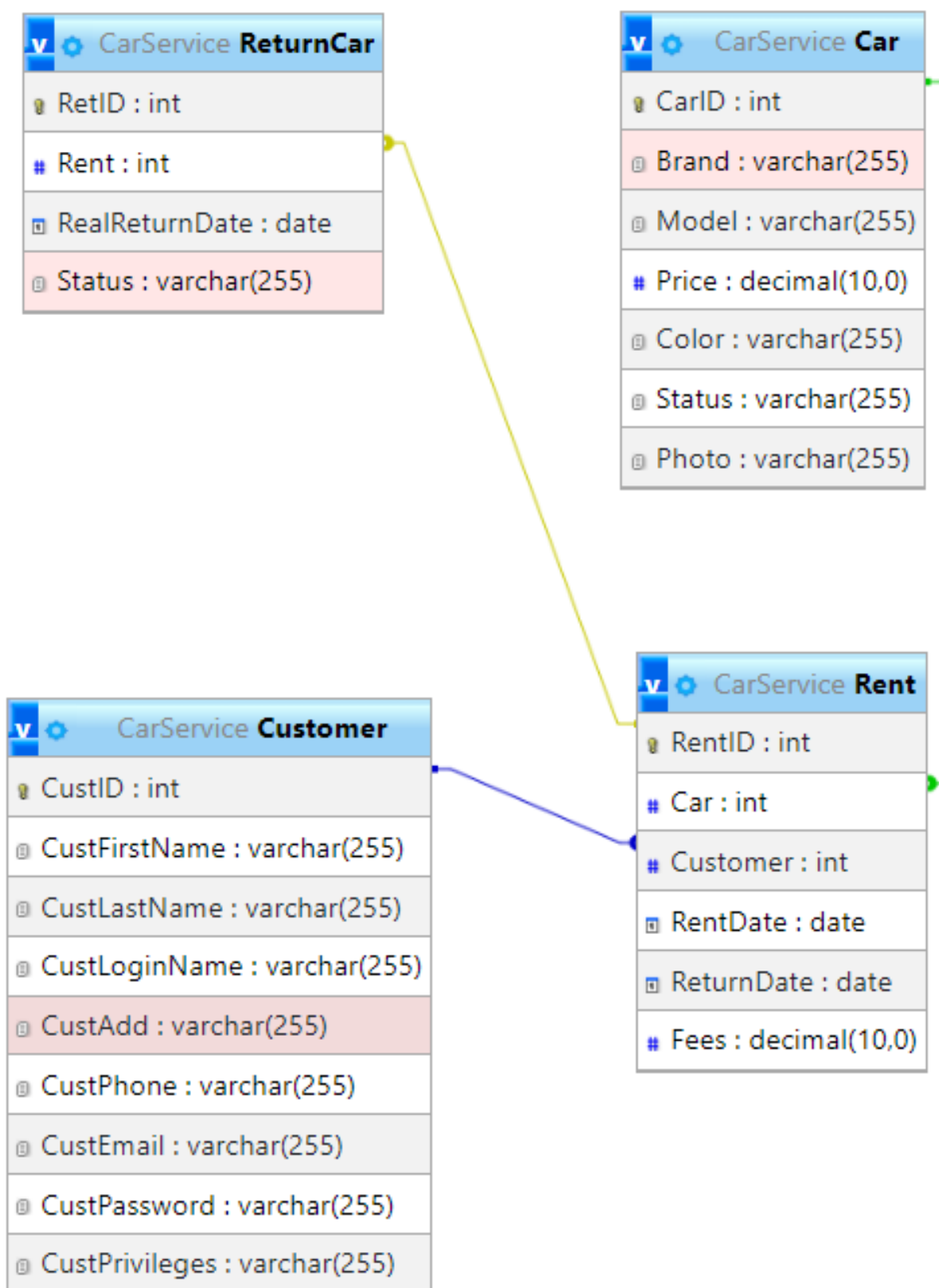
Таблиця Б.3 – Попередня структура таблиці Rent

<b>Ім'я поля</b>	<b>Тип даних</b>	<b>Довжина</b>	<b>Призначення</b>
RentID	int	10	Ідентифікатор замовлення
Car	int	10	Ідентифікатор авто
Customer	int	10	Ідентифікатор користувача
RentDate	date	-	Дата подачі авто
ReturnDate	date	-	Дата повернення авто користувачем
Fees	decimal	10	Фактична сума оренди

Таблиця Б.4 – Попередня структура таблиці ReturnCar

<b>Ім'я поля</b>	<b>Тип даних</b>	<b>Довжина</b>	<b>Призначення</b>
RetID	int	10	Ідентифікатор рядка повернення авто
Rent	int	10	Ідентифікатор замовлення
RealReturnDate	date	-	Фактична дата повернення авто користувачем
Status	varchar	255	Статус виконання замовлення

