

Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації  
Кафедра комп'ютерних наук

Бакалаврська кваліфікаційна робота на тему:

РОЗРОБКА WEB-РЕСУРСУ ГОТЕЛЮ

Виконав: студент (4) курсу, групи 4КН-216  
спеціальності 122 – Комп'ютерні науки  
(шифр і назва напрямку підготовки, спеціальності)

*Янковий*

Янковий Є.В.  
(прізвище та ініціали)

Керівник: к.т.н., доцент каф.КН

*Денисюк*

Денисюк В.О.  
(прізвище та ініціали)

«04» червня 2025 р.

Рецензент: к.т.н./професор каф. АІТ

*Папінов*

Папінов В.М.  
(прізвище та ініціали)

«05» червня 2025 р.

Допущено до захисту

Зав. кафедри Яровий А.А.

«06» червня 2025 р.

Вінниця ВНТУ - 2025 рік

Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації  
Кафедра комп'ютерних наук  
Рівень вищої освіти перший (бакалаврський)  
Галузь знань – 12 Інформаційні технології  
Спеціальність – 122 Комп'ютерні науки  
Освітньо-професійна програма – Комп'ютерні науки

**ЗАТВЕРДЖУЮ**

Завідувач кафедри КН  
проф., д.т.н. Яровий А.А.

«05» березня 2025 р.

**ЗАВДАННЯ  
НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**  
Янковому Євгенію Володимировичу

1. Тема роботи: Розробка WEB-ресурсу готелю.

Керівник роботи: Денисюк Валерій Олександрович, к.т.н., доцент  
затверджені наказом вищого навчального закладу «20» березня 2025 року № 97

2. Строк подання студентом роботи 30 травня 2025 р.

3. Вихідні дані до роботи :

об'єктно-орієнтована мова програмування; середовище розробки Visual Studio; зручний інтерфейс; швидке завантаження сторінок; інформаційне наповнення; забезпечення статичного та динамічного наповнення сайту.

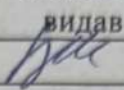
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

аналіз можливостей HTML; аналіз можливостей каскадних таблиць стилів CSS; аналіз можливостей клієнтської мови програмування JavaScript; постановка завдання; розробка структури сайту; реалізація web-сайтів; розробка інтерфейсу сайту; вибір колірної гама; розробка динамічного контенту; тестування роботи сайту.

5. Перелік графічного матеріалу:

схема алгоритму роботи модуля; діаграми класів та компонентів програмного забезпечення; загальний вигляд інтерфейсу користувача; приклад роботи програми.

## 6. Консультанти розділів проекту (роботи)

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата                                                                        |                                                                                     |
|--------|-------------------------------------------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
|        |                                           | завдання видав                                                                      | завдання прийняв                                                                    |
| 1-3    | Денисюк В.О., к.т.н., доцент              |  |  |
|        |                                           |                                                                                     |                                                                                     |
|        |                                           |                                                                                     |                                                                                     |

7. Дата видачі завдання 05 травня 2025 року

## КАЛЕНДАРНИЙ ПЛАН

| №з/п | Назва та зміст етапу                              | Термін виконання |            | Примітка |
|------|---------------------------------------------------|------------------|------------|----------|
|      |                                                   | початок          | закінчення |          |
| 1    | Аналіз предметної області WEB-ресурсу готелю      | 05.05.25         | 07.05.25   |          |
| 2    | Постановка задачі                                 | 08.05.25         | 11.05.25   |          |
| 3    | Розробка програмного модуля WEB-ресурсу готелю    | 12.05.25         | 15.05.25   |          |
| 4    | Обґрунтування засобів розробки WEB-ресурсу готелю | 16.05.25         | 19.05.25   |          |
| 5    | Програмна реалізація WEB-ресурсу готелю           | 21.05.25         | 26.05.25   |          |
| 6    | Тестування WEB-ресурсу готелю                     | 27.05.25         | 29.05.25   |          |
| 7    | Розробка інструкції користувача                   | 30.05.25         | 01.06.25   |          |
| 8    | Оформлення матеріалів до захисту БКР              | 02.06.25         | 04.06.25   |          |
|      |                                                   |                  |            |          |

Студент

Янковий  
(підпис)

Янковий Є.В.  
(прізвище та ініціали)

Керівник роботи

Денисюк  
(підпис)

Денисюк В.О.  
(прізвище та ініціали)

## **АНОТАЦІЯ**

Бакалаврська кваліфікаційна робота складається з 89 сторінок формату А4, на яких є 25 рисунка, список використаних джерел містить 27 найменувань.

У роботі розглянуто актуальну задачу розробки програмного модуля, який забезпечує роботу WEB-ресурсу готелю. Розглянуто основні можливості застосування веб-сайтів у готельній галузі. Проаналізовано та досліджено основні методи розв'язання задачі. Було розглянуто існуючі структури сайтів, обрано найкращу. У розробці сайту використано сучасні технічні засоби та мови, такі як: CSS3, JavaScript, HTML, PHP. Виконано ретельний підбір інформації та розроблено зображення, розроблено логотип. Обрано гармонійну кольорову гаму, повністю розкриває можливості та послуги готелю. Можна рекомендувати розміщення WEB-ресурс готелю у мережі.

Ключові слова: WEB-ресурс, готель, CSS3, JavaScript, HTML, PHP.

## **ABSTRACT**

The bachelor's qualification work consists of 89 pages of A4 format, on which there are 25 figures, the list of used sources contains 27 names.

The paper considers the current task of developing a software module that ensures the operation of the hotel's WEB resource. The main possibilities of using websites in the hotel industry are considered. The main methods of solving the problem are analyzed and researched. The existing structures of sites were considered, the best one was selected. In the development of the site, modern technical means and languages were used, such as: CSS3, JavaScript, HTML, PHP. A careful selection of information was carried out and images were developed, a logo was developed. A harmonious color scheme was chosen, fully revealing the capabilities and services of the hotel. We can recommend placing the hotel's WEB resource on the network.

Keywords: WEB resource, hotel, CSS3, JavaScript, HTML, PHP.

## ЗМІСТ

|                                                                             |    |
|-----------------------------------------------------------------------------|----|
| ВСТУП.....                                                                  | 3  |
| 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ WEB-РЕСУРСУ ГОТЕЛЮ.....                         | 4  |
| 1.1 Аналіз галузі WEB-ресурсів.....                                         | 5  |
| 1.2 Аналіз відомих програм аналогів.....                                    | 7  |
| 1.3 Постановка задачі.....                                                  | 11 |
| 1.4 Висновок до розділу 1.....                                              | 15 |
| 2 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ WEB-РЕСУРСУ ГОТЕЛЮ.....                       | 16 |
| 2.1 Розробка структури WEB-ресурсу.....                                     | 16 |
| 2.2 Розробка UML-діаграм WEB- ресурсу готелю.....                           | 18 |
| 2.3 Висновок до розділу 2.....                                              | 27 |
| 3 ПРОГРАМНА РЕАЛІЗАЦІЯ WEB-РЕСУРСУ ГОТЕЛЮ.....                              | 28 |
| 3.1 Обґрунтування вибору мови програмування.....                            | 28 |
| 3.2 Створення WEB-ресурсу.....                                              | 35 |
| 3.3 Тестування WEB-ресурсу.....                                             | 55 |
| 3.4 Висновок до розділу 3.....                                              | 60 |
| ВИСНОВКИ.....                                                               | 61 |
| СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....                                         | 62 |
| ДОДАТОК А (ОБОВ'ЯЗКОВИЙ) ПРОТОКОЛ ПЕРЕВІРКИ<br>КВАЛІФІКАЦІЙНОЇ РОБОТИ ..... | 65 |
| ДОДАТОК Б (ОБОВ'ЯЗКОВИЙ) ЛІСТИНГ ПРОГРАМИ.....                              | 66 |
| ДОДАТОК В (ОБОВ'ЯЗКОВИЙ) ІЛЮСТРАТИВНА ЧАСТИНА.....                          | 73 |
| ДОДАТОК Г (ДОВІДНИКОВИЙ) ІНСТРУКЦІЯ КОРИСТУВАЧА.....                        | 85 |

## ВСТУП

**Актуальність досліджень.** Подорожуючи, людина має забезпечити себе в дорозі житлом, харчуванням, знайти тимчасовий притулок на новому місці. Для цього існують готелі та ресторани, які цілком задовольняють фізичні потреби людини. Але також є психологічні потреби. Людина має відчувати зручність і комфорт на новому місці, забезпечити собі захист, послуги та дозволя. На початку подорожі, кожний подорожуючий спочатку буде займатися пошуком інформації, про місце куди він поїде, цінову політику, можливість переночувати у тому чи іншому готелі. Гарний рівень реклами готелів та ресторанів забезпечує їм приток клієнтів, популярність, забезпечує вимушено переселених необхідною інформацією. Рекламу можна здійснювати через засоби мас-медіа, такі як журнали, телебачення та Інтернет. В сучасному світі саме Інтернет є передовим засобом надання реклами. Популяризація і розкриття всього асортименту послуг того чи іншого готелю здійснюється через їх сайти. Отже, з вищенаведеного можна зробити висновок, що розробка WEB-ресурсу готелю є актуальною та потребує подальшого дослідження.

**Метою дослідження** є розширення функціональних можливостей WEB-ресурсу готелю.

**Об'єктом дослідження** є процеси, які виконуються WEB-ресурсом готелю.

**Предметом дослідження** є алгоритми та програмне забезпечення, що втілює WEB-ресурс готелю.

**Задачі дослідження:**

- дослідження вже наявних аналогічних рішень;
- вибір та обґрунтування підходу до вирішення поставленого завдання;
- розробка структури та дизайну WEB-ресурсу готелю;

- програмне створення WEB-ресурсу готелю;
- перевірка функціонування ресурсу та аналіз результатів тестування;
- створення користувацької інструкції з експлуатації WEB-ресурсу.

#### **Апробація роботи.**

Робота пройшла апробацію на конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2025). Вінницький національний технічний університет, 2025р.», доповідь «Розробка WEB-ресурсу готелю» [1].

**Публікація:** електронні тези конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2025), 2025р.» [1].

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ WEB-РЕСУРСУ ГОТЕЛЮ

## 1.1 Аналіз галузі WEB-ресурсів

Аналіз галузі веб-ресурсів охоплює вивчення поточного стану, тенденцій розвитку, основних гравців та технологій, що використовуються у створенні та підтримці веб-сайтів і веб-додатків. Це вимагає всебічного дослідження у розробці [2]. Проведемо детальний структурований огляд галузі WEB-ресурсів [3-5]. Визначення галузі веб-ресурсів.

Галузь веб-ресурсів охоплює такі царини програмних розробок:

- створення сайтів (від візиток до великих порталів);
- розробку веб-додатків (наприклад, CRM, інтернет-магазини, SaaS);
- хостинг, підтримку та SEO-оптимізацію;
- цифровий маркетинг та аналітику користувацької поведінки.

Розглянемо поточний стан галузі WEB-ресурсів. Розвиток відбувається за популярними напрямками:

- корпоративні сайти і вітрини;
- інтернет-магазини (e-commerce);
- освітні платформи, блоги, новинні портали;
- SaaS-платформи (як-от Canva, Notion).

Використовуються ключові технології:

- фронтенд - HTML5, CSS3, JavaScript, фреймворки (React, Vue, Angular);
- бекенд - PHP 8+, Node.js, Python (Django/Flask), Ruby on Rails, Java, .NET;
- CMS - WordPress, Joomla, Drupal;
- бази даних - MySQL, PostgreSQL, MongoDB;

- хостинг - Cloud (AWS, Azure, GCP), VPS, shared;
- інтеграції - API, сторонні сервіси (оплата, аналітика, чат-боти).

На ринки присутня конкуренція та використання сучасних технологій:

- глобальний ринок зростає завдяки диджиталізації бізнесів;
- наявна висока конкуренція серед фрілансерів, студій, SaaS-платформ для no-code/low-code створення сайтів;
- лідери ринку - Wix, Squarespace, Shopify, WordPress, Webflow.

Тенденції та перспективи полягають у використанні факторів:

- мобільна оптимізація (mobile-first design);
- PWA (Progressive Web Apps);
- штучний інтелект у веб-дизайні (генератори, оптимізація контенту);
- безпека та конфіденційність (HTTPS, GDPR, cookies management);
- Headless CMS - контент окремо від відображення;
- Serverless-архітектури та JAMstack.

Проблеми та виклики полягають у:

- забезпеченні кібербезпеки;
- підтримці швидкодії та доступності;
- адаптації під нові пристрої та формати (smart TV, голосові інтерфейси);
- конкуренції з платформами-конструкторами.

