

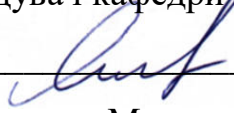
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ



к.т.н, доц. Михелєв І.Л.

" ____ " ____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "магістр"

на тему:

**«Дослідження моделей управління та розробка інформаційної підсистеми
автовокзала»**

Виконала: студентка групи 6141 м

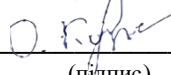

(підпис)

Стецюк В.В.

Керівник роботи:

доцент кафедри ІУСТ, к.т.н.,

(посада, науковий ступінь вчене звання)


(підпис)

Гайдаєнко О.В.

м. Миколаїв – 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій
Спеціальність 122 "Комп'ютерні науки"
Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми

В. Ажищев

к.т.н, доц. Ажищев В.Ф.

" ____ " _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ на здобуття ступеня вищої освіти "магістр"

Студенту Стецюк Валерія Володимирівна
(прізвище, ім'я, по батькові)

1. Тема роботи: Дослідження моделей управління та розробка інформаційної підсистеми автовокзала

Керівник роботи Гайдаєнко Оксана Володимирівна

Затверджені наказом ректора № 1355-уч. від "14" листопада 2025 року.

2. Термін подання роботи: 15 грудня 2025 р.

3. Вихідні дані до проєкту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем, матеріали наукового стажування

4. Перелік питань, що належать до розробки (найменування розділів)

1. Дослідження систем використання вебзастосунків для організації та контролю робочого процесу компанії. 2. Проектування архітектури та розробка концепції вебзастосунку. 3. Розробка проєктних рішень щодо вебзастосунку для організації та контролю робочого процесу компанії. 4. Техніко економічне обґрунтування розробки вебзастосунку. 5. Охорона праці. 6. Охорона навколишнього середовища. Висновки. Додатки.

5. Перелік презентаційних матеріалів Мета та об'єкт дослідження. Архітектура розробки інформаційної підсистеми. Діаграма варіантів використання. Логічна та фізична модель. Результат роботи. Технології, які використовувались для розробки сайту.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	К.т.н., доц. Гайдаєнко О.В	05.09.2024	22.12.2024
2	К.т.н., доц. Гайдаєнко О.В	23.12.2024	15.06.2025
3	К.т.н., доц. Гайдаєнко О.В	15.06.2025	25.09.2025
4	К.т.н., доц. Гайдаєнко О.В	25.09.2025	14.11.2025
5	К.т.н., доц. Гайдаєнко О.В	25.09.2025	14.11.2025
6	К.т.н., доц. Гайдаєнко О.В	25.09.2025	14.11.2025

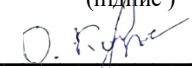
7. Дата видачі завдання 05 вересня 2024 р.**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів роботи	Примітка
1	Дослідження та аналіз предметної галузі	22.12.2024	
2	Розробка концепції	15.03.2025	
3	Розділ з розробки проєктних рішень по системі (та її частинах)	15.06.2025	
4	Реалізація проєкту та розробка робочої документації	25.09.2025	
5	Обчислення економічної ефективності	14.11.2025	
6	Вирішення питань охорони праці та навколишнього середовища	14.11.2025	
7	Оформлення пояснювальної записки	14.11.2025	
8	Подання магістерської роботи на перевірку на плагіат	16.11.2025	
9	Подання магістерської роботи рецензенту	20.11.2025	
10	Подання магістерської роботи на державний захист	15.12.2025	

Студент


 (підпис)

Керівник роботи


 (підпис)
Стецюк В.В.

(ПІБ)

Гайдаєнко О.В.

(ПІБ)

АНОТАЦІЯ

Магістерська робота містить 79 сторінок, 27 рисунків, 9 таблиць та 4 додатків. У процесі виконання було використано 18 джерел.

Актуальність теми присвячена автовокзалам, які щодня обслуговують велику кількість пасажирів, забезпечуючи планування та виконання міжміських і міжобласних перевезень. Зростання вимог до якості сервісу, необхідність оперативної обробки даних та оптимізації бізнес-процесів зумовлюють потребу в автоматизації роботи автовокзалу. Впровадження інформаційних систем дозволяє зменшити кількість ручних операцій, мінімізувати помилки, прискорити обмін інформацією та підвищити комфорт пасажирів.

Метою роботи є розробка та впровадження вебзастосунку – інформаційної підсистеми автовокзалу, яка забезпечує автоматизоване управління рейсами, бронюванням та продажем квитків, взаємодію між пасажирами, операторами й адміністраторами, а також формування звітності в режимі реального часу.

Об'єктом дослідження є процес управління діяльністю автовокзалу.

Предметом дослідження є адміністративно-інформаційна підсистема, яка забезпечує взаємодію між пасажирами, операторами та адміністраторами.

У процесі виконання кваліфікаційної роботи проведено аналіз предметної області, досліджено існуючі рішення, визначено вимоги до системи, розроблено архітектуру та концептуальну, логічну й фізичну моделі бази даних. Створено структуру вебзастосунку, діаграми варіантів використання, класів і послідовності. Запропоновано технічні, програмні, організаційні та інформаційні рішення для впровадження системи.

Кваліфікаційна робота складається зі вступу, шести розділів, висновків, списку використаних джерел та додатків.

Ключові слова: автовокзал, вебзастосунок, бронювання квитків, інформаційна система, автоматизація.

ABSTRACT

The master's thesis contains 79 pages, 27 figures, 9 tables, and 4 appendices. A total of 18 sources were used in the research process.

The relevance of the topic is determined by the need to improve the operation of bus stations, which serve a large number of passengers every day and ensure the planning and execution of intercity and interregional transportation. Increasing service quality requirements, the need for rapid data processing, and the optimization of business processes create the demand for automating bus station operations. The implementation of information systems reduces the number of manual operations, minimizes errors, accelerates information exchange, and enhances passenger comfort.

The aim of the thesis is to develop and implement a web application – an information subsystem of a bus station – that provides automated management of routes, ticket booking and sales, interaction among passengers, operators, and administrators, as well as real-time reporting.

The object of the study is the process of managing the operation of a bus station.

The subject of the study is the administrative and informational subsystem that ensures interaction among passengers, operators, and administrators.

During the research, an analysis of the subject area was conducted, existing solutions were examined, and system requirements were defined. The architecture and the conceptual, logical, and physical models of the database were developed. The structure of the web application, as well as use case, class, and sequence diagrams, were created. Technical, software, organizational, and informational solutions for system implementation were proposed.

The thesis consists of an introduction, six chapters, conclusions, a list of references, and appendices.

Keywords: bus station, web application, ticket booking, information system, automation.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1. ДОСЛІДЖЕННЯ СИСТЕМ ВИКОРИСТАННЯ	
ВЕБЗАСТОСУНКІВ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ	
РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ.....	11
1.1 Аналіз предметної області.....	11
1.2 Загальний опис вебзастосунку.....	11
1.3 Структура вебзастосунку	13
1.3.1 Аналіз існуючих рішень для розробки вебзастосунку для	
організації та контролю робочого процесу компанії	16
1.3.2 Функціональні можливості вебзастосунку	17
1.3.3 Архітектура вебзастосунку	18
1.4 Вимоги та архітектура вебзастосунку для віртуальної організації...21	
1.5 Постановка задачі та опис діяльності організації.....	23
Висновок до розділу 1.....	24
РОЗДІЛ 2. ПРОЕКТУВАННЯ АРХІТЕКТУРИ ТА РОЗРОБКА	
КОНЦЕПЦІЇ ВЕБЗАСТОСУНКУ	26
2.1 Побудова моделі варіантів використання	26
2.2 Побудова концептуальної моделі.....	28
2.3 Рішення з інформаційного забезпечення.....	30
2.3.1 Логічна модель	30
2.3.2 Фізична модель	32
2.3.3 Діаграма класів	33
2.3.4 Діаграма послідовності.....	35
2.4 Рішення з технічного забезпечення.....	36
2.5 Рішення з математичного забезпечення.....	38
2.5.1 Математична модель	38
2.6 Рішення з програмного забезпечення.....	39
Висновок до розділу 2.....	40

РОЗДІЛ 3. РОЗРОБКА ПРОЕКТНИХ РІШЕНЬ ЩОДО ВЕБЗАСТОСУНКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ.....	41
3.1 Загальносистемні рішення.....	41
3.1.1 Представлення схеми організаційної структур.....	41
3.1.2 Представлення схеми функціональної структури.....	42
3.1.3 Описання функцій, що автоматизуються.....	43
3.1.4 Загальний опис системи.....	44
3.2 Рішення з організаційного забезпечення.....	45
Висновок до розділу 3.....	46
РОЗДІЛ 4. ТЕХНІКО ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ РОЗРОБКИ ВЕБЗАСТОСУНКУ	48
4.1 Загальна характеристика роботи.....	48
4.2 Розрахунок витрат на проектування та впровадження системи.....	48
4.2.1 Техніко - економічний аналіз.....	48
4.2.2 Вартісний аналіз.....	49
4.3 Розрахунок економічної ефективності розробки і впровадження системи.....	50
Висновок до розділу 4.....	51
РОЗДІЛ 5. ОХОРОНА ПРАЦІ.....	52
5.1 Вступ.....	52
5.2 Аналіз небезпечних і шкідливих факторів.....	52
5.2.1 Освітленість робочого місця.....	52
5.2.2 Вплив випромінювання.....	54
5.2.3 Вплив електричного струму.....	54
5.2.4 Виробничий шум.....	55
5.2.5 Умови мікроклімату.....	56
5.2.6 Організація робочого місця.....	57
5.3 Розрахунок освітленості робочого місця.....	57
5.4 Розробка заходів щодо зменшення впливу небезпечних і шкідливих	

факторів.....	58
5.4.1. Заходи щодо зниження величини виникаючих зарядів статичної електрики.....	58
5.4.2 Заходи щодо зниження шуму на робочому місці.....	59
Висновок до розділу 5.....	60
РОЗДІЛ 6. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА.....	61
6.1 Вступ.....	61
6.2 Загальні відомості про забруднення.....	61
6.3 Заходи щодо запобігання забруднення навколишнього середовища	62
Висновок до розділу 6.....	63
ВИСНОВКИ.....	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	65
Додаток А Інструкція користувача.....	67
Додаток Б Технічне завдання на розробку.....	74
Додаток В Екранні форми.....	76
Додаток Г Фрагмент програмного коду	78

ВСТУП

Магістерська робота виконана на тему «Дослідження моделей управління та розробка інформаційної підсистеми автовокзалу».

Актуальність теми полягає у необхідності вдосконалення процесів управління та обслуговування пасажирів у транспортній сфері шляхом впровадження сучасних інформаційних технологій. Автоматизація автовокзалу дозволяє оптимізувати роботу персоналу, підвищити якість сервісу для користувачів та забезпечити прозорість і зручність процесів бронювання, оплати та моніторингу рейсів.

Об'єктом дослідження є процес управління діяльністю автовокзалу.

Предметом дослідження є адміністративно-інформаційна підсистема, яка забезпечує взаємодію між пасажирями, операторами та адміністраторами.

Метою роботи є підвищення ефективності управління автовокзалом за рахунок розробки вебзастосунку, який автоматизує основні бізнес-процеси, пов'язані з бронюванням квитків, формуванням розкладів і контролем перевезень.

У розділі 1 виконано аналіз предметної області, розглянуто структуру та функціональні можливості вебзастосунку, наведено аналіз існуючих систем, що дозволило визначити недоліки та сформулювати вимоги до майбутньої системи.

У розділі 2 здійснено проєктування архітектури та концепції вебзастосунку, побудовано моделі варіантів використання, концептуальну, логічну й фізичну моделі даних, діаграми класів і послідовності. Також визначено технічне, математичне та програмне забезпечення, необхідне для реалізації системи.

У розділі 3 розроблено проєктні рішення щодо вебзастосунку для організації та контролю робочого процесу автовокзалу. Подано схеми організаційної та функціональної структур, описано автоматизовані функції й наведено загальний опис системи.

У розділі 4 виконано розрахунок економічної ефективності розробки та впровадження вебзастосунку, проаналізовано витрати й обґрунтовано доцільність реалізації проєкту.

У розділі 5 розглянуто питання охорони праці під час роботи з комп'ютерним обладнанням, наведено аналіз шкідливих факторів та запропоновано заходи для покращення умов праці розробників.

У розділі 6 розглянуто питання охорони навколишнього середовища. Визначено основні джерела забруднення, пов'язані з використанням комп'ютерної техніки, та наведено рекомендації щодо зменшення негативного впливу інформаційних технологій на довкілля.

Результати роботи спрямовані на підвищення ефективності управління діяльністю автовокзалу, покращення якості обслуговування пасажирів та створення основи для подальшого розвитку інформаційних систем у транспортній галузі.

РОЗДІЛ 1 ДОСЛІДЖЕННЯ СИСТЕМ ВИКОРИСТАННЯ ВЕБЗАСТОСУНКІВ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ

1.1 Аналіз предметної області

Сучасні інформаційні технології активно впроваджуються у різні сфери діяльності людини, зокрема у транспортну галузь. Вони дозволяють підвищити ефективність роботи підприємств, автоматизувати рутинні процеси та покращити якість обслуговування користувачів. Одним із важливих напрямів цифровізації є створення вебзастосунків, які забезпечують зручний доступ до послуг у режимі онлайн.

Предметною областю є робота автовокзалу та процес онлайн-бронювання автобусних квитків.

До основних елементів належать: розклад рейсів, перевізники, пасажирів, касири, квитки та інформаційні повідомлення. Зазвичай ці процеси виконуються вручну або частково автоматизовано, що призводить до черг, затримок і складності отримання актуальної інформації.

Основна мета – спростити пошук рейсів, зробити процес бронювання швидким і зручним, забезпечити користувачів доступом до актуальних даних у будь-який час. Розробка вебзастосунку дозволяє оптимізувати взаємодію між пасажирами, перевізниками та адміністрацією автовокзалу, підвищити якість обслуговування та зменшити навантаження на персонал.

1.2 Загальний опис вебзастосунку

Вебзастосунки – це інтерактивні застосунки, що складаються з двох частин:

Одна завантажується у браузер, коли користувач набирає відповідну команду. Друга розміщується на веб-сервері, що містить усі дані, закладені у

застосунку. Він отримує запит від користувача, знаходить інформацію та відправляє у браузер.

Вебзастосунки можна використовувати у різних сферах діяльності, наприклад, для взаємодії з клієнтами або для налаштування бізнес процесів у компанії.

Зокрема, вони дозволяють нам обмін інформацією між співробітниками та клієнтами, здійснювати транзакції, продавати та рекламувати товари та послуги дистанційно, працювати з списками завдань чи нотатками, перекладати текст [1].

Існує кілька типів вебзастосунків, які можуть бути статичними або динамічними та складатися з однієї чи кількох сторінок. Вони можуть містити багато функцій і елементів дизайну сайту. Деякі з них включають у себе анімацію та систему керування змістом.

Статичні вони не є інтерактивними, вебзастосунок відображає інформацію, яка не змінюється та не оновлюється, але щоб її змінити, то потрібно відредагувати та повторно завантажити код. Статичні веб-програми завантажуються швидко, тому їх використовують для простого веб-сайту портфоліо.

Динамічні надають нову та актуальну інформацію залежно від місця, часу та користувача. Хоча динамічні вебзастосунки вимагають трохи більше часу завантаження, ніж статичні, вони компенсують цей недолік за допомогою отримання даних у реальному часі. Вони краще підійдуть для магазину з постійно змінюваним інвентарем.

Окрім статично-динамічного розподілу, вебзастосунки є односторінковими або багатосторінковими. Односторінкові завантажують всю інформацію на одній сторінці і там є кнопка до дій, де користувач відразу замовляє товар чи послугу. Це забезпечує таку перевагу, як швидкість, але також приносить проблеми, пов'язані з масштабуванням та безпекою.

А от багатосторінкові розділяють функціонал на кілька сторінок, дозволяючи краще організувати великий обсяг інформації, що спрощує

масштабування та забезпечує вищий рівень безпеки, хоча потребує більше часу на завантаження та переходи між сторінками.

У створенні вебзастосунку є деякі переваги та недоліки, які представлені в таблиці 1.1.

Таблиця 1.1 – Переваги та недоліки вебзастосунків

Переваги	Недоліки
Швидкий вихід на ринок	Немає офлайн-доступності
Економічно ефективна розробка	Обмежений доступ до функцій мобільного пристрою
Простота обслуговування та налаштування	Зниження обізнаності користувачів
Виявлення в пошуковій системі	Затримка при перезавантаженні кожної сторінки

Вебзастосунок автовокзалу – це платформа для пасажирів, яка забезпечує зручний і швидкий доступ до інформації про розклад рейсів, бронювання квитків, новини та послуги вокзалу.

Для досягнення поставленої мети постає актуальним питання розробка вебзастосунку є автоматизація роботи автовокзалу та спростити процес бронювання автобусних квитків, покращивши загальний досвід подорожей.

Цільовою аудиторією є люди, які регулярно подорожують автобусами або планують поїздки.

1.3 Структура вебзастосунку

Структура веб-сайту означає ієрархію, порядок і організацію сторінок сайту, накреслених відповідно до бажаного досвіду користувача. Він об'єднує всі веб-сторінки за допомогою навігаційної системи меню, внутрішніх посилань і вмісту.

