

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра мережних технологій факультету інформатики

РОЗРОБКА ВЕБ-САЙТУ ДЛЯ ТУРИСТИЧНОГО АГЕНТСТВА

Текстова частина до курсової роботи
за спеціальністю “Інженерія програмного забезпечення” 121

Керівник курсової роботи
к.ф-м.н., с.в. Гречко А. В.

(підпис)

“ ____ ” _____ 2020 р.

Виконала студентка Ровніна Т. Д.

“ ____ ” _____ 2020 р.

Київ 2020

Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

Кафедра інформатики факультету інформатики

ЗАТВЕРДЖУЮ

Зав. кафедри мережних технологій,
доцент, к. тех. н.

_____ О. В. Франчук

“ ____ ” _____ 2019 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студентці 3-го курсу, факультету інформатики, Ровніній Тетяні Дмитрівні

ТЕМА: Розробка веб-сайту для туристичного агентства

Вихідні дані:

- адаптація інтерфейсу під типові розширення екрану;
- підтримка браузеру Google Chrome.

Зміст ТЧ до курсової роботи:

Індивідуальне завдання

Вступ

1. Аналіз предметної області. Постановка завдання курсової роботи

2. Теоретичні відомості

3. Опис реалізації програмного продукту

Висновки

Список використаних джерел

Додатки

Дата видачі “ ____ ” _____ 2019 р. Керівник _____

Завдання отримав _____

Тема: Розробка веб-сайту для туристичного агентства

Календарний план виконання роботи:

№ п/п	Назва етапу курсової роботи	Термін виконання етапу	Примітка
1.	Отримання завдання на курсову роботу.	02.10.2019	
2.	Огляд технічної літератури за темою роботи.	12.12.2019	
3.	Проектування схеми бази даних.	01.01.2020	
4.	Розробка бази даних.	30.01.2020	
5.	Розробка серверної частини веб-сайту.	15.03.2020	
6.	Розробка клієнтської частини веб-сайту.	01.04.2020	
7.	Написання текстової частини.	10.04.2020	
8.	Створення слайдів для доповіді та написання доповіді.	15.04.2020	
9.	Обговорення отриманих результатів з керівником.	20.04.2020	
11.	Остаточне оформлення текстової частини та слайдів.	09.05.2020	
12.	Захист курсової роботи.	15.05.2020	

Студент _____

Керівник _____

“ _____ ” _____

Зміст

Перелік умовних позначень.....	7
Анотація	8
Вступ	9
1. Аналіз предметної області. Постановка завдання курсової роботи ...	11
1.1. Аналіз сучасного стану питання та обґрунтування теми	11
1.2. Огляд існуючих аналогів розробки	11
1.3. Постановка завдання	15
2. Теоретичні відомості.....	17
2.1. Опис засобів розробки.....	17
2.1.1. JavaScript як мова програмування.....	17
2.1.2. Веб-сервер Node.js	17
2.1.3. Шаблонізатор Pug.....	18
2.1.4. Додаткові бібліотеки	19
2.1.4.1. Бібліотека JQuery.....	19
2.1.4.2. Набір інструментів Bootstrap.....	19
2.1.5. Середовище розробки WebStorm	19
2.2. Вибір СКБД.....	20
2.2.1. Переваги і недоліки SQLite.....	21
2.2.2. Переваги і недоліки MySQL	22
2.2.3. Переваги і недоліки PostgreSQL.....	22
Висновок з вибору СКБД	23
3. Опис реалізації програмного продукту.....	24
3.1. Аналіз технічного завдання	24
3.2. Архітектура системи.....	25

3.3. Реалізація бази даних	26
3.3.1. Розробка ER-моделі	26
3.3.2. Розробка реляційної моделі	28
3.4. Опис розробки серверної частини	33
3.5. Опис клієнтської частини	36
3.5.1. Опис функціоналу клієнтської частини.....	36
3.5.2. Опис інтерфейсу	36
3.6. Тестування програми і результати її виконання.....	39
Висновки	48
Список використаних джерел	49
Додаток А. Запити створення БД	51
Додаток Б. Код програми	56
Код з файлу server.js.....	56
Код з файлу client.js.....	66
Код з файлу style.css.....	71
Код з файлу index.pug	78
Код з файлу navbar.pug	80
Код з файлу scripts.pug.....	81
Код з файлу success.pug	81
Код з файлу tour-view.pug.....	81
Код з файлу tours.pug	82
Код з файлу comments.pug.....	83
Код з файлу contacts.pug	84
Код з файлу document.pug	85
Код з файлу filters.pug	86

Код з файлу footer.pug.....	87
Код з файлу header.pug.....	87
Код з файлу hotel-tours.pug	88
Код з файлу hotel-view.pug	88
Код з файлу hotels.pug.....	89

Перелік умовних позначень

БД	– база даних;
СКБД	– система керування базами даних;
СКРБД	– система керування реляційними базами даних;
ER-модель	– модель Entity-Relationship (модель «сутність-зв'язок»)

Анотація

Курсова робота Ровніної Т. Д. на тему “Розробка веб-сайту для туристичного агентства”, 89 с., 10 табл., 21 рис., 14 джерел, 2 додатка.

Об’єктом дослідження є пропозиції та веб-системи організацій, які пропонують туристичні послуги.

Метою роботи є отримання інформаційної веб-системи для взаємодії потенційних клієнтів з туристичним агентством через інтернет.

Створений програмний продукт містить такі складові:

- базу даних з необхідною інформацією;
- сервер для обробки запитів;
- веб-сайт для роботи з клієнтом.

Для демонстрації роботи веб-сайту створено вигадане туристичне агентство «Around The World», яке містить вигадані пропозиції турів.

В результаті виконання роботи були досліджені теоретичні основи та прикладні підходи до реалізації процесу створення веб додатків і реалізована інформаційна веб-система.

NODE.JS, EXPRESS, MVC, DATABASE, SQL, MYSQL, CLIENT-SERVER, WEB, PUG, JAVASCRIPT, TRAVEL AGENCY

Вступ

Актуальність даної роботи визначається тим, що у сучасному світі дуже поширене ведення бізнесу через інтернет ресурси. Це дає велику кількість переваг, серед яких – доступ до даних у будь-який зручний час і з будь-якої точки світу. Туристичний бізнес не є виключенням. Для привернення уваги та інформування потенційних і існуючих клієнтів туристичні агентства звертаються до створення публічних веб-сайтів з інформацією про туроператорів, наявні подорожі та інші можливі дані, які цікавлять людей. Інтернет сторінки також вирішують проблему зі скороченням витраченого часу на певного клієнта агенції та забезпеченням його більш якісним обслуговуванням.

Метою роботи є отримання готової інформаційної веб-системи для взаємодії людей з туристичним агентством за допомогою персональних комп'ютерів, мобільних телефонів, планшетів та інші пристроїв, які мають доступ до всесвітньої глобальної мережі.

Об'єктом дослідження є пропозиції та веб-системи організацій, які пропонують туристичні послуги.

Обрані наступні методи дослідження:

- спостереження – вивчення послуг туристичних агентств, що дає змогу отримати первинну інформацію у вигляді сукупності тверджень;
- порівняння – встановлення подібності або відмінності існуючих систем;
- аналіз – розбиття досліджуваного об'єкта на частини.

Серед джерел дослідження можна виділити такі основні:

- веб-сайти агентств: «Поехали с нами», «Join UP!», «TUI»;
- рекомендації по створенню баз даних: Дж. К. Дейт «Введение в системы баз данных»;

- рекомендації по створенню графічного інтерфейсу: С. Круг «Веб дизайн: книга Стива Круга или «не заставляйте меня думать!», Д. Норман «Дизайн привычных вещей»,

Створений програмний продукт буде містити такі складові:

- базу даних з необхідною інформацією;
- сервер для обробки запитів;
- веб-сайт для роботи з клієнтом.

Для демонстрації роботи веб-сайту буде використовуватись вигадане туристичне агентство «Around The World», яке буде містити вигадані пропозиції турів.

Виконання даної роботи зачіпає такі проблеми спеціальності, як веб-програмування, інтеграція з СКБД і сховищами даних, застосування розподіленої архітектури програмного забезпечення, створення зрозумілих і простих користувацьких інтерфейсів, ефективної програмної розробки.

Під час розробки проекту було використано таке програмне забезпечення:

- мова програмування JavaScript;
- середовище розробки WebStorm;
- платформа для виконання мережевих застосунків – Node.js;
- розподілена система керування версіями файлів – Git;
- продукти MySQL для взаємодії з базою даних.

Робота складається зі вступу, трьох основних розділів, висновків, списку використаних джерел та додатків.

1. Аналіз предметної області. Постановка завдання курсової роботи

1.1. Аналіз сучасного стану питання та обґрунтування теми

Успіх туристичного бізнесу завжди залежав від кількості клієнтів. Щоб збільшити популярність свого агентства, працівники застосовували різні методи: привабливі вивіски, роздача флаєрів тощо. З розвитком та розповсюдженням інтернету збільшити увагу до організації стало у рази простіше. На сьогодні створення веб-сайту організації є необхідним для покращення ефективності роботи та способу заявити про себе великому загалу користувачів глобальної мережі. Також це дозволяє зменшити кількість працівників, а отже і кількість витрат.

До основних складових публічних сайтів туристичних агентств можна віднести:

- інформація про агентство;
- контактні дані;
- актуальні пропозиції турів.

Для автоматизації оформлень замовлень клієнтами у системі повинен бути передбачений інтерфейс, що дозволить зручний пошук за критеріями та надасть повну інформацію про кожен тур.

Система дозволить агентству та клієнту обмінюватись актуальною інформацією щодо деталей запропонованих турів, надавати відгуки інших користувачів та пришвидшить процес оформлення замовлення.

1.2. Огляд існуючих аналогів розробки

На сьогодні існує дуже багато організацій, які займаються туристичним бізнесом. Серед них були обрані та проаналізовані веб-сайти найбільш популярних: «Поехали с нами», «JoinUp!», «TUI».

«Поехали с нами» — одне з найбільших туристичних агентств України. Мережа нараховує більш ніж 300 офісів в 78 містах. Вони співпрацюють з

різними туристичними операторами і продають їх пропозиції. Інтерфейс веб-сайту зображений на рисунку 1.1.

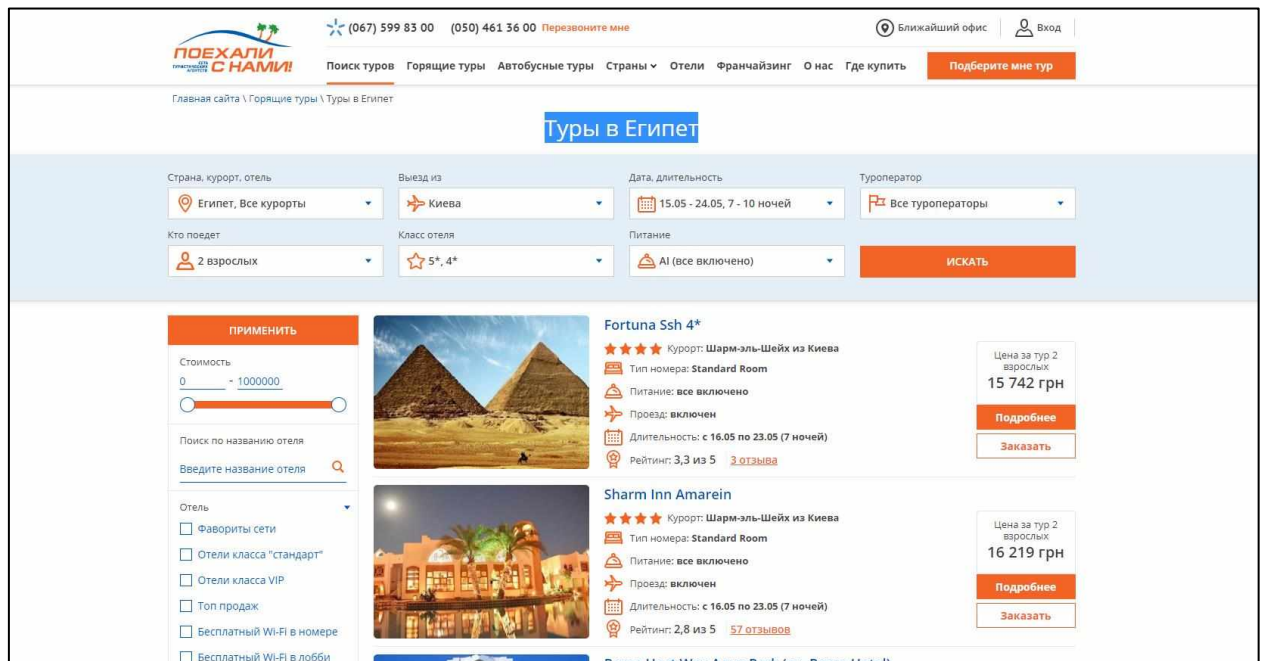


Рисунок 1.1 - Интерфейс сайту «Поехали с нами»

Веб-сайт «Поехали с нами» має такі позитивні сторони:

- наявність контактної інформації про агентство;
- реєстрація на сайті;
- детальна інформація про тури;
- можливість фільтрації за багатьма критеріями;
- наявність відгуків про готель;
- можливість оформити тур онлайн.

Але не зважаючи на велику кількість переваг також є недоліки:

- одномовність (російська мова);
- забагато зайвої інформації;
- дратуюче постійне появлення чату з консультантом;
- сторінка фіксованого розміру.

Також окремої уваги заслуговують кольори веб-сайту. Яскравий помаранчевий колір привертає увагу клієнта до деталей, а білий і ніжно блакитний кольори слугують добрим фоном.

«JoinUp!» — багатопрофільний український туроператор. На рисунку 1.2 можна побачити інтерфейс сайту.

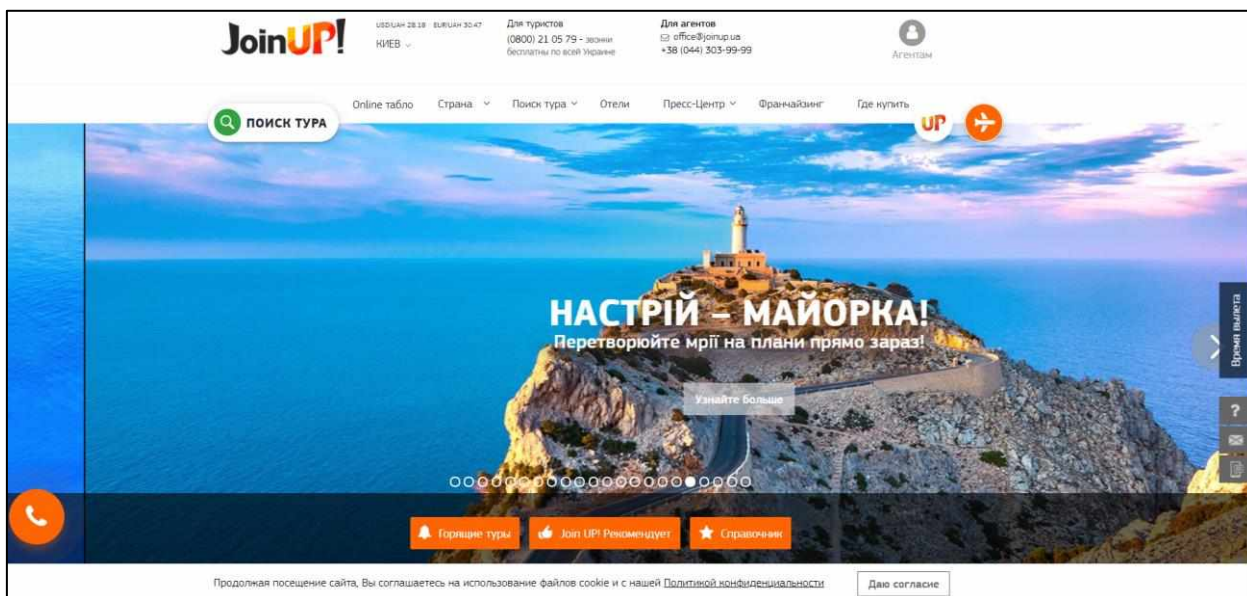


Рисунок 1.2 - Інтерфейс сайту «JoinUp!»

Веб-сайт має схожі характеристики, що і «Поехали с нами». На додачу наявна можливість переглянути рейси літаків на Online табло. З недоліків було виявлено не дуже зручний перегляд турів. Інформація на сайті доволі одноманітно розташована і погано привертає увагу.

«TUI» — англо-німецька багатонаціональна компанія для подорожей та туризму. Це найбільша у світі компанія для дозвілля. Їй належать туристичні агенції, готелі, авіакомпанії, круїзні судна та роздрібні магазини. Інтерфейс веб-сайту надано на рисунку 1.3.

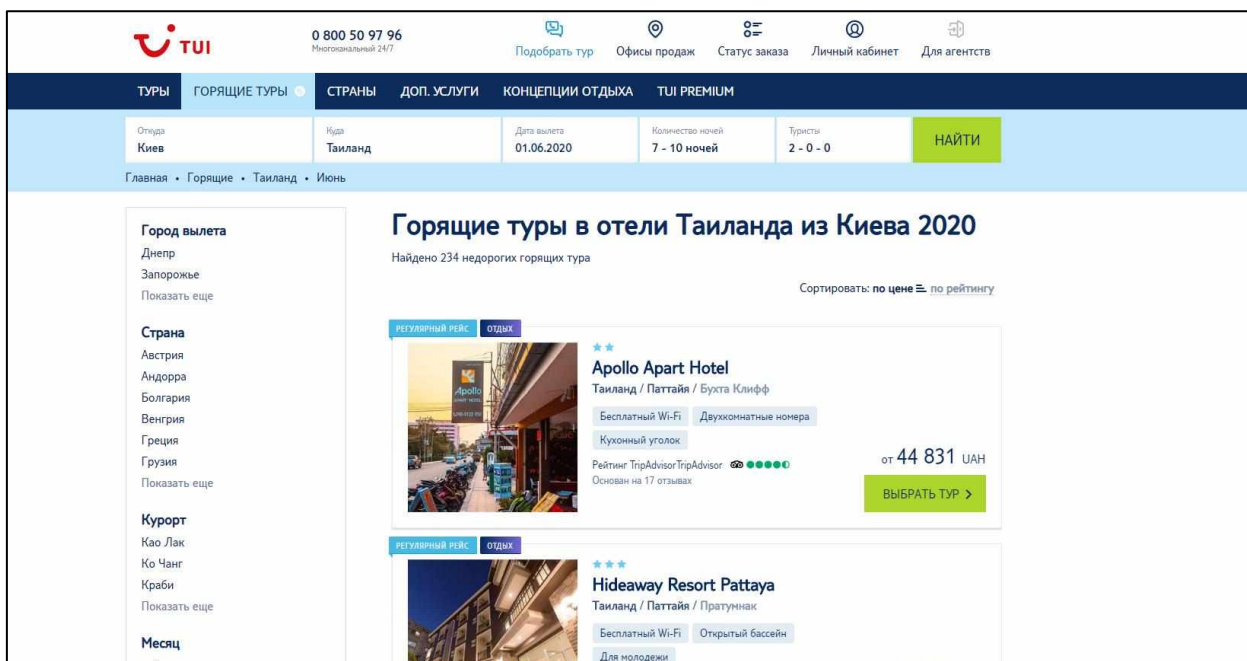


Рисунок 1.3 – Интерфейс сайту «TUI»

Переваги і недоліки сайту схожі на попередні зазначені. Також серед плюсів наявна багатомовність, але не одразу зрозуміло, де її знайти. Інтерфейс привабливий і зрозумілий.

