

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ**

Факультет автоматизації і інформаційних технологій

Кафедра Автоматизації інформаційних технологій

Спеціальність «151 Автоматизація та комп'ютерні технології»

ЗАТВЕРДЖУЮ:

Зав. Кафедрою АКІТ

_____ Запривода А.В.

(підпис)

“ ___ ” _____ 20__ р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ ВИПУСКНУ РОБОТУ СТУДЕНТОВІ

Краснощок Андрій Романович

(прізвище, ім'я, по батькові)

1. Тема роботи __ Автоматизація насосної станції _____

затверджена наказом по університету від « 01 » квітня 2024_р. № 595/2

2. Термін здачі студентом закінченої роботи _____

3. Науковий керівник _____

4. Зміст пояснювальної записки (перелік питань, що їх належить розробити)

5. Інформаційні слайди

6. Консультанти роботи

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Прийом програмного продукту

№	Представник комісії	Дата	Підпис

8. Дата видачі завдання

Керівник _____ / _____ / Запривода А.В. _____
 (підпис) (ПІБ)

Завдання прийняв до виконання _____ / Краснощок А.Р.

/

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ**

Автоматизації і інформаційних технологій

Управління проєктами

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО АТЕСТАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО РІВНЯ «БАКАЛАВР»**

на тему:

«Управління проєктом з оренди транспортних засобів»

Краснощок Андрій Романович

Київ 2024 р.

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ**

Автоматизації і інформаційних технологій

Автоматизація та комп'ютерні технології

ЗАТВЕРДЖУЮ

Завідувач кафедри УП

Запривода А.В.

„___” _____ 2024 року

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО АТЕСТАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО РІВНЯ «БАКАЛАВР»**

на тему:

«Автоматизація насосної станції»

Виконала: студент IV курсу, групи _____

Спеціальність: «151 Автоматизація та комп'ютерні технології»

Освітньо-професійна програма: Автоматизація та комп'ютерні технології»

Краснощок Андрій Романович

Керівник к.т.н, доц. Запривода А.В.

Рецензент _____

(прізвище та ініціали)

Київ 2024 р.

АНОТАЦІЯ

Шепелева С. С. Управління проектом з оренди транспортних засобів.

Атестаційна випускна робота бакалавра за спеціальністю: 126

«Інформаційні системи та технології», освітньо-професійної програми: «Управління проектами». – Київський національний університет будівництва і архітектури. – Київ, 2024.

Дипломна робота присвячена розробці та реалізації проекту каршерингу з метою створення інноваційної транспортної послуги.

Кінцевим результатом є розробка та впровадження ефективної системи каршерингу, спрямованої на задоволення потреб клієнтів та підвищення якості транспортних послуг у місті.

Ключові слова: каршерінг, транспортне обслуговування, проект, безпека, екологія, управління проектами.

SUMMARY

Shepelieva S.S. Vehicle rental project management.

Attestation graduation thesis of a bachelor in the specialty: 126

"Information systems and technologies", educational and professional program: "Project management". – Kyiv National University of Construction and Architecture. – Kyiv, 2024.

The thesis is devoted to the development and implementation of a car sharing project with the aim of creating an innovative transport service.

The final result is the development and implementation of an effective car sharing system aimed at meeting the needs of customers and improving the quality of transport services in the city.

Keywords: car sharing, transport service, project, safety, ecology, project management.

ЗМІСТ

Вступ	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ.	11
АКТУАЛЬНІСТЬ ТЕМИ	
1.1 Аналіз предметної області	11
1.2 Постановка задачі	11
1.3 Значення сервісів з оренди автомобілів у світі та в Україні	12
1.4 Формулювання місії	12
1.5 Аналіз конкурентів	12
1.6 SWOT-аналіз	17
1.7 WBS-структура	18
1.8 OBS-структура	22
2 ОПИС СЕРВІСУ З ПОЗИЦІЇ РЕАЛІЗАЦІЇ ПРОЄКТУ	23
2.1 Назва проєкту, його опис	23
2.2 Визначення методики та закріплення ролей	25
2.3 Таблиця оновлень	25
2.4 Вимоги до проєкту	25
2.5 Команда розробки та тестування проєкту	27
2.6 Матриця відповідальності	28
2.7 Бюджет проєкту	29
2.8. Ризики	32
3 ІНФОРМАЦІЙНО-МАТЕМАТИЧНА МОДЕЛЬ ПРОЄКТУ	35
3.1 Мобільні додатки для iOS Android	35
3.2 Веб-додаток	37
3.3 Серверна частина	38
3.4 Система управління поїздками	38
3.5 Система оплати	40
3.6 Система оцінок та відгуків	44
3.7 Адміністративна панель	47

3.8 Безпека	51
3.9 Інфраструктура	53
4 ЕРГОНОМІКА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ	54
4.1 Стандарти ISO 9001:2015 та ISO 14000	54
4.2 Вплив ергономіки на процес розробки каршерінгового проєкту	55
4.3 Рівень якості проєкту	56
4.4 Система управління якістю у проєкті	58
4.5 Якість у процесах управління проєктами	59
4.6 Політика у сфері якості «U-Share»	60
4.7 Цілі у сфері якості «U-Share»	61
4.8 Управління документацією та протоколами	63
4.9 Вимоги у сфері якості до каршерінгового проєкту	65
4.10 Вплив на ресурси каршерінговим проєктом «U-Share»	68
Висновок	70
Список використаних джерел	72
Додатки	74

ВСТУП

Оренда різного виду транспортних засобів є невід'ємною частиною життя багатьох мешканців міст. Найпопулярнішим видом транспорту для переміщення є автомобілі. Саме тому в роботі буде розглядатися каршерінг. «Каршерінг – це сервіс оренди авто на короткостроковий період» [1].

Розвиток каршерінгу є світовим трендом і може вирішити проблеми мешканців, пов'язаних з пересуванням на відносно великі відстані у містах. Оскільки розвиток оренди автомобілів знаходиться на початковій стадії в Україні, то розробка та впровадження проєктів каршерінгу у великих містах країни є актуальним та перспективним.

Метою цього дослідження є розробка та впровадження проєкту тимчасової оренди автомобілів «U-Share» в Україні для підвищення транспортної зручності. Це дозволить забезпечити місцевих та туристів зручним і бюджетним засобом пересування, створити нову бізнес-модель, яка буде прибутковою для ініціаторів проєкту.

Для досягнення мети необхідно вирішити наступні завдання::

- аналіз ринку каршерінгу в Україні;
- порівняння основних конкурентів для визначення переваг та недоліків;
- розробка плану проєкту;
- розробка веб-сторінки для сервісу.

Результати дослідження можуть бути використані в різних сферах, таких як управління та планування міст, транспортних компаніях та

громадському транспорті, розробці програмного забезпечення для транспортних систем та сфери інформаційних технологій.

Практичне значення дослідження полягає в тому, що його результати можуть бути використані для реалізації проєктів каршерінгу в українських містах.

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ. АКТУАЛЬНІСТЬ ТЕМИ

1.1 Аналіз предметної області

Каршерінг – це сервіс оренди авто на короткостроковий період. Це послуга, яка дозволяє користувачам орендувати авто на годину, кілька годин або кілька днів за допомогою мобільного додатку або веб-платформи (сайту). Ця послуга стає дуже популярною у великих містах в умовах загострення транспортних проблем, зростання вартості володіння автомобілем та підвищення екологічної відповідальності. В Україні попит на послуги каршерінгу збільшується у зв'язку зі збільшенням кількості міського населення та транспортними проблемами у великих містах. За статистикою керманіча даної послуги Getmancar у 2022 році попит зріс на 20-30% від попередніх років. Після початку повномасштабної війни в Україні був занепад у сфері каршерінгу, але вже в кінці 2023 року повернувся до довоєнних показників [2].

1.2 Постановка задачі

Метою цього дослідження є розробка та впровадження проекту використання автомобілів «U-Share» в Україні для підвищення транспортної зручності. Завдання включають аналіз ринку послуги каршерінгу в Україні, розробку програмного забезпечення веб-платформи, створення всіх необхідних пунктів проекту для якісної реалізації. Крім того, буде проведено порівняльний аналіз з іншими сервісами каршерінгу на українському ринку для виявлення конкурентних переваг та можливостей для подальшого розвитку.

1.3 Значення сервісів з оренди автомобілів у світі та в Україні

Каршерінг набуває все більшої популярності у всьому світі. За даними PwC очікується, що світовий ринок райдшерінгу зросте до 54,3 млрд доларів США до 2028 року [3]. Це пов'язано з наступними факторами:

- Зростаюча урбанізація: все більше людей переїжджають до міст, де власний автомобіль недоречний через затори на дорогах, проблеми з паркуванням та високі витрати на обслуговування.
- Зміна споживчої поведінки: люди все більше цінують доступність і гнучкість, коли можна швидко пристосуватись до різкої зміни ситуацій. Наприклад, не ходять транспорт через мітинги або зламався власний автомобіль.
- Розвиток технологій: мобільні додатки та пристрої Інтернету речей роблять спільне користування автомобілями простішим і доступнішим.

Хоча український ринок каршерінгу знаходиться на початковій стадії розвитку, він вже демонструє динамічне зростання. Це видно за стрімкою появою нових конкурентів як от Zity або U.Go. Саме тому можна вважати сервіс дуже актуальним та корисним.

1.4 Формулювання місії

Розробка та впровадження каршерінгу «U-Share» є актуальною та перспективною темою. Сервіс завдяки своїм конкурентним перевагам має потенціал стати одним з лідерів на ринку спільного користування автомобілями в Україні.

1.5 Аналіз конкурентів

Аналіз конкурентів на ринку є важливим кроком для успішного розвитку проекту. Це дослідження допомагає зібрати важливі дані про існуючі компанії в галузі, сильні та слабкі сторони, а також поточні та потенційні загрози. Конкурентний аналіз надає можливість зрозуміти

динаміку ринку, виявити тенденції та переваги, які допоможуть створити ефективну стратегію впровадження проєкту. Також є можливість визначити ніші та можливості для інновацій та покращення сервісу. Аналіз конкурентів (табл. 1.1) дозволяє зрозуміти фактори успіху та невдач конкурентів, дозволяючи коригувати стратегічний розвиток бізнесу та його конкурентоспроможність.

Таблиця 1.1

Переваги	Конкурент	Недоліки
Оренда авто можлива маючи лише телефон з додатком та водійське посвідчення	Getmancar	Відсутній фільтр по електромобілям
Можливість побудови маршруту та його ціни	MOBILECAR	Немає фільтру за дитячим кріслом (актуально для молодих батьків)
За пальне, паркування, миття авто по місту платити не треба (закладено у вартість)		Відсутня система мотивування ощадливого керування
		Регіон оренди лише Київ та Дніпро
Більше ніж 8 видів тарифів	AE CarSharing	Неможна залишити машину в області, лише в самому місті
Можна придбати додаткове страхування		При потраплянні в ДТП (неважливо від ролі) платиш компанії часткові виплати
Є можливість додати до «улюблених» конкретний автомобіль, щоб не втратити його	Zity	

Кінець таблиці 1.1

Спеціальне забарвлення автопарків для туризму		
Перші 2 години використання безкоштовні	U.Go	Є ліміти на пробіг за кількістю орендованих годин
Фіксована ціна при оренді на день		
Збільшується ліміт пробігу на вихідних		
Можливість оренди вантажних авто	Getaround	
Варіант «зарезервувати» авто за деякий час до фактичної оренди		
Можливість надати своє авто в оренду (після проходження спеціальної перевірки)		

Уся інформація, подана у таблиці, була проаналізована шляхом користування офіційними додатками компаній.

Даний аналіз покладений в основу проектування функціональності власного сервісу. Для цього можна скористатись різними методиками та підходами. Найчастіше використовується метод брейншторму. Він дозволяє згенерувати велику кількість варіантів покращення сервісу. Для реалізації методу брейншторму з метою фіксування всіх ідей, можна скористатись інструментом Miro.

Miro - це потужний онлайн-інструмент для співпраці, який ідеально підходить для брейншторму та розробки ідей. Команда може використовувати Miro для створення цифрової дошки, на якій кожен учасник може додавати ідеї в режимі реального часу.