Таким чином, галузь веб-ресурсів є однією з найбільш динамічних і технічно розвинених. Її розвиток безпосередньо пов'язаний з потребами бізнесу, користувацькими звичками та технічними інноваціями. Професійне зростання в цій галузі вимагає постійного навчання та адаптації до нових технологій.

## 1.2 Аналіз відомих програм аналогів

Проведемо аналіз галузі WEB-ресурсів готелів, адаптованого до сучасних реалій.

Галузь готельного бізнесу активно використовує веб-ресурси для бронювання, інформування, просування послуг і підтримки зв'язку з клієнтами. Веб-сайт готелю сьогодні є не лише візитівкою, а й ефективним інструментом продажів і взаємодії з туристами.

Розглянемо детально призначення веб-ресурсів готелів.

Вони виконують такі основні функції:

- онлайн-бронювання номерів;
- демонстрація фото/відео номерів, території, послуг;
- надання інформації про ціни, акції, умови проживання;
- відгуки клієнтів і рейтинг;
- зв'язок з адміністрацією (форма зворотного зв'язку, онлайн-чат, месенджери);
- маркетинг (блог, SEO, просування в Google Maps).

Поточний стан ринку полягає у використанні таких типів веб-ресурсів готелів:

- офіційні сайти готелів;
- сторінки готелів на платформах ОТА (Booking.com, Airbnb, Expedia);
- агрегатори та туристичні портали.

На ринку наявний високий рівень цифровізації:

- великі готельні мережі мають професійні сайти з повною функціональністю;
- малий і середній бізнес часто використовує шаблонні рішення (Wix, WordPress, Tilda);

- зростає потреба в мобільній версії сайту та підтримці багатомовності.

Серед використовуваних технології представлені 3 таких.

1) Фронтенд:

- HTML5, CSS3, JavaScript;
- анімації, інтерактивні елементи (галереї, мапи, калькулятори вартості).

2) Бекенд:

- PHP, Node.js, Python;
- CMS: WordPress (найпопулярніший), Joomla, Drupal.

3) Додаткові сервіси:

- платіжні шлюзи (Stripe, Fondy, LiqPay);
- інтеграція з системами бронювання (Booking Engine, Channel Manager);
- онлайн-чати (LiveChat, Tawk.to, Facebook Messenger).

Основні виклики на цьому ринку:

- застарілі сайти з поганою мобільною адаптацією;
- відсутність безпечного HTTPS-з'єднання;
- слабка SEO-оптимізація (низька видимість у Google);
- відсутність автоматизації бронювання (все через телефон або email);
- недостатній захист персональних даних клієнтів (GDPR).

Сучасні тренди, які набувають широкого використання:

- PWA (прогресивні веб-додатки);
- штучний інтелект для підбору номерів, автоматичних відповідей;
- VR/360°-тури по номерах та готелю;
- мобільні додатки готелів;
- Big Data та аналітика поведінки користувачів.

Отже, можна зробити Висновки про те, що готельний бізнес потребує сучасних, швидких і зручних веб-ресурсів, які:

- залучають клієнтів;
- дають можливість легко бронювати;
- є інструментом цифрового маркетингу.

Інвестиції в професійний веб-ресурс сьогодні – це не витрата, а ключовий фактор конкурентоздатності готелю.

Станом на 2024 рік в Україні функціонувало 2017 готелів. Це число включає різні типи закладів розміщення, такі як готелі, мотелі та хостели [6-8].

Структура готельного ринку України на 2024 рік така:

- міжнародні готельні мережі - 1,1%;
- національні мережеві готелі - 5,1%;
- незалежні готелі різних форматів - 80,3%;
- мотелі - 9,6%;
- хостели - 3,9%.

Простежується стійка динаміка зростання. У порівнянні з попередніми роками, кількість готелів зросла на 9,1%, мотелів — на 10,7%, а хостелів — на 28,7%. Це свідчить про стійкість і адаптивність готельного сектору до сучасних викликів, зокрема завдяки розвитку внутрішнього туризму [9-11].

Також простежуються перспективи розвитку. У період з 2024 по 2026 роки в Україні планується реалізація щонайменше 45 нових готельно-апартаментних комплексів на 6670 номерів та 108 котеджних містечок готельного типу на 3097 будинків. Ці проекти спрямовані на задоволення зростаючого попиту на якісні послуги розміщення, особливо в регіонах з активним розвитком туризму.

Дослідження дозволило обрати лідерів серед готельних мереж:

- Optima Hotels - 64 готелі;
- Ribas Hotels Group - 21 готель;

- Premier Hotels & Resorts - 11 готелів;
- Accor (міжнародна мережа) - 9 готелів;
- Radisson - 5 готелів

Регіональні особливості полягають у тому, що більшість міжнародних готелів зосереджені у великих містах, наприклад:

- Київ - 14 готелів;
- Львів - 4 готелі;
- Одеса - 2 готелі;
- Трускавець та Буковель - по 1 готелю.

Це свідчить про концентрацію міжнародного готельного бізнесу в туристично привабливих та економічно активних регіонах.

У якості приклада, розглянемо розвиток готельної справи у Вінницькій області.

Станом на 2025 рік, у Вінницькій області функціонує понад 250 готелів та інших закладів розміщення. Це число включає як великі готелі, так і міні-готелі, апарт-готелі, хостели та гостьові будинки [12].

Наявна динаміка розвитку готельного ринку Вінниччини. Так відбувається зростання кількості готелів: після 2015 року спостерігається поступове збільшення кількості готелів та аналогічних засобів розміщення в області. Це пов'язано з розвитком внутрішнього туризму та підвищенням попиту на якісні послуги проживання. Зростає різноманітність пропозицій: У регіоні представлені як великі готелі, так і малі заклади на 10–15 номерів, що дозволяє задовольнити потреби різних категорій туристів.

У якості прикладів популярних готелів у Вінницькій області можна навести:

- Hotel France - Один із найпопулярніших готелів у Вінниці, що отримав високі оцінки від гостей;
- FENIX TOWN - Сучасний готель з високим рівнем обслуговування та зручним розташуванням;

- Aristokrat - Готель, що пропонує комфортні номери та широкий спектр послуг для гостей;
- Hotel Dobrodiu - Затишний готель з привітним персоналом та приємною атмосферою;
- HIDE PARK Hotel - Готель, що поєднує сучасний дизайн та високий рівень сервісу.

Ці готелі отримали позитивні відгуки від гостей та мають високі рейтинги на платформах бронювання.

Отже, готельний сектор Вінницької області демонструє стабільне зростання та розвиток. Різноманітність пропозицій дозволяє туристам обирати заклади відповідно до своїх уподобань та бюджету. Зростання кількості готелів та підвищення якості послуг свідчать про позитивні тенденції в туристичній галузі регіону. Це ще раз підтверджує актуальність розробки WEB-ресурсу готелю.

### **1.3 Постановка задачі**

Одним з нових видів комунікації є всесвітня комп'ютерна мережа Internet. Мережа Internet є середовищем, що охоплює широкий спектр засобів подання і передачі інформації. На даний момент часу вирішені проблеми, які пов'язані з технічною стороною передачі інформації. Таким чином, наявність технічної бази і широке коло користувачів стимулює зростання числа інформації, наведеної у WWW. Таку інформацію частіше всього розміщують на так званих сторінках або сайтах.

Популярність в мережі того чи іншого браузеру можна подивитися на відповідних сайтах [13, 14]. Наведемо нижче результати (рис.1.1 а, б). Позначення такі:

- Chrome - Google Chrome,
- Edge - Microsoft Edge,

- IE - Microsoft Internet Explorer,
- Firefox - Mozilla Firefox (identified as Mozilla before 2005),
- Mozilla - The Mozilla Suite (identified as Firefox from 2005),
- Safari - Apple Safari (and Konqueror. Both identified as Mozilla before 2007),
- Opera – Opera,
- Netscape - Netscape Navigator (identified as Mozilla after 2006).

| Браузер                   | Особливості                                                                                     | Орієнтовна частка ринку (%) |
|---------------------------|-------------------------------------------------------------------------------------------------|-----------------------------|
| <b>Google Chrome</b>      | Швидкий, багатий на функції, великий вибір розширень, синхронізація з іншими сервісами Google.  | 65-70%                      |
| <b>Mozilla Firefox</b>    | Настроюваний, відкритий код, фокус на приватності, велика спільнота розробників.                | 4-5%                        |
| <b>Microsoft Edge</b>     | Інтеграція з Windows, підтримка нових веб-стандартів, режим InPrivate для приватного перегляду. | 8-10%                       |
| <b>Safari</b>             | Оптимізований для macOS, тісна інтеграція з екосистемою Apple.                                  | 15-18% (переважно на macOS) |
| <b>Opera</b>              | Вбудований VPN, інтегрований месенджер, турбо режим для швидкого завантаження сторінок.         | 2-3%                        |
| <b>Brave</b>              | Фокус на приватності, вбудований блокувальник реклами, нагороди за перегляд реклами.            | 2-3%                        |
| <b>Vivaldi</b>            | Настроюваний інтерфейс, підтримка багатьох вкладок, інтеграція з поштою та календарем.          | <1%                         |
| <b>Tor Browser</b>        | Анонімний браузер для захисту приватності, використовується для обходу цензури.                 | <1%                         |
| <b>UC Browser</b>         | Популярний в Азії, компресія даних для швидшого завантаження сторінок на повільних з'єднаннях.  | Переважає в Азії            |
| <b>DuckDuckGo Browser</b> | Фокус на приватності, не відстежує користувачів, простий інтерфейс.                             | <1%                         |

a)

| 2025      | Chrome | Edge   | Firefox | Safari | Opera |
|-----------|--------|--------|---------|--------|-------|
| February  | 78.4 % | 11.7 % | 3.9 %   | 4.3 %  | 0.9 % |
| January   | 79.4 % | 10.7 % | 3.9 %   | 3.9 %  | 1.4 % |
| 2024      | Chrome | Edge   | Firefox | Safari | Opera |
| December  | 79.6 % | 10.5 % | 3.9 %   | 3.9 %  | 1.4 % |
| November  | 78.5 % | 11.1 % | 4.0 %   | 4.0 %  | 1.6 % |
| October   | 77.9 % | 11.2 % | 4.2 %   | 4.1 %  | 1.6 % |
| September | 78.6 % | 11.1 % | 4.0 %   | 3.6 %  | 1.6 % |
| August    | 79.5 % | 10.7 % | 4.0 %   | 3.2 %  | 1.5 % |
| July      | 79.8 % | 10.7 % | 4.0 %   | 3.2 %  | 1.4 % |
| June      | 78.8 % | 10.9 % | 4.3 %   | 3.5 %  | 1.9 % |
| May       | 78.0 % | 10.9 % | 4.5 %   | 3.7 %  | 2.2 % |
| April     | 77.4 % | 10.9 % | 4.8 %   | 3.8 %  | 2.3 % |
| March     | 77.6 % | 10.7 % | 4.6 %   | 3.7 %  | 2.2 % |
| February  | 77.5 % | 10.5 % | 4.6 %   | 3.6 %  | 2.0 % |
| January   | 78.1 % | 10.4 % | 4.7 %   | 3.8 %  | 2.1 % |

б)

Рисунок 1.1 – Процент використання браузерів у мережі за 2024-2025 рр.

Як бачимо, Google Chrome впевнено підвищує свою популярність серед користувачів завдяки зручності та швидкості. Найбільшу частину користувачів ІЕ складають люди старшого віку з країн СНД, які користуються браузерами за замовчуванням поставленими в системі.

Поширений браузер це Internet Explorer він використовується як стандартний браузер на операційній системі Windows. Плюс цього браузера в тому, що він займає мало дискового простору, і на перших місяцях стабільно працює. А мінус цього браузера в тому, що він починає видавати помилки після декількох місяців роботи, має місце блокування сценаріїв JavaScript, підвисання при роботі з Flash і некоректна робота з CSS.

Опера - в основному він призначений для роботи з низькою швидкістю Internet-з'єднання. У нього існують спеціальні режими для роботи з низькою швидкістю Інтернету, такі як турбо. Має розширений функціонал і не містить надлишкових плагінів розширення.

Google Chrome - легкий для користувача браузер, який швидко відкриває сторінки. Після установки офіційної версії браузера, ви будете

безкоштовно отримувати свіжі оновлення. Цей браузер гарний, як для високої швидкості інтернет-з'єднання, так і для низької. Гарно працює під час перевірки сценаріїв, швидко завантажує сторінки і підтримує всі версії CSS.