Проаналізувавши три веб-сайти можна зробити порівняльний аналіз за такими основними критеріями: зручність інтерфейсу, привабливий дизайн, відсутність зайвої інформації, наявність контактних даних, багатомовність, наявність форми зворотного зв'язку, зручний пошук.

Наявність чи відсутність цих вимог у кожному з розглянутих сайтів наведено у таблиці 1.1.

Таблиця 1.1 – Таблиця порівняння існуючих систем

Критерії	Поехали с нами	JoinUp	TUI
зручність інтерфейсу	+	-	-
привабливий дизайн	+	-	+

Продовження таблиці 1.1

відсутність зайвої інформації	-	-	-
контактні дані	+	+	+
багатомовність	-	-	+
форма зворотного зв'язку	+	+	+
зручний пошук	+	-	+

1.3. Постановка завдання

Метою проекту є створення веб-сайту для туристичного агентства, щоб полегшити взаємодію клієнтів з працівниками організації. Для забезпечення функціонування сайту потрібно створити базу даних, яка буде містити актуальну інформацію, та сервер для зв'язку між базою та клієнтом.

Веб-сайт має реалізовувати наступний функціонал:

- надавати інформацію про агентство;
- забезпечувати зв'язок з персоналом організації;
- містити тури двох категорій: звичайні та зі знижками;
- надавати перелік і опис готелів;
- дозволяти зручний пошук за різними категоріями: дата, країна, туроператор, трансфер, категорія готелю, тип харчування тощо;
- можливість коментувати та переглядати відгуки на готель;
- надавати можливість оформлення заявки на тур;
- надсилати на пошту клієнта деталі про результати обробки замовлення.

Сайт має відповідати наступним вимогам:

- забезпечувати адаптивність сторінок до розмірів екрану;
- відповідати вимогам мінімального дизайну;

- надавати інформацію українською мовою;
- забезпечувати контроль інформації, яку вводить користувач, з метою запобігання введення помилкових даних;
- контент сторінки має бути написаний на зрозумілій для користувача мові;
- забезпечувати естетичну і функціональну однаковість;
- кольорова гама сайту має бути функціональною, тобто виділяти деталі кольором, який буде гармонічно поєднуватись з фоном сайту;
- сторінки мають містити ознаки характерні для обраного агентства.

2. Теоретичні відомості

2.1. Опис засобів розробки

2.1.1. JavaScript як мова програмування

JavaScript – прототипно-орієнтована скриптова мова програмування призначена для створення інтерактивних веб-сторінок.

Основні переваги використання JavaScript:

- підтримується майже усіма браузерами;
 - простий в освоєнні і застосуванні;
- має багато бібліотек з додатковим функціоналом;
- підключається до HTML-сторінки і працює без компіляції;
 - дозволяє розвантажити сервер, адже виконується на стороні користувача.

Основна ідея JavaScript полягає в можливості зміни значень HTML-атрибутів і властивостей середовища відображення в процесі перегляду веб-сайту користувачем. Також при використанні AJAX перезавантаження сторінки не відбувається.

2.1.2. Веб-сервер Node.js

Node.js – це програмна платформа, яка робить JavaScript мовою загального призначення. Її також називають середовищем виконання JS. Вона вміє зв'язуватися з зовнішніми бібліотеками, викликати команди з коду і виконувати роль веб-сервера. Нижче будуть приведені переваги Node.js в порівнянні з іншими серверними фреймворками.

Завдяки Node.js з'являється можливість використовувати одну мову програмування на клієнті і сервері. Якщо розробник має навички в JavaScript, йому буде легше вивчити Node.js. Зменшується код, що присутній і на стороні клієнта, і на стороні сервера. Об'єкти з однаковими назвами можуть виконувати абсолютно різні функції в браузері і на сервері.

Node.js розроблявся спеціально для веб-програмування. Він вільно взаємодіє з найбільш популярними базами даних і допомагає отримати доступ до протоколів передачі даних: http, udp, https, tcp.

Node.js пришвидшує написання програми. Створення робочого коду займає небагато часу. Якщо архітектура добре продумана, то в подальшому труднощів з тим, щоб розширити сайт на Node.js теж не з'явиться.

Вбудована бібліотека JavaScript з кожним оновленням розширюється. Кількість готових модулів і зовнішніх бібліотек безперервно зростає, в чому сприяє використанню менеджера пакетів npm.

Node.js використовує рушій JavaScript V8. Він раціонально витрачає пам'ять, непогано оптимізований і дає функціонал по профілізації процесора і пам'яті.

Технологія стрімко покращується завдяки великій спільноті. Node.js має відкритий код, тому веб-розробники можуть писати різні модулі та пакети і ділитися ними.

2.1.3. Шаблонізатор Pug

Шаблонізатор – програмне забезпечення, що дозволяє використовувати HTML-шаблони для генерації кінцевих HTML-сторінок. Застосування шаблонізатора значно прискорює верстку сайту, за рахунок повторного використання одних і тих же блоків коду, а також істотно спростить надалі оновлення інформації на готових сторінках.

Існує безліч шаблонізаторів для різних мов програмування. Саме для JavaScript можна використати Hogan.js, Nunjunks, Twig, Pug тощо. Але оскільки в якості сервера був обраний Node.js, то для нього краще підійде шаблонізатор Pug.

Pug (jade) – це HTML препроцесор, написаний на JS для роботи на платформі Node.js, з метою прискорення верстки інтерфейсу сайту.

Переваги Pug:

- відутність відкриваючих і закриваючих тегів;
- вкладеність регулюється відступами.

З недоліків можна виділити чутливість шаблонізатору до відступів та пробілів.

2.1.4. Додаткові бібліотеки

2.1.4.1. Бібліотека JQuery

jQuery — популярна JavaScript-бібліотека з відкритим кодом. Бібліотека містить набір корисних функцій, що спрощує взаємодію з HTML і CSS, обробку подій, управління анімацією тощо. Також jQuery надає можливість оновлювати контент сторінки без її перезавантаження за допомогою AJAX-запитів.

2.1.4.2. Набір інструментів Bootstrap

Bootstrap — це безкоштовний набір інструментів з відкритим кодом, призначений для створення веб-сайтів та веб-додатків, який містить шаблони CSS та HTML для представлення форм, кнопок, навігації та інших компонентів інтерфейсу. Також він містить додаткові розширення JavaScript.

Використання компонентів Bootstrap пришвидшує верстку сайту та робить його інтерфейс більш привабливим.

2.1.5. Середовище розробки WebStorm

На сьогодні існує багато різних середовищ редагування коду. Серед них найбільш відомими є: Sublime text, Atom, Visual Studio, WebStorm, Eclipse та інші. У них є свої переваги і недоліки. Для розробки з використанням мови програмування JavaScript і серверу Node.js компанія JetBrains рекомендує використовувати середовище розробки WebStorm.

WebStorm – це інтегроване середовище розробки для JavaScript, HTML та CSS, яке розроблене на основі платформи IntelliJ IDEA. Цей інструмент розроблено компанією JetBrains і є платним. Але є можливості отримати безкоштовну ліцензію.

Далі будуть приведені переваги WebStorm:

- розумний редактор коду: WebStorm використовує розумне автодоповнення коду, перевірку помилок на льоту, швидку навігацію по коду і рефакторинг для мов JavaScript і TypeScript, а також для популярних фреймворків;
- інтеграція з системами контролю версій: у WebStorm наявний простий і універсальний інтерфейс для роботи з Git, GitHub, Mercurial і іншими системами контролю версій. Робити комміти, переглядати внесені зміни і вирішувати конфлікти можна прямо в IDE;
- відлагоджувач: налагодити додаток можна прямо в IDE. У WebStorm є можливість ставити точки зупину в вихідному коді, переглядати стек викликів, стежити за значеннями змінних і використовувати інтерактивну консоль;
- програми на Node.js: WebStorm підтримує повноцінне налагодження Node.js додатків. Новий додаток можна створити, використовуючи шаблон Node.js Express, а необхідні модулі встановити за допомогою вбудованого в WebStorm менеджера npm.

2.2. Вибір СКБД

Система керування базами даних (СКБД) – це програмне забезпечення, за допомогою якого користувачі можуть визначати, створювати та підтримувати базу даних, а також здійснювати контрольований доступ до неї.

Вибір СКБД є одним з важливих етапів при розробці застосунків. Система повинна забезпечувати реляційну модель роботи з даними, тобто, щоб зберігати і працювати з даними, такий тип СКБД має мати певну

структуру таблиці і відповідати аспектам цілісності. Кожна таблиця складається з безлічі атрибутів і має унікальний ключ, що ідентифікує об'єкт.

Обраний програмний продукт повинен задовольняти як поточні так і майбутні потреби користувачів. Отже, для розробки проекту необхідно обрати СКБД, яка найбільше буде відповідати висунутим вимогам.

Розглянемо три найбільш розповсюджені реляційні СКБД, які можна було б використати для виконання даної роботи: SQLite, MySQL, PostgreSQL.

2.2.1. Переваги і недоліки SQLite

SQLite – це бібліотека, яка легко вбудовується в додаток. Будучи файловою БД, вона надає досить широкий набір інструментів для роботи з нею, в порівнянні з серверними БД. При роботі з SQLite звернення відбуваються за допомогою функціональних і прямих викликів файлів, що містять дані. Це підвищує швидкість і продуктивність операцій.

Переваги SQLite:

- файлова структура: база даних складається з одного файлу, тому її дуже легко переносити на різні машини;
- стандартизована: SQLite використовує SQL, але деякі функції опущені (RIGHT OUTER JOIN або FOR EACH STATEMENT);
- добре підходить для розробки і тестування: SQLite, зі своїм багатим набором функцій, може надати більш ніж достатній функціонал.

Недоліки SQLite:

- відсутність призначеного для користувача управління;
- відсутність додаткових налаштувань.

SQLite варто використовувати у таких випадках: вбудовані програми, система доступу до дискової пам'яті, тестування. У випадку, коли це програми розраховані на багато користувачів або додатки, що використовують великі обсяги даних, краще SQLite не використовувати.

2.2.2. Переваги і недоліки MySQL

MySQL – це найпопулярніша з усіх серверних БД. Вона пропонує широкий функціонал, хоча і не повністю реалізує SQL-стандарти. Розібратися в ній досить просто, так як в мережі про неї є багато інформації.

Переваги MySQL:

- простота: MySQL легко встановлюється. Існує багато сторонніх інструментів для роботи з цією БД;
- багато функцій: підтримує більшу частину функціоналу SQL.
- безпека: в MySQL вбудовано багато функцій безпеки;
- потужність: MySQL може працювати з великими обсягами даних, і непогано підходить для масштабованих додатків;
- швидкість: нехтування деякими стандартами дозволяє MySQL працювати продуктивніше.

Недоліки MySQL:

- обмеження: MySQL не може зробити все, що завгодно, і в ній присутні певні обмеження функціоналу.
- питання надійності: деякі операції реалізовані менш надійно, ніж в інших СКРБД.
- застій в розробці: робота над цим продуктом сильно загальмована.

MySQL варто використовувати у цих випадках: розподілені операції, висока безпека, веб-сайти. Випадки, коли не варто використовувати MySQL: SQL-сумісність, одночасні операції читання і запис, потреба в багатьох функціях.

2.2.3. Переваги і недоліки PostgreSQL

PostgreSQL – це найбільш просунута СКРБД, що орієнтується в першу чергу на повну відповідність стандартам і розширюваність. PostgreSQL

намагається повністю відповідати SQL-стандартам. Ця БД відмінно справляється з одночасною обробкою декількох завдань.

Переваги PostgreSQL:

- повна SQL-сумісність;
- підтримка сторонніми організаціями: PostgreSQL підтримується досвідченою спільнотою та використовується в багатьох інструментах, пов'язаних з СКРБД;
- можливість розширення: PostgreSQL можна програмно розширити за рахунок збережених процедур;
- об'єктно-орієнтованість: PostgreSQL - не тільки реляційна, а й об'єктно-орієнтована СУБД.

Недоліки PostgreSQL:

- продуктивність: в простих операціях читання PostgreSQL може поступатися іншим БД;
- популярність: через свою складність інструмент не дуже популярний;
- хостинг: проблематично знайти відповідного провайдера через перераховані вище фактори.

Коли варто використовувати PostgreSQL: цілісність даних, складні процедури, інтеграція. Краще не використовувати у випадках простих ситуацій і коли важлива швидкість.

Висновок з вибору СКБД

Розглянуті нами СКБД є непоганими продуктами і мають як свої переваги так і недоліки. Спираючись на вимоги проекту було обрано базу даних MySQL. Вона є найбільш зручною у використанні та може легко впоратись з поставленими задачами проекту.

3. Опис реалізації програмного продукту

3.1. Аналіз технічного завдання

У цьому розділі будуть детально розглянуті пункти технічного завдання, викладеного у підрозділі 1.3.

Інформація про агентство має складатись з назви, логотипу і контактних даних. А саме має бути адреса і її місце розташування на карті, посилання в соціальних мережах і телефон.

Про кожен тур мають бути відомі такі дані: країна і місто призначення, дата початку туру, кількість ночей, трансфер, назва туроператора, назва готелю, категорія готелю, приблизна ціна за одну людину. Додатково може бути наявний опис готелю, програма туру та ціна зі знижкою.

Клієнту потрібно надати можливість переглядати не тільки тури, а й готелі. Відгуки про готель мають бути доступні як на сторінці готелю так і на сторінці туру. Якщо коментарів багато, потрібно розміщувати їх на різних сторінках.

Операція оформлення заявки на тур має бути швидкою та зрозумілою. Клієнту треба надати можливість переглянути детальну інформацію про тур, що сподобався, та сформулювати на нього заяву без зайвого клопоту. Для оформлення турист має мати можливість ввести свої контактні дані, кількість людей, обрати тип харчування з запропонованих та за бажанням додати коментар до замовлення. Після підтвердження на пошту замовника має прийти повідомлення про успішне проходження операції, в якому міститься код замовленого туру.

Дизайн веб-сайту повинен бути зрозумілим і не містити зайвого. Важливі елементи такі як кнопки, знижки тощо – мають бути виділені іншим кольором, який буде гармонічно дивитись з обраним фоном. Дизайн має містити кольори характерні обраній туристичній агенції, щоб у клієнта складалось більше асоціацій з організацією.

Щоб запобігти введення помилкових даних клієнтом потрібно обмежувати вибір дати, надавати перелік можливої інформації та робити підказки до полів.

3.2. Архітектура системи

У цьому підрозділі наводиться опис архітектури системи та взаємодії її елементів. Архітектура системи туристичного агентства представлена на рисунку 3.1.

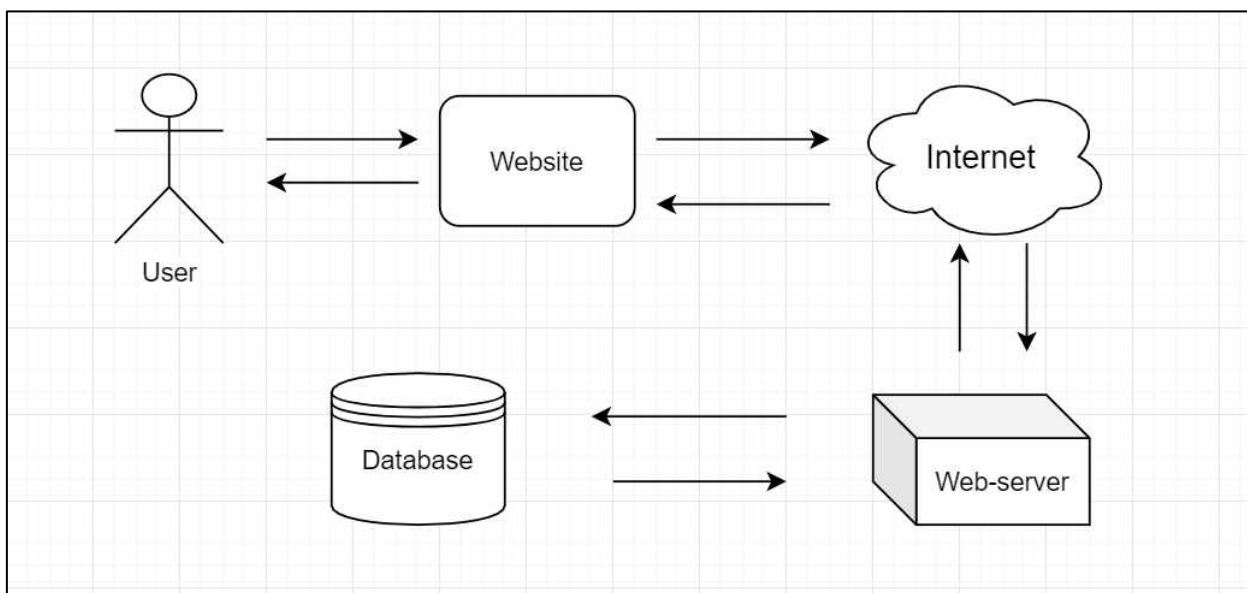


Рисунок 3.1 – Архітектура системи тур агентства «Around the world»

До складу системи входять такі компоненти:

- база даних MySQL;
- веб-сервер Node.js;
- веб-сайт на стороні клієнта.

