МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ЕКОНОМІКО-ПРАВНИЧИЙ ФАХОВИЙ КОЛЕДЖ ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ»

Циклова комісія математичних дисциплін та інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «ПРОЄКТУВАННЯ ТА РОЗРОБКА ВЕБДОДАТКУ З БРОНЮВАННЯ НОМЕРІВ У ГОТЕЛІ»

Виконав: здобувач освіти 4 курсу, групи К121-22
Спеціальність 121 Інженерія програмного забезпечення
(шифр і спеціальність)
Даніїл ТОМЧЕНКО
(ім'я та ПРІЗВИЩЕ)
Керівник Олена ШИТІКОВА
(ім'я та ПРІЗВИЩЕ)
Рецензент викладач Запорізького авіаційного фахового коледжу ім. О. Г. Івченка, Артур ТІМЕНКО
(посада, вчене звання, науковий ступінь, ім'я та ПРІЗВИЩЕ)

Запоріжжя
2026

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ВСП «Економіко-правничий фаховий коледж ЗНУ»

Освітньо–професійний ступінь фаховий молодший бакалавр

Спеціальність 121 – Інженерія програмного забезпечення
(шифр і назва)

ЗАТВЕРДЖУЮ

Голова циклової комісії математичних дисциплін та інформаційних технологій

_____ Т.М. Смолянкова
(підпис)

“ 05 ” червня 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

ТОМЧЕНКО Даніілу Денисовичу
(ПРІЗВИЩЕ, ім’я та по– батькові)

1. Тема роботи «Проєктування та розробка вебдодатку з бронювання номерів у готелі»

Керівник роботи к.т.н. Шитікова О.В.
(науковий ступінь, вчене звання, прізвище, ім’я, по батькові)

Затверджені наказом ВСП ЕПФК ЗНУ від «01» грудня 2025 р. № 1728-с

2. Строк подання студентом роботи 05.06.2026

3. Вихідні дані до роботи 1. Постановка задачі.
2. Перелік джерел.

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

1.Сучасне представлення веброзробки
2. Розробка проєкту вебдодатку з бронювання номерів у готелі
3. Програмна реалізація вебзастосунку

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

презентація до захисту

6. Консультанти розділів роботи

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

7. Дата видачі завдання 01.12.2025

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Розробка плану роботи:		
	постановка задачі	Грудень 2025	виконано
2.	Збір вихідних даних, обробка методичних		
	та теоретичних джерел	Січень 2026	виконано
3.	Розробка першого розділу:		
	огляд предметної області	Лютий 2026	виконано
4.	Розробка другого розділу:		
	розробка проєкту вебсайту та вибір технологій	Квітень 2026	виконано
5.	Розробка третього розділу:		
	розробка застосунку	Травень 2026	виконано
6.	Оформлення і нормоконтроль кваліфікаційної роботи та перевірка на плагіат	Червень 2026	виконано
7.	Захист кваліфікаційної роботи	16.06.2026	виконано

Здобувач освіти _____ Даніїл ТОМЧЕНКО
(підпис) (ім'я ПРІЗВИЩЕ)

Керівник роботи _____ Олена ШИТІКОВА
(підпис) (ім'я ПРІЗВИЩЕ)

Нормоконтроль пройдено

Нормоконтролер _____ Юлія БОРИСОВСЬКА
(підпис) (ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Кваліфікаційна робота: 52 сторінки, 1 таблиця, 24 ілюстрації, 15 позицій у переліку посилань.

Об’єкт дослідження – процес автоматизації надання послуг у сфері готельного бізнесу засобами вебтехнологій.

Предмет дослідження – методи, засоби та технології проєктування і програмної реалізації вебдодатку для бронювання номерів у готелі.

Метою кваліфікаційної роботи є проєктування та розробка ефективного вебдодатка для готелю, що забезпечить автоматизацію процесу резервування номерів та підвищення якості обслуговування клієнтів.

У кваліфікаційній роботі досліджено архітектурні особливості та специфіку роботи сучасних вебресурсів, а також проведено порівняльний аналіз діючих онлайн-платформ бронювання номерів. Використавши ці дані як підґрунтя, було розроблено програмний проєкт вебзастосунку для автоматизації взаємодії з клієнтами. Структуру зберігання даних спроєктовано у СУБД MySQL. Інтерфейсна та серверна частини реалізовані із застосуванням мови HTML, таблиць стилів CSS та скриптів PHP. Створений застосунок забезпечує двосторонню трансляцію даних до бази, дозволяючи ідентифікованим користувачам повноцінно працювати з інформаційною системою.

ГОТЕЛЬ, БРОНЮВАННЯ, ВЕБЗАСТОСУНОК, MYSQL, PHP, HTML, CSS

SUMMARY

Qualification work: 52 pages, 1 table, 24 illustrations, 15 positions in the list of references.

The object of the research is the process of automating the provision of services in the hotel business using web technologies.

The subject of the research is methods, tools and technologies for designing and software implementation of a web application for booking hotel rooms.

The purpose of the qualification work is to design and develop an effective web application for a hotel that will automate the process of booking rooms and improve the quality of customer service.

The qualification work investigated the architectural features and specifics of the work of modern web resources, as well as conducted a comparative analysis of existing online room booking platforms. Using this data as a basis, a software project of a web application was developed to automate interaction with customers. The data storage structure was designed in the MySQL database. The interface and server parts were implemented using HTML, CSS style sheets and PHP scripts. The created application provides two-way data transmission to the database, allowing identified users to fully work with the information system.

HOTEL, RESERVATION, WEB APPLICATION, MYSQL, PHP, HTML, CSS

ЗМІСТ

Завдання на кваліфікаційну роботу студенту 2
Реферат 4
Summary 5
Вступ 7
1 Сучасне предствлення веброзробки 9
1.1 Основні поняття веброзробки 9
1.2 Типи та призначення вебсайтів 10
1.3 Аналіз обраної предметної області 12
1.4 Огляд існуючих систем онлайн-бронювання готелів 14
1.5 Засоби та технології створення вебзастосунків 18
1.6 Висновки за розділом 1 20
2 Розробка проєкту вебдодатку для бронювання номерів у готелі 22
2.1 Технічне завдання 22
2.2 Проєктування структури вебсайту 24
2.3 UML-діаграма прецедентів 29
2.4 Схема бази даних 33
2.5 Проєктування інтерфейсу 34
3 Програмна реалізація вебзастосунку 38
3.1 Створення серверної частини 38
3.2 Створення клієнтської частини 42
Висновки 49
Перелік використаних джерел 51

ВСТУП

У сучасних умовах стрімкого розвитку інформаційних технологій цифровізація бізнес-процесів є невіддільною складовою успішного функціонування підприємств готельної індустрії. Наявність зручного та надійного вебдодатку для онлайн-бронювання дозволяє закладам розміщення оптимізувати процеси взаємодії з клієнтами, автоматизувати облік номерного фонду та значно підвищити рівень конкурентоспроможності на ринку послуг. Розробка власного програмного рішення забезпечує гнучкість у налаштуванні функціоналу під конкретні потреби готелю та сприяє покращенню користувацького досвіду, що робить тему даної роботи своєчасною та актуальною.

Об’єкт дослідження – процес автоматизації надання послуг у сфері готельного бізнесу засобами вебтехнологій.

Предмет дослідження – методи, засоби та технології проєктування і програмної реалізації вебдодатку для бронювання номерів у готелі.

Метою кваліфікаційної роботи є проєктування та розробка ефективного вебдодатку для готелю, що забезпечить автоматизацію процесу резервування номерів та підвищення якості обслуговування клієнтів.

Для досягнення поставленої мети необхідно вирішити такі завдання:

1. Здійснити аналіз предметної області та дослідити існуючі програмні рішення (вебсайти та системи) для бронювання готельних номерів;

2. Сформулювати вимоги до системи та розробити технічне завдання на створення вебдодатку;

3. Здійснити проєктування архітектури та логічної структури вебдодатку, а також розробити інтуїтивно зрозумілий інтерфейс користувача (UI/UX);

4. Спроєктувати структуру бази даних для збереження інформації про клієнтів, номерний фонд та статус бронювань;

5. Здійснити програмну реалізацію вебдодатку та провести тестування його компонентів.

Під час виконання роботи застосовувалися методи системного та порівняльного аналізу для дослідження аналогів. Програмна реалізація вебдодатку здійснювалася з використанням сучасних технологій веброзробки у середовищі Visual Studio Code.

Структура та обсяг роботи. Кваліфікаційна робота складається зі вступу, трьох розділів, висновків та переліку використаних джерел. Перший розділ містить теоретичні відомості щодо особливостей розробки вебдодатків, проведення аналізу предметної області, огляд існуючих сучасних систем бронювання готелів і вибір засобів та технологій веброзробки. Другий розділ присвячено етапу проєктування системи: наведено технічне завдання, функціональні вимоги, діаграми варіантів використання, схему бази даних, а також розроблено макети (wireframes) користувацького інтерфейсу. Третій розділ містить опис практичної реалізації кожного з компонентів вебдодатку, зокрема клієнтської та серверної частин, налаштування бази даних, а також результати тестування розробленого програмного продукту.

1 СУЧАСНЕ ПРЕДСТВЛЕННЯ ВЕБРОЗРОБКИ

1.1 Основні поняття веброзробки

Стрімкий розвиток інформаційно-комунікаційних технологій та глобальної мережі Інтернет зумовив перехід більшості бізнес-процесів у цифрове середовище [1]. Фундаментом цього переходу є веброзробка – галузь програмної інженерії, що спеціалізується на проєктуванні, створенні та підтримці вебсайтів і вебдодатків. Для забезпечення коректного розуміння принципів побудови системи бронювання готельних номерів необхідно визначити базові поняття та терміни цієї предметної області.

У сучасній науковій та технічній літературі розрізняють поняття вебсайту та вебдодатка:

1. Вебсайт – це сукупність пов'язаних між собою вебсторінок, що розміщені під одним доменним ім'ям і здебільшого носять інформаційний характер [2].

2. Вебдодаток – це клієнт-серверна програма, в якій клієнтом виступає веббраузер, а сервером – вебсервер. На відміну від класичного вебсайту, вебдодаток характеризується високим рівнем інтерактивності та орієнтований на виконання користувачем конкретних завдань [3] (наприклад, здійснення онлайн-бронювання, управління обліковим записом, проведення транзакцій) [4].

Більшість сучасних вебдодатків функціонують на базі архітектури «клієнт-сервер», яка передбачає поділ обчислювального навантаження між постачальником ресурсів (сервером) та споживачем (клієнтом). Процес розробки такого програмного забезпечення традиційно поділяється на дві основні складові:

1. Клієнтська частина (Frontend) – це публічна частина вебдодатка, з якою безпосередньо взаємодіє користувач. Вона відповідає за відображення

графічного інтерфейсу (UI), обробку дій користувача (UX) та відправлення запитів до сервера. Базовими технологіями фронтенд-розробки є мова гіпертекстової розмітки HTML, каскадні таблиці стилів CSS та мова програмування JavaScript [5].

2. Серверна частина – це прихована від користувача програмно-апаратна складова, що розміщується на сервері. Вона відповідає за бізнес-логіку вебдодатка, обробку запитів від клієнтської частини, автентифікацію та авторизацію користувачів, а також за взаємодію з базою даних [6].

Невіддільним компонентом будь-якого сучасного вебдодатка є база даних – структурований масив інформації, що зберігається на обчислювальному пристрої та керується системою управління базами даних (СУБД). У контексті готельного бізнесу база даних є критично важливою, оскільки саме в ній зберігається інформація про клієнтів, доступний номерний фонд, тарифи та актуальні статуси бронювань.

Обмін даними між клієнтською та серверною частинами здійснюється за допомогою мережевих протоколів, основним з яких є HTTP або його захищена версія HTTPS. Захищений протокол забезпечує шифрування даних під час їхньої передачі, що є обов'язковою вимогою для систем, які обробляють персональні дані клієнтів та платіжну інформацію.

Таким чином, розробка вебдодатка для бронювання номерів у готелі є комплексним процесом, що вимагає інтеграції клієнтських та серверних технологій, проєктування надійної архітектури баз даних та забезпечення високого рівня безпеки інформаційного обміну.

1.2 Типи та призначення вебсайтів

Сучасний інформаційний простір характеризується значним розмаїттям вебресурсів, які суттєво відрізняються за архітектурою, масштабом, функціональними можливостями та цільовим призначенням [7]. Для

обґрунтування вибору оптимального формату програмного рішення для автоматизації бізнес-процесів готелю доцільно розглянути загальноприйняту класифікацію вебсайтів.