Міро для проєкту каршерінгу під час командного мозкового штурму використовувалося наступним чином:

1. Команда створює цифрову дошку на Міро, де кожен учасник може додавати ідеї.
2. Учасники групують ідеї за темами або категоріями для кращої візуалізації.
3. Команди візуалізують ідеї та розробляти концепції, використовуючи різні інструменти Міро, такі як фігури, стрілки та текстові блоки.
4. Учасники обговорюють ідеї та вносять зміни в режимі реального часу, залишаючи коментарі та нотатки на дошці.
5. Після мозкового штурму команда може зберегти дошку, щоб отримати доступ до результатів і продовжити роботу над нею в майбутньому.

Таким чином було отримано результат на рис. 1.1 та рис 1.2 для проєкту щодо ідей, які можна реалізувати.



Рисунок 1.1

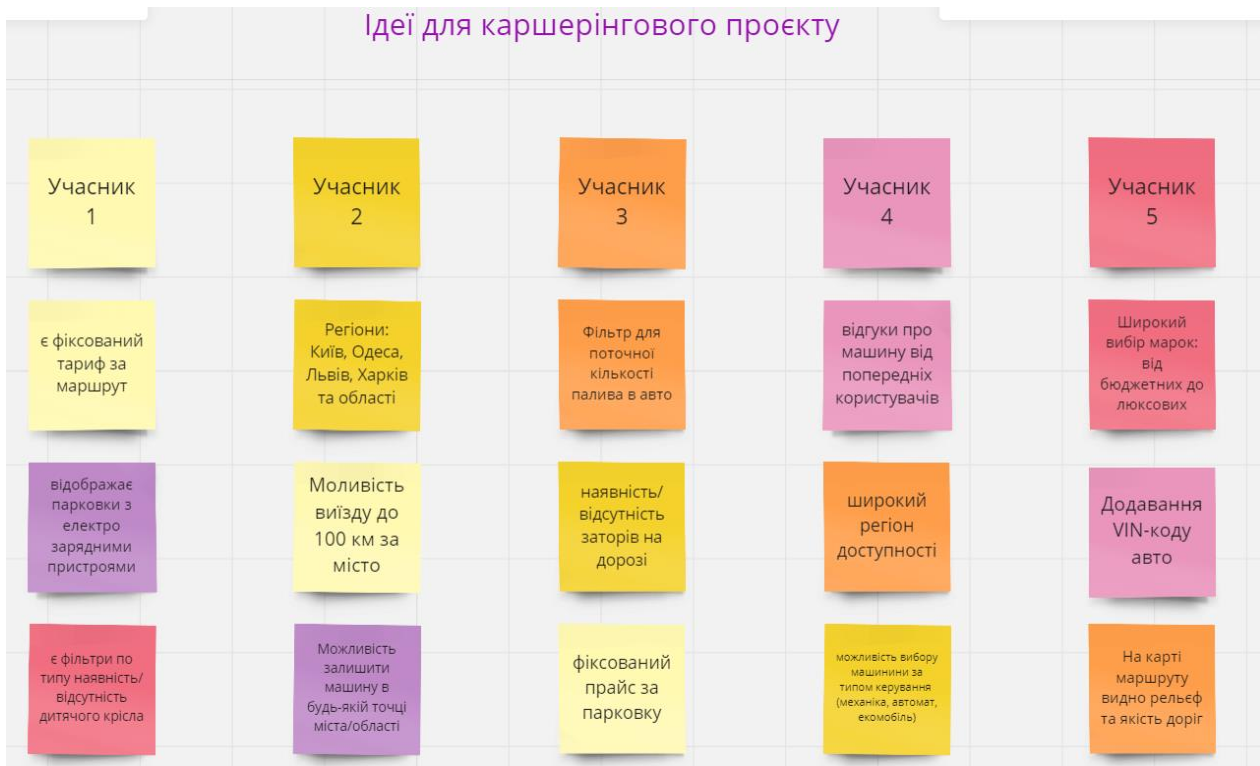


Рисунок 1.2, аркуш 15

Розуміння проблеми, яка розв'язується за допомогою проекту, має вирішальне значення для успішної та ефективної реалізації. Кілька ключових аспектів:

- 1) Цілеспрямованість: зосередження на конкретних завданнях, що покроково приводять до головної мети.
- 2) Детальне планування: визначення ресурсів, термінів та стратегії виконання проекту.
- 3) Попередження проблем: аналіз ризиків дозволяє запобігти непередбачуваним обставинам.
- 4) Пріоритетність: виконання завдань шляхом їхнього розподілення за важливістю.
- 5) Мотивування команди: робота на спільний результат є одним з важливих факторів командоформування.

Шляхом детального аналізу основних компаній у сфері каршерінгу було визначено слабкі сторони такого сервісу. Варіанти їхнього усунення:

- маленька зона покриття: розширити до 100 кілометрів за межею міста;
- безпека користувача: заохочення обережному водінню шляхом надання бонусів на наступні поїздки після завершення попередніх без пригод;
- зручність для користувача: простий процес вибору авто, закладання у вартість оренди витрати на паливо та мийку.
- розширення зон обслуговування: підписання угод з найбільш можливою кількістю заправних станцій, де будуть заправлятися машини з автопарку.

1.6 SWOT-аналіз

В рамках даного проєкту був проведений SWOT-аналіз, заснований на детальному аналізі діяльності конкурентів. SWOT-аналіз – це ефективний інструмент, який дозволяє не тільки оцінювати сильні та слабкі сторони, але й виявляти можливості та загрози, які можуть вплинути на успіх проєкту «U-Share».

Сильні сторони:

- великий асортимент видів та марок автомобілів;
- великий список доповнень за бажаннями клієнта;
- професійне обслуговування технічною підтримкою;
- доступні ціни;
- великий радіус обслуговування в області.

Слабкі сторони:

- відсутність стратегії розвитку на перспективу;
- поки ще невелика популярність;
- відсутність повної версії додатку/веб-платформи.

Можливості:

- зростання території обслуговування;
- просування за допомогою реклами;
- нові бонуси та ідеї заохочення клієнтів;
- актуальність сервісу на ринку.

Загрози:

- велика конкуренція;
- інфляція в країні;
- нестабільний курс валют.

Таким чином, проведений SWOT-аналіз є важливим етапом у процесі стратегічного планування проєкту «U-Share». Це забезпечує чітке розуміння поточної ситуації і допомагає виробити стратегію подальшого розвитку.

1.7 WBS-структура

Структурна декомпозиція робіт (WBS) – це опис роботи, яка буде виконана для проєкту [4]. Це ієрархія завдань, що відображає розуміння проєктною групою складу роботи.

1. Ініціація проєкту

- 1.1 Визначення цілей та завдань проєкту
 - 1.1.1 Проведення дослідження ринку
 - 1.1.2 Ідентифікація зацікавлених сторін
- 1.2 Розробка проєктного статуту
 - 1.2.1 Підготовка документа проєктного статуту
 - 1.2.2 Отримання схвалення від ключових зацікавлених сторін
- 1.3 Проведення наради по проєкту
 - 1.3.1 Підготовка питань для наради
 - 1.3.2 Проведення наради

2. Планування проєкту

- 2.1 Розробка плану управління проєктом
 - 2.1.1 Визначення обсягу та результатів

- 2.1.2 Створення WBS
- 2.2 Розробка фінансового плану
 - 2.2.1 Оцінка витрат
 - 2.2.2 Створення бюджету
- 2.3 Розробка розкладу
 - 2.3.1 Визначення заходів
 - 2.3.2 Послідовність заходів
 - 2.3.3 Оцінка тривалості заходів
- 2.4 Розробка плану управління ризиками
 - 2.4.1 Ідентифікація ризиків
 - 2.4.2 Проведення аналізу ризиків
- 2.5 Розробка плану ресурсів
 - 2.5.1 Ідентифікація необхідних ресурсів
 - 2.5.2 Розподіл ресурсів

3. Дизайн та розробка системи

- 3.1 Дизайн архітектури системи
 - 3.1.1 Визначення вимог до системи
 - 3.1.2 Проєктування компонентів системи
- 3.2 Розробка програмного забезпечення
 - 3.2.1 Розробка інтерфейсу користувача
 - 3.2.2 Реалізація серверної частини
 - 3.2.3 Інтеграція з системами GPS та оплати

- 3.3 Закупівля та налаштування обладнання
 - 3.3.1 Закупівля транспортних засобів
 - 3.3.2 Встановлення пристроїв відстеження
- 3.4 Тестування
 - 3.4.1 Розробка плану тестування
 - 3.4.2 Проведення проміжного тестування
 - 3.4.3 Проведення приймального тестування

4. Маркетинг та продажі

- 4.1 Розробка маркетингової стратегії
 - 4.1.1 Ідентифікація цільової аудиторії
- 4.2 Розробка маркетингових матеріалів
 - 4.2.1 Створення кампаній у соціальних мережах
- 4.3 Запуск маркетингової кампанії
 - 4.3.1 Виконання онлайн-маркетингу
 - 4.3.2 Виконання офлайн-маркетингу

5. Операції та управління

- 5.1 Операційне планування
 - 5.1.1 Визначення операційних процедур
 - 5.1.2 Розробка плану технічного обслуговування
- 5.2 Налаштування обслуговування клієнтів
 - 5.2.1 Створення центру підтримки клієнтів
 - 5.2.2 Розробка протоколів підтримки клієнтів

- 5.3 Навчання персоналу
 - 5.3.1 Розробка навчальних матеріалів
 - 5.3.2 Проведення навчальних сесій
- 5.4 Запуск операцій
 - 5.4.1 Виконання тестового запуску
 - 5.4.2 Повноцінний запуск

6. Моніторинг та контроль

- 6.1 Моніторинг продуктивності
 - 6.1.1 Моніторинг продуктивності системи
 - 6.1.2 Моніторинг використання транспортних засобів
- 6.2 Контроль якості
 - 6.2.1 Проведення регулярних аудитів
 - 6.2.2 Впровадження ініціатив з підвищення якості

7. Закриття проєкту

- 7.1 Підсумковий огляд та звітність
 - 7.1.1 Проведення підсумкового огляду проєкту
 - 7.1.2 Підготовка підсумкового звіту
- 7.2 Передача та перехід
 - 7.2.1 Передача операційній команді
 - 7.2.2 Перехід на підтримку
- 7.3 Закриття проєкту
 - 7.3.1 Закриття документації проєкту

- 7.3.2 Архівування файлів проєкту

Ця структура WBS забезпечує ретельне планування каршерінгового проєкту «U-Share», охоплюючи кожен етап циклу від ініціації до закриття.

1.8 OBS-структура

Організаційна структура проєкту є важливим елементом управління. Це допомагає визначити обов'язки персоналу, координує дії між різними відділами та забезпечує ефективне використання ресурсів.

Склад команди для виконання проєкту «U-Share»:

- менеджер проєкту – написання щотижневих/щоденних звітів, розв'язання локальних проблем в команді, комунікація із зацікавленими сторонами;
- замовник та інвестори – фінансування, формування вимог, затвердження результатів проєкту;
- команда розробки, дизайну та тестування – написання коду та тестування продукту;
- команда безпеки та якості – втілення в проєкті державних стандартів та вимог;
- команда технічної підтримки – отримання результату розробки задля подальшої його підтримки.

Створення чіткої OBS-структури дозволяє забезпечити ефективне управління проєктом.

ОПИС СЕРВІСУ З ПОЗИЦІЇ РЕАЛІЗАЦІЇ ПРОЄКТУ

«Статут проєкту – документ, розроблений ініціатором або спонсором проєкту, який формально авторизує існування проєкту і надає керівнику проєкту повноваження використовувати ресурси організації.

Статут проєкту є вихідним документом стадії ініціації проєкту разом із наказом про призначення команди проєкту та аналізом зацікавлених сторін» [5].

Зазвичай статут містить подальшу інформацію.

2.1 Назва проєкту, його опис

В таблиці 2.1 наведено приклад опису проєкту, який зазвичай розміщується в статуті проєкту.