Клієнт взаємодіє із системою використовуючи браузер комп'ютера, в якому відображається сайт організації. Вся інформація, що пов'язана з системою, зберігається у базі даних. Основним компонентом системи, що розробляється є сервер, який буде обробляти запити, що надходять з боку

клієнта, і брати дані з бази даних. Взаємодія між сервером і клієнтським веб-сайтом буде відбуватись за допомогою HTTP-протоколу. Щоб отримати дані з БД, сервер використовує SQL-запити.

Можна вважати, що в системі використовується шаблон MVC (Model-View-Controller). Він передбачає поділ системи на три взаємопов'язані частини: модель даних, інтерфейс користувача та модуль керування. У запропонованій схемі роль моделі даних виконує база даних, інтерфейс користувача представлений у вигляді сайту, а за модуль керування відповідає веб-сервер.

3.3. Реалізація бази даних

У даному розділі будуть детально описані етапи створення БД туристичного агентства «Around the world». Запити для створення і результати реалізації таблиць подано в Додатку А. Інформація, якою були заповнені таблиці, частково вибиралася з сайтів туристичних організацій і частково була вигадана.

3.3.1. Розробка ER-моделі

Першим етапом у розробці бази даних є побудова її ER-моделі.

ER (Entity-Relationship) - модель (модель «сутність-зв'язок») — графічна мова моделювання, яка дозволяє за допомогою узагальнених конструкцій блоків описувати об'єкти БД, їх властивості, зв'язки і обмеження.

На рисунку 3.2 зображена схема ER-моделі для бази даних туристичного агентства «Around the world».

Було створено такі типи сутностей: заяви, тури, туроператори, трансфери, харчування, готелі, країни, відгуки.

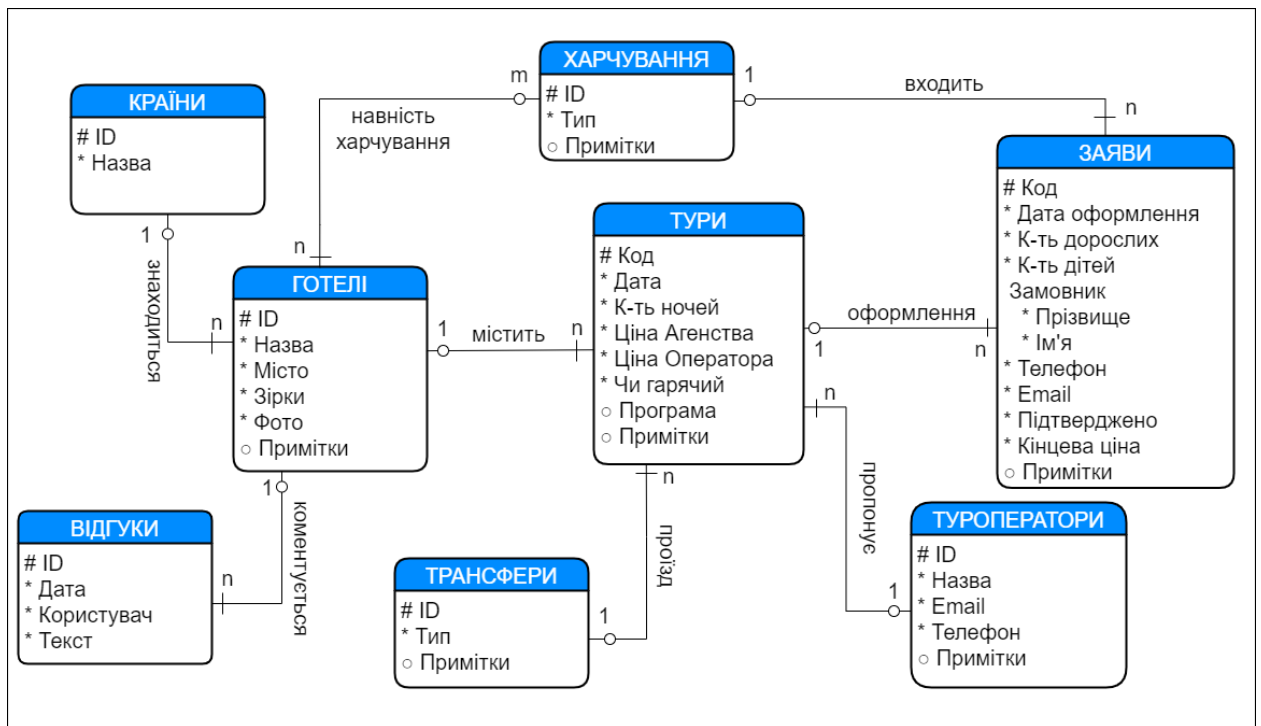


Рисунок 3.2 – ER-модель бази даних туристичного агентства

Сутність «Заяви» відповідає за документ, який оформлюється на певний тур при замовленні його клієнтом. Ключ – «код», тому що лише він може однозначно ідентифікувати екземпляр даної сутності.

Сутність «Тури» відповідає за тури, які пропонує агентство. Ключ – «код».

Сутність «Туроператори» відповідає за туристичних операторів, з якими співпрацює агентство і які надають організації свої тури. Ключ – «ID».

Сутність «Трансфери» відповідає за спосіб перевезення, який буде входити в послуги туру. Наприклад, це може бути літак або автобус. Ключ – «ID».

Сутність «Харчування» відповідає за тип харчування, що пропонує готель. Це може бути все включене, тільки сніданок, трьох разове харчування тощо. Ключ – «ID».

Сутність «Готелі» відповідає за готель, в якому будуть проживати туристи під час відпочинку. Ключ – «ID».

Сутність «Країни» відповідає за країну, у якій знаходиться готель. Оскільки країни можуть часто повторюватись було вирішено винести їх в окрему сутність. Ключ – «ID».

Сутність «Відгуки» відповідає за коментарі, які залишають туристи до готелю. Ключ – «ID».

3.3.2. Розробка реляційної моделі

Наступним кроком у розробці БД є створення реляційної моделі даних спираючись на побудовану ER-модель.

Реляційна модель даних — логічна модель даних, де в ролі відношень виступає таблиця, а атрибутами є її рядки.

У таблицях 3.1 – 3.9 подані представлення реляційних моделей туристичного агентства.

Таблиця 3.1 – Реляційна модель Documents для сутності «Заяви»

Кл юч	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	code	INT	NOT NULL			Атрибут простий обов'язковий «Код».
FK	toure_code	INT	NOT NULL	NO ACTION	CASCADE	Зв'язок між сутностями «Тури» та «Заяви». Код туру. При видаленні туру, заява не змінюється. При оновленні, оновлюється і заява.
FK	food_id	CHAR(2)	NOT NULL	NO ACTION	CASCADE	Зв'язок між сутностями «Харчування» та «Заяви». Код харчування. При видаленні типу харчування, заява не змінюється. При оновленні, оновлюється і заява.
	registrDate	DATETIM E	NULL			Атрибут простий необов'язковий «Дата оформлення». За замовчення поточна дата
	adults	INT	NOT NULL			Атрибут простий обов'язковий «Кількість дорослих».
	children	INT	NULL			Атрибут простий необов'язковий «Кількість дітей». За замовченням 0.

Продовження таблиці 3.1

	clientSurname	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Прізвище замовника».
	clientName	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Ім'я замовника».
	phone	VARCHAR (13)	NOT NULL			Атрибут простий обов'язковий «Телефон».
	email	VARCHAR (50)	NOT NULL			Атрибут простий обов'язковий «Email».
	confirmed	BOOL	NOT NULL			Атрибут простий обов'язковий «Підтверджено». За замовченням false.
	price	DECIMAL	NOT NULL			Атрибут простий обов'язковий «Кінцева ціна»
	notes	VARCHAR (225)	NULL			Атрибут простий не обов'язковий «Примітки».

Таблиця 3.2 – Реляційна модель Tours для сутності «Тури»

Ключ	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	code	INT	NOT NULL			Атрибут простий обов'язковий «Код».
FK	operator_id	INT	NOT NULL	NO ACTION	CASCADE	Зв'язок між сутностями «Тури» та «Туроператори». При видаленні туроператора, тур не змінюється. При оновленні, оновлюється і тур.
FK	hotel_id	INT	NOT NULL	NO ACTION	CASCADE	Зв'язок між сутностями «Тури» та «Готелі». При видаленні готелю, тур не змінюється. При оновленні, оновлюється і тур.
FK	transfer_id	INT	NOT NULL	NO ACTION	CASCADE	Зв'язок між сутностями «Тури» та «Трансфери». При видаленні трансферу, тур не змінюється. При оновленні, оновлюється і тур.
	tourDate	DATE	NOT NULL			Атрибут простий обов'язковий «Дата»
	nights	INT	NOT NULL			Атрибут простий обов'язковий «Кількість ночей»
	price_new	DECIMAL	NULL			Атрибут простий необов'язковий «Ціна агентства»

Продовження таблиці 3.2

	price	DECIMAL	NOT NULL			Атрибут простий обов'язковий «Ціна оператора»
	isHot	BOOL	NOT NULL			Атрибут простий обов'язковий «Чи гарячий»
	program	VARCHAR (1000)	NULL			Атрибут простий необов'язковий «Програма»
	notes	VARCHAR (225)	NULL			Атрибут простий необов'язковий «Примітки»

Таблиця 3.3 – Реляційна модель Hotels для сутності «Готелі»

Кл юч	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	id_hotel	INT	NOT NULL			Атрибут простий обов'язковий «ID».
FK	country_id	CHAR(2)	NOT NULL	NO ACTION	CASCADE	Зв'язок між сутностями «Країни» та «Готелі». При видаленні міста, готель не змінюється. При оновленні, оновлюється і готель.
	name_hotel	VARCHAR (30)	NOT NULL			Атрибут простий обов'язковий «Назва».
	stars	INT	NOT NULL			Атрибут простий обов'язковий «Зірки»
	city	VARCHAR (30)	NOT NULL			Атрибут простий обов'язковий «Місто».
	photo	VARCHAR (225)	NOT NULL			Атрибут простий обов'язковий «Фото»
	description	VARCHAR (1000)	NULL			Атрибут простий необов'язковий «Примітки»

Таблиця 3.4 – Реляційна модель Operators для сутності «Туроператори»

Кл юч	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	id_operator	INT	NOT NULL			Атрибут простий обов'язковий «Id»
	name_operator	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Назва».
	email	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Email».

Продовження таблиці 3.4

	phone	VARCHAR (18)	NOT NULL			Атрибут простий обов'язковий «Телефон».
	notes	VARCHAR (255)	NULL			Атрибут простий не обов'язковий «Примітки».

Таблиця 3.5 – Реляційна модель Transfers для сутності «Трансфери»

Кл юч	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	id_transfer	INT	NOT NULL			Атрибут простий обов'язковий «Id».
	type_transf er	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Тип».
	notes	VARCHAR (225)	NULL			Атрибут простий не обов'язковий «Примітки».

Таблиця 3.6 – Реляційна модель Food для сутності «Харчування»

Кл юч	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	id_food	CHAR(2)	NOT NULL			Атрибут простий обов'язковий «Id».
	type_food	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Тип».
	notes	VARCHAR (225)	NULL			Атрибут простий не обов'язковий «Примітки».

Таблиця 3.7 – Реляційна модель Food_availability для зв'язку «Наявність харчування»

Кл юч	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PPK FK	food_id	CHAR(2)	NOT NULL	CASCADE	CASCADE	При оновленні або видаленні харчування оновлювати/ видалити і зв'язок
PPK FK	hotel_id	INT	NOT NULL	CASCADE	CASCADE	При оновленні або видаленні готелю оновлювати/ видалити і зв'язок

Таблиця 3.8 – Реляційна модель Comments для сутності «Відгуки»

Ключ	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	id_comment	INT	NOT NULL			Атрибут простий обов'язковий «Id».
FK	hotel_id	INT	NOT NULL	CASCADE	CASCADE	Зв'язок між сутностями «Відгуки» і «Готелі». При оновленні або видаленні готелю оновлювати/ видалити коментар
	comment_date	DATETIME	NULL			Атрибут простий обов'язковий «Дата». За замовченням поточна дата
	user	VARCHAR (20)	NOT NULL			Атрибут простий обов'язковий «Користувач».
	email	VARCHAR (50)	NOT NULL			Атрибут простий обов'язковий «Email».
	text	VARCHAR (1000)	NOT NULL			Атрибут простий обов'язковий «Текст».

Таблиця 3.9 – Реляційна модель Countries для сутності «Країни»

Ключ	Ім'я атрибуту	Тип	N / NN	ON DELETE	ON UPDATE	Пояснення
PK	id_country	CHAR(3)	NOT NULL			Атрибут простий обов'язковий «Id».
	name_country	VARCHAR (16)	NOT NULL			Атрибут простий обов'язковий «Назва».

Первинний ключ подано в першому рядку кожної таблиці. У випадку зв'язку багато до багатьох, ця інформація знаходиться у першому і другому рядках, так як ключ складається із двох частин, які одночасно є і зовнішніми ключами.

Якщо у відношенні присутній зовнішній ключ, то його значення повинно або відповідати значенню первинного ключа деякого кортежу в його базовому відношенні, або бути рівним NULL. Стратегії та обґрунтування їх вибору знаходиться у таблицях у стовпчиках «ON DELETE», «ON UPDATE» і «Пояснення».

Нижче будуть перераховані корпоративні обмеження цілісності:

- заборонено оформлювати тур на дати менші за поточну;
- одну заяву можна оформити лише на один тур;
- кількість дорослих у заяві має бути більше нуля;
- ціна туру у копійках завжди «00»;
- нова ціна туру має бути менша за звичайну ціну;
- кількість зірок готелю може бути від 1 до 5;
- тип харчування обирається з певного переліку: ОВ (без харчування), ВВ (сніданок), НВ (сніданок + вечеря), FB (трьох разове харчування), AI (все включене).

3.4. Опис розробки серверної частини

Серверна частина застосунку знаходиться у файлі `server.js`, код якого наведено у Додатку Б.

Сервер був реалізований за допомогою фреймворку `Express.js` для створення веб-застосунків `Node.js`. Також у роботі були використані наступні модулі:

- `body-parser`. Використовується для обробки `POST` запитів;
- `mysql`. Модуль для підключення до БД `MySQL`;
- `nodemailer`. Використовується для відправки повідомлень на пошту.

Робота серверу здійснюється на основі прийому `HTTP`-запитів від клієнтської частини програми. У даному проекті використовуються такі типи запитів: `GET`, `POST`. Далі наведено перелік запитів, що реалізує система.

Запити GET:

- URL '/' – отримати головну сторінку програми. У відповідь надсилається шаблон головної сторінки та інформація про туристичних операторів для їх відображення на сторінці;
- URL '/about' – отримати сторінку з контактами агентства. У відповідь надсилається шаблон блоку з контактами;
- URL '/tours' – отримати сторінку з усіма турами агентства. У відповідь надсилається шаблон блоку з короткою інформацією по усім турам та фільтри;
- URL '/hot-tours' – отримати сторінку з турами зі знижкою. У відповідь надсилається шаблон блоку з короткою інформацією по турам зі знижкою та фільтри;
- URL '/hotels' – отримати сторінку з усіма готелями, у які пропонуються тури. У відповідь надсилається шаблон блоку з короткою інформацією по усім готелям та фільтри;
- URL '/api/tour' – отримати сторінку з описом певного туру. Код туру передається як параметр в частині URL. У відповідь надсилається шаблон блоку з повною інформацією по обраному туру і готелю, що належить до цього туру. Також надаються відгуки до готелю;
- URL '/api/hotel' – отримати сторінку з описом певного готелю. Код готелю передається як параметр в частині URL. У відповідь надсилається шаблон блоку з повною інформацією по обраному готелю та можливі тури у цей готель. Також надаються відгуки до готелю;
- URL '/filters' – отримати сторінку з відфільтрованими турами або готелями. Параметри фільтрування передаються в частині URL. У відповідь надсилається шаблон блоку з короткою інформацією по відфільтрованим турам/готелям.

Запити POST:

- URL `'/buyTour'` – оформлення заявки на тур. Данні заявки передаються в тілі запиту;
- URL `'/comment'` – додавання відгуку на готель. Данні коментаря передаються в тілі запиту.

Також реалізовано такі функції:

- `sendMessage(docId, client)` – відправити повідомлення на пошту клієнта про успішне оформлення заявки;
- `addComment(comment)` – додати відгук до готелю у БД;
- `buyTour(id, client)` – придбати тур. Додати в БД заяву на тур;
- `parseDate(date, txt)` – перетворити дату у формат «дд-мм-рррр». Місяць може бути зазначений як числом так і назвою;
- `getFilters()` – повертає всі параметри для фільтрації;
- `queryHotels(filters)` – формує SQL-запит використовуючи параметри фільтрації, якщо вони надані, для отримання з БД готелів;
- `queryTours(filters)` – формує SQL-запит використовуючи параметри фільтрації, якщо вони надані, для отримання з БД турів;
- `getAllHotels(filters)` – отримати всі готелі з можливою фільтрацією;
- `getAllTours(filters)` – отримати всі тури з можливою фільтрацією;
- `getHotTours(filters)` – отримати всі тури зі знижками з можливою фільтрацією;
- `getTour(id)` – отримати інформацію про тур за його кодом;
- `getHotel(id)` – отримати інформацію про готель за його кодом;
- `getComments(hotel)` – отримати відгуки до певного готелю;
- `getOperators()` – отримати інформацію про туроператорів, з якими співпрацює агентство.

3.5. Опис клієнтської частини

Функціонал клієнтської частини застосунку знаходиться у файлі `client.js`, а шаблони сторінок у файлах з розширенням `. pug`. Код наведено у Додатку Б.

3.5.1. Опис функціоналу клієнтської частини

Під час розробки клієнтської частини використовувались бібліотеки `Bootstrap` і `jQuery`. Для зв'язку із сервером і оновлення контенту сторінку без її перезавантаження використовувались `AJAX`-запити.