Залежно від технології формування контенту та способу взаємодії з користувачем, вебресурси поділяються на дві базові категорії:

1. Статичні вебсайти – ресурси, що складаються з незмінного набору HTML-сторінок. Інформація на таких сайтах залишається статичною до моменту її ручного редагування розробником. Вони не використовують бази даних і призначені переважно для одностороннього інформування користувача.

2. Динамічні вебсайти та вебдодатки – системи, здатні генерувати контент «на льоту» у відповідь на запити користувача. Вони активно використовують серверні мови програмування та взаємодіють із системами управління базами даних (СУБД). Саме до цієї категорії належать сучасні системи онлайн-бронювання [8].

За цільовим призначенням та специфікою бізнес-логіки виокремлюють такі основні типи вебсистем:

1. Сайти-візитки та лендінги. Невеликі вебресурси, що містять загальну інформацію про компанію, її послуги та контактні дані. Їхня головна мета – первинне представлення бізнесу в мережі Інтернет та швидке залучення цільової аудиторії.

2. Корпоративні вебсайти. Повноцінні інформаційні представництва компаній, що можуть містити розгорнуті каталоги послуг, новини, закриті розділи для співробітників (B2B-портали) та модулі зворотного зв'язку.

3. Електронна комерція та системи онлайн-бронювання. Багатофункціональні вебдодатки, призначені для здійснення комерційних транзакцій у мережі. До цієї групи належать інтернет-магазини, системи продажу квитків та сервіси резервування готельних номерів. Вони вимагають складної серверної інфраструктури, інтеграції з платіжними шлюзами,

забезпечення високого рівня безпеки даних та синхронізації з внутрішніми системами обліку (наприклад, статусом зайнятості номерного фонду) [9].

4. Інформаційні портали та агрегатори. Великі вебресурси, що об'єднують структуровану інформацію з багатьох джерел. У контексті готельного бізнесу прикладами є глобальні системи дистрибуції (GDS) та агрегатори (наприклад, Booking.com або Agoda), які збирають пропозиції від тисяч різних готелів.

З огляду на поставлену мету кваліфікаційної роботи – проєктування вебдодатку з бронювання номерів у готелі – розроблюваний програмний продукт класифікується як динамічний вебдодаток із функціоналом електронної комерції. Такий тип системи вимагає реалізації інтерактивного користувацького інтерфейсу, надійної серверної логіки для обробки запитів на резервування, а також проєктування реляційної або нереляційної бази даних для зберігання історії транзакцій та актуального стану номерного фонду готелю.

1.3 Аналіз обраної предметної області

Предметною областю даного дослідження є сфера надання готельних послуг, зокрема бізнес-процес онлайн-бронювання номерного фонду. Індустрія гостинності є однією з найбільш динамічних галузей економіки, що вимагає високого рівня клієнтоорієнтованості та оперативності в обслуговуванні. У зв'язку з цим автоматизація процесів взаємодії з клієнтами стає критичним фактором конкурентоспроможності будь-якого закладу розміщення [10].

Традиційний підхід до резервування номерів, що передбачає телефонні дзвінки або електронні листи, поступово втрачає свою ефективність через залежність від людського фактора, обмеженість робочого часу персоналу та незручність для сучасного споживача. На зміну йому прийшли системи

онлайн-бронювання, які дозволяють цифровізувати повний цикл надання послуги [11].

Детальний аналіз предметної області вимагає ідентифікації ключових учасників (акторів) системи та основних інформаційних об'єктів. Взаємодія у межах розроблюваного вебдодатку здебільшого здійснюється між двома основними категоріями користувачів:

1. Клієнт (Гість) – зовнішній користувач системи, головною метою якого є пошук вільного номера за заданими критеріями (дата заїзду/виїзду, кількість осіб, категорія номера), ознайомлення з умовами проживання, здійснення бронювання та отримання підтвердження.

2. Адміністратор (Менеджер готелю) – внутрішній користувач, який здійснює управління номерним фондом, обробку вхідних заявок, контроль статусів бронювань, а також управління ціноутворенням та інформаційним наповненням вебдодатку.

Основними інформаційними сутностями предметної області, які підлягають подальшому моделюванню в базі даних, є:

1. Номерний фонд: сукупність номерів готелю з їхніми унікальними характеристиками (категорія, місткість, вартість, перелік зручностей, поточний статус доступності).

2. Бронювання (Транзакція): запис, що фіксує факт резервування певного номера конкретним клієнтом на визначений період часу, з відповідним статусом (очікує підтвердження, підтверджено, скасовано тощо).

3. Профіль клієнта: сукупність персональних та контактних даних гостя, необхідних для ідентифікації та зворотного зв'язку.

Життєвий цикл процесу онлайн-бронювання концептуально складається з таких етапів: пошук доступних пропозицій у заданому діапазоні дат → вибір оптимального варіанту → введення персональних даних клієнта → підтвердження резервування → автоматична фіксація змін у базі даних готелю та відправка квитанції (ваучера) клієнту.

Отже, впровадження спеціалізованого вебдодатку дозволяє розв'язати низку актуальних проблем предметної області: забезпечення цілодобового доступу до послуг готелю (режим 24/7), усунення ризику овербукінгу (подвійного бронювання одного номера на ті самі дати), оптимізація робочого навантаження персоналу та підвищення загального рівня сервісу.

1.4 Огляд існуючих систем онлайн-бронювання готелів

Для визначення оптимальної архітектури, базових функціональних вимог та сучасних підходів до проєктування інтерфейсу користувача (UI/UX) розроблюваного вебдодатка, було проведено аналіз наявних на ринку програмних рішень. Дослідження охоплює як глобальні системи-агрегатори, так і корпоративні вебсайти окремих готельних мереж, що дозволяє виявити їхні сильні та слабкі сторони.

1. Глобальна система бронювання Booking.com Booking.com є одним із найпопулярніших у світі агрегаторів готельних послуг. Головною перевагою системи є потужна пошукова система з розгалуженим деревом фільтрів (за ціною, зручностями, відстанню до центру тощо) та складна система рейтингування на основі відгуків користувачів. Проте інтерфейс платформи є значно перевантаженим інформаційними блоками, маркетинговими тригерами («залишився 1 номер», «цей готель переглядають 5 осіб») та рекламою, що може розпорошувати увагу користувача і ускладнювати процес прямого бронювання для конкретного закладу [12].

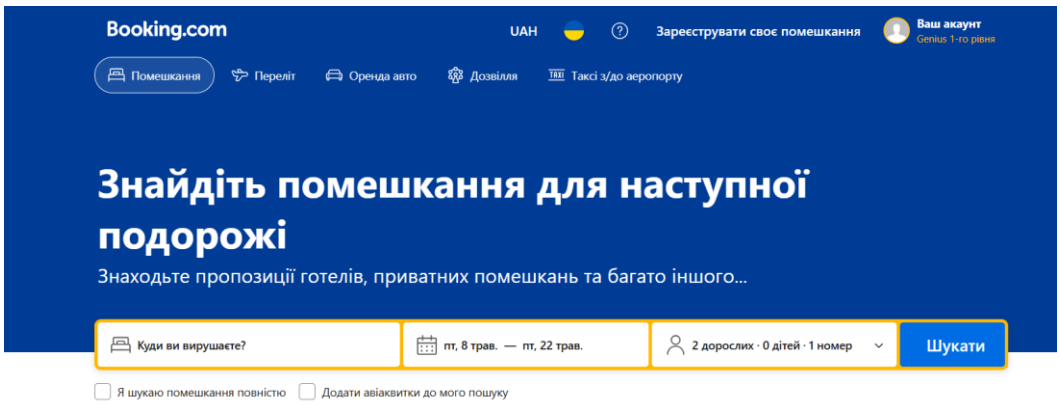


Рисунок 1.1 – Головна сторінка системи бронювання Booking.com

2. Вебсайт міжнародної мережі готелів Hilton (hilton.com) Корпоративний вебсайт мережі Hilton є прикладом високоякісного монобрендового вебдодатка. Система характеризується строгим корпоративним дизайном, інтуїтивно зрозумілою навігацією та фокусом на програмі лояльності (Hilton Honors). Процес бронювання розбитий на чіткі логічні кроки: вибір готелю, вибір категорії номера, додавання додаткових послуг (сніданок, трансфер) та оплата. Важливою деталлю є використання інтерактивного календаря з відображенням мінімальної вартості за кожну добу [13].

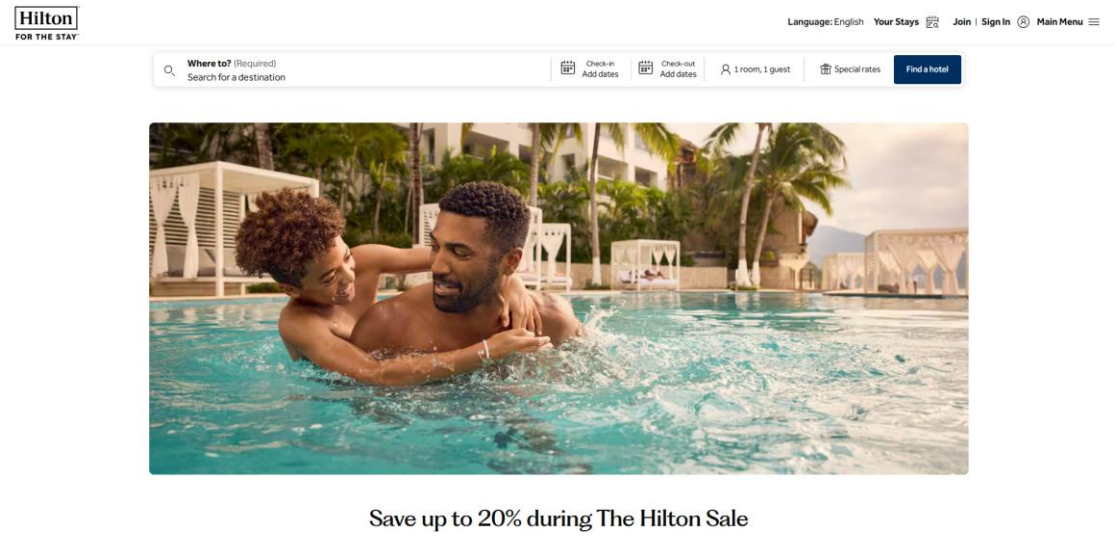


Рисунок 1.2 – Інтерфейс вибору номерів на вебсайті мережі Hilton

3. Вебсайт національної мережі Premier Hotels and Resorts (phnr.com) Даний вебресурс репрезентує український ринок готельних послуг. Вебдодаток мережі Premier Hotels відрізняється лаконічним дизайном та наявністю фіксованого (sticky) блоку пошуку номерів, який завжди залишається в полі зору користувача під час скролінгу сторінки. До переваг можна віднести якісну візуалізацію номерного фонду (великі фотографії, детальний опис зручностей) та мультимовність. Недоліком є іноді тривалий час очікування відповіді від сервера під час формування списку доступних номерів на обрані дати [14].

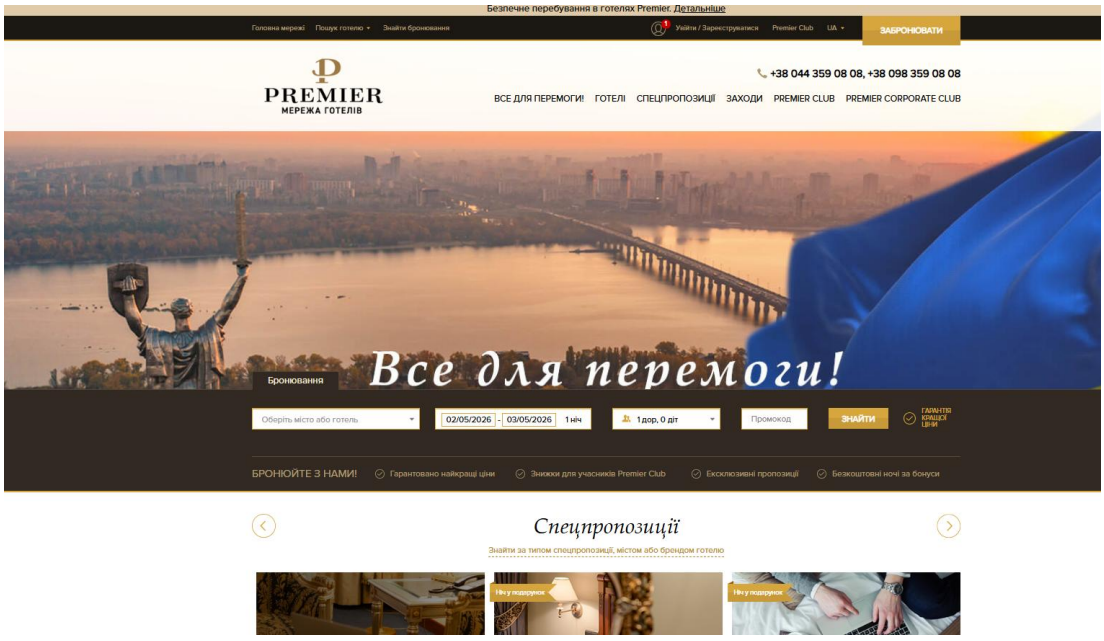


Рисунок 1.3 – Блок пошуку та бронювання на вебсайті Premier Hotels and Resorts

4. Вебсайт української готельної компанії Ribas Hotels Group (ribashotelsgroup.ua) Вебсайт компанії Ribas є прикладом сучасного підходу до веброзробки з використанням актуальних UI/UX трендів. Інтерфейс системи вирізняється мінімалізмом, використанням великої типографіки та високою швидкістю завантаження (оптимізований frontend). Форма бронювання реалізована у вигляді зручного спливаючого вікна (модального вікна) або

окремого віджета, що дозволяє користувачеві здійснити резервування, не переходячи на інші сторінки та не втрачаючи контексту обраного готелю [15].