Таблиця 2.1

Назва	Каршерінговий проєкт “«U-Share»“
Опис	Каршерінговий сервіс, який дозволяє брати авто в короткострокову оренду за фіксованими цінами в будь-який час дня та ночі.
Бачення	
Опис	Українці відмовляються від купівлі дорогих автомобілів, які не вигідно обслуговувати та утримувати на користь орендованих автомобілів, витрати на які обмежуються часом використання. Все, що потрібно користувачеві – це водійське посвідчення та встановлений додаток.
Цільова аудиторія	Мешканці чи відвідувачі найбільших міст України – Києва, Дніпра, Одеси, Львова, яким потрібен автомобіль на нетривалий час, а також ті, хто не має власного автомобіля, але є необхідність отримати доступ до транспортного засобу у будь-який час доби. Ці люди цінують свій час і віддають перевагу самостійному водінню.

Закінчення таблиці 2.1

Проблеми та їх рішення	<ul style="list-style-type: none"> • Мала область покриття. Сервіс працює у 4 містах (Київ, Дніпро, Одеса, Львів), а також за їх межами з можливістю виїзду та завершення оренди за 100 кілометрів від міста у будь-який бік. • Незаплановані витрати. Користувач може розрахувати суму оренди за допомогою фіксованого тарифу на подорожі. • Безпека. Після поїздки користувачі можуть залишити коментар про досвід використання транспортного засобу, справність складових, наявність додаткових автомобільних атрибутів. • Неакуратне водіння. При акуратному керуванні відповідно до всіх правил дорожнього руху користувачеві нараховуються бонуси, які надалі реалізуються у вигляді меншої оплати за поїздку. • Невикористаний оплачений залишок оренди. Програма дозволяє передавати оплачений невикористаний час оренди або кілометри іншому водієві, залишок суми автоматично повертається на картку.
Додаткові послуги	<ul style="list-style-type: none"> • великий перелік фільтрів для вибору авто; • мінімальний рівень палива; • тип коробки, наявність дитячого крісла тощо; • додаток попереджає про перевищення швидкості; • на карті видно пробки, нерівності рельєфу, різкі повороти; • паркувальні місця стандартні та з можливістю підзарядки; • голосовий помічник.

2.2 Визначення методики та закріплення ролей

Дуже важливим є визначення фреймворку, за яким буде виконуватися розробка проєкту. В даному випадку було обрано Scrum. Планується 20 спринтів. Зазначається дата початку проєкту, його поточний стан.

Обов'язково вписується ім'я того, хто виконує роль власника проєкту (це може бути як сам клієнт, так і відповідальна назначена ним людина, яка чітко бачить продукт – бізнес-аналітик/проєктний менеджер/продуктовий менеджер тощо). Поруч має бути і список команди, яка буде займатися проєктом.

2.3 Таблиця оновлень

Таблиця оновлень (табл 2.2.) вимагається на початку, коли формується загальне враження та основна інформація про проєкт, щоб вся команда була в курсі, що додалося або прибралось. Обов'язково підписується клієнтом та стороною-виконувачем/продуктовою компанією (зазвичай відповідає проєктний менеджер). Після кожного оновлення знову підписується (письмово затверджується) обома сторонами.

Таблиця 2.2

Дата	Хто?	Зміни
12.01.2024	@SofiaShepelieva	Створила сторінку Project charter
14.01.2024	@teammember1	Змінив гіперпосилання Team members
26.01.2024	@teammember2	Зробила єдину таблицю бюджету з гіперпосиланнями на складові

2.4 Вимоги до проєкту

Вимоги до проєкту (табл 2.3) зазначаються технічною командою, але підтримуючи зв'язок та погодження з замовником.

Таблиця 2.3

Вимога	Опис
1. Масштабованість для інших пристроїв залежно від розміру екрану	для моделей Iphone, які підтримують оновлення, для Windows, для телефонів Android
Швидкість відповіді на дзвінок до служби підтримки	max 1 хв
Швидкість відповіді на запит про аварію до служби підтримки	max 5 хв
Швидкість отримання чека на пошту про подорож	max 5 хв
Швидкість відображення машин на карті згідно з локацією користувача	max 5 сек
Швидкість отримання смс під час реєстрації	max 1 хв
Довжина пароля має бути не менше 8 і не більше 14 символів	
Літерна частина пароля повинна містити як малі, так і великі (великі) літери	
Не можна використовувати попередні паролі	
Пароль не повинен збігатися з логіном, ім'ям, датою народження або прізвищем користувача	
Пароль не повинен містити трьох і більше однакових символів поспіль	
У системі доступні кнопки визначення рівня надійності пароля	"слабкий", "низький", "середній", "високий", "дуже високий".

Закінчення таблиці 2.3

Після трьох спроб входу в систему система безпеки може заблокувати обліковий запис користувача	
Додаток пропонує такі ж функції на пристрої iOS, що і на пристрої Android	
Місткість програми	не більше 150 Мб
Програма запускається лише тоді, коли користувач запускає його	
До 500 користувачів можуть завантажити права водія одночасно	
Можна налаштувати 3 мови користувача	українська, англійська
Коли користувач переміщається сторінкою програми, він має доступ до кнопок "налаштування" , "акаунт" , "радар "	
Зміна колірної гами всієї кнопки при застосуванні темної теми	

2.5 Команда розробки та тестування проєкту

У таблиці 2.4 зазначається позиція, ім'я, контактна пошта, основні навички та чи є ця людина внутрішньою чи найманою (аутсорс).

Таблиця 2.4

Team member	Name	E-mail	Stack	Internal/External
Designer	Софія	sofiashepeleva0440@gmail.com	C++/C#	Internal
FE dev	Ім'я 1	Пошта1	React Native	Internal

Закінчення таблиці 2.4

BE dev - 1	Ім'я 5	Пошта5	SWIFT/Java	Internal
BE dev - 2	Ім'я 2	Пошта2	SWIFT/Java	Internal
QA	Ім'я 3	Пошта3	Java, JS, C#, Python	Internal
PM	Ім'я 4	Пошта4	Jira, Confluence	Internal

2.6 Матриця відповідальності

Безперечно після зазначення команди та учасників проєкту наступним кроком є визначення відповідальностей під час розробки проєкту. У таблиці 2.5 матриці зазначено не всю структуру, а саме дії управління проєктом.

При заповненні використовуються 4 літери:

R – Responsible (виконання);

A – Accountable (відповідальність);

C – Consult before doing (консультація);

I – Inform after doing (інформується після виконання).

Таблиця 2.5

Фаза	Задача	CEO	DEV	QA	PM
Ініціація	Виявлення проблеми				R&A
	Розробка рішення	C&I	C		R&A
Планування	Розробка Roadmap		C&I		R&A
	Розрахування бюджету		C		R&A
	Аналіз конкурентів				R&A

Закінчення таблиці 2.5

	Написання плану комунікацій		C&I		R&A
	Створення уставу проєкту	I	C		R&A
	Створення Deliverables		C		R&A
	Estimation		C&I		R&A
Розробка	Виконання		R	I	A
	Завершення задач спринту	I	R	I	A
Тестування	Виявлення проблем		I	R	A
	Розробка рішення		I	R	A
	Рішення проблеми		I	R	A
Завершення	Документування досвіду				R&A
	Звіт про завершення	I			R&A
	Договір супроводу	I	C		R&A

2.7 Бюджет проєкту

Сумарний бюджет проєкту складається з двох розділів: витрати на заробітну плату і технічну складову, необхідна для виконання задач.

Сумарний бюджет у табл. 2.6:

Таблиця 2.6

Загальна вартість	Розрахунково
Заробітна плата	\$232 726
Доставка та техніка	\$148 180
Інші витрати	\$0
Всього	\$380 906
Непередбачувані витрати (%)	15,00%
Сума непередбачуваних витрат	\$57 136
Загальна сума	\$438 042

Заробітна плата в табл. 2.7:

Таблиця 2.7

Посада	Всього мін, год	Всього макс, год	Ціна додаткови х год, \$	Всього мін сума, \$	Всього макс сума, \$
Проектний менеджер	320	320	\$20	\$6 400	\$6 400
UX/UI дизайнер	608	896	\$20	\$12 160	\$17 920
Розробник BE Java 1	888	1260	\$25	\$22 200	\$31 500
Розробник BE Java 2	888	1260	\$25	\$22 200	\$31 500
Розробник FE dev	680	898	\$35	\$23 800	\$31 430
QA	420	700	\$30	\$12 600	\$21 000
Clean up UX/UI	24	24	\$20	\$480	\$480
Чистка BE Java 1	24	24	\$25	\$600	\$600
Чистка BE Java 2	24	24	\$24	\$576	\$576
Чистка FE dev	24	24	\$30	\$720	\$720
Чистка QA	24	24	\$25	\$600	\$600
Всього	3924	5454		\$102 336	\$142 726

Витрати на технічну складову в табл. 2.8:

Таблиця 2.8

Найменування	Години	Сума	Підсумок
			\$22 400,00
Компанія соціальних досліджень	240	\$50,00	\$12 000,00
Аналіз поточної ситуації	40	\$25,00	\$1 000,00
Проектне планування	40	\$35,00	\$1 400,00
Експерти	320	\$25,00	\$8 000,00
Обладнання			\$25 380,00
Macbook	1	\$2 500,00	\$2 500,00
Lenovo note	1	\$1 000,00	\$1 000,00
Android phone	1	\$400,00	\$400,00
Iphone	1	\$700,00	\$700,00
Закупівля серверів	12	\$65,00	\$780,00
Вартість серверів	2	\$10 000,00	\$20 000,00
Програмне забезпечення			\$80 000,00
Ліцензійне забезпечення	1600	\$50,00	\$80 000,00
Jira стандарт	10	\$45,00	\$450,00
Miro Team	10	\$48,00	\$480,00
Навчання та підтримка			\$12 000,00
Вартість командних подорожей	150	\$50,00	\$7 500,00
Вартість навчання	150	\$30,00	\$4 500,00
Додаткове обладнання			\$8 400,00
Дизайн та рекламне оформлення автомобілів	80	\$45,00	\$3 600,00
Оренда електромобілів	160	\$10,00	\$1 600,00

Закінчення таблиці 2.8

Оренда бізнес-автомобілів	160	\$10,00	\$1 600,00
Оренда стандартних автомобілів	160	\$10,00	\$1 600,00
		Загальна вартість	\$148 180,00

2.8. Ризики

Дуже важливим під час проекту є правильне визначення ризиків. Що може піти не так? Зламається техніка, звільниться людина, пандемія.

Прорахувати все неможливо, але імовірні ситуації обов'язково треба брати до уваги. Їх приклад наведено в табл. 2.9.

Таблиця 2.9

	Ризик	Рівень	Час на вирішення	Рішення
1	Затримка постачання 12 серверів напередодні святкових днів	Низький	+1-2 тижні	Розірвання договору з попереднім постачальником, підписання з альтернативним постачальником
2	Труднощі застосування цифрової системи доступу на ретро авто	Високий	n/a	Введення ретро-авто винести за проєкт (окремо)

Закінчення таблиці 2.9

3	Проблеми налагодження безперебійного обміну даними сервер-автомобіль через унікальність проекту	Середній	+1-3 тижні	Фахівці outstaff на світовому ринку (знайти, провести переговори)
4	Затягування підписання угод про безкоштовне паркування з державними органами	Середній	+ 1-5 тижнів	Повідомити ЗМІ (засоби масової інформації)
5	Схвалення оплат криптовалютою з боку регулюючих органів	Високий	неможливість випуску релізу	Викреслити опцію з релізу, замінити на альтернативні оплати
6	Інформація щодо зон завершення оренди за містом не надана до кінця 4 спринту (1 місяць робіт)	Низький		Карту завершення оренди в MVP (мінімально життєздатний продукт) звузити до межі
7	Пандемія	Низький	+2-3 тижні	Перехід на повну дистанційну роботу

Закінчення таблиці 2.9

8	Війна	Високий	+2-4 тижні	Вивезення на безпечну територію працівників (за можливості)
---	-------	---------	---------------	--

ІНФОРМАЦІЙНО-МАТЕМАТИЧНА МОДЕЛЬ ПРОЄКТУ

Для створення застосунку «U-Share» мають бути виконані такі етапи розробки архітектури додатка:

1. Мобільні додатки для iOS Android
2. Веб-додаток
3. Серверна частина
4. Система управління поїздками
5. Система оплати
6. Система оцінок та відгуків
7. Адміністративна панель
8. Безпека
9. Інфраструктура

3.1 Мобільні додатки для iOS Android

Для створення мобільного додатку використовується кросплатформенне рішення, таке як React Native. Воно дозволяє розробляти кросплатформенні мобільні додатки на JavaScript, використовуючи той один код як для iOS, так і для Android, що спрощує час та підтримку додатку.