Навігація по сторінкам реалізована шляхом обробки дій користувача. При активізації певного елемента керування формується посилання, яке відправляється на сервер. У свою чергу сервер реагує на нього і надає необхідний контент сторінки.

Окрім серверних операцій використовуються функції, які виконуються безпосередньо клієнтом:

- переходи між сторінками відгуків;
- змінення елементів слайдеру для туристичних операторів;
- перевірка даних, що були введені клієнтом;
- відкриття прихованої форми для відгуків;
- відкриття форми для оформлення туру.

3.5.2. Опис інтерфейсу

Відповідно до стандартів оформлення веб-сайтів структура кожної сторінки складається з трьох основних частин:

- `Header` (шапка) – блок у верхній частині сторінки сайту. Містить логотип тур агентства, його назву та меню. Постійно присутній на сторінці;
- `Body` (тіло сайту) – основний блок відображення контенту, який змінюється в залежності від сторінки;

- Footer (підвал) – блок в нижній частині сторінки, який містить посилання на сайт GitHub з проектом. Постійно присутній на сторінці.

Макет інтерфейсу користувача надано на рисунку 3.3. Головна сторінка є базовою і при переході на інші сторінки змінюється лише блок BODY.

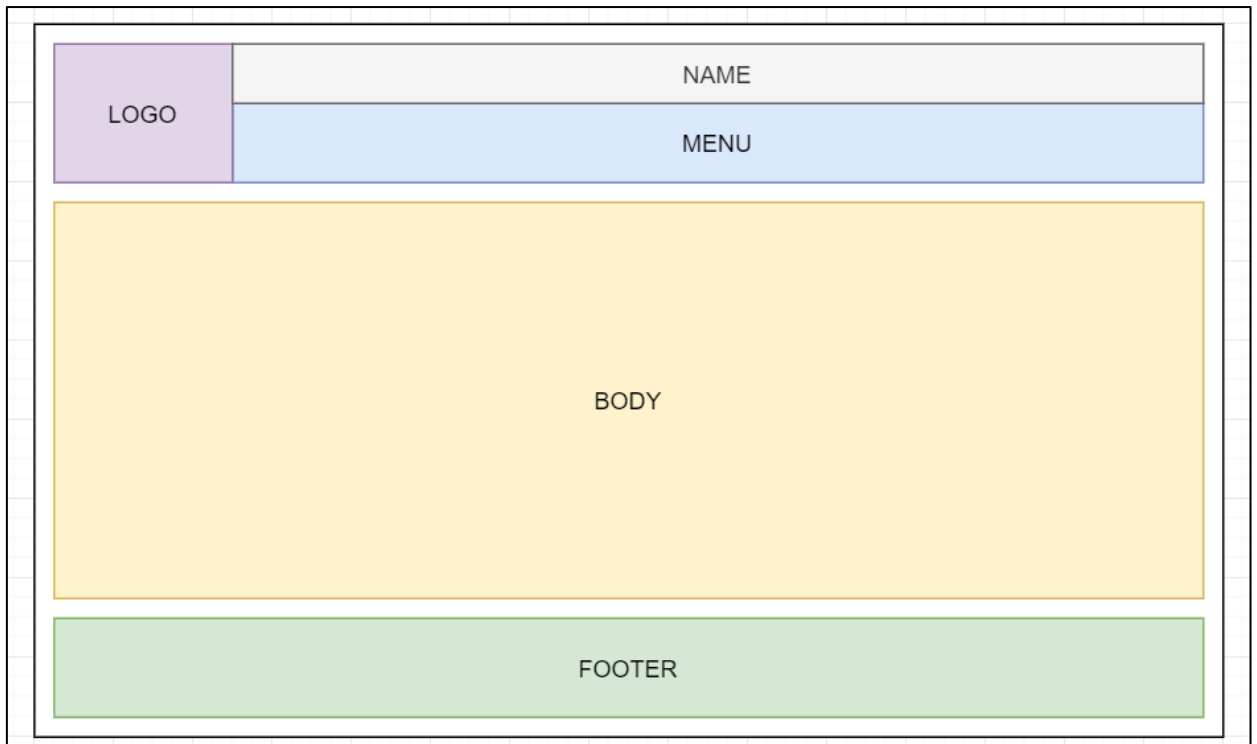


Рисунок 3.3 – Макет інтерфейсу користувача

Веб-сайт містить декілька сторінок, з кожної з яких можна потрапити на головну сторінку та сторінки доступні у верхньому меню. Мапу сайту організації зображено на рисунку 3.4. Далі буде наведено детальний опис кожної сторінки.

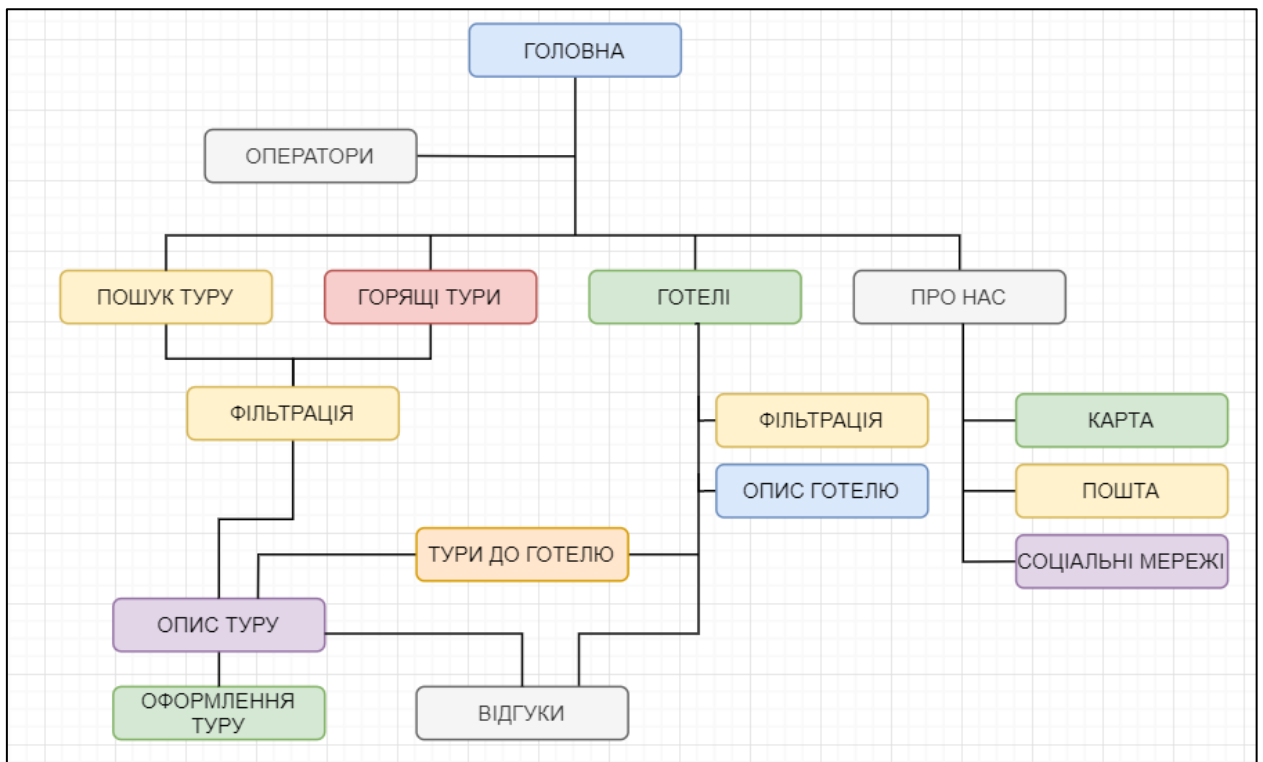


Рисунок 3.4 – Мапа сайту туристичного агентства

Головна сторінка. Містить таблицю з перевагами тур агентства «Around the world». Також наявний слайдер з короткою контактною інформацією про туроператорів, з якими співпрацює організація.

Пошук туру. На цій сторінці знаходяться усі тури від поточної дати, які пропонує агентство, та блок фільтрації за критеріями: країна, дата від, дата до, трансфер, туроператор, тип харчування, категорія готелю. При натисканні на картку туру відкривається його детальний опис.

Горячі тури. Сторінка схожа на попередню, але містить тури лише зі знижками.

Готелі. На цій сторінці знаходяться усі готелі, в які агентство пропонує тури, та блок фільтрації за критеріями: країна, тип харчування, категорія готелю. При натисканні на картку готелю відкривається його детальний опис.

Про нас. Сторінка містить контактні дані туристичного агентства: адресу, пошту, телефон, посилання на соціальні мережі.

Опис готелю. На цій сторінці знаходиться детальний опис готелю, перелік доступних турів у цей готель та відгуки. Також можна залишити свій відгук.

Опис туру. Містить детальний опис туру та відгуки. Можна залишити свій відгук та оформити заяву на цей тур.

Для оформлення туру клієнт має ввести такі необхідні дані: прізвище, ім'я, email, телефон, кількість дорослих і дітей, обраний тип харчування. За бажанням можна додати коментарі з питаннями і побажаннями.

Щоб залишити відгук на готель клієнт має надати: ім'я, email і текст коментаря.

Кольори веб-сайту складаються з різних відтінків синього, оскільки це колір логотипу вигаданого туристичного агентства «Around the world».

3.6. Тестування програми і результати її виконання

Нижче наведено опис кінцевого варіанту користувацького інтерфейсу та його тестування.

Щоб потрапити на сторінку туристичного агентства, потрібно в стрічку адреси ввести посилання «<http://localhost:8888/>». Завантажуючи сайт користувач одразу бачить головну сторінку, яка представлена на рисунку 3.5. Також на цю сторінку можна потрапити натиснувши на логотип та назву туристичного агентства розміщених вгорі сторінки у меню. На сторінці є слайдер з контактними даними туристичних операторів та інформація чому треба обрати саме це агентство.

Знизу сторінки є надпис «Copyright: Rovnina Tetiana», якщо на нього натиснути, то відбудеться перехід на сайт GitHub, де розміщено проект.

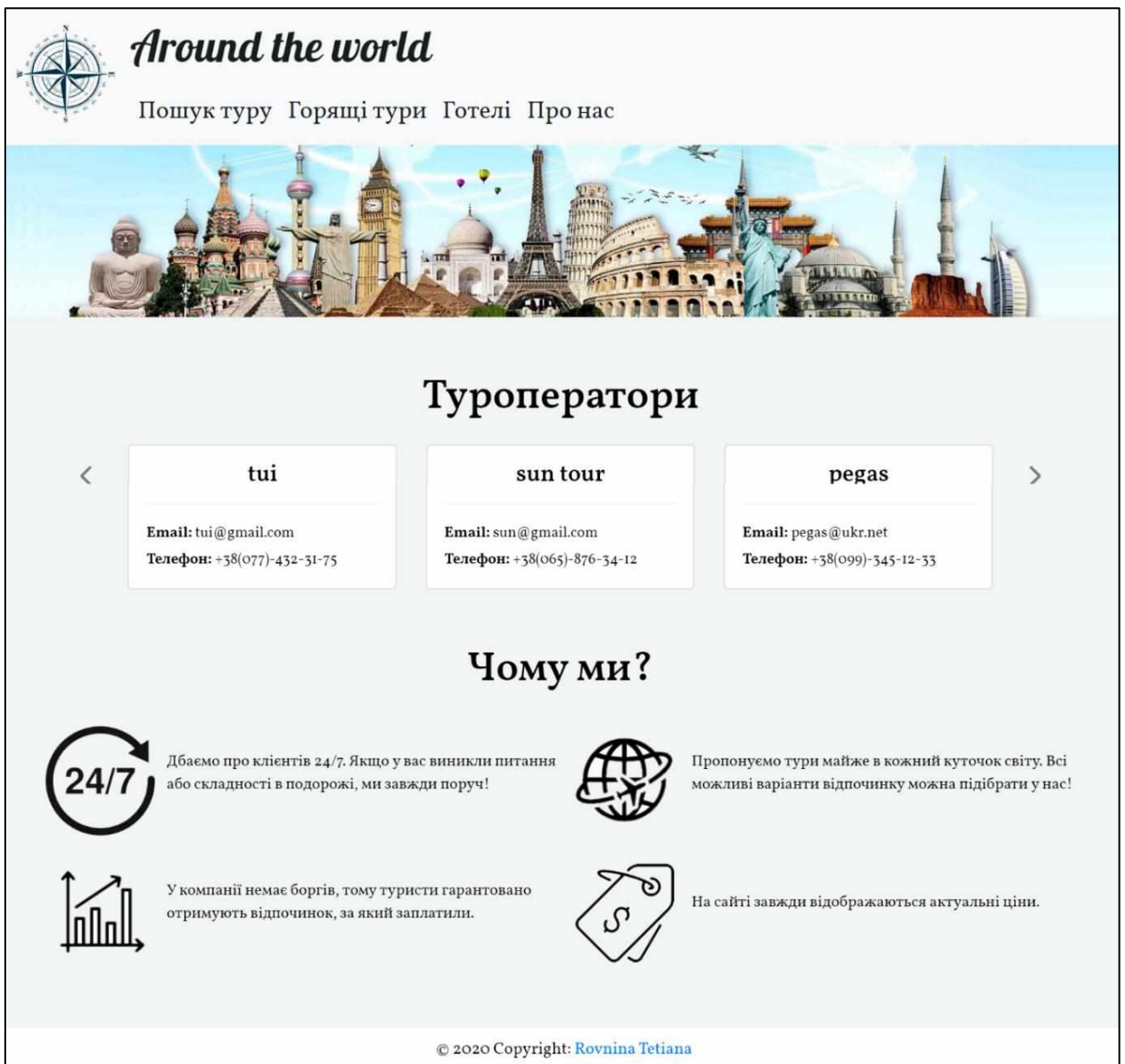


Рисунок 3.5 – Інтерфейс головної сторінки

Вгорі сторінки наявне меню, яке також присутнє на всіх інших сторінках. Якщо обрати пункт «Про нас», то перейдемо на сторінку з контактами організації. Її зображено на рисунку 3.6.

Сторінка містить контакти організації та гугл-карту з відміченим місцем розташування офісу туристичного агентства. Натиснувши на email, відбудеться перехід у пошту з можливістю написати лист на дану адресу. Відповідно при натисканні на іконки соціальних мереж: Facebook, Instagram, Telegram – відкриються відповідні сторінки.

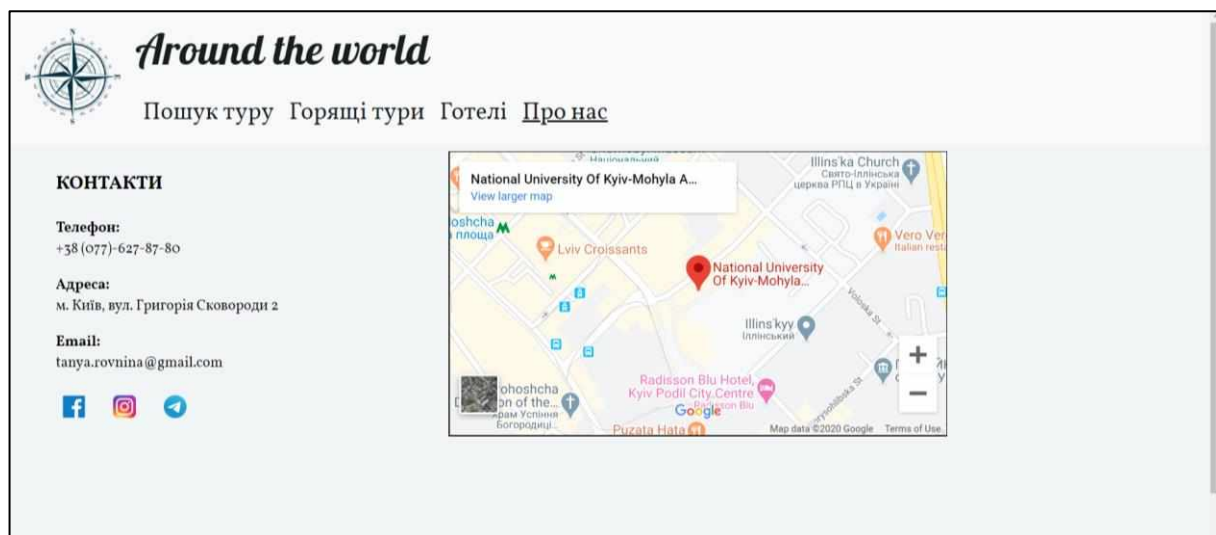


Рисунок 3.6 – Інтерфейс сторінки «Про нас»

Якщо у меню обрати «Пошук туру», то з'явиться сторінка з короткою інформацією по усім турами та блок фільтрації за критеріями. Вигляд сторінки зображено на рисунку 3.7.

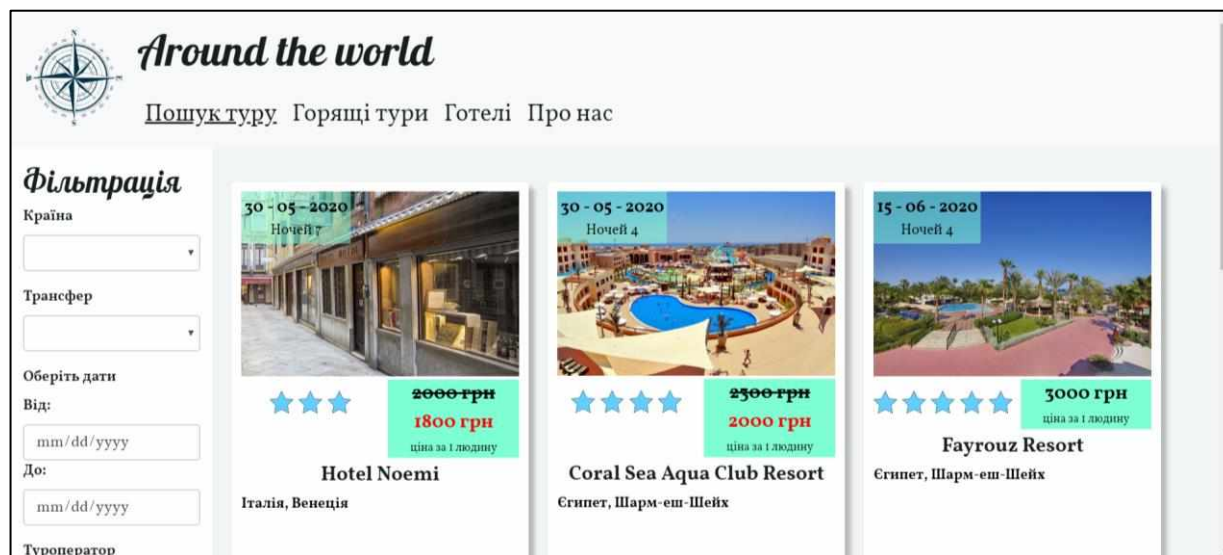


Рисунок 3.7 – Інтерфейс сторінки «Пошук турів»

Фільтрація

Країна

Трансфер

Оберіть дати

Від:

mm/dd/yyyy

До:

mm/dd/yyyy

Туроператор

Тип харчування

☐ AI (все включене)
 ☐ BB (сніданок)
 ☐ FB (3 р. харчування)
 ☐ HB (сніданок+ вечеря)
 ☐ OB (без харчування)

Категорія готелю

☐ 5*
 ☐ 4*
 ☐ 3*
 ☐ 2*
 ☐ 1*

Застосувати

Скинути

Повний вигляд блоку фільтрації зображений на рисунку 3.8.