Mozilla Firefox - браузер середнього класу, відрізняється стабільною роботою. Швидкість відкриття сторінок не висока, але і не мала. Цей браузер підійде для простих користувачів, які хочуть послухати музику, подивитися картинки і т.д. Але він має суттєвий мінус – з наступними доповненнями дуже перевантажує систему плагінами і «корисними можливостями» які на ділі виходять недієздатними, зайвими додатками, дуже зажерливий в плані пам'яті.

Apple Safari - хороший браузер від фірми Apple. Він відрізняється стабільністю та легкістю роботи в Інтернеті, має вбудований завантажувач файлів. Підійде як для роботи в Інтернеті, так і для простого серфінгу в мережі, але набув популярності лише серед користувачів Apple MAC.

Оперуючи даними наведеними вище, можна зробити висновок, що розробка сайту під такі браузери як Firefox, Safari, Opera, Google Chrome буде більш оптимальною ніж для роботи під IE – сучасні сайти працюють використовуючи динамічну розмітку та плаваючий дизайн який не відображається в старих версіях IE, інформаційне наповнення буде відображатися в старих версіях браузера, але графічна складова буде більш врізана. Тому, щоб представити найкраще сайт, слід завжди використовувати нові та якісні браузери.

Отже, у процесі розробки веб-сайту необхідно розв'язати такі задачі:

1. визначити принципи реалізації web-сайтів;
2. розробити структуру сайту;
3. розробити статичні і динамічні графічні зображення наповнення сайту;

4. провести варіантний аналіз програмного забезпечення з метою вибору засобів реалізації сайту;
5. протестувати розроблений web-сайт.

В першому розділі було виконано детальний аналіз предметної області, визначено суть технічної проблеми та особливості кожного з видів інструментів для реалізації сайту – а саме, мов html, css та js. Також зроблено огляд існуючих варіантів редакторів для подальшої роботи і реалізації технічного завдання. Аналізувалися переваги і деякі особливості існуючих браузерів і властивості їх роботи з кодом сторінки. Визначені їх переваги і недоліки, які будуть враховані при розробці web-сайту.

Постановка задач дослідження.

Веб-сайт, який розробляється, надає можливість під час роботи над ним, вивчити особливості програмування для середовища WEB, реалізувати ідеї за допомогою створених графічних зображень (статичних і динамічних), розкрити обрану тему (в даному випадку – сайт-реклама готелю), і, як результат – відточити власні навички та представити користувачам мережі новий функціональний сайт на обрану тему. Сайт можна розцінювати як рекламний проект, направлений на поширення інформації про послуги та можливості обраного готелю.

#### **1.4 Висновок до розділу 1**

Проведені дослідження підтвердили актуальність розробки WEB-ресурсу готелю. Проналізовано галузь веб-ресурсів. Виконано постанову подальших досліджень та розробки. Визначено основну групу браузерів, на які орієнтована розробка.

## 2 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ WEB-РЕСУРСУ ГОТЕЛЮ

### 2.1 Розробка структури WEB-ресурсу

Починаючи розробку сайту, слід розглянути основні етапи, з яких складається побудова структури і наповнення контентом сайту.

Розробник мусить пройти всі п'ять етапів, при розробці будь якого проекту, але чіткої послідовності при виконанні поставленої задачі – немає, це гнучка схема, яка лише зазначає основні цілі, але пропускати їх небажано [15].

#### 1. Передпроектна підготовка:

- передпроектне дослідження – необхідно ознайомитися з поставленою задачею та темою;
- розробка технічного завдання – затверджується кінцевий варіант технічного завдання, включаючи вимоги до дизайну і вимоги до технічної частини проекту, розробка структури сайту.

#### 2. Розробка дизайну:

- дизайн-концепція сайту – креативна ідея, розробка основної графічної частини дизайну сайту на прикладі головної сторінки, адаптація до браузерів;
- технічний дизайн – створюються графічні шаблони типових сторінок сайту на основі головної сторінки затвердженої в концепції.

3. Верстка – створення html-сторінок на основі затвердженого дизайну типових сторінок.

#### 4. Програмна частина проекту:

- інтеграція сайту з системою управління – вдосконалення елементів керування сайтом для зручної роботи,

програмування, налагодження серверу, забезпечення безпеки проекту, контроль якості;

- програмування, запуск проекту – на цьому етапі доробляється функціонал.

5. Інформаційне наповнення сайту – якісна гарна картинка не є основною частиною роботи, вона лише підтримує та підкреслює вміст сайту.

6. Тестування сайту – налаштування дизайну та відображення сторінок в різних браузерях може варіюватися. Деякі браузери, як Internet Explorer, можуть некоректно відображати плаваючу верстку сайту чи напівпрозорі елементи сторінок.

Окремі стандарти CSS недоступні для старіших браузерів, тому слід передбачити можливість коректного відображення налаштувань сайту в конкретних випадках.

#### 7. Закінчення роботи, його експлуатація.

Отже, можна зробити висновок, що процес розробки сайту – досить кропітка робота, головним чином через різницю в підтримці різними браузерами багатьох елементів CSS чи JavaScript. Якщо сайт буде перевантажений зображеннями, то буде втрачатися його інформація. Має бути врівноваження між вмістом і оформленням, простота реалізації у випадках неможливості відкриття у старих браузерах.

Проблема пошуку інформації також є однією з найважливіших, адже про країну можна скласти враження лише відвідавши її. На щастя, мережа Internet забезпечує бажаючих великою кількістю доступної інформації, її слід підібрати і систематизувати, розбавивши зображеннями і доповнивши фотографіями. Тому, робимо висновок, що основна проблема – це реалізація в різних браузерах. Другорядна проблема, але не менш важлива – підбір і створення графічного оформлення за допомогою існуючих редакторів і програм.

Також слід врахувати такий принцип, як принцип кроссбраузерності – тобто можливість коректного і однакового відображення створеного сайту у різних браузерах. У першому розділі вже згадувалось, що ІЕ не здатен правильно сприймати деякі вказівки CSS, також він блокує скрипти JS.

Існує велика кількість браузерів. Якщо користувач працює з Windows, то за замовчуванням, йому доступний Internet Explorer (далі – ІЕ), але додатково можна встановити такі як Firefox, Safari, Opera, Google Chrome. Користувачам Apple MAC запропонований Safari, або для їх операційної системи також підходить Firefox 3 чи Google Chrome 1. Не дивлячись на те, що всі останні браузери краще всього працюють з веб-сторінками і обробляють сценарії, цього не можна сказати про ІЕ 6, який все ще є найбільш поширеним браузером в мережі. Навіть ІЕ 7 містить помилки при обробці стилів і блокує деякі сценарії JavaScript [16].

## **2.2 Розробка UML-діаграм WEB-ресурсу готелю**

UML-діаграма класів WEB-ресурсу готелю, можна представити як класи "Сторінка", "Користувач", "Форма", "Готель", "Номер", "Послуга" (рис.2.1).

Припущення для UML-моделі - моделюємо систему сайту готелю. Основні сутності:

- Hotel;
- Room;
- Service;
- Review;
- User;
- ContactForm;
- Page (базовий клас для AboutPage, KitchenPage, MapPage тощо).

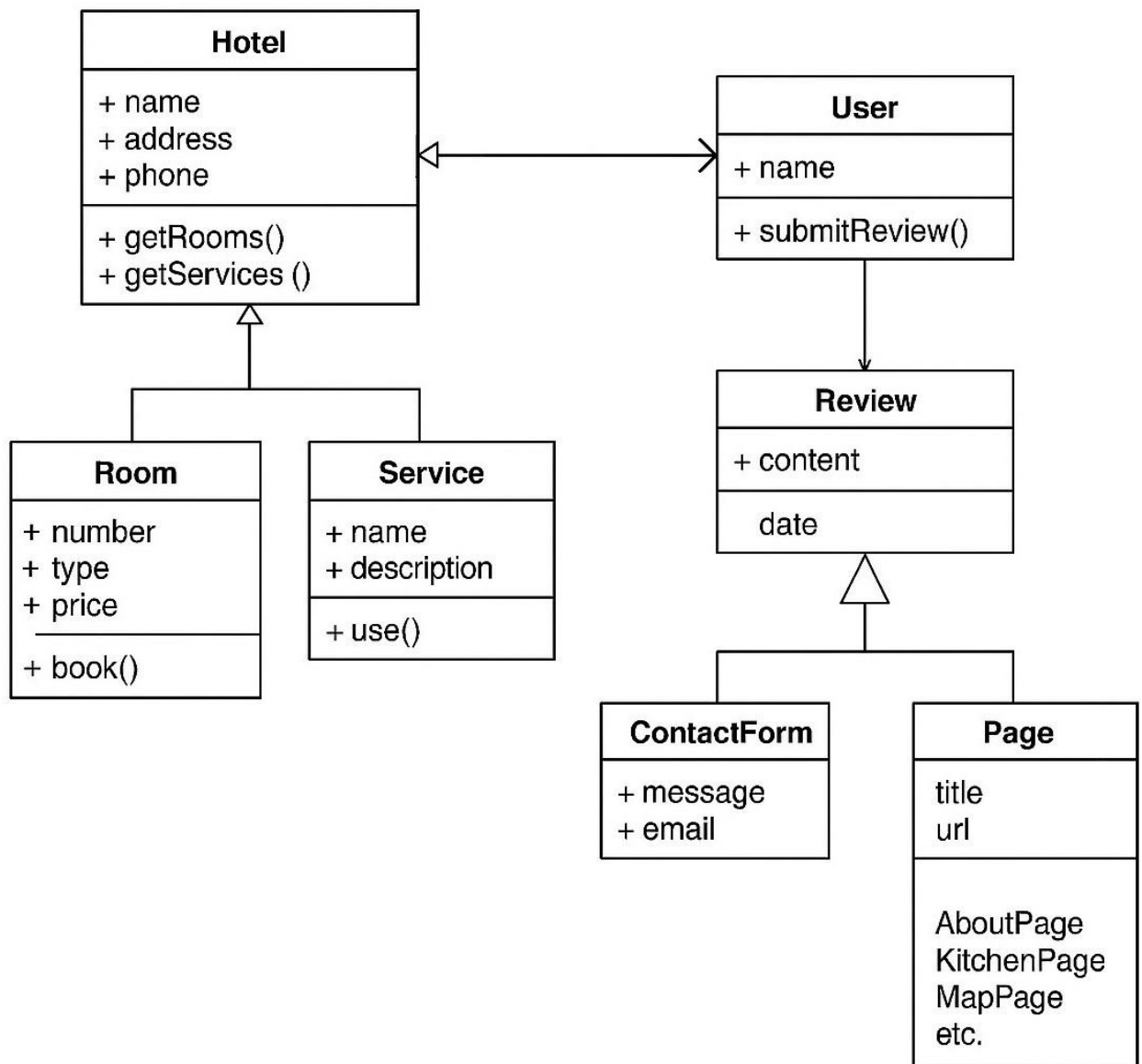


Рисунок 2.1 - UML-діаграма класів WEB- ресурсу готелю

Розглянемо основні класи UML-діаграма класів WEB- ресурсу готелю.

#### 1. Hotel.

- Атрибути:

- + name - назва готелю;
- + address - адреса;
- + phone - телефон.

- Методи:

- + getrooms () - повертає список номерів;
- + getservices () - повертає список послуг.

- Зв'язки: має асоціацію з багатьма Room і Service.

## 2. Room.

- Атрибути:
  - + number - номер кімнати;
  - + type - тип (економ, люкс тощо);
  - + price вартість.
- Методи:
  - + book() – забронювати кімнату.
- Зв'язок: належить до одного Hotel

## 3. Service.

- Атрибути:
  - + name – назва послуги;
  - + description - опис.
- Методи:
  - + use() – використати послугу.
- Зв'язок: надається готелем.

## 4. User

- Атрибути:
  - + name – ім'я користувача;
  - + email – електронна пошта.
- Методи:
  - + submitReview() – залишити відгук
- Зв'язки: має один або кілька Review.

## 5. Review.

- Атрибути:
  - + content - текст відгуку;
  - + date - дата залишення.
- Зв'язок: належить до User.

## 6. ContactForm.

- Атрибути:
  - + `message` - повідомлення;
  - + `email` - контактна пошта.
- Методи: + `submit()` - надіслати форму.
- Зв'язок: незалежний, використовується для зворотного зв'язку.

## 7. Page (суперклас).

- Атрибути:
  - + `title` - назва сторінки;
  - + `url` - шлях до сторінки.
- Підкласи:
  - `AboutPage`, `KitchenPage`, `MapPage`, тощо.
- Зв'язок: спадкування (усі сторінки наслідують загальні атрибути від `Page`).