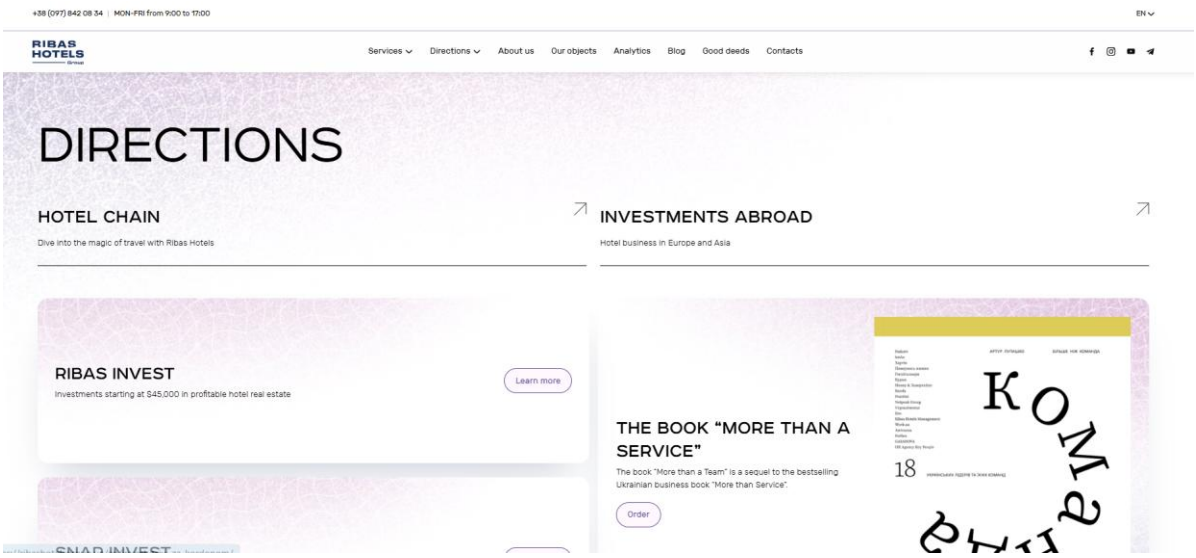


Рисунок 1.4 – Модуль онлайн-бронювання на вебсайті Ribas Hotels Group

Розглянуті аналоги сайтів бронювання номерів мають як свої переваги, так і недоліки. Для подальшої роботи було виділено наступний функціонал, що буде реалізовано в розроблюваному вебдодатку:

- інтуїтивно зрозуміла система пошуку вільних номерів з використанням інтерактивного календаря для вибору дат заїзду та виїзду;
- мінімалістичний інтерфейс користувача без перевантаження зайвими інформаційними блоками та маркетинговими тригерами;
- фіксований (sticky) блок пошуку або зручне модальне вікно бронювання, що дозволяє оформлювати замовлення без втрати контексту поточної сторінки;
- покроковий та логічний процес оформлення резервування (вибір категорії, перегляд деталей, підтвердження);
- детальна візуалізація номерного фонду з якісними зображеннями та чітким описом наявних зручностей;

- оптимізована взаємодія з базою даних для забезпечення високої швидкості формування списку доступних номерів та миттєвого відгуку системи.

1.5 Засоби та технології створення вебзастосунків

Для успішної реалізації вебдодатку бронювання готельних номерів необхідно було обрати оптимальний стек технологій. Процес вибору базувався на таких критеріях: специфіка предметної області (готельний бізнес), вимоги до швидкодії та безпеки, вартість підтримки інфраструктури та забезпечення високої доступності для кінцевих користувачів.

Після аналізу існуючих рішень для розробки було обрано класичний стек вебтехнологій: HTML/CSS/JavaScript для клієнтської частини (Frontend), PHP для серверної частини (Backend) та MySQL для керування базами даних. Вибір кожної з технологій є обґрунтованим з огляду на наступні фактори.

1. Клієнтська частина: HTML5, CSS3.

Незважаючи на популярність SPA-фреймворків, для вебдодатку готелю оптимальним є використання класичної тріади (HTML/CSS/JS). По-перше, сайти готелів критично залежать від пошукової оптимізації (SEO) [5]. Класичний HTML-рендеринг забезпечує найкращу індексацію сторінок (описів номерів, послуг) пошуковими роботами. По-друге, функціонал клієнтської частини (валідація форм реєстрації, динамічний вибір дат заїзду у календарі, слайдери фотографій номерів) не є настільки складним, щоб виправдати завантаження об'ємного ядра фреймворку. Використання чистого JavaScript (Vanilla JS) дозволяє уникнути надмірної складності (overengineering), забезпечує миттєве завантаження сторінок та стабільну роботу навіть на мобільних пристроях зі слабким інтернет-з'єднанням.

2. Серверна частина: PHP

Альтернативні варіанти, що розглядались – Node.js, Python (Django), Ruby on Rails.

PHP залишається однією з найоптимальніших мов для розробки систем бронювання середнього масштабу. На відміну від Node.js, який орієнтований на асинхронні події реального часу (наприклад, чати), або Python, який частіше використовується для складних обчислень та машинного навчання, PHP був створений спеціально для веброзробки [3].

Проєкти на PHP підтримуються 99% сучасних хостинг-провайдерів (шаред-хостинги), що суттєво знижує фінансові витрати готелю на розгортання та щомісячну підтримку сервера порівняно з утриманням VPS для Node.js або Python.

PHP має глибоку та безшовну інтеграцію з СУБД MySQL. Наявність вбудованих механізмів обробки сесій (що критично важливо для системи авторизації та збереження стану «кошика» під час бронювання) дозволяє реалізувати безпечний бекенд з мінімальною кількістю сторонніх залежностей.

3. Система управління базами даних: MySQL

Альтернативи: PostgreSQL, NoSQL-рішення (наприклад, MongoDB).

Система бронювання номерів має чітко структуровані дані з жорсткими зв'язками (Користувач -> Замовлення -> Номер -> Платіж).

Використання документо-орієнтованих NoSQL баз (MongoDB) у цьому випадку є помилковим, оскільки вони погано справляються зі складними транзакціями, де важливо, щоб один і той самий номер не був заброньований двома клієнтами одночасно на ті ж дати [8].

Порівняно з PostgreSQL, яка є більш потужною, але й більш вимогливою до ресурсів, MySQL є швидшою для операцій читання (SELECT-запити для пошуку вільних номерів складатимуть понад 80% усіх запитів до БД). Реляційна модель MySQL гарантує цілісність даних (відповідність принципам ACID) та ідеально вписується в екосистему обраного бекенду (архітектура LAMP/WAMP).

Таким чином, обраний інструментарій не є випадковим, а являє собою збалансоване рішення, що гарантує надійність системи, високу швидкість розробки та простоту подальшого адміністрування вебдодатку персоналом готелю.

1.6 Висновки за розділом 1

У даному розділі було проведено комплексне теоретичне дослідження предметної області та проаналізовано передумови для розробки вебдодатку бронювання готельних номерів. За результатами виконаної роботи можна зробити наступні висновки:

1. Досліджено специфіку предметної області. Визначено, що автоматизація процесів взаємодії з клієнтами є критично важливою вимогою для сучасного готельного бізнесу. Впровадження системи онлайн-бронювання дозволяє підвищити якість обслуговування, забезпечити цілодобовий доступ користувачів до послуг та оптимізувати адміністративні процеси, мінімізуючи ризик помилок через людський фактор.

2. Проведено аналіз існуючих рішень та сайтів-аналогів. Детальний огляд наявних на ринку систем бронювання дозволив виділити їхні сильні та слабкі сторони. Завдяки цьому було сформовано концептуальне бачення майбутнього продукту, визначено базовий набір необхідного функціоналу (фільтрація за датами, перегляд опису номерів, зручна форма реєстрації) та виявлено інтерфейсні недоліки конкурентів, яких слід уникнути під час власної розробки.

3. Здійснено обґрунтований вибір засобів розробки. На основі аналізу вимог до майбутньої системи було порівняно сучасні технологічні інструменти. Доведено, що для реалізації поставленої задачі найбільш раціональним є використання класичного стеку вебтехнологій: HTML/CSS/JavaScript для створення інтуїтивно зрозумілої клієнтської

частини, мови PHP для побудови стабільної серверної логіки та СУБД MySQL для надійного збереження даних. Такий вибір гарантує баланс між швидкістю розробки, продуктивністю додатку та простотою його подальшого супроводу.

Отже, результати теоретичного дослідження, проведений аналіз аналогів та затверджений стек технологій повністю формують необхідну науково-практичну та інформаційну базу. Це дає змогу перейти до наступного етапу кваліфікаційної роботи – безпосереднього проєктування архітектури вебдодатку та його програмної реалізації.

2 РОЗРОБКА ПРОЄКТУ ВЕБДОДАТКУ ДЛЯ БРОНЮВАННЯ НОМЕРІВ У ГОТЕЛІ

2.1 Технічне завдання

2.1.1 Найменування та область застосування

Розроблюваний програмний продукт отримав назву «Вебдодаток з бронювання номерів у готелі».

Основне призначення вебдодатку полягає в автоматизації та спрощенні комунікації між клієнтами та співробітниками готелю, забезпечуючи швидкий, прозорий та зручний доступ до послуг бронювання.

2.1.2 Підстава для розробки

Програма розробляється на підставі наказу № 1728-с «Про затвердження тем кваліфікаційних робіт студентів 4 курсу освітньо-професійного ступеня «фаховий молодший бакалавр» денної форми здобуття освіти ВСП «Економіко-правничий фаховий коледж ЗНУ» від 01.12.2025 року.

2.1.3 Призначення розробки

Розроблений програмний продукт призначений для інформатизації та автоматизації процесів взаємодії між клієнтами та адміністрацією готельного комплексу. Упровадження даного вебзастосунку спрямоване на вирішення наступних експлуатаційних та функціональних завдань:

1) забезпечення надійних механізмів реєстрації, авторизації та автентифікації користувачів із метою захисту персональних даних та розмежування рівнів доступу;

2) динамічне відображення актуального номерного фонду готелю із деталізацією ключових характеристик апартаментів (категорія, місткість, опис, вартість за добу);

3) реалізація модуля онлайн-бронювання номерів, що включає алгоритми перевірки доступності кімнат на обрані дати та автоматичний розрахунок загальної вартості проживання;

4) систематизація, збереження та виведення історії транзакцій (бронювань) в індивідуальному кабінеті користувача;

5) забезпечення можливості оперативного оновлення статусів об'єктів (номерів) та управління інформаційним контентом через безперервну взаємодію з реляційною базою даних;

6) організація інтерактивного каналу комунікації шляхом імплементації модуля зворотного зв'язку для обробки звернень та запитів від клієнтів.

2.1.4 Технічні вимоги до програмного продукту

На етапі проєктування вебзастосунку було сформовано наступний комплекс технічних та функціональних вимог:

1) реалізація надійної клієнт-серверної архітектури для забезпечення безперебійного обміну даними між користувацьким інтерфейсом та сервером із подальшою їх обробкою;

2) розробка захищеної підсистеми реєстрації, авторизації та автентифікації користувачів із застосуванням методів криптографічного хешування паролів та управління сесіями;

3) розмежування прав доступу до функціоналу системи: забезпечення можливості вільного перегляду номерного фонду для неавторизованих

відвідувачів (режим «Гість») та надання доступу до створення бронювань і перегляду історії транзакцій виключно авторизованим користувачам;

4) забезпечення динамічного генерування контенту на вебсторінках (каталог номерів, ціни, описи) на основі актуальних даних, отриманих із реляційної бази даних;

5) гарантування логічної ізольованості даних: унеможливлення перегляду, модифікації або видалення облікових записів та персональних бронювань одних користувачів іншими;

6) проєктування ергономічного, інтуїтивно зрозумілого інтерфейсу (UI) з використанням каскадних таблиць стилів (CSS) для забезпечення комфортної навігації та коректного відображення елементів управління;

7) організація безпечного та структурованого зберігання облікових даних користувачів, інформації про об'єкти готелю та запитів зворотного зв'язку на сервері під управлінням СУБД MySQL;

8) захист системи від несанкціонованої зміни інформації рядовими користувачами та забезпечення цілісності даних шляхом імплементації дворівневої валідації (на стороні клієнта через JavaScript та на стороні сервера через PHP), а також використання параметризованих запитів (PDO) для запобігання SQL-ін'єкціям.