Структура має включати всі веб-сторінки сайту, систему категорій для їх організації та засоби для переходу відвідувачів від одного елемента до іншого. Вона має гарантувати, що найважливішу інформацію можна знайти першою, запрошуючи відвідувачів продовжувати досліджувати сам додаток, включаючи такі елементи, як категорія та підкатегорія, навігація та система зв'язків.

Категорії створюють основу для структури веб-сайту, групування сторінок сайту зі схожим вмістом це є полегшенням для відвідувачів пошуку того, що їм потрібно. Багатофункціоналі сайти, які мають велику кількість категорій, повинні додатково розділити свій вміст на підкатегорії.

Навігація веб-сайту допомагає відвідувачам легко знаходити потрібний вміст. Зазвичай вона представлена у вигляді меню, яке може бути розташоване вгорі сторінки.

Добре продумана система посилань допомагає користувачам легко переміщуватися веб-сайтом. Залежно від структури сайту, відвідувачі можуть покладатися на різні типи посилань для зручності навігації.

СТА (заклики до дії) – кнопки, які мотивують користувачів виконати конкретну дію, наприклад, купити товар або підписатися.

Внутрішні посилання – з'єднують сторінки одного й того самого сайту, допомагаючи користувачам знайти додаткову інформацію або переходити між розділами.

Контекстні посилання – ведуть до контенту за межами сайту, наприклад, на сторінки партнерів, новинні статті чи блоги, що доповнюють основний вміст [2].

Структура вебзастосунку визначає основні компоненти та їх взаємодію, забезпечуючи ефективність, зручність і доступність для всіх користувачів.

Ключові елементи структури автовокзалу:

головна сторінка – тут буде розміщено логотип та навігаційне меню, яке дозволяє переходити до основних розділів: Головна, Розклад, Бронювання, Контакти та Особистий кабінет. Також на головній сторінці буде форма для пошуку рейсів, де користувачі можуть вибирати напрямок, дату та час поїздки.

У верхній частині сторінки буде панель користувача з кнопкою для входу в Особистий кабінет або реєстрації нового користувача;

сторінка розкладу рейсів містить форму фільтрації, яка дозволяє користувачам вибирати рейси за різними критеріями, такими як напрямок, дата, час відправлення та тип транспорту. Під формою фільтрації відображається список рейсів, в якому для кожного рейсу вказані деталі, зокрема час відправлення та прибуття, маршрут, а також ціна квитка. Біля кожного рейсу знаходиться кнопка "Забронювати", що дає змогу користувачу перейти на сторінку бронювання для вибору місця та оплати квитка;

на сторінці бронювання, користувач може вибрати рейс, що його цікавить, із детальною схемою автобуса, де він може вибрати вільне місце. Далі пасажир заповнює форму введення особистих даних, вказуючи своє ім'я, прізвище та контактні дані для підтвердження бронювання. Після цього йде процес оплати, після успішної оплати користувач отримує підтвердження бронювання разом із квитком;

у особистому кабінеті користувач може переглядати розділ "Мої бронювання", де відображається перелік усіх його заброньованих рейсів разом з інформацією про статус кожного бронювання (підтверджено, в очікуванні, скасовано тощо). В "Налаштуваннях профілю" є можливість змінювати контактні дані, пароль та інші налаштування;

на сторінці контактів і довідки буде розміщена форма зворотного зв'язку, яка дозволяє користувачам залишати свої запити, коментарі або скарги. Це допоможе швидко отримати відповіді на питання або вирішити проблеми;

щодо інтерактивної карти маршруту, вона надасть можливість користувачам переглядати маршрут рейсу, показавши точку відправлення, важливі зупинки та місце прибуття.

Ця структура дозволяє створити повноцінний вебзастосунок для автовокзалу з усіма необхідними функціями для користувачів. Вона забезпечує простоту у навігації, доступ до актуальної інформації про рейси та можливість швидкого бронювання квитків.

1.3.1 Аналіз існуючих рішень для розробки вебзастосунку для організації та контролю робочого процесу

Розробка вебзастосунків для організації та контролю робочого процесу автовокзалу є актуальною темою, враховуючи зростання потреби в автоматизації та зручності користувацьких сервісів. Більшість сучасних систем орієнтовані на інтеграцію розкладів, управління квитками та взаємодію з клієнтами.

Приклади вже існуючих рішень для розробки вебзастосунку автовокзалу:

FlixBus – сервіс автобусних перевезень у Європі. Його функціонал включає пошук рейсів за напрямком, датою та часом, онлайн-бронювання квитків із вибором місця, інтеграцію з платіжними системами, а також сповіщення про затримки чи зміни в розкладі. Серед особливостей сервісу – інтерактивна карта маршрутів та мобільний додаток із синхронізацією веб-версії;

Ecolines – автобусний перевізник у Східній Європі. Сервіс пропонує можливість переглядати розклад рейсів та вибирати місце в автобусі, а також користуватися особистим кабінетом для управління бронюваннями. Інтеграція з Google Maps дозволяє перевіряти маршрут. Особливості Ecolines включають підтримку різних мов і валют, а також оплату через міжнародні платіжні системи;

EasyWay – інформаційний портал громадського транспорту, який надає розклад автобусів і маршруток. Сервіс оснащений інтерактивною картою зупинок і маршрутів та інтеграцією з системами моніторингу транспорту. Його особливостями є підтримка міст і міжміських перевезень, відображення затримок у реальному часі та можливість відслідковувати рейси. Платформа підтримує кілька мов для зручності користувачів.

Infobus – призначений для пошуку і бронювання автобусних квитків. Його функціонал включає систему бронювання, яка відображає реальні місця на рейсах, а також інтеграцію з API локальних перевізників для надання актуальної інформації. Додаток генерує маршрути, що включають проміжні зупинки. Серед

особливостей — можливість відслідковувати рейси в реальному часі та підтримка кількох мов для зручності користувачів з різних країн.

Недоліки існуючих рішень:

- багато платформ не адаптовані для невеликих автовокзалів або специфічних мов;
- не всі системи підтримують розширені функції для управління персоналом чи технічними процесами;
- часто сервіси працюють окремо від адміністративних інструментів.

1.3.2 Функціональні можливості вебзастосунку

Функціональні можливості вебзастосунку є ключовими аспектами, які визначають його призначення та ефективність у задоволенні потреб користувачів.

Вебзастосунок забезпечує зручний доступ до всіх необхідних функцій для планування та здійснення подорожей, а саме:

- пошук рейсів — користувач може знаходити рейси за містом відправлення та прибуття, застосовувати фільтри за датою, часом, тривалістю подорожі та вартістю квитків;
- розклад рейсів представлений у зручному вигляді з інформацією про час відправлення, прибуття, доступні місця та вартість квитків;
- детальна інформація про маршрути — відображається список наявних автобусних напрямків із можливістю перегляду зупинок на інтерактивній карті;
- онлайн-бронювання квитків дозволяє користувачам обрати рейс і забронювати місце в автобусі (якщо це підтримується перевізником);
- вибір місця — користувач може переглядати схему автобуса та обирати місце залежно від наявності, забезпечуючи комфорт під час подорожі;
- оплата квитків здійснюється через інтегровані платіжні системи, що гарантує безпечні та швидкі фінансові операції; після оплати

користувач отримує електронний квиток або підтвердження бронювання електронною поштою чи SMS;

- особистий кабінет — користувач може створювати обліковий запис, керувати особистими даними, переглядати історію бронювань, скасовувати поїздки чи подавати запит на відшкодування відповідно до політики скасування;
- сповіщення — система інформує пасажирів про зміни в розкладі, затримки чи оновлення статусу бронювання;
- відгуки та оцінки — після поїздки користувач може залишити відгук про якість обслуговування перевізника;
- доступність — додаток працює на будь-яких пристроях (ПК, планшет, смартфон) та доступний 24/7.

Переваги вебзастосунку:

- спрощений доступ до інформації про рейси та квитки;
- підвищений рівень безпеки під час фінансових операцій;
- зменшення черг і паперової роботи;
- доступність сервісу з будь-якого пристрою у будь-який час;
- інтерактивна карта маршрутів із позначенням зупинок;
- інтеграція з платіжними системами та електронними картами.

Таким чином, функціональні можливості вебзастосунку для автовокзалу створюють зручний і ефективний інструмент для пасажирів, спрощуючи процес планування поїздки, бронювання та оплати квитків. Це рішення підвищує рівень комфорту користувачів і сприяє автоматизації роботи автовокзалу завдяки впровадженню сучасних цифрових технологій.

1.3.3 Архітектура вебзастосунку

Побудова архітектури застосунку — це комплекс заходів, який спрямований на те, щоб чітко визначити, як буде побудована система.

Загалом відокремлюють три типи верхньорівневої архітектури вебзастосунку, це:

моноліт – це архітектурний підхід, при якому всі компоненти та модулі вебзастосунку об'єднані в єдину, нерозривну систему. У монолітній архітектурі всі функціональні частини, такі як обробка запитів, бізнес-логіка, доступ до бази даних та інтерфейс користувача, працюють в рамках одного великого коду. Це означає, що зміни в одній частині системи можуть впливати на інші, що ускладнює масштабування та оновлення додатку;

мікросервіси – це архітектурний підхід, який передбачає розподіл великого додатку на кілька невеликих, автономних сервісів, кожен з яких виконує певну функцію або набір функцій. Ці сервіси взаємодіють між собою через мережу за допомогою чітко визначених інтерфейсів і протоколів;

серверлес – це архітектурний підхід, який дозволяє розробникам зосередитись на написанні коду, не турбуючись про управління серверами чи інфраструктурою. У серверлес-системах відповідальність за розгортання, масштабування та управління ресурсами повністю перекладається на хмарного провайдера.

Три типи верхньорівневої архітектури вебзастосунків представлені на рисунках 1.1, 1.2, 1.3.

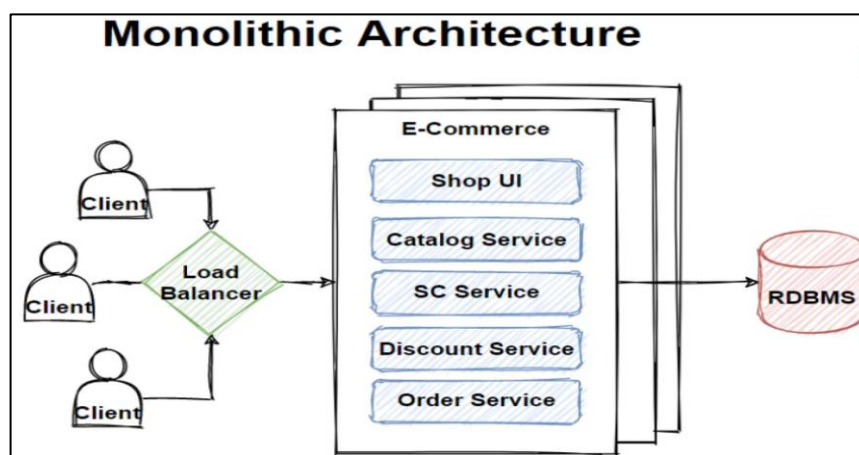


Рисунок 1.1 – Моноліт

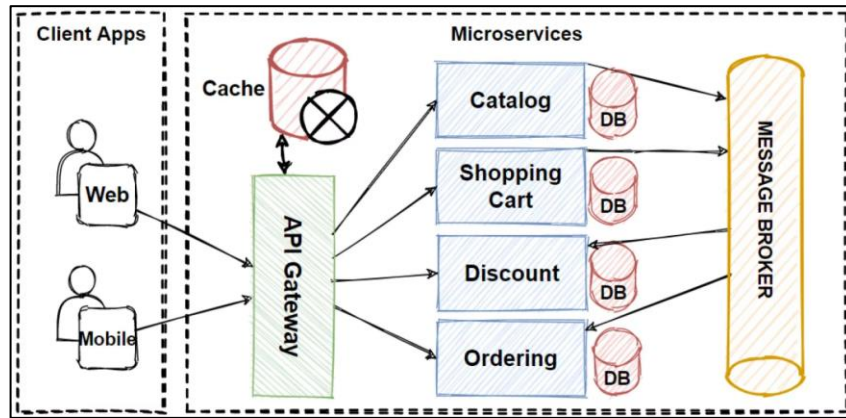


Рисунок 1.2 – Мікросервіси

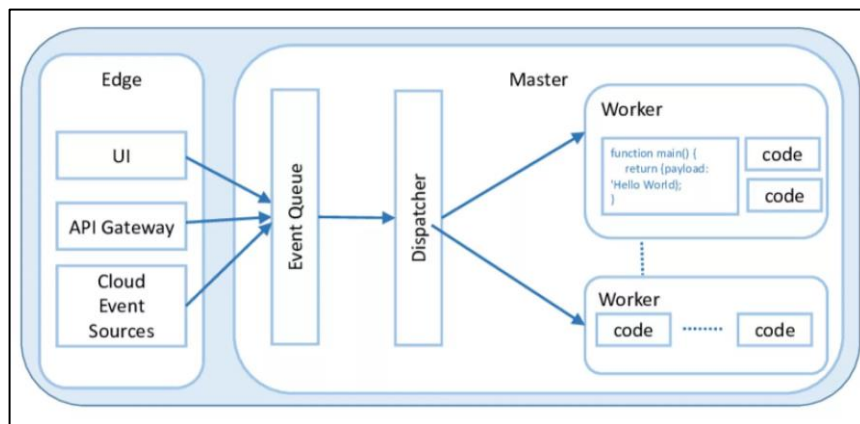


Рисунок 1.3 – Серверлесс

Архітектура вебзастосунку представлена на рисунку 1.4.

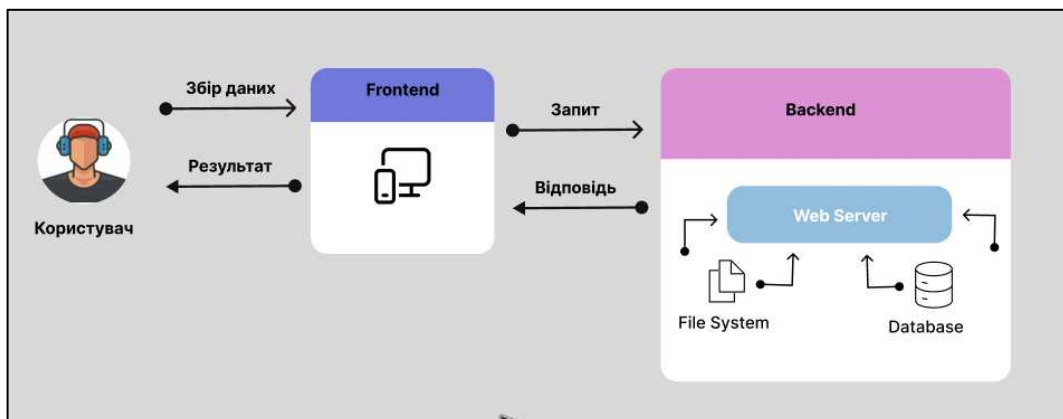


Рисунок 1.4 – Архітектура вебзастосунку

Архітектура додатку розроблена за принципами багаторівневого дизайну, що забезпечує гнучкість, масштабованість, безпеку та зручність обслуговування. Нижче наведено основні рівні архітектури:

1. клієнтська частина відповідає за взаємодію з користувачем і забезпечує динамічний інтерфейс, включаючи такі компоненти:
 - інтерфейс користувача, він є інтуїтивно - зрозумілим для перегляду розкладу, бронювання квитків, історії поїздок;
 - мобільна адаптація забезпечення доступу з різних пристроїв;
 - обробка асинхронних запитів для обміну даними з сервером у реальному часі.
2. серверна частина реалізує бізнес-логіку, обробляє запити клієнтів і працює з базою даних, включаючи такі частини:
 - управління користувачами: автентифікація, профілі;
 - обробка бронювань: створення, зміна та скасування бронювань;
 - платіжна інтеграція: підтримка онлайн-оплат ;
 - генерація розкладів і управління рейсами: обробка даних про рейси в реальному часі;
 - надсилання email, SMS або push-сповіщень.
3. база даних відповідає за зберігання структурованих даних і забезпечення їх цілісності. Дані, які зберігаються:
 - користувачі: облікові записи, історія дій;
 - рейси: маршрути, час, вартість, доступні місця;
 - бронювання: статуси (оплачено/скасовано), платіжні дані;
 - сповіщення: історія відправлених повідомлень.

Ця архітектура гарантує ефективну, безпечну та зручну роботу вебзастосунку автовокзалу для всіх учасників [3].

1.4 Вимоги та архітектура вебзастосунку для віртуальної організації

Вебзастосунок для віртуальної організації має бути побудований таким чином, щоб забезпечити ефективну взаємодію між користувачами, спрощення процесів і зручність для всіх. Далі прописані вимоги такого додатку:

1. Інтерфейс користувача:

- простий і зрозумілий дизайн з інтуїтивно зрозумілими елементами;
- адаптивність до різних пристроїв (ПК, смартфони).
- мульти-мовна підтримка для користувачів з різних країн.

2. Функціональні вимоги:

- реєстрація та авторизація користувачів;
- можливість пошуку інформації в системі (пошук по базі даних, фільтрація за різними критеріями);
- перегляд та редагування особистих даних;
- система повідомлень (сповіщення про оновлення, запити, новини).

