

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ, КОМП'ЮТЕРНОЇ ТА ПРОГРАМНОЇ
ІНЖЕНЕРІЇ
КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ УПРАВЛІННЯ**

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри

_____ Литвиненко О.Є.
«_____» _____ 2021 р.

**ДИПЛОМНИЙ ПРОЄКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)**

**ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ
«БАКАЛАВР»**

Тема: _____ «Програмний модуль сервісу бронювання квартир»

Виконавець: _____ Слюсаренко О.К.

Керівник: _____ Нечипорук В.В.

Нормоконтролер: _____ Тупота Є.В.

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет Кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютеризованих систем управління

Спеціальність 123 «Комп'ютерна інженерія»

(шифр, найменування)

Освітньо-професійна програма «Системне програмування»

Форма навчання денна

ЗАТВЕРДЖУЮ
Завідувач кафедри

Литвиненко О.Є.

« » 2021 р.

ЗАВДАННЯ

на виконання дипломної роботи (проєкту)

Слюсаренку Олександрю Костянтиновичу

(прізвище, ім'я, по батькові випускника в родовому відмінку)

1. Тема дипломної роботи (проєкту) «Програмний модуль сервісу бронювання квартир»

затверджена наказом ректора від «04» лютого 2021 р. № 135/ст.

2. Термін виконання роботи (проєкту): з 17 травня 2021 р. по 20 червня 2021 р.

3. Вихідні дані до роботи (проєкту): програмна документація, ДСТУ, мова програмування JavaScript, середовище розробки WebStorm

4. Зміст пояснювальної записки: аналіз принципів побудови програмних модулів сервісів бронювання квартир, програмний модуль сервісу бронювання квартир, програмне забезпечення модуля сервісу бронювання квартир.

5. Перелік обов'язкового графічного (ілюстративного) матеріалу:

1) структурна схема програмного модуля;

2) функціональна схема програмного модуля;

3) схема алгоритму бронювання квартири;

4) схема алгоритму залишення відгуку;

5) зовнішній вигляд основних вікон програмного модуля сервісу бронювання квартир.

6. Календарний план-графік

№ пор.	Завдання	Термін виконання	Відмітка про виконання
1.	Провести огляд літератури за темою дипломного проекту та аналіз існуючих систем.	17.05.2021 – 18.05.2021	
2.	Зробити вибір середовища програмування і компонентів програмної утиліти	18.05.2021 – 20.05.2021	
3.	Розробити структуру програмних засобів системи	21.05.2021 – 22.05.2021	
4.	Розробити програмні засоби. Провести відладку програмних засобів	23.05.2021 – 05.06.2021	
5.	Написати пояснювальну записку	06.05.2021 – 10.06.2021	
6.	Підготувати графічний та ілюстративний матеріал	11.06.2021 – 12.06.2021	
7.	Підготувати доповідь та презентацію	12.06.2021 – 13.06.2021	

7. Дата видачі завдання: « 17 » травня 2021 р.

Керівник дипломної роботи (проекту) _____ Нечипорук В.В.
(підпис керівника) (П.І.Б.)

Завдання прийняв до виконання _____ Слюсаренко О.К.
(підпис студента)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту «Програмний модуль сервісу бронювання квартир»: 57 сторінок, 17 рисунків, 15 літературних джерел, 3 таблиць.

ПОДОРОЖІ, ОНЛАЙН-БРОНЮВАННЯ, ОРЕНДА КВАРТИРИ, ПОШУК КВАРТИРИ, ВІДПУСКА, ТУР.

Об'єкт – процес онлайн-бронювання квартир.

Предмет – програмний модуль сервісу бронювання для оренди квартири.

Мета дипломного проекту – розробити програмний модуль сервісу бронювання квартири.

Метод проектування – мова програмування *JavaScript*, інтегроване середовище розробки програмного забезпечення *WebStorm*.

Результати дипломного проекту рекомендується використовувати під час пошуку квартир для подорожей, а також для оренди квартир на довгостроковий час.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	6
ВСТУП.....	7
РОЗДІЛ 1 ПРИНЦИПИ ПОБУДОВИ ПРОГРАМНИХ МОДУЛІВ	
СЕРВІСІВ БРОНЮВАННЯ КВАРТИР	10
1.1. Основні поняття та принципи систем онлайн-бронювання	10
1.2. Переваги і недоліки сервісів онлайн-бронювання квартир.....	13
1.3. Постановка завдання проектування	19
1.4. Висновки до розділу	20
РОЗДІЛ 2 ПРОГРАМНИЙ МОДУЛЬ СЕРВІСУ БРОНЮВАННЯ	
КВАРТИР.....	21
2.1. Функціональні вимоги.....	21
2.2. Структура програмного модуля	22
2.3. Інформаційне забезпечення	26
2.4. Висновки до розділу	29
РОЗДІЛ 3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ СЕРВІСУ	
БРОНЮВАННЯ КВАРТИР	32
3.1. Системні вимоги.....	32
3.2. Алгоритми реалізації методів роботи модуля.....	32
3.3. Інструкція користувача програмного модуля	41
3.4. Тестування роботи програмного модуля.....	45
3.5. Висновки до розділу	50
ВИСНОВКИ.....	52
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	56
ДОДАТОК А.....	58

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

IDS – Internet Distribution System (інтернет системи бронювання)

ADS – Alternative Distribution System (альтернативні системи бронювання)

GDS – Global Distribution System (глобальні системи бронювання)

CRS – Common Reporting Standard (централізована система продаж)

B2C – Business-to-Consumer (продавець-клієнт)

АСУ – Автоматизована система управління

ВСТУП

Туристична галузь є однією з найбільш перспективних, динамічних і прибуткових галузей світової економіки. Формування нових туристичних маршрутів, створення цілої низки туристичних атракцій для подорожувачів – це все сприяє зростанню туризму у світовій економіці. Одним із ключових чинників зростання конкурентоспроможності на ринку в даний час це створення власних сервісів в глобальній комп'ютерній мережі Інтернет.

В даний час Інтернет є найбільш зручним і популярним джерелом інформації, так як до Інтернету має доступ практично кожний, через це практично усі компанії мають свій інтернет ресурс, для розміщення власних сервісів, які дозволяють створювати заклади із будь-якої точки світу.

Також ще є два варіанти бронювання за допомогою комп'ютера: готель має власну Інтернет-сторінку або є членом однієї з систем Інтернет-бронювання (Академсервіс, WEB International, Nota Bene і ін.). На сьогоднішній день існує 4 глобальні системи бронювання: Amadeus, Galileo, Worldspan, Sabre. Вони використовуються більш ніж 800 тисячами тур агентств по всьому світу для бронювання туристичних послуг, в тому числі готельного розміщення. Системи стають все більш популярні з-за широти пропонованих можливостей при досить невеликих витратах.

Основним завданням сервісу бронювання квартир є швидкість, завдяки якій користувач за лічені хвилини отримує гарантію проживання. Для цього необхідно лише заповнити форму, відправити заявку і отримати підтвердження. Сервіс бронювання відправить всі необхідні документи користувачу на електронну пошту. Також у сервісі бронювання квартир повна відсутність мовного бар'єру, він доступний у будь-який час доби, так як сервіс сам перевіряє наявність квартири на обрані дати і після підтвердження вносить дані в систему.

Актуальність дипломної роботи полягає в тому, що в ній докладно розглядається процес бронювання, як важлива частина прибутковості сервісів,

розбираються функції сучасних сервісів, позитивні та негативні сторони і технологій сервісів бронювання квартир, так як вимоги клієнтів до якості сервісу звернутися стають все більш жорсткими. Обслуговування має бути швидким, ненав'язливим і якісним – це головні критерії, якими керується потенційний гість при виборі сервісу бронювання квартири. Багато сервісів пропонують клієнтам різноманітні способи попереднього замовлення місць: по телефону, факсу, пошті або через Інтернет. Саме від того як організовані зовнішні і внутрішні комунікації, як швидко надходить інформація від клієнта до сервісу і скільки часу потрібно на обробку заявки, залежить думка гостя про сервіс в цілому. Так як процес обслуговування починається з попереднього замовлення місць, тому необхідно щоб у клієнта склалося позитивну думку про організацію роботи служби бронювання.

Саме ці критерії і послужили для вибору теми даного дипломного проекту.

Об'єкт проектування – сервіс бронювання квартир.

Предмет проектування – програмний модуль сервісу для оренди квартири.

Мета дипломного проекту – розробка програмного модуля сервісу бронювання квартири.

Методи проектування – мова програмування *JavaScript*, інтегроване середовище розробки програмного забезпечення *WebStorm*.

Практичне значення отриманих результатів. Розроблена в даному дипломному проекті підсистема керування виконуючими пристроями дозволяє створити верстат з числовим програмним керуванням з високою точністю і швидкістю для обробки деталей.

В першому розділі описано основні поняття та принципи систем онлайн-бронювання; переваги та недоліки онлайн бронювання та розглянуто сучасні системи бронювання квартир та готелів. На основі цього було прийняті рішення щодо завдань до програмного модуля.

В другому розділі сформульовано функціональні вимоги до програмного модуля; розроблено структуру програмного модуля, яка складається з структурної схеми серверного блоку та клієнтського блоку; розроблено діаграму прецедентів в

якій зображено загальну логічну послідовність виконання функцій програмного модуля; розроблено функціональну схему.

В третьому розділі встановлено системні вимоги програмного модуля; програмно реалізовано алгоритми автентифікації користувачів, реалізація списку квартир і пошук, бронювання, відгуки; описано інтерфейс користувача, а також проведено ручне тестування програмного модуля.

РОЗДІЛ 1

ПРИНЦИПИ ПОБУДОВИ ПРОГРАМНИХ МОДУЛІВ СЕРВІСІВ БРОНЮВАННЯ КВАРТИР

1.1. Основні поняття та принципи систем онлайн-бронювання

Онлайн-бронювання – це бронювання, яке здійснюється через Інтернет. Даний термін застосовується по відношенню до бронювання номерів в готелі, квитків (це може бути авіа, залізничні, автобусні тощо), місць в ресторанах і театрах, прокат автомобілів тощо.

Інтернет системи бронювання (*IDS*), відомі також як Альтернативні Системи Бронювання (*ADS*), з'явилися на початку 90-х років XX століття і виступили в якості альтернативи вже існуючій системі *GDS* (Глобальні системи бронювання). На відміну від *GDS*, доступ до якої мають тільки агенти туристичних компаній, *ADS* дає можливість користуватися послугами бронювання не тільки туристичним агентам, а й приватним особам. Зайшовши на будь-який з порталів системи *ADS*, кожен клієнт може самостійно зробити вибір потрібного йому готелю і тип номера на чітко визначені дати, забронювати свій запит в режимі реального часу і моментально отримати підтвердження на адресу своєї електронної пошти [1].

Сервіс, який надає свій продукт в системах *ADS* і *GDS*, відразу отримує доступ до мільйонів потенційних клієнтів, і більш ніж 800 тисячам туристичних агентств. В такому випадку сервіс значно збільшує свою ефективність за рахунок зниження різного роду витрат.

Надана інформація про сервіси, включаючи докладний опис номерів, типів цін, відразу потрапляє на тисячі каналів існуючих систем.

Кафедра КСУ				НАУ 21 10 32 000 ПЗ			
<i>Виконав</i>	Слосаренко О.К.			ПРИНЦИПИ ПОБУДОВИ ПРОГРАМНИХ МОДУЛІВ СЕРВІСІВ БРОНЮВАННЯ КВАРТИР	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	Нечипорук В.В.				Д	10	57
<i>Консульт.</i>					123 СП-437		
<i>Норм. контр.</i>	Тупота С. В.						
<i>Голова комісії.</i>	Литвиненко О. Є.						

Потенційний клієнт з будь-якого куточка світу в будь-який момент може самостійно або за допомогою агента вибрати найбільш цікавий для нього варіант.

У той же час, отримуючи інформацію про бронювання з великої кількості джерел, сервіс витрачає мінімальні ресурси на її обробку. Централізована Система Продаж (*CRS*) дає можливість зберігати всю наявну інформацію про наявність вільних місць у готелі, ціни на них в одному місці - в так званому електронному «особистому кабінеті» готелю. Якраз саме звідси у готелю є можливість управляти продажами через всі канали, використовуючи при цьому тільки один і інтуїтивно зрозумілий інтерфейс [2].