2.2 Проєктування структури вебсайту

Перед безпосередньою програмною реалізацією вебзастосунку було проведено етап проєктування його архітектури та навігаційної структури. Ефективність взаємодії користувача з системою (UX) та забезпечення інтуїтивно зрозумілого інтерфейсу значною мірою залежать від продуманої логіки переходів між сторінками. Для досягнення цієї мети було враховано наступні аспекти:

1. Структуру вебзастосунку логічно розподілено на загальнодоступні (публічні) та захищені (приватні) модулі. До публічного простору належать головна сторінка, каталог номерного фонду з детальною інформацією про об'єкти, сторінка контактної інформації та форми зворотного зв'язку, а також інтерфейси реєстрації та авторизації. До захищеного простору, доступного виключно після успішної автентифікації, належать модуль оформлення бронювання конкретного номера та панель перегляду історії особистих транзакцій (бронювань).

2. Розроблено єдине навігаційне меню, розміщене у верхньому колонтитулі кожної сторінки для забезпечення безперешкодного доступу до ключових розділів. Навігаційна панель є контекстно-залежною: її контент генерується динамічно на основі стану користувацької сесії. Для неавторизованих відвідувачів виводяться посилання для входу та створення облікового запису, тоді як для авторизованих користувачів ці елементи замінюються на посилання до панелі керування власними бронюваннями та функцію безпечного виходу із системи.

3. Маршрутизацію користувача спроєктовано за принципом мінімізації кроків до цільової дії. Наприклад, перехід до сторінки бронювання здійснюється безпосередньо з інформаційної картки обраного номера. Крім того, реалізовано алгоритми автоматичного перенаправлення (редиректу): у разі спроби неавторизованого доступу до захищеного функціоналу система перериває процес і направляє користувача на сторінку входу. На всіх сторінках передбачено зрозумілі кнопки повернення (наприклад, «На головну» або «Повернутися до номерів»), що унеможливлює потрапляння користувача в «навігаційний тупик».

На рисунку 2.1 представлено ієрархію сторінок вебзастосунку.

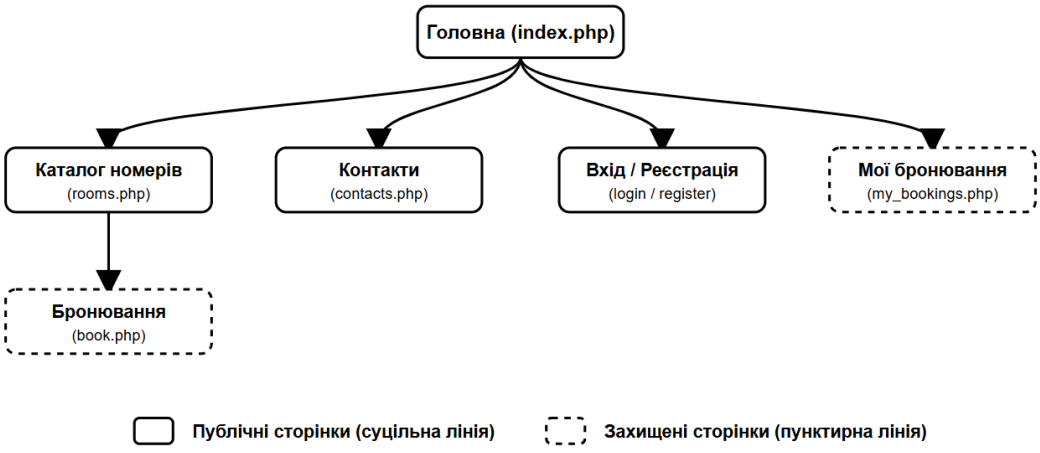


Рисунок 2.1 – Ієрархія сторінок вебсайту

Після того як ієрархія та логіка взаємодії модулів системи були остаточно визначені, виникає необхідність деталізації користувацького інтерфейсу, що передбачає проєктування структурних елементів та контентних блоків у межах кожної сторінки.

2.2.1 Структура сторінок сайту

Візуалізація архітектури інтерфейсу користувача та просторового розташування елементів на вебсторінках здійснювалася шляхом проєктування Wireframe-макетів у спеціалізованому програмному середовищі MockFlow. Процес розробки графічних схем було розпочато зі створення базового макета головної сторінки вебзастосунку (рис. 2.2).

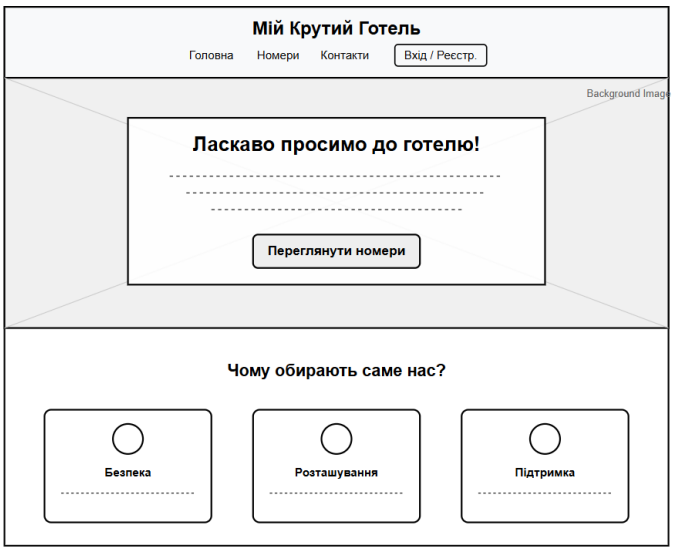


Рисунок 2.2 – Wireframe-макет головної сторінки вебсайту

Структурно головна сторінка складається з верхнього колонтитула (header) із відцентрованим заголовком і горизонтальним меню, головного графічного банера (Hero Image) та контентного блоку з перевагами готелю. З головної сторінки забезпечується інтуїтивно зрозуміла навігація до всіх ключових розділів вебзастосунку, зокрема: «Номери» (каталог), «Контакти», сторінок автентифікації («Вхід / Реєстрація»), а також до персонального розділу «Мої бронювання» для авторизованих клієнтів.

Окремої уваги заслуговує сторінка каталогу номерів. Вона є основним функціональним ядром проєкту і містить вичерпну інформацію про доступний номерний фонд: візуальне представлення, опис характеристик апартаментів, вартість проживання та функціональні елементи (кнопки) для ініціалізації процесу бронювання (рис. 2.3).

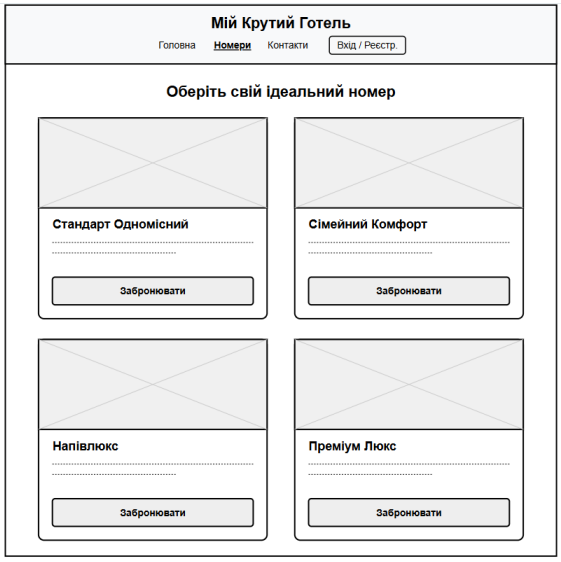


Рисунок 2.3– Wireframe-макет сторінки вебсайту «Номери»

Проєктування схематичних макетів (wireframing) є фундаментальним етапом життєвого циклу створення вебзастосунку. Насамперед, застосування цієї методології дає змогу сформувати чітку візуальну модель інформаційної архітектури ще до написання програмного коду, абстрагуючись від графічного оформлення. Це дозволяє оптимізувати користувацький досвід (UX) та налагодити інтуїтивну навігацію на стадії планування, запобігаючи необхідності масштабного рефакторингу коду в майбутньому і суттєво економлячи час. Крім того, структурні прототипи виступають універсальним інструментом узгодження технічних рішень, гарантуючи єдність концепції розроблюваного продукту. Як наслідок, попереднє макетування є обов'язковою передумовою створення якісної, зручної та ергономічної інформаційної системи.

2.2.2 UML-діаграма розгортання

Діаграми розгортання мови UML виступають невід'ємним інструментом на етапі проєктування та впровадження інформаційної системи. Вони забезпечують наочне графічне відображення фізичної топології

розроблюваного рішення, ілюструючи розподіл програмних артефактів по апаратних обчислювальних вузлах.

Нижче на рисунку 2.4 наведено UML-діаграму розгортання, яка ілюструє фізичну архітектуру вебзастосунку та описує ланцюг інформаційної взаємодії від клієнтського пристрою (браузера) через обробник на вебсервері безпосередньо до сервера бази даних.

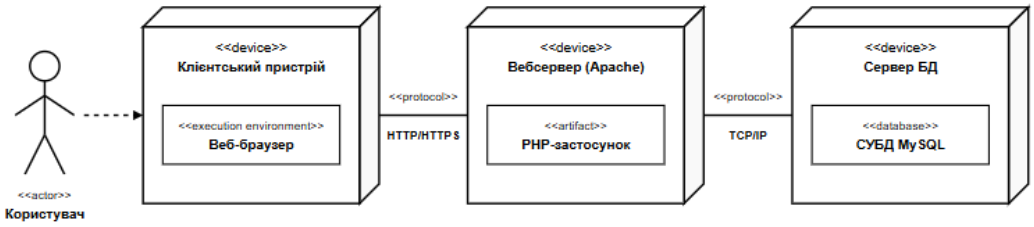


Рисунок 2.4 – UML-діаграма розгортання

Таким чином, моделювання фізичної архітектури за допомогою UML-діаграм розгортання виступає важливим етапом створення інформаційної системи. Цей метод забезпечує системне бачення інфраструктури розробленого вебзастосунку, створюючи технічне підґрунтя для раціонального розподілу ресурсів, підвищення продуктивності обробки даних та посилення безпеки серверного середовища.

2.3 UML-діаграма прецедентів

Діаграма прецедентів (Use Case diagram), що зображена на рисунку 2.5, дозволяє чітко визначити та структуризувати функціональні вимоги до програмного продукту. Деталізований опис взаємозв'язків між ідентифікованими акторами (ролями користувачів) та прецедентами (варіантами використання системи) наведено у таблиці 2.1.

Таблиця 2.1 – Зв’язок акторів і прецедентів

Актор	Прецедент
Клієнт	Перегляд загальної інформації про готель на головній сторінці. Ознайомлення з каталогом номерного фонду та детальними характеристиками апартаментів. Реєстрація нового персонального облікового запису в системі. Авторизація (вхід до системи) та безпечний вихід з облікового запису. Здійснення онлайн-бронювання обраного готельного номера. Перегляд історії та статусів власних транзакцій (бронювань) в особистому кабінеті. Відправлення запитів або повідомлень адміністрації через форму зворотного зв'язку
Адміністратор	Управління номерним фондом (додавання нових номерів, редагування описів, зміна вартості, видалення). Моніторинг та обробка клієнтських бронювань (перегляд, підтвердження, зміна статусу або скасування). Перегляд облікових записів зареєстрованих клієнтів готелю. Опрацювання звернень та повідомлень, отриманих від користувачів через сторінку контактів. Оновлення контенту та інформаційних блоків вебзастосунку

Опис прецедентів актора «Клієнт»:

1. Створення облікового запису та авторизація

- Надає можливість неавторизованому відвідувачу створити персональний профіль та здійснити вхід до захищеної частини системи.
- Основний потік подій: Користувач переходить на сторінку реєстрації, заповнює форму з особистими даними (ім'я, email, пароль). Після успішного внесення даних до бази користувач здійснює вхід, вводячи облікові дані у форму авторизації, та отримує доступ до функціоналу бронювання.