3. Інтеграція з іншими сервісами:

- підключення до зовнішніх API для обміну даними з іншими системами;
- можливість інтеграції з платіжними системами для обробки фінансових операцій;

4. Безпека:

- Шифрування особистих даних користувачів.
- Регулярне оновлення і підтримка системи безпеки для захисту.

5. Продуктивність:

- висока швидкість завантаження сторінок і обробки запитів;
- система масштабування для обробки великої кількості користувачів одночасно.

6. Моніторинг і аналітика:

- інтеграція з аналітичними системами для відслідковування активності користувачів і використання ресурсу;
- можливість створення звітів і статистики по діяльності.

Архітектура вебзастосунку:

1. Клієнтська частина (Frontend):

- використання сучасних технологій для розробки інтерфейсу;
- взаємодія з серверною частиною;
- адаптивний дизайн для забезпечення коректної роботи на всіх пристроях.

2. Серверна частина (Backend):

- використання фреймворків та мов програмування;
- зберігання даних у базах даних;
- взаємодія з API сторонніх сервісів для інтеграції (платіжні системи, чати).

3. База даних:

- структура бази даних повинна забезпечувати швидкий доступ до інформації, підтримувати зв'язки між різними таблицями;
- використання реляційних баз даних залежно від вимог до зберігання даних.

4. Безпека:

- захист персональних даних користувача та його платіжних систем.

Ця архітектура допомагає створити ефективний вебзастосунок, який буде надавати високоякісні послуги користувачам і забезпечувати зручну та безпечну роботу для організації.

1.5 Постановка задачі та опис діяльності організації

Автовокзал є транспортним вузлом, який забезпечує перевезення пасажирів міжміськими та міжнародними автобусними маршрутами. Основними завданнями організації є розробка зручного доступу до інформації про рейси, бронювання квитків, обслуговування пасажирів та оптимізація управління операційними процесами.

З метою вдосконалення сервісу та адаптації до сучасних потреб користувачів необхідно розробити вебзастосунок, який дозволить спростити взаємодію пасажирів із системою, забезпечуючи такі ключові функції як:

- пошук і фільтрацію рейсів за різними параметрами (місто відправлення та прибуття, дата, час);
- бронювання квитків із можливістю вибору місця;
- інтеграцію з платіжними системами для здійснення онлайн-оплати;
- доступ до розкладу рейсів із відображенням актуальних даних;
- персоналізований особистий кабінет для управління бронюваннями;
- сповіщення про зміни в розкладі або статус бронювання;
- надання додаткових послуг, таких як інтерактивна карта маршрутів і інформація про доступні зупинки.

Таким чином, розробка вебзастосунку допоможе автоматизувати важливі процеси на автовокзалі та зробить обслуговування пасажирів зручнішим і швидшим. Завдяки додатку люди зможуть легко шукати рейси, бронювати квитки та оплачувати їх онлайн, не витрачаючи час на черги чи паперову роботу.

Також додаток запропонує корисні функції, як-от сповіщення про рейси, інтерактивна карта маршрутів, що зробить подорожі ще комфортнішими.

Загалом, це рішення покращить роботу автовокзалу, а пасажирів отримають сучасний та зручний сервіс для планування своїх поїздок.

Висновок до розділу 1

Вебзастосунки є важливими інструментами для полегшення доступу до інформації та автоматизації різних процесів. Вони можуть бути статичними або динамічними, а також односторінковими або багатосторінковими. Додатки для автовокзалів, зокрема, забезпечують зручне бронювання квитків, надають актуальну інформацію про розклад рейсів і доступні місця, а також забезпечують інтерактивні функції для користувачів. Архітектура таких додатків передбачає чітке розподілення компонентів для забезпечення ефективної роботи та масштабування. Вебзастосунок автовокзалу спрощує процес подорожей і покращує обслуговування клієнтів завдяки автоматизації та інтеграції сучасних технологій.

У цьому розділі було розглянуто основні аспекти розробки вебзастосунку для організації та контролю робочого процесу компанії. Спочатку було надано загальний опис вебзастосунку, а також детально розглянута його структура, зокрема функціональні можливості та архітектура. Було проведено аналіз існуючих рішень для подібних вебзастосунків, що дозволяє оцінити переваги та недоліки доступних варіантів.

Таким чином, розділ забезпечує теоретичну частину для подальшого проектування та реалізації вебзастосунку.

РОЗДІЛ 2 ПРОЕКТУВАННЯ АРХІТЕКТУРИ ТА РОЗРОБКА КОНЦЕПЦІЇ ВЕБЗАСТОСУНКУ

2.1 Побудова моделі варіантів використання

Модель варіантів використання – це модель того, як різні типи користувачів взаємодіють із системою для вирішення проблеми. Таким чином, вона описує цілі користувачів, взаємодію між користувачами та системою, а також необхідну поведінку системи для досягнення цих цілей.

Модель варіантів використання складається з низки елементів моделі. Найважливішими елементами моделі є: варіанти використання, актори та зв'язки між ними.

Зображена діаграма складається з:

- ✓ акторів – ролі акторів виконують пасажир, оператор та адміністратор автовокзалу;
- ✓ граничних меж системи – відображення всіх можливих варіантів використання вебзастосунку автовокзалу.

Відповідно до діаграми, система включає три ролі: пасажир, оператор та адміністратор.

Пасажир має можливість перейти на головну сторінку вебзастосунку, переглянути розклад рейсів, пройти реєстрацію або авторизацію. Після входу в систему пасажиру доступна функція купівлі квитків, перегляду придбаних квитків, а також скасування квитка за потреби.

Оператор має розширені права доступу. Він може додавати нові рейси до системи, редагувати наявні або скасовувати їх, переглядати статистику бронювань та продажів квитків, а також здійснювати перевірку квитків за QR-кодом або ідентифікатором.

Адміністратор виконує функції управління всією системою. Йому доступні можливості керування користувачами (додавання, редагування, блокування),

перегляду загальної звітності по роботі системи та зміни основних налаштувань: тарифів, валюти, параметрів затримки тощо.

Таким чином, діаграма варіантів використання ілюструє взаємодію кожного типу користувачів із системою та демонструє основну логіку її функціонування, це все представлено на рисунку 2.1 [4].

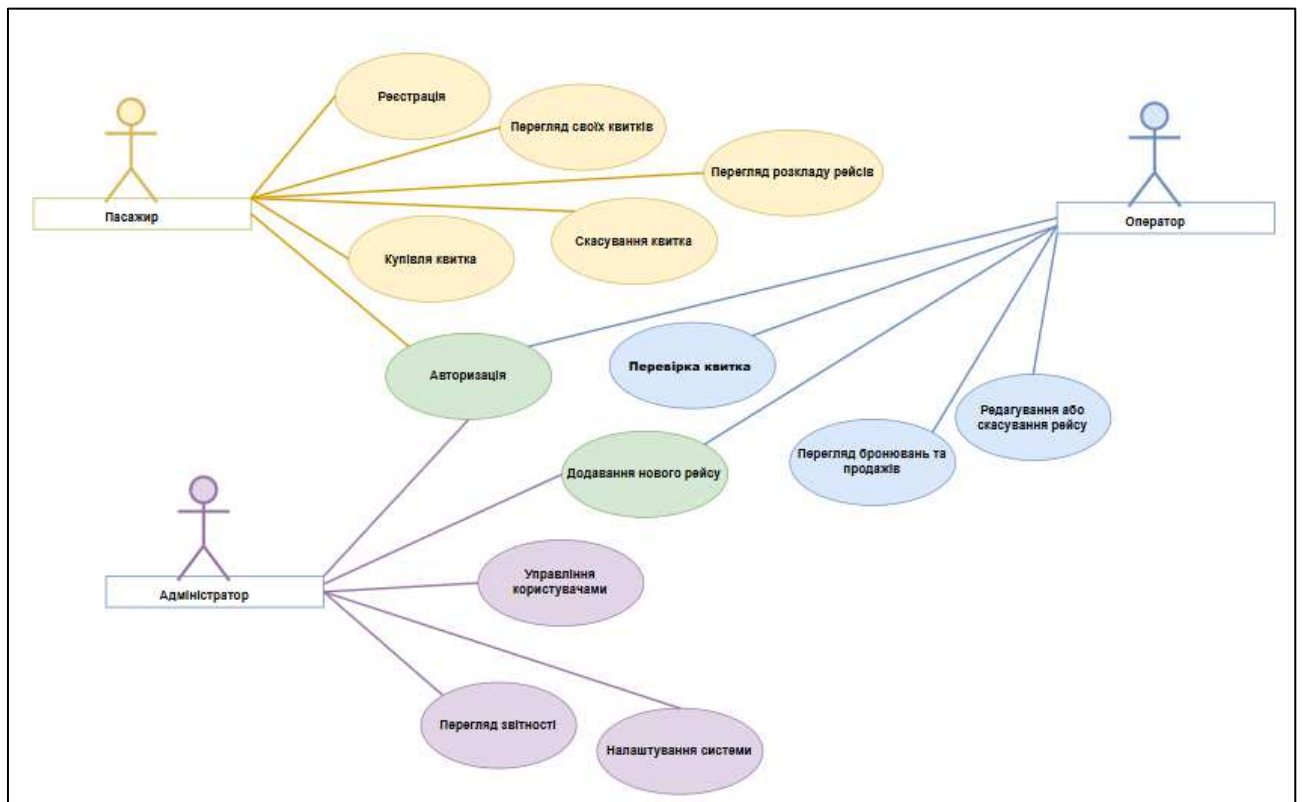


Рисунок 2.1 – Use Case Diagram

Основні варіанти використання (Use Cases):

1. для пасажирів:

- ✓ перегляд розкладу рейсів – користувач бачить список рейсів з датою, часом, маршрутом, наявністю місць;
- ✓ реєстрація – створення нового облікового запису;
- ✓ авторизація – вхід у систему для доступу до особистого кабінету;
- ✓ купівля квитка – вибір рейсу, місця, введення даних, оплата квитка;
- ✓ перегляд своїх квитків – в особистому кабінеті;
- ✓ скасування квитка – якщо передбачено політикою.

2. для оператора:

- ✓ додавання нового рейсу – формування розкладу;
- ✓ редагування або скасування рейсу – зміна параметрів або повне видалення рейсу;
- ✓ перегляд бронювань та продажів – статистика по рейсам;
- ✓ перевірка квитка – за допомогою QR-коду або ID.

3. для адміністратора:

- ✓ управління користувачами – блокування, редагування прав доступу;
- ✓ перегляд звітності – звіти про фінансові операції, статистику продажів;
- ✓ налаштування системи – управління параметрами (валюта, тарифи, час затримки).

2.2 Побудова концептуальної моделі

Під час проектування інформаційної підсистеми автовокзалу було виділено основні сутності, які беруть участь у функціонуванні системи. Між цими сутностями встановлено логічні зв'язки, що дозволяють ефективно реалізувати бізнес-процеси, зокрема бронювання квитків, керування рейсами, облік пасажирів.

Основні сутності системи:

- ✓ пасажир – зареєстрований користувач системи, який має змогу переглядати розклад, купувати квитки та керувати своїми бронюваннями;
- ✓ квиток – документ, що підтверджує право пасажирів на поїздку певним рейсом, він є пов'язаним з конкретним пасажиром, автобусом і рейсом;
- ✓ рейс – окрема поїздка, що має дату, час відправлення, маршрут і пов'язана з конкретним автобусом;
- ✓ автобус – транспортний засіб, що виконує рейси, має номер, кількість місць і тип;

- ✓ маршрут – містить інформацію про пункти відправлення та призначення, проміжні зупинки, загальний час у дорозі;
- ✓ оператор – співробітник, що створює та редагує рейси, перевіряє квитки;
- ✓ адміністратор – керує системою в цілому: правами доступу, користувачами, статистикою.

Концептуальну модель представлено на рисунку 2.2.

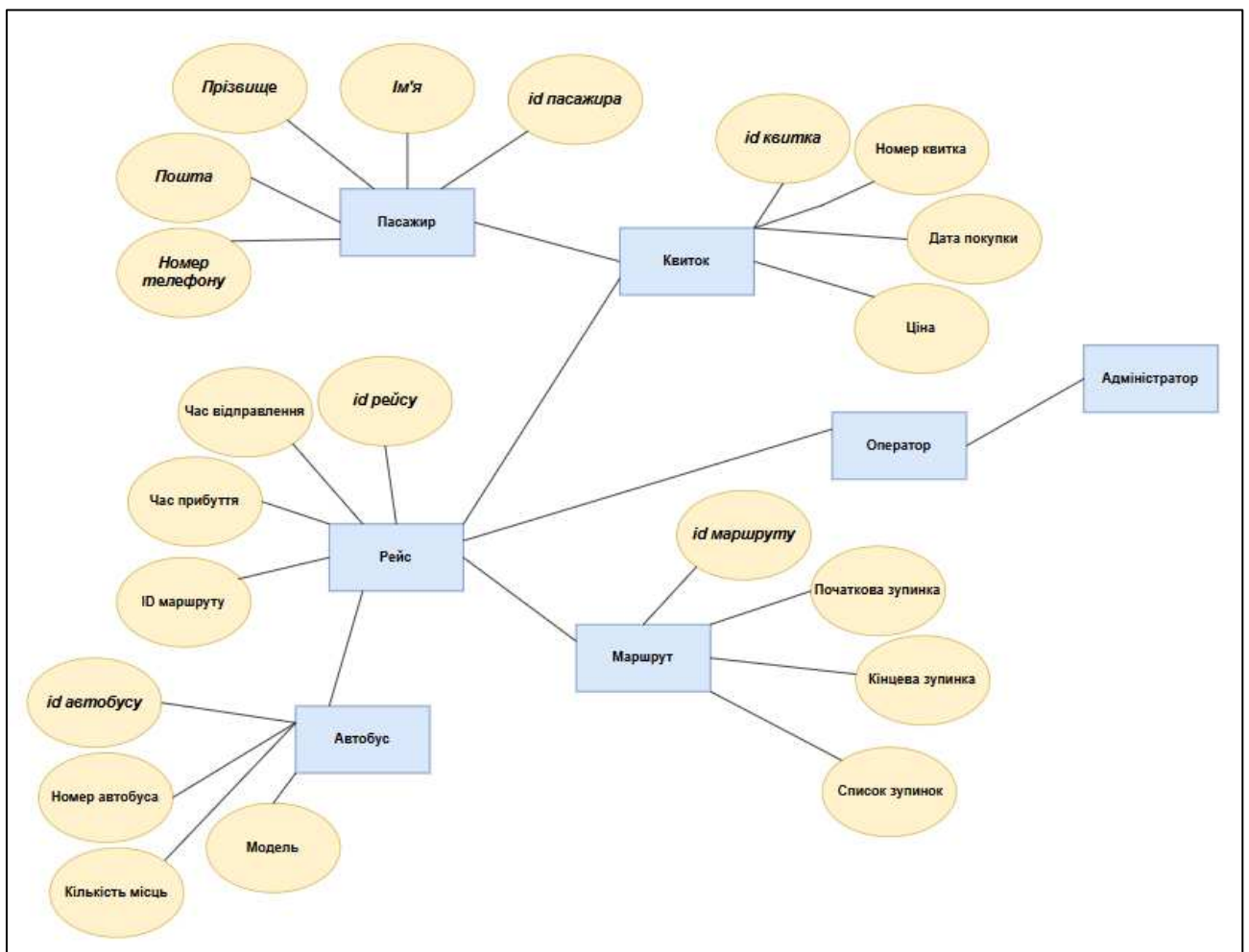


Рисунок 2.2 – Концептуальна модель

Ця модель відображає основні бізнес-процеси:

- 1) пасажир шукає маршрут, бронює місце, купує квиток, їде на рейсі;
- 2) оператор створює рейси, призначає автобуси;
- 3) адміністратор керує системою та користувачами.

2.3 Рішення з інформаційного забезпечення

2.3.1 Логічна модель

Логічна модель даних використовується для визначення структури елементів даних і встановлення зв'язків між ними. Логічна модель даних додає додаткову інформацію до елементів концептуальної моделі даних. Перевага використання логічної моделі даних полягає в тому, що вона забезпечує фундамент для створення основи для фізичної моделі.

У логічній моделі бази даних системи «Інформаційна підсистема автовокзалу» виділено основні сутності, які відображають ключові об'єкти та процеси, які представлені в таблицях 2.1, 2.2, 2.3, 2.4, 2.5, 2.6.