Системами онлайн-бронювання називаються системи, в яких відображено наявність реальних вільних дат для бронювання житла, і в яких інформування власника житла про бронювання квартири клієнтом відбувається менш ніж за одну хвилину з моменту бронювання. Тобто бронювання можливо в будь-який момент часу, за будь-який мінімальний період часу до початку броні, тобто більш ніж за одну хвилину.

Існує два види бронювання, які можна зустріти на сайтах готельного та туристичного бізнесу:

Істинне онлайн-бронювання – це коли клієнт вибирає, бронює і оплачує номер самостійно через системи *ADS* або безпосередньо на сайті готелю. Для оплати клієнт використовує кредитну або дебетову карту, дані якої передаються в готель безпосередньо, і через кілька секунд клієнт отримує ваучер, який підтверджує бронювання номера або квартири.

Псевдоонлайн-бронювання або бронювання за запитом – це коли клієнт заповнює заявку на сайті готелю і відправляє її адміністратору. Протягом деякого часу клієнт чекає дзвінка співробітника служби бронювання для підтвердження броні [3].

На сьогодні системи онлайн-бронювання представляє собою автоматизовану систему управління (АСУ), яка реалізує комплекс по вирішенню проблем за технологією Клієнт-сервер.

У цей комплекс входять модуль створення запиту на бронювання; модуль інтернет оплати завдяки інтернет рішенням; модуль інформування користувача та власника квартири, модуль зв'язку між користувачем та власником, модуль контролю якості, модуль контроль вільних дат для бронювання та інші [4].

Будь-яка сучасна система являє собою складний комплекс функціональних ланок, від злагодженості роботи якого залежить успішність існування продукту на ринку [5].

Перш за все, потрібно з'ясувати принципи побудови та механізми функціонування типової системи онлайн-бронювання квартир.

Принципи онлайн-бронювання:

1) Пошук доступних пропозицій. Користувач на сайті системи бронювання вибирає маршрут, дати, вартість, кількість осіб, розташування, й інші параметри, в залежності від предмету онлайн-бронювання. Наприклад, для бронювання готелів характерні наступні основні критерії відбору: країна, місто, готель, дата заїзду, дата виїзду, кількість осіб, кількість і вік дітей.

2) Заповнення форми з контактною і платіжною інформацією (на ряді сайтів інформація заноситься в кілька етапів). Тут зазвичай користувачеві пропонується заповнити такі поля: ПІБ, телефон, E-mail, тип банківської картки, номер картки, ім'я власника картки, секретний номер CVV2 (CVC2).

3) Здійснення платежу. Зазвичай при проходженні платежу гроші спочатку блокуються на карті користувача, а потім, за умови підтвердження правильності всіх даних, списуються.

4) Отримання документа, що підтверджує бронювання. Клієнт отримує документ встановленого зразка про успішне завершення бронювання, який гарантує йому надання послуги. Це можуть бути електронний авіаквиток, ваучер на заселення та інші [6].

Ймовірним покупцем продукту може стати будь-який користувач Інтернету. Навіть сайти, які позиціонують себе як ознайомчий буклет приводять більше 20% від усіх клієнтів.

Етапи сервісів онлайн-бронювання:

1) Сайт зі списком послуг, реквізитами, цінами, картинками та іншою інформацією. В цьому випадку сайт представляє собою додатковий рекламний засіб, що дозволяє користувачам ознайомитися в режимі постійного доступу з інформацією про фірму, їх діяльність та координатами. Недолік – інформація на сайті статична без врахування думки користувачів.

2) Інтерактивний сайт. Використовується для швидкого зв'язку клієнтів з фірмою через Інтернет. Сайт здатний видавати інформацію за запитом користувача, відповідати на питання, має засоби зворотного зв'язку. Такий сайт являє собою електронний офіс фірми, що сприяє збільшенню числа потенційних клієнтів. Недолік – неможливість безпосередньої участі в реалізації послуг.

3) Сайт-магазин. Він здатний приймати платежі, бронювати, укладати договір з клієнтом, сповіщати клієнтів. Такий сайт виконує функцію повноцінного паралельного механізму реалізації послуг, що дозволяє розвантажити менеджерів, прискорити виконання рутинних операцій. Недолік – ризик, що пов'язаний з можливістю злому сайту, шахрайство.

Описані вище системи працюють за принципом *B2C (Business-to-Consumer)* взаємодія «продавець-клієнт» [7].

Комп'ютерні системи онлайн-бронювання мають велику кількість переваг над іншими каналами збуту продукції і послуг в сфері соціально-культурного сервісу і туризму. Вони надають можливість зробити бронювання в найкоротший термін, комп'ютерні системи економлять не тільки час, а ще витрати на телефон, здійснювання контролю всіх операцій, аналіз статистичних даних і інші [8].

1.2. Переваги і недоліки сервісів онлайн-бронювання квартир

Кожен рік кількість туристів, які обирають перевагу самостійній організації свого відпочинку тільки збільшується. Це сприяє розвитку створених сервісів, а й появі нових, в цілях яких є задоволення всіх потреб туриста. На даний момент

часу ринок програмних засобів, що пропонують онлайн-бронювання є великий, при цьому кожний із них від недоліків не застрахований.

Переваги онлайн-бронювання:

1) Швидкість. Онлайн-бронювання допомагає отримати гарантію приживання за пару хвилин. Всього необхідно заповнити форму на сайті, відправити заявку і отримати підтвердження. Після чого система онлайн-бронювання відправить всі необхідні документи користувачу на електронну пошту. Системі буде відомо які апартаменти зарезервовані, тому їх власник зможе заздалегідь все підготувати. Відміна онлайн-бронювання також проходить в онлайн, система оголошує дати, до початку яких користувач має можливість відмовитися від бронювання без штрафу. Але строки повернення коштів на карту встановленні в кожній системі по своєму.

2) Відсутність мовного бар'єру. В системі онлайн-бронювання користувача можуть попросити лише ввести свої персональні данні латиницею, і це може бути лише доповненням.

3) Доступне в будь-який час доби. Користувач лише заповнює форму, електронна система сама перевіряє наявність апартаментів на обрані дати і після підтвердження вносить дані в систему.

4) Великий вибір квартир, зручний інтерфейс пошуку і бронювання, який дозволяє групувати найбільш підходящі варіанти за різними критеріями.

5) Особистий кабінет в системі. Завдяки якому можна отримувати інформацію майбутніх поїздках, передивлятися інформацію, є можливість самостійно змінити своє бронювання.

6) Наявність фотографій пропонованих варіантів розміщення, які супроводжуються відгуками, туристів які там побували.

Недоліки онлайн-бронювання:

1) В системах онлайн-бронювання трапляються збої.

2) Конфіденційність даних. Система не тільки зберігає ваші персональні дані, дані карти, системи має можливість відслідковують місцезнаходження користувача, тому ціни можуть бути в одній країні вище, ніж в іншій.

3) Багато сервісів онлайн-бронювання відслідковують вашу активність, якщо вам сподобався якийсь варіант, але ви його не забронювали, то сервіс може підіймати вартість даного варіанту.

4) Не завжди рейтинг та відгуки можуть бути правдивим та фото відповідати дійсності.

5) Багато сервісів мають слабку технічну підтримку.

Існує доволі багато сервісів, які відповідають вимогам користувачів, але варто розуміти, що кожен користувач по різному може оцінювати та віддавати переваги.

1.2.1. Сервіс бронювання квартир *Airbnb*

Airbnb допоможе знайти кращі пропозиції готелів та квартир за допомогою безлічі фільтрів, які дозволяють звузити область пошуку. У каталозі представлені готелі з високими оцінками в центральних районах міста. Це підійде більшості мандрівників, які шукають апартаменти для відпочинку на кілька днів.

Результати пошуку також включають квартири, окремі кімнати, студії з двома ліжками, готелі. Серед популярних напрямків виділяються: Токіо, Париж, Лондон, Лос-Анджелес і Барселона. Кожен номер має докладний опис: наявність телебачення, *Wi-Fi*, сніданків, окремої кухні і розташування на карті. Відгуки допомагають скласти уявлення про представлений житло. Виберіть необхідну дату заїзду на календарі і сайт *Airbnb* надасть всі відповідні варіанти за вашим пошуком (рис. 1.1).

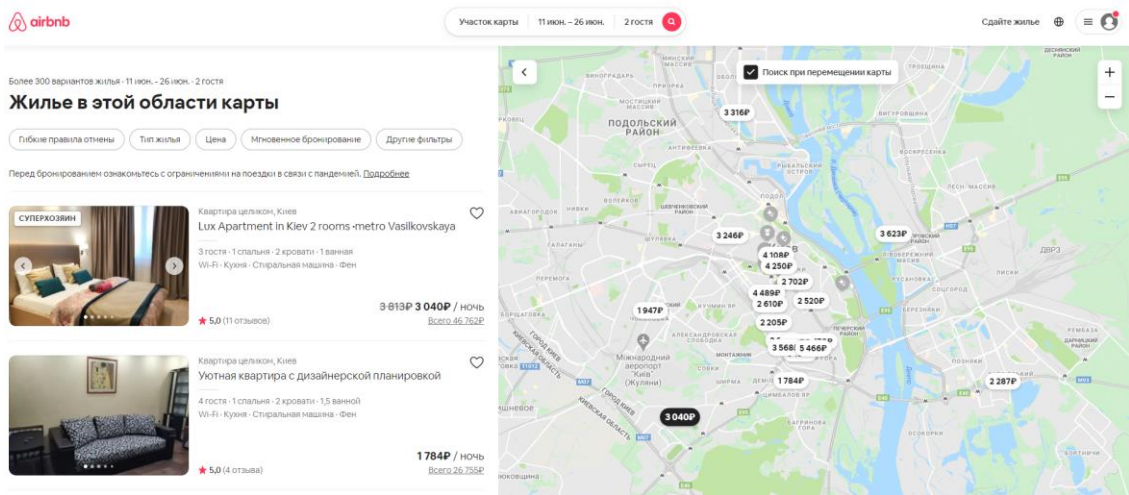


Рис. 1.1. Приклад роботи сервісу *Airbnb*

Даний сервіс має такі переваги:

- 1) Зв'язок з власником, завдяки якому користувач має можливість уточнити всі деталі та задати запитання, які йому цікаві.
- 2) Бонуси від власника. Зазвичай, в орендованій квартирі є чай, кава, печиво або навіть якась мінімальна їжа. Це все може залишити власник, або минулі мешканці.
- 3) Можливість оплати частинами, або в розстрочку.
- 4) Наявність великої і точної системи відгуків.

Недоліки:

Необхідність спілкуватися з власником перед бронюванням. Тобто перед тим як забронювати обрану квартиру, потрібно написати її господареві про можливість її бронювання. Також є – квартири з опцією миттєвого бронювання. Тут перед бронюванням зв'язуватися з власником не потрібно, але все рівно потрібно буде зв'язатися з власником перед заселенням.

1.2.2. Сервіс бронювання готелів *Booking.com*

Booking.com пропонує широкий асортимент варіантів проживання: апартаментів, гостьових будинків, сімейних пансіонатів, хостелів (Рис. 1.2). Клієнтам сайту доступна можливість бронювання авіаквитків онлайн, оренди

авто, замовлення ресторанного столика, таксі до аеропорту. Сервіс *Booking.com* підтримує 40 мов. Загальна кількість пропозицій становить 1 476 702 номерів, 121 276 напрямків 228 куточків світу.

The image shows a screenshot of the Booking.com website. On the left, there is a sidebar with filters under the heading "Выбрать по критериям:". The filters are categorized into "Сгруппированные фильтры" (3 and 4 stars), "Популярно" (Hotels, Superdeal, One double bed, Breakfast included, Apartments, Two separate beds, Hostels, Very good: 8+), and "Ваш бюджет" (Price ranges from UAH 0 to UAH 6,200+ per night). On the right, there are two hotel listings. The first is "Гостиница Измайлово Альфа" (4 stars, 8.5 rating) with a price of UAH 4,653. The second is "Apartments on Nikitskiy Bulvar 8" (9.8 rating) with a price of UAH 4,653. Both listings include photos, location information, and a "Сравнить" button.