Фільтрація не потребує заповнення усіх полів. Для тестування турів були введені такі параметри:

- країна: Греція;
- трансфер: літак;
- дата від: 07.05.2020;
- дата до: 31.07.2020;
- туроператор: — ;
- тип харчування: сніданок;
- категорія готелю: 5*, 4*, 3*, 2*.

Результат фільтрування надано на рисунку 3.9. Якщо обрати кнопку «Скинути», то знову з’являються всі тури.

Рисунок 3.8 – Блок фільтрації

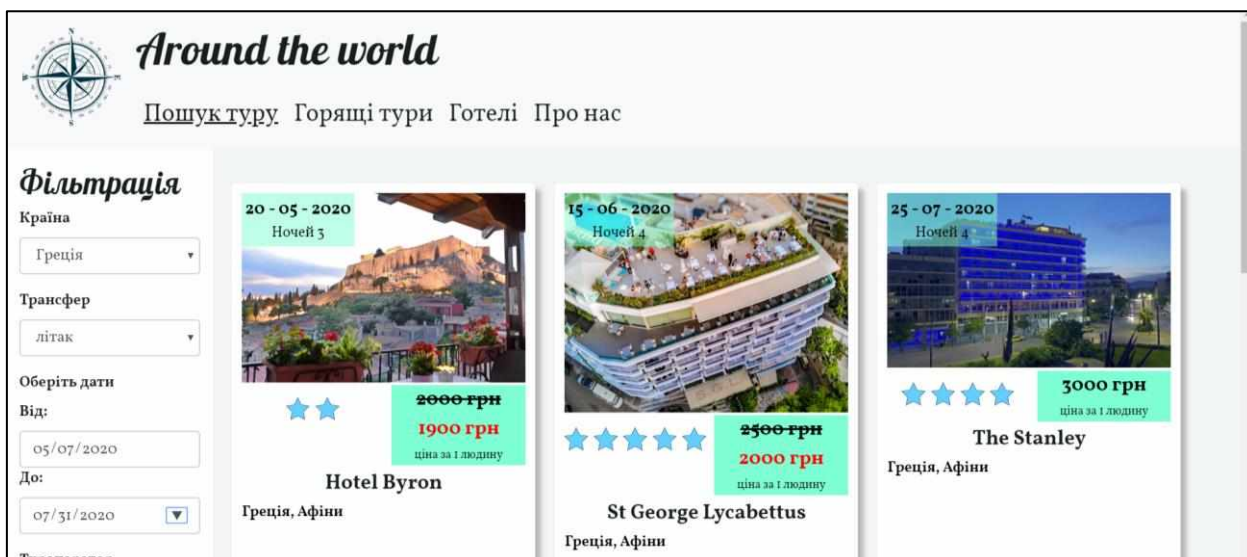


Рисунок 3.9 – Результат фільтрації турів

Якщо у меню обрати «Горящі тури», то з'явиться сторінка з турами зі знижками та блок фільтрації. Вигляд сторінки зображено на рисунку 3.10.

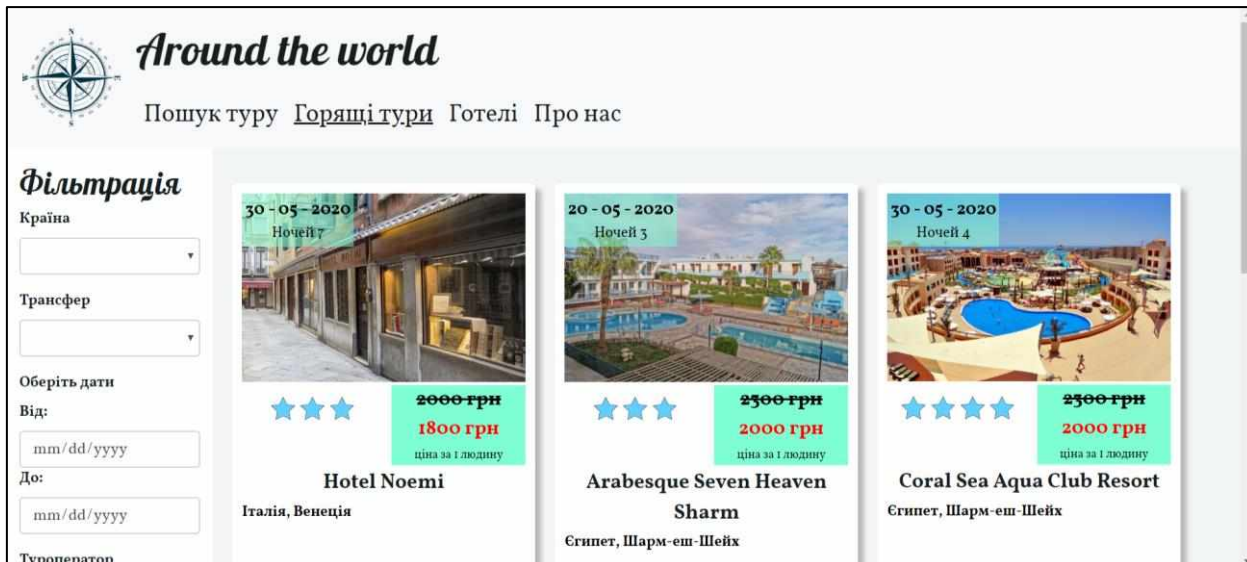


Рисунок 3.10 – Інтерфейс сторінки «Горящі тури»

При натисканні на картку туру відкривається сторінка з детальним описом туру та готелю, який йому належить. На рисунках 3.11 – 3.12 зображено опис одного з турів.

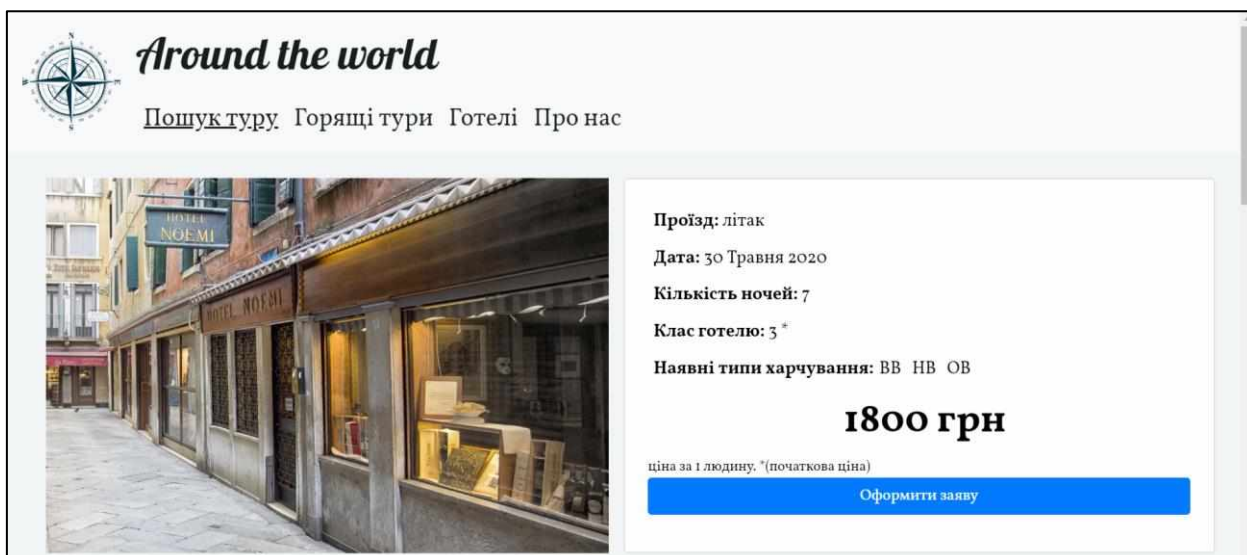


Рисунок 3.11 – Інтерфейс опису туру. Початок

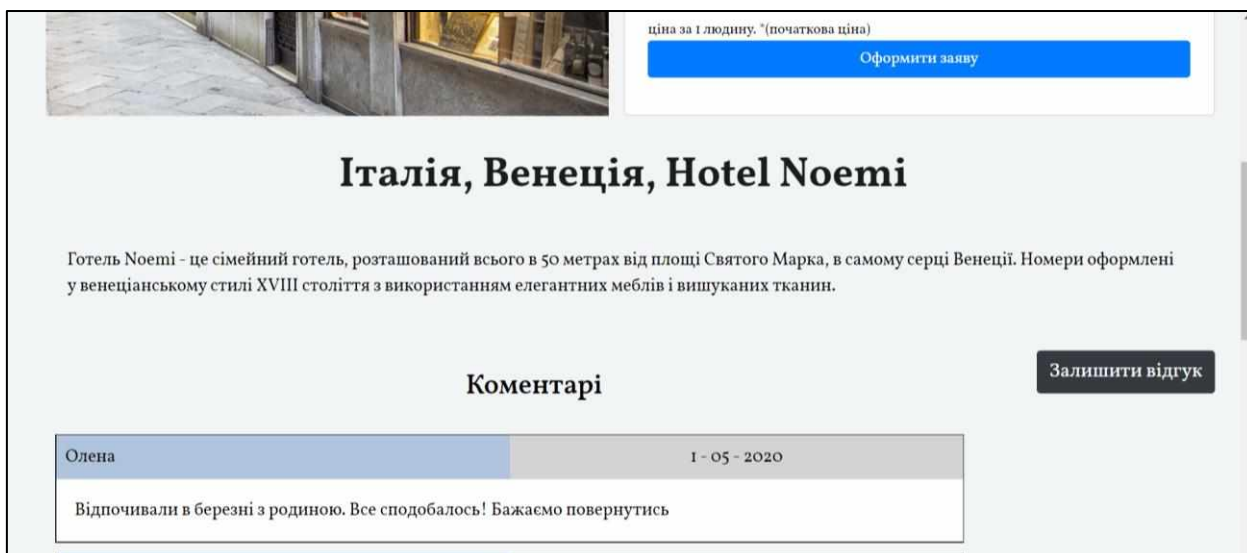


Рисунок 3.12 – Інтерфейс опису туру. Продовження

Знизу сторінки надано блок з відгуками. Якщо їх більше ніж 5 тоді з'являється ще одна сторінка для відгуків. Для переключення сторінок знизу наявна пагінація. При натисканні на кнопку «Залишити відгук» з'являється форма для введення даних зображена на рисунку 3.13.

Готель Noemi - це сімейний готель, розташований всього в 50 метрах від площі Святого Марка, в самому серці Венеції. Номери оформлені у венеціанському стилі XVIII століття з використанням елегантних меблів і вишуканих тканин.

Email *

Email

Ім'я *

Ім'я

Залишити відгук

Коментар *

Текст...

Залишити

Коментарі

Рисунок 3.13 – Форма для заповнення відгуку на готель

Щоб оформити заяву на тур треба натиснути на кнопку «Оформити заяву». Форма для заповнення зображена на рисунку 3.14.

ціна за 1 людину. * (початкова ціна)

Заповніть форму

Прізвище *

Іванов

Ім'я *

Іван

Email *

ivanov@gmail.com

Телефон *

099-657-98-90

Тип їжі*

ВВ (сніданок)

Кількість дорослих *

Кількість дітей *

Коментар

Текст...

© 2020 Copyright: Rovnina Tetiana

Рисунок 3.14 – Форма для заповнення заяви на тур

Для оформлення необхідно заповнити всі поля із зірочкою. Якщо забути про них або ввести неправильні значення, то систем надає підказку під відповідним полем. Після оформлення заяви на тур на пошту приходить

повідомлення про успішне виконання операції. Повідомлення надано на рисунку 3.15.

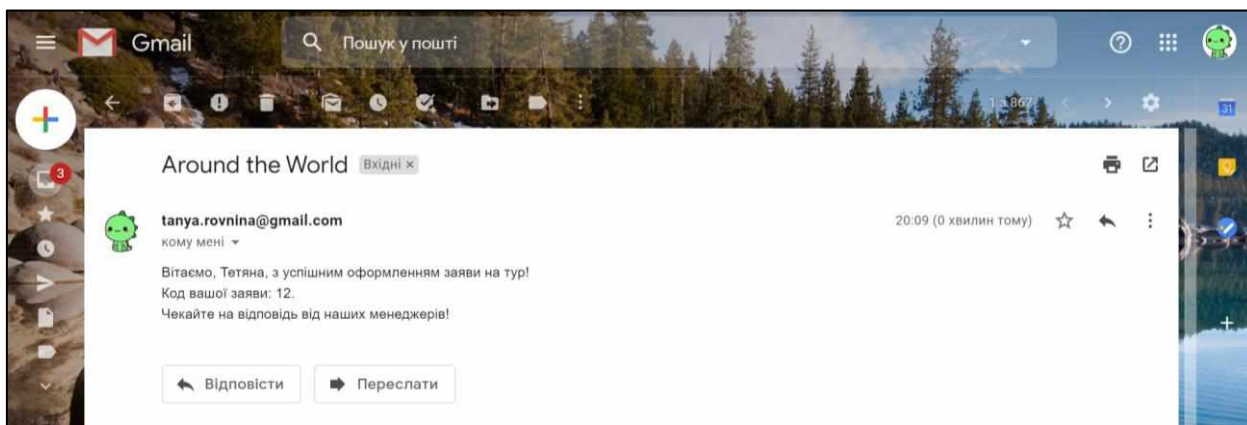


Рисунок 3.15 – Повідомлення про успішне оформлення туру

Якщо у верхньому меню обрати «Готелі», то з'явиться сторінка з короткою інформацією по усім готелям, в які тур агентство пропонує тури. Також вона містить блок фільтрації. Вигляд сторінки зображено на рисунку 3.16.

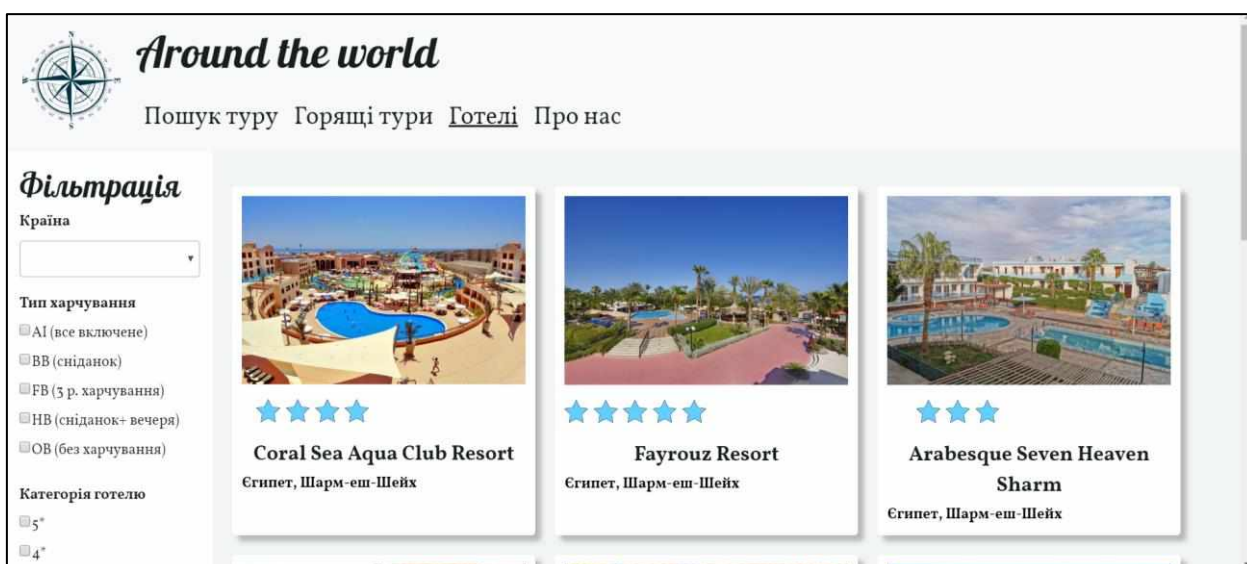


Рисунок 3.16 – Інтерфейс сторінки Готелі

При натисканні на картку готелю відкривається сторінка з детальним описом цього готелю та наявними у цей готель турами. Сторінка схожа на сторінку з описом туру, але має деякі відмінності. На рисунках 3.17 – 3.18 зображено опис одного з готелів.