Таблиця 2.1

Атрибути сутності «Пасажир»

Атрибут	Опис даних
Id пасажир	Унікальний ідентифікатор пасажир
Ім'я	Ім'я пасажир
Прізвище	Прізвище пасажир
Пошта	Електронна пошта
Телефон	Номер телефону

Таблиця 2.2

Атрибути сутності «Квиток»

Атрибут	Опис даних
Id квитка	Унікальний ідентифікатор квитка
Місце	Місце відправки
Статус	(Броньований/Оплачений/Скасований)
Id рейсу	Унікальний ідентифікатор рейсу
Id пасажир	Унікальний ідентифікатор пасажир

Таблиця 2.3

Атрибути сутності «Рейс»

Атрибут	Опис даних
Id рейсу	Унікальний ідентифікатор рейсу
Id автобуса	Унікальний ідентифікатор автобуса
Id маршруту	Унікальний ідентифікатор маршруту
Час відправлення	Дата та час відправлення автобуса
Час прибуття	Дата та час прибуття автобуса
Вартість	Ціна

Таблиця 2.4

Атрибути сутності «Маршрут»

Атрибут	Опис даних
Id маршруту	Унікальний ідентифікатор маршруту
Початкова зупинка	Місце відправки
Кінцева зупинка	Місце прибуття
Тривалість	Тривалість поїздки
Відстань	Відстань між зупинками

Таблиця 2.5

Атрибути сутності «Автобус»

Атрибут	Опис даних
Id автобуса	Унікальний ідентифікатор автобуса
Номер транспорту	Номер автобусу
Модель	Модель автобусу
Кількість місць	Кількість місць в салоні автобуса

Таблиця 2.6

Атрибути сутності «Системний користувач»

Атрибут	Опис даних
Id користувача	Унікальний ідентифікатор користувача
Логін	Логін системного користувача

Пароль	Пароль системного користувача
Роль	(Адміністратор/Оператор)

Створена логічна модель представлена на рисунку 2.3

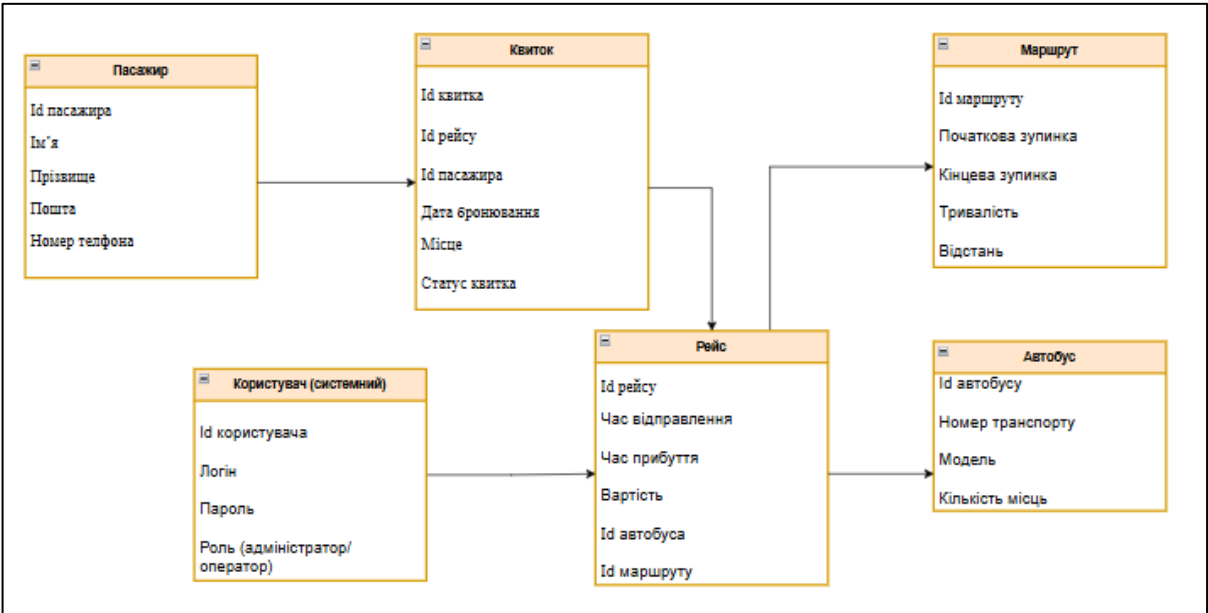


Рисунок 2.3 – Логічна модель

2.3.2 Фізична модель

Фізична модель даних описує специфічну для бази даних реалізацію моделі даних. Він пропонує абстракцію бази даних і допомагає створити схему. Це пов'язано з великою кількістю метаданих, які пропонує фізична модель даних. Фізична модель даних також допомагає візуалізувати структуру бази даних шляхом копіювання ключів стовпців бази даних, обмежень, індексів, тригерів та інших СУБД особливості [5].

Фізична модель представлена в таблиці 2.7.

Таблиця 2.7

Фізична модель даних

Ім'я об'єкта фізичної мод.	Найменування поля	Тип даних	Розмір поля (байт)	Індекс	Ключ
-------------------------------	----------------------	-----------	-----------------------	--------	------

Пасажир	id_пасажира	int	4	ІК	Первинний
	ПІБ	varchar	100		
	Телефон	varchar	15		
	Email	varchar	100		
Автобус	id_автобуса	int	4	ІК	Первинний
	Номер_транспорту	varchar	20		
	Модель	varchar	50		
	Кількість_місць	int	4		
Маршрут	id_маршруту	int	4	ІК	Первинний
	Початкова_зупинка	varchar	100		
	Кінцева_зупинка	varchar	100		
	Тривалість	time	8		
	Відстань	int	4		
Рейс	id_рейсу	int	4	ІК	Первинний
	Час_відправлення	datetime	8		
	Час_прибуття	datetime	8		
	Вартість	int	8		
	id_автобуса	int	4		Зовнішній
	id_маршруту	int	4		Зовнішній
Квиток	id_квитка	int	4	ІК	Первинний
	Дата_бронювання	datetime	8		
	Місце	int	4		
	Статус	varchar	20		
	id_рейсу	int	4		Зовнішній
	id_пасажира	int	4		Зовнішній
Користувач	id_користувача	int	4	ІК	Первинний
	Логін	varchar	50		Унікальний
	Пароль	varchar	255		
	Роль	enum			

2.3.3 Діаграма класів

Діаграма класів – це статична структурна діаграма, що демонструє властивості системи, класи, операції та зв'язки між об'єктами для опису

структури системи. На цій діаграмі не показано, коли і в якій послідовності відбуваються дії, лише структура та зв'язки.

Діаграма класів представляє статичну структуру системи бронювання автобусних квитків. Вона описує основні сутності (класи), які беруть участь у системі, їхні атрибути, методи (операції), а також зв'язки між цими класами, вона представлена на рисунку 2.4 [6].

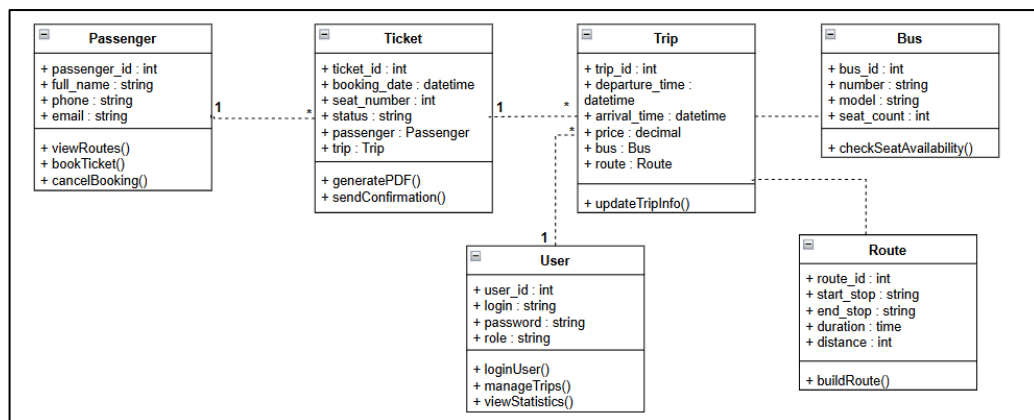


Рисунок 2.4 – Діаграма класів

Опис класів:

- 1) пасажир – представляє особу, яка користується системою для перегляду маршрутів, бронювання квитків та скасування бронювання.
 - ✓ Атрибути: ID, ПІБ, телефон, email.
 - ✓ Дії: перегляд маршрутів, бронювання квитка, скасування бронювання.
- 2) автобус – представляє транспортний засіб, який використовується для здійснення рейсів.
 - ✓ Атрибути: ID, номер, модель, кількість місць.
 - ✓ Дія: перевірка наявності вільних місць.
- 3) маршрут – описує шлях від початкової зупинки до кінцевої.
 - ✓ Атрибути: ID, початкова і кінцева зупинка, тривалість, відстань.
 - ✓ Дія: побудова маршруту.

- 4) рейс – представляє заплановану поїздку за певним маршрутом з використанням конкретного автобуса.
- ✓ Атрибути: ID, час відправлення та прибуття, вартість, пов'язаний автобус і маршрут.
 - ✓ Дія: оновлення інформації про рейс.
- 5) квиток – представляє бронювання, зроблене пасажиром на конкретний рейс.
- ✓ Атрибути: ID, дата бронювання, номер місця, статус, пов'язаний пасажир і рейс.
 - ✓ Дії: генерація PDF квитка, надсилання підтвердження.
- 6) користувач – представляє особу, яка входить у систему та має певні права (наприклад, адміністратор, оператор).
- ✓ Атрибути: ID, логін, пароль, роль.
 - ✓ Дії: авторизація, керування рейсами, перегляд статистики.

Зв'язки між класами:

Один пасажир може мати багато квитків (1:M). Пасажир може забронювати декілька квитків на різні рейси.

Один рейс може мати багато квитків (1:M). На один рейс можуть бути заброньовані квитки багатьма пасажирами.

Один користувач (наприклад, адміністратор) може керувати багатьма рейсами (1:M).

Кожен рейс має один автобус (1:1). Для кожного рейсу використовується конкретний автобус.

Кожен рейс має один маршрут (1:1). Кожен рейс здійснюється за певним маршрутом.

2.3.4 Діаграма послідовності

Діаграма послідовності (Sequence Diagram) - використовуються для моделювання логіки сценаріїв використання, показуючи інформацію, що передається між об'єктами в системі під час виконання сценарію.

У цьому прикладі розглянемо сценарій купівлі квитка пасажиром через вебзастосунок автовокзалу, який представлено на рисунку 2.5 [7].

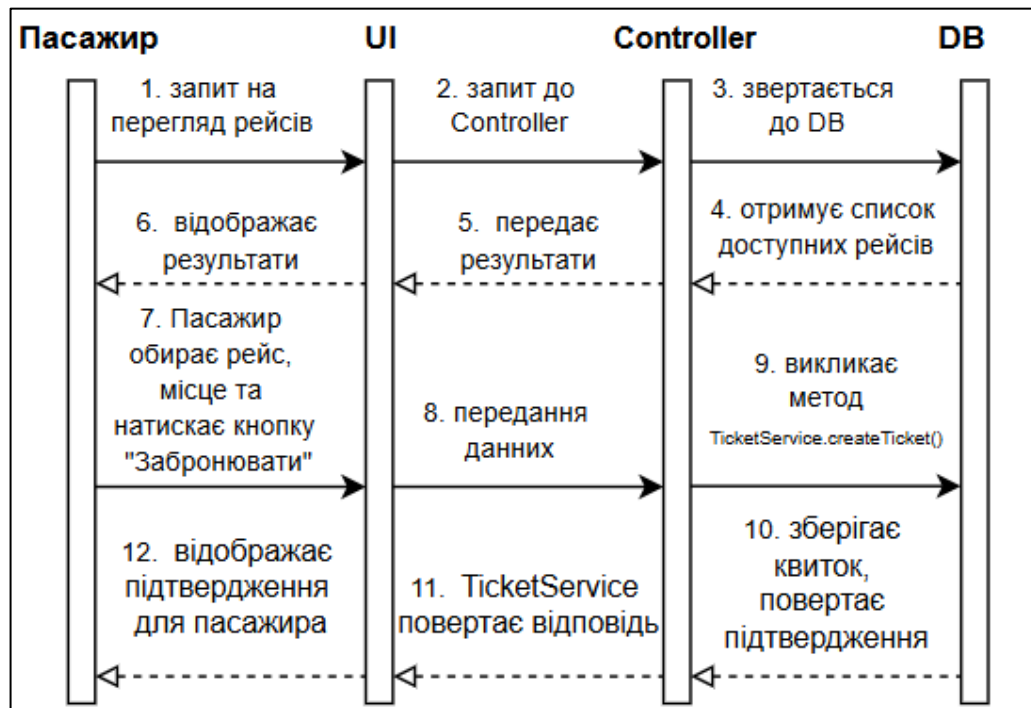


Рисунок 2.5 – Діаграма послідовності

Учасники сценарію (об'єкти):

- ✓ Пасажир.
- ✓ Інтерфейс користувача (UI).
- ✓ Сервер додатка (Controller).
- ✓ Сервіс квитків (TicketService).
- ✓ База даних (DB).

Умови сценарію:

- ✓ Рейс і місце мають бути доступними.
- ✓ Пасажир має бути авторизований.

2.4 Рішення з технічного забезпечення

Для забезпечення ефективної роботи інформаційної підсистеми автовокзалу необхідне використання таких технічних засобів:

1. серверне обладнання: він відповідає за обробку запитів від клієнтів, зберігання бази даних та логіки додатку.

Мінімальні характеристики:

- ✓ процесор: Intel Xeon або AMD Ryzen 7;
- ✓ оперативна пам'ять: 16–32 GB;
- ✓ накопичувач: SSD 512 GB або більше;
- ✓ операційна система: Linux (наприклад, Ubuntu Server), Windows Server;
- ✓ підключення до мережі: не менше 100 Мбіт/с.

2. робочі станції для персоналу: використовуються операторами, диспетчерами та адміністраторами для керування рейсами, обробки запитів, перегляду звітності.

Характеристики ПК:

- ✓ процесор: Intel Core i5 або AMD Ryzen 5;
- ✓ оперативна пам'ять: від 8 GB;
- ✓ жорсткий диск: SSD 256 GB;
- ✓ операційна система: Windows 10/11 або Linux;
- ✓ периферія: монітор, клавіатура, мишка, принтер для друку квитків.

3. мобільні пристрої: застосовуються контролерами для перевірки електронних квитків, а також пасажирями для перегляду розкладу та купівлі квитків.

Для контролерів:

- ✓ планшет або смартфон з камерою для сканування QR-кодів;
- ✓ підтримка Android 10+ або iOS 14+.

Для пасажирів:

- ✓ смартфони з доступом до Інтернету та браузером або мобільним додатком.

4. інфраструктура:

- ✓ Wi-Fi у приміщенні вокзалу для пасажирів та працівників;
- ✓ сканери квитків (QR-сканери) ;
- ✓ джерела безперебійного живлення (UPS) ;
- ✓ хмарна інфраструктура або VPS для розміщення сервера.

Архітектура розгортання: передбачається використання хмарного хостингу або віртуального сервера (VPS), що дозволяє масштабувати систему при збільшенні навантаження, забезпечує доступність 24/7 і зменшує витрати на локальне обслуговування.

2.5 Рішення з математичного забезпечення

2.5.1 Математична модель

Математична модель описує основні кількісні залежності у системі.

Позначимо:

P – кількість пасажирів, які хочуть придбати квитки на рейс;

R – загальна кількість рейсів, які здійснюються;

C – максимальна кількість квитків, які можна продати на один рейс;

S – кількість місць в автобусі для одного рейсу.

Основні умови:

- 1) кількість проданих квитків не може перевищувати кількість місць в автобусі:

$$C \leq S$$

Це гарантує, що не буде продано більше квитків, ніж реально місць в транспорті.

- 2) кількість пасажирів, які придбали квитки на рейс, не повинна перевищувати кількість доступних квитків:

$$P \leq C$$

Це означає, що кількість пасажирів не може бути більшою за кількість проданих квитків.

- 3) якщо система обслуговує R рейсів, тоді загальна кількість пасажирів за всі рейси буде:

$$P_{total} = \sum_{i=1}^R P_i$$

де P_i – кількість пасажирів на i -му рейсі [8].

2.6 Рішення з програмного забезпечення

Для реалізації вебзастосунку обрано стек технологій:

Back-end:

PHP (Laravel) – сучасний фреймворк, що забезпечує швидку розробку, безпеку, масштабованість і має вбудовані засоби для роботи з БД та авторизації користувачів.

PHP – це найпоширеніша мова програмування загального призначення з відкритим початковим кодом. PHP сконструйована спеціально для ведення веброзробок, а її код може впроваджуватися безпосередньо в HTML. PHP-скрипти виконуються на сервері і генерують HTML, яка видається клієнту. Клієнт отримуватиме результат виконання певного скрипта, але не знатиме який саме код його згенерував. Вебсервер можна налаштувати для обробки всіх файли HTML через PHP так, що клієнти навіть не зможуть дізнатися, чи використовується PHP [9].

Front-end:

HTML, CSS (Bootstrap), JavaScript (jQuery) – для створення адаптивного, зручного та інтуїтивно зрозумілого інтерфейсу користувача.

Bootstrap — це безкоштовний фреймворк з відкритим кодом для розробки фронтенд-сайтів та вебзастосунків. Розроблений для забезпечення адаптивної розробки веб-сайтів, орієнтованих на мобільні пристрої, Bootstrap надає набір синтаксисів для дизайну шаблонів.

Як фреймворк, Bootstrap містить основи адаптивної веб-розробки, тому розробникам потрібно лише вставити код у попередньо визначену систему сітки.

Фреймворк Bootstrap побудовано на мові гіпертекстової розмітки (HTML) , каскадних таблицях стилів (CSS) та JavaScript . Веб-розробники, що використовують Bootstrap, можуть створювати веб-сайти набагато швидше, не витрачаючи час на турботу про основні команди та функції [10].

База даних:

MySQL – реляційна СУБД, що добре інтегрується з Laravel, забезпечує ефективну роботу з даними та підтримує транзакції.

Інструменти для розробки:

DBeaver – для візуалізації БД, PhpStorm – для написання коду, Postman – для тестування API.