Рис. 1.2. Приклад роботи сервісу *Booking.com*

Характеристики *Booking.com*:

- 1) внесення оплати при заселенні;
- 2) можливість скасування бронювання номера за 24 години до заїзду;
- 3) відгуки користувачів;
- 4) відгуки користувачів;
- 5) зручність інтерфейсу;
- 6) вибір оптимальних місць проживання - вілл, готелів, хостелів, номерів бізнес-класу;
- 7) фільтрування цін онлайн, відгуків, кількості зірок, варіантів харчування, розміщення, зручностей;
- 8) надання унікальних пропозицій, акцій, знижок онлайн;
- 9) додаткові опції (близьке розташування аеропорту, можливість автопрокату);

10) надання вигідних умов турагентам. Програма забезпечує простоту бронювання номерів клієнтів турагентств, заробляючи комісію.

1.2.3. Сервіс бронювання готелів *Hotels*

На сервісі *Hotels* представлений широкий вибір варіантів житла. Забронювати можна не тільки номер в готелі, а й приватні апартаменти (квартиру) або заміський будинок (рис. 1.3). Онлайн сервіс пропонує більше сотні тисяч готелів по всьому світу. На *Hotel* є бонусна програма, яка дозволяє накопичувати добу, а потім використовувати їх для оплати свого проживання.

На головній сторінці є розділ з найпопулярнішими і вигідними пропозиціями в різних країнах. Ви можете купити тур в Париж чи Рим зі знижкою до 70%.

The screenshot shows the Hotels.com website interface. At the top, there's a navigation bar with the Hotels.com logo, a language/currency selector (Ukrainian/UAH), a help link, and a user account link. Below that is a search bar with 'Київ, Україна' entered, and filters for arrival (пн, 7 черв.), departure (сб, 12 черв.), and guests (1 номер для 2 гостей). A 'ШУКАТИ' button is on the right. The main content area shows search results for 'Київ, Україна'. On the left, there are filters: 'Уточніть результати: 641 готель, будинки і квартири', 'Частина назви' (input field), 'Популярні фільтри' (checkboxes for 'Будинки і квартири', 'Зі сніданком', 'з басейном', 'Wi-Fi включено у вартість номера', 'У вартість включено автостоянку', 'для гостей з тваринами'), 'Ціна (за 5 днів): 0 UAH - 25 000 UAH+' (price range slider), and 'Кількість зірок' (radio buttons 1-5). The main results area features a 'Повідомлення для мандрівників' banner about COVID-19. Below it are two hotel listings. The first is 'Готель «Хрещатик»' (Хрещатий Яр) with a price of 19 580 UAH (13% discount) and amenities like 'Басейн', 'Доступна автостоянка', 'Трансфер з/до аеропорту', 'Спа', 'Тренажерний зал', and 'Бар'. The second is 'Premier Hotel Rus' (Печерський район) with a price of 4 944 UAH (9% discount) and amenities like 'Доступна автостоянка' and 'Доступ з домашніми улюбленицями'.

Рис. 1.3. Приклад роботи сервісу *Hotels*

Використання Hotels вимагає проходження простої реєстрації. Авторизованим користувачам доступний розширений функціонал проекту, включаючи програму *Welcome Rewards*. Особистий кабінет відображає відомості про резервованих номерах, історії замовлень. Створення закладок скоротить час подальшого пошуку кращих готелів світу. Бронювання готелів гарантована перевіреним комерційним сервером, використовують шифрування інформації кредиток. На багато пропозицій на сайті діє безкоштовна скасування бронювання. Доступний вибір між оплатою на сайті або в готелі за фактом заселення.

1.3. Постановка завдання проектування

Сервіси онлайн-бронювання квартир використовуються для вирішення різноманітних завдань, таких як створення можливості здавати в оренду власного житла, оренди мандрівниками власного житла. Сервіс надає платформу для встановлення контакту між власником квартири та гостем, а також відповідає за обробку транзакцій.

Метою дипломного проекту є розробка програмного модуля сервісу бронювання квартир.

Відповідно до проведеного аналізу переваг та недоліків сучасних програмних рішень в області сервісу бронювання квартир та відповідно до мети проектування в даному проекті поставлено наступні завдання:

- проаналізувати сучасні програмні рішення та методи створення сервісів бронювання квартир;
- реалізувати можливість роботи програмного модуля з серверною базою даних для оперування інформацією користувачів;
- розробити структурі схеми програмного модуля: клієнтську частину та серверну;
- реалізувати алгоритм створення профілю користувача, з можливістю перегляду заброньованих подорожей;

– реалізувати алгоритм бронювання відповідних апартаментів за відповідними датами;

– реалізувати в програмному модулі функції залишення відгуку, відміни бронювання та приватні повідомлення між орендатором та власником квартири;

– протестувати роботу програмного модуля.

1.4. Висновки до розділу

1) Після проведення аналізу існуючих програмних рішень сформовано вимоги до програмного модуля, які забезпечать його універсальність:

– мати сучасні програмні рішення та методи створення сервісів бронювання квартир;

– мати можливість роботи програмного модуля з серверною базою даних для оперування інформацією користувачів;

– мати структурі схеми програмного модуля: клієнтську частину та серверну;

– мати алгоритм створення профілю користувача, з можливістю перегляду заброньованих подорожей;

– мати алгоритм бронювання відповідних апартаментів за відповідними датами;

– мати в програмному модулі функції залишення відгуку, відміни бронювання та приватні повідомлення між орендатором та власником квартири.

2) Проаналізовано методи створення програмного модуля сервісу бронювання квартир та обрано мову програмування *JavaScript*, та інтегроване середовище розробки програмного забезпечення *WebStorm*.

РОЗДІЛ 2

ПРОГРАМНИЙ МОДУЛЬ СЕРВІСУ БРОНЮВАННЯ КВАРТИР

2.1. Функціональні вимоги

До функціональних вимог програмного модуля сервісу бронювання квартир належать:

- можливість роботи програмного модуля з серверною базою даних для оперування інформацією користувачів;
- можливість реєстрації нових профілів за допомогою введених даних, таких як електронна пошта, ім'я та фамілія, а також пароль;
- можливість перегляду та пошуку квартир за допомогою назви, або переміщенням карти за допомогою комп'ютерної миші;
- можливість перегляду сторінки квартири з відповідною інформацією, такою як: фотографії, назва, опис, зручності (кондиціонер, сніданок та інші), відруги.
- можливість перегляду вільних дат відповідної квартири.
- можливість обирання дати для бронювання квартири.
- можливість відправлення запиту користувачем на бронювання в системі;
- можливість залишення відгуку на сторінці квартири.
- можливість відправлення і отримання повідомлень між користувачем та власником квартири;
- можливість відхилення бронювання квартири;

Кафедра КСУ				НАУ 21 10 32 000 ПЗ			
<i>Виконав</i>	Слосаренко О.К.			ПРОГРАМНИЙ МОДУЛЬ СЕРВІСУ БРОНЮВАННЯ КВАРТИР	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	Нечипорук В.В.				Д	21	57
<i>Консульт.</i>					123 СП-437		
<i>Норм. контр.</i>	Тупота С. В.						
<i>Голова комісії.</i>	Литвиненко О. Є.						

2.2. Структура програмного модуля

На основі проведеного аналізу функціональних вимог програмного модуля сервісу бронювання квартир побудовано структурну схему. Структурна схема складається з двох частин – структурна схема серверного блоку (рис. 2.1) та структурна схема клієнтського блоку (рис. 2.2).

До структурної схеми серверного блоку входять наступні компоненти:

- база даних, елементами якої є дані користувачів; дані квартири (власник, вартість за добу, відгуки, опис та інші), дані приватних повідомлень між орендаром та власником квартири;
- блок роботи з базою даних, за допомогою якого встановлюється зв'язок між серверною частиною програмного модуля та самою базою даних;
- блок оновлення даних користувача. В ході роботи даного блоку користувач при успішній реєстрації отримує ключ, завдяки якому може повторно по ньому входити в свій профіль та користуватися функціями сервісу;
- блок оновлення даних квартири. При бронюванні користувачами, для всіх інших користувачів квартири будуть змінені.
- блок оновлення даних бронювання. В ході операції дані користувача про його бронювання заносяться в базу даних.
- блок збереження особистих даних користувача, до яких належать відомості про електронні адреси, паролі та користувацькі ідентифікатори;
- блок збереження даних квартири, при яких в базі оновлюється вільні дати для бронювання.
- блок роботи користувача з бронюванням, при якому користувач отримує дані про вільні дати для обраної квартири, обирає необхідні йому дати.

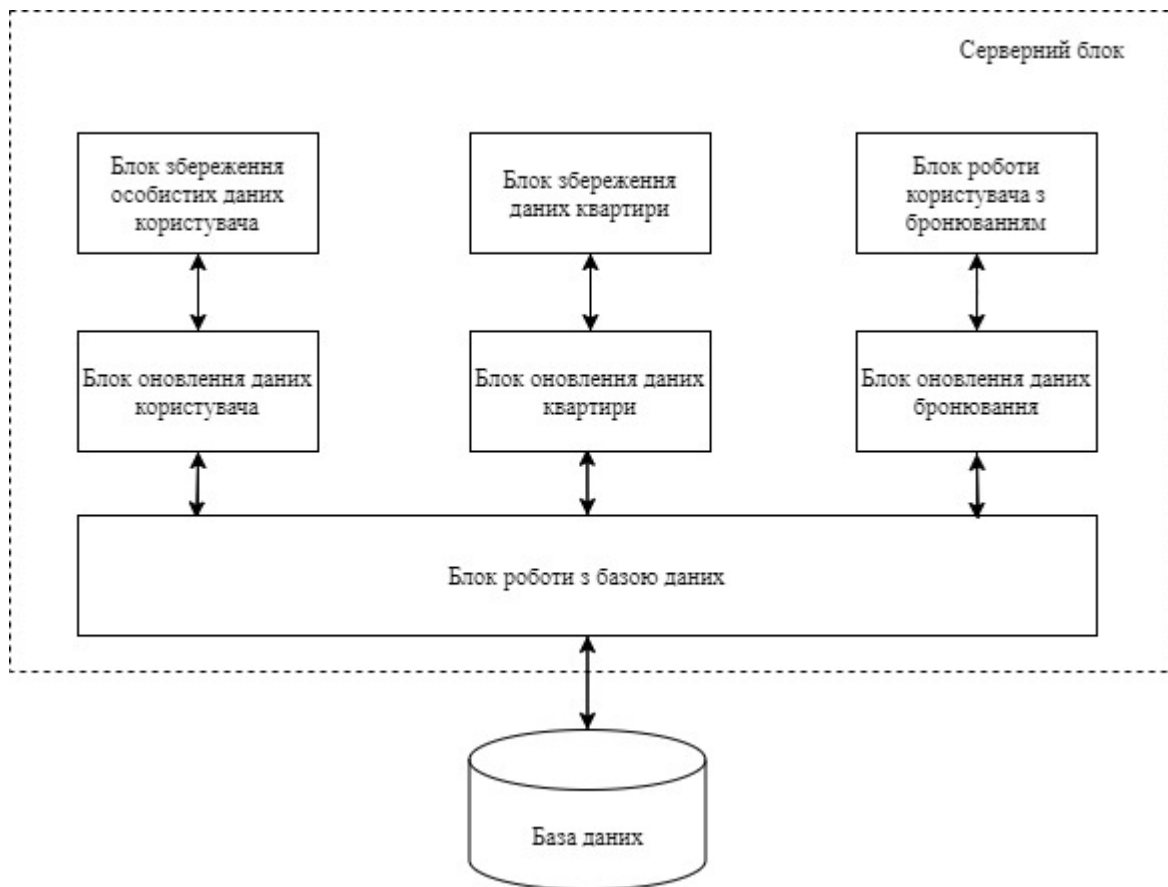


Рис. 2.1. Структурна схема серверного блоку програмного модуля

До структурної схеми клієнтського блоку входять наступні компоненти:

- база даних, елементами якої є дані користувачів; дані квартири (власник, вартість за добу, відгуки, опис та інші), дані приватних повідомлень між орендаром та власником квартири;
- блок роботи з базою даних, за допомогою якого встановлюється зв'язок між серверною частиною програмного модуля та самою базою даних;
- інтерфейс користувача, який виконує функцію передачі інформації між користувачем та блоком роботи з базою даних;
- блок роботи з профілем користувача, що надає можливість програмному модулю авторизувати користувача. Цей блок в свою чергу поділяється на блок створення нового профілю для входу та блок використання зареєстрованого профілю для входу;

– блок роботи з бронюванням користувача, що надає можливість користувачу робити бронювання обраної квартири з відповідними обраними датами, а також передавати їх в базу для подальшого оновлення в системі;

– блок роботи з профілем квартири, завдяки якому користувач має можливість побачити вільні дати, ціну за ніч та відповідні характеристики обраної квартири;

– блок відображення квартири, який функціонує через блок роботи з профілем квартири. Цей блок збирає усі можливі квартири в базі даних та надає їх користувачу;

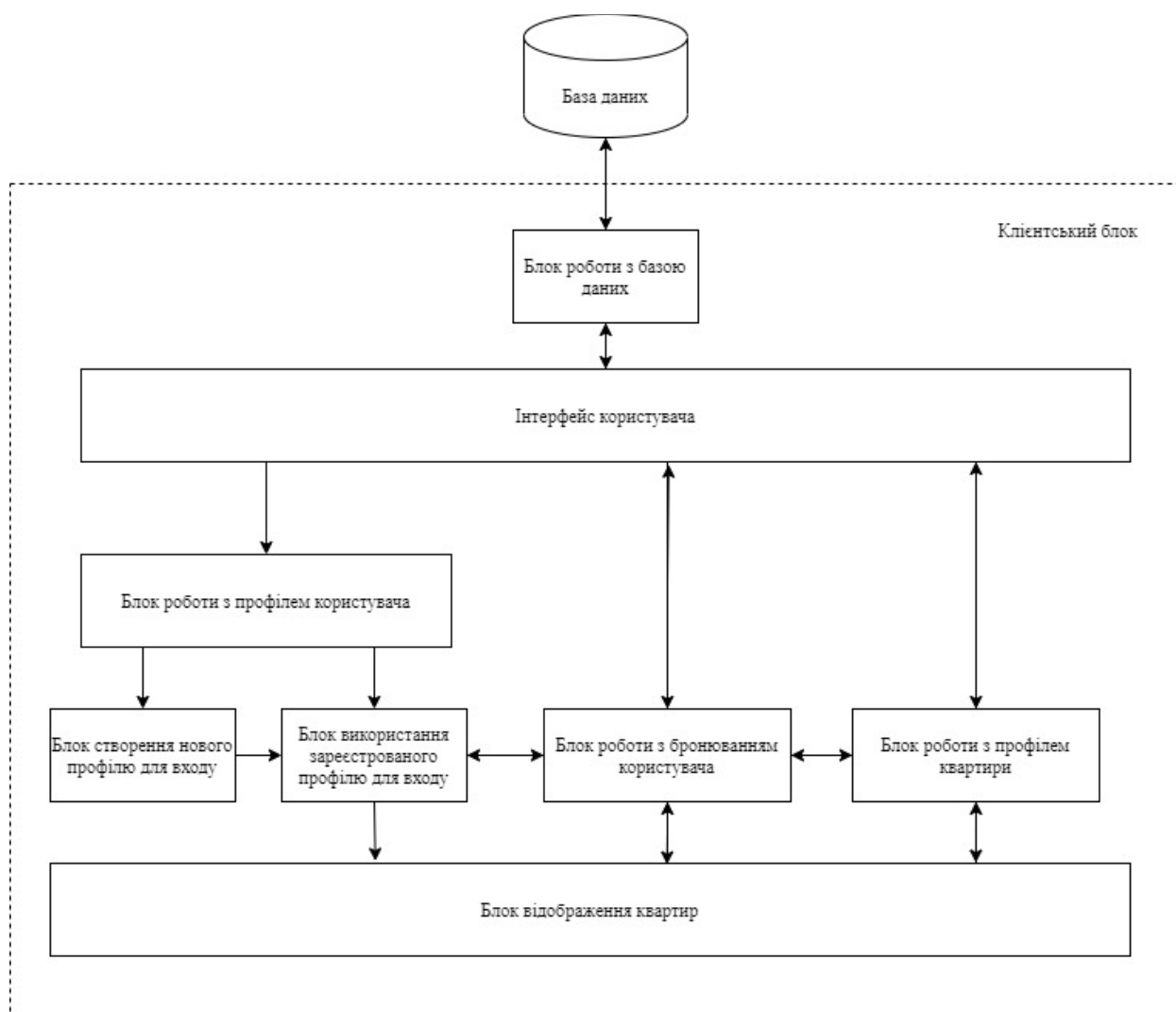


Рис. 2.2. Структурна схема клієнтського блоку програмного модуля

Загальна логічна послідовність виконання функцій програмного модуля сервісу бронювання квартир наведена на діаграмі прецедентів (рис. 2.3).

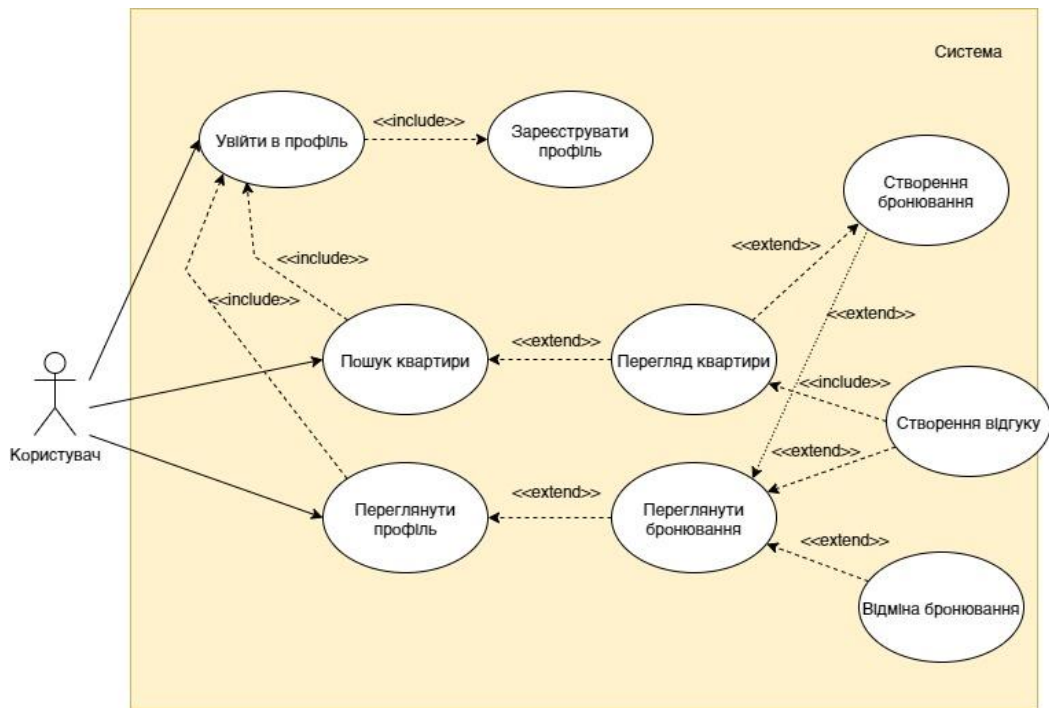


Рис.2.3. Діаграма прецедентів програмного модуля

Діаграма прецедентів починається з того, що користувач повинен увійти в сервіс використовуючи один з двох варіантів використання, а саме здійснити вхід в програмний модуль за допомогою зареєстрованого профілю або за допомогою реєстрації нового профілю користувача.

У випадку обрання прецеденту входу в систему за допомогою реєстрації нового профілю користувачу стають доступні такі розширення як: пошук та перегляд квартир, перегляд профіля з можливістю відслідковувати бронювання, робити бронювання, робити відміну бронювання та створення відгуку.

У випадку обрання прецеденту перегляду квартири користувач має можливість побачити всю інформацію стосовно даної квартири та зробити бронювання, у разі бронювання квартири дані передаються в прецедент перегляд бронювання, де користувач може відслідковувати активні бронювання.

У випадку обрання прецеденту перегляду профіля користувач отримує доступ до перегляду активних бронювань, при роботі з цими бронюваннями він має можливість здійснити відміну, і якщо дата закінчення бронювання квартири закінчилася, тоді користувачу стає активним залишення відгуку.

2.3. Інформаційне забезпечення

Відповідно до вимог програмного забезпечення побудовано функціональну схему (рис. 2.4).

Відповідно до функціональної схеми програмного модуля кожен блок має свої вхідні та вихідні параметри.

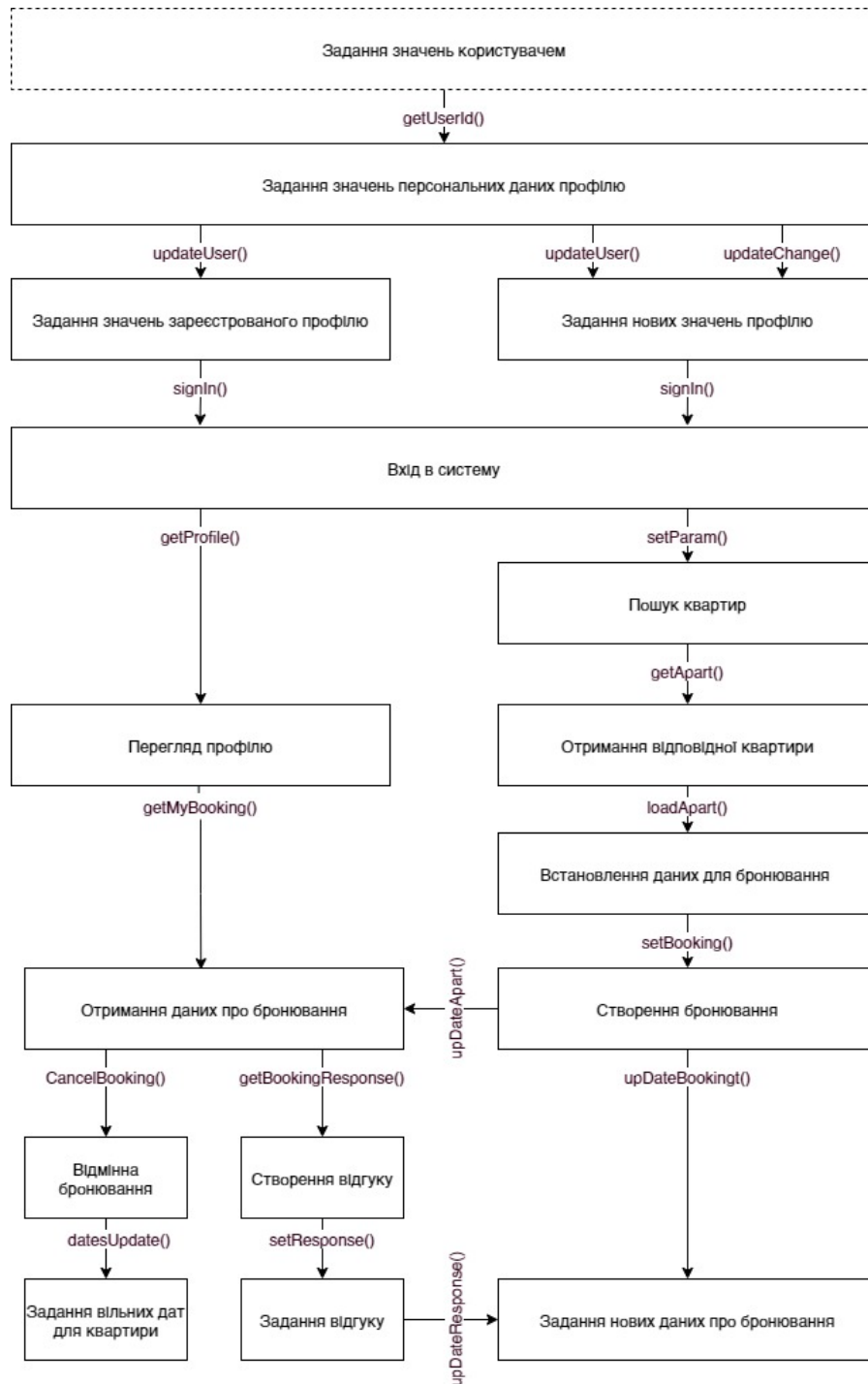


Рис. 2.4. Функціональна схема програмного модуля

1) Блок задання значень персональних даних профілю.