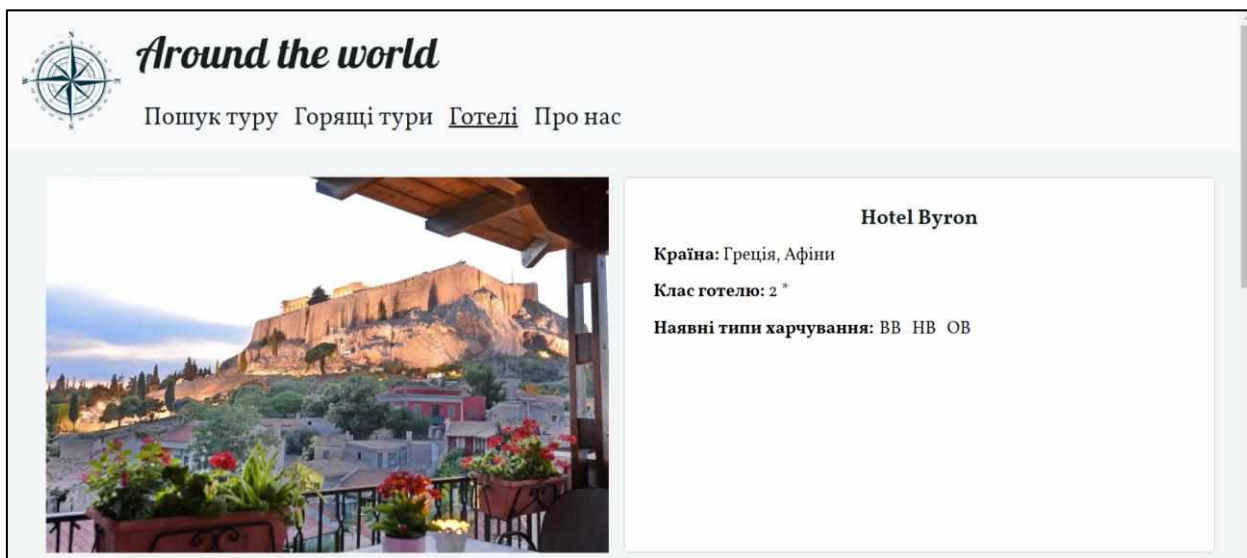


Рисунок 3.17 – Інтерфейс опису готелю. Початок



Рисунок 3.18 – Інтерфейс опису готелю. Продовження

Якщо в таблиці «Доступні тури» натиснути на кнопку «Перегляд» навпроти будь-якого туру, то перейдемо на сторінку з описом цього туру.

Висновки

При виконанні курсової роботи була створена інформаційна веб-система для взаємодії користувачів з туристичним агентством за допомогою інтернету.

Веб-сайт розроблено для одного типу користувачів і може адаптуватись під різну ширину екранів. Дизайн сайту відповідає вимогам мінімальності та містить контент українською мовою. Програма надає інформацію про агентство і містить актуальну інформацію про тури та готелі. Користувач може залишити відгук на готель та оформити тур не виходячи з дому.

Застосунок розроблено на мові програмування JavaScript у середовищі WebStorm. Для створення шаблонів сторінок був використаний шаблонізатор PUG. У даній роботі була використана програмна платформа Node.js. Для реалізації веб-сервера використовувався модуль Express.js. Для розробки інтерфейсу користувача була використана бібліотека JQuery 3.4.1 та набір інструментів Bootstrap 4.4.1. В якості СРБД був обраний MySQL.

Застосунок може бути покращено наступними шляхами:

- розробка системи для персоналу агентства;
- реєстрація користувачів на сайті;
- оповіщення зареєстрованих клієнтів о нових пропозиціях;
- розміщення веб-сайту у хмарі.

Список використаних джерел

1. «Поехали с нами». Веб-сайт туристичного агентства [Електронний ресурс]. – Режим доступу: URL: <https://www.poechalisnami.ua/>. – Назва з екрану.
2. «JoinUp!». Веб-сайт туристичного агентства [Електронний ресурс]. – Режим доступу: URL: <https://joinup.ua/>. – Назва з екрану.
3. «TUI». Веб-сайт туристичного агентства [Електронний ресурс]. – Режим доступу: URL: <https://www.tui.ua/>. – Назва з екрану.
4. «Booking.com». Веб-сайт системи бронювання житла [Електронний ресурс]. – Режим доступу: URL: <https://www.booking.com/>. – Назва з екрану.
5. Веб-сайт компанії JetBrains. WebStorm [Електронний ресурс]. – Режим доступу: URL: <https://www.jetbrains.com/ru-ru/webstorm/>. – Назва з екрану.
6. Веб-сайт бібліотеки «Bootstrap» [Електронний ресурс]. – Режим доступу: URL: <https://getbootstrap.com/>. – Назва з екрану.
7. SQLite vs MySQL vs PostgreSQL: A Comparison Of Relational Database Management Systems [Електронний ресурс]. – Режим доступу: URL: <https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems>. – Назва з екрану.
8. В чем преимущества Node.js? [Електронний ресурс]. – Режим доступу: URL: <https://artjoker.ua/ru/blog/v-chem-preimushchestva-nodejs/>. – Назва з екрану.
9. Дональд А. Норман. Дизайн привычных вещей / Д. А. Норман; [пер. з англійської М. Бакалов] – К.: Клуб Сімейного Дозвілля, 2019. – 320 с.

10. Стівен Круг. Веб дизайн: книга Стива Круга или «не заставляйте меня думать!», 2-е издание / С. Круг; [пер. з англійської Ю. Асотова, В. Овчінникова] – СПб.: Символ-Плюс, 2008. – 224 с.
11. Стівен Круг. Как сделать сайт удобным. Юзабилити по методу Стива Круга / С. Круг – СПб: Пітер – 2010. – 208 с.
12. 10 Usability Heuristics for User Interface Design. [Електронний ресурс]. – Режим доступу: URL: <https://www.nngroup.com/articles/ten-usability-heuristics/>. – Назва з екрану.
13. Матеріали з лекцій курсу «Реляційні бази даних» – основні теоретичні відомості. [Електронний ресурс]. – Режим доступу: URL: <https://distedu.ukma.edu.ua/course/view.php?id=50>. – Назва з екрану.
14. Сайт для побудови діаграм «Draw.io». [Електронний ресурс]. – Режим доступу: URL: <https://www.draw.io/>. – Назва з екрану.

Додаток А. Запити створення БД

Запити для створення БД проекту у системі MySQL і результати їх виконання.

Запит для створення БД:

```
CREATE DATABASE travel;
```

Нижче наведено запит для створення таблиці «comments».

```
CREATE TABLE comments (  
    id_comment int NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    hotel_id int NOT NULL,  
    comment_date datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,  
    user varchar(20) NOT NULL,  
    email varchar(50) NOT NULL,  
    text varchar(1000) DEFAULT NULL,  
    FOREIGN KEY(hotel_id) REFERENCES hotels(id_hotel) ON DELETE CASCADE ON UPDATE  
    CASCADE  
);
```

Результат виконання запиту зображено на рисунку 4.1.

Field	Type	Null	Key	Default	Extra
id_comment	int(11)	NO	PRI	NULL	auto_increment
hotel_id	int(11)	NO	MUL	NULL	
comment_date	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
user	varchar(20)	NO		NULL	
email	varchar(50)	NO		NULL	
text	varchar(1000)	YES		NULL	

Рисунок 4.1 – Таблиця «comments»

Нижче наведено запит для створення таблиці «countries».

```
CREATE TABLE countries (  
    id_country char(3) NOT NULL PRIMARY KEY,  
    name varchar(16) NOT NULL UNIQUE  
);
```

Результат виконання запиту зображено на рисунку 4.2.

Field	Type	Null	Key	Default	Extra
id_country	char(3)	NO	PRI	NULL	
name_country	varchar(16)	NO	UNI	NULL	

Рисунок 4.2 – Таблица «countries»

Нижче наведено запит для створення таблиці «documents».

```
CREATE TABLE documents (
  code int NOT NULL AUTO_INCREMENT PRIMARY KEY ,
  tour_code int NOT NULL,
  food_id char(2) NOT NULL,
  registrDate datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
  adults int NOT NULL,
  children int NOT NULL DEFAULT 0,
  clientSurname varchar(20) NOT NULL,
  clientName varchar(20) NOT NULL,
  phone varchar(13) NOT NULL,
  email varchar(50) NOT NULL,
  confirmed bool NOT NULL,
  price decimal(6,2) NOT NULL,
  notes varchar(225) DEFAULT NULL,
  FOREIGN KEY(tour_code) REFERENCES tours(code) ON UPDATE CASCADE,
  FOREIGN KEY(food_id) REFERENCES food(id_food) ON UPDATE CASCADE
);
```

Результат виконання запиту зображено на рисунку 4.3.

Field	Type	Null	Key	Default	Extra
code	int(11)	NO	PRI	NULL	auto_increment
tour_code	int(11)	NO	MUL	NULL	
food_id	char(2)	NO	MUL	NULL	
registrDate	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
adults	int(11)	NO		NULL	
children	int(11)	NO		0	
clientSurname	varchar(20)	NO		NULL	
clientName	varchar(20)	NO		NULL	
phone	varchar(13)	NO		NULL	
email	varchar(50)	NO		NULL	
confirmed	tinyint(1)	YES		0	
price	decimal(6,2)	NO		NULL	
notes	varchar(225)	YES		NULL	

Рисунок 4.3 – Таблица «documents»

Нижче наведено запит для створення таблиці «food».

```
CREATE TABLE food (
```

```

id_food char(2) NOT NULL PRIMARY KEY,
type_food varchar(20) NOT NULL,
notes varchar(225) DEFAULT NULL
);

```

Результат виконання запиту зображено на рисунку 4.4.

Field	Type	Null	Key	Default	Extra
id_food	char(2)	NO	PRI	NULL	
type_food	varchar(20)	NO		NULL	
notes	varchar(225)	YES		NULL	

Рисунок 4.4 – Таблиця «food»

Нижче наведено запит для створення таблиці «food_availability».

```

CREATE TABLE food_availability(
    food_id char(2) NOT NULL,
    hotel_id int NOT NULL,
    PRIMARY KEY(food_id, hotel_id),
    FOREIGN KEY(food_id) REFERENCES food(id_food) ON UPDATE CASCADE,
    FOREIGN KEY(hotel_id) REFERENCES hotels(id_hotel) ON UPDATE CASCADE
);

```

Результат виконання запиту зображено на рисунку 4.5.

Field	Type	Null	Key	Default	Extra
food_id	char(2)	NO	PRI	NULL	
hotel_id	int(11)	NO	PRI	NULL	

Рисунок 4.5 – Таблиця «food_availability»

Нижче наведено запит для створення таблиці «hotels».

```

CREATE TABLE hotels (
    id_hotel int NOT NULL AUTO_INCREMENT PRIMARY KEY,
    country_id char(3) NOT NULL,
    city varchar(30) NOT NULL,
    name_hotel varchar(20) NOT NULL,
    stars int NOT NULL,
    photo varchar(225) NOT NULL,
    notes varchar(225) DEFAULT NULL,
    FOREIGN KEY(country_id) REFERENCES countries(id_country) ON UPDATE CASCADE
);

```

Результат виконання запиту зображено на рисунку 4.6.

Field	Type	Null	Key	Default	Extra
id_hotel	int(11)	NO	PRI	NULL	auto_increment
country_id	char(3)	NO	MUL	NULL	
name_hotel	varchar(30)	NO		NULL	
city	varchar(30)	NO		NULL	
stars	int(11)	NO		NULL	
photo	varchar(225)	NO		NULL	
description	varchar(1000)	YES		NULL	

Рисунок 4.6 – Таблиця «hotels»

Нижче наведено запит для створення таблиці «operators».

```
CREATE TABLE operators (
  id_operator int NOT NULL AUTO_INCREMENT PRIMARY KEY,
  name_operator varchar(20) NOT NULL,
  phone char(18) NOT NULL,
  email varchar(50) NOT NULL,
  notes varchar(225) DEFAULT NULL
);
```

Результат виконання запиту зображено на рисунку 4.7.

Field	Type	Null	Key	Default	Extra
id_operator	int(11)	NO	PRI	NULL	auto_increment
name_operator	varchar(20)	NO		NULL	
phone	varchar(18)	NO		NULL	
email	varchar(20)	NO		NULL	
notes	varchar(225)	YES		NULL	

Рисунок 4.7 – Таблиця «operators»

Нижче наведено запит для створення таблиці «transfers».

```
CREATE TABLE transfers (
  id_transfer int NOT NULL AUTO_INCREMENT PRIMARY KEY,
  type_transfer varchar(20) NOT NULL,
  notes varchar(225) DEFAULT NULL
);
```

Результат виконання запиту зображено на рисунку 4.8.

Field	Type	Null	Key	Default	Extra
id_transfer	int(11)	NO	PRI	NULL	auto_increment
type_transfer	varchar(20)	NO		NULL	
notes	varchar(225)	YES		NULL	

Рисунок 4.8 – Таблица «transfers»

Нижче наведено запит для створення таблиці «tours».

```
CREATE TABLE tours (
  code int NOT NULL AUTO_INCREMENT PRIMARY KEY,
  operator_id int NOT NULL,
  hotel_id int NOT NULL,
  transfer_id int NOT NULL,
  tourDate date NOT NULL,
  nights int NOT NULL,
  price_new decimal(6,2) DEFAULT NULL,
  price decimal(6,2) NOT NULL,
  isHot bool NOT NULL,
  program varchar(1000) DEFAULT NULL,
  notes varchar(225) DEFAULT NULL,
  FOREIGN KEY(operator_id) REFERENCES operators(id_operator) ON UPDATE CASCADE,
  FOREIGN KEY(hotel_id) REFERENCES hotels(id_hotel) ON UPDATE CASCADE,
  FOREIGN KEY(transfer_id) REFERENCES transfers(id_transfer) ON UPDATE CASCADE
);
```

Результат виконання запиту зображено на рисунку 4.9.

Field	Type	Null	Key	Default	Extra
code	int(11)	NO	PRI	NULL	auto_increment
operator_id	int(11)	NO	MUL	NULL	
hotel_id	int(11)	NO	MUL	NULL	
transfer_id	int(11)	NO	MUL	NULL	
tourDate	date	NO		NULL	
nights	int(11)	NO		NULL	
price_new	decimal(6,2)	YES		NULL	
price	decimal(6,2)	NO		NULL	
isHot	tinyint(1)	NO		NULL	
program	varchar(1000)	YES		NULL	
notes	varchar(225)	YES		NULL	

Рисунок 4.9 – Таблица «tours»

Додаток Б. Код програми

Код з файлу server.js

```
let express = require('express');
let server = express();
server.listen(8888);
console.log('Server is running on port 8888');
server.use(express.static(__dirname));

let bodyParser = require('body-parser');
server.use(bodyParser.urlencoded({extended: false}));
server.use(bodyParser.json());

let mysql = require('mysql');
let con = mysql.createConnection({
  host: "127.0.0.1",
  user: "root",
  password: "Dream@06365",
  insecureAuth: true,
  database: "travel"
});
con.connect(function (err) {
  if (err) throw err;
});

let nodemailer = require("nodemailer");
let transporter = nodemailer.createTransport({
  host: 'smtp.gmail.com',
  port: 587,
  secure: false,
  auth: {
    user: "tanya.rovnina@gmail.com", //testEmailAccount.user,
    pass: "sparrow.com" //testEmailAccount.pass
  }
});

//обробка запиту на отримання головної сторінки
server.get('/', function (req, res) {
  getOperators().then(r => {
    res.render(__dirname + "/views/index.pug", {operators: r});
  });
});

//обробка запиту на отримання сторінки з контактами
server.get('/about', function (req, res) {
  res.render(__dirname + "/views/contacts.pug");
});

//обробка запиту на отримання сторінки з усіма турами
```



```

server.get('/tours', function (req, res) {
  getAllTours().then(tours => {
    getFilters().then(obj => {
      obj.tours = tours;
      obj.category = "tours";
      res.render(__dirname + "/views/tours.pug", obj);
    });
  });
});

//обробка запиту на отримання сторінки з горящими турами
server.get('/hot-tours', function (req, res) {
  getHotTours().then(hot => {
    getFilters().then(obj => {
      obj.tours = hot;
      obj.category = "hot";
      res.render(__dirname + "/views/tours.pug", obj);
    });
  });
});

//обробка запиту на отримання сторінки з готелями
server.get('/hotels', function (req, res) {
  getAllHotels().then(hotel => {
    getFilters().then(filters => {
      res.render(__dirname + "/views/hotels.pug", {
        hotels: hotel,
        countries: filters.countries,
        foods: filters.foods,
        category: "hotels",
      });
    });
  });
});

//обробка запиту на отримання сторінки з описом певного туру
server.get('/api/tour', function (req, res) {
  getTour(req.query.id).then(tour => {
    res.render(__dirname + "/views/tour-view.pug", tour);
  });
});

//обробка запиту на отримання сторінки з описом певного готелю
server.get('/api/hotel', function (req, res) {
  getHotel(req.query.id).then(hotel => {
    getHotelTours(req.query.id).then(t=>{
      hotel.tours = t;
      res.render(__dirname + "/views/hotel-view.pug", hotel);
    });
  });
});

```

```

    });
});

//обробка запиту на фільтрування турів/готелів
server.get('/filters', function (req, res) {
    let filters = {
        country: req.query.country,
        food: req.query.food,
        stars: req.query.stars,
        dateFrom: req.query.dateFrom,
        dateTo: req.query.dateTo,
        transfer: req.query.transfer,
        operator: req.query.operator
    }

    if (req.query.category === "hotels") {
        getAllHotels(filters).then(hotel => {
            getFilters().then(filters => {
                res.render(__dirname + "/views/hotels.pug", {
                    hotels: hotel,
                    countries: filters.countries,
                    foods: filters.foods,
                    category: "hotels",
                });
            });
        });
    } else if (req.query.category === "hot") {
        getHotTours(filters).then(hot => {
            getFilters().then(obj => {
                obj.tours = hot;
                obj.category = "hot";
                res.render(__dirname + "/views/tours.pug", obj);
            });
        });
    } else {
        getAllTours(filters).then(tours => {
            getFilters().then(obj => {
                obj.tours = tours;
                obj.category = "tours";
                res.render(__dirname + "/views/tours.pug", obj);
            });
        });
    }
});

//обробка запиту на оформлення туру
server.post('/buyTour', function (req, res) {
    let notes = "";
    if (req.body.notes !== undefined)
        notes = req.body.notes;