## Основні взаємозв'язки.

- Aggregation/Composition:
  - `Hotel` агрегує `Room` та `Service`.
- Association:
  - `User` асоціюється з `Review`.
- Generalization:
  - `Page` → `AboutPage`, `MapPage`, `KitchenPage` (усі підкласи є сторінками сайту).

Також для якісної розробки WEB-ресурсу готелю доречно побудувати такі UML-діаграми:

- Use Case Diagram – дозволяє моделювати взаємодію користувача з сайтом;

- Component Diagram - дозволяє визначитися із структурою HTML, CSS, JS, Backend;
- Activity Diagram - показує потік дій у веб-сценарії;
- Sequence Diagram - описує, як користувач взаємодіє з елементами (надсилання форм, завантаження сторінок тощо).

#### 1. Use Case Diagram (UML-діаграма варіантів використання).

Показує, як користувачі взаємодіють із сайтом (рис.2.2).

Наприклад:

- Гість → Переглядає номери;
- Користувач → Надсилає повідомлення через форму;
- Адміністратор → Редагує сторінки.

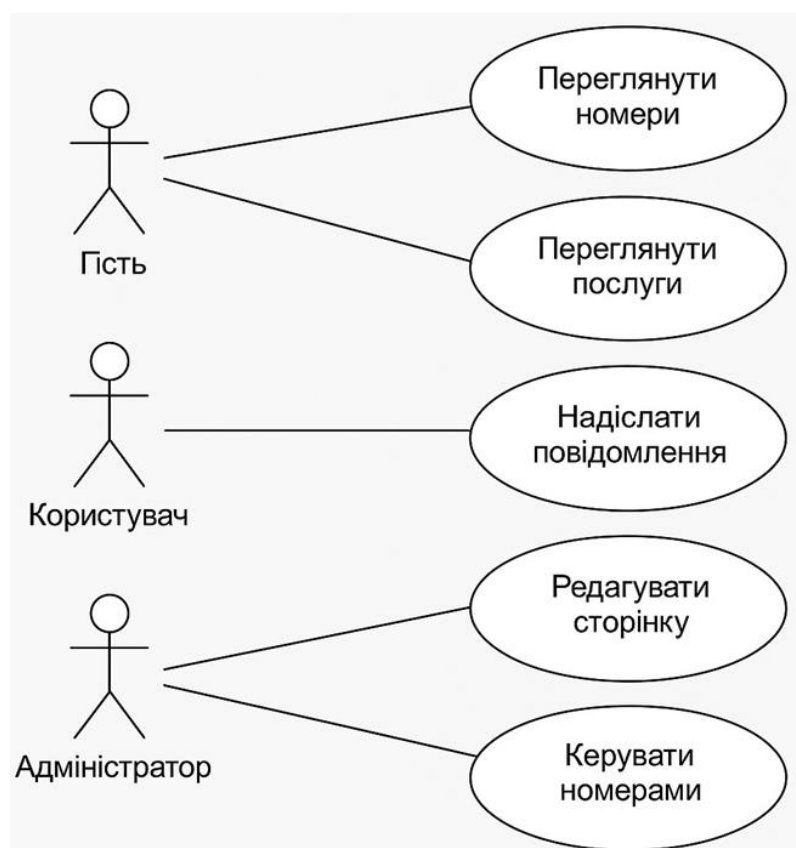


Рисунок 2.2 - UML-діаграма варіантів використання WEB- ресурсу готелю

2. Component Diagram (діаграма компонентів). Відображає структуру веб-додатку — HTML, CSS, JS, серверна частина (PHP, Python тощо).

Наприклад:

```
[HTML Page] --> [CSS Stylesheet]
               \--> [JavaScript Script]
               \--> [PHP Form Handler]
```

3. Activity Diagram (діаграма діяльності). Показує потік дій у веб-сценарії:

```
Відкриття сторінки → Введення у форму →
                    → Надсилання → Підтвердження
```

4. Sequence Diagram (діаграма послідовності). Може описувати, як користувач взаємодіє з елементами HTML.

Наприклад:

```
User → Браузер → HTML-форма →
      → PHP-обробник → База даних
```

UML діаграма компонентів (UML Component Diagram) представляє спрощену архітектуру вебсайту готелю (рис.2.3).

UML-діаграма компонентів WEB-ресурсу готелю демонструє залежності між основними елементами сторінки. Допомогає зрозуміти залежності між файлами вебпроєкту та ієрархію компонентів, які забезпечують зовнішній вигляд та функціональність сторінки.

Компоненти та їхні зв'язки такі.

1. Web Browser. Залежить від:

- HTML Document – браузер завантажує HTML-документ для відображення сторінки.

2. HTML Document. Назва: *Готель Готель*. Залежить від:

- CSS Stylesheet — HTML-файл імпортує CSS для стилізації;
- Image — HTML-файл використовує зображення (наприклад, логотип).

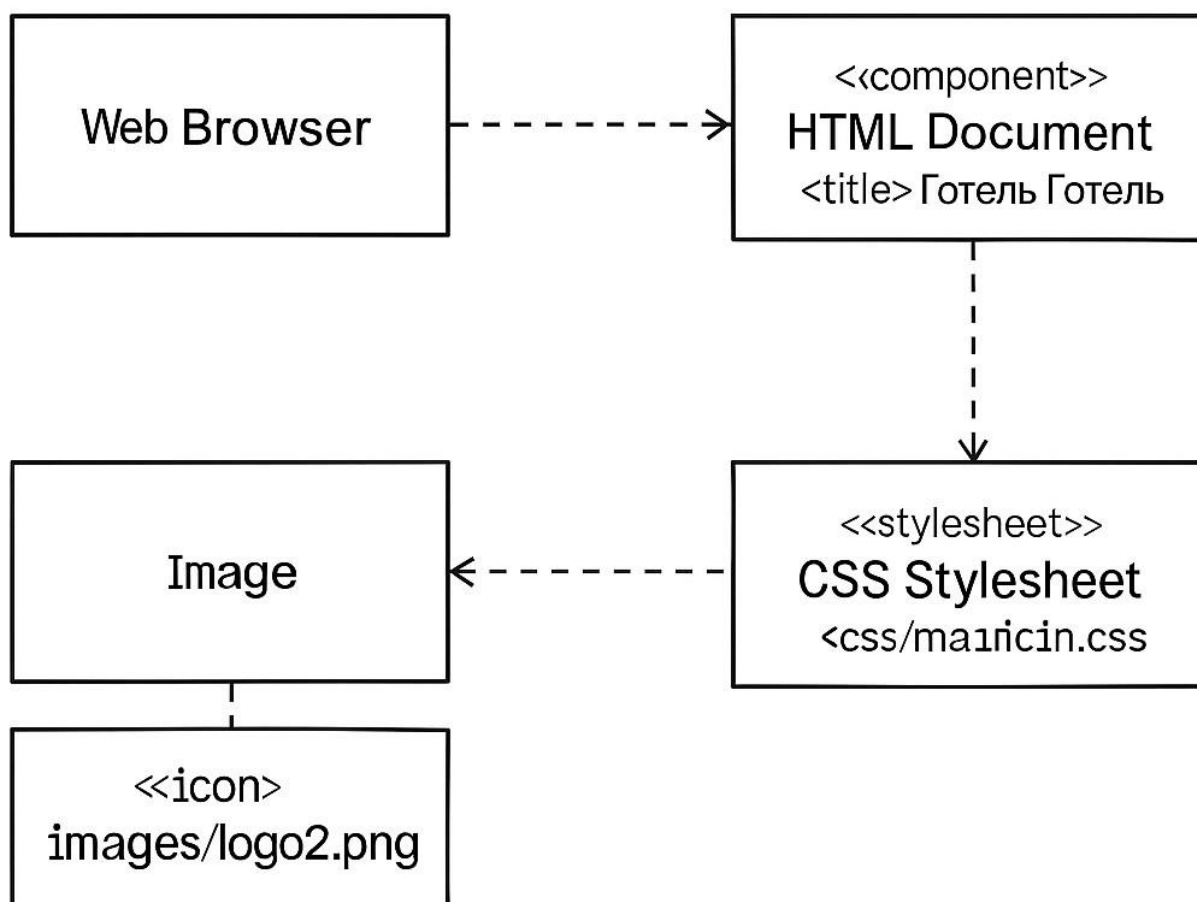


Рисунок 2.3 - UML-діаграма компонентів WEB- ресурсу готелю

### 3.CSS Stylesheet. Шлях:

- `css/main.css` — використовується для оформлення сторінки.

### 3. Image. Шлях:

- `images/logo2.png` — використовується як візуальний елемент (логотип сайту).

Взаємодії. Браузер рендерить HTML, який у свою чергу підключає:

- CSS-файл для стилів;
- Зображення для візуального контенту.

UML-діаграма активності (Activity Diagram) WEB-ресурсу готелю представляє процес завантаження вебсторінки готелю в браузері (рис.2.4). Моделює типовий потік дій при завантаженні HTML-сторінки: від ініціації до повного виводу оформленого контенту з графікою. Вона корисна для візуального розуміння залежностей між HTML, CSS та зображеннями у вебпроекті.

UML-діаграма активності WEB-ресурсу готелю складається з наступних кроків:

1. Початок процесу — ініціація завантаження вебсторінки браузером.
2. Display page — відображення HTML-документа. Назва сторінки: *"Готель «Готель»"*. Цей крок ініціює вивід контенту на екран.
3. Load CSS — завантаження зовнішнього стилю: `css/main.css`. Стилзація елементів сторінки.
4. Load image — завантаження зображення: `images/logo2.png`. Виводиться логотип сайту в шапці сторінки.
5. Завершення — коли всі ресурси завантажено, сторінка повністю готова до взаємодії.

UML-діаграма послідовності (UML Sequence Diagram) надана на рисунку 2.5.

Компоненти взаємодії:

- користувач - відвідувач сайту;
- веб-браузер - клієнт, що інтерпретує HTML;
- веб-сервер - обслуговує запити до HTML/CSS/PHP;
- `guest.php` - PHP-файл, який може взаємодіяти з БД та відображати відгуки.

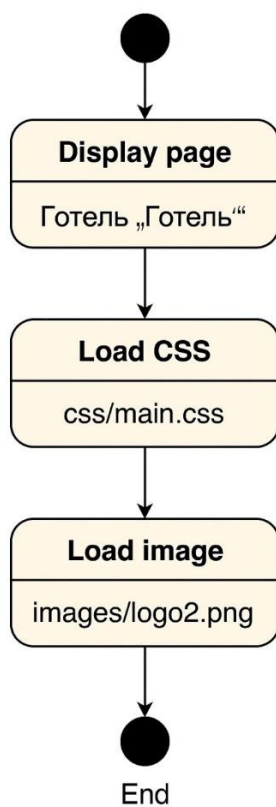


Рисунок 2.4 - UML-діаграма активності WEB- ресурсу готелю

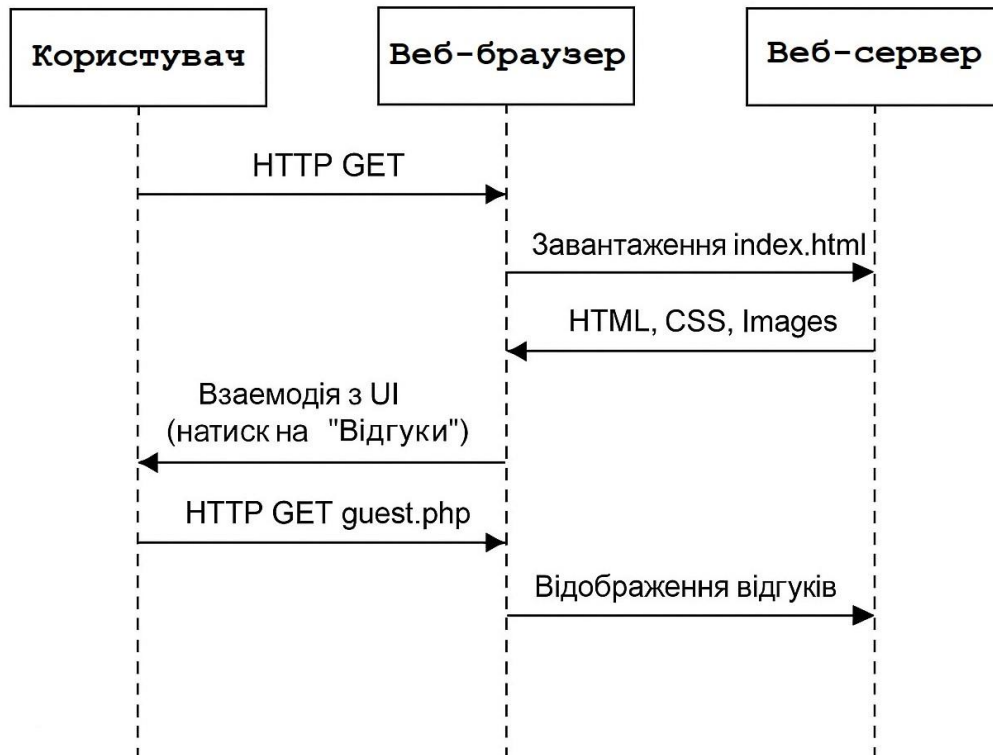


Рисунок 2.5 - UML-діаграма послідовності WEB-ресурсу готелю

Опис сценарію.