Вхідними даними блоку є функція *getUserId()*, за допомогою якого система отримує приватне значення ключа користувача. В ході роботи даного блоку відбувається подія, яка перевіряє співпадіння вхідних даних користувача з тими даними, що є в базі даних. Після проходження цієї перевірки блок вирішує куди відправити користувача до блоку задання зареєстрованого профілю, або до блоку задання нових значень профілю.

2) Блок задання зареєстрованих персональних даних користувача.

Вхідними даними блоку є функція *updateUser()*, дана функція виконується лише тоді, коли в системі існує даний користувач.

3) Блок задання нових значень профілю.

Вхідними даними блоку є функція *updateChange()* та *updateUser()*, завдяки цим функціям генерується та зберігається інформація. Всередині блоку відбувається процес по генерації унікального ключа для користувача та оновлення інформації в базі даних.

Результат виконання перелічених блоків є успішна автентифікація користувача в системі. По закінченню роботи блоків користувач має можливість отримати доступ до функцій сервісу.

4) Блок вхід в систему.

Вхідними даними представленого блоку є функція *signIn()*, яка надає можливість користувачу отримати доступ до головних функцій сервісу, а саме: блок пошук квартир, завдяки якому користувач знаходить для себе квартиру там може зробити її бронювання; блок перегляд профілю, завдяки якому користувач має можливість отримувати дані про свої бронювання та працювати з ними.

5) Блок пошуку квартир.

Вхідними даними представленого блоку є функція *setParam()*, яка надає можливість користувачу отримати повний список всіх квартир в системі, функція передає значення – картинка, назва, місто, та вартість за ніч та ідентифікатор квартири. В ході виконання блоку виконується функція *getApart()*.

6) Блок отримання відповідної квартири

Вхідними даними представленого блоку є функція *getApart()*, ця функція відправляє ідентифікатор квартири, після чого відкривається нова сторінка сервісу, яка надає можливість користувачу детально переглянути наявну інформацію стосовно квартири.

7) Блок встановлення даних для бронювання

Вхідними даними представленого блоку є функція *loadApart()*, яка надає можливість користувачу переглянути вільні дати обраної квартири, обрати вільні дати.

8) Блок створення бронювання

Вхідними даними представленого блоку є функція *setBooking()*, отримує ідентифікатор квартири, дату початку та кінця бронювання, цей блок опрацьовує інформацію надану користувачем та передає її *upDateBooking()*.

9) Блок задання нових даних про бронювання

Вхідними даними представленого блоку є функція *upDateResponse()* і *upDateBooking()*, перша функція оновлює сторінку квартири додаючи туди відгук з інформацією про час бронювання; друга функція задає користувачу відповідне бронювання, в залежності від встановлених їм параметрів.

10) Блок перегляд профілю

Вхідними даними представленого блоку є функція *getProfile()*, яка надає можливість користувачу відкрити власний профіль та отримати перегляд бронювання.

11) Блок отримання даних про бронювання

Вхідними даними представленого блоку є функція *getMyBooking()* і *upDateApart()*, перша функція дає отримати список бронювання, які користувач робив за весь час; друга функція створює нове бронювання та додає його до списку бронювання користувача.

12) Блок відмінна бронювання

Вхідними даними представленого блоку є функція *CancelBooking()*, яка надає можливість користувачу відмінити бронювання.

Функція активується після натискання користувачем на кнопку «Відмінна бронювання».

13) Блок задання вільних дат для квартири

Вхідними даними представленого блоку є функція *datesUpdate()*, яка отримує на вхід дані про скасоване бронювання користувача та оновлює вільні дати для квартири.

14) Блок створення відгуку

Вхідними даними представленого блоку є функція *getBookingResponse()*, функція стає активним після проходження часу бронювання, цей блок надає можливість користувачу створити відгук стосовно своєї подорожі.

15) Блок задання відгуку

Вхідними даними представленого блоку є функція *setResponse()*, яка отримує на вхід дані, що ввів користувач в вікні відгука та передає їх до бази даних.

2.4. Висновки до розділу

1) До функціональних вимог програмного модуля сервісу бронювання квартир належать:

- можливість роботи програмного модуля з серверною базою даних для оперування інформацією користувачів;
- можливість реєстрації нових профілів за допомогою введених даних, таких як електронна пошта, ім'я та фамілія, а також пароль;
- можливість перегляду та пошуку квартир за допомогою назви, або переміщенням карти за допомогою комп'ютерної миші;
- можливість перегляду сторінки квартири з відповідною інформацією, такою як: фотографії, назва, опис, зручності (кондиціонер, сніданок та інші), відруги;
- можливість перегляду вільних дат відповідної квартири;
- можливість обирання дати для бронювання квартири;

- можливість відправлення запиту користувачем на бронювання в системі;
- можливість залишення відгуку на сторінці квартири;
- можливість відправлення і отримання повідомлень між користувачем та власником квартири;
- можливість відхилення бронювання квартири.

2) Розроблено структурну схему, що складається з двох частин – структурна схема серверного блоку та структурна схема клієнтського блоку.

До структурної схеми серверного блоку входять наступні компоненти:

- база даних, елементами якої є дані користувачів; дані квартири (власник, вартість за добу, відгуки, опис та інші), дані приватних повідомлень між орендаром та власником квартири;

- блок роботи з базою даних, за допомогою якого встановлюється зв'язок між серверною частиною програмного модуля та самою базою даних;

- блок оновлення даних користувача. В ході роботи даного блоку користувач при успішній реєстрації отримує ключ, завдяки якому може повторно по ньому входити в свій профіль та користуватися функціями сервісу;

- блок оновлення даних квартири. При бронюванні користувачами, для всіх інших користувачів квартири будуть змінені;

- блок оновлення даних бронювання. В ході операції дані користувача про його бронювання заносяться в базу даних;

- блок збереження особистих даних користувача, до яких належать відомості про електронні адреси, паролі та користувацькі ідентифікатори;

- блок збереження даних квартири, при яких в базі оновлюється вільні дати для бронювання;

- блок роботи користувача з бронюванням, при якому користувач отримує дані про вільні дати для обраної квартири, обирає необхідні йому дати.

До структурної схеми клієнтського блоку входять наступні компоненти:

- база даних, елементами якої є дані користувачів; дані квартири (власник, вартість за добу, відгуки, опис та інші), дані приватних повідомлень між орендаром та власником квартири;

- блок роботи з базою даних, за допомогою якого встановлюється зв'язок між серверною частиною програмного модуля та самою базою даних;
- інтерфейс користувача, який виконує функцію передачі інформації між користувачем та блоком роботи з базою даних;
- блок роботи з профілем користувача, що надає можливість програмному модулю авторизувати користувача. Цей блок в свою чергу поділяється на блок створення нового профілю для входу та блок використання зареєстрованого профілю для входу;
- блок роботи з бронюванням користувача, що надає можливість користувачу робити бронювання обраної квартири з відповідними обраними датами, а також передавати їх в базу для подальшого оновлення в системі;
- блок роботи з профілем квартири, завдяки якому користувач має можливість побачити вільні дати, ціну за ніч та відповідні характеристики обраної квартири;
- блок відображення квартири, який функціонує через блок роботи з профілем квартири. Цей блок збирає усі можливі квартири в базі даних та надає їх користувачу.

3) Відповідно до вимог програмного забезпечення побудовано функціональну схему:

- блок задання значень персональних даних профілю;
- блок задання зареєстрованих персональних даних користувача;
- блок задання нових значень профілю;
- блок вхід в систему;
- блок пошуку квартир;
- блок отримання відповідної квартири;
- блок створення бронювання;
- блок перегляд профілю;
- блок отримання даних про бронювання;
- блок відмінна бронювання;
- блок створення відгуку.

РОЗДІЛ 3

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ СЕРВІСУ БРОНЮВАННЯ КВАРТИР

3.1. Системні вимоги

Програмний сервіс бронювання квартир представляється у вигляді інтернет сайту. Для роботи з сервісом необхідний комп'ютер з такими мінімальними системними вимогами:

- ОС *Windows XP* і вище;
- 32-розрядний (*x86*) або 64-розрядний (*x64*) процесор від 1.5 ГГц;
- 2 гігабайти (ГБ) оперативної пам'яті (*RAM*);
- від 100 мегабайт (*МБ*) місця на диску;
- доступ до Інтернету (для оновлення каталогу).

3.2. Алгоритми реалізації методів роботи модуля

Відповідно до поставлених завдань проектування програмного модуля реалізовано:

- 1) Автентифікацію користувачів в системі.

Функціонал:

- користувачі можуть зареєструватися або увійти з створеним попередньо обліковим записом;
- обробка неправильних даних при вході або реєстрація нового профілю;
- користувачі, які увійшли мають можливість отримати доступ до головних функцій програмного модуля;

Кафедра КСУ				НАУ 21 10 32 000 ПЗ			
<i>Виконав</i>	Слосаренко О.К.			ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ СЕРВІСУ БРОНЮВАННЯ КВАРТИР	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	Нечипорук В.В.				Д	32	57
<i>Консульт.</i>					123 СП-437		
<i>Норм. контр.</i>	Тупота С. В.						
<i>Голова комісії.</i>	Литвиненко О. Є.						

- користувачі, які не авторизовані можуть переглядати сторінку заставки;
- зареєстровані користувачі перенаправляються на сторінку пошуку.

За вміст виведеного бокового меню відповідає екземпляр класу *User*, код якого наведено нижче [9]:

```
class User < ApplicationRecord
  validates :password_digest, :fname, :lname, presence: true
  validates :email, :session_token, presence: true, uniqueness: true
  validates :password, length: {minimum: 6}, allow_nil: true
  attr_reader :password
  after_initialize :ensure_session_token
```

З форми автентифікацію зчитуються дані пошти, та пароль. Після чого йде перевірка на правильність написання, пароль повинен містити не менше 6 символів, а пошта повинна проходити регулярне вираження. Та виконується ініціалізація входу.

Перевірка на валідність пошти виконується наступна [10]:

```
validates_format_of :email, :with => /\A[^\@ ]+@[^\@\.]+[^\@\.]+|z/
//
def self.find_by_credentials(email, password)
  user = User.find_by(email: email)
  return user if user && user.is_password?(password)
  nil
end
def is_password?(password)
  BCrypt::Password.new(self.password_digest).is_password?(password)
end
def self.generate_session_token
  SecureRandom::urlsafe_base64
end
def password=(password)
  @password = password
```

```

    self.password_digest = BCrypt::Password.create(password)
end
def ensure_session_token
    self.session_token ||= User.generate_session_token
end
def reset_session_token!
    self.session_token = User.generate_session_token
    self.save!
    self.session_token
end
end

```

В наведеному вище програмному кодї йде перевірка на існування даного користувача в системі за його даними пошти та паролю. Якщо профіль відсутній в базі даних користувач про це інформується та йому надається можливість створити новий профіль в систем [11].

2) Реалізація списку квартир і пошуку.

Функціонал:

- списки відображаються відповідно до положення карти. Коли користувачі зменшують масштаб і рухаються, відображаються в списках ті квартири, які зараз знаходяться на мапі;
- користувачі можуть фільтрувати списки, вибираючи діапазон дат на сторінці заставки. Квартири, які не відповідають не відобразатимуться;
- кожен список містить короткий текстовий опис, картинку, інформацію про місцезнаходження, ціну.