```

```

    let client = {
      surname: req.body.surname,
      name: req.body.name,
      email: req.body.email,
      phone: req.body.phone,
      food: req.body.food,
      adults: req.body.adults,
      children: req.body.children,
      notes: notes
    }

    buyTour(req.query.id, client).then(doc => sendMessage(doc, client));
    res.render(__dirname + "/views/success.pug");
  });

  //обробка запиту на додавання вігуку
  server.post('/comment', function (req, res) {
    let comment = JSON.parse(req.body.comment);
    addComment(comment).then(comment =>
      res.render(__dirname + "/views/comments.pug", {comments: comment})
    );
  });

  //отримати туроператорів
  function getOperators() {
    return new Promise((resolve, reject) => {
      let sql = "SELECT * FROM operators";
      con.query(sql, function (err, res) {
        if (err) throw err;
        resolve(res);
      });
    });
  }

  //отримати коментарі
  function getComments(hotel) {
    return new Promise((resolve, reject) => {
      let sql = "SELECT * " +
        "FROM comments " +
        "WHERE hotel_id = " + hotel + " " +
        "ORDER BY comment_date DESC";
      con.query(sql, function (err, comment) {
        if (err) throw err;

        comment.forEach(c => {
          c.comment_date = parseDate(new Date(c.comment_date), false);
        });
        resolve(comment);
      });
    });
  }

```

```

        });
    });
}

// отримати готель за айді
function getHotel(id) {
    return new Promise((resolve, reject) => {
        let sql = "SELECT * " +
            "FROM hotels INNER JOIN countries ON hotels.country_id = " +
            "countries.id_country " +
            "WHERE id_hotel = " + id;
        con.query(sql, function (err, hotel) {
            if (err) throw err;
            sql = "SELECT * " +
                "FROM food_availability INNER JOIN food ON " +
                "food_availability.food_id = food.id_food " +
                "WHERE hotel_id = " + id;
            con.query(sql, function (err, food) {
                if (err) throw err;

                getComments(id).then(comment =>
                    resolve({hotel: hotel[0], comments: comment, foods: food})
                );
            });
        });
    });
}

// отримати тур за айді
function getTour(id) {
    return new Promise((resolve, reject) => {
        let sql = "SELECT * " +
            "FROM tours INNER JOIN transfers ON transfer_id = id_transfer " +
            "WHERE code = " + id;
        con.query(sql, function (err, tour) {
            if (err) throw err;
            tour[0].tourDate = parseDate(new Date(tour[0].tourDate), true);

            getHotel(tour[0].hotel_id).then(obj => {
                obj.tour = tour[0];
                resolve(obj);
            });
        });
    });
}

// отримати тури до певного готелю
function getHotelTours(hotel) {
    return new Promise((resolve, reject) => {
        getAllTours().then(res => {
            let tours = new Array();

```

```

        res.forEach(t => {
            if (t.hotel_id == hotel)
                tours.push(t);
        });

        resolve(tours);
    });
});
}

// отримати всі горящі тури
function getHotTours(filters) {
    let sql = queryTours(filters) + " AND isHot = true ";

    return new Promise((resolve, reject) => {
        con.query(sql, function (err, result) {
            if (err) throw err;
            result.forEach(tour => {
                tour.tourDate = parseDate(new Date(tour.tourDate), false);
            });

            resolve(result);
        });
    });
}

// отримати всі тури
function getAllTours(filters) {
    let sql = queryTours(filters);

    return new Promise((resolve, reject) => {
        con.query(sql, function (err, result) {
            if (err) throw err;
            result.forEach(tour => {
                tour.tourDate = parseDate(new Date(tour.tourDate), false);
            });

            resolve(result);
        });
    });
}

// отримати всі готелі з можливою фільтрацією
function getAllHotels(filters) {
    let sql = queryHotels(filters);

    return new Promise((resolve, reject) => {
        con.query(sql, function (err, result) {
            if (err) throw err;
            resolve(result);
        });
    });
}

```

```

    });
  });
}

//запит для отримання турів
function queryTours(filters) {
  let now = new Date();
  let today = now.getFullYear() + "-" + (now.getMonth() + 1) + "-" +
now.getDate();

  let sql = "SELECT * " +
    "FROM (((tours T INNER JOIN hotels Ht ON T.hotel_id = Ht.id_hotel) " +
    "INNER JOIN countries Ct ON Ht.country_id = Ct.id_country) " +
    "INNER JOIN transfers Tr ON transfer_id = Tr.id_transfer) " +
    "INNER JOIN operators O ON operator_id = O.id_operator ";

  if (filters) {
    sql += "WHERE T.hotel_id IN (SELECT Th.id_hotel " +
      "FROM (" + queryHotels(filters) + ") Th)";
    if (filters.operator != "")
      sql += " AND operator_id = " + filters.operator;
    if (filters.transfer != "")
      sql += " AND transfer_id = " + filters.transfer;
    if (filters.dateFrom != "")
      sql += " AND tourDate >= '" + filters.dateFrom + "'";
    else
      sql += " AND tourDate > '" + today + "'";
    if (filters.dateTo != "")
      sql += " AND tourDate <= '" + filters.dateTo + "'";
  } else
    sql += "WHERE tourDate > '" + today + "'";

  return sql;
}

//запит для отримання готелів
function queryHotels(filters) {
  let sql = "SELECT * " +
    "FROM hotels H INNER JOIN countries C ON H.country_id = C.id_country";
  if (filters) {
    if (filters.country != "" || filters.stars != "" || filters.food != "")
      sql += " WHERE ";
    if (filters.country != "")
      sql += "H.country_id = '" + filters.country + "'";
    if (filters.country != "" && filters.stars != "")
      sql += " AND ";
    if (filters.stars != "")
      sql += "stars IN (" + filters.stars + ")";
    if ((filters.country != "" || filters.stars != "") && filters.food !=
    "")
      sql += " AND "

```

```

        if (filters.food !== "")
            sql += "H.id_hotel IN (SELECT F.hotel_id " +
                "FROM food_availability F " +
                "WHERE F.food_id IN (" + filters.food + "))";
    }

    return sql;
}

// отримати всі фільтри
function getFilters() {
    let filters = {};
    return new Promise((resolve, reject) => {
        let sql = "SELECT * FROM countries";
        con.query(sql, function (err, res1) {
            if (err) throw err;
            filters.countries = res1;

            sql = "SELECT * FROM food";
            con.query(sql, function (err, res2) {
                if (err) throw err;
                filters.foods = res2;

                sql = "SELECT * FROM transfers";
                con.query(sql, function (err, res3) {
                    if (err) throw err;
                    filters.transfers = res3;

                    getOperators().then(res4 => {
                        filters.operators = res4;
                        resolve(filters);
                    });
                });
            });
        });
    });
}

// перетворити дату у формат дд-мм-рррр
function parseDate(date, txt) {
    let newDate = "";
    newDate += date.getDate() + " ";

    if (txt) {
        switch (date.getMonth()) {
            case 0:
                newDate += " Січня ";
                break;
            case 1:
                newDate += " Лютого ";

```

```

        break;
    case 2:
        newDate += " Березня ";
        break;
    case 3:
        newDate += " Квітня ";
        break;
    case 4:
        newDate += " Травня ";
        break;
    case 5:
        newDate += " Червня ";
        break;
    case 6:
        newDate += " Липня ";
        break;
    case 7:
        newDate += " Серпня ";
        break;
    case 8:
        newDate += " Вересня ";
        break;
    case 9:
        newDate += " Жовтня ";
        break;
    case 10:
        newDate += " Листопада ";
        break;
    case 11:
        newDate += " Грудня ";
        break;

    default:
        break;
}
} else {
    let m = (date.getMonth() + 1)
    if (m < 10)
        newDate += "- 0" + m + " - ";
    else
        newDate += "- " + m + " - ";
}

newDate += date.getFullYear();

return newDate;
}

//придбати тур
function buyTour(id, client) {

```



```

    return new Promise((resolve, reject) => {
        let sql = "SELECT price " +
            "FROM tours " +
            "WHERE code = " + id;
        con.query(sql, function (err, res) {
            if (err) throw err;
            let price = res[0].price;

            sql = "INSERT INTO documents " +
                "(tour_code, food_id, email, clientSurname, clientName, phone,
adults, children, notes, price) " +
                "VALUES ('" + id + "', '" + client.food + "', '" + client.email
+ "', '" + client.surname + "', '"
                + client.name + "', '" + client.phone + "', '" + client.adults
+ "', '"
                + client.children + "', '" + client.notes + "', '" + price +
"')";

            con.query(sql, function (err, result) {
                if (err) throw err;
                resolve(result.insertId);
            });
        });
    });
}

//додати відгук до готелю
function addComment(comment) {
    return new Promise((resolve, reject) => {
        let sql = "INSERT INTO comments " +
            "(hotel_id, email, user, text) " +
            "VALUES ('" + comment.hotel + "', '" + comment.email + "', '" +
comment.user + "', '" + comment.text + "')";
        con.query(sql, function (err, result) {
            if (err) throw err;

            getComments(comment.hotel).then(res =>
                resolve(res)
            );
        });
    });
}

//відправити повідомлення користувачу
async function sendMessage(docId, client) {
    transporter.sendMail({
        from: "tanya.rovnina@gmail.com",
        to: client.email,
        subject: "Around the World",
        text: "Вітаємо, " + client.name + ", з успішним оформленням заяви на
тур! " +
            "\nКод вашої заяви: " + docId + ".\n" +

```

```

        "Чекайте на відповідь від наших менеджерів!",
    });
}

```

Код з файлу client.js

```

window.jQuery = $;
window.$ = $;

```

```

//відкрити опис певного туру
$(document).on('click', '.tour_card .tour-img, .tour_card .tour-title',
function () {
    let $this = $(this);
    let id = $this.closest('.tour_card').data('tour-id');

    changeContent('/api/tour?id=' + id);
});

```

```

//опис туру за готелем
$(document).on('click', '.tour-view', function () {
    let $this = $(this);
    let id = $this.data('tour-id');

    changeContent('/api/tour?id=' + id);
});

```

```

//відкрити опис певного готелю
$(document).on('click', '.hotel_card .hotel-img, .hotel_card .hotel-title',
function () {
    let $this = $(this);
    let id = $this.closest('.hotel_card').data('hotel-id');

    changeContent('/api/hotel?id=' + id);
});

```

```

//відкрити форму для відгуку
$(document).on('click', '#open-comment', function () {
    let form = $('#comment-form');
    if (form.hasClass('hidden'))
        form.removeClass('hidden');
    else
        form.addClass('hidden');
});

```

```

//відкрити форму з заявою
$(document).on('click', '#open-statement', function () {
    $('#document').removeClass('hidden');

});

//закрити форму
$(document).on('click', '.close_img', function () {
    $('#document').addClass("hidden");
});

//показати всі тури
$(document).on('click', '#href_tours', function () {
    clearSelect();
    $('#href_tours').addClass("active");

    changeContent('/tours');
});

//показати горящі тури
$(document).on('click', '#href_hot', function () {
    clearSelect();
    $('#href_hot').addClass("active");

    changeContent('/hot-tours');
});

//показати всі готелі
$(document).on('click', '#href_hotels', function () {
    clearSelect();
    $('#href_hotels').addClass("active");

    changeContent('/hotels');
});

//інформація про агенство
$(document).on('click', '#href_about', function () {
    clearSelect();
    $('#href_about').addClass("active");

    changeContent('/about');
});

// замінити основний блок інформацією з сервера за посиланням
function changeContent(url) {

```

```

$.ajax({
    url: url,
    method: 'get',
    dataType: 'json',
    error: function (xhr) {
        $('#main_block').empty();
        $('#main_block').append(xhr.responseText);
        $('#page1').removeClass('hidden');

        //встановити мінімальну дату
        $('#f-dateFrom').attr('min', minDate());
        $('#f-dateTo').attr('min', minDate());

        //повернути скрол нагору
        document.body.scrollTop = 0;
        document.documentElement.scrollTop = 0;
    },
});
}

// прибрати підкреслення з пунктів меню
function clearSelect() {
    let links = $('.nav-link');
    for (let i = 0; i < links.length; i++) {
        let id = links[i].id;
        $('##' + id).removeClass("active");
    }
}

// Додати коментар до готелю
$(document).on('click', '#add-comment', function () {
    let $this = $(this);
    let hotel = $this.data('hotel-id');

    let user = $('#user').val();
    let email = $('#user-email').val();
    let text = $('#comment').val();

    let comment = {
        "hotel": hotel,
        "user": user,
        "email": email,
        "text": text
    }

    $.ajax({
        url: '/comment',
        method: 'post',
        data: {comment: JSON.stringify(comment)},
    });
});

```

```

        dataType: 'json',
        error: function (xhr) {
            $('#comments-block').empty();
            $('#comments-block').append(xhr.responseText);
            $('#page1').removeClass('hidden');
            $('#comment-form').addClass('hidden');
        },
    });
});

// прибрати фільтри
$(document).on('click', '#clear-filters', function () {
    let $this = $(this);
    let category = $this.closest('.filters').data('filter-id');
    ;

    if (category === "tours")
        $('#href_tours').click();
    else if (category === "hot")
        $('#href_hot').click();
    else if (category === "hotels")
        $('#href_hotels').click();
});

// застосувати фільтри
$(document).on('click', '#use-filters', function () {
    let $this = $(this);
    let category = $this.closest('.filters').data('filter-id');
    ;

    let country = $('#filter-country').val();

    let food = "";
    let checkboxes = document.getElementsByName('food');
    for (var i = 0; i < checkboxes.length; i++) {
        if (checkboxes[i].checked)
            food += "" + checkboxes[i].value + ",";
    }
    food = food.substring(0, food.length - 1);

    let stars = "";
    checkboxes = document.getElementsByName('hotel');
    for (var i = 0; i < checkboxes.length; i++) {
        if (checkboxes[i].checked)
            stars += checkboxes[i].value + ",";
    }
    stars = stars.substring(0, stars.length - 1);

```

```

    let dateFrom, dateTo, transfer, operator = "";
    if (category != "hotels") {
        dateFrom = $('#f-dateFrom').val();
        dateTo = $('#f-dateTo').val();
        transfer = $('#filter-transfer').val();
        operator = $('#filter-operator').val();
    }

    let url = "/filters?category=" + category + "&" +
        "country=" + country + "&food=" + food + "&stars=" + stars + "&" +
        "dateFrom=" + dateFrom + "&dateTo=" + dateTo + "&" +
        "transfer=" + transfer + "&operator=" + operator;

    changeContent(url);
});

//пагінація коментарів
$(document).on('click', '.page-link', function () {
    let $this = $(this);
    let page = $this.data('page');

    hideComments();
    $('#page' + page).removeClass("hidden");
});

//сховати сторінки з коментарями
function hideComments() {
    let links = $('.comments-page');
    for (let i = 0; i < links.length; i++) {
        let id = links[i].id;
        $('##' + id).addClass("hidden");
    }
}

//встановити мінімальну дату
function minDate() {
    var today = new Date();
    var dd = today.getDate();
    var mm = today.getMonth()+1; //January is 0!
    var yyyy = today.getFullYear();
    if(dd<10){
        dd='0'+dd
    }
    if(mm<10){
        mm='0'+mm
    }
    today = yyyy+'-'+mm+'-'+dd;
    return today;
}

```

```
}
```

Код з файлу style.css

```
body {  
    margin: 0;  
    background-color: #f0f5f5;  
    font-family: Vollkorn;  
    font-size: 15px;  
    color: black;  
}  
  
.navbar-light .navbar-nav .nav-link {  
    font-size: 25px;  
    color: #141f1f;  
}  
  
.nav-link, .page-link{  
    cursor: pointer;  
}  
  
.nav-link.active {  
    text-decoration: underline;  
}  
  
.logo {  
    width: 100px;  
    height: 100px;  
}  
  
.title {  
    color: #141f1f;  
    font-family: Pattaya;  
    font-size: 45px;  
    margin-right: 2em;  
}  
  
/*основний блок*/  
.content {  
    min-height: 68vh;  
    padding-bottom: 50px;  
}  
  
footer {  
    background: white;  
    font-size: 17px;  
    left: 0;  
    bottom: 0;
```

```

padding: 10px;
width: 100%;
}

/*інформація про компанію*/
.information, .map {
display: inline-block;
vertical-align: top;
margin-bottom: 10px;
}

.map {
border: 1px solid black;
width: 100%;
height: 300px;
margin-top: 1%;
margin-left: 2%;
}

.information {
width: 40%;
margin: 1% 3%;
}

.information a {
color: black;
text-decoration: none;
}

/*шаринг в соцмережах*/
.fa {
display: inline-block;
width: 40px;
margin: 2%;
}

.fa img {
width: 60%;
}

/* головна сторінка */
#operators-block{
padding: 3% 0;
}

#home h3{
text-align: center;
margin: 2%;
font-size: 42px;
}

```



```

}

#home table{
    margin: 1% 3%;
}

#home .icon{
    width: 10%;
}

#home .txt{
    width: 30%;
    font-size: 16px;
}

.carousel-block{
    position: relative;
    margin: 0 10%;
    overflow: hidden;
}

.carousel-item h4{
    text-align: center;
}

.carousel-item span{
    margin: 1%;
}

/*Фільтрація*/
#filter-nav {
    height: 100%;
    width: 100%;
    top: 0;
    left: 0;
    padding: 1%;
    background: white;
    color: #141f1f;
}

#filter-nav .title {
    font-size: 2.5vw;
    margin: 0;
}

#filter-nav .f-title {
    font-weight: bold;
}

```

```

#filter-nav button{
    width: 100%;
    margin: 1%;
}

#tours_block, .short-inf, #hotels_block {
    margin: 0;
}

/* каптка тпы */
#tours, #hotels, #description_block, .description-card, #comments-block,
#description_block form {
    padding: 2%;
}

.tour_card, .hotel_card {
    width: 31%;
    cursor: pointer;
    border: none;
    margin: 1%;
    padding: 1%;
    box-shadow: 0.4em 0.4em 5px rgba(122, 122, 122, 0.5);
}

.tour_img, #home_img, .hotel_img, #success_img {
    width: 100%;
    object-fit: contain;
    vertical-align: top;
}

.block-row {
    display: inline-block;
}

.stars {
    display: inline-block;
    width: 60%;
    padding: 4%;
    text-align: center;
}

.stars_img {
    width: 18%;
    margin: 1%;
}

.price {
    text-align: center;
    float: right;
    margin-top: 1%;
}