1. Користувач ідкриває головну сторінку сайту готелю, переглядає інформацію, взаємодіє з меню навігації.
2. Браузер надсилає HTTP-запит на сервер для отримання `index.html`.
3. Сервер відповідає HTML-документом, стилями CSS, та зображеннями (наприклад, логотип, фото міста).
4. Користувач переглядає сторінку та натискає на посилання «Відгуки».
5. Браузер надсилає новий запит до `guest.php`.
6. Сервер генерує динамічну сторінку з відгуками (наприклад, з бази даних).
7. Браузер відображає сторінку з відгуками.

### **2.3 Висновок до розділу 2**

У результаті досліджень та розробки з'ясовано основні етапи розробки сайту; обрані основні класи програмної реалізації WEB-ресурсу готелю; розглянуто UML-діаграму варіантів використання, яка показує взаємодію користувачів із сайтом; розроблено UML-діаграма компонентів ресурсу готелю, яка демонструє залежності між основними елементами сторінки; досліджено UML-діаграму активності WEB-ресурсу, яка представляє процес завантаження веб-сторінки готелю в браузері; UML-діаграму послідовності дій, яка описує взаємодію між об'єктами в системі.

## **3 ПРОГРАМНА РЕАЛІЗАЦІЯ WEB-РЕСУРСУ ГОТЕЛЮ**

### **3.1 Обґрунтування вибору мови програмування**

Аналіз можливостей HTML.

Подорожуючи по просторам Всесвітньої павутини, можна звернути увагу, як різняться між собою оформлення і можливості веб-сторінок. Але не кожен користувач знає, що все це різноманіття реалізована на основі розмітки HTML.

Мова розмітки гіпертексту HTML (Hyper Text Markup Language) є стандартним засобом представлення інформації у середовищі World Wide Web (WWW) у вигляді веб-сторінок. Завдяки незалежності від платформи і стандартизованому характеру HTML документи, написані з його використанням, можна безперешкодно продивлятися і модифікувати на комп'ютерах з різним програмним і апаратним забезпеченням [15]. Для зручного пояснення надалі слід навести пояснення основних понять в мові HTML:

Гіперпосилання – фрагмент тексту, який є вказівником на інший файл або об'єкт. Гіперпосилання забезпечують можливість переходу від одного документа до іншого.

HTML-файл, HTML-сторінка, веб-сторінка, веб-документ – документ, створений у вигляді гіпертексту. Такі файли мають, як правило, розширення html чи htm. Веб-сторінки складаються з набору окремих елементів, які містять форматовані дані. Елементи можуть бути вкладеними один до одного.

Тег – позначення початку та кінця елемента. Тег – це команда, що визначає правила форматування тексту. Теги визначають межі дії елементів та відділяють елементи один від одного. Теги заключаються у дужки «<» та «>», кінцевий тег ще має косу риску «/» [15].

Форматування структури сторінки відбувається за допомогою тегів, і створює «каркас» веб-документа – розділення між формами і текстом, таблиці та форми, дозволяє виділяти області для зображень та фонів, задавати META-теги (містить службову інформацію про ім'я розробника сторінки та його електронну адресу, ключові слова для пошуку, застосування і реєстрації в мережі [15]), розділяти сторінку на окремі фрейми-спеціальні окремо завантажені у браузер області не залежні одне від одного, тощо. За допомогою HTML також можна додавати стилі та оформлення до сторінки, наприклад заливати сторінки кольором, визначати різний вигляд для посилань, і т.д. Але для налаштування зовнішнього вигляду краще використовувати CSS, про що буде більш детально в наступному пункті.

Стандарт мови HTML пройшов довгий шлях еволюції, і наразі найпопулярнішою і найпоширенішою в мережі є версія HTML 4.01. Контроль за розробкою відповідних специфікацій по мірі їх розвитку здійснювався міжнародним консорціумом W3C (World Wide Web Consortium). У 2008 році з легкої руки W3C у світ вийшла перша пробна версія HTML5.

Нова версія розмітки має набагато більш широкий функціонал але не підтримується всіма браузерами. Підтверджена робота у таких браузерах як Internet Explorer 9, Firefox 4 і Google Chrome (і версіях вище). Якщо в них запустити сторінку створену за допомогою HTML5, то на моніторі буде такий результат як на рис.1.1. Якщо ж запустити ту саму сторінку у версії Internet Explorer 8 (далі IE) чи нижче, то сторінка відобразиться так, як на рис. 1.2. Тобто, можна зробити висновок, що новий варіант HTML є більш зручним і допомагає власними силами налаштувати привабливий вигляд сторінки. Не дивлячись на те, що всі останні браузери краще всього працюють з веб-сторінками і обробляють сценарії, цього не можна сказати про IE 6, який все ще є найбільш поширеним браузером в мережі. Навіть IE 7

містить помилки при обробці стилів і блокує деякі сценарії JavaScript та не розуміє новий HTML [15, 16].



Рисунок 3.1 – Сторінка з HTML5 і елементом canvas



Рисунок 3.2 – Сторінка з HTML5 в старішій версії браузера

Тому зараз мова розмітки гіпертексту використовується для виконання своєї безпосередньої цілі – налаштування розмітки всередині документа і заповнення його інформацією. Для решти функцій є CSS, php та js.

Аналіз можливостей каскадних таблиць стилів CSS.

Таблиці стилів дозволяють налаштовувати зовнішній вигляд практичного будь-якого елемента HTML-документа. Наприклад, з їх допомогою можна задати колір, шрифт та інші параметри. З використанням таблиць стилів можна забути про необхідність фізичного форматування тексту а також про настройку відображення таблиць, списків та інших елементів веб-документів. Крім того, таблиці стилів дозволяють налаштовувати навіть параметри відображення змісту документа, недоступні при використанні лише HTML-елементів (відступи абзаців, стилі рамок малюнків, навіть інтервали між буквами тексту і вільне позиціонування елементів сторінки) [17].

По-справжньому оцінити CSS можна тоді, коли приходить час змінювати дизайн всього сайту. Наприклад, у нас є сайт, який містить дуже багато сторінок і має лістинги (для цього використовують спеціальний тег `<pre>`). По якійсь причині необхідно змінити колір шрифту тексту в кожному лістингу, для цього (при умові використання лише можливостей HTML) довелося би вносити зміни до кожного тегу. А при використанні каскадних таблиць достатньо змінити (чи додати) один запис до таблиці стилів.

Таблиці стилів можна розташовувати всередині кожного веб-документа, але правилом гарного тону вважається виносити каскадні таблиці стилів в окремі файли. Це дозволяє зручно налаштовувати стилі окремо від веб-документа і зекономити кількість коду та сторінок, помістивши посилання на одну таблицю стилів в потрібні веб-сторінки. У випадку розмі

До появи CSS дизайнери веб-сторінок були обмежені можливостями розмітки і оформлення мови HTML.

HTML є головною основою для створення сторінок в мережі, але він не є інструментом формування дизайну. Він здатний забезпечити базове оформлення сторінки в браузері через найпростіше оформлення тексту, зображень та таблиць та інших елементів сторінок. Але спроби дизайнера надати сторінці привабливого вигляду через інструменти HTML можуть забрати забагато часу і стрічок коду, в результаті це не окупиться оскільки веб-сторінки виходять громіздкими і повільно загрузаються.

Таблиці стилів CSS мають наступні переваги.

1. Більше можливостей форматування, ніж в HTML. В CSS можна форматувати абзаци по мірі їх появи в тексті (наприклад, з абзацним відступом і з довільним інтервалом між абзацами) та змінювати між стрічні інтервали (відстань між двома сусідніми стрічками тексту в абзаці);
2. При використанні CSS можна вирішувати яким шляхом додати фонове зображення на сторінку. Воно може відображатися у вигляді безшовної мозаїки, повторюватися, замоцуватися тощо;
3. Стилi каскадних таблиць займають набагато менше місця, ніж форматуючі команди HTML. Наприклад, тег `<font>`. Використовуючи CSS дизайнер може зменшити розмір веб-сторінки. В результаті вони будуть стильно виглядати і швидко завантажуватися;
4. Таблиці стилів полегшують процес оновлення сайту. Можна зібрати всі стилі в єдиний зовнішній файл і зв'язати його з всіма сторінками сайту. Коли дизайнер редагує стиль, зміни моментально переносить зміни у всі елементи сторінок сайту, де є посилання на описані в таблиці стилі. Можна повністю змінити зовнішній вигляд сайту шляхом редагування єдиного файлу таблиці стилів [17].

Аналіз можливостей клієнтської мови програмування JavaScript. По-справжньому ефективні сторінки створюються з використанням скриптів. Скрипти можна створювати на різних мовах, але у мережі найбільш поширеними є PHP і JS [16, 17].

Скрипти, це невеличкі програми (названі сценаріями), написані автором документу і виконуючі у своїй більшості неважкі задачі по обробці даних чи прикрашанні сторінки.

Мова програмування сценаріїв JavaScript є значно спрощеною версією популярної мови Java. Мова Java - компільована мова програмування, орієнтована на підтримку різних платформ. На відміну від Java, JavaScript є інтерпретуємим (як і інші мови для сценаріїв). Це означає, що при завантаженні HTML-документа браузер знаходить в ньому всі програми на js, перевіряє їх на наявність помилок, після чого виконує команду за командою. Цю мову програмування тому і називають «клієнтською» бо фактично скрипт виконується на стаціонарному комп'ютері користувача, а не на сервері. Таким чином можна зробити висновок, що js дозволяє значно знизити рівень завантаженості на веб-серверах за рахунок того, що основні сценарії для роботи зі сторінкою будуть виконуватися на стороні користувача. Також скрипти використовують для створення інтерактивних сторінок, для написання програм, наприклад, легкої браузерної гри чи калькулятора. Більшість сервісів і сайтів використовують js для визначення браузера, операційної системи і так далі, щоб більш коректно відобразити сторінку на конкретному комп'ютері та знати який підключати файл стилів до головної сторінки. Особливо це актуально для користувачів IE, адже він у більшості випадків некоректно відображає стандарти CSS.

Саме JavaScript дозволяє взаємодіяти з cookie (куки – невеликі фрагменти даних, які відправляються веб-сервером і зберігаються на комп'ютерах користувача), зберігати частину інформації на стороні

користувача, і використовувати її при повторному відвідуванні цієї особою сайту.

Також важливо виділити головні плюси цієї мови.

1. На відміну від HTML5, підтримується практично всіма браузерами;
2. Дуже простий в освоєнні та використанні;
3. Має велику бібліотеку готових до використання скриптів;
4. Дозволяє розвантажити сервер, адже вся «робота» виконується на стороні користувача.

Але є і один мінус у JavaScript - елементи меню та інші частини сторінки створені за допомогою js, не можуть індексуватися пошуковими системами.

Таким чином, скрипти дозволяють зробити сайт більш живим та цікавим, саме тому в подальшій роботі над роботою він буде активно використовуватися.

Отже, вирішено реалізацію виконати за допомогою мов PHP та JavaScript [16]. Саме JavaScript дозволяє взаємодіяти з cookie (невеликі фрагменти даних, які відправляються веб-сервером і зберігаються на комп'ютерах користувача), зберігати частину інформації на стороні користувача, використовувати її при повторному відвідуванні цієї особою сайту. Також важливо виділити головні плюси цієї мови: на відміну від HTML5 [18], підтримується практично усіма браузерами; дуже простий в засвоєнні та використанні; має велику бібліотеку готових до використання скриптів; дозволяє розвантажити сервер, адже вся «робота» виконується на стороні користувача. Єдиний недолік - елементи меню та інші частини сторінки, створені за допомогою JavaScript, не можуть індексуватися пошуковими системами. Але скрипти дозволяють зробити сайт більш живим та цікавим, саме тому в розробці і використано JavaScript. Аналіз основних браузерів за популярністю показав, що розробка сайту під такі браузери як

Google Chrome, Microsoft Edge, Safari, Opera, Firefox буде більш оптимальною [18, 19]. Враховано принцип кроссбраузерності, можливість коректного і однакового відображення створеного сайту у різних браузерах.