Реалізація відображення в списку квартир в залежності від переміщення мапи реалізовано наступним чином:

```

class Listing < ApplicationRecord
  //...
  def self.in_bounds(bounds)
    self.where("lat < ?", bounds[:northEast][:lat])
  end
end

```

```

    .where("lat > ?", bounds[:southWest][:lat])
    .where("long > ?", bounds[:southWest][:lng])
    .where("long < ?", bounds[:northEast][:lng])
  end
  //...
end

```

Реалізація взаємодії з картою реалізована наступним чином:

```

class ListingMap extends React.Component {
  constructor(props) {
    super(props);
    this.searchParams =
new URLSearchParams(`${this.props.history.location.hash}`);
    const lat = parseFloat(this.searchParams.get('lat')) || 40.716880;
    const long = parseFloat(this.searchParams.get('long')) || -73.948810;
    const startDate = this.searchParams.get('checkin') || null;
    const endDate = this.searchParams.get('checkout') || null;
    this.center = {
      lat: lat,
      lng: long }
    this.mapOptions = {
      center: this.center,
      zoom: 8 }
    this.dates = {
      startDate: startDate,
      endDate: endDate }}
  componentDidMount() {
    this.map = new google.maps.Map(this.mapNode, this.mapOptions);
    this.MarkerManager = new MarkerManager(this.map);
    this.registerListeners();
    this.MarkerManager.updateMarkers(this.props.listings); }

```

```

registerListeners() {
  google.maps.event.addListener(this.map, 'idle', () => {
    const directions = this.map.getBounds().toJSON();
    const bounds = {
      northEast: { lat: directions.north, lng: directions.east },
      southWest: { lat: directions.south, lng: directions.west }}
    this.props.updateFilter({
      "bounds": bounds,
      "dates": this.dates
    }))))}

```

3) Бронювання.

Функціонал:

- користувач може вибрати діапазон дат для бронювання. Якщо дати вже зайняти, тоді користувачу не буде дозволено обирати ці дати;
- користувач має можливість отримати доступ до власних бронювань у розділі «Профіль»;
- користувач має можливість скасувати бронювання, якщо дата початку ще не настала;
- користувач має можливість написати відгук про бронювання, якщо дата завершення вже минула.

Реалізація обрання користувачем діапазону дат, які вільні для обраної квартири представлено нижче [12]:

```

class Listing < ApplicationRecord
  //...
  def self.available_in_bounds(bounds, dates)
    overlap = Booking.select('listing_id')
      .where('? BETWEEN start_date AND end_date
              OR ? BETWEEN start_date AND end_date',
              dates[:startDate], dates[:endDate])
    too_long = Booking.select('listing_id')

```

```

        .where('? < start_date AND ? > end_date',
              Date.parse(dates[:startDate]), Date.parse(dates[:endDate]))
        self.in_bounds(bounds).where.not(id: overlap).where.not(id: too_long)
    end
end
end

```

Для відображення реалізації кроків бронювання квартири користувачем в системі розроблена відповідна схема алгоритму (рис. 3.1).

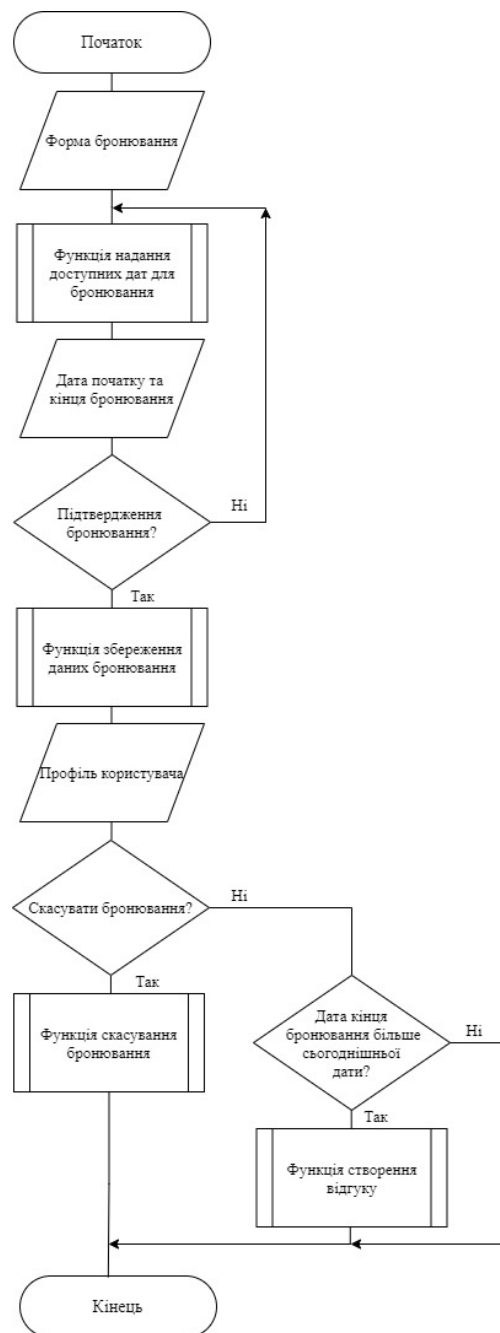


Рис. 3.1. Схема алгоритму бронювання квартири

Реалізація відміни бронювання реалізовано наступним чином:

```
class BookingIndexItem extends React.Component {
  // ...
  render() {
    const cancelButton = moment(booking.start_date).isAfter(moment(), 'day') ? (
      <button onClick={() => this.cancelBooking(booking.id)}
        className='cancel-booking-button'>
          Cancel Booking
        </button> ) : ( null )
  }
}
```

В наведеному програмного коді при натисканні на кнопку «Відмінити бронювання» виконується отримання ідентифікатору бронювання, дату початку та кінця, після чого бронювання стає неактивним, і кнопка також.

Реалізація залишення відгуку стосовно бронювання реалізовано наступним чином:

```
const reviewButton = moment(booking.end_date).isBefore(moment(), 'day') ? (
  <button onClick={() => dispatch(openReviewModal('create-review',
    booking.listing.id))} className='review-button'>
    Write Review
  </button>
) : ( null) }
```

В наведеному програмного коді при натисканні на кнопку «Залишити відгук» виконується відкриття вікна форми, де користувач може поділитися своїми враженнями.

4) Відгуки.

Функціонал:

- користувачі можуть залишити відгук про бронювання, а також надають оцінки у вигляді оцінок від однієї до п'яти;
- користувачі можуть оцінювати бронювання за шістьма категоріями, такі як: точність, комунікація, чистота, розташування, реєстрація та вартість;

– у списках відображається загальний середній рейтинг, який перераховується кожного разу, коли подається новий рейтинг;

– списки також відображають середній рейтинг для кожної з шести категорій. Рейтинг кожної категорії також перераховується кожного разу, коли подається новий рейтинг.

Для відображення реалізації кроків залишення відгуку користувача в системі розроблена відповідна схема алгоритму (рис. 3.2).



Рис. 3.2. Схема алгоритму залишення відгуку

Реалізація обробки встановлених параметрів на формі відгуку представлена нижче [13]:

```
class ReviewIndex extends React.Component {  
  findAvgRating() {  
    const { reviews } = this.props;  
    const ratings = {  
      accuracy: 0,  
      communication: 0,  
      cleanliness: 0,  
      location: 0,  
      check_in: 0,  
      value: 0  
    }  
    for (let i = 0; i < reviews.length; i++) {  
      let review = reviews[i];  
      for (let key in ratings) {  
        ratings[key] += review[key]} }  
    for (let key in ratings) {  
      ratings[key] = (ratings[key] / reviews.length) }  
    return ratings;  
  }  
}
```

В даному програмному коді відбувається обробка параметрів відгуку користувача. Для кожної оцінки присвоюється ключ, після чого, завдяки циклу, сервіс зберігає відповідну оцінку та формує середній рейтинг завдяки $ratings[key] = (ratings[key] / reviews.length)$

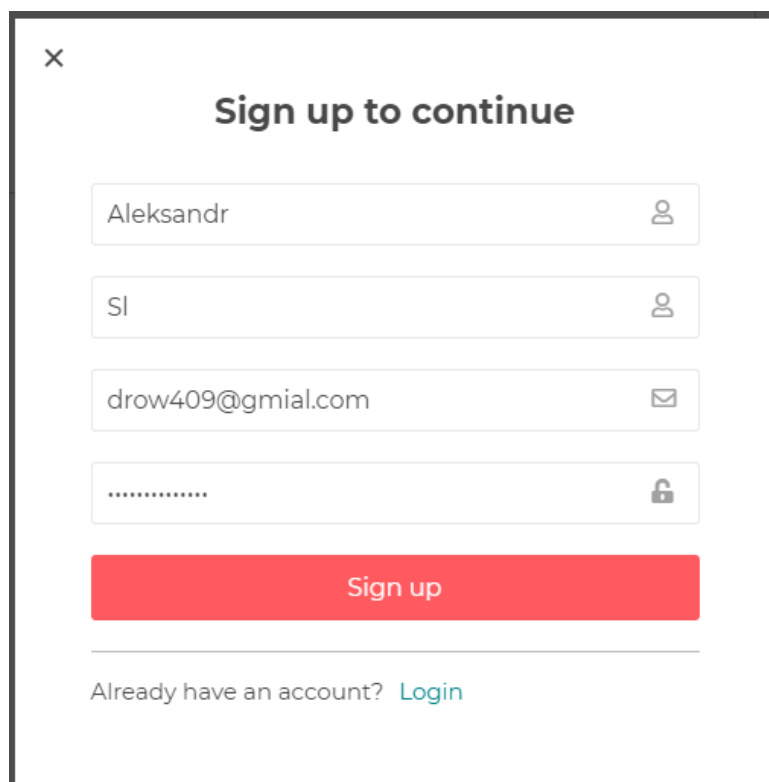
3.3. Інструкція користувача програмного модуля

Програмний модуль сервісу бронювання квартир представляється у вигляді інтернет сайту. Для запуску необхідно комп'ютер з мінімальною ОС *Windows XP* і вище та доступом до Інтернету.

При відкритті сайту користувач потрапляє до вікна входу в сервіс. На цьому вікні користувач програмного модуля повинен обрати один з запропонованих варіантів входу, який дозволить йому почати роботу з головними функціями додатку.

Користувач може або увійти в систему за уже дійсним профілем, або створити новий.

Для того, щоб зареєструвати аккаунт нового користувача користувач повинен заповнити своє ім'я, прізвище, пошту та пароль за допомогою якого відбуваються усі подальші процеси входження користувача на свій профіль додатку (рис. 3.3).



The image shows a registration form titled "Sign up to continue". It contains four input fields: a name field with "Aleksandr", a last name field with "SI", an email field with "drow409@gmial.com", and a password field with masked characters. Each field has a small person icon to its right. Below the fields is a red "Sign up" button. At the bottom, there is a link: "Already have an account? [Login](#)".

Рис. 3.3. Реєстрація нового користувача

Після користувач потрапляє на головне вікно сайту, де розташована форма для заповнення, в яку потрібно ввести параметри для пошуку квартири (рис.3.4).

Подорожувати як смисл ЖИТТЯ

Знайди квартиру на будь-який смак

куди?

Київ

CHECK-IN: 07/01/2021 CHECKOUT: 06/15/2021

June 2021

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Рис. 3.4. Форма пошуку квартири

Після натискання пошук кнопки «Пошук» користувач отримує список всіх доступних квартир за його параметром (рис. 3.5).

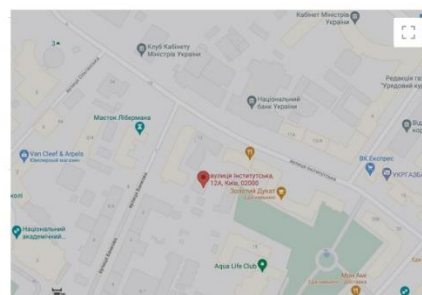
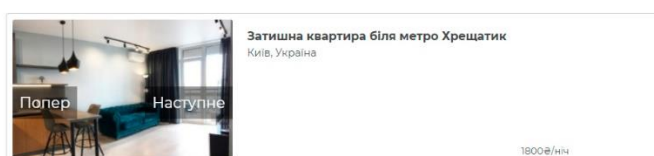


Рис. 3.5. Знайдені квартири за введеними характеристиками

Після натискання на список, відкривається вікно квартири, в якому користувач має можливість побачити більше картинок квартири, прочитати опис та які зручності є доступні в даній квартирі (рис.3.6).