```

```

        padding: 0% 8%;
        background-color: aquamarine;
    }

    .price span {
        font-weight: bold;
        font-size: 20px;
    }

    .tour-title, .hotel-title {
        text-align: center;
        font-weight: bold;
        font-size: 21px;
        color: #141f1f;
        margin: 1%;
    }

    .tour-dates {
        font-weight: bold;
        font-size: 18px;
    }

    .date_info {
        position: absolute;
        background-color: rgba(127, 255, 212, 0.5);
        text-align: center;
        padding: 1%;
    }

    .old {
        text-decoration: line-through;
    }

    .new {
        color: red;
    }

    /* опис typu */
    #description_block {
        font-size: 18px;
    }

    #description_block .tour-title{
        margin-top: 3%;
        font-size: 42px;
    }

    .short-inf span {
        width: 100%;
    }

```

```

        margin: 1%;
    }

    .short-inf a {
        color: white;
        text-decoration: none;
    }

    .price-inf {
        text-align: center;
        font-size: 40px;
        font-weight: bold;
    }

    .description {
        padding: 3%;
    }

    /* комментарии */
    #comment-form, .comment {
        border: #141f1f 1px solid;
    }

    #comment-form{
        padding: 2%;
    }

    .hidden {
        display: none;
    }

    #open-comment {
        float: right;
        font-size: 20px;
    }

    #comment-form button {
        width: 30%;
    }

    .comment-title {
        margin-top: 3%;
        text-align: center;
    }

    .comment {
        width: 80%;
        margin-bottom: 1%;
    }

```

```

.comment-user {
    width: 50%;
    padding: 1%;
    background: lightsteelblue;
}

.comment-date {
    text-align: center;
    padding: 1%;
    width: 50%;
    background: lightgrey;
}

.comment-txt {
    margin: 0;
    padding: 2%;
    background: white;
}

.comment-head {
    margin: 0;
}

/* форма для оформления туру */
#document {
    z-index: 900;
    width: 100%;
    height: 100%;
    position: fixed;
    top: 0;
    left: 0;
    background: rgba(105, 114, 123, 0.7)
}

#document form {
    border: 1px solid grey;
    background: #d9d9d9;
    border-radius: 10px;
    position: relative;
    max-height: 85vh;
    overflow: auto;
    font-size: 18px;
    margin: 3% 10%;
    padding: 2% 3%;
}

#document h3{
    margin-bottom: 3%;
    text-align: center;
    font-weight: bold;
}

```

```

        color: darkblue;
    }

    .close_img {
        position: absolute;
        top: -2px;
        right: 4px;
        width: 4em;
        cursor: pointer;
    }

    #success .message{
        margin: 4%;
        text-align: center;
    }

    /* таблиця з турами у готеля */
    .table-tours{
        margin: 2%;
        text-align: center;
    }

    .table-tours .head{
        background: #2c75d2;
    }

    .table-tours td, .table-tours th{
        padding: 1%;
        width: 8%;
        border: #141f1f 1px solid;
    }

    /*адаптивність*/
    @media only screen and (max-width: 800px) {
        .tour_card, .hotel_card, .comment{
            width: 100%;
        }
    }

    @media only screen and (max-width: 764px) {
        #filter-nav .title {
            font-size: 32px;
        }
    }
}

```

Код з файлу index.pug

```
doctype html
```

```

html(lang='uk')
  include header.pug
  body
    include navbar.pug
    #main_block.content
      #home
        img(src='../public/images/panorama.jpg', alt='panorama')
        #operators-block.carousel.slide.carousel-multi-item(data-
ride='carousel')
          h3 Туроператори
          // стрілки контролю
          a.carousel-control-prev(href='#operators-block',
role='button', data-slide='prev')
            img.carousel-control-prev-
icon(src='../public/images/left.png', alt='Previous')
          a.carousel-control-next(href='#operators-block',
role='button', data-slide='next')
            img.carousel-control-next-
icon(src='../public/images/right.png', alt='Next')
          // картки туроператорів
          .carousel-block(role='listbox')
            // перший слайд
            .carousel-item.active
              - var i = 0;
              while i < operators.length && i < 3
                .col-md-4(style='float:left')
                  .card.p-3
                    h4.card-title=
operators[i].name_operator
                    hr
                    span
                      b Email:
                      = " " + operators[i].email
                    span
                      b Телефон:
                      = " " + operators[i].phone
              - i++
            // інші слайди
            while i < operators.length
              .carousel-item
                -var j = 0;
                while i < operators.length && j < 3
                  .col-md-4(style='float:left')
                    .card.p-3
                      h4.card-title=
operators[i].name_operator
                      hr
                      span
                        b Email:
                        = " " + operators[i].email
                      span
                        b Телефон:

```

```

= " " + operators[i].phone
- j++
- i++
div
  h3 Чому ми?
  table
    tr
      td.icon
        img(src='../public/images/24_7.png',
alt='24/7')
      td.txt
        p Дбаємо про клієнтів 24/7. Якщо у вас виникли
питання або складності в подорожі, ми завжди поруч!
      td.icon
        img(src='../public/images/world.png',
alt='world')
      td.txt
        p Пропонуємо тури майже в кожний куточок світу.
Всі можливі варіанти відпочинку можна підібрати у нас!
    tr
      td.icon
        img(src='../public/images/diagram.png',
alt='diagram')
      td.txt
        p У компанії немає боргів, тому туристи
гарантовано отримують відпочинок, за який заплатили.
      td.icon
        img(src='../public/images/price.png',
alt='price')
      td.txt
        p На сайті завжди відображаються актуальні
ціни.
  include footer.pug
  include scripts.pug

```

Код з файлу navbar.pug

```

// верхня панель з меню, яка завжди закріплена вгорі
nav.navbar.navbar-expand-lg.navbar-light.bg-light
  a.navbar-brand(href='/')
    img.logo(src='../public/images/logo.png', alt='Logo')
  div
    a.navbar-brand(href='/')
      span.title Around the world
      button.navbar-toggler(type='button', data-toggle='collapse', data-
target='#navbarNavAltMarkup', aria-controls='navbarNavAltMarkup', aria-
expanded='false', aria-label='Toggle navigation')
      span.navbar-toggler-icon
    #navbarNavAltMarkup.collapse.navbar-collapse
    br

```



```

ul.navbar-nav.mr-auto
  li#href_tours.nav-link Пошук туру
    span.sr-only(current)
  li#href_hot.nav-link Горячі тури
  li#href_hotels.nav-link Готелі
  li#href_about.nav-link Про нас

```

Код з файлу scripts.pug

```

// підключення javaScript
script(src='../node_modules/jquery/dist/jquery.min.js')
script(src='../node_modules/bootstrap/dist/js/bootstrap.min.js')
script(src='../client.js')

```

Код з файлу success.pug

```

doctype html
html(lang='uk,en')
  include header.pug
  body
    #success.content
      h1.message Дякуємо за те, що обрали наше турагенство! Очікуйте
відповіді від нас на пошту!
      img(src='../public/images/plane.png', alt='plane')
      include footer.pug
      include scripts.pug

```

Код з файлу tour-view.pug

```

#description_block
  div.row.short-inf
    img.col-md-6(src= hotel.photo, alt='hotel photo')
    div.card.col-md-6.description-card(data-tour-id= tour.code)
      span
        b Проїзд:
        = " " + tour.type_transfer
      span
        b Дата:
        = " " + tour.tourDate
      span
        b Кількість ночей:
        = " " + tour.nights
      span

```

```

        b Клас готелю:
        = " " + hotel.stars + " *"

    span
        b Наявні типи харчування:
        for food in foods
            span= food.id_food

    if tour.price_new
        span.price-inf= tour.price_new + " грн"
    else
        span.price-inf= tour.price + " грн"

    small ціна за 1 людину. *(початкова ціна)

    button#open-statement.btn.btn-primary Оформити заяву
div
    h3.tour-title= hotel.name_country + ", " + hotel.city+",
    "+hotel.name_hotel
    p.description= hotel.description
    if tour.program
        h3.comment-title Програма туру
        p.description= tour.program

    button#open-comment.btn.btn-dark Залишити відгук
    #comment-form.hidden
        .form-row
            .form-group.col-md-4
                label(for='user-email') Email *
                input#user-email.form-control(type='email',
placeholder='Email')
            .form-group.col-md-4
                label(for='user') Ім'я *
                input#user.form-control(type='text', placeholder='Ім'я')
        .form-group
            label(for='comment') Коментар *
            textarea#comment.form-control(rows='3', cols='45',
placeholder='Текст...')
            button#add-comment.btn.btn-primary(data-hotel-id=tour.hotel_id)
Залишити

    h3.comment-title Коментарі
    include comments.pug
include document.pug

```

Код з файлу tours.pug

```

#tours_block.row
    include filters.pug

```

```

#tours.row.col-md-10.col-sm-8
for tour in tours
  .tour_card.card(data-tour-id= tour.code)
    .date_info
      span.tour-dates= tour.tourDate
      br
      span= "Ночей " + tour.nights
    img.img-fluid.tour-img(src=tour.photo, alt='photo')
  div.block-row
    .stars.row
      - var n = 0;
      while n < tour.stars
        img(src='../public/images/star.png', alt='*')
        - n++
    .price
      if tour.price_new
        span.old= tour.price + " грн"
        br
        span.new= tour.price_new + " грн"
      else
        span= tour.price + " грн"
      br
      small ціна за 1 людину
    span.tour-title= tour.name_hotel
    span
      b= tour.name_country + ", "+ tour.city

```

Код з файлу comments.pug

```

#comments-block
  - var n = 0;
  - var id = 1;
  - var len = comments.length / 5 + 1;
  while n < comments.length
    - var j = 0;
    - var page = 'page'+ id++;
    div.comments-page.hidden(id=page)
      while j < 5 && n < comments.length
        div.comment
          div.row.comment-head
            span.comment-user= comments[n].user
            span.comment-date= comments[n].comment_date
            p.comment-txt= comments[n].text
          - j++
          - n++

nav
  ul.pagination.justify-content-center

```

```

- var i = 1;
while i < len
  li.page-item.page-link(data-page=i)= i
- i++;

```

Код з файлу contacts.pug

```

#contacts.content.container.text-center.text-md-left.row
.col-md-5.col-lg-4.text-center.text-md-left.mb-3.mb-md-0.information
h5.font-weight-bold.text-uppercase.mt-3.mb-4 Контакти
p
  b Телефон:
  br
  | +38 (077)-627-87-80
p
  b Адреса:
  br
  | м. Київ, вул. Григорія Сковороди 2
p
  b Email:
  br
  a(href='mailto:tanya.rovnina@gmail.com') tanya.rovnina@gmail.com
  // шаринг в соцмережах
div

a.fa(href='https://www.facebook.com/profile.php?id=100015201939759',
target='_blank')
  img(src='../public/images/fb.png', alt='facebook')
a.fa(href='https://www.instagram.com/sleep.walker66/',
target='_blank')
  img(src='../public/images/insta.png', alt='instagram')
a.fa(href='https://t.me/sleep_walker11', target='_blank')
  img(src='../public/images/telegram.png', alt='telegram')
  // карта
.col-md-7.col-lg-6.col-xl-6

iframe.map(src='https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d2539.74
01147007763!2d30.51779931538038!3d50.464563979477305!2m3!1f0!2f0!3f0!3m2!1i10
24!2i768!4f13.1!3m3!1m2!1s0x40d4ce3f8783f067%3A0x20e77fb079389395!2z0J3QsNGG0
ZbQvtC90LDQu9GM0L3QuNC5INGD0L3RltCy0LXRgNGB0LjRgtC10YIgItCa0LjRlNCy0L4t0JzQvt
Cz0LjQu9GP0L3RgdGM0LrQsCDQsNC60LDQtNC10LzRltGPIg!5e0!3m2!1suk!2sua!4v15422825
14758', allowfullscreen='')
  | </div>
  | </div>

```

Код з файлу document.pug

```
// сторінка для заповнення даних на тур
#document.hidden
  form(name='docForm', action='/buyTour?id='+tour.code, method='post')
    h3 Заповніть форму
    img.close_img(src='../public/images/close.png', alt='close')
    .form-row
      .form-group.col-md-4
        label(for='surname') Прізвище *
        input#surname.form-control(type='text', name='surname',
placeholder='Іванов', required)
      .form-group.col-md-1
      .form-group.col-md-4
        label(for='name') Ім'я *
        input#name.form-control(type='text', name='name',
placeholder='Іван', required)
      .form-row
      .form-group.col-md-4
        label(for='email') Email *
        input#email.form-control(type='email', name='email',
pattern='[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}$',
placeholder='ivanov@gmail.com', required)
      .form-group.col-md-1
      .form-group.col-md-4
        label(for='phone') Телефон *
        input#phone.form-control(type='tel', name='phone',
pattern='[0-9]{3}-[0-9]{3}-[0-9]{2}-[0-9]{2}', placeholder='099-657-98-90',
required)
      .form-row
      .form-group.col-md-3
        label(for='food') Тип їжі*
        select#food.form-control(name='food', required)
        for f in foods
          option(value=f.id_food)= f.id_food + " (" + f.type_food
+ ")"
      .form-group.col-md-1
      .form-group.col-md-3
        label(for='adults') Кількість дорослих *
        input#adults.form-control(type='number', name='adults',
min='1', required)
      .form-group.col-md-1
      .form-group.col-md-3
        label(for='children') Кількість дітей *
        input#children.form-control(type='number', min='0',
name='children', required)
      .form-group.col-md-11
        label(for='notes') Коментар
        textarea#notes.form-control(rows='4', cols='45',
placeholder='Текст...')

    button.btn.btn-primary(type='submit') Оформити
```

Код з файлу filters.pug

```
// Фільтрація
#filter-nav.col-md-2.col-xs-12.col-sm-4
  span.title Фільтрація
  div.filters(data-filter-id=category)
    .form-group
      label.f-title(for='filter-country') Країна
      select#filter-country.form-control
        option(value='', selected='')
        for country in countries
          option(value=country.id_country)= country.name_country
    if transfers
      .form-group
        label.f-title(for='filter-transfer') Трансфер
        select#filter-transfer.form-control
          option(value='', selected='')
          for transfer in transfers
            option(value=transfer.id_transfer)=
transfer.type_transfer
    if tours
      .form-group
        label.f-title(for='filter-date') Оберіть дати
        #filter-date
          div
            label.f-title(for='f-dateFrom') Від:
            br
            input#f-dateFrom.form-control(type='date',
name='calendar')
          div
            label.f-title(for='f-dateTo') До:
            br
            input#f-dateTo.form-control(type='date',
name='calendar')
        if operators
          .form-group
            label.f-title(for='filter-operator') Туроператор
            select#filter-operator.form-control
              option(value='', selected='')
              for operator in operators
                option(value=operator.id_operator)=
operator.name_operator
          .form-group
            label.f-title(for='filter-food') Тип харчування
            #filter-food
              for food in foods
                input#food.id_food(type='checkbox',
value=food.id_food, name='food',
```

```

        label(for=food.id_food)=      food.id_food  +  "  ("  +
food.type_food + ") "
        br
    .form-group
        label.f-title(for='filter-hotel') Категорія готелю
        #filter-hotel
            input#hotel1(type='checkbox', name='hotel', value='5')
            label(for='hotel1')  5*
            br
            input#hotel2(type='checkbox', name='hotel', value='4')
            label(for='hotel2')  4*
            br
            input#hotel3(type='checkbox', name='hotel', value='3')
            label(for='hotel3')  3*
            br
            input#hotel4(type='checkbox', name='hotel', value='2')
            label(for='hotel4')  2*
            br
            input#hotel5(type='checkbox', name='hotel', value='1')
            label(for='hotel5')  1*
            br
        button#use-filters.btn.btn-secondary Застосувати
        button#clear-filters.btn.btn-secondary Скинути

```

Код з файлу footer.pug

```

footer
    .footer-copyright.text-center
        | © 2020 Copyright:
        a(href='https://github.com/TRovnina') Rovnina Tetiana

```

Код з файлу header.pug

```

//шапка сайту
head
    // метаданні та назва сайту
    meta(charset='utf-8')
    title Around the World
    // підключення css
    link(rel='stylesheet',                                type='text/css',
href='../node_modules/bootstrap/dist/css/bootstrap.min.css')
    link(rel='stylesheet',                                type='text/css',
href='../public/stylesheets/style.css')

```

Код з файлу hotel-tours.pug

```
table.table-tours
  tr.head
    th Дата
    th Ночі
    th Ціна (грн)
    th Трансфер
    th Оператор
    th Перегляд туру
  - var i = 0;
  for tour in tours
    tr
      td= tour.tourDate
      td= tour.nights
      td= tour.price
      td= tour.type_transfer
      td= tour.name_operator
      td
        button.btn.btn-outline-primary.tour-view(data-tour-id=tour.code) Перегляд
```

Код з файлу hotel-view.pug

```
#description_block
  div.row.short-inf
    img.col-md-6(src= hotel.photo, alt='hotel photo')
    div.card.col-md-6.description-card(data-hotel-id= hotel.code)
      h3.hotel-title= hotel.name_hotel
      span
        b Країна:
        = " " + hotel.name_country + ", " + hotel.city
      span
        b Клас готелю:
        = " " + hotel.stars + " *"

      span
        b Наявні типи харчування:
        for food in foods
          span= food.id_food

  div
    p.description= hotel.description

  h3.comment-title Доступні тури
  include hotel-tours.pug

  button#open-comment.btn.btn-dark Залишити відгук
```



```

#comment-form.hidden
  .form-row
    .form-group.col-md-4
      label(for='user-email') Email *
      input#user-email.form-control(type='email',
placeholder='Email')
    .form-group.col-md-4
      label(for='user') Имя *
      input#user.form-control(type='text', placeholder='Имя')
  .form-group
    label(for='comment') Комментарий *
    textarea#comment.form-control(rows='3', cols='45',
placeholder='Текст...')
    button#add-comment.btn.btn-primary(data-hotel-id=hotel.id_hotel)
Залишити

h3.comment-title Комментарий
#comments-block
  include comments.pug

```

Код з файлу hotels.pug

```

#hotels_block.row
  include filters.pug
  #hotels.row.col-md-10
    for hotel in hotels
      .hotel_card.card(data-hotel-id= hotel.id_hotel)
        img.img-fluid.hotel-img(src=hotel.photo, alt='photo')
        .stars.row
          - var n = 0;
          while n < hotel.stars
            img(src='../public/images/star.png', alt='')
            - n++
        span.hotel-title= hotel.name_hotel
        span
          b= hotel.name_country + ", " + hotel.city

```