Приклад створення простої форми входу на React Native:

```
import React, { useState } from 'react';
import { View, TextInput, Button, StyleSheet, Text } from 'react-native';
const LoginScreen = ({ navigation }) => {
  const [username, setUsername] = useState("");
  const [password, setPassword] = useState("");
  const handleLogin = () => {
```

```

    navigation.navigate('Home');
  };
  return (
    <View style={styles.container}>
      <Text style={styles.label}>Ім'я користувача</Text>
      <TextInput
        style={styles.input}
        value={username}
        onChangeText={setUsername}
        placeholder="Введіть ім'я користувача"
      />
      <Text style={styles.label}>Пароль</Text>
      <TextInput
        style={styles.input}
        value={password}
        onChangeText={setPassword}
        placeholder="Введіть пароль"
        secureTextEntry
      />
      <Button title="Вхід" onPress={handleLogin} />
    </View>
  );
};
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    padding: 16,
  },
  label: {

```

```

    fontSize: 18,
    marginBottom: 8,
  },
  input: {
    borderWidth: 1,
    borderColor: '#ccc',
    padding: 8,
    marginBottom: 16,
    borderRadius: 4,
  },
});
export default LoginScreen;

```

3.2 Веб-додаток

Веб-додаток створюється за допомогою HTML, CSS, JavaScript.

Приклад простої форми з прикладу входу за допомогою HTML та JavaScript:

```

<h2>«U-Share»</h2>

<form action="/action_page.php">

  <div class="imgcontainer">

  </div>

  <div class="container">

    <label for="uname"><b>Ім'я користувача</b></label>

    <input type="text" placeholder="" name="test" required>

    <label for="psw"><b>Пароль</b></label>

    <input type="password" placeholder="" name="test" required>

```

```
<button type="button" onclick="redirectToNewPage()">Вхід</button>
```

```
<script>
```

3.3 Серверна частина

Серверна частина буде складатись з декількох пунктів – бекенд-сервіс та база даних.

Бекенд-сервіс відповідає за обробку запитів від клієнтських додатків, зберігання даних та виконання бізнес-логіки. Для цього використовується Node.js та Express, так як є масштабування завдяки своїй асинхронній природі та великій кількості доступних модулів.

База даних зберігає інформацію про користувачів, поїздки, платежі та відгуки. Використання реляційних баз даних таких як PostgreSQL забезпечує ефективне зберігання та доступ до даних. Ця база підтримує складні запити та транзакції.

Приклад створення схеми бази даних для користувачів:

```
CREATE TABLE users (
  id SERIAL PRIMARY KEY,
  username VARCHAR(50) UNIQUE NOT NULL,
  password VARCHAR(100) NOT NULL,
  email VARCHAR(100) UNIQUE NOT NULL,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

3.4 Система управління поїздками

Система управління поїздками дозволяє користувачам замовляти та автоматично підбирати найближче авто, враховуючи такі фактори як відстань, рейтинг та час, необхідний щоб дістатися до транспорту. Алгоритм наведено на рис. 3.1:

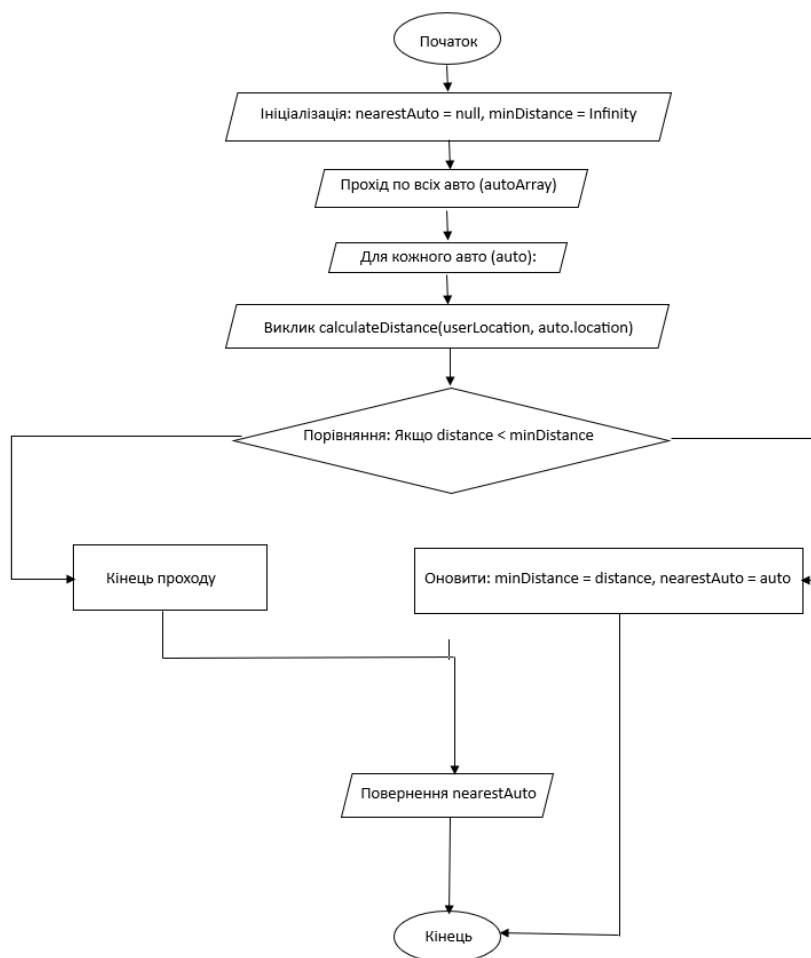


Рисунок 3.1

Приклад алгоритму на JavaScript:

```

function findNearestAuto(autos, userLocation) {
  let nearestAuto = null;
  let minAuto= Infinity;
  auto.forEach(auto => {
    const auto = calculateAuto(userLocation, auto.location);
    if (distance < minDistance) {
      minDistance = distance;
      nearestAuto= driver;
    }
  });
};
  
```

```

    return nearestAuto;
}
function calculateDistance(loc1, loc2) {
    const R = 6371;
    const dLat = (loc2.lat - loc1.lat) * (Math.PI / 180);
    const dLon = (loc2.lon - loc1.lon) * (Math.PI / 180);
    const a =
        Math.sin(dLat / 2) * Math.sin(dLat / 2) +
        Math

```

3.5 Система оплати

Система оплати забезпечує безпечну обробку платежів та автоматично списує кошти після завершення поїздки.

Основні компоненти системи оплати:

1. Для інтеграції з платіжними системами будемо використовувати Stripe. Вона забезпечує найкращий захист інтернет-платежів та відповідає найвищому рівню сертифікації міжнародних стандартів безпеки.

```

const express = require('express');

const bodyParser = require('body-parser');

const stripe = require('stripe')('your_stripe_secret_key');

const app = express();

app.use(bodyParser.json());

app.post('/create-payment-intent', async (req, res) => {

    const { amount, currency } = req.body;

    try {

```

```
const paymentIntent = await stripe.paymentIntents.create({  
  
  amount,  
  
  currency,  
  
});  
  
res.send({  
  
  clientSecret: paymentIntent.client_secret,  
  
});  
  
} catch (error) {  
  
  res.status(500).send({ error: error.message });  
  
}  
  
});  
  
const port = 3000;  
  
app.listen(port, () => {  
  
  console.log(`Server running on port ${port}`);  
  
});
```

2. Автоматична оплата списання коштів після поїздки:

```
import React, { useState } from 'react';  
  
import { loadStripe } from '@stripe/stripe-js';
```

```
import { Elements, CardElement, useStripe, useElements } from
'@stripe/react-stripe-js';

const stripePromise = loadStripe('your_stripe_public_key');

const CheckoutForm = () => {

  const stripe = useStripe();

  const elements = useElements();

  const [paymentError, setPaymentError] = useState(null);

  const [paymentSuccess, setPaymentSuccess] = useState(null);

  const handleSubmit = async (event) => {

    event.preventDefault();

    const card = elements.getElement(CardElement);

    const { error, paymentMethod } = await stripe.createPaymentMethod({

      type: 'card',

      card,

    });

    if (error) {

      setPaymentError(error.message);

      setPaymentSuccess(null);

    } else {
```

```
    setPaymentError(null);

    setPaymentSuccess('Payment successful!');

  }

};

return (

  <form onSubmit={handleSubmit}>

    <CardElement />

    <button type="submit" disabled={!stripe}>

  </button>

  </form>

);
```

3. Для безпеки платіжної системи сайту використовується шифрування за допомогою сертифікатів SSL/TLS, так як саме це є найпопулярнішим методом:

```
const fs = require('fs');

const https = require('https');

const express = require('express');

const app = express();

const options = {

  key: fs.readFileSync('server.key'),

  cert: fs.readFileSync('server.cert')

};
```

```
https.createServer(options, app).listen(3000, () => {
  console.log('HTTPS сервер працює на порту 3000');
});
```

3.6 Система оцінок та відгуків

Після завершення поїздки користувачі можуть залишати оцінки та відгуки, що допомагає підтримувати якість сервісу.

Основні компоненти системи оцінок та відгуків:

1. Форма оцінки поїздки на React та обробку запиту на Node.js:

```
import React, { useState } from 'react';

import axios from 'axios';

const RatingForm = ({ tripId }) => {

  const [rating, setRating] = useState(0);

  const [comment, setComment] = useState("");

  const handleSubmit = async (event) => {

    event.preventDefault();

    try {

      await axios.post('/api/rate-trip', { tripId, rating, comment });

      alert('Дякуємо за вашу оцінку!');

    } catch (error) {
```

```
console.error('Error submitting rating:', error);

alert('Не вдалося залишити оцінку. Спробуйте ще раз.');
```

```
}

};

return (

<form onSubmit={handleSubmit}>

  <label>

    Оцініть поїздку:

    <select value={rating} onChange={(e) => setRating(e.target.value)}
required>

      <option value="" disabled>Виберіть оцінку</option>

      <option value="1">1</option>

      <option value="2">2</option>

      <option value="3">3</option>

      <option value="4">4</option>

      <option value="5">5</option>

    </select>

  </label>

  <label>
```

Коментар:

```

    <textarea value={comment} onChange={(e) =>
setComment(e.target.value)} />

    </label>

    <button type="submit">Відправити</button>

  </form>

);

};

export default RatingForm;

app.post('/api/rate-trip', async (req, res) => {

  const { tripId, rating, comment } = req.body;

  try {

    await db.query('INSERT INTO ratings (trip_id, rating, comment)
VALUES ($1, $2, $3)', [tripId, rating, comment]);

    res.status(200).send('Оцінку успішно збережено');

  } catch (error) {

    console.error('Error saving rating:', error);

    res.status(500).send('Не вдалося зберегти оцінку');

  }

});

```

2. Збір та аналіз відгуків (дані зберігаються в PostgreSQL):

```
CREATE TABLE ratings (
    id SERIAL PRIMARY KEY,
    trip_id INTEGER NOT NULL,
    rating INTEGER NOT NULL,
    comment TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

3.7 Адміністративна панель

Адміністративна панель дозволяє адміністраторам керувати обліковими записами користувачів та автомобілями, переглядати статистику і звіти.

1. Управління обліковими записами користувачів – додавання в групи, редагування, видалення.