2. Огляд номерного фонду готелю

- Надає користувачу можливість ознайомитися з доступними для проживання апартаментами та їхніми характеристиками.
- Основний потік подій: Користувач переходить до публічного розділу «Номери» (каталог), переглядає перелік доступних кімнат, ознайомлюється з їхнім візуальним представленням, описом зручностей, місткістю та актуальною вартістю проживання.

3. Оформлення бронювання номера

- Надає авторизованому користувачу інструментарій для створення заявки на резервування обраного номера на певний період.
- Основний потік подій: Авторизований клієнт обирає номер із каталогу, переходить до форми оформлення бронювання, вказує бажані дати заїзду та виїзду, перевіряє контактну інформацію та натискає кнопку підтвердження. Система валідує дані та фіксує транзакцію у базі даних.

4. Перегляд історії бронювань

- Забезпечує клієнту доступ до структурованої інформації про всі його поточні та минулі резервування.
- Основний потік подій: Користувач авторизується в системі, переходить до захищеного розділу «Мої бронювання», після чого система динамічно генерує та відображає таблицю з переліком його замовлень, датами та їхніми поточними статусами.

5. Відправлення запиту до адміністрації

- Надає можливість користувачу надіслати повідомлення або запитання представникам готелю.
- Основний потік подій: Користувач переходить до розділу «Контакти», заповнює спеціальну форму зворотного зв'язку (вводить ім'я, email, текст звернення) та ініціює відправку. Система зберігає звернення для подальшого опрацювання адміністратором.

Опис прецедентів актора «Адміністратор»:

1. Управління клієнтськими бронюваннями

- Надає адміністратору можливість контролювати та обробляти вхідні заявки на резервування номерів від клієнтів.
- Основний потік подій: Адміністратор здійснює вхід до панелі керування, переходить до списку бронювань, переглядає деталі нових заявок (дані клієнта, обраний номер, дати), звіряє доступність та змінює статус замовлення (наприклад, «Підтверджено» або «Скасовано»).

2. Управління номерним фондом

- Забезпечує можливість адміністратору актуалізувати інформацію про об'єкти готелю, що відображаються у каталозі.
- Основний потік подій: Адміністратор входить до системи, переходить до розділу управління номерами, заповнює форму для додавання нового номера (або обирає існуючий для редагування), вказує опис, завантажує фотографії, встановлює ціну та натискає «Зберегти». Система оновлює відповідні записи в СУБД.

3. Моніторинг профілів клієнтів

- Надає адміністратору інструменти для перегляду та адміністрування облікових записів зареєстрованих користувачів.
- Основний потік подій: Адміністратор відкриває розділ управління клієнтами, де система виводить перелік зареєстрованих осіб. Адміністратор має змогу обрати конкретний профіль для детального огляду контактної інформації або історії його активності в системі.

4. Опрацювання звернень користувачів

- Забезпечує адміністратору можливість своєчасно реагувати на запити та відгуки, залишені клієнтами.
- Основний потік подій: Адміністратор відкриває модуль вхідних повідомлень, детально вивчає зміст запиту клієнта, отримує його контактні дані (email або телефон) та надає кваліфікований зворотний зв'язок поза межами системи (за допомогою корпоративної пошти або телефонного дзвінка).

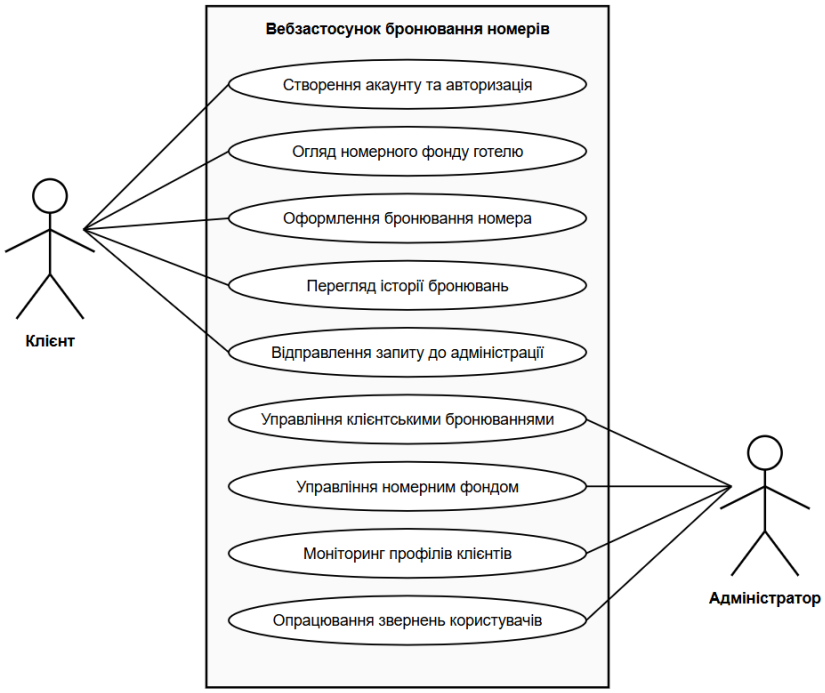


Рисунок 2.5 – UML-діаграма прецедентів

2.4 Схема бази даних

Проєктування бази даних було виконано відповідно до технічного завдання розроблюваного вебзастосунку.

Модель бази даних системи бронювання готелю складається з п’яти сутностей:

1. users – дані про облікові записи зареєстрованих клієнтів (ідентифікаційні та контактні дані, хешовані паролі);

2. room_types – загальна інформація про категорії номерного фонду (назва, опис, місткість, базова вартість за добу та візуальне представлення);

3. rooms – дані про конкретні фізичні номери готелю та їхній поточний експлуатаційний статус;

4. bookings – інформація про транзакції бронювання (дати заїзду та виїзду, загальна вартість замовлення, статус резервування);

5. contact_messages – дані про звернення та повідомлення, надіслані користувачами через контактну форму.

Кожній сутності моделі відповідає окрема таблиця в базі даних. Серед атрибутів кожної таблиці обов'язково присутній атрибут «id», що генерується автоматично та є первинним ключем (Primary Key).

За допомогою зв’язків між таблицями (через зовнішні ключі – Foreign Keys) було реалізовано цілісну реляційну модель. Зокрема, таблиця конкретних кімнат (rooms) логічно підпорядкована таблиці категорій (room_types), а таблиця bookings слугує сполучною ланкою, що фіксує зв'язок між клієнтом (users) та забронюванним об'єктом (rooms). Для ідентифікації авторства звернень таблицю повідомлень (contact_messages) також інтегровано в загальну структуру через зв'язок із таблицею користувачів (рис. 2.6).

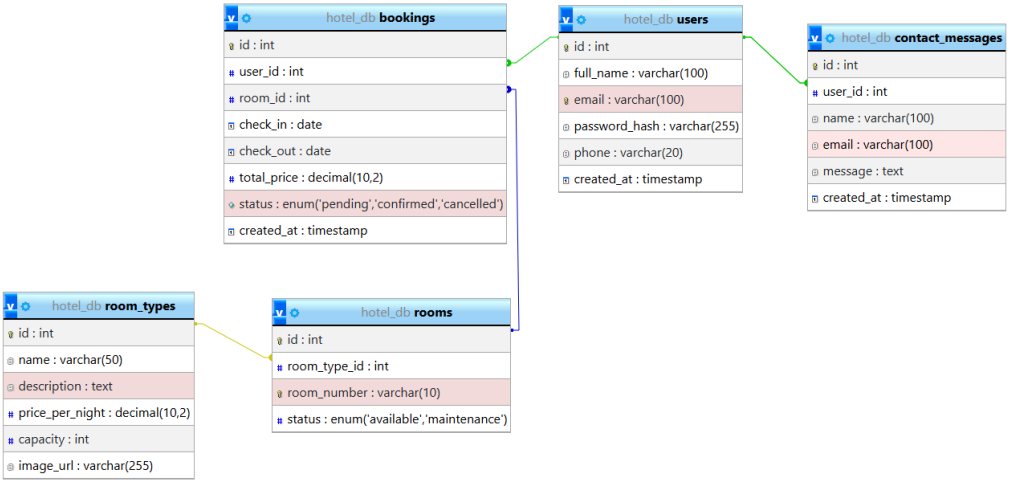


Рисунок 2.6 – Схема бази даних

2.5 Проєктування інтерфейсу

З метою забезпечення високого рівня ергономіки та зручності взаємодії клієнтів із системою бронювання, було розроблено інтуїтивно зрозумілий графічний інтерфейс вебзастосунку.

На рисунку 2.7 представлено макет екранної форми реєстрації нового користувача, тоді як на рисунку 2.8 наведено інтерфейс форми авторизації для доступу до існуючого облікового запису.

Навігація між модулями створення профілю та входу здійснюється за допомогою відповідних гіперпосилань, інтегрованих безпосередньо у візуальні блоки карток. Ініціалізація процесу обробки введених облікових даних відбувається після натискання акцентованих кнопок «Зареєструватися» або «Увійти». Технічно перехід між цими формами реалізовано за класичною моделлю багатосторінкового сайту – шляхом перенаправлення користувача на відповідні окремі вебсторінки (HTML-документи).

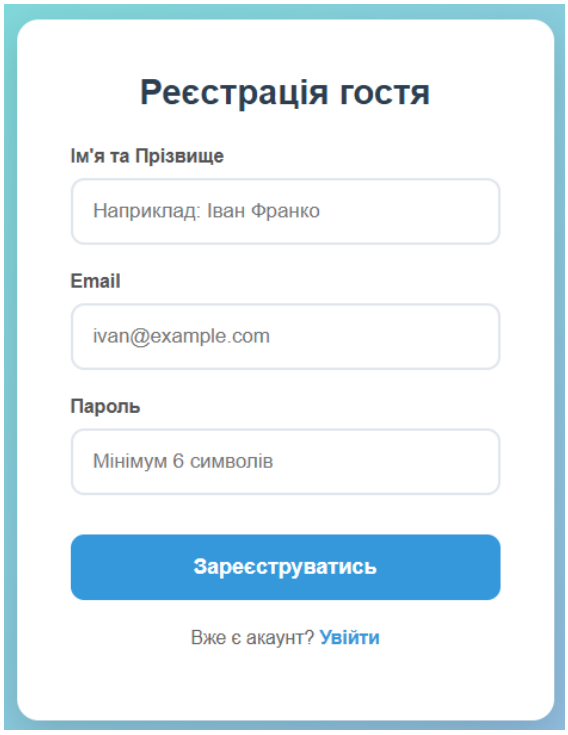


Рисунок 2.7 – Макет форми реєстрації в системі

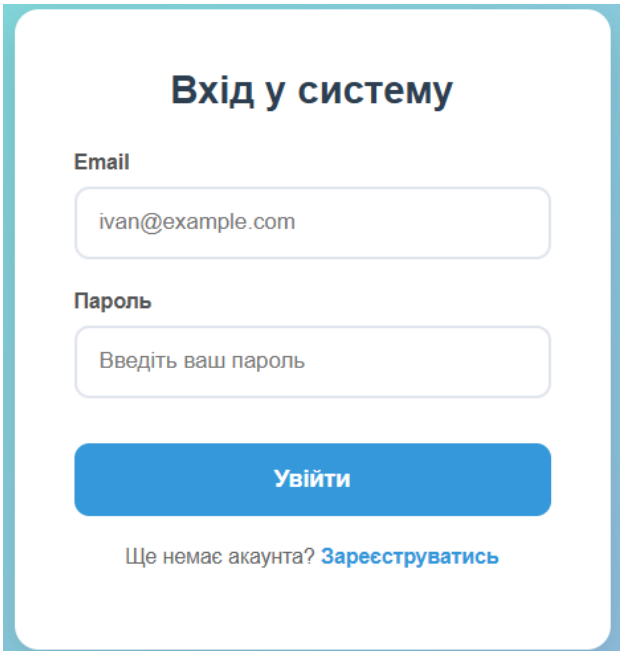


Рисунок 2.8 – Макет форми входу до системи

На рисунку 2.9 представлено макет вебсторінки каталогу доступного номерного фонду готелю. Для забезпечення зручності сприйняття об'ємного візуального та текстового контенту, замість застарілого табличного подання, інформацію було структуровано у вигляді адаптивної сітки незалежних карток (Grid/Flexbox layout). Такий підхід дозволяє ергономічно розмістити фотографії апартаментів, опис їхніх характеристик та елементи взаємодії (кнопки бронювання), запобігаючи перевантаженню інтерфейсу та забезпечуючи комфортне гортання сторінки як на десктопних, так і на мобільних пристроях.