Висновок до розділу 2

У другому розділі було виконано проєктування архітектури та розробку концепції вебзастосунку для інформаційної підсистеми автовокзалу. Було побудовано модель варіантів використання, яка відображає основних користувачів системи (пасажир, оператор, адміністратор) та їхню взаємодію з системою. На основі бізнес-вимог сформовано концептуальну модель із ключовими сутностями, що забезпечують реалізацію основних функцій – реєстрації, купівлі квитків, управління рейсами та адміністративного контролю.

Крім того, розроблено логічну модель даних, яка деталізує структуру взаємопов'язаних об'єктів, та створено фізичну модель для подальшої реалізації бази даних. Завдяки побудові діаграми класів було визначено основні класи системи, їх атрибути та зв'язки. Отримані результати створюють цілісну основу для подальшої реалізації програмної частини вебзастосунку та забезпечують відповідність системи вимогам користувачів і бізнес-процесам автовокзалу.

РОЗДІЛ 3. РОЗРОБКА ПРОЕКТНИХ РІШЕНЬ ЩОДО ВЕБЗАСТОСУНКУ ДЛЯ ОРГАНІЗАЦІЇ ТА КОНТРОЛЮ РОБОЧОГО ПРОЦЕСУ КОМПАНІЇ

3.1 Загальносистемні рішення

3.1.1 Представлення схеми організаційної структур

Організаційна структура підприємства, діяльність якого підтримується розробленим вебзастосунком, є ієрархічною. На верхньому рівні знаходиться директор автовокзалу, який здійснює загальне керівництво роботою підприємства. Йому підпорядковуються:

- адміністратор інформаційної системи, у підпорядкуванні якого функціонує IT-відділ та служба технічної підтримки;
- операційний відділ, що включає диспетчерів та інспекторів контролю перевезень;
- касовий відділ, до складу якого входять старші касири та касири;
- технічний відділ, який представлений технічними спеціалістами та майстрами з обслуговування транспорту.

Така структура забезпечує чіткий розподіл ролей і відповідальностей між підрозділами, що сприяє ефективному функціонуванню автовокзалу. Розроблена інформаційна система охоплює як адміністративні, так і користувацькі процеси підприємства. На рисунку 3.1 представлено узагальнену організаційну структуру інформаційної підсистеми автовокзалу.

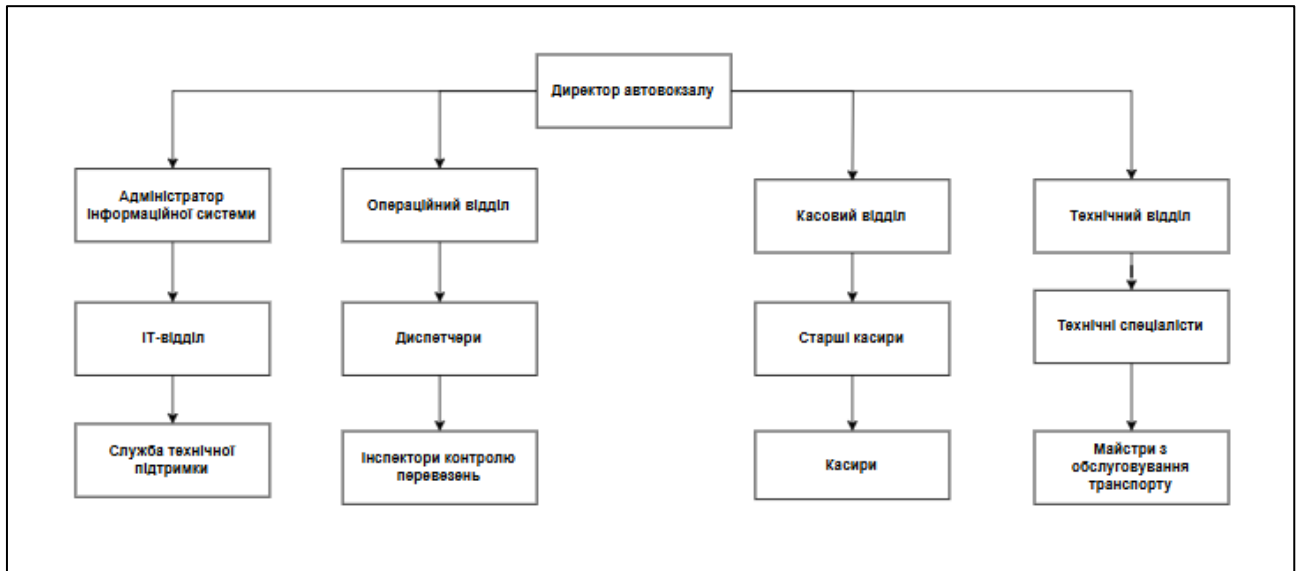


Рисунок 3.1 – Організаційна структура інформаційної системи автовокзалу

3.1.2 Представлення схеми функціональної структури

Функціональна структура вебзастосунку відображає логічний поділ системи на окремі підсистеми та модулі, які взаємодіють між собою для забезпечення безперебійної роботи інформаційної підсистеми автовокзалу. Така структура дозволяє організувати функції вебзастосунку таким чином, щоб кожен компонент відповідав за конкретний напрям діяльності та забезпечував чітку взаємодію між користувачами й системою.

До основних підсистем вебзастосунку належать:

- підсистема управління рейсами – призначена для створення, редагування та видалення рейсів, призначення автобусів і водіїв, внесення змін до графіка руху. Цей модуль забезпечує актуальність та своєчасне оновлення інформації про доступні поїздки;
- підсистема бронювання – відповідає за обробку заявок пасажирів, вибір місць у салоні, проведення онлайн-оплати та формування електронних квитків. Вона забезпечує зручність для користувачів та зменшує навантаження на касирів;
- підсистема управління користувачами – включає функції реєстрації та авторизації, визначення ролей (пасажир, оператор, адміністратор), а

також управління рівнями доступу. Це забезпечує безпеку й контроль за діями всіх учасників процесу;

- підсистема звітності та аналітики – дозволяє формувати статистичні дані щодо кількості поїздок, проданих квитків, завантаженості рейсів тощо. Аналітичні дані можуть використовуватися адміністрацією для оптимізації роботи та прийняття управлінських рішень;
- підсистема сповіщень – забезпечує автоматичне інформування пасажирів та персоналу про зміни у розкладі рейсів, затримки, підтвердження бронювання чи скасування поїздки. Вона сприяє підвищенню рівня сервісу та оперативності комунікації.

Усі підсистеми взаємодіють між собою через серверну частину, яка обробляє запити, забезпечує збереження та актуальність даних у базі, а також координує роботу всіх модулів. Така архітектура дозволяє системі функціонувати стабільно, забезпечувати швидку обробку запитів та гарантовану цілісність інформації.

На рисунку 3.4 представлено функціональна структура вебзастосунку автовокзалу.

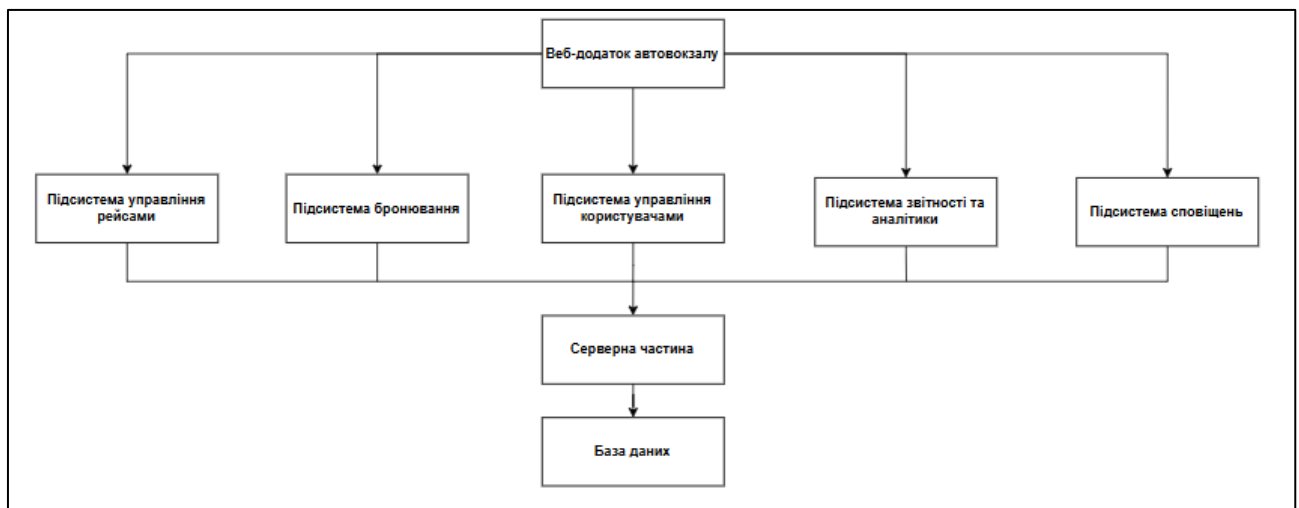


Рисунок 3.2 – Функціональна структура вебзастосунку

3.1.3 Описання функцій, що автоматизуються

Розроблений вебзастосунок автоматизує такі основні бізнес-процеси:

- формування та оновлення розкладу рейсів – оператор вводить дані про нові поїздки, зміни чи скасування рейсів;
- бронювання місць і продаж квитків – пасажери обирають напрямок, місце та оплачують квиток онлайн;
- управління користувачами – адміністратор контролює права доступу, додає або блокує користувачів;
- облік та звітність – система автоматично формує звіти за періодом, маршрутом або перевізником;
- інформування клієнтів – автоматична розсилка сповіщень про зміни у рейсах або статус бронювання;
- моніторинг стану системи – контроль за працездатністю серверів, обсягом бази даних, активністю користувачів.

Автоматизація цих функцій дозволяє мінімізувати людський фактор, підвищити швидкість обробки запитів та забезпечити зручність для користувачів.

3.1.4 Загальний опис системи

Розроблений вебзастосунок являє собою сучасну клієнт–серверну інформаційну систему, призначену для автоматизації основних бізнес-процесів автовокзалу. Система забезпечує централізоване збереження даних, зручний доступ користувачів до необхідної інформації та оперативне виконання операцій, пов'язаних із бронюванням квитків, формуванням розкладу та управлінням рейсами.

Архітектура вебзастосунку побудована за принципом розподіленої багатошарової системи. Серверна частина реалізована з використанням PHP (Laravel 11), що забезпечує надійність.

Клієнтська частина використовує HTML, CSS (Bootstrap) та JavaScript (jQuery), завдяки чому інтерфейс залишається легким, адаптивним і зручним для взаємодії з будь-якого пристрою, комп'ютера, планшета чи смартфона.

За збереженням та обробкою даних відповідає реляційна база даних MySQL, що гарантує цілісність, узгодженість та швидкий доступ до інформації. Адміністрування системи та бази даних здійснюється за допомогою інструментів DBeaver та PhpStorm.

Функціонально вебзастосунок складається з таких основних модулів:

- модуль реєстрації та авторизації користувачів, забезпечує створення облікових записів, автентифікацію та розмежування прав доступу між пасажирами, операторами та адміністраторами;
- модуль управління рейсами, призначений для введення, редагування та актуалізації інформації про рейси, автобуси та водіїв;
- модуль бронювання і продажу квитків, дозволяє пасажирам вибирати місця, оформлювати бронювання, здійснювати оплату та отримувати електронні квитки;
- модуль формування звітів, генерує зведені дані щодо кількості поїздок, продажів, завантаженості рейсів і фінансових показників;
- модуль сповіщень і повідомлень, автоматично інформує користувачів про зміни в розкладі, затримки рейсів, підтвердження або скасування бронювань.

Інтерфейс користувача розроблено з урахуванням принципів UX-дизайну: він має логічну структуру, інтуїтивну навігацію та дружній дизайн.

3.2 Рішення з організаційного забезпечення

Організаційне забезпечення системи включає комплекс заходів, спрямованих на ефективне впровадження та використання вебзастосунку в діяльності автовокзалу. До них належать:

1. Розподіл відповідальності між користувачами:

- адміністратор відповідає за технічну підтримку, створення користувачів, моніторинг системи;
- оператор забезпечує введення даних про рейси, бронювання, квитки;
- касир обслуговує клієнтів, проводить оплату квитків;
- пасажир користується функціями пошуку, бронювання та оплати онлайн.

2. Регламент роботи системи:

- створення резервних копій бази даних щодня;
- контроль журналів подій для відстеження дій користувачів;
- проведення регулярного технічного обслуговування сервера.

3. Навчання персоналу:

- проведення інструктажів з користування вебзастосунком;
- підготовка короткої інструкції користувача;
- тестування знань працівників перед впровадженням системи.

4. Підтримка і супровід системи:

- моніторинг працездатності серверної частини;
- оновлення програмного забезпечення та безпеки;
- збір відгуків користувачів для подальшого вдосконалення системи.

Завдяки цьому організаційне забезпечення гарантує безперервну роботу системи, чітку координацію між користувачами та підвищення ефективності процесів управління на автовокзалі.

Висновок до розділу 3

У третьому розділі було розроблено проектні рішення щодо вебзастосунку для організації та контролю робочого процесу автовокзалу. Розглянуто організаційну та функціональну структури системи, описано функції, які підлягають автоматизації, та надано загальну характеристику вебзастосунку.

Запропоновані рішення забезпечують ефективну взаємодію між користувачами, оптимізацію робочих процесів і підвищення якості обслуговування пасажирів. Організаційне забезпечення визначає послідовність дій під час впровадження системи, що сприяє її надійній та стабільній роботі.

РОЗДІЛ 4. ТЕХНІКО ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ РОЗРОБКИ ВЕБЗАСТОСУНКУ

4.1 Загальна характеристика роботи

Розробка вебзастосунку для інформаційної підсистеми автовокзалу спрямована на автоматизацію основних процесів: формування розкладу, бронювання квитків, оплати та управління користувачами.

Створення такого програмного продукту дозволяє зменшити навантаження на персонал, скоротити час обслуговування пасажирів і підвищити рівень сервісу.

Вебзастосунок розроблено із застосуванням сучасних технологій PHP (Laravel 11), MySQL, JavaScript (jQuery) та CSS (Bootstrap). Система функціонує у форматі «клієнт–сервер», що забезпечує безперебійну роботу, масштабованість і зручність у користуванні.

Впровадження вебзастосунку є економічно доцільним, оскільки витрати на його розробку та підтримку значно менші, ніж постійні витрати, пов'язані з ручною обробкою даних і паперовими квитками.

4.2 Розрахунок витрат на проектування та впровадження системи

4.2.1 Техніко - економічний аналіз

Перед початком розрахунків необхідно оцінити основні техніко-економічні показники, що впливають на вартість створення та впровадження системи. До них належать обсяг робіт, тривалість розробки, кількість залучених фахівців, а також витрати на використання технічних і програмних ресурсів.

Розробка вебзастосунку потребує витрат, пов'язаних із проектуванням, програмуванням, тестуванням, впровадженням і подальшим супроводом. Також до складу витрат входить оплата праці розробників, використання

електроенергії, оренда або придбання хостингу, а також амортизаційні витрати на обладнання.

Витрати на розробку вебзастосунку включають оплату праці розробників, витрати на технічне забезпечення, програмне забезпечення та супровід системи, представлено в таблиці 4.1.

Таблиця 4.1

Витрати на створення системи

№	Витрати	Вартість, грн
1	Оплата праці розробника (2 місяці *25 000грн)	50 000
2	Витрати на електроенергію та інтернет	2 000
3	Придбання та налаштування домену та хостингу	3 000
4	Програмне забезпечення	5 000
5	Амортизаційні витрати на обладнання	4 000
6	Витрати на супровід та технічну підтримку (1 місяць)	5 000
Разом		69 000

Отже, загальна вартість розробки та впровадження вебзастосунку становить 69 000 грн.

4.2.2 Вартісний аналіз

Для оцінки економічної ефективності необхідно порівняти поточні витрати автовокзалу до і після впровадження системи.

До впровадження вебзастосунку обробка замовлень здійснювалася вручну, що потребувало значних витрат часу та оплати праці додаткового персоналу.

Після впровадження системи очікується:

- зменшення кількості помилок у бронюваннях – на 30-40%;
- скорочення витрат на паперову документацію – до 10 000 грн/рік;

- економія часу операторів і касирів – до 25% робочого часу;
- зменшення навантаження на адміністрацію через автоматизацію звітності.

4.3 Розрахунок економічної ефективності розробки і впровадження системи

Для визначення доцільності впровадження вебзастосунку необхідно оцінити його економічну ефективність. Ефект від використання системи проявляється у зниженні витрат на обробку інформації, скороченні часу виконання операцій, підвищенні продуктивності праці персоналу та зменшенні кількості помилок під час бронювання й продажу квитків.

Окрім цього, автоматизація дозволяє скоротити витрати на паперову документацію, оптимізувати роботу касирів і операторів, а також зменшити потребу у додатковому персоналі.

Економічна ефективність впровадження вебзастосунку визначається за формулою:

$$E = \frac{P}{B} * 100\%$$

де, Е – економічна ефективність впровадження;

Р – прибуток або економія від упровадження, грн;

В – витрати на розробку системи, грн.

Очікувана щорічна економія (за рахунок скорочення паперових витрат і трудових ресурсів) становить 35 000 грн.