Додавання:

```
app.post('/admin/add-user', async (req, res) => {
    const { name, email, password } = req.body;
    const hashedPassword = await bcrypt.hash(password, 10);
    try {
```

```

    await db.query('INSERT INTO users (name, email, password) VALUES
($1, $2, $3)', [name, email, hashedPassword]);

    res.status(200).send('Користувача додано успішно');

  } catch (error) {

    console.error('Error adding user:', error);

    res.status(500).send('Не вдалося додати користувача');

  }

});

```

Редагування:

```

app.put('/admin/edit-user/:id', async (req, res) => {

  const { id } = req.params;

  const { name, email } = req.body;

  try {

    await db.query('UPDATE users SET name = $1, email = $2 WHERE id =
$3', [name, email, id]);

    res.status(200).send('Користувача оновлено успішно');

  } catch (error) {

    console.error('Error updating user:', error);

    res.status(500).send('Не вдалося оновити користувача');

  }

});

```

```
});
```

Видалення користувача:

```
app.put('/admin/edit-user/:id', async (req, res) => {

  const { id } = req.params;

  const { name, email } = req.body;

  try {

    await db.query('UPDATE users SET name = $1, email = $2 WHERE id =
$3', [name, email, id]);

    res.status(200).send('Користувача оновлено успішно');

  } catch (error) {

    console.error('Error updating user:', error);

    res.status(500).send('Не вдалося оновити користувача');

  }

});
```

2. Статистика та звіти – перегляд аналітики по поїздках та доходах. Для отримання цих статистик використовуємо JavaScript:

```
app.get('/admin/trip-statistics', async (req, res) => {

  try {
```

```
    const result = await db.query('SELECT COUNT(*) AS total_trips,
AVG(rating) AS average_rating FROM trips JOIN ratings ON trips.id =
ratings.trip_id');

    res.json(result.rows[0]);

  } catch (error) {

    console.error('Error fetching trip statistics:', error);

    res.status(500).send('Не вдалося отримати статистику поїздок');

  }

});

app.get('/admin/revenue-statistics', async (req, res) => {

  const { startDate, endDate } = req.query;

  try {

    const result = await db.query('SELECT SUM(amount) AS total_revenue
FROM payments WHERE created_at BETWEEN $1 AND $2', [startDate,
endDate]);

    res.json(result.rows[0]);

  } catch (error) {

    console.error('Error fetching revenue statistics:', error);

    res.status(500).send('Не вдалося отримати статистику доходів');

  }

}
```

```
});
```

3.8 Безпека

Безпека є критичним аспектом для додатку. Вона захищає особисті дані та платежі.

Основні заходи безпеки

1. Для шифрування даних використовується SSL/TLS для веб-сервера nginx:

```
server {  
  
    listen 443 ssl;  
  
    server_name «U-Share».com;  
  
    ssl_certificate /etc/ssl/certs/ «U-Share».com.crt;  
  
    ssl_certificate_key /etc/ssl/private/ «U-Share».com.key;  
  
    ssl_protocols TLSv1.2 TLSv1.3;  
  
    ssl_ciphers HIGH:!aNULL:!MD5;  
  
    location / {  
  
        root /var/www/html;  
  
        index index.html index.htm;  
  
    }  
  
    error_log /var/log/nginx/error.log;  
  
    access_log /var/log/nginx/access.log;
```

```
}
```

2. Для авторизації користувачів використовується захист маршрутів.

Приклад коду:

```
const authenticateJWT = (req, res, next) => {  
  
  const token = req.header('Authorization')?.split(' ')[1];  
  
  if (token) {  
  
    jwt.verify(token, 'your_jwt_secret', (err, user) => {  
  
      if (err) {  
  
        return res.sendStatus(403);  
  
      }  
  
      req.user = user;  
  
      next();  
  
    });  
  
  } else {  
  
    res.sendStatus(401);  
  
  }  
  
};  
  
app.get('/protected', authenticateJWT, (req, res) => {
```

```
res.send('Це захищений маршрут');

});
```

3. Захист від різноманітних атак. Наприклад, від DDoS атак захищає хостинг Azure, а для захисту Brute force обмежується кількістю спроб входу.

Приклад:

```
const rateLimit = require('express-rate-limit');

const loginLimiter = rateLimit({

  windowMs: 15 * 60 * 1000,

  max: 5,

  message: 'Забгато спроб входу, спробуйте пізніше'

});

app.post('/login', loginLimiter, async (req, res) => {

});
```

3.9 Інфраструктура

Для забезпечення надійності та масштабованості проєкту буде використано:

- хостинг: Azure.
- інструменти для моніторингу та логування додатку, веб-сторінки, сервера тощо.
- система резервного копіювання даних.

ЕРГОНОМІКА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Ергономіка інформаційних технологій відіграє важливу роль не тільки в кінцевому продукті, але й у процесі розробки проєкту. Застосування ергономічних принципів на етапах планування, розробки та тестування ІТ-рішень може значно підвищити продуктивність команди, знизити ризик помилок і забезпечити високу якість і зручність у використанні продуктів/сервісів.

4.1 Стандарти ISO 9001:2015 та ISO 14000

Стандарти ISO 9001:2015 та ISO 14000 для систем управління якістю та екологічного менеджменту створюють основу для ергономіки ІТ та підкреслюють її важливість.

ISO ДСТУ 9001:2015:

- Розділ 4.2 "Вимоги до процесів" стандарту рекомендує враховувати ергономічні міркування при проєктуванні та розробці ІТ-систем.
- Розділ 8.2 "Вимірювання, аналіз та поліпшення" стандарту рекомендує використовувати ергономічні показники для оцінки ефективності ІТ-систем.

ISO 14000:

•Розділ 4.3.1 "Визначення екологічних аспектів" стандарту рекомендує враховувати вплив ІТ-систем на навколишнє середовище, в тому числі ергономічні міркування, які можуть призвести до неефективного використання ресурсів.

•Розділ 4.4.1 стандарту "Постановка екологічних цілей і завдань" рекомендує ставити цілі і завдання для поліпшення ергономіки ІТ-систем і зменшення їх негативного впливу на навколишнє середовище.

4.2 Вплив ергономіки на процес розробки каршерінгового проєкту

Планування:

- На етапі планування проєкту слід провести ергономічний аналіз для визначення потреб і очікувань користувачів ІТ-системи.

- На основі цього аналізу слід розробити ергономічні вимоги до ІТ-системи та включити їх в умови введення в експлуатацію.

Розробка:

- При розробці ІТ-систем слід використовувати ергономічні методи та інструменти, такі як створення прототипів, тестування користувачів та аналіз даних.

- Важливо враховувати ергономічні принципи при розробці користувацьких інтерфейсів, навігації, довідкових систем та інших елементів ІТ-систем.

Тестування:

- На етапі тестування ІТ-системи слід проводити ергономічні тести для виявлення та виправлення помилок, а також для оцінки зручності та ефективності ІТ-системи.

Переваги ергономічного підходу

- Підвищення продуктивності: ергономічні ІТ-системи дозволяють командам працювати швидше, ефективніше і з меншою кількістю помилок.

- Зниження витрат: ергономічний підхід може допомогти зменшити витрати на розробку, тестування та підтримку.

- Підвищення задоволеності користувачів: ергономічні ІТ-системи більш зручні, прості у використанні.

- Зменшення впливу на навколишнє середовище: ергономічні ІТ-системи можуть зменшити негативний вплив на навколишнє середовище, включаючи економію енергії та ресурсів.

Застосування цих принципів при розробці проєкту каршерінгу може значно підвищити ефективність, зручність і екологічність цього проєкту.

4.3 Рівень якості проєкту

Визначення рівня проєкту у сфері управління якістю, виявлення слабких сторін та планування заходів щодо їх усунення в табл. 4.1.

Таблиця 4.1

Загальне управління якістю (TQM)											Відсутність професійного управління якістю
Зосередження на клієнтах											Зосередження на виробничому процесі
Дії зосередженні на фактах											Дії зосередженні на думках
Діяльність орієнтована на результат											Діяльність орієнтована на методи та засоби
Орієнтація на дію											Тільки розмови
Роботи щодо якості на всіх рівнях та у всіх підрозділах											Роботи щодо якості зосередженні в деяких підрозділах
Професійне керівництво											Аматорське керівництво
Кожен приймає участь у роботах щодо якості											Тільки підрозділ якості

-залучити до даного проєкту менеджера з досвідом, зацікавивши незвичайними викликами, з якими можемо стикнутися впродовж розробки та впровадження;

-звернутися до експертів за порадами та оцінкою релевантності реалізації проєкту. На основі отриманих порад керувати нинішнім та подальшим процесом.

2) Хто бере участь у контролі якості: в основному в компаніях є спеціальний підрозділ, який займається і контролює якість. Спиратися на цю схему.

Вирішення:

-впровадити в кожний підрозділ людину, що відповідатиме за якість;
-підтримувати мотивацію співробітників на високому рівні задля того, щоб кожен працював на совість.

4.4 Система управління якістю у проєкті

Управління проєктами включає безперервне планування, організацію, моніторинг та контроль за всіма аспектами для досягнення зовнішніх та внутрішніх цілей.

Якість процесів управління проєктами надає якості продуктів/сервісів проєкту, отже й успіх організації.

Визначення цілі системи управління якістю проєкту:

1. Забезпечення високої якості веб-сайту (швидке завантаження, зрозумілий інтерфейс, недовге та нескладне орендування та здання машини на парковку).
2. Забезпечення безпеки даних (надійність збереження платіжних карток, даних з документів та історії поїздок).
3. Вчасна реалізація проєкту.

4. Забезпечення високої якості обслуговування клієнтів (зручний та інтуїтивно-логічний інтерфейс; швидка відповідь команди підтримки; легкі та змістовні покрокові інструкції).

5. Постійне вдосконалення процесів (під час реалізації та після впровадження).

4.5 Якість у процесах управління проєктами

Стратегічний процес: визначення пріоритетів концепцій стратегічного процесу, які визначають важливість для досягнення найвищої якості під час управління проєктом:

- Задоволення потреб та бажань споживачів та інших зацікавлених осіб.

Вивчення потреб користувачів: провести детальний аналіз потреб і очікувань користувачів щодо функціональності та зручності використання сайту. Це дозволить забезпечити, що розроблений продукт відповідає їхнім вимогам і очікуванням, що і вимагає замовник.

- Проєкт виконується як безліч спланованих та взаємопов'язаних підпроцесів?

Планування та виконання проєкту: розробити докладний план проєкту, який включатиме всі необхідні підпроцеси та їх взаємозв'язки.

Впроваджувати ефективні методи управління проєктом, такі як методологія Agile та фреймворк Scrum, щоб забезпечити зосередженість на результаті.

- Для досягнення цілей проєкту обов'язковим є зосередження як на якості процесу, так і на якості продукту/сервісу.

Якість продукту: встановити високі стандарти якості для розробки та випуску продукту/сервісу. Проводити регулярні тестування та оцінки якості, щоб переконатися, що сайт відповідає вимогам користувачів і замовника.

- Як проєктний менеджмент відповідає за створення "середовища" управління якістю?

Управління якістю процесу: впроваджувати систему управління якістю, яка включає в себе процеси контролю якості, визначення показників продуктивності та виявлення можливостей для покращення.

- Як проєктний менеджмент відповідає за безперервнi покращення?

Безперервне покращення: забезпечувати систематичні огляди процесів та продукту для виявлення можливостей для покращення. Впроваджувати нові методики та практики для постійного удосконалення робочих процесів та виробничої діяльності.

4.6 Політика у сфері якості «U-Share»

Ми, команда «U-Share», зобов'язуємося до постійного вдосконалення та забезпечення високої якості наших послуг у сфері каршерінгу. Наша політика у сфері якості відповідає наступним принципам:

1. Відповідність цілям та контексту організації:

- Ми розробляємо та впроваджуємо наш проєкт з урахуванням стратегічних цілей та контексту нашої діяльності.

2. Основа для постановки та перегляду цілей у сфері якості:

- Наша політика становить основу для встановлення та перегляду цілей у сфері якості, спрямованих на досягнення високої якості продукту та задоволення потреб наших клієнтів.

3. Зобов'язання до виконання застосовних вимог:

- Ми зобов'язуємося виконувати всі застосовні вимоги стандартів якості та регуляторів у сфері каршерінгу.

4. Постійне покращення системи управління якістю:

- Ми прагнемо до постійного покращення нашої системи управління якістю шляхом впровадження нових технологій, процесів та методів роботи.

Асоціація «U-Share» забезпечує високу якість послуг з каршерінгу, щоб забезпечити максимальне задоволення користувачів та стимулювати розвиток цього напрямку транспортного обслуговування. Ми прагнемо до найвищих стандартів безпеки, зручності та ефективності, дотримуючись найсуворіших вимог щодо збереження та захисту особистих даних користувачів. Наша організація зобов'язується постійно вдосконалювати свої процеси та пропонувати інноваційні рішення для поліпшення якості сервісу «U-Share».

4.7 Цілі у сфері якості «U-Share»

1. Забезпечення безпеки та захисту даних:

- Специфічна: запровадити двофакторну аутентифікацію для всіх користувачів платформи «U-Share» до 1 квітня 2024 року.