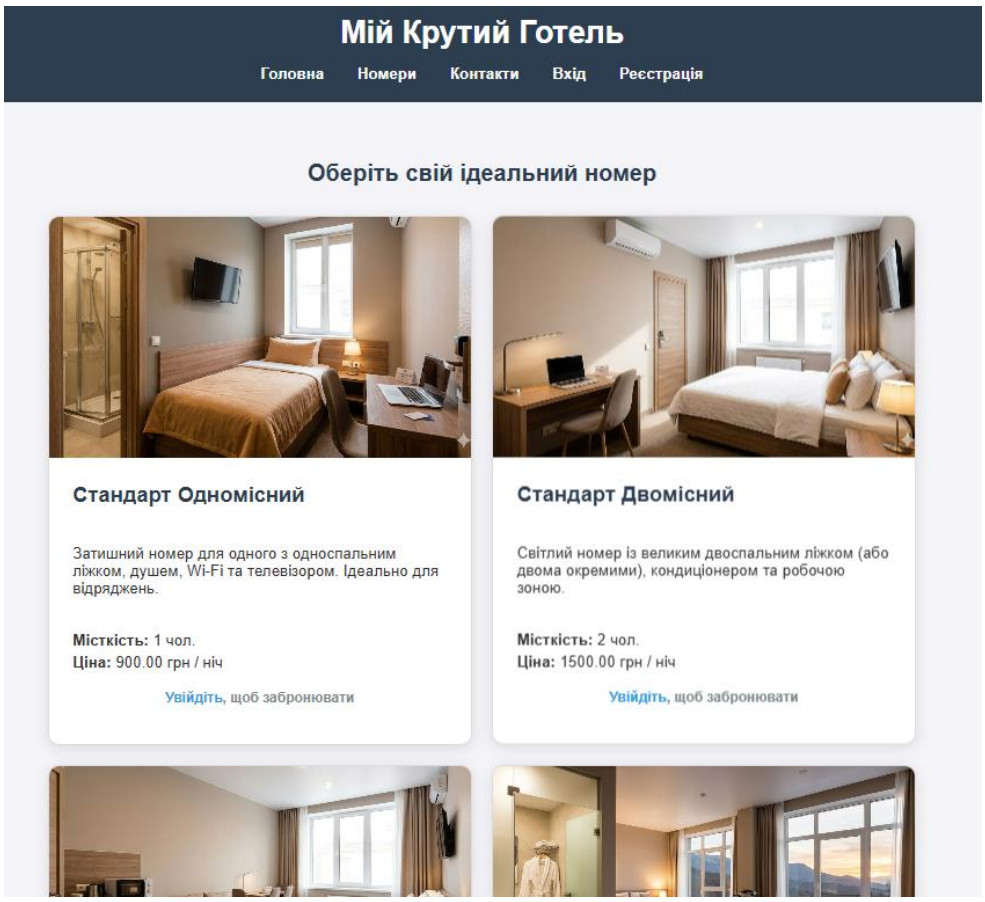


Рисунок 2.9 – Макет відображення вебсторінки із доступними номерами

Формування візуального стилю системи базувалося на принципах мінімалізму та чіткої візуальної ієрархії. З метою полегшення сприйняття об'ємних блоків даних було застосовано спокійні, ненав'язливі відтінки, що не відвертають увагу від цільових дій (бронювання номерів). Завдяки цьому вдалося врахувати всі ключові аспекти правильної візуалізації контенту та забезпечити позитивний користувацький досвід (User Experience).

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБЗАСТОСУНКУ

3.1 Створення серверної частини

Архітектура спроєктованого вебзастосунку передбачає постійну двосторонню взаємодію з даними: їх збір, обробку та подальше збереження, а також динамічне вилучення для відображення актуального контенту (наприклад, каталогу номерів, статусів бронювання чи персональних даних клієнта). Для забезпечення надійного зберігання, цілісності та швидкої вибірки цієї інформації було розроблено реляційну базу даних із використанням системи керування базами даних MySQL.

3.1.1 Реалізація бази даних застосунку

На основі попередньо спроєктованої реляційної моделі було здійснено етап фізичного розгортання бази даних: створено відповідні таблиці, визначено типи даних для атрибутів (полів) та налаштовано зв’язки за допомогою зовнішніх ключів. Практична імплементація схеми бази даних та її подальше адміністрування виконувалися за допомогою візуального інструментарію вебзастосунку phpMyAdmin. У результаті було успішно згенеровано всі необхідні структури даних для функціонування готельної системи. Приклад конфігурації полів однієї зі створених таблиць проілюстровано на рисунку 3.1.

#	Ім'я	Тип	Зіставлення	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
1	id	int			Ні	*Немає*		AUTO_INCREMENT	Більше
2	name	varchar(50)	utf8mb4_unicode_ci		Ні	*Немає*			Більше
3	description	text	utf8mb4_unicode_ci		Так	*NULL*			Більше
4	price_per_night	decimal(10,2)			Ні	*Немає*			Більше
5	capacity	int			Ні	*Немає*			Більше
6	image_url	varchar(255)	utf8mb4_unicode_ci		Так	*NULL*			Більше

Рисунок 3.1 – Структура сутності «room_types»

Графічне середовище phpMyAdmin забезпечує гнучкий інструментарій для конфігурації кожного поля таблиці, дозволяючи точно визначати його назву, тип даних, розмірність та додаткові специфічні атрибути (наприклад, первинні ключі чи обмеження NOT NULL). Зокрема, таблиця room_types акумулює в собі вичерпну інформацію щодо категорій номерного фонду готелю, яка є необхідною для забезпечення повноцінного функціонування як клієнтської частини каталогу, так і панелі адміністратора. Демонстрацію фактичного наповнення цієї таблиці тестовими записами наведено на рисунку 3.2.

←T→	id	name	description	price_per_night	capacity	image_url
	1	Стандарт Одномісний	Затишний номер для одного з односпальним ліжком, д...	900.00	1	img/std_single.jpg
	2	Стандарт Двомісний	Світлий номер із великим двоспальним ліжком (або д...	1500.00	2	img/std_double.jpg
	3	Сімейний Комфорт	Просторий номер для всієї родини. Включає двоспаль...	2200.00	4	img/family.jpg
	4	Напівлюкс	Номер покращеного планування із зоною для відпочин...	3000.00	2	img/junior_suite.jpg
	5	Преміум Люкс	Розкішні двокімнатні апартаменти з власною джакузі...	5500.00	2	img/lux.jpg

Рисунок 3.2 – Вміст таблиці «room_types»

3.1.2 Реалізація функцій

Усі операції з обробки даних, зокрема додавання нових записів, їх зчитування, оновлення та видалення, було реалізовано за допомогою серверної мови програмування PHP у поєднанні зі структурованими запитами SQL. Динамічне виведення інформації на вебсторінки «Каталог номерів» та

«Контакти» здійснюється шляхом виконання PHP-скриптів, які встановлюють з’єднання з базою даних, вилучають необхідні масиви даних та інтегрують їх у HTML-розмітку сторінки. Фрагмент програмного коду, що відповідає за отримання списку номерів із таблиці room_types, наведено нижче:

```
$sql = "SELECT * FROM room_types";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
   while($row = $result->fetch_assoc()) {
      // вивід даних у картки
   }
}
```

Алгоритм реалізації сторінки вибору номерів базується на виконанні SQL-запиту SELECT для вилучення всіх доступних категорій номерного фонду. Програмний код забезпечує підключення до сервера бази даних, перевірку успішності запиту та подальшу ітераційну обробку результатів за допомогою циклу while. Кожен отриманий запис (кортеж) інкапсулюється в окремий HTML-блок (картку), де дані з полів name, price_per_night та image_url динамічно підставляються у відповідні елементи інтерфейсу. Це забезпечує актуальність відображуваної інформації без необхідності статичного редагування HTML-коду.

```
<?php
// Ініціалізація з'єднання з базою даних
require_once 'db_connect.php';

// Формування SQL-запиту для отримання категорій номерів
$sql = "SELECT id, name, description, price_per_night, image_url, capacity FROM room_types";
$result = $conn->query($sql);

// Перевірка наявності записів у базі даних
if ($result->num_rows > 0) {
   // Циклічна обробка кожного рядка результатів вибірки
   while($row = $result->fetch_assoc()) {
      echo '<div class="room-card">';
```

```
        echo '          <img src="img/' . htmlspecialchars($row['image_url']) . '" alt="' .
htmlspecialchars($row['name']) . '">';
        echo '    <div class="room-info">';
        echo '        <h3>' . htmlspecialchars($row['name']) . '</h3>';
        echo '        <p>' . htmlspecialchars($row['description']) . '</p>';
        echo '        <div class="details">';
        echo '            <span>Місткість: ' . $row['capacity'] . ' осіб</span>';
        echo '            <span class="price">Ціна: ' . $row['price_per_night'] . ' грн/доба</span>';
        echo '        </div>';
        echo '          <a href="booking.php?type_id=' . $row['id'] . '" class="btn-book">Обрати
номер</a>';
        echo '    </div>';
        echo '</div>';
    }
} else {
    echo "<p>Наразі доступних номерів не знайдено.</p>";
}
?>
```

Процес реєстрації нового користувача в системі реалізовано через обробку HTTP-запиту методом POST. Серверна частина виконує фільтрацію вхідних даних для запобігання XSS-атакам та перевірку унікальності електронної пошти за допомогою умовного оператора. Ключовим аспектом безпеки є застосування алгоритму password_hash() для перетворення відкритого пароля користувача у незворотний криптографічний хеш перед його збереженням у базі даних. Фізичне внесення даних здійснюється за допомогою підготовлених SQL-запитів, що гарантує захист від SQL-ін'єкцій та забезпечує цілісність інформаційної структури системи.

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Збір та первинна фільтрація даних з форми
    $full_name = mysqli_real_escape_string($conn, $_POST['full_name']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $password = $_POST['password'];
    $phone = mysqli_real_escape_string($conn, $_POST['phone']);

    // Хешування пароля для безпечного зберігання
    $hashed_password = password_hash($password, PASSWORD_DEFAULT);
```

```
    // Перевірка наявності користувача з такою ж поштою
    $check_user = "SELECT id FROM users WHERE email = '$email'";
    $res = $conn->query($check_user);

    if ($res->num_rows > 0) {
        $error = "Користувач з таким Email вже існує!";
    } else {
        // SQL-запит на додавання нового користувача
        $sql = "INSERT INTO users (full_name, email, password_hash, phone)
                VALUES ('$full_name', '$email', '$hashed_password', '$phone')";