Тоді рівень ефективності:

$$E = \frac{35\,000}{69\,000} * 100\% \approx 50,7\%$$

Термін окупності проекту розраховується за формулою:

$$T = \frac{B}{P} = \frac{69\,000}{35\,000} \approx 1,97 \text{ року}$$

Отже, впровадження вебзастосунку є економічно доцільним, очікувана окупність інвестицій становить близько 2 років, після чого система почне приносити чисту економію.

Висновок до розділу 4

У цьому розділі проведено техніко-економічне обґрунтування розробки вебзастосунку для автоматизації роботи автовокзалу. Розраховано загальні витрати на створення системи, виконано економічний аналіз і визначено рівень ефективності її впровадження.

Результати показали, що проєкт є економічно вигідним, сприяє підвищенню продуктивності праці персоналу, скороченню витрат і покращенню якості обслуговування пасажирів.

РОЗДІЛ 5. ОХОРОНА ПРАЦІ

5.1 Вступ

У сучасних умовах інформатизації суспільства значна частина працівників проводить більшість робочого часу за персональним комп'ютером. Робоче місце розробника інформаційних систем належить до категорії робіт з незначними фізичними навантаженнями, проте має низку шкідливих і небезпечних факторів: електромагнітне випромінювання, шум, недостатнє освітлення, статична електрика та напруження зору.

Метою цього розділу є аналіз умов праці працівника, який займається розробкою вебзастосунків, виявлення шкідливих і небезпечних факторів, а також розроблення заходів для забезпечення безпечних умов праці.

5.2 Аналіз небезпечних і шкідливих факторів

5.2.1 Освітленість робочого місця

Освітлення має значний вплив на наше самопочуття та працездатність. Недостатня кількість світла на робочому місці може спричинити перевтому очей, головний біль і загальне виснаження. Натомість добре продумане освітлення допомагає підтримувати концентрацію, покращує настрій і зменшує напруження. Саме тому важливо ретельно обирати джерела світла та правильно розміщувати їх у робочому просторі.

Існує кілька видів освітлення, які допомагають забезпечити оптимальні умови для роботи:

1. природне освітлення — денне світло є найкориснішим для зору та загального самопочуття. Якщо є можливість, варто розташувати робочий стіл поблизу вікна. Водночас необхідно уникати прямого потрапляння сонячних променів на екран комп'ютера, щоб запобігти появі відблисків;

2. загальне освітлення – цей тип забезпечує рівномірне освітлення всього приміщення. Для цього зазвичай використовують стельові або точкові світильники, які створюють комфортне базове світлове тло;

3. локальне (робоче) освітлення – це спрямоване світло, необхідне безпосередньо на робочій поверхні. Настільні лампи або світильники з можливістю регулювання яскравості дозволяють сфокусувати освітлення там, де це потрібно найбільше.

Найкращим рішенням вважається поєднання кількох видів освітлення, це допомагає створити збалансований рівень світла та забезпечити комфорт для очей упродовж усього робочого дня.

Щоб створити правильне та зручне освітлення в офісі, варто дотримуватись кількох основних вимог:

- достатня яскравість, робоча зона має бути освітлена настільки, щоб очі не втомлювалися під час роботи. Рекомендований рівень освітленості – не менше 500 люкс;
- рівномірність світла, важливо уникати різких тіней і відблисків, які викликають дискомфорт. Освітлення має бути м'яким і рівномірно розподіленим по всій поверхні столу;
- вибір ламп, найкраще підходять LED-лампи, адже вони енергоефективні, довговічні та не мерехтять. Люмінесцентні лампи можуть створювати дискомфорт через часті коливання яскравості;
- регулювання освітлення, можливість змінювати яскравість і колірну температуру дозволяє підлаштовувати освітлення під різний час доби та тип завдань;
- колірна температура, для робочої атмосфери рекомендується нейтральне або холодне біле світло, яке допомагає підтримувати концентрацію й підвищує працездатність;
- естетика та зручність, настільна лампа має бути не лише функціональною, а й гармонійно вписуватись у дизайн інтер'єру, створюючи приємну атмосферу для роботи [11].

5.2.2 Вплив випромінювання

Робота за комп'ютером завжди пов'язана з певним впливом електромагнітного та, у меншій мірі, рентгенівського випромінювання, яке походить від монітора. Хоча сучасні рідкокристалічні (РК) екрани є значно безпечнішими порівняно зі старими електронно-променевими моніторами, вони все ж створюють певне навантаження на органи зору та нервову систему.

Щоб мінімізувати можливий негативний вплив, важливо дотримуватись встановлених санітарних норм і правил роботи за комп'ютером. Зокрема, відстань від очей до екрана має становити приблизно 50–70 см, адже це допомагає знизити напруження очних м'язів і запобігає швидкій втомі. Також частота оновлення екрана повинна бути не меншою ніж 75 Гц, щоб уникнути мерехтіння, яке може викликати подразнення очей і головний біль.

Не менш важливо робити регулярні перерви — приблизно кожні дві години безперервної роботи за комп'ютером. У цей час корисно відвести погляд від монітора, зробити кілька вправ для очей або просто трохи порухатись. Такі прості дії допомагають знизити втому, підтримують зір і сприяють збереженню загального самопочуття [12].

5.2.3 Вплив електричного струму

Усі комп'ютерні пристрої живляться від електромережі, тому існує ризик ураження електричним струмом.

Основні причини — пошкодження ізоляції проводів, використання несправних подовжувачів або перевантаження мережі.

Для запобігання небезпеці необхідно:

- встановити комп'ютер на тверду, рівну поверхню. Забезпечити щонайменше 10 см вільного простору з усіх боків, щоб повітря могло

вільно циркулювати. Недостатня вентиляція може призвести до перегрівання або навіть займання;

- не ставити пристрої надто близько один до одного та не перекривати вентиляційні отвори, тому що, це може спричинити рециркуляцію гарячого повітря;
- стежити, щоб дроти не були пережаті меблями, не лежали на проході та не створювали небезпеку спотикання. Усі кабелі повинні бути під'єднані до відповідних роз'ємів;
- не розміщувати комп'ютер на м'яких поверхнях (ліжко, диван, килим), оскільки це блокує вентиляцію;
- використовувати обладнання лише в межах рекомендованого температурного діапазону;
- не вставляти сторонні предмети у вентиляційні отвори, це може викликати коротке замикання, ураження струмом або пожежу;
- не користуватися пристроєм, підключеним до електромережі, під час грози. Якщо пристрій працює від батареї, спершу від'єднайте всі кабелі;
- уникати потрапляння води чи напоїв на обладнання. Навіть невелика кількість рідини може спричинити коротке замикання;
- перед чищенням відключити обладнання від розетки. Використовуйте м'яку тканину, злегка зволожену водою. Не застосовуйте спиртові чи аерозольні засоби — вони можуть містити легкозаймисті речовини [13].

5.2.4 Виробничий шум

Шум — це поєднання звуків різної частоти та інтенсивності, що виникають унаслідок коливань частинок у твердому, рідкому або газоподібному середовищі. Під шумом розуміють будь-який небажаний або дратівливий для людини звук, який заважає сприйняттю інформації чи викликає дискомфорт.

Однією з основних характеристик шуму є частотний склад. Якщо переважають звуки з частотою до 400 Гц, шум вважається низькочастотним. У

межах 400–1000 Гц, це середньочастотний шум. При частоті понад 1000 Гц, високочастотний шум.

Різні типи шуму по-різному впливають на організм людини. Низькочастотний шум інтенсивністю до 100 дБ зазвичай не спричиняє помітної шкоди слуху. Для середньочастотного безпечний рівень становить 85 – 90 дБ. Високочастотний шум найнебезпечніший: уже при 75 – 85 дБ він може викликати втому, дратівливість і негативно впливати на нервову систему.

Шум є особливо небезпечним тим, що його шкідливий вплив проявляється поступово і майже непомітно. Людський організм не має природного захисту від тривалого впливу шуму, тому його дія може призводити до зниження слуху, порушення сну, головного болю та загального виснаження [14].

5.2.5 Умови мікроклімату

Мікроклімат виробничих приміщень – це сукупність метеорологічних умов (температура, вологість, швидкість руху повітря, теплове випромінювання), які впливають на самопочуття та працездатність людини. Вимірюють їх на висоті 2 м від підлоги.

Оптимальні умови мікроклімату забезпечують комфорт і нормальний тепловий стан організму: температура 18–20 °С, відносна вологість 40 – 60 %, швидкість руху повітря 0,1–0,2 м/с.

Підвищена температура може спричинити перегрів або тепловий удар, прискорення серцебиття, падіння тиску, надмірне потовиділення. Низька температура, переохолодження, звуження судин, порушення кровообігу та ризик простудних захворювань.

При надмірній вологості ускладнюється випаровування поту, що знижує працездатність. Рух повітря впливає на теплообмін: слабкі потоки до 36 °С освіжають, а понад 40 °С – пригнічують. Різкі коливання параметрів мікроклімату порушують терморегуляцію організму [15].

5.2.6 Організація робочого місця

Робоче місце повинно бути організоване таким чином, щоб забезпечити комфортні, безпечні умови праці, що сприяють підтриманню високої працездатності та зниженню ризику професійного перевтомлення.

Основні параметри включають:

- висота робочого столу – 725 ± 20 мм, що забезпечує зручне розташування рук і мінімізує навантаження на спину;
- висота сидіння стільця – 420–550 мм, із можливістю регулювання під зріст користувача, бажано, щоб спинка підтримувала поперековий відділ хребта;
- екран монітора має бути розташований на рівні очей або трохи нижче, на відстані 50 – 70 см, щоб уникати перенапруження зору та шиї;
- клавіатура повинна розміщуватись на відстані 10 – 30 см від краю столу, під кутом, що забезпечує природне положення кистей;
- освітлення робочої зони має бути м'яким, без відблисків на екрані, з рівнем освітленості не менше 300 лк, бажано використовувати комбіноване природне та штучне освітлення;
- провітрювання приміщення повинно здійснюватися регулярно для підтримання оптимального мікроклімату.

Також необхідно робити короткі перерви тривалістю 10–15 хвилин кожні 2 години роботи за комп'ютером. У цей час рекомендується виконувати вправи для очей, розминку для шиї, рук і спини, що допомагає зменшити втому та покращити кровообіг.

Дотримання цих вимог сприяє збереженню здоров'я програміста, запобігає розвитку зорового і фізичного перевтомлення, а також підвищує ефективність праці [11].

5.3 Розрахунок освітленості робочого місця

Правильна організація освітлення робочого місця є важливою для будь-якого підприємства, і це зумовлено щонайменше двома причинами. По-перше, дотримання норм освітленості є законодавчо обов'язковим. Якщо під час перевірки виявлять порушення вимог, підприємство може отримати штраф, а у разі повторного порушення – більш серйозні санкції. По-друге, належне освітлення безпосередньо впливає на здоров'я і працездатність працівників, знижує втому очей і ризик виробничих травм.

Освітленість робочого місця розраховується за формулою:

$$N = \frac{E * S * Kz * z}{\Phi * \eta}$$

Де N – кількість джерел світла;

E – норма освітленості;

S – площа робочої поверхні;

Kz – коефіцієнт запасу;

z – коефіцієнт нерівномірності освітлення (обидва залежать від виду лампочки, ступеня забруднення приміщення);

Φ – світловий потік 1 джерела в Люменах (характеристика вказується на упаковці);

η – коефіцієнт використання світлового потоку (враховує висоту стін, площу об'єкта, тип світильника, на якій висоті він буде встановлений, колір покриття стелі, стін, підлоги) [16].

5.4 Розробка заходів щодо зменшення впливу небезпечних і шкідливих факторів

5.4.1. Заходи щодо зниження величини виникаючих зарядів статичної електрики

Статична електрика може накопичуватися на обладнанні, пластикових поверхнях і людському тілі, що створює ризик пошкодження електроніки та небезпеку для людини. Основні заходи поділяються на чотири групи:

1. заземлення полягає у підключенні обладнання або антистатичних виробів до землі через провідник із опором менше 10 Ом. Це дозволяє розрядити накопичений статичний заряд і запобігти його небезпечному впливу;

2. статичні ліквідатори нейтралізують заряд, генеруючи позитивні та негативні іони, які видаляють статичну електрику з поверхні обладнання або виробів;

3. антистатичні засоби, рідини або спреї наносяться на пластикові, гумові або текстильні поверхні. Основний компонент, четвертинна сіль амонію, яка підвищує вологість поверхні та збільшує її провідність. Використання таких засобів дозволяє швидко усунути накопичений статичний заряд на об'єктах;

4. електростатичне екранування захищає чутливі електронні компоненти від впливу зовнішньої статичної електрики. Це дозволяє запобігти пошкодженню обладнання при зберіганні або перевезенні;

5. контроль вологості повітря, оптимальна відносна вологість 50–65% сприяє природному розсіюванню статичної електрики. При підвищеній вологості на поверхнях утворюється тонка водяна плівка, яка знижує електричний опір. У сухому повітрі (40–50%) заряд накопичується швидше. Для підвищення вологості використовуються розпилювачі водяного туману або системи зволоження повітря, які повинні бути встановлені з урахуванням безпеки [17].

5.4.2 Заходи щодо зниження шуму на робочому місці

Шум на робочому місці може спричиняти втому, зниження концентрації, дратівливість та зменшення продуктивності. Щоб мінімізувати його вплив, використовуються такі заходи:

- використання обладнання з низьким рівнем шуму, при виборі комп'ютерної техніки, принтерів, сканерів та інших пристроїв бажано віддавати перевагу моделям, що працюють тихо. Таке обладнання зменшує загальний рівень шуму в приміщенні, створюючи комфортніші умови для роботи;
- системний блок комп'ютера можна розміщувати в спеціальних шумоізоляційних нішах або під столом, щоб знизити рівень шуму від вентиляторів і жорстких дисків. Це особливо ефективно для офісів з великою кількістю комп'ютерів;
- розміщення принтера в окремому приміщенні, принтери і багатофункціональні пристрої під час друку створюють значний шум. Перенесення їх у спеціальну кімнату або відокремлену зону дозволяє зменшити шумову навантаженість для співробітників, які працюють за комп'ютерами;
- використання звукопоглинальних матеріалів, для зниження відлуння і шуму від зовнішніх джерел застосовують килимові покриття, звукопоглинальні панелі на стінах, підвісні стелі, щільні штори. Такі матеріали зменшують відбиття звукових хвиль і створюють більш комфортну акустичну атмосферу.

Висновок до розділу 5

У п'ятому розділі було проведено аналіз умов праці працівника, визначено основні небезпечні та шкідливі фактори, а також запропоновано заходи для їх усунення або зменшення впливу.

Запропоновані рекомендації забезпечують безпечні умови праці, відповідають санітарно-гігієнічним нормам та сприяють підвищенню ефективності роботи персоналу.

РОЗДІЛ 6. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ

Охорона навколишнього середовища є важливою складовою діяльності будь-якого підприємства чи установи, зокрема й у сфері інформаційних технологій.

Хоча процес розробки вебзастосунків не пов'язаний безпосередньо з викидами шкідливих речовин чи промисловим виробництвом, інформаційна діяльність опосередковано впливає на стан довкілля через споживання електроенергії, використання комп'ютерного обладнання, матеріалів та утворення електронних відходів.

Метою цього розділу є оцінка можливого впливу діяльності в сфері розробки програмного забезпечення на навколишнє середовище та визначення заходів, спрямованих на його мінімізацію.

6.2 Загальні відомості про забруднення

Під забрудненням навколишнього середовища розуміють надходження в біосферу будь-яких твердих, рідких і газоподібних речовин або видів енергії (теплоти, звуку, радіоактивності і т. п.) у кількостях, що шкідливо впливають на людину, тварин і рослини як безпосередньо, так і непрямым шляхом.

В умовах офісної роботи та ІТ-сфери найбільший вплив на довкілля мають:

- споживання електроенергії комп'ютерним обладнанням;
- використання витратних матеріалів (папір, картриджі, батарейки);
- утворення електронних відходів (застарілі комп'ютери, монітори, периферійні пристрої);
- викиди вуглекислого газу від електростанцій, які забезпечують енергопостачання офісів і серверів.

Крім того, значний екологічний вплив мають центри обробки даних (серверні ферми), які споживають великі обсяги електроенергії для забезпечення безперервної роботи серверів і систем охолодження. Надмірне споживання енергії та нераціональне використання комп'ютерної техніки сприяють збільшенню вуглецевого сліду компаній.

Згідно з даними міжнародних екологічних досліджень, ІТ-індустрія виробляє близько 2–3% глобальних викидів вуглекислого газу, що співставно з авіаційною галуззю. Тому навіть діяльність, яка не має безпосереднього впливу на довкілля, повинна враховувати екологічні стандарти та принципи сталого розвитку [18].

6.3 Заходи щодо запобігання забруднення навколишнього середовища

Для зменшення негативного впливу ІТ-діяльності на довкілля доцільно впроваджувати комплекс заходів екологічного спрямування.

Основні з них такі:

1) енергоефективність роботи офісу, а саме:

- використання енергозберігаючого обладнання;
- вимкнення комп'ютерів та моніторів у неробочий час;
- застосування систем автоматичного керування освітленням.