### **3.2 Створення web-ресурсу**

Розглянемо реалізацію web-сайтів.

Питання структури сайту є одним з початкових і основних при розробці сайту, оскільки безпосередньо відповідає за ієрархію сторінок і зручність навігації.

Структура сайту - це внутрішній устрій сайту, його «кістяк», розташування сторінок, розділів, підрозділів, додаткових матеріалів. Першочерговим завданням дизайнера є створення чіткого порядку з хаотичного скупчення інформації.

З позиції розробника, структуру сайту умовно можна поділити на два рівні — логічний і фізичний.

На логічному рівні структура сайту є сукупністю сторінок, які об'єднані між собою єдиним дизайном, стилем і посиланнями.

На фізичному рівні структура сайту є впорядкованим набором файлів різного типу (HTML-сторінки, зображення, програми, мультимедійні файли). Продумана і зручна файлова структура допомагає розробнику оптимізувати свою роботу, а також буде зрозумілою для інших фахівців, що працюють над проектом [20, 21].

Сайти з невдалою логікою структури не лише ускладнюють роботу відвідувачів, вони позбавляють всякого бажання на них знаходитися. Коли відвідувачі не в змозі знайти потрібну інформацію, вони схильні припускати, що на цьому сайті її взагалі немає. Розчаровані відвідувачі навряд чи запам'ятають цей ресурс, а тим більше захочуть ще раз його відвідати.

Вдалий дизайн в першу чергу має орієнтуватися на користувача, який без особливих зусиль може знайти і використати будь яку інформацію, що міститься у сайті. При цьому повинні бути абсолютно зрозумілими логічна схема сайту і логічний взаємозв'язок між окремими сторінками.

Хороша навігація сайту означає, що користувачі точно знають, де знаходяться, де розташовані елементи сайту і як використовувати ці елементи. Правильна структура інформації дозволяє користувачам без побоювання продовжувати мандрування сайтом, і бути впевненими в тому, що вони завжди зможуть без зусиль повернутися до раніше переглянутих сторінок.

На вдало організованих сайтах відвідувачі можуть вільно переміщатися у пошуках потрібної інформації, не турбуючись про структуру сайту. Піклування про структуру сайту є завданням розробника, а не відвідувачів.

Розглянемо більш детально логічну організацію сайту, існує кілька видів організації сайту на цьому рівні: лінійна, ієрархічна, мережна і комбінована.

Лінійна структура сайту зображена на рисунку 3.3.



Рисунок 3.3 – Лінійна структура сайту

Строго лінійна структура сайту використовується досить рідко, оскільки дозволяє відвідувачам сайту переходити тільки від попередньої сторінки до наступної, і ніяк інакше. Таке подання інформації з однієї сторони полегшує процес взаємодії з відвідувачами: уже на етапі створення сайту можна досить точно прогнозувати поведінку відвідувачів. З іншої точки зору, подібна організація структури сайту не залишає відвідувачам вибору, що може їм не сподобатися [21].

Застосовується лінійна структура сайту для різних онлайн-презентацій, опису покрокових процесів. Основна перевага застосування такої моделі організації сайту полягає в передбачуваності поведінки відвідувача на сайті, завдяки чому можливо вгадати дії відвідувачів ще на етапі створення сайту.

Отже, лінійна структура сайту не підходить для сайтів з великим обсягом інформації.

Ієрархічна структура сайту зображена на рисунку 3.4.

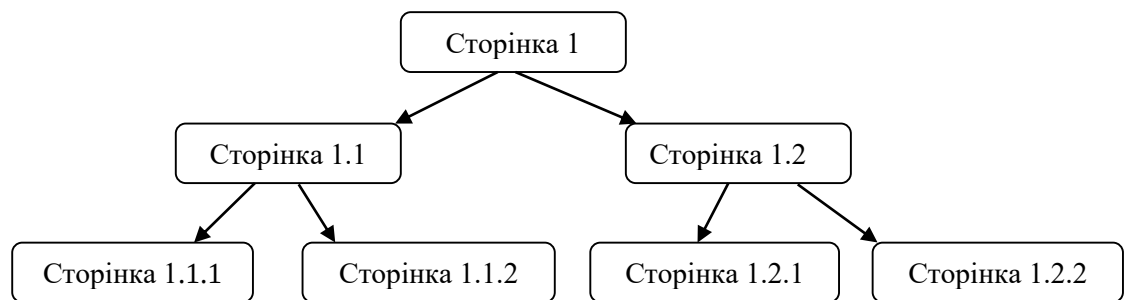


Рисунок 3.4 – Ієрархічна структура сайту

Дана структура сайту передбачає застосування головної сторінки, на якій розміщено меню з посиланнями на розділи сайту, що розташовані на наступних сторінках. Розділи можуть містити в собі посилання на підрозділи чи іншу детальну інформацію. Ієрархічна структура дозволяє відвідувачам сайту за своїм бажанням управляти глибиною відвідування сайту. Відвідувачі можуть заходити тільки на сторінки верхніх рівнів ієрархії, або ж спуститися до нижніх рівнів. При цьому воля вибору відвідувача залежить від ширини дерева [21].

Якщо відвідувачам для досягнення кінцевої мети необхідно зробити занадто багато натискань клавіші «миші», то ієрархічна структура сайтів може виявитися в цьому випадку занадто вузькою. Відвідувачів дратує нескінченне натискання, що не приносить очікуваних результатів.

У той же час, занадто широке дерево пошуку, ґрунтуючись на дуже великій кількості варіантів вибору, може змусити відвідувачів витратити багато часу на вивчення наданих варіантів.

Мережна структура сайту зображена на рисунку 3.5.

Така структура сайту є найбільш поширеною структурою в Інтернеті. За такою організацією можна перейти з будь-якої сторінки на будь-яку іншу, що є з нею логічно взаємозв'язаною. В подібній структурі посилання на інші розділи сайту формуються залежно від бажання відвідувача.

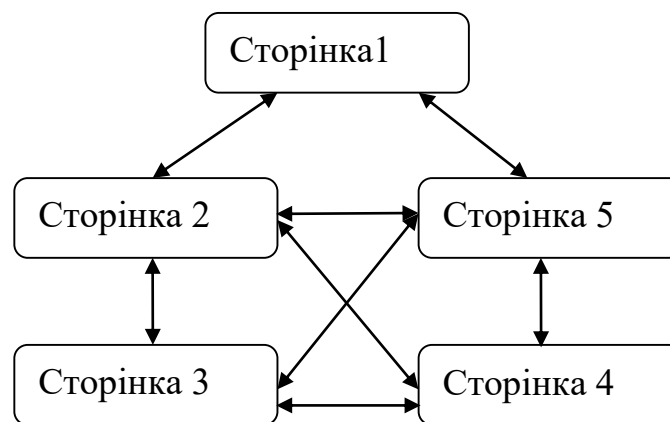


Рисунок 3.5 – Мережна структура сайту

Комбінована структура сайту є поєднанням кількох попередніх структур. Наприклад, на сайті може бути передбачена ієрархічна структура, яка в певному місці може містити виконання покрокових дій (лінійну структуру) [20, 21].

Враховуючи тенденції в Інтернеті та особливості навігації, зроблено висновок, що мережна структура сайту є найбільш оптимальною та зручною для навігації по інформаційно насиченому сайті. Кожна сторінка буде зв'язана і міститиме посилання на сусідні сторінки, тому меню розташується збоку від контейнеру з контентом і матиме всі необхідні посилання. Це забезпечить швидкість та мобільність при переході по сторінкам і дозволить «тримати все під рукою». Далі, розглянемо фізичну структуру сайту.

Всі файли з розширенням \*.html будуть знаходитися в «кореневій» папці. Оскільки кожна з сторінок матиме посилання на решту – то принципової різниці яка сторінка є головною по ієрархії – немає. Але прийнято мати «головну» сторінку, index.html, саме з неї і розпочинатиме роботу сайт при запуску.

Отже, сайт міститиме наступні сторінки:

- головна;
- календар свят;
- новорічна традиція;
- відео;
- фотогалерея.

Посилання на ці сторінки містяться в боковому меню, яке буде виводитися при завантаженні кожної з перелічених сторінок у незмінному вигляді. Розміщення у виділеному збоку секторі дозволить зорозділити простір сайту на область для контенту, область для футеру і шапки, і область для меню. Меню також можна було б розмістити у стрічку, над областю контенту, але було обрано саме такий спосіб розташування навігації.

Звичайно, назви сторінок будуть даватися на англійській мові, кодування файлу UTF-8. В «кореневій» папці будуть міститися окремі папки з зображеннями використаними під час роботи і завантаженими на сайт, папки з скриптами і стилями. Відповідно до ієрархії вкладень у кореневій папці, слід передбачити коректне посилання всередині сторінок на використані ресурси [17, 22].

У складі сайту мають бути присутні файли, які можна віднести до службових. Зазвичай, їх розміщують в «корені» сайту. Це файли, що призначені для виконання певних завдань і мають обов'язкове стандартне ім'я, а також файли, назви яких є довільними і визначаються розробниками. Такими файлами є index.html і style.css. Але для зручності, всі стилі відведено в окрему папку css. Папка img вміщуватиме всі зображення –

створені власноруч, фотографії для завантаження в альбом будуть за адресою `img/gal`. Папка `zoom` і весь її вміст (включно зі стилями і скриптами) відповідає за налагодження галереї – збільшення-зменшення зображень по натисненню клавіші «миші». При цьому використовувалися вільні дані з Інтернету, вони збережені в окремій папці, щоб не що це за файл чи за що він відповідає. Наприклад, `tradition.html` – сторінка з інформацією про традиції Японії, `reset.css` – каскадна таблиця стилів, що «скидає» базові налагодження оформлення сторінок в браузері тощо.

Враховуючи сучасні тенденції та особливості навігації, з'ясовано, що мережна структура сайту є найбільш оптимальною та зручною для навігації по інформаційно насиченому сайті [21, 23]. Отже, враховуючи основні моменти в фізичній організації, обраємо мережну структуру сайту.

Розробка інтерфейсу сайту.

Зручний інтерфейс є першим кроком для налагодження діалогу між відвідувачем. Інтерфейс сайту – це сукупність засобів, за допомогою яких користувач взаємодіє з веб-сайтом або веб-ресурсом через браузер. Однією з головних вимог є однозначний вигляд і функціональність при роботі на різних браузерах [24].

Класичним та найпопулярнішим методом створення інтерфейсу є використання `html` із застосуванням `css` і `js`, чи будь-яких інших скрипкових мов на стороні сервера. Проте різна реалізація інших специфікацій в браузерах викликає проблеми при розробці веб-ресурсів та подальшої їх підтримки. Крім того, можливість користувача налаштовувати багато параметрів браузера (таких як розмір шрифту, кольору, відключення сценаріїв) може перешкодили коректній роботі інтерфейсу. Отже, слід передбачити гнучкість та простоту реалізації інтерфейсу.

В цілях простоти реалізації та можливості відображення в будь-яких варіантах налаштувань браузера, вирішено виконати блокове розташування

контенту та меню, меню представити у вигляді списків та зорово поділити області ресурсу на зони, представлені на рисунку 3.6.

Шапкою сайту називають верхню частину сторінки, яка служить для індивідуалізації сайту, тобто для розпізнавання його з-поміж інших сайтів на подібну тематику. Вона є аналогом заголовку до теми всього ресурсу. Це найголовніший елемент загального дизайну сайту, де зазвичай знаходиться зображення чи напис пов'язаний з тематикою сайту. Є кілька варіантів реалізації шапки сайту, багато випадків в мережі, коли найбільша увага надана шапці, зроблено її анімованою чи з великою кількістю графіки. У випадку готелю, слід обрати простий текстовий вибір шапки (рис. 3.7), щоб занадто не перевантажувати сторінку графікою (надто велика кількість зображень та анімацій складає враження про «не серйозність» організації).