        if ($conn->query($sql) === TRUE) {
            header("Location: login.php?success=registered");
            exit();
        } else {
            $error = "Помилка при реєстрації: " . $conn->error;
        }
    }
}
?>
```

3.2 Створення клієнтської частини

Невід'ємним компонентом розробленого вебзастосунку є клієнтська частина, яка забезпечує безпосередню взаємодію з користувачем. Структурування контенту та логічне розміщення всіх функціональних елементів було реалізовано за допомогою мови гіпертекстової розмітки HTML. Візуальне оформлення, що повністю відповідає попередньо спроєктованим UI-макетам, виконано із застосуванням каскадних таблиць стилів (CSS). Завдяки чіткому дотриманню принципів вебергономіки вдалося створити інтуїтивно зрозумілий користувацький інтерфейс. Використання стриманої кольорової палітри сприяє комфортному сприйняттю графічної та текстової інформації, а продумана архітектура сторінок забезпечує зручну навігацію системою бронювання. На рисунку 3.3 представлена головна сторінка готельного вебзастосунку, яка виконує роль центрального вузла для переходу до всіх інших розділів сайту.

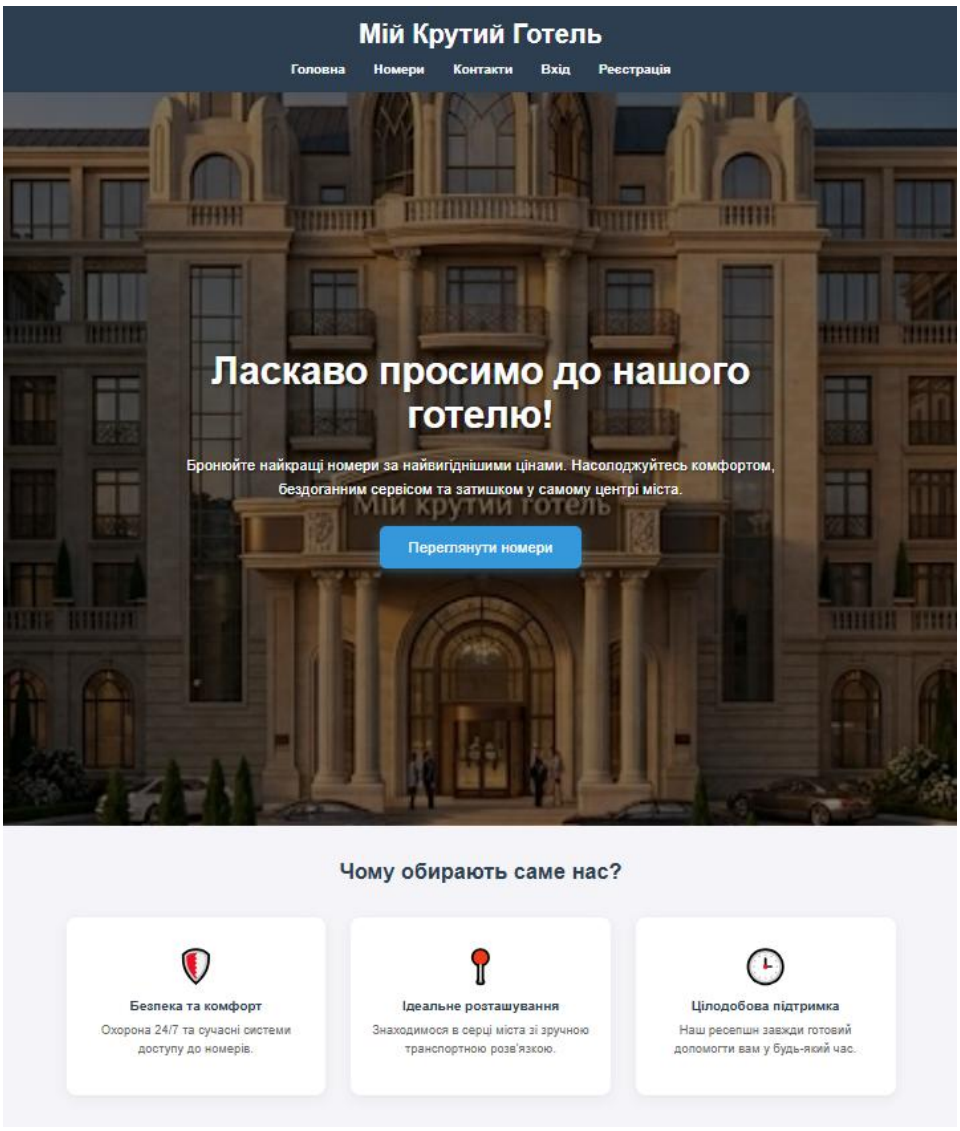


Рисунок 3.3 – Головна сторінка вебсайту

Структура головної вебсторінки включає інформативний заголовок, панель навігації для швидкого переходу між функціональними розділами, а також тематичне графічне зображення (банер), що візуально ідентифікує цільове призначення застосунку – надання готельних послуг. За допомогою відповідного розділу меню користувач отримує можливість детально ознайомитися з усім переліком апартаментів, доступних для проживання та подальшого бронювання (рис. 3.4).

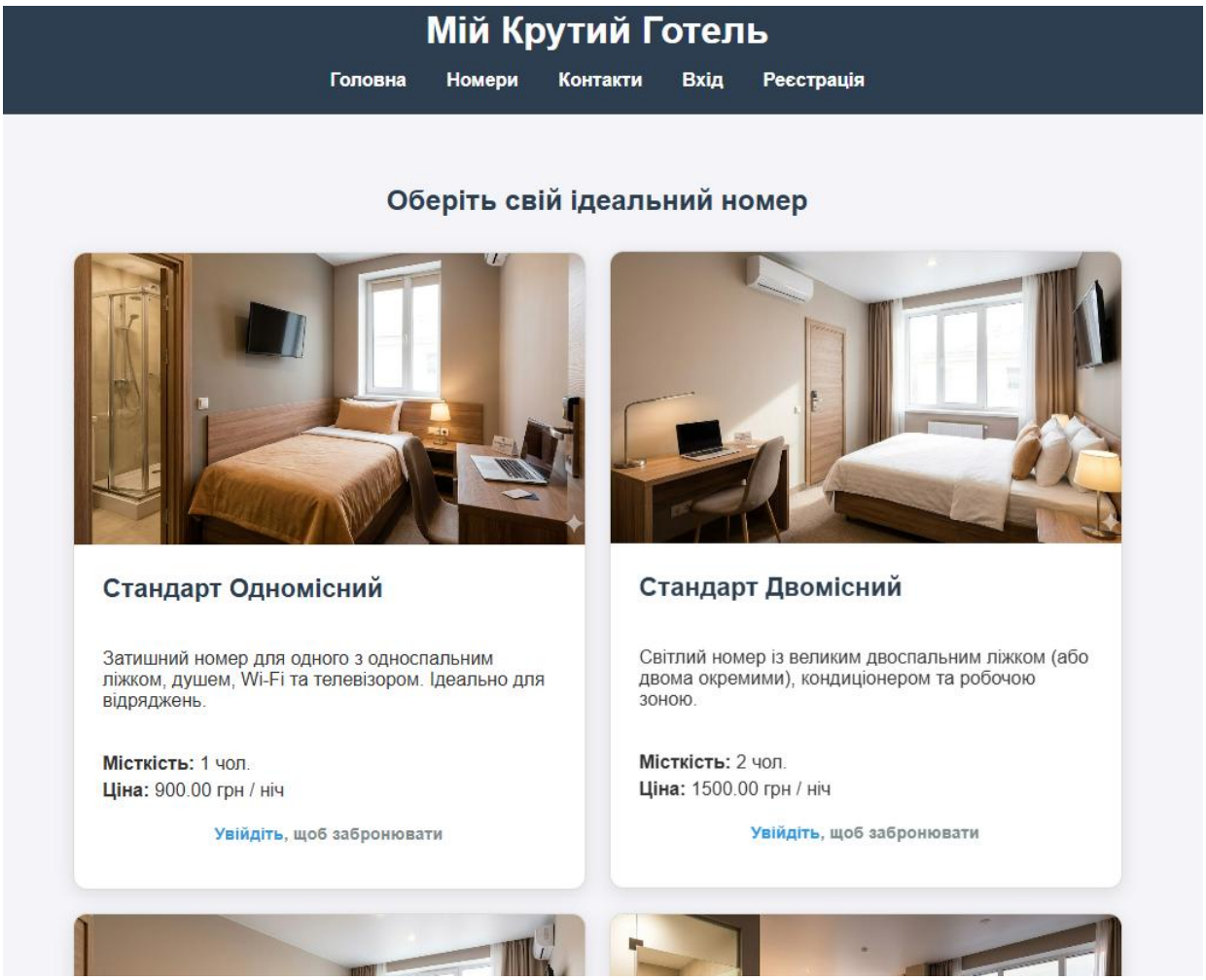


Рисунок 3.4 – Вебсторінка «Номери»

Інформацію щодо всього доступного номерного фонду готелю було ергономічно структуровано у вигляді адаптивної сітки карток з використанням сучасних технологій компонування (CSS Flexbox/Grid). Для кожної категорії апартаментів зазначено назву, вартість проживання за добу, а також наведено репрезентативну фотографію для візуального ознайомлення з умовами розміщення. Окрім цього, в структуру вебзастосунку було інтегровано сторінку «Контакти», на якій реалізовано зручну форму зворотного зв'язку. Вона надає користувачам можливість оперативно звернутися до адміністрації закладу для вирішення будь-яких питань, уточнення деталей бронювання або залишення відгуків (рис. 3.5).

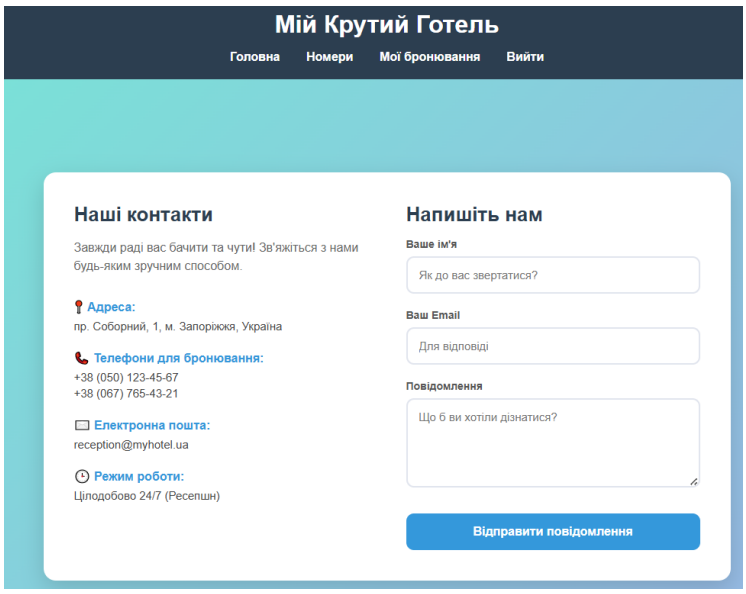


Рисунок 3.5 – Вебсторінка «Контакти»

Для неавторизованих відвідувачів функціонал вебзастосунку обмежений переважно ознайомчим рівнем: користувачі можуть переглядати номерний фонд, цінову політику та контактну інформацію. Однак, для повноцінної взаємодії із системою – безпосереднього оформлення бронювання готельних номерів та управління власними замовленнями – клієнту необхідно пройти процедуру реєстрації з подальшою авторизацією. Вебсторінки з розробленими формами для створення облікового запису та входу в систему проілюстровано на рисунках 3.6 та 3.7 відповідно.

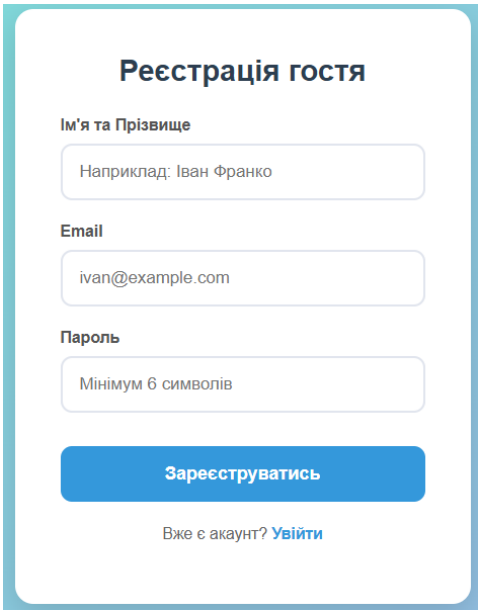


Рисунок 3.6 – Форма для реєстрації облікового запису

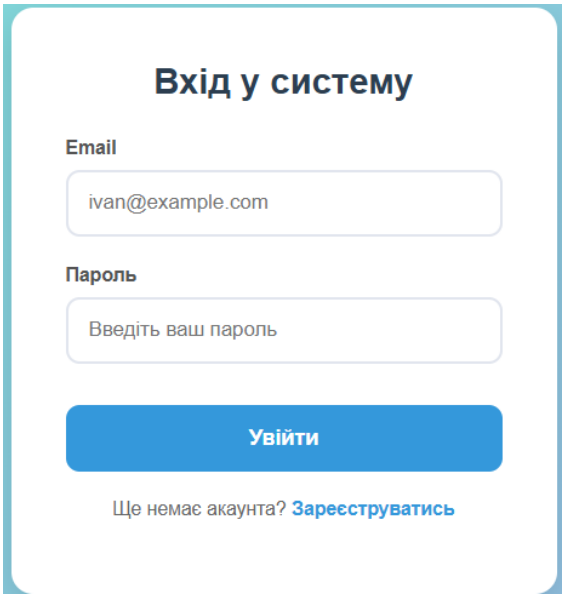


Рисунок 3.7 – Форма для авторизації

У разі успішної автентифікації в системі, вебзастосунок автоматично перенаправляє клієнта до інтерфейсу вибору номерного фонду, де представлено всі доступні для резервування категорії апартаментів. Безпосередньо під переліком номерів інтегровано інтерактивну форму для оформлення замовлення. Задля забезпечення швидкої навігації та миттєвого доступу до процесу бронювання, у головне горизонтальне меню було додано відповідне гіперпосилання, яке автоматично фокусує увагу користувача на секції з формою подачі заявки (рис. 3.8).

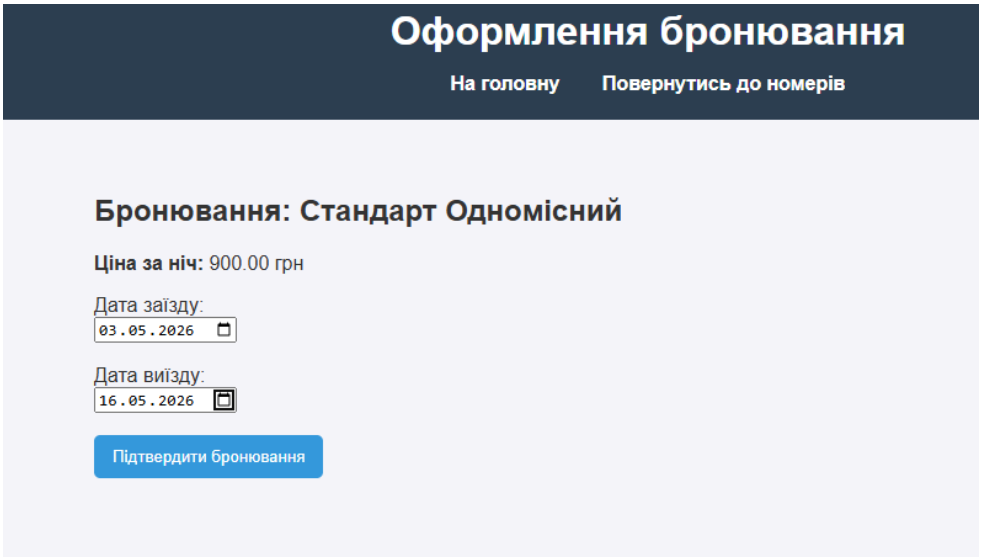


Рисунок 3.8 – Вебсторінка із формою підтвердження бронювання

Процес фіналізації замовлення передбачає введення клієнтом часових параметрів проживання, а саме дат заїзду та виїзду. Після ініціалізації запиту шляхом натискання цільової кнопки, система автоматично здійснює розрахунок загальної вартості послуг на основі актуальних тарифів. У разі коректного заповнення всіх полів, на екрані з'являється діалогове вікно або інформаційне повідомлення, яке підтверджує успішне завершення операції резервування та відображає фінальну суму до сплати (рис. 3.9). Це забезпечує користувачу миттєвий зворотний зв'язок та підтверджує фіксацію транзакції в базі даних.

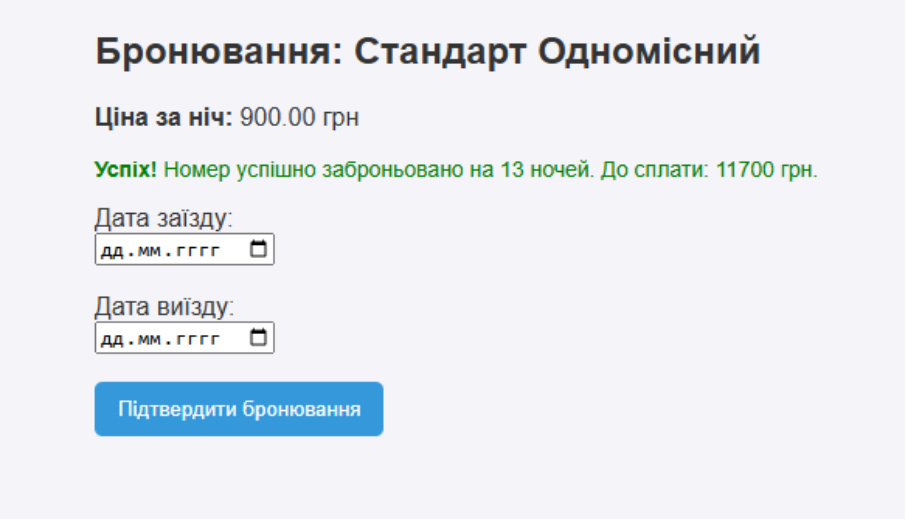
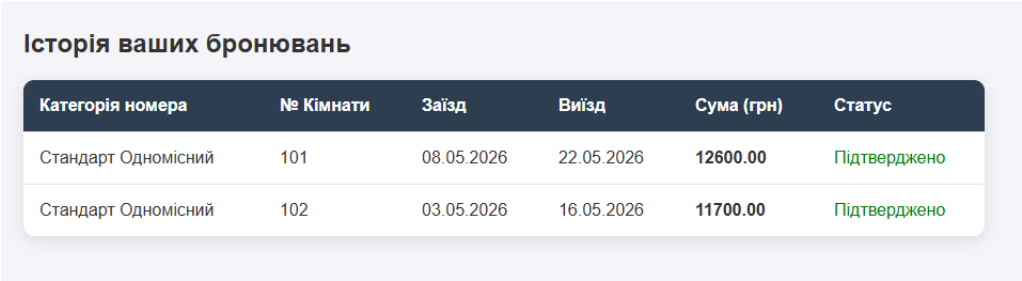


Рисунок 3.9 – Підтвердження бронювання

Після завершення введення необхідних реквізитів та активації кнопки «Забронювати», система здійснює валідацію даних і виводить інформаційне повідомлення про успішне формування запиту. Програмна логіка вебзастосунку забезпечує миттєве занесення транзакції до реляційної таблиці bookings, після чого користувач отримує можливість здійснювати моніторинг статусу та деталей свого замовлення у відповідному розділі особистого кабінету (рис. 3.10).

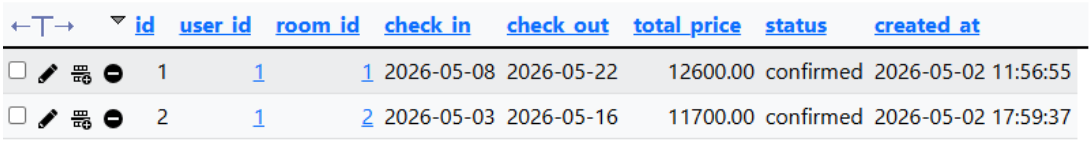


Історія ваших бронювань

Категорія номера	№ Кімнати	Заїзд	Виїзд	Сума (грн)	Статус
Стандарт Одномісний	101	08.05.2026	22.05.2026	**12600.00**	Підтверджено
Стандарт Одномісний	102	03.05.2026	16.05.2026	**11700.00**	Підтверджено

Рисунок 3.10 – Історія бронювань в особистому кабінеті

Водночас адміністратор системи має повний доступ до керування та аналізу нових заявок через інтерфейс адміністрування бази даних phpMyAdmin, де кожне бронювання відображається як окремий структурований запис із повним набором ідентифікаторів (рис. 3.11).



←T→	id	user id	room id	check in	check out	total price	status	created at
	1	1	1	2026-05-08	2026-05-22	12600.00	confirmed	2026-05-02 11:56:55
	2	1	2	2026-05-03	2026-05-16	11700.00	confirmed	2026-05-02 17:59:37

Рисунок 3.11 – Таблиця «Bookings»