- Вимірювана: збільшити рівень безпеки платформи, зменшивши кількість випадків несанкціонованого доступу на 30% до кінця 2024 року.

- Досяжна: впровадити механізми моніторингу та аналізу активності користувачів для виявлення потенційних загроз безпеці.

- Реалістична: провести навчання персоналу щодо правил обробки та захисту конфіденційної інформації до 1 жовтня 2024 року.

- Часова обмеженість: закінчити впровадження нових заходів безпеки до 31 березня 2024 року.

2. Покращення обслуговування клієнтів:

- Специфічна: зменшити час очікування відповіді на запити користувачів до 5 хвилин до кінця 2024 року.

- Вимірювана: збільшити задоволеність клієнтів обслуговуванням на 20% до кінця 2024 року за результатами опитувань.

- Досяжна: організувати додаткові тренінги для служби підтримки клієнтів для підвищення їхньої ефективності та професійної компетентності.

- Реалістична: розробити та впровадити систему автоматизованої обробки запитів користувачів до 1 серпня 2024 року.

- Часова обмеженість: переглянути та вдосконалити процеси обробки запитів користувачів протягом останніх шести місяців 2024 року.

3. Покращення стану транспортного засобу:

- Специфічна: зменшити середній вік транспортних засобів в флоті «U-Share» до 8 років до кінця 2024 року.

- Вимірювана: підвищити рівень задоволеності користувачів станом автомобілів на 25% за результатами опитувань, проведених до кінця 2024 року.

- Досяжна: провести систематичні технічні перевірки та планове технічне обслуговування для всього автомобільного флоту відповідно до регламентованих строків та стандартів безпеки.

- Реалістична: укласти договори з автосалонами щодо регулярного оновлення транспортного засобу та забезпечення його якості та надійності.

- Часова обмеженість: запровадити програму заміни старих транспортних засобів на нові до 31 грудня 2024 року та підтримувати середній вік флоту на необхідному рівні після цього.

4. Постійне покращення системи якості:

- Специфічна: провести внутрішній аудит системи управління якістю та ідентифікувати пріоритетні напрями покращень до 1 серпня 2024 року.

- Вимірювана: підвищити ефективність системи управління якістю на 15% до кінця 2024 року, враховуючи показники внутрішніх аудитів та рівень задоволеності клієнтів.

- Досяжна: запровадити систему внутрішнього контролю за процесами управління якістю та створити план дій щодо покращення слабких сторін системи.

- Реалістична: розробити та впровадити план заходів щодо покращення системи управління якістю до 1 жовтня 2024 року.

- Часова обмеженість: провести оцінку ефективності впроваджених змін до кінця 2024 року та внести необхідні коригування.

Ці цілі відображають прагнення команди до високої якості та задоволення потреб клієнтів у проєкті «U-Share».

4.8 Управління документацією та протоколами

1. Вступ

У цьому документі викладена процедура управління документацією в системі управління якістю організації «U-Share», спрямована на забезпечення ефективного контролю, збереження та доступу до документів та протоколів.

2. Сфера застосування управління документацією

Система управління документацією охоплює всі аспекти документації, пов'язаної з процесами каршерінгу в проєкті «U-Share». Ця система відповідає міжнародним стандартам якості та регуляторам у сфері каршерінгу.

3. Процедура управління документацією. За всі процеси зберігання, сортування та знищення відповідних документів відповідає проєктний менеджер з якості компанії.

3.1. Документація, що включає документи політики та цілей у сфері якості, керівництво з управління якістю, процедури та записи, що

стосуються діяльності «U-Share», ретельно розробляється та затверджується керівництвом організації.

3.2. Кожен співробітник, що займається процесами каршерінгу, має право вносити пропозиції щодо змін у документацію. Зміни до документів системи управління якістю можуть бути внесені тільки після затвердження керівництвом «U-Share».

3.3. Персонал організації «U-Share», залучений до процесів каршерінгу, зобов'язаний ознайомитися з усією документацією системи управління якістю та змінами до неї, що відбувається після затвердження змін керівництвом.

3.4. Усі зовнішні документи, що надходять від інших організацій та інформаційних джерел, реєструються та зберігаються згідно з внутрішніми вимогами «U-Share».

3.5. Система зберігання документації та протоколів розроблена згідно з системою кодування папок та забезпечує легкий доступ до неї для всіх співробітників «U-Share».

3.6. Застарілі або неактуальні документи зберігаються окремо від діючих документів та підлягають відповідній обробці та знищенню згідно з процедурами організації.

3.7. Система управління документацією забезпечує конфіденційність інформації, зібраної в ході діяльності організації, на всіх її рівнях проведення.

Цей документ введений в дію з метою забезпечення високої якості керування документацією та протоколами в організації «U-Share» та є обов'язковим для виконання всіма співробітниками, що займаються процесами каршерінгу.

4.9 Вимоги у сфері якості до каршерінгового проєкту

Вимоги, яким має відповідати каршерінговий проєкт «U-Share» та заходи для їх впровадження наведено в табл 4.2.

Таблиця 4.2

№	Вимога	Пункт за ISO ДСТУ 9001:2015	Заходи для впровадження вимог	Статус наявності в організації
1.	Доступ до всієї необхідної документації	4.3 "Визначення області застосування" 7.5 "Документована інформація"	Забезпечити доступ аудиторам до усіх необхідних документів, включаючи політику якості, процедури, записи про аудити та іншу відповідну документацію	+
2.	Наявність відповідних ресурсів	7.1 "Ресурси"	Забезпечити наявність необхідних ресурсів для проведення аудиту, включаючи аудиторський персонал та необхідне обладнання	Призначити аудиторський персонал

Продовження таблиці 4.2

3.	Підготовка персоналу	7.2 "Компетентність" 7.3 "Свідомість"	Підготувати персонал проєкту до проведення аудиту, включаючи ознайомлення з процедурами аудиту та вимогами стандарту	Провести захід задля ознайомлення всіх членів проєкту з процесом та постановами аудиту
4.	Проведення внутрішніх аудитів	9.2 "Оцінка внутрішнього аудиту".	Перед проведенням зовнішнього аудиту рекомендується провести внутрішні аудити для внутрішньої перевірки відповідності системи управління якістю вимогам стандарту	Провести попередній внутрішній аудит
5.	Дотримання вимог стандарту	Розділи 4 "Контекст організації" та 5 "Лідерство"	Забезпечити, щоб усі процеси та практики, що використовуються в проєкті «U-Share», відповідали вимогам стандарту ISO ДСТУ 9001:2015	+

Закінчення таблиці 4.2

6.	Аналіз результатів аудиту	9.1 "Моніторинг, вимірювання, аналіз та оцінка"	Провести аналіз результатів аудиту та вжити необхідні заходи для виправлення виявлених неузгодженостей та покращення системи управління якістю	Виконати після пройдення аудиту
7.	Документування результатів	7.5 "Документована інформація"	Документувати всі результати аудиту, включаючи виявлені невідповідності та заходи, що були прийняті для їх виправлення	Виконати після пройдення аудиту
8.	Проведення післяаудиторських заходів	Розділ 10 "Покращення"	Провести післяаудиторські заходи для перевірки ефективності вжитих заходів та переконання відповідності системи управління якістю вимогам стандарту після внесення змін	Виконати після пройдення аудиту

4.10 Вплив на ресурси каршерінговим проектом «U-Share»

Проект має не тільки відповідати загальним вимогам якості, а й вимогам у різних сферах, які виділяються і впливають на якість продукту/сервісу. Вплив «U-Share» показано в таблиці 4.3.

Таблиця 4.3

Сфера	Високий	Середній	Низький
Енергетичні ресурси	X		
Водні ресурси			X
Паливні ресурси	X		
Забруднення ґрунту		X	
Забруднення повітря	X		
Логістика	X		
Хімічні відходи, побутова хімія			X
Сміття, відходи		X	

Запобіжні заходи для зменшення впливу каршерінгового проекту на енергетичні ресурси (електромобілі):

1. Використання електромобілів з високою енергоефективністю та низьким рівнем викидів.
2. Запровадження системи ефективного управління зарядкою, яка б регулювала споживання електроенергії з урахуванням пікових навантажень.
3. Використання відновлюваних джерел енергії для зарядки електромобілів, таких як сонячні батареї або вітрові установки.

Запобіжні заходи для зменшення впливу на паливні ресурси:

1. Запровадження гібридних та електричних автомобілів, що вимагають менше палива або його повного відсутності.

2. Стимулювання використання автомобілів з високою паливною ефективністю та низьким рівнем викидів.

Запобіжні заходи для зменшення забруднення повітря:

1. Використання автомобілів з екологічно чистими двигунами або електромобілів, що не викидають шкідливих речовин.

2. Впровадження програм збільшення обсягу електротранспорту та зменшення кількості автомобілів з ДВЗ на дорогах.

Запобіжні заходи для зменшення впливу на логістику:

1. Оптимізація маршрутів та розкладів для зменшення кількості поїздок та витрат на транспортування.

2. Використання технологій оренди автомобілей для зменшення кількості порожніх пробігів та оптимізації використання ресурсів.

ВИСНОВОК

Проект «U-Share» – це проект в області каршерінгу, метою якого є надання жителям України можливості брати в короткострокову оренду автомобілі. В ході цього проекту було розглянуто кілька важливих етапів та аспектів

Основні досягнення проекту:

1. Розробка та впровадження платформи:

«U-Share» створює та впроваджує просту у використанні платформу, за допомогою якої користувачі можуть легко знаходити, бронювати та орендувати автомобілі. Інтуїтивно зрозумілий інтерфейс і високий рівень автоматизації процесів значно спрощують використання сервісу.

2. Екологічна відповідальність:

Одним з найважливіших аспектів «U-Share» є акцент на використанні електромобілів. Це може значно зменшити викиди CO₂ в атмосферу, що позитивно впливає на навколишнє середовище. Крім того, використання електромобілів допоможе знизити залежність від традиційних джерел палива в деякій частині жителів міст.

3. Якісне управління ресурсами:

Проект передбачає використання новітніх технологій для моніторингу та управління автопарком. Це включає аналіз даних про використання автомобіля, стан акумулятора, потреби в технічному обслуговуванні та багато іншого. Це дозволяє «U-Share» забезпечувати високу надійність і безпеку своїх сервісів.

4. Високий рівень задоволеності клієнтів:

Збираючи та аналізуючи відгуки користувачів, «U-Share» може постійно вдосконалювати свої послуги. Орієнтація на клієнта – один з ключових пріоритетів компанії, здатний задовольнити потреби і очікування користувачів. Тому було проведено дуже детальний аналіз конкурентів на ринку.

5. Розширення на ринку:

«U-Share» активно працює над своєю на українському ринку. У найближчому майбутньому компанія планує пропонувати послуги в чотирьох містах країни. А згодом, за успішності проєкту, розширюватися й далі.

Проєкт «U-Share» є важливим кроком у розвитку транспортних послуг в Україні. Завдяки екологічній орієнтації та використанню передових технологій, «U-Share» має всі шанси стати лідером на каршерінговому ринку. Реалізація цього проєкту дозволить команді та користувачам змінити ситуацію з швидким переміщенням містами та областями. Це може стати початком революційних змін у транспортній сфері.

Список використаних джерел

1. [Аласичний прокат і каршерінг] [Електронний ресурс] // Режим доступу: <https://autoprokat.net.ua/ua/articles/%D0%BAlasichnij-prokat-i-karsher%D1%96ng.html>
2. [Попит на каршерінг повернувся до довоєнних показників Getmancar] [Електронний ресурс] // Режим доступу: <https://ain.ua/2023/08/10/popyt-na-karshering-povernuvsyia-do-dovoyennyh-pokaznykiv-getmancar/>
3. [PwC Assurance Transparency Report 2023] // Режим доступу: <https://www.pwc.com/ua/uk/about/pwc-assurance-transparency-report-2023.pdf>
4. [Work Breakdown Structures - Dynamics 365 Project Operations] [Електронний ресурс] // Режим доступу: <https://learn.microsoft.com/ru-ru/dynamics365/project-operations/prod-pma/work-breakdown-structures>
5. [Статут проєкту] [Електронний ресурс] // Режим доступу: <https://uk.wikipedia.org/wiki/%D0%A1%D1%82%D0%AT%D1%83%D1%82%D0%п%D1%р%D0%E%D1%94%D0%AT%D1%у>
6. [Головна сторінка] [Електронний ресурс] // Режим доступу: <https://uk.wikipedia.org>
7. Петров О.В. Сучасні тенденції розвитку каршерінгу в Україні / О.В. Петров // Економіка і управління на транспорті. – К.: – ЕУТ, 2021 – № 12. – С. 34-40.
8. Іванова Т.М. Вплив каршерінгових послуг на міську мобільність / Т.М. Іванова, С.В. Коваль // Транспортні системи і технології. – К.: – ТСТ, 2022 – № 3. – С. 55-60.
9. Мюллер Т. Каршеринг: мобільність без обмежень / Т. Мюллер // Пер. з англ. – Київ: Видавництво «Освіта», 2019. – 300 с.