2) раціональне використання ресурсів:

- перехід на електронний документообіг, що зменшує витрати паперу;
- повторне використання чорновиків для внутрішніх потреб;
- перехід на електронну звітність.

3) утилізація електронних відходів:

- передача відпрацьованої техніки спеціалізованим підприємствам для утилізації;
- окремий збір батарейок, ламп та картриджів;
- здача застарілого обладнання для вторинної переробки.

4) застосування “зелених” технологій у розробці

- оптимізація роботи серверів, що знижує споживання енергії;
- використання хмарних рішень із енергоефективних дата-центрів;
- вибір хостинг-провайдерів, які застосовують відновні джерела енергії.

Реалізація цих заходів дозволяє не лише знизити навантаження на екосистему, а й підвищити корпоративну відповідальність компанії, яка впроваджує сучасні екологічні принципи.

Висновок до розділу 6

У шостому розділі розглянуто основні джерела негативного впливу інформаційної діяльності на навколишнє середовище та запропоновано заходи для їх мінімізації.

Використання енергоефективного обладнання, зменшення паперового документообігу, впровадження екологічних стандартів і відповідальна утилізація техніки забезпечують екологічну безпеку діяльності компанії.

Отже, впровадження розробленого вебзастосунку є екологічно безпечним і відповідає вимогам сталого розвитку.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи була реалізована розробка проєкту інформаційної підсистеми автовокзалу. Проведено детальний аналіз предметної області, визначено основні проблеми в організації роботи та недоліки існуючих систем управління, що стало підґрунтям для формулювання вимог до майбутньої системи.

На основі цих вимог розроблено концепцію інформаційної підсистеми, яка забезпечує автоматизацію основних процесів діяльності автовокзалу – від пошуку рейсів і бронювання квитків до керування користувачами та рейсами.

Реалізація запропонованого проєкту дозволяє підвищити ефективність роботи автовокзалу, покращити якість обслуговування пасажирів, зменшити витрати часу персоналу та забезпечити зручність користування сервісом для всіх учасників процесу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Як створити веб-додаток: типи, переваги, принцип роботи. URL: <https://wezom.com.ua/ua/blog/kak-sozdat-veb-prilozhenie>
2. What makes a good website structure? Everything you need to know. URL: <https://www.wix.com/blog/webstructure#:~:text=A%20website%20structure%20refers%20to,menus%2C%20internal%20linking%20and%20content.>
3. Як обрати архітектуру для веб додатку. URL: <https://blog.ithillel.ua/articles/web-application-architecture>
4. Concept: Use-Case Model. URL: https://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/concepts/use_case_model_CD178AF9.html#:~:text=A%20use%2Dcase%20model%20is,system%20in%20satisfying%20these%20goals.
5. Що таке моделювання даних? Типи (концептуальний, логічний, фізичний). URL: <https://www.guru99.com/uk/data-modelling-conceptual-logical.html>
6. Що таке діаграма класів UML і найкращий творець діаграм класів UML. URL: <https://www.mindonmap.com/uk/blog/what-is-uml-class-diagram/>
7. Діаграма послідовності (Sequence Diagrams). URL: <https://www.maxzosim.com/sequence-diagrams/>
8. Стецюк В. В. ДОСЛІДЖЕННЯ МОДЕЛЕЙ УПРАВЛІННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ АВТОВОКЗАЛА. XVI Міжнародна науково-технічна конференція «Інновації в суднобудуванні та океанотехніці». Миколаїв, 2025. С. 598 – 600.
9. Що таке PHP та що з цим можна робити? URL: <https://www.php.net/manual/uk/introduction.php>
10. What is Bootstrap? URL: <https://www.techtarget.com/whatis/definition/bootstrap#:~:text=The%20Bootstrap%20framework%20is%20built,about%20basic%20commands%20and%20functions.>
11. Правильне освітлення робочого місця. URL: <https://tvoysvet.com.ua/nov-statji/pravilne-osvitlennya-robochogo-miscya>

12. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин. URL: <https://zakon.rada.gov.ua/rada/show/v0007282-98#Text>
13. Техніка безпеки при роботі з електронним та електричним обладнанням. URL: <https://surl.li/bhtacb>
14. Шум на виробництві та його вплив на організм людини. URL: <https://oppb.com.ua/news/shum-na-vyrobnyctvi-ta-yogo-vplyv-na-organizm-lyudyny>
15. Мікроклімат виробничих приміщень. URL: <https://smu.dsp.gov.ua/mikroklimat-vyrobnychkh-prymishchen/>
16. Розрахунок освітленості приміщення. URL: <https://brille.ua/rozrakhunok-osvitlenosti-prymishchennia/#osveshennost-pomesheniya>
17. Чотири заходи з ліквідації статичної електрики. URL: <https://ua.esd-antistatic.com/news/four-measures-to-eliminate-static-electricity-44546855.html>
18. Поняття забруднення, класифікація забруднень довкілля та джерела забруднення. Реферат. URL: <https://ru.osvita.ua/vnz/reports/ecology/21286/>

Додаток А

Інструкція користувача

Цей мобільний додаток призначений для зручного бронювання квитків, перегляду розкладу автобусів, оплати поїздок, отримання сповіщень та керування власним профілем. Інтерфейс інтуїтивний і пристосований для щоденного використання пасажиром.

Після запуску додатку відображається заставка з логотипом, що виконує функцію завантаження, що представлено на рисунку А.1.

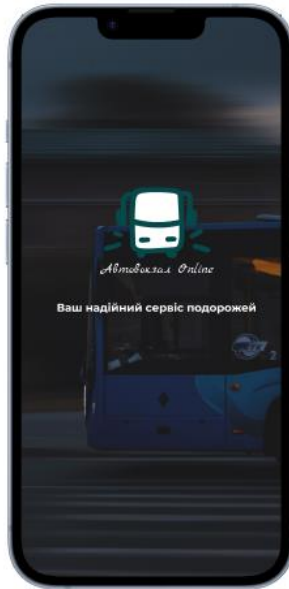


Рисунок А.1 – Заставка додатку

Далі користувач може ознайомитися з можливостями сервісу, які представлені на рисунку А.2.

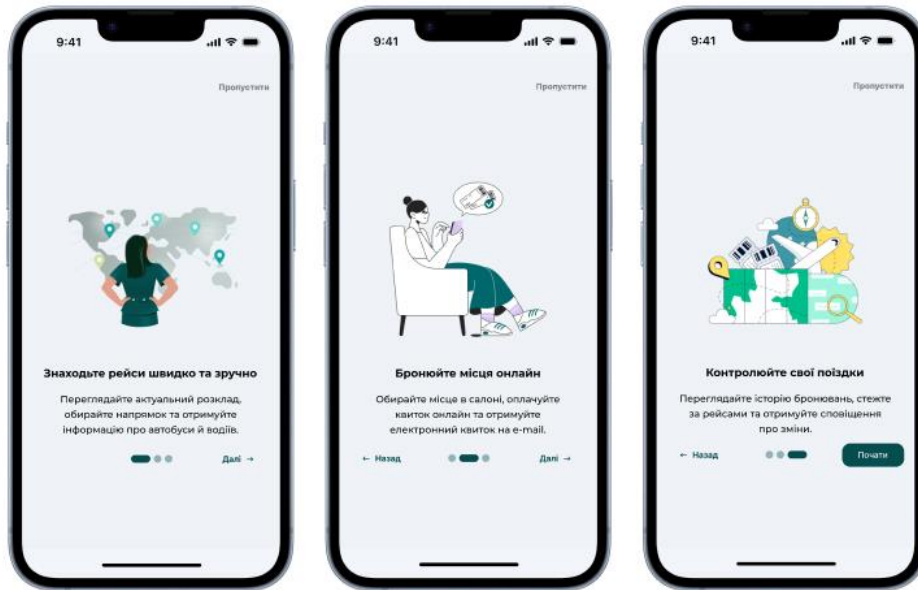


Рисунок А.2 – Онбордінг

Для доступу до всіх функцій користувачу необхідно створити акаунт, ввівши: ім'я та прізвище, електронну пошту, номер телефону та пароль, як показано на рисунку А.3.

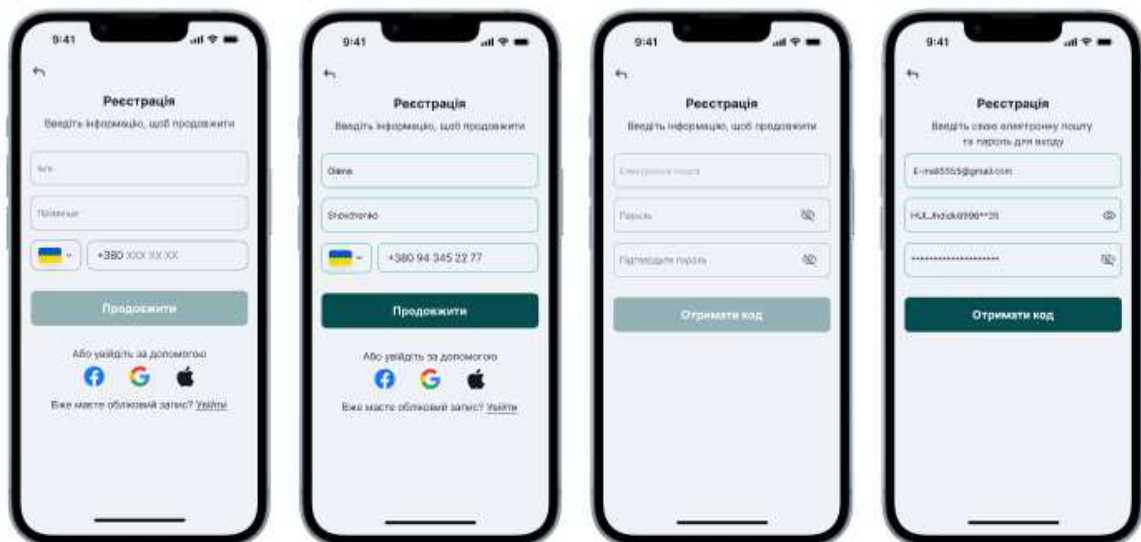


Рисунок А.3 – Екрани реєстрації

Після правильно введених даних на ваш номер телефону прийде код, який потрібно ввести, щоб підтвердити вашу реєстрацію, це все показано на рисунку А.4

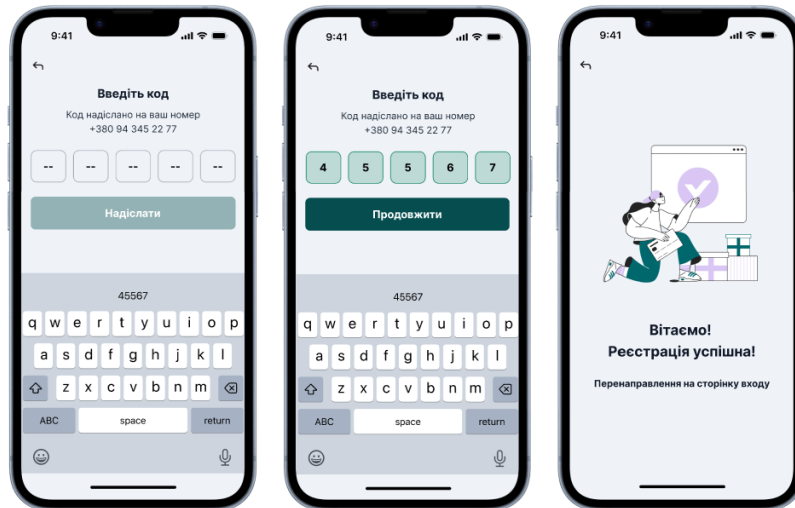


Рисунок А.4 – Успішна реєстрація

Якщо у вас вже є аккаунт, то є можливість увійти за допомогою електронної пошти чи те номеру телефону та пароля. Також підтримується кнопка «Забув пароль», що зображено на рисунку А.5.

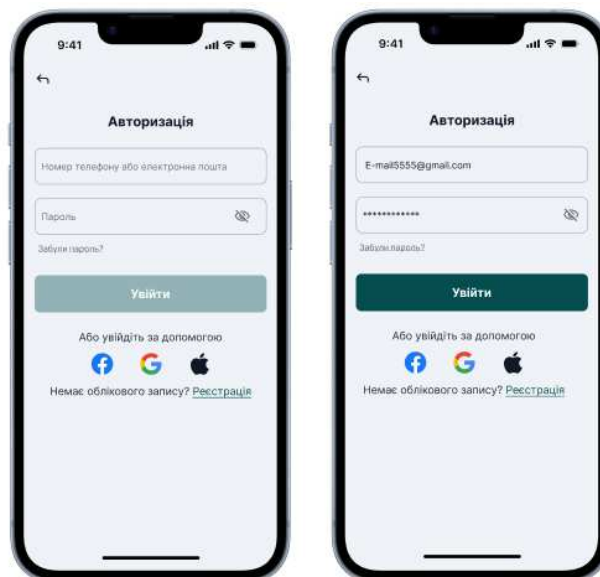


Рисунок А.5 – Авторизація

Після входу до аккаунту, ви потрапляєте на головну сторінку, де можете обрати напрямок (звідки – куди) та дату поїздки, а також переглянути знижки, що представлено на рисунку А.6.

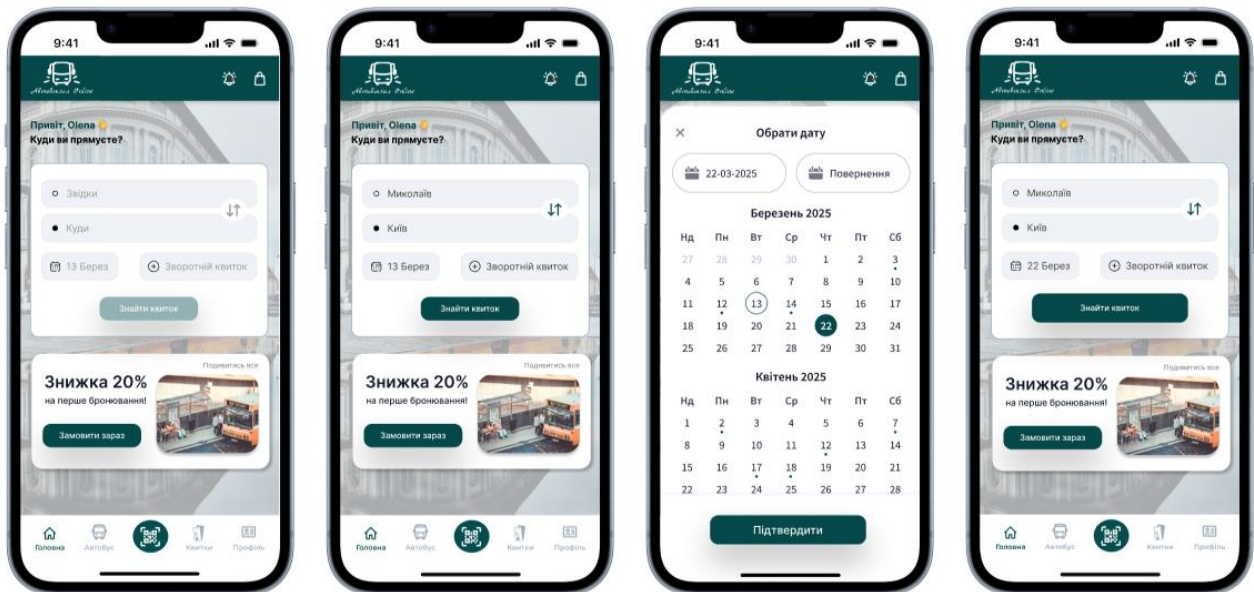


Рисунок А.6 – Головний екран додатку

Після введення дати та напрямку відкривається список доступних квитків, при натисканні подивитись деталі на обраний рейс показується: місце відбуття та прибуття, години, ціна, доступні місця, відстань, знижки та кнопка «Обрати місце». Користувач обирає місце в автобусі і вже після вибору необхідно натиснути «Підтвердити», що зображено на рисунку А.7.

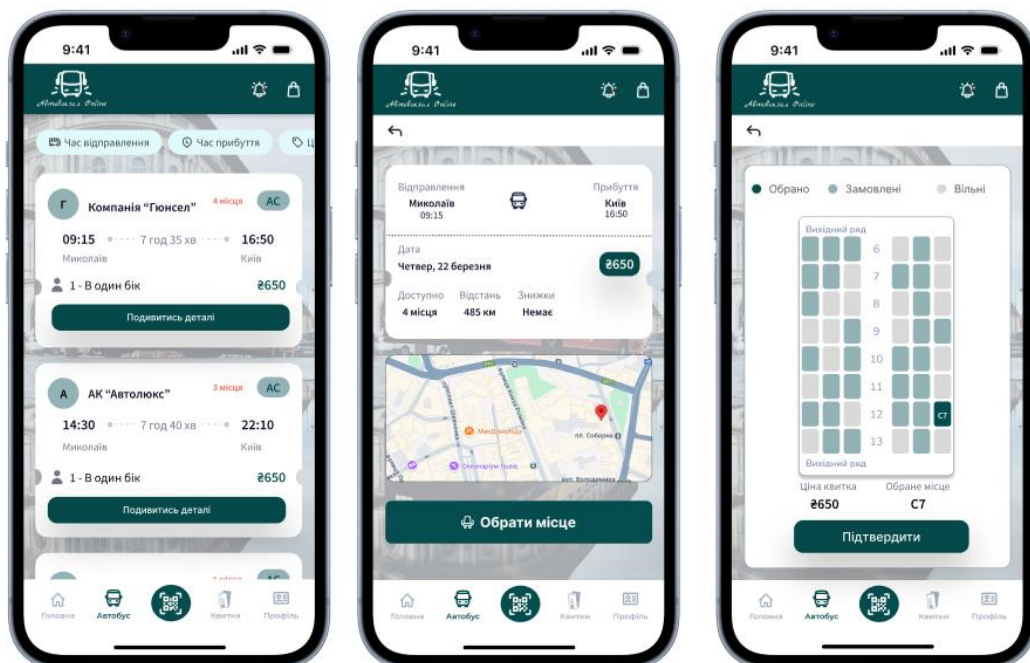


Рисунок А.7 – Перегляд та обирання місця в автобусі

Після вибору місця користувачу потрібно ввести особисті дані, які необхідні для оформлення квитка, а саме: ім'я та прізвище, як показано на рисунку А.8.