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було досягнуто поставлену мету – розроблено вебзастосунок для автоматизації процесів бронювання номерного фонду готелю. Проведене дослідження існуючих готельних платформ та сервісів бронювання дозволило виявити актуальні функціональні вимоги, впровадити найкращі практики інтерфейсних рішень та уникнути типових ергономічних помилок. Сформульоване технічне завдання забезпечило чітку структуризацію вимог до функціональності, безпеки та дизайну майбутньої інформаційної системи.

У першому розділі було опрацьовано теоретичні відомості щодо архітектури сучасних вебзастосунків корпоративного сегмента. Аналіз чотирьох репрезентативних прикладів існуючих вебсистем дозволив визначити ключові тенденції у сфері веброзробки та сформувати перелік необхідних сервісів для забезпечення конкурентоспроможності готельного вебсайту. Крім того, на основі сформованих вимог було здійснено обґрунтований вибір засобів та технологій веброзробки, оптимальних для подальшої програмної реалізації проєкту.

Другий розділ присвячено проєктуванню системи. На основі розробленого технічного завдання було визначено функціональні та нефункціональні вимоги до програмного продукту. Створення wireframe-макетів дозволило візуалізувати структуру інтерфейсу та оптимізувати шлях користувача від ознайомлення з послугами до фіналізації бронювання. Розроблена діаграма варіантів використання (Use Case) чітко розмежувала ролі користувачів та адміністратора в системі, а спроєктована реляційна модель бази даних заклала фундамент для цілісного управління інформаційними потоками.

У третьому розділі детально описано етапи програмної реалізації всіх компонентів вебзастосунку. Було розроблено:

- Клієнтську частину із застосуванням HTML та CSS, що забезпечило адаптивне та коректне відображення контенту на різних пристроях.
- Серверну частину на базі мови PHP, яка відповідає за логіку обробки запитів, автентифікацію користувачів та розрахунок вартості послуг.
- Базу даних (MySQL), що гарантує надійне збереження персональних даних клієнтів та актуальних статусів номерного фонду.

Для реалізації проєкту було використано професійний інструментарій, зокрема середовище розробки Visual Studio Code та локальний сервер OpenServer, що забезпечило ефективний цикл розробки та налагодження програмного коду.

Результатом виконання роботи став повнофункціональний готельний вебсайт, який відповідає сучасним вимогам до вебсистем. Застосунок надає користувачам можливість ознайомитися з категоріями номерів, здійснити реєстрацію, обрати дати проживання та отримати миттєвий розрахунок вартості замовлення через особистий кабінет. Впровадження форми зворотного зв’язку забезпечило ефективний канал комунікації з адміністрацією.

Таким чином, усі завдання кваліфікаційної роботи виконані в повному обсязі, а поставлена мета досягнута. Розроблений вебзастосунок є дієвим інструментом для цифрової трансформації готельного бізнесу, що сприяє залученню клієнтів та оптимізації операційної діяльності підприємства.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Цифровізація бізнесу: основні етапи та переваги для підприємств. URL: https://inmind.com.ua/cifrovizaciya-biznesu (дата звернення: 01.05.2026).

2. Веб-сайт: визначення, структура та класифікація. URL: https://uk.wikipedia.org/wiki/Вебсайт (дата звернення: 01.05.2026).

3. Що таке веб-додаток і чим він відрізняється від сайту. URL: https://itproger.com/ua/news/shcho-take-veb-dodatok (дата звернення: 02.05.2026).

4. Архітектура клієнт-сервер: принципи роботи та застосування. URL: https://ru.wikipedia.org/wiki/Клиент_–_сервер (дата звернення: 02.05.2026).

5. Frontend розробка: базові технології HTML, CSS та JavaScript. URL: https://dou.ua/lenta/articles/frontend-guide/ (дата звернення: 03.05.2026).

6. Протокол HTTPS: чому він важливий для безпеки веб-ресурсів. URL: https://hostiq.ua/wiki/what-is-https/ (дата звернення: 03.05.2026).

7. Класифікація сучасних веб-сайтів та їх цільове призначення. URL: https://web-mate.com.ua/blog/vydy-veb-saytiv (дата звернення: 04.05.2026).

8. Динамічні веб-додатки та реляційні бази даних. URL: https://dev.mysql.com/doc/ (дата звернення: 04.05.2026).

9. E-commerce платформи та системи онлайн-бронювання. URL: https://retailers.ua/uk/news/tehnologii (дата звернення: 04.05.2026).

10. Індустрія гостинності в умовах цифровізації. URL: https://hoteliero.com/uk/analytics/ (дата звернення: 05.05.2026).

11. Booking Engines: як системи бронювання змінюють готельний бізнес. URL: https://bnovo.ru/blog/booking-engine/ (дата звернення: 05.05.2026).

12. Офіційний сайт агрегатора готельних послуг Booking.com. URL: https://www.booking.com/ (дата звернення: 06.05.2026).

13. Офіційний сайт міжнародної мережі готелів Hilton. URL: https://www.hilton.com/ (дата звернення: 06.05.2026).

14. Офіційний сайт мережі Premier Hotels and Resorts. URL: https://www.phnr.com/ (дата звернення: 06.05.2026).

15 Офіційний сайт української готельної мережі Ribas Hotels Group. URL: https://ribashotelsgroup.ua/ (дата звернення: 07.05.2026).

Декларація

академічної доброчесності

здобувача освіти ВСП «Економіко-правничий фаховий коледж ЗНУ»

Я, Томченко Даніїл Денисович, здобувач освіти 4 курсу, спеціальності 121 Інженерія програмного забезпечення, освітньо-професійної програми Розробка програмного забезпечення, групи К121-22, адреса електронної пошти daniiltomcenko9@gmail.com

- підтверджую, що написана мною кваліфікаційна робота на тему «Проєктування та розробка вебдодатку з бронювання номерів у готелі» відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст. 42 Закону України «Про освіту», зі змістом яких ознайомлений/ознайомлена;

- заявляю, що надана мною для перевірки електронна версія роботи є ідентичною її друкованій версії;

- згоден/згодна на перевірку моєї роботи на відповідність критеріям академічної доброчесності у будь-який спосіб, у тому числі за допомогою інтернет-системи, а також на архівування моєї роботи в базі даних цієї системи.

Дата 05.06.2026 Підпис  Даніїл ТОМЧЕНКО

Дата _____ Підпис _____ Олена ШИТІКОВА