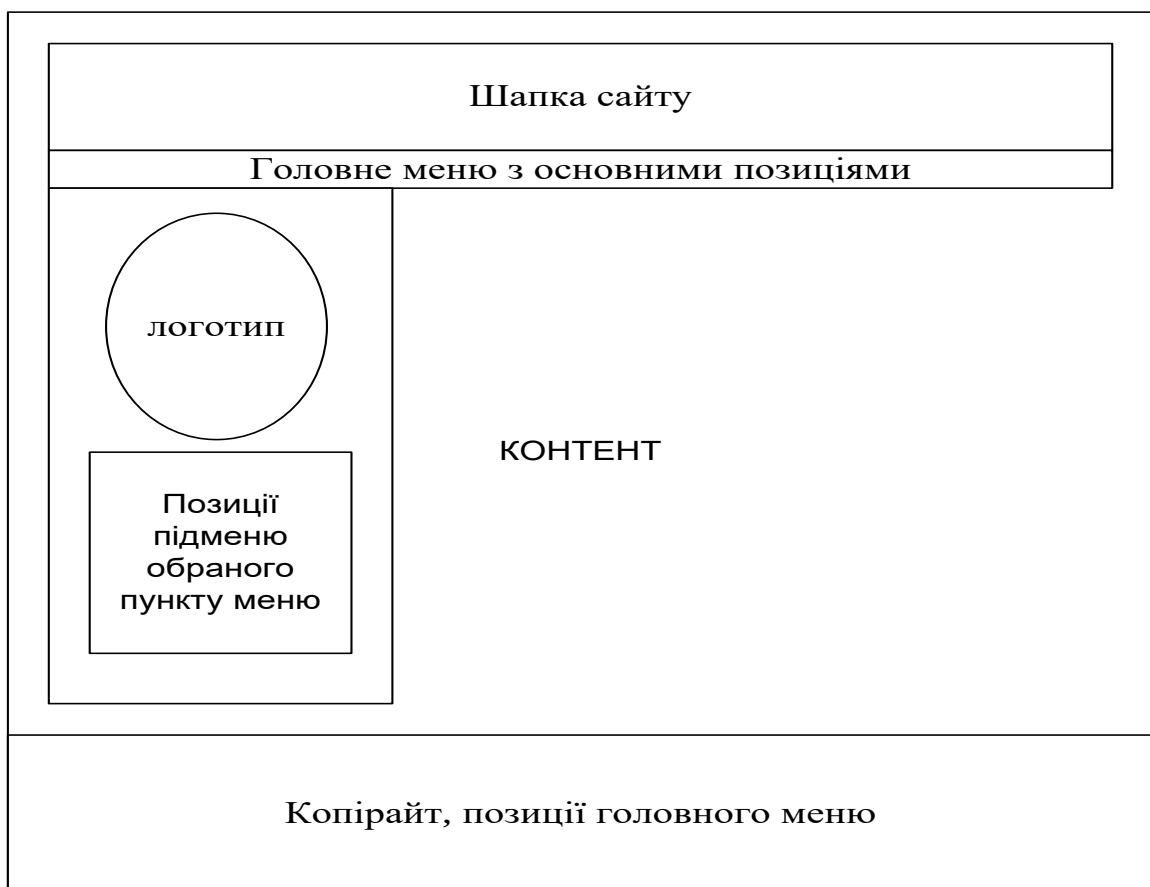


Рисунок 3.6 – Схематичне представлення інтерфейсу сайту



Рисунок 3.7 – Шапка сайту.

Головне меню вирішено винести окремим блоком під шапкою, щоб користувач завжди міг обрати головні пункти меню. Підпункти обрано виносити в бокове підменю, яке прикрашене логотипом. Це дозволить завжди мати потрібні пункти «під рукою» та не перевантажувати сторінку надто великою кількістю випадających меню, які б могли залазити на область контенту. До того ж випадające меню реалізуються через сценарії та таблиці стилів, деякі браузері можуть некоректно відображувати такі меню і здвигати їх.

Вибір колірної гами.

Гармонічний підбір кольорів для оформлення сайту грає важливу роль, оскільки сприяє зоровому сприйняттю тексту. Колір не повинен тиснути на зір і змушувати примружуватися щоб розібрати елементи сайту, а навпаки – плавно підкреслювати все. Має бути легкий контраст. Жовтий та чорний чудово контрастують один з одним. Але є і інші кольори, які здатні підкреслювати один одного, або зливатися чи виділятися. Чітких правил співставлення кольорів не існує, кожен художник це визначає для себе особисто сам, але коли справа доходить до дизайну, то можна виділити деякі правила «гарного тону». Наприклад, «кислотні» кольори в надмірній кількості чи в якості заднього фону сторінки дуже знижують якість перегляду сторінки, це знак поганого смаку [17, 22].

Для визначення гармонійного поєднання різних кольорів користуються колірними (колориметричними) колами. Прийнято вважати, що перші експерименти з упорядкування кольорів провів Ісаак Ньютон у 1676 році.

Він показав, що за допомогою скляної призми можна спочатку розкласти сонячний промінь на складові кольори (спектр), а потім, за допомогою збиральної лінзи знову отримати біле світло. Ньютон виділив з усього спектру сім кольорів («Чапля Осінь Жде Завзято, Буде Сани Фарбувати») і зв'язав кожен з них з нотою на октаві. Століття по тому Йоган Вольфган Гете починає вивчати психологічний вплив кольору. Він звернув увагу, що синій дає відчуття прохолоди, спокою, впевненості, а жовтий створює ефект тепла. Гете створював колірний круг, який показує психологічний ефект кожного кольору. Він поділив усі кольори на дві групи - сторона плюс (від червоного через помаранчевий до жовтого) і сторона мінус (від зеленого через синій до фіолетового). Кольори з боку плюс викликають ефект збудження і бадьорості. Кольори з боку мінус можуть викликати спокій, меланхолію, розслабленість. В даний час користуються формою теорії кольору, розробленої німецьким теоретиком мистецтва Іоханесом Іттенем. Іоханес Іттен розробив «колірні акорди» і змінив колірний круг. В основі колірного кола Іттена лежать три кольори - червоний, жовтий і синій (первинна тріада) (рис.3.8) [25].



Рисунок 3.8 – Коло кольорів Іттена

Згідно Іттена, є кольори першого порядку, другого і третього. Перші формують наступні кольори і є базовими, після них можна створити іншу частину палітри. Побудувавши так коло кольорів, ми отримуємо ідеальну послідовність, де кожний колір має своє чітко визначене місце. Послідовність кольорів має ту ж відповідність, що і у веселці чи спектрі кольорів.

Така візуалізація палітри допомагає підібрати гармонічні пари кольорів, для подальшого використання у дизайні. Є кілька правил для підбору кольорів по цьому колу. Переглянемо їх:

- **Однотонна схема.** У такій схемі використовується колір одного тону (Hue), але з різними значеннями насиченості (Saturation) і яскравості (Brightness). Ця схема справляє враження акуратності і витонченості. Монохроматичні кольори добре поєднуються один з одним, виробляючи заспокійливий ефект. Монохроматична схема приємна на вигляд, особливо при використанні синіх або зелених тонів. Основний тон може комбінуватися з нейтральними тонами також, як наприклад чорний і сірий. Тим не менш при використанні цієї схеми можуть виникнути труднощі з виділенням найбільш важливих елементів, схемі бракує колірного контрасту;
- **Аналогова схема кольорів.** В аналоговій колірній схемі використовуються два кольори, розташовані поруч один з одним. Зазвичай один колір використовується як домінуючий, в той час як інші кольори є допоміжними. Аналогова колірна схема подібна монохроматичній, але пропонує більше нюансів;
- **Контрастна (комплементарна) схема.** Комплементарна колірна схема створюється за допомогою двох кольорів розташованих один проти одного на колірному колі (такі кольори називають *complementary* - додаткові). При використанні контрастної колірної схеми важливо вибрати домінуючий колір і

використовувати додатковий колір для акцентів. Наприклад, основний колір для фону і додатковий колір, щоб виділяти важливі елементи. Комплементарна колірна схема найбільш контрастна з усіх колірних схем і найбільш сильно привертає увагу, але в цій схемі важче досягти балансу кольорів, ніж в монохроматичному або аналогової схемах, особливо при використанні м'яких, теплих відтінків [25];

- Роздільна комплементарна колірна схема є модифікацією стандартної комплементарної схеми. Її відмінність полягає в тому, що можна варіювати тон протилежних кольорів також, як це робиться в аналоговій колірній схемі. Роздільна комплементарна схема пропонує більше нюансів, ніж комплементарна, але більш важка в балансуванні. У такій схемі рекомендується використовувати один теплий колір проти діапазону холодних кольорів (наприклад, червоний проти синього і синьо-зеленого);
- Тріада. У тріаді використовується три кольори рівновіддалені один від одного на колірному колі. Ця схема популярна серед художників, тому що вона пропонує сильний візуальний контраст, зберігаючи при цьому баланс і колірну насиченість. Тріада не така контрастна, як комплементарна схема, але виглядає більш збалансованою і гармонійною;
- Колірна схема «прямокутник», подвійна комплементарна. Ця схема найбільш насичена кольорами, ніж всі інші. У ній використовується чотири кольори - дві пари протилежних (комплементарних) кольорів. У подвійній комплементарній схемі важко досягти гармонії. Якщо використовувати в композиції всі чотири кольори в рівних пропорціях, схема буде здаватися

розбалансованою, тому треба правильно визначити домінуючий і підлеглі кольори для розстановки акцентів.

Розглянувши ці правила, було вирішено вибрати схему за допомогою роздільної комплементарної колірної схеми (рис.3.6), щоб добитися гармонічно-врівноваженого дизайну і колірної палітри з плавними переходами. Для плавності переходів і контрасту будуть використовуватися також напівтони обраних кольорів (рис.3.7).

Базові кольори, обрані для роботи, це помаранчевий та синій. Представлені на рисунку 3.9 відтінки лише підкреслюватимуть атмосферу сайту і створюватимуть фон для тексту. Розглянемо більш детально значення кожного з обраних базових кольорів, щоб зрозуміти який вплив матиме палітра на користувача.

Помаранчевий – це теплий осінній колір, в уяві українців він завжди пов'язаний з достатком, збором урожаю, переходом від однієї пори року до іншої, циклічністю і господарством. Синій – це колір миру, спокою, стійкості, гармонії, довіри, порядку, води, неба, волі (рис.3.10) [26].



Рисунок 3.9 – Обрані основні кольори за принципом «Тріади»



Рисунок 3.10 – Додаткові кольори і напівтони

Таким чином, запропонована кольорова гама (помаранчевий і синій) представляє єдність тепла і свіжості, землі, гармонії і праці. Теплі кольори чудово поєднуються з освіжаючими синіми відтінками. Отже, ці кольори обрані за правилом, можна вважати доцільними для демонстрування обраної теми.

Розробка динамічного контенту.

Вся величезна кількість існуючих сайтів може бути розділена на два види:

- динамічні;
- статичні.

Статичними прийнято називати сайт, що складається з незмінних, тобто статичних HTML-сторінок. Динамічні сайти можуть «підлаштовуватися» під своїх відвідувачів, реагувати на їх дії.

Головною ознакою динамічного сайту є чітка розподіленість – окремо від html-документа, знаходяться в інших директоріях скрипти та каскадні таблиці стилів.

Сторінка посилається у своєму коді на потрібні документи, і таким чином можна забезпечити окремий документ для «підкачки» даних і заповнення контенту.

Крім того скрипти на js та php дозволяють налаштувати інтерактивність сайту – щоб оформлення чи якійсь плагіни відповідали на ті чи інші дії користувача. Все це включається в поняття динамічного контенту.

У випадку сайту готелю обрано динамічну структуру сайту, і яскравим прикладом динамічного наповнення є гостьова книга. На сервері зберігається база даних, в якій зберігаються всі записи для гостьової – відгуки.

Користувач, перейшовши на сторінку «гостьова» має можливість залишити свій відгук, і сторінка має відреагувати на його бажання і відображати старі повідомлення (до його репосту) і нові (вже з новим доданим відгуком).

Для цього створено кілька скриптів, винесених в окремий файл і зв'язаних через теги з основним документом веб-сторінки. Вони наведені нижче з коментарями (рис.3.11 а, рис.3.11 б).

Також слід передбачити щоб форма містила потрібні мітки, для передачі даних зі своїх полів. Більш детально код всієї сторінки гостьової наведено в додатках. Наведемо скрипт, який відповідає за додавання на веб-сторінці тільки-но написаного відгуку (рис.3.12).

У результаті досліджень та розроки реалізовано коди index.html та guest.php (додаток В).

*Код index.html.*

Створює головну сторінку веб-сайту готелю:

- відображає головну сторінку готелю;
- дає змогу переглядати інші сторінки через меню;
- містить навігацію, коротку інформацію про готель, стилізований зовнішній вигляд (через CSS);
- надає перехід на guest.php (PHP-код для відгуків).