10. Сміт А. Майбутнє транспорту: каршеринг як альтернатива приватним авто / А. Сміт // Пер. з англ. – Львів: Видавництво «Літопис», 2021. – 320 с.

Додатки

1. Код головної сторінки сайту

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<style>
```

```
body {font-family: Arial, Helvetica, sans-serif;}
```

```
form {border: 1px solid #f1f1f1;}
```

```
input[type=text], input[type=password] {
```

```
    width: 100%;
```

```
    padding: 12px 20px;
```

```
    margin: 8px 0;
```

```
    display: inline-block;
```

```
    border: 1px solid #ccc;
```

```
    box-sizing: border-box;
```

```
}
```

```
button {
```

```
    background-color: #4CAF50;
```

```
color: white;

padding: 14px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 100%;

}

button:hover {

    opacity: 0.8;

}

.cancelbtn {

    width: auto;

    padding: 10px 18px;

    background-color: #f44336;

}

.imgcontainer {

    text-align: center;

    margin: 24px 0 12px ;

}
```

```
img.avatar {  
    width: 20%;  
    border-radius: 0%;  
}  
  
.container {  
    padding: 16px;  
}  
  
span.psw {  
    float: right;  
    padding-top: 16px;  
}  
  
.close:hover {background: #f44336;}  
  
.popup {  
    display: none;  
    position: fixed;  
    top: 50%;  
    left: 50%;
```

```
transform: translate(-50%, -50%);

background: #f9f9f9;

padding: 20px;

border: 1px solid #ccc;

z-index: 100;

width: 80%;

max-width: 400px;

box-shadow: 0px 0px 10px rgba(0,0,0,0.1);

}

.popup form {

    width: 100%;

}

.popup input[type=text], .popup input[type=password] {

    width: calc(100% - 24px);

    margin: 8px 0;

}

.popup button {

    width: auto;

    padding: 10px 18px;
```

```
}
```

```
h2 {  
    text-align: center;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>U-Share</h2>
```

```
<form id="loginForm" action="/action_page.php">
```

```
    <div class="imgcontainer">
```

```
        
```

```
    </div>
```

```
    <div class="container">
```

```
        <label for="email"><b>Ім'я користувача</b></label>
```

```
        <input type="text" id="email" name="email" required><br><br>
```

```
        <label for="loginPassword"><b>Пароль</b></label>
```

```

    <input type="password" id="loginPassword" name="loginPassword"
required>

```

```

    <button type="button" onclick="validateLoginPassword()">Вхід</button>
</div>
</form>

```

```

<div class="container">

```

```

    <button type="button"
onclick="document.getElementById('popup').style.display='block'">Реєстрац
ія</button>

```

```

</div>

```

```

<div id="popup" class="popup">

```

```

    <h2>Форма Реєстрації</h2>

```

```

    <form id="registerForm">

```

```

        <label for="registerEmail">Ваш email:</label>

```

```

        <input type="text" id="registerEmail" name="registerEmail"
required><br><br>

```

```

        <label for="registrationPassword">Пароль:</label>

```

```

        <input type="password" id="registrationPassword"
name="registrationPassword" required><br><br>

```

```

        <label for="confirmPassword">Підтвердити пароль:</label>

```

```

    <input type="password" id="confirmPassword"
name="confirmPassword" required><br><br>

    <label for="driverLicense">Номер водійського посвідчення:</label>

    <input type="text" id="driverLicense" name="driverLicense"
required><br><br>

    <button type="button" onclick="registerUser()">Готово</button>

    <button type="button"
onclick="document.getElementById('popup').style.display='none'">Закрити<
/button>

</form>

</div>

<script>

    document.getElementById('loginForm').addEventListener('keydown',
function(event) {

        if (event.key === 'Enter') {

            event.preventDefault(); // Запобігаємо відправленню форми

            validateLoginPassword(); // Викликаємо функцію перевірки паролю

        }

    });

function validateLoginPassword() {

    var password = document.getElementById('loginPassword').value;

    if (password === "123") {

```

```
        window.location.href = "dop/button.html"; // Перенаправляємо на
наступну сторінку

    } else {

        alert("Невірний пароль! Введіть коректний пароль."); // Виводимо
повідомлення про неправильний пароль

    }

}

function registerUser() {

    var email = document.getElementById('registerEmail').value;

    var password = document.getElementById('registrationPassword').value;

    var confirmPassword =
document.getElementById('confirmPassword').value;

    var driverLicense = document.getElementById('driverLicense').value;

    if (password !== confirmPassword) {

        alert("Паролі не співпадають! Будь ласка, перевірте і спробуйте
ще раз.");

        return;

    }

    // Можна додати логіку для реєстрації користувача

    alert("Реєстрація успішна!");
```

```
// Закрити роруп
document.getElementById('popup').style.display = 'none';
}
</script>
```

```
</body>
```

```
</html>
```

2. Код меню сайту проєкта

```
<!DOCTYPE html>
```

```
<html lang="uk">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Навігація</title>
```

```
<style>
```

```
body {
```

```
    font-family: Arial, Helvetica, sans-serif;
```

```
    margin: 0;
```

```
    padding: 0;
```

```
    display: flex;
```

```
    justify-content: center;
```

```
align-items: center;

height: 100vh;

background-color: #f9f9f9;

}
```

```
.container {

display: flex;

flex-direction: column;

align-items: center;

}
```

```
button {

background-color: #4CAF50;

color: white;

padding: 14px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 200px;

font-size: 16px;

text-align: center;

}
```

```
button:hover {  
    opacity: 0.8;  
}
```

```
.bottomright {  
    position: absolute;  
    bottom: 8px;  
    right: 16px;  
    font-size: 18px;  
}
```

```
.popup {  
    display: none;  
    position: fixed;  
    top: 0;  
    left: 50%;  
    transform: translateX(-50%);  
    background: #f9f9f9;  
    padding: 20px;  
    border: 1px solid #ccc;  
    z-index: 100;
```

```
}

.close:hover {
    background: #f44336;
}

.imgcontainer {
    text-align: center;
    margin: 1px 0 0px 0;
}

img.avatar {
    width: 50%;
    border-radius: 0%;
}

</style>

</head>

<body>

<form action="/action_page.php">
```

```
<div class="imgcontainer">  
    
</div>
```

```
<div class="container">  
  <button type="button" onclick="my()">Особистий кабінет</button>  
  <button type="button" onclick="filtr()">Фільтри</button>  
  <button type="button" onclick="reservation()">Пошук авто</button>  
  <button type="button" onclick="faq()">FAQ</button>  
</div>
```

```
<div class="bottomright">  
  <button type="button" onclick="exit()">Назад</button>  
</div>
```

```
<script>  
function reservation() {  
  window.location.href = "reservation.html";  
}
```

```
function filtr() {  
    window.location.href = "filtr.html";  
}
```

```
function faq() {  
    window.location.href = "faq.html";  
}
```

```
function my() {  
    window.location.href = "my.html";  
}
```

```
function exit() {  
    window.location.href = "../start.html";  
}
```

```
</script>
```

```
</body>
```

```
</html>
```

3. Код сторінки питань та відповідей сайту

```
<!DOCTYPE html>
```

```
<html lang="uk">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Технічна підтримка</title>

  <style>

    body {

      font-family: Arial, Helvetica, sans-serif;

      margin: 0;

      padding: 0;

      background-color: #f9f9f9;

    }

    .button-container {

      margin: 20px;

      text-align: center;

    }

    .button {

      display: inline-block;

      padding: 10px 20px;

      background-color: #4CAF50;
```

```
color: white;

text-decoration: none;

border-radius: 5px;

margin-bottom: 10px;

}
```

```
.overlay {

display: none;

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

background: rgba(0, 0, 0, 0.7);

z-index: 999;

}
```

```
.popup {

display: none;

position: fixed;

left: 50%;

top: 50%;
```

```
transform: translate(-50%, -50%);  
  
background: white;  
  
padding: 20px;  
  
box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);  
  
z-index: 1000;  
  
}
```

```
.popup:target,  
.overlay:target + .popup {  
  
    display: block;  
  
}
```

```
.close {  
  
    position: absolute;  
  
    top: 10px;  
  
    right: 10px;  
  
    text-decoration: none;  
  
    font-size: 24px;  
  
    color: #333;  
  
}
```

```
.bottomright {
```

```
position: absolute;

bottom: 8px;

right: 16px;

font-size: 18px;

background-color: #f1f1f1;

}
```

```
.imgcontainer {

    text-align: center;

    margin: 24px 0 12px 0;

}
```

```
img.avatar {

    width: 20%;

    border-radius: 0%;

}
```

```
button {

    background-color: #4CAF50;

    color: white;

    padding: 14px 20px;

    margin: 8px 0;
```

```
border: none;

cursor: pointer;

width: 100%;

}

</style>

</head>

<body>

<form action="/action_page.php">

  <div class="imgcontainer">

  </div>

  <div class="button-container">

    <a href="#popup6" class="button">Як відкрити авто?</a>

  </div>

  <div id="popup6" class="overlay"></div>

  <div class="popup" id="popup6-content">

    <a href="#" class="close">&times;</a>

    <p>Треба лише набрати код підтвердження на панелі авто.</p>
```

```
</div>
```

```
<div class="button-container">
```

```
  <a href="#popup1" class="button">Щось трапилось під час  
поїздки?</a>
```

```
</div>
```

```
<div id="popup1" class="overlay"></div>
```

```
<div class="popup" id="popup1-content">
```

```
  <a href="#" class="close">&times;</a>
```

```
  <p>Якщо додаток не працює, не вдається завершити оренду або  
відкрити/закрити двері:</p>
```

```
  <ul>
```

Переконайтеся у додатку, що ви почали оренду (в режимі "Інспектування" або "Паркування" автомобіль не заведеться) і спробуйте ще раз.

Не забудьте тримати натиснутою педаль гальма, коли заводите автомобіль.

Іноді потрібно з зусиллям покачати кермо в різні сторони.

<p>Якщо автомобіль не заводиться або глохне:</p>

Переконайтеся у додатку, що ви почали оренду і спробуйте ще раз.

Не забудьте тримати натиснутою педаль гальма, коли заводите автомобіль.

Іноді потрібно з зусиллям покачати кермо в різні сторони.

<p>Якщо колесо спущене:</p>

Негайно зупиніться, не продовжуйте рух зі спущеним колесом!

Зв'яжіться з нами.

<p>Якщо сталося ДТП:</p>


```

    <li>Зверніться в дорожню поліцію за телефоном 102 та зв'яжіться
з нами.</li>

```

```

</ul>

```

```

</div>

```

```

<div class="button-container">

```

```

    <a href="#popup2" class="button">Як завести машину? Де ключ?</a>

```

```

</div>

```

```

<div id="popup2" class="overlay"></div>

```

```

<div class="popup" id="popup2-content">

```

```

    <a href="#" class="close">&times;</a>

```

```

    <p>Якщо в машині є кнопка "Старт-стоп", то ключ не потрібен -
заводьте машину кнопкою.</p>

```

```

    <p>Якщо кнопки "Старт-стоп" немає, то ви знайдете ключ від
машини в бардачку.</p>

```

```

</div>

```

```

<div class="button-container">

```

```

    <a href="#popup3" class="button">Чи потрібно заправляти
машину?</a>

```

```

</div>

```

```
<div id="popup3" class="overlay"></div>
```

```
<div class="popup" id="popup3-content">
```

```
<a href="#" class="close">&times;</a>
```

```
<p>Зазвичай, вам не потрібно заправляти автомобіль, ми заправляємо їх самі.</p>
```