## Структурна модель БД для вебсайту для сервісу прокату автомобілів





**Реалізація HTML розмітки форми авторизації користувача для вебсайту  
прокату автомобілів**

```
<form class="row g-3" method="post" action="process.php">
  <div class="mb-3 col-md-12">
    <h2 class="form-label">Log in</h2>
  </div>
  <div class="mb-3 col-md-12">
    <label class="form-label">Login</label>
    <input type="text" class="form-control" name="loginName"
aria-describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-12">
    <label class="form-label">Password</label>
    <input type="password" class="form-control"
name="password">
  </div>
  <div class="mb-3 col-md-6">
    <a class="btn btn-danger" id="cancel"
href="index.php">Cancel</a>
  </div>
  <div class="mb-3 col-md-6">
    <button class="btn btn-success" type="submit"
name="login">Login</button>
  </div>
</form>
```

## Реалізація HTML розмітки форми реєстрації користувача для вебсайту прокату автомобілів

```
<form class="row g-3" method="post" action="process.php">
  <div class="mb-3 col-md-12">
    <h2 class="form-label">Sign Up</h2>
  </div>
  <div class="mb-3 col-md-6">
    <label for="exampleInputEmail1" class="form-label">First
Name</label>
    <input type="text" class="form-control" name="firstName"
aria-describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-6">
    <label for="exampleInputEmail1" class="form-label">Last
Name</label>
    <input type="text" class="form-control" name="lastName"
aria-describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-6">
    <label for="exampleInputEmail1" class="form-label">Login
Name</label>
    <input type="text" class="form-control" name="loginName"
aria-describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-6">
    <label for="exampleInputEmail1" class="form-
label">Add</label>
    <input type="text" class="form-control" name="add" aria-
describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-6">
    <label for="exampleInputEmail1" class="form-
label">Phone</label>
    <input type="text" class="form-control" name="phone" aria-
describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-6">
    <label for="exampleInputEmail1" class="form-label">e-
Mail</label>
    <input type="email" class="form-control" name="eMail"
aria-describedby="emailHelp">
  </div>
  <div class="mb-3 col-md-12">
    <label for="exampleInputPassword1" class="form-
label">Password</label>
    <input type="password" class="form-control"
name="password">
  </div>
```

```
<div class="mb-3 col-md-6">
  <a class="btn btn-danger" id="cancel"
href="index.php">Cancel</a>
</div>
<div class="mb-3 col-md-6">
  <button class="btn btn-success" type="submit"
name="register">Submit</button>
</div>
</form>
```

## HTML розмітка таблиці позицій автомобілів на сторінці адміністратора сайту

```

<!-- Table -->
<table class="table table-hover table-striped text-center"
id="tableSelect">
  <!-- Head table -->
  <thead class="table-dark">
    <th scope="col">Photo</th>
    <th scope="col">Brand</th>
    <th scope="col">Model</th>
    <th scope="col">Price</th>
    <th scope="col">Color</th>
    <th scope="col">Status</th>
    <th scope="col">Action</th>
  </thead>
  <!-- Row table -->
  <tbody>
    <tr>
      <td><img src = <?php echo $row['Photo']?>
width="200px" height="133px"></td>
      <td><?php echo $row['Brand'] ?></td>
      <td><?php echo $row['Model'] ?></td>
      <td><?php echo $row['Price'] ?></td>
      <td><?php echo $row['Color'] ?></td>
      <td><?php echo $row['Status'] ?></td>
      <td>
        <a href="carUpdate.php?CarID=<?php echo
$row['CarID']?>" class="btn btn-success">Update</a>
        <a href="delete.php?CarID=<?php echo
$row['CarID']?>" class="btn btn btn-danger">Delete</a>
      </td>
    </tr>
  <?php
  }
  ?>
</tbody>
</table>
<!-- Modal Insert Car -->
<div class="modal fade" id="InsertAutoFormModal" data-bs-
backdrop="static" data-bs-keyboard="false" tabindex="-1" aria-
labelledby="staticBackdropLabel" aria-hidden="true">
  <form action="process.php" method="POST"
enctype="multipart/form-data">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-title"
id="staticBackdropLabel">Insert car</h5>

```

```

        <button type="button" class="btn-close" data-bs-
dismiss="modal" aria-label="Close"></button>
    </div>
    <div class="modal-body">
        <div class="form-group">
            <label class="form-label">Brand</label>
            <input type="text" class="form-control"
name="Brand" id="Brand" value="Hyundai">
        </div>
        <div class="form-group">
            <label class="form-label">Model</label>
            <input type="text" class="form-control"
name="Model" id="Model" value="Accent">
        </div>
        <div class="form-group">
            <label class="form-label">Price</label>
            <input type="number" class="form-control"
name="Price" id="Price" min="0" value="23">
        </div>
        <div class="form-group">
            <label class="form-label">Color</label>
            <input type="text" class="form-control"
name="Color" id="Color" value="white">
        </div>
        <div class="form-group">
            <label class="form-label">Status</label>
            <input type="text" class="form-control"
name="Status" id="Status" value="free">
        </div>
        <div class="form-group">
            <label class="form-label"
for="typeNumber">Photo</label>
            <input type="file" class="form-control"
name="Photo" id="Photo" accept=".jpg, .png, .svg" require>
        </div>
    </div>
    <div class="modal-footer">
        <button type="button" class="btn btn-danger" data-
bs-dismiss="modal">Close</button>
        <button type="submit" class="btn btn-success"
name="carInsert">Insert</button>
    </div>
</div>
</form>
</div>

```