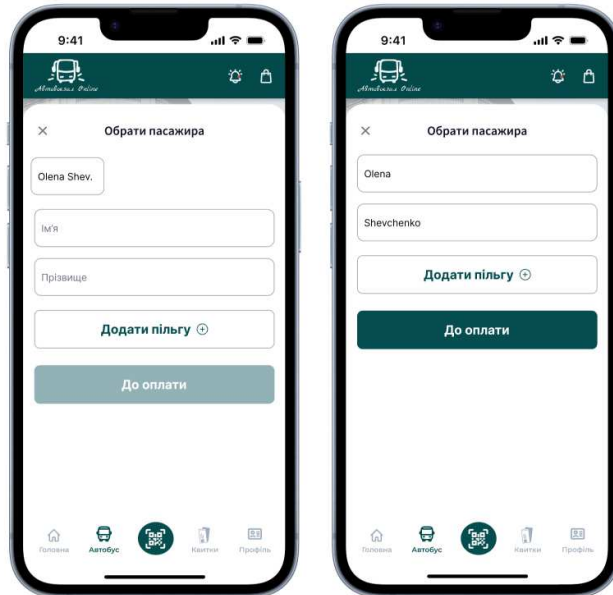


Рисунок А.8 – Дані пасажирів

Після підтвердження даних користувач переходить до оплати. Можна обрати один із доступних способів, які зображені на рисунку А.9.

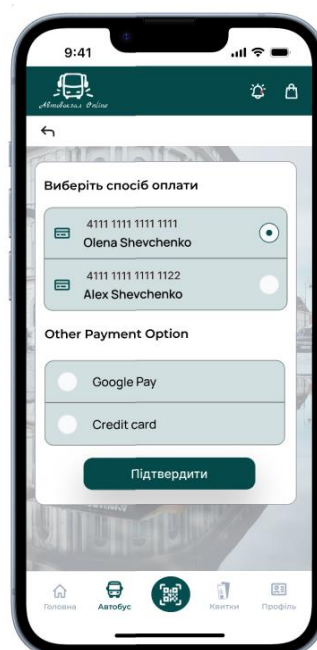


Рисунок А.9 – Оплата квитка

Після успішного здійснення оплати квиток автоматично зберігається у вкладці «Квитки», що знаходиться в нижньому меню додатку.

Також там користувач може переглянути активний (поточний) квиток та відкрити архів, де містяться всі раніше придбані та використані квитки, це все представлено на рисунку А.10.

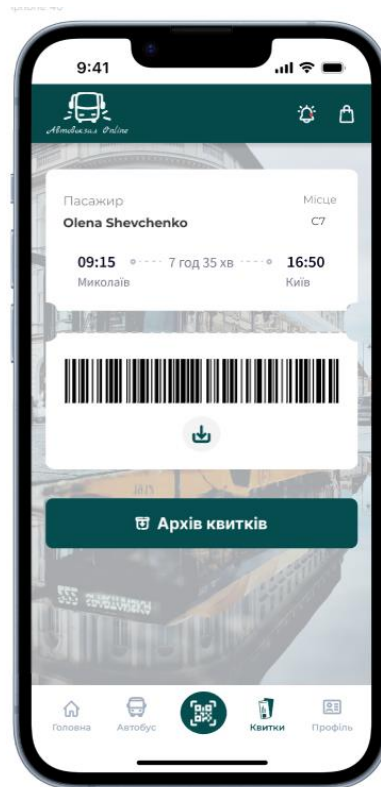


Рисунок А.10 – Сторінка «Квитки»

У розділі профілю користувач може редагувати особисті дані, переглядати історію поїздок, налаштовувати сповіщення, переглядати досягнення. У системі передбачена гейміфікація, користувач отримує нагороди та бейджі за активність, це ви можете побачити на рисунку А.11.

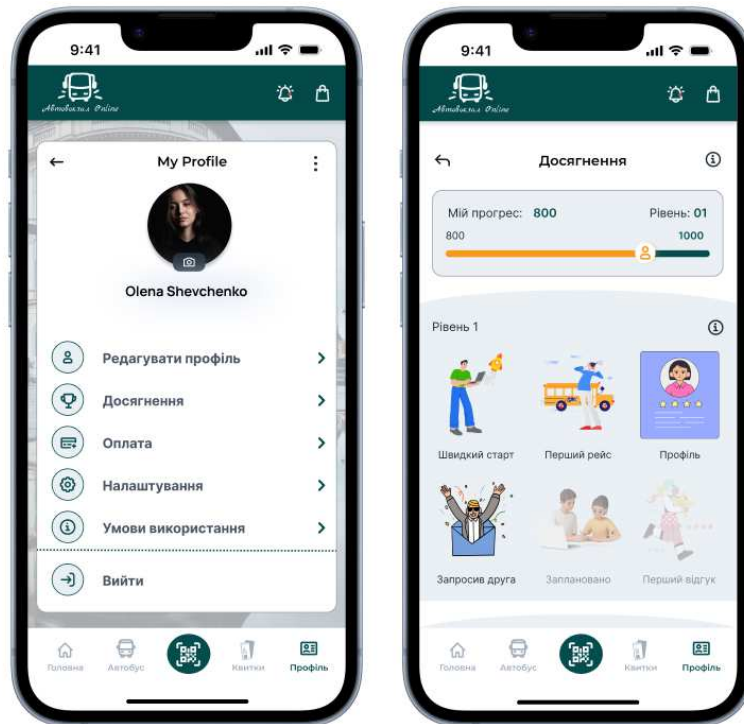


Рисунок А.11 – Профіль та досягнення користувача

Додаток Б

Технічне завдання на розробку

Веб-додаток автовокзалу призначений для забезпечення зручного та швидкого доступу до послуг бронювання, купівлі квитків і перегляду інформації про рейси.

Система спрямована на автоматизацію роботи автовокзалу, оптимізацію процесів обслуговування пасажирів та підвищення ефективності управління перевезеннями.

Метою розробки є створення інформаційної підсистеми, яка дозволить:

- скоротити черги в касах;
- надати користувачам можливість бронювати квитки онлайн;
- забезпечити актуальність даних про рейси;
- полегшити роботу персоналу;
- підвищити прозорість та керованість процесів.

Веб-додаток повинен мати структуровану архітектуру та складатися з взаємопов'язаних сторінок із чітким функціональним призначенням.

Основні сторінки системи:

- онбордінг (3 екрани);
- реєстрація;
- авторизація;
- головна сторінка;
- пошук рейсів (квитки);
- вибір місця в автобусі;
- оплата квитка;
- мої квитки;
- профіль користувача;
- досягнення.

Проект розробляється на основі дизайнерського макету (Figma), де передбачені інтуїтивна навігація та адаптивний користувацький інтерфейс.

Дизайн відповідає стандартам HTML5, CSS3 та використовує фреймворк Bootstrap. Деякі елементи реалізуються через JavaScript (jQuery). Серверна частина працює на PHP (Laravel 11).

Функціональні вимоги:

1. Користувач (пасажир) може:

- здійснювати пошук рейсів за датою та напрямком;
- переглядати квитки та деталі рейсу;
- бронювати та купувати квитки онлайн;
- обирати місце в автобусі;
- отримувати електронний квиток (QR-код);
- переглядати історію поїздок;
- редагувати профіль і налаштування;
- отримувати сповіщення (затримка, зміна часу, нагадування);
- отримувати досягнення за активність (бейджі).

2. Персонал автовокзалу (адміністратор / оператор / касир) може:

- створювати та редагувати рейси;
- призначати автобуси та водіїв;
- переглядати статистику продажів;
- керувати ролями користувачів;
- формувати звіти та аналітику.

Додаток В

Екранні форми

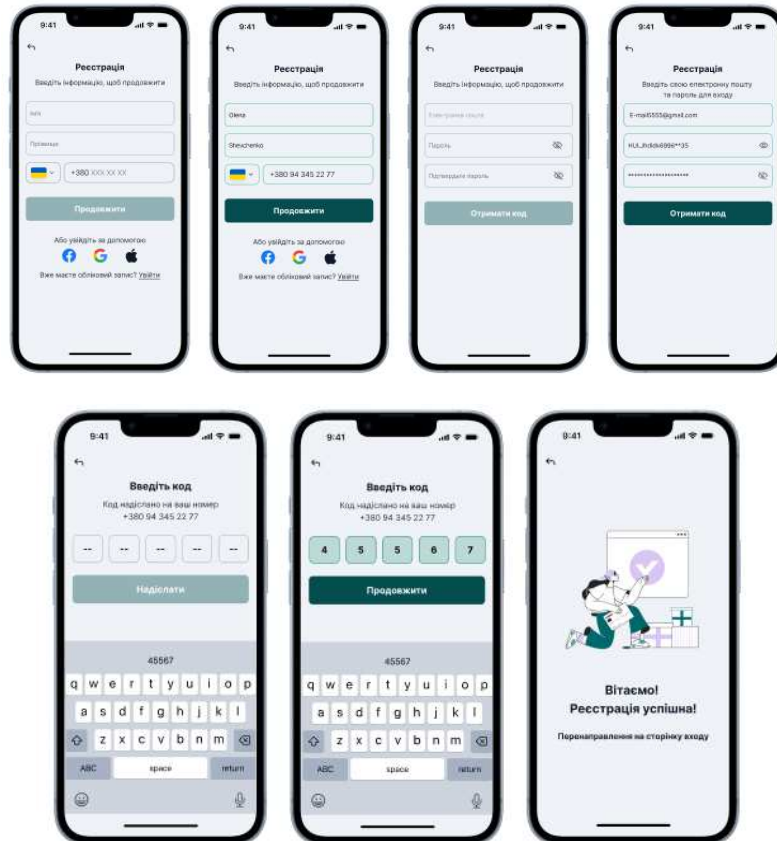


Рисунок В.1 – Форма реєстрації

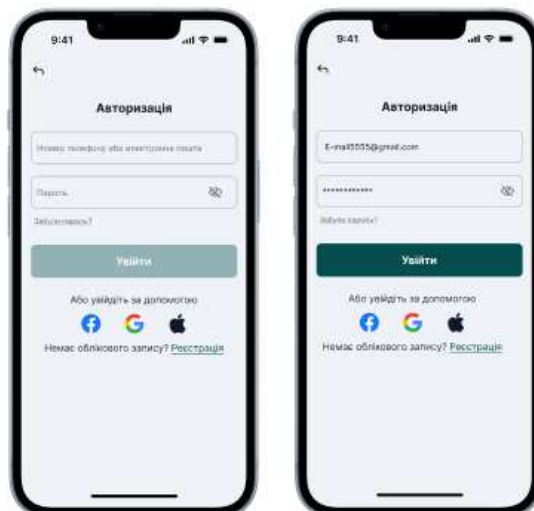


Рисунок В.2 – Форма авторизації

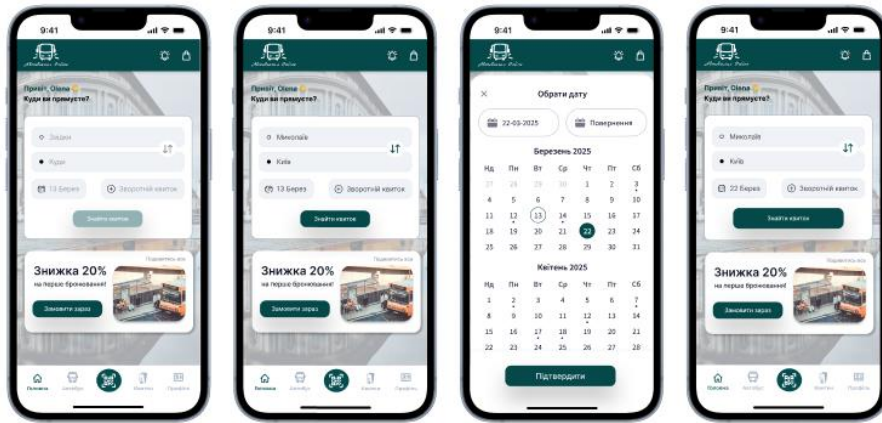


Рисунок В.3 – Пошук автобуса

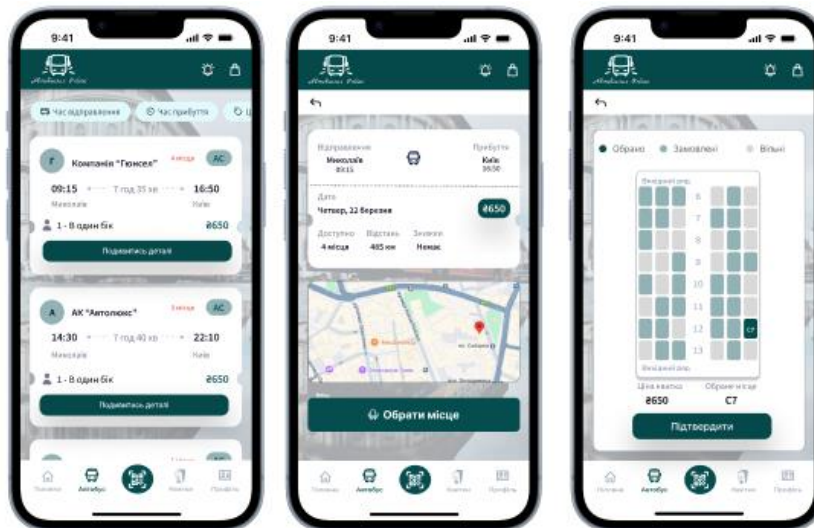


Рисунок В.4 – Вибір автобуса та місця в ньому

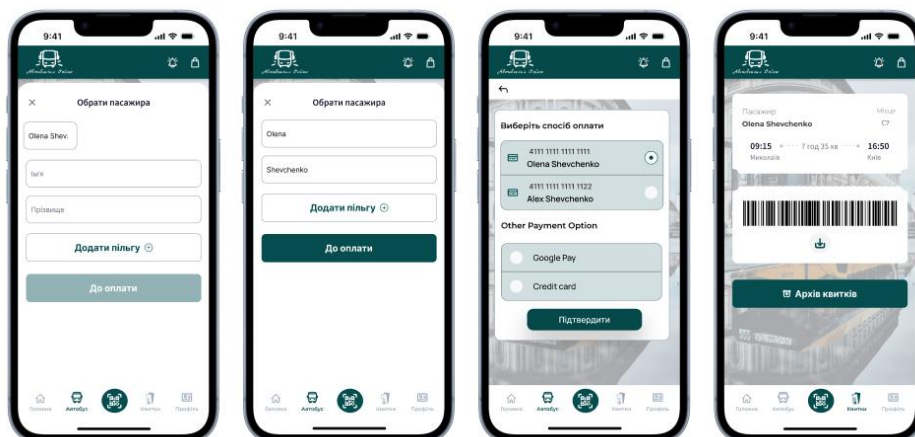


Рисунок В.5 – Оформлення квитка

Додаток Г

Фрагмент програмного коду

```
// Реєстрація користувача
// UserController.php
public function register(Request $request)
{
    $request->validate([
        'name' => 'required|string|max:255',
        'email' => 'required|email|unique:users',
        'password' => 'required|min:6|confirmed',
    ]);

    $user = User::create([
        'name' => $request->name,
        'email' => $request->email,
        'role' => 'passenger',    // роль за замовчуванням
        'password' => Hash::make($request->password),
    ]);

    return redirect()->route('login')->with('success',
'Reєстрація успішна!');
}

// Авторизація користувача (касира / адміністратора)
// AuthController.php

public function login(Request $request)
{
    $credentials = $request->only('email', 'password');

    if (Auth::attempt($credentials)) {
        if (Auth::user()->role === 'admin') {
            return redirect('/dashboard/admin');
        }
    }
}
```

```

    }
    if (Auth::user()->role === 'operator') {
        return redirect('/dashboard/operator');
    }
    return redirect('/dashboard/passenger');
}

return back()->withErrors(['email' => 'Невірний логін або
пароль']);
}

// Бронювання місця пасажиром
// BookingController.php

public function bookTicket(Request $request)
{
    $request->validate([
        'route_id' => 'required',
        'seat_number' => 'required',
        'payment_method' => 'required',
    ]);

    Booking::create([
        'user_id' => Auth::id(),
        'route_id' => $request->route_id,
        'seat_number' => $request->seat_number,
        'status' => 'підтверджено',
    ]);

    return redirect()->back()->with('success', 'Квиток
заброньовано!');
}

```