```
<p>Вартість палива включена до наших цін.</p>
```

```
<p>Якщо під час поїздки рівень палива впаде нижче 25%</p>
```

```
<ul>
```

```
<li>Будь ласка, завітайте на заправну станцію WOG або ОККО.</li>
```

```
<li>Заправте повний бак звичайним неетильованим 95.</li>
```

```
<li>Оплата там не потрібна. І ви отримаєте бонус!</li>
```

```
</ul>
```

```
</div>
```

```
<div class="button-container">
```

```
<a href="#popup4" class="button">Які є обмеження?</a>
```

```
</div>
```

```
<div id="popup4" class="overlay"></div>
```

```
<div class="popup" id="popup4-content">
```

```
<a href="#" class="close">&times;</a>
```

```
<p>Водій</p>
```


Водій повинен мати при собі дійсне посвідчення водія

За кермом може перебувати лише зареєстрований у U-share водій

<p>Дотримуйтесь правил безпеки</p>

Не керуйте авто у нетверезому вигляді (і не використовуйте нічого, що може негативно позначитися на вашій здатності керувати автомобілем)

Кожна людина, яка знаходиться в автомобілі, повинна використовувати свій ремінь безпеки

<p>Ставтеся до автомобіля дбайливо</p>

Забороняється будь-яке буксирування

Заборонені уроки водіння, гонки та спортивні заходи

Забороняється заміна чи видалення деталей чи рідин

Забороняються будь-які дії, спрямовані на свідоме заподіяння шкоди салону або екстер'єру автомобіля

<p>Тільки приватні поїздки</p>

Автомобіль заборонено використовувати для отримання комерційної вигоди

Автомобіль не можна здавати в оренду чи суборенду

Заборонено надавати послуги таксі та будь-які інші відплатні транспортні послуги

Заборонено використовувати автомобіль для незаконної діяльності

<p>Дотримуйтесь чистоти</p>

Куріння в автомобілі заборонено!

Приберіть сміття із салону до завершення оренди

Перевозіть свійських тварин лише у спеціальних кошиках чи сумках

</div>

```
<div class="bottomright">  
    <button type="button" onclick="exit()">назад</button>  
</div>  
</form>  
  
<script>  
    function exit() {  
        window.location.href = "button.html";  
    }  
</script>  
  
</body>  
</html>  
  
4. Код сторінки фільтрів сайту  
  
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
<meta charset="UTF-8">  
  
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<style>
```

```
.dropbtn {
```

```
  background-color: #4CAF50;
```

```
  color: white;
```

```
  padding: 16px;
```

```
  font-size: 16px;
```

```
  border: none;
```

```
  cursor: pointer;
```

```
}
```

```
.dropbtn:hover, .dropbtn:focus {
```

```
  background-color: #3e8e41;
```

```
}
```

```
.input-box {
```

```
  border-box: box-sizing;
```

```
  background-image: url('searchicon.png');
```

```
  background-position: 14px 12px;
```

```
  background-repeat: no-repeat;
```

```
  font-size: 16px;
```

```
  padding: 14px 20px 12px 45px;
```

```
  border: none;
```

```
border-bottom: 1px solid #ddd;
}

.input-box:focus {outline: 3px solid #ddd;}

.dropdown {
    position: relative;
    display: inline-block;
}

.dropdown-content {
    display: none;
    position: absolute;
    background-color: #f6f6f6;
    min-width: 230px;
    overflow: auto;
    border: 1px solid #ddd;
    z-index: 1;
}

.dropdown-content a {
    color: black;
```

```
padding: 12px 16px;

text-decoration: none;

display: block;

}

.dropdown-content a:hover {background-color: #3e8e41; color: white;}

.dropdown a:hover {background-color: #ddd;}

.show {display: block;}

.buttons {

    text-align: center;

    margin-top: 10px;

}

.buttons button {

    margin: 5px;

}

.form-label {

    display: block;
```

```
margin-bottom: 5px;

font-weight: bold;

    color: #4CAF50;

    text-align: center;
}
```

```
.form-select {

    width: 100%;

    padding: 10px;

    border-radius: 5px;

    border: 1px solid #ddd;

        color: #4CAF50;

        text-align: center;
}
```

```
.imgcontainer {

    text-align: center;

    margin: 24px 0 12px 0;
}
```

```
img.avatar {

    width: 20%;

    border-radius: 0%;
```

```
}  
  
.bottomright {  
    position: absolute;  
    <!-- bottom: 8px;  
    right: 16px;  
    font-size: 18px;  
    background-color: #f1f1f1; -->  
    left: 30px;  
    bottom: 8px;  
    border: 3px solid #f1f1f1;  
}  
  
button {  
    background-color: #4CAF50;  
    color: white;  
    padding: 14px 20px;  
    margin: 8px 0;  
    border: none;  
    cursor: pointer;  
    width: 100%;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<form action="/action_page.php">
```

```
  <div class="imgcontainer">
```

```
    
```

```
  </div>
```

```
<form>
```

```
  <label class="form-label" for="transport-select">Тип транспорту</label>
```

```
  <select class="form-select" name="transport" id="transport-select">
```

```
    <option value="">-- Оберіть тип транспорту --</option>
```

```
    <option value="petersburg">Легковий</option>
```

```
    <option value="samara">Позашляховик</option>
```

```
  </select>
```

```
</form>
```

```
<form>
```

```
<label class="form-label" for="engine-select">Двигун</label>
```

```
<select class="form-select" name="engine" id="engine-select">
```

```
<option value="">-- Оберіть тип палива --</option>
```

```
<option value="diesel">Дизель</option>
```

```
<option value="electric-hybrid">Електро</option>
```

```
<option value="electric-hybrid">Гібрид</option>
```

```
</select>
```

```
</form>
```

```
<form>
```

```
<label class="form-label" for="transmission-select">Коробка</label>
```

```
<select class="form-select" name="transmission" id="transmission-select">
```

```
<option value="">-- Оберіть тип трансмісії --</option>
```

```
<option value="automatic">Автомат</option>
```

```
<option value="manual">Механіка</option>
```

```
</select>
```

```
</form>
```

```
<form>
```

```
<label class="form-label" for="transmission-select">Додаткові послуги
```

```
</label>
```

```
<select class="form-select" name="transmission" id="transmission-select">  
  <option value="">-- Додати послугу --</option>  
  <option value="automatic">Дитяче крісло</option>  
</select>  
</form>
```

```
<button type="button" onclick="reservation()">Пошук авто</button>
```

```
<div class="bottomright" >  
  <button type="" onclick="exit()">назад</button>  
</div>
```

```
<script>  
  function reservation() {  
  
    window.location.href = "reservation.html";
```

```
}  
  
function exit() {  
  
    window.location.href = "button.html";  
  
}  
  
</script>  
  
</body>  
  
</html>
```

5. Код сторінки оренди авто

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
    <meta charset="UTF-8">  
  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  
<title>Пошук</title>  
  
  
<style>  
  
.button {  
  
    background-color: #4CAF50;  
  
    color: white;  
  
    padding: 10px 20px;  
  
    margin: 8px 0;
```

```
border: none;

cursor: pointer;

width: 100%;

text-align: center;

}
```

```
.button:hover {

  opacity: 0.8;

}
```

```
.map-container {

  display: none;

  position: fixed;

  top: 72%;

  left: 50%;

  transform: translate(-50%, -50%);

  z-index: 100;

  width: 300px;

  height: 187px;

}
```

```
.bottomright {  
    position: absolute;  
    bottom: 8px;  
    right: 16px;  
    font-size: 18px;  
  
}
```

```
.imgcontainer {  
    text-align: center;  
    margin: 24px 0 12px 0;  
  
}
```

```
img.avatar {  
    width: 20%;  
    border-radius: 0%;  
  
}
```

```
.button {
```

```
background-color: #4CAF50;
color: white;
padding: 10px 20px;
margin: 8px 0;
border: none;
cursor: pointer;
width: 100%;
text-align: center;
}
```

```
.button:hover {
    opacity: 0.8;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<form action="/action_page.php">
```

```
<div class="imgcontainer">
```

```

```

```
</div>
```

```
</form>
```

```
<button class="button" onclick="openMap()">Обрати автомобіль</button>
```

```
<div id="map" class="map-container">
```

```
<div id="googleMap" style="width:100%;height:100%;"></div>
```

```
<div id="infoWindowContent">
```

```
<button class="button" onclick="reserveCar()">Забронювати</button>
```

```
</div>
```

```
<button class="button" onclick="closeMap()">Закрити</button>
```

```
</div>
```

```
<div class="bottomright" >
```

```
<button type="button" onclick="exit()">назад</button>

</div>

<script>

function exit() {

    window.location.href = "button.html";

}

</script>

<script>

var map;

function openMap() {

    document.getElementById('map').style.display = 'block';

    initMap();

}

function closeMap() {

    document.getElementById('map').style.display = 'none';

}
```

```
// оновлення карти Google

function initMap() {

    // Координати центра карти (здесь указаны координаты Лондона)

    var center = {lat: 50.4053397, lng: 30.5193513};

    if (map) {

        map = null;

    }

    // об'єкт карти

    map = new google.maps.Map(document.getElementById('googleMap'), {

        zoom: 13,

        center: center

    });

    var carMarker = new google.maps.Marker({

        position: center,

        map: map,

        icon: 'https://maps.google.com/mapfiles/kml/shapes/cabs.png' //
```

```
});

// Створення інф вікна
var infoWindow = new google.maps.InfoWindow({
    content: document.getElementById('infoWindowContent')
});

// Інформаційне вікно
carMarker.addListener('click', function() {
    infoWindow.open(map, carMarker);
});
}

// Резерв машини
function reserveCar() {
    alert('Автомобіль заброньований на годину!');
    alert('Код підтвердження N546G');
}

</script>

<!-- Загрузка Google Карт -->
```

```
<script  
src="https://maps.googleapis.com/maps/api/js?key=ВАШ_API_КЛЮЧ&call  
back=initMap" async defer></script>
```

```
</body>
```

```
</html>
```

6. Код сторінки особистого кабінету користувача

```
<!DOCTYPE html>
```

```
<html lang="uk">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Особистий кабінет</title>
```

```
  <style>
```

```
    body {  
      font-family: Arial, sans-serif;  
      background-color: #f0f0f0;  
      margin: 0;  
      padding: 0;  
    }
```

```
    .container {  
      max-width: 600px;
```

```
margin: 0 auto;

padding: 20px;

background-color: #fff;

border-radius: 8px;

box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);

}
```

```
.avatar {

display: block;

margin: 0 auto;

width: 20%;

border-radius: 00%;

}
```

```
h1 {

text-align: center;

margin-top: 20px;

}
```

```
.info {

margin-bottom: 20px;

}
```

```
.info label {  
    display: block;  
    font-weight: bold;  
    margin-bottom: 5px;  
}
```

```
.info input[type="text"] {  
    width: 100%;  
    padding: 10px;  
    border-radius: 5px;  
    border: 1px solid #ccc;  
}
```

```
.bottomright {  
    position: absolute;  
    bottom: 8px;  
    right: 16px;  
    font-size: 18px;  
}
```

```
button {
```

```
        background-color: #4CAF50;

        color: white;

        padding: 14px 20px;

        margin: 8px 0;

        border: none;

        cursor: pointer;

        width: 100%;

        border-radius: 5px;

    }

</style>

</head>

<body>

<div class="container">

    <form action="/action_page.php">

        <br>

        <h1>Особистий кабінет користувача</h1>

        <br>

        <div class="info">

            <label for="bonus">Бонусний баланс: <span>0 грн.</span></label>

            <br>

        </div>

    </div>

</body>

</html>
```

```
<div class="info">

    <label for="phone">Номер телефону: <span>
+380933562898</span></label>

</div>

<div class="info">

    <label for="driver_license">Номер посвідчення водія: <span>
КВІ491479</span></label>

</div>

<div class="info">

    <label for="email">Пошта: <span> Test@gmail.com</span></label>

</div>

<div class="bottomright">

    <button type="button" onclick="exit()">назад</button>

</div>

</form>

</div>
```

```
<script>  
    function exit() {  
        window.location.href = "button.html";  
    }  
</script>  
</body>  
</html>
```