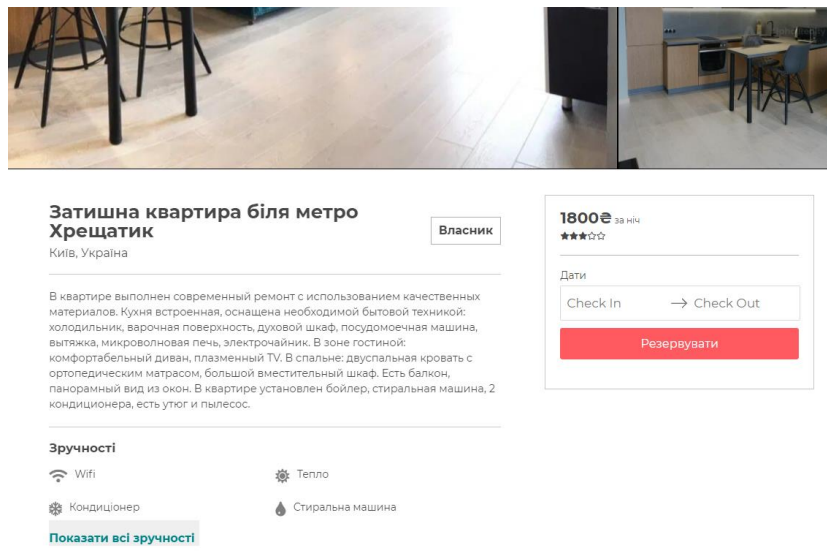


Рис. 3.6. Вікно опису квартири

Якщо прокрутити сторінку вниз, користувач зможе побачити календар квартири, в якому зарезервовані дати, коли квартира вже орендована помічена кольором. Також після календаря йде відгуки, де можна побачити не тільки середню оцінку, а за кожну характеристику (рис. 3.7).

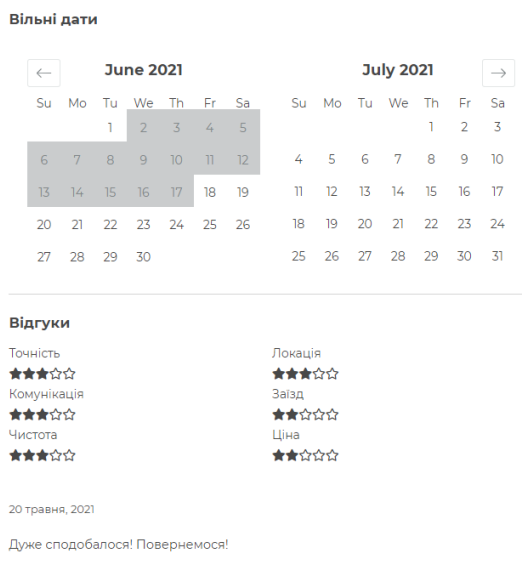


Рис. 3.7. Вільні дати і відгуки

Для бронювання користувачу потрібно обрати дати в календарі, та натиснути «Резервувати» (рис. 3.8).

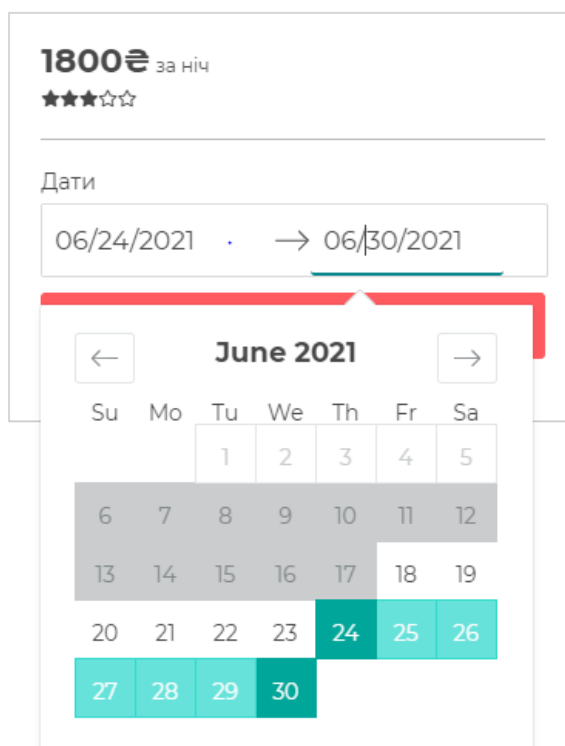



Рис. 3.8. Вибір дат для резервування квартири

Одразу відкриється вікно профіля, де користувач може побачити свої власні бронювання (рис. 3.9).

Майбутні поїздки

У вас найближчі поїздки. Керуйте та переглядайте свої бронювання тут.

24 червня 2021 - 30 червня 2021 Відмінити бронювання



Затишна квартира біля метро Хрещатик
Київ, Україна

10800€

Рис. 3.9. Майбутні поїздки

Після поїздки користувач може, по бажанню, залишити відгук. Для цього йому потрібно зайти в профіль, і справа зверху карточки поїздки буде кнопка «Відгук». При натисканні на яку відкриється вікно (рис. 3.10), в якому користувач може оцінити характеристики квартири та надати розвернутий відгук.

Точність	★★★★★	Локація	★★★★☆
Комунікація	★★★★☆	Заїзд	★★★★☆
Чистота	★★★★☆	Ціна	★★★★☆

Розповідь як вам ця поїздка?

Submit

Рис. 3.10 Форма відгуку

3.4. Тестування роботи програмного модуля

Для тестування програмного модуля сервісу бронювання квартир використовується ручне тестування. В ньому застосовуються властивості програмного модуля для знаходження помилок у роботі. Процес та результати тестування представлені у вигляді тестових випадків.

Суть тестового випадку полягає в створенні та проходженні спеціального алгоритму тестування програми. Тестовий випадок складається з кроків, умов та параметрів, завдяки яким перевіряється можливість виникнення невизначеної

ситуації в роботі програмного модуля. Кожен тестовий має містити наступні складові:

- визначення мети тестового випадку;
- певна умова, що передує виконанню основних дій випадку, яка повинна привести програмний модуль в необхідний стан, для початку перевірки;
- кроки тестування, які модулюють використання програмного модуля людиною;
- опис очікуваних наслідків, тобто що повинно відбутися шляхом даних кроків;
- статус випадку, який підтверджує або заперечує відповідність очікуваного результату до фактичного.

Тестування роботи програмного модуля проводиться над його певними можливостями, а саме: повторне бронювання квартири, бронювання квартири на некоректні дати, створення відгуку.

3.4.1. Тестування повторне бронювання квартири

Для тестування на помилки можливості бронювання квартири на коректні дані користувач введе дату прибуття та виїзду на доступні дані, після бронювання інший користувач спробує забронювати на таку саму дату обрану квартиру. Результат даного тестового випадку є відсутність для другого користувача вибору повторно дати (табл. 3.1).

Висновком даного тестового випадку (табл. 3.1) є те, що бронювання успішно зберігається в базі даних, і ці дати неможливо забронювати повторно.

Тестування повторне бронювання квартири

Дія	Очікуваний результат	Фактичний результат
Передумова		
Відкрити програмний модуль та виконати вхід в профіль на першому користувачі	Відбувається перехід до головного вікна програми	Пройдено
Відкрити програмний модуль та виконати вхід в профіль на другому користувачі	Відбувається перехід до головного вікна програми	Пройдено
Кроки тесту		
Обрати квартиру для тестування та відкрити її вікно	Вікно відкрилося, дані квартири згружені	Пройдено
Відкрити форму для бронювання	Форма відкрилася	Пройдено
Обрати дати для бронювання 20.06.2021	Дати успішно обрані	Пройдено
Натиснути кнопку «Забронювати»	Квартира заброньована, відкрилося вікно профіля з майбутніми подорожами	Пройдено
Перейти на профіль другого користувача	Вхід виконано	Пройдено
Обрати ту саму квартиру та відкрити її вікно бронювання	Вікно відкрилося, дані квартири згружені	Пройдено
Обрати дати для бронювання 20.06.2021	Дати неможливо обрати, вони вже заброньовані	Пройдено

Постумова		
Завершити роботу програмного модуля на першому користувачі	Робота програмного модуля завершена	Пройдено
Завершити роботу програмного модуля на другому користувачі	Робота програмного модуля завершена	Пройдено

3.4.2. Бронювання квартири на некоректні дати

Для тестування на помилки можливості бронювання квартири на некоректні дані користувач забронює квартиру на дату, яка пройшла; забронює квартиру без вводу дати заїзду та дати виїзду (табл. 3.2).

Таблиця 3.2

Тестування бронювання квартири на некоректні дати

Дія	Очікуваний результат	Фактичний результат
Передумова		
Відкрити програмний модуль та виконати вхід в профіль	Відбувається перехід до головного вікна програми	Пройдено
Кроки тесту		
Обрати квартиру для тестування та відкрити її вікно	Вікно відкрилося, дані квартири згружені	Пройдено
Відкрити форму для бронювання	Форма відкрилася	Пройдено

Обрати дати для бронювання 04.07.2020	Дати не обрані	Пройдено
Не заповнюючи дані натиснути кнопку «Забронювати»	Дати не обрані	Пройдено
Постумова		
Завершити роботу програмного модуля	Робота програмного модуля завершена	Пройдено

Висновком даного тестового випадку (табл. 3.2) є те, що бронювання працює таким чином, що обробляє виключення і не дає користувачу ввести негативні значення.

3.4.3. Тестування створення відгуку

Для тестування на помилки можливості сервісу створення відгуку для бронювання, яке вже закінчилося. В ході тестування користувач спробує залишити відгук для бронювання, яке ще не закінчилося; залишити пустий відгук.

Таблиця 3.3

Тестування створення відгуку

Дія	Очікуваний результат	Фактичний результат
Передумова		
Відкрити програмний модуль та виконати вхід в профіль	Відбувається перехід до головного вікна програми	Пройдено

Кроки тесту		
Відкрити профіль користувача	Сторінка профілю відкрилася, для користувача доступний список його бронювань	Пройдено
Натиснути на кнопку «Відгук»	Кнопка не активна, вона знаходиться в стані invisible	Пройдено
Обрати бронювання, яке все закінчилося	Бронювання обране	Пройдено
Натиснути на кнопку «Відгук»	Відкривається форма створення відгуку	Пройдено
Залишивши пусті поля натисніть « <i>Submit</i> »	Форма не виконає операції та буде чекати поки користувач щось не введе	Пройдено
Постумова		
Завершити роботу програмного модуля	Робота програмного модуля завершена	Пройдено

Результат даного тестового випадку є відсутність можливості створення відгуку для бронювання, яке не закінчилося; неможливість збереження пустого відгуку. (табл. 3.3).

3.5. Висновки до розділу

1) Визначено мінімальні системні вимоги програмного модуля сервісу бронювання квартир. До них належать:

- ОС *Windows XP* і вище;
- 32-розрядний (*x86*) або 64-розрядний (*x64*) процесор від 1.5 ГГц;

- 2 гігабайти (ГБ) оперативної пам'яті (RAM);
- від 100 мегабайт (МБ) місця на диску;
- доступ до Інтернету (для оновлення каталогу).

2) Програмно реалізовано алгоритми:

– алгоритм автентифікацію користувачів в системі, з таким функціоналом: реєстрація або вхід користувачів; обробка неправильних даних;

– алгоритм реалізація списку квартир і пошуку, з таким функціоналом: відображення списку в залежності від положення карти; фільтрація списків;

– алгоритм бронювання, з таким функціоналом: користувач може вибрати діапазон дат для бронювання; користувач має можливість отримати доступ до своїй бронювань, користувач має можливість написати відгук про бронювання;

– алгоритм відгуків, з таким функціоналом: користувачі можуть писати відгуки про минулі бронювання, а також надають оцінки у вигляді зірочок – від однієї до п'яти; відображення середнього рейтингу; оновлення рейтингу.

3) Описано інтерфейс користувача, який здійснює передачу інформації між користувачем та компонентами програмного модуля сервісу бронювання квартир.

4) Проведено тестування програмного модуля. Було використано ручне тестування, в якому застосовувалися властивості програмного модуля для знаходження помилок у роботі. Результати тестування представлені у вигляді позитивних тестових випадків.

За допомогою тест кейсів перевірені наступні можливості програми:

- повторне бронювання квартири;
- бронювання квартири на некоректні дати;
- створення відгуку.

ВИСНОВКИ

1) Після проведення аналізу існуючих програмних рішень сформовано вимоги до програмного модуля, які забезпечать його універсальність:

– мати сучасні програмні рішення та методи створення сервісів бронювання квартир;

– мати можливість роботи програмного модуля з серверною базою даних для оперування інформацією користувачів;

– мати структурі схеми програмного модуля: клієнтську частину та серверну;

– мати алгоритм створення профілю користувача, з можливістю перегляду заброньованих подорожей;

– мати алгоритм бронювання відповідних апартаментів за відповідними датами;

– мати в програмному модулі функції залишення відгуку, відміни бронювання та приватні повідомлення між орендатором та власником квартири.

2) Проаналізовано методи створення програмного модуля сервісу бронювання квартир та обрано мову програмування JavaScript, та інтегроване середовище розробки програмного забезпечення WebStorm.

3) До функціональних вимог програмного модуля сервісу бронювання квартир належать:

– можливість роботи програмного модуля з серверною базою даних для оперування інформацією користувачів;

– можливість реєстрації нових профілів за допомогою введених даних, таких як електронна пошта, ім'я та фамілія, а також пароль;

– можливість перегляду та пошуку квартир за допомогою назви, або переміщенням карти за допомогою комп'ютерної миші;

– можливість перегляду сторінки квартири з відповідною інформацією, такою як: фотографії, назва, опис, зручності (кондиціонер, сніданок та інші), відруги.

– можливість перегляду вільних дат відповідної квартири;

– можливість обирання дати для бронювання квартири;

– можливість відправлення запиту користувачем на бронювання в системі;

– можливість залишення відгуку на сторінці квартири;

– можливість відправлення і отримання повідомлень між користувачем та власником квартири;

– можливість відхилення бронювання квартири.

4) Розроблено структурну схему, що складається з двох частин – структурна схема серверного блоку та структурна схема клієнтського блоку.

До структурної схеми серверного блоку входять наступні компоненти:

– база даних, елементами якої є дані користувачів; дані квартири (власник, вартість за добу, відгуки, опис та інші), дані приватних повідомлень між орендаром та власником квартири;

– блок роботи з базою даних, за допомогою якого встановлюється зв'язок між серверною частиною програмного модуля та самою базою даних;

– блок оновлення даних користувача. В ході роботи даного блоку користувач при успішній реєстрації отримує ключ, завдяки якому може повторно по ньому входити в свій профіль та користуватися функціями сервісу;

– блок оновлення даних квартири. При бронюванні користувачами, для всіх інших користувачів квартири будуть змінені;

– блок оновлення даних бронювання. В ході операції дані користувача про його бронювання заносяться в базу даних;

– блок збереження особистих даних користувача, до яких належать відомості про електронні адреси, паролі та користувацькі ідентифікатори;

– блок збереження даних квартири, при яких в базі оновлюється вільні дати для бронювання;

– блок роботи користувача з бронюванням, при якому користувач отримує дані про вільні дати для обраної квартири, обирає необхідні йому дати.

До структурної схеми клієнтського блоку входять наступні компоненти:

– база даних, елементами якої є дані користувачів; дані квартири (власник, вартість за добу, відгуки, опис та інші), дані приватних повідомлень між орендаром та власником квартири;

– блок роботи з базою даних, за допомогою якого встановлюється зв'язок між серверною частиною програмного модуля та самою базою даних;

– інтерфейс користувача, який виконує функцію передачі інформації між користувачем та блоком роботи з базою даних;

– блок роботи з профілем користувача, що надає можливість програмному модулю авторизувати користувача. Цей блок в свою чергу поділяється на блок створення нового профілю для входу та блок використання зареєстрованого профілю для входу;

– блок роботи з бронюванням користувача, що надає можливість користувачу робити бронювання обраної квартири з відповідними обраними датами, а також передавати їх в базу для подальшого оновлення в системі;

– блок роботи з профілем квартири, завдяки якому користувач має можливість побачити вільні дати, ціну за ніч та відповідні характеристики обраної квартири;

– блок відображення квартири, який функціонує через блок роботи з профілем квартири. Цей блок збирає усі можливі квартири в базі даних та надає їх користувачу;

5) Відповідно до вимог програмного забезпечення побудовано функціональну схему.

6) Визначено мінімальні системні вимоги програмного модуля сервісу бронювання квартир.

7) Програмно реалізовано алгоритми:

– алгоритм автентифікацію користувачів в системі, з таким функціоналом: реєстрація або вхід користувачів; обробка неправильних даних;

– алгоритм реалізація списку квартир і пошуку, з таким функціоналом: відображення списку в залежності від положення карти; фільтрація списків;

– алгоритм бронювання, з таким функціоналом: користувач може вибрати діапазон дат для бронювання; користувач має можливість отримати доступ до своєї бронювань, користувач має можливість написати відгук про бронювання;

– алгоритм відгуків, з таким функціоналом: користувачі можуть писати відгуки про минулі бронювання, а також надають оцінки у вигляді зірочок – від однієї до п'яти; відображення середнього рейтингу; оновлення рейтингу.

8) Описано інтерфейс користувача, який здійснює передачу інформації між користувачем та компонентами програмного модуля сервісу бронювання квартир.

9) Проведено тестування програмного модуля. Було використано ручне тестування, в якому застосовувалися властивості програмного модуля для знаходження помилок у роботі. Результати тестування представлені у вигляді позитивних тестових випадків.

За допомогою тест кейсів перевірені наступні можливості програми:

- повторне бронювання квартири;
- бронювання квартири на некоректні дати;
- створення відгуку.

10) Практична цінність проекту полягає в наданні можливості користувачу використовувати програмний модуль під час пошуку квартир для подорожей, а також для оренди квартир на довгостроковий час.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1) Роглев Х.Й. «Основы готельного менеджменту» – Київ: Пролісок – 2010. – 300 с.
- 2) Маркетинг туризма : учебное пособие / И.В. Гончарова, Т.П. Розанова, М.А. Морозов, Н.С. Морозова. – М. : Федеральное агентство по туризму – 2014. – 224 с.
- 3) Лисяков Ю. Современные информационные технологии. // Гостиничный и ресторанный бизнес. – 2005.
- 4) Браймер Р.А. Основы управління в індустрії гостинності: – М.: Аспект Пресс. – 2001. – 382 с.
- 5) Джанджугазова Є.А. Маркетинг в індустрії гостинності. М.: Видавничий центр "Академія". – 2003.
- 6) Орлик О. В. Можливості Інтернету у формуванні, просуванні й реалізації продуктів та послуг / О. В. Орлик, К. Ф. Дем'янчук // Інформатика та інформаційні технології. – Одеса: ОНЕУ. – 2015. – 91 с.
- 7) Шаховалов М.М., Інтернет-технології в туризмі: Навчальний посібник /М.М.. Шаховалов; АЛТГАКИ, кафедра інформатики. – Барнаул: Видво. – 2013. – 251 с
- 8) Морозов М.А., Морозова Н.С. Информационные технологии в социально-культурном сервисе и туризме. Оргтехника. – М.: Изд-во. центр "Академия". – 2002. – 240 с.
- 9) Лазаро Исси Коэн, Джозеф Исси Коэн. Полный справочник по HTML, CSS и JavaScript; – ЭКОМ Паблишерз - М., – 2016. – 311 с.
- 10) Херман Дэвид Сила JavaScript. 68 способов эффективного использования JS; – М., – 2015. – 952 с.
- 11) Кингсли-Хью К.Е. JavaScript 1.5: учебный курс / К.Э. Кингсли-Хью. – М., – 2013. – 272 с.
- 12) Крокфорд Д. JavaScript Сильные стороны; – М., – 2016. – 262 с.

13) Макфарланд Дэвид JavaScript. Подробное руководство; – Эксмо: М., – 2015. – 608 с.

14) ДСТУ 3008-95 Документація. Звіти у сфері науки і техніки. Структура і правила оформлення.

15) Бойченко С.В., Іванченко О.В. Положення про дипломні роботи (проекти) випускників Національного авіаційного університету. – К.: НАУ, 2017. – 63 с.

ДОДАТОК А

Лістинг коду форми бронювання програмного модуля

```
class BookingForm extends React.Component {
  constructor(props) {
    super(props);
    this.state = props.booking; // {startDate: null, endDate: null}
    this.handleSubmit = this.handleSubmit.bind(this);
    // this.handleChange = this.handleChange.bind(this);
    this.isBlocked = this.isBlocked.bind(this);
  }
  isBlocked(day1) {
    return this.props.bookedDates.some(day2 => {
      return isSameDay(day1, day2);
    })
  }
  handleSubmit(e) {
    const { userId, booking, formType, createBooking, receiveBooking,
openLoginModal } = this.props;
    const startDate = moment(this.state.startDate).format("YYYY/MM/DD");
    const endDate = moment(this.state.endDate).format("YYYY/MM/DD");
    e.preventDefault();
    if (userId) {
      createBooking({
        start_date: startDate,
        end_date: endDate, // not showing up in controller?
        guest_id: userId,
        listing_id: this.props.listing.id
      });
      this.props.history.push('/bookings');
    }
  }
}
```

```

    } else {
      openLoginModal();
    }
  }
  renderStars(num) {
    const stars = [];
    for (let i = 0; i < 5; i++) {
      if (i < num) {
        stars.push(<i key={i} className="fas fa-star"></i>)
      } else {
        stars.push(<i key={i} className="far fa-star"></i>)
      }
    }
    return stars;
  }
  render() {
    return (
      <div className='booking-form-container'>
        <div className='booking-form-info'>
          <div className='price-info'>
            <p className='booking-price'>${this.props.listing.price}</p>
            <p className='per-night'>per night</p>
          </div>
          <p className='booking-stars'>{this.renderStars(this.props.avgRating)}</p>
        </div>
        <h1 className='booking-form-dates-title'>Dates</h1>
        <div className='booking-form-date-picker'>
          <DateRangePicker
            startDate={this.state.startDate} // momentPropTypes.momentObj or null,
            startDateId="your_unique_start_date_id" // PropTypes.string.isRequired,

```

```

    endDate={this.state.endDate} // momentPropTypes.momentObj or null,
    endDateId="your_unique_end_date_id" // PropTypes.string.isRequired,
    onDatesChange={({ startDate, endDate }) =>
      this.setState({ startDate, endDate })
    } // PropTypes.func.isRequired,
    focusedInput={this.state.focusedInput} // PropTypes.oneOf([START_DATE,
    END_DATE]) or null,
    onFocusChange={focusedInput => this.setState({ focusedInput })} //
    PropTypes.func.isRequired,
    numberOfMonths={1}
    startDatePlaceholderText="Check In"
    endDatePlaceholderText="Check Out"
    isDayBlocked={this.isBlocked}
  />
</div>
<button onClick={this.handleSubmit} className='reservation-button'>
  Reserve
</button>
</div>
)
}
}
export const RECEIVE_ALL_BOOKINGS = 'RECEIVE_ALL_BOOKINGS';
export const RECEIVE_BOOKING = 'RECEIVE_BOOKING';
export const REMOVE_BOOKING = 'REMOVE_BOOKING';
export const receiveAllBookings = bookings => {
  return {
    type: RECEIVE_ALL_BOOKINGS,
    bookings
  }
}

```

```

}
export const receiveBooking = booking => {
  return {
    type: RECEIVE_BOOKING,
    booking
  }
}
export const removeBooking = booking => {
  return {
    type: REMOVE_BOOKING,
    bookingId: booking.id
  }
}
export const fetchAllBookings = () => {
  return dispatch => {
    return BookingAPIUtil.fetchAllBookings()
      .then(bookings => {
        return dispatch(receiveAllBookings(bookings))
      })
  }
}
export const fetchBooking = id => {
  return dispatch => {
    return BookingAPIUtil.fetchBooking(id)
      .then(booking => {
        return dispatch(receiveBooking(booking))
      })
  }
}
export const createBooking = (booking) => {

```

```
return dispatch => {
  return BookingAPIUtil.createBooking(booking)
    .then(booking => {
      return dispatch(receiveBooking(booking))
    })
}
}

export const updateBooking = booking => {
  return dispatch => {
    return BookingAPIUtil.updateBooking(booking)
      .then(booking => {
        return dispatch(receiveBooking(booking))
      })
  }
}

export const deleteBooking = id => {
  return dispatch => {
    return BookingAPIUtil.deleteBooking(id)
      .then(booking => {
        return dispatch(removeBooking(booking))
      })
  }
}
}
```