Ось що він робить поетапно.

#### 1. Оголошення HTML-документа

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 5.0
    Transitional//EN">
```

#### 2. <head> — Встановлення метаданих сторінки.

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

```
<title>Готель "Готель"</title>
<link href="css/main.css" rel="stylesheet">
```

Детально:

- встановлює кодування UTF-8;
- вказує заголовок у вкладці браузера;
- підключає CSS-файл зі стилями.

3. `<body>` — опис основної структури сторінки.

Верхній банер:

```
<div id="banner">
  
  <ul id="siteTools">...</ul>
</div>
```

Детально:

- виводить логотип готелю;
- показує інструменти: посилання на форму зворотного зв'язку, карту, поле пошуку.

Навігаційне меню:

```
<ul id="nav">...</ul>
```

Меню з переходами по сторінках: Головна, Про нас, Кухня, Послуги, Номери, Контакти.

Основний контент сторінки:

```
<div id="mainWrap">
  <div id="sidebar">...</div>
  <div id="content">...</div>
</div>
```

Ліва панель (sidebar) містить привітання і посилання на:

- карту та акції,
- партнерів та відгуки (guest.php).



```

if(strlen($_POST['mess']) > $maxlenMes)
{
    $mess=substr(check($_POST['mess']),0,$maxlenMes);
    $warn.="Занадто довге повідомлення. Зменшіть до $maxlenMes
символів!";
}else{
    $mess=check($_POST['mess']);
}
}
if (empty($warn)) {
    $f=fopen($filename,"a") or die("Не можу відкрити файл!");
    flock($f,LOCK_EX);
    fwrite($f,"$name|$mail|$url|$city|.time()."| $mess\r\n");//запис
даних у файл
    fflush($f); // записати у файл на сервері
    flock($f,LOCK_UN);
    fclose($f); // закрити файл
    Header("Location: http://".$_SERVER["SERVER_NAME"].
$_SERVER["SCRIPT_NAME"].'?adet=1');
    exit(); }?>

```

б)

Рисунок 3.11 – Скрипт для зберігання відгуків: а) початок ; б) закінчення

Центральна частина (content) забезпечує:

- вивід текст-привітання з інформацією про місто Вінницю;
- фото міста (home\_photo.png);
- рекламний текст про готель та його зручності.

Футер (нижня частина сайту):

```

<div id="footer">
    <div id="col1">...</div>
    <div id="col2">...</div>
</div>

```

- колонка 1 - повторює меню навігації;
- колонка 2 – містить інформацію про авторські права, контактні дані, ім'я розробника.

```

function datestring($date) {
    $monstring=array("січня", "лютого", "березеня", "квітня", "травеня", "червня",
    "липня", "серпня", "вересеня", "жовтня", "листопада", "січня");
    $m=$monstring[$date['mon']-1];
    $daystring=array("у неділю", "у понеділок", "у вівторок", "у середу", "у
    четвер", "у п'ятницю", "у суботу");
    $d=$daystring[$date['wday']];
    if(strlen($date['minutes'])==1)
    {
        $min="0".$date['minutes'];
    }else{
        $min=$date['minutes'];
    }
    return "Повідомлення написано $d, ".$date['mday']." $m ".$date['year']."
    року, в ".$date['hours']." год. $min хв.";
}
if (is_readable($filename)) { // якщо файл читається
    $data=file($filename);// перекинути файл у масив рядків
    $totlines=count($data);//скільки записів
    if (!isset($totlines)) {$totlines=0;}
    $int=(int)($totlines/$maxlines); // скільки сторінок?
    if ($int*$maxlines < $totlines):
        $pages=$int+1;
    else:
        $pages=$int;
    endif;
    if(@$page < 1 || @$page > $pages) {$page=1;} // поточна сторінка
    for ($link=1;$link<=$pages;$link++) // створює посилання на інші
    сторінки ВД
    {
        if ($link!=$page)
        {
            @$line.="<a
href='".$_SERVER["SCRIPT_NAME"]."?page=$link'>[$link]</a>";
        }
        if ($link==$page)
        {
            @$line.=" $link ";
        }
    }
    if (!isset($line)) {$line=1;}
    echo "<p>Відгуків $totlines, Сторінок <b>$line</b></P>";
}
?>

```

Рисунок 3.12 - Скрипт додавання написаного відгуку

*Kod guest.php.*

Цей код - це HTML-сторінка готелю, яка поєднує в собі інтерфейс і РНР-логіку для гостьової книги (відгуків). Розберемося по частинах, що робить цей код:

## 1. Заголовок сторінки (HTML + CSS)

```
<!DOCTYPE html>
<head>
<meta charset="UTF-8">
<title>Готель "Вінничина"</title>
<link href="css/main.css" rel="stylesheet">
<style> .prokrutka { height: 490px; width:
600px; overflow: auto; } </style>
</head>
```

Визначає мета-дані, стилі, зовнішній CSS, та інлайн-стиль для блоку з прокруткою (відгуки).

## 2. Верхній банер та навігація

```

<ul>
  <li><a href="form.html">Написати нам</a></li>
  ...
</ul>
```

Відображає логотип і посилання: «написати», «карта», пошук.

## 3. Меню сайту

```
<ul id="nav">
  <li><a href="index.html">Головна</a></li>
  <li><a href="about.html">Про нас</a></li>
  ...
</ul>
```

Горизонтальне меню з навігацією по сайту.

## 4. Бічна панель

```
<div id="sidebar">
  <h2>Ласкаво просимо</h2>
  <a href="map.html">Ми на карті</a>
  ...
```

</div>

Кнопки переходу до розділів: карта, акції, партнери, відгуки.

## 5. PHP: Обробка форми гостьової книги.

Головні дії:

- перевірка та очищення введених даних (функція `check()`);
- валідація: ім'я, email, місто, повідомлення;
- збереження даних у файл `guest.csv`;
- переадресація після збереження.

```
if ($_POST['add']) {
    // перевірка обов'язкових полів
    // обрізання довгих полів
    // запис у guest.csv
}
```

## 6. HTML-форма для введення відгуків.

```
<form action=<?echo $_SERVER["SCRIPT_NAME"];?>
method="post">
    <input type="text" name="name">
    <input type="text" name="mail">
    <textarea name="mess"></textarea>
    <input type="submit" value="Відправити"
name="add">
</form>
```

Форма для залишення відгуку: ім'я, e-mail, сторінка, місто, повідомлення.

## 7. PHP: Вивід збережених відгуків:

- читання з файлу `guest.csv`;
- підрахунок кількості записів і сторінок;
- формування сторінок і пагінація;

- виведення записів у таблиці.

```
$data = file($filename);  
...  
list($name, $mail, $url, $city, $time, $mess) =  
explode("|", ...);
```

#### 8. Дата написання відгуку

```
function datestring($date) {  
    формує повідомлення: "написано у понеділок, 1 травня 2025 року, в  
    14:23"
```

### 3.3 Тестування web-ресурсу

Розглянемо дві можливі стратегії тестування:

- низхідне;
- висхідне.

Перш за все необхідно пояснити термінологію. Такі терміни як «низхідне тестування», «низхідна розробка», «низхідне проектування» часто використовуються як синоніми. На практиці перші два і є синонімами (вони містять певну стратегію при тестування і створенні класів/модулів), але низхідне проектування – це інший незалежний процес. Спроектований об'єкт таким методом, може тестуватися і низхідним і висхідним методом.

Висхідна розробка чи тестування часто порівнюється з монолітним тестуванням (при такому тестування вважається, що елементи системи серйозного початкового тестування не проходили). Висхідне тестування передбачає початок тестування з термінальних класів (тобто класів, що не використовують методи інших класів). Тому можна зробити висновок, що висхідне тестування є протилежним методом відносно низхідного тестування.

Низхідне тестування починається з верхнього, головного класу (чи модуля) об'єкту. Суворая, коректна процедура підключення наступного послідовного елемента, що піддається тестуванню класу, не існує. Єдине правило, яким слід керуватися при виборі наступного класу, полягає в тому, що ним повинен бути один з класів, методи якого викликаються класом, який перед цим пройшов тестування.

Для тестування сайту було обрано низхідний метод, оскільки він дозволяє на ранніх етапах побачити помилки, в складових частинах складних об'єктів [27].

Крім того, під час тестування сайту перевіряється його відповідність технічному завданню (наявність створених зображень та анімацій), перевіряються його технічні характеристики.

Також розглядається те, як сайт переглядається на різних моніторах – робоча область сайту, де розміщений контент і меню, займає 900 пікселів, і тому буде вміщуватися навіть на найменших моніторах.

Результати перегляду сайту в різних браузерях однакові і наведений на рисунках 3.13 - 3.19. Підібрані кольори і шрифти коректно відображаються, час завантаження мінімальний, всі гіперпосилання працюють, розглянемо переходи між сторінками та особливості їх вмісту нижче.

Для доступу на головну сторінку, достатньо через браузер відкрити файл index.html. Результати відкриття сторінок сайту наведені на рисунках 3.13 – 3.19.

На сайті представлено 6 головних сторінок (див. меню у верхній частині сайту), які мають посилання на дочірні сторінки та сторінки загальної важливості (такі як «Ми на карті», «Написати нам»).

Всі сторінки пов'язані одна з одною і можна зручно переміщуватися між ними. На сторінці «Написати нам» є форма зворотного зв'язку, яка створена на CSS3. На сторінці «Відгуки» розміщена гостьова книга, створена за допомогою скриптів.

На таких сторінках, як «Кухня» і «Послуги» є додаткова панель меню, яка доступна лише в обраній вами під категорії. Тобто, обравши «Кюхню» ви зможе роздивитися меню тої чи іншої кухні, але перейшовши на головну кнопку меню «Головна» ви втратите доступ до цього підменю. Користуючись послугами сайту Google, до сайту було додано карту розміщення готелю.



Рисунок 3.13 - Головна сторінка сайту

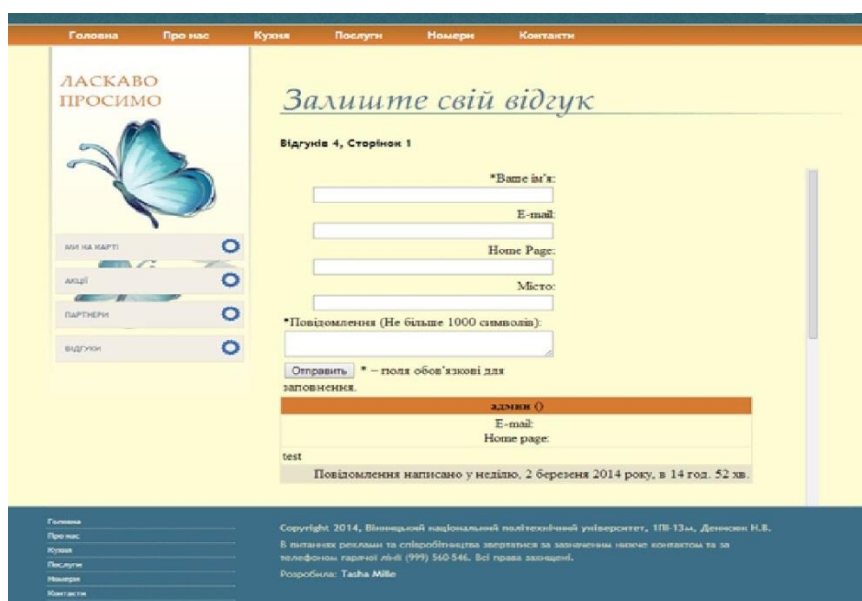


Рисунок 3.14 - Гостьова книга сайту

| Головна                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Про нас | Кухня | Послуги | Наша команда | Контакти |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|-------|---------|--------------|----------|
| <h2 style="text-align: center;">МЕНЮ</h2> <p><b>Заказати страву:</b></p> <ul style="list-style-type: none"> <li>Європейська кухня</li> <li>Українська кухня</li> </ul>  <div> <input type="radio"/> На карті         </div> <div> <input checked="" type="radio"/> Меню         </div> <div> <input type="radio"/> Багетери         </div> <div> <input type="radio"/> ВІДЛІКИ         </div>                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |         |       |         |              |          |
| <h2 style="text-align: center;">Кухня</h2> <p><i>До уваги наступна кухня: європейська та українська. Більш детально меню по кухні у них, у меню зліва.</i></p> <p>Крім того, завжди до нашого ресторану може будь-хто бажаний, не тільки обидві готелів.</p> <p>Під час проживання в готелі, надати 20% знижки на замовлення їдальні у номері. Ви можете не спускатися вниз до ресторану, а продовжити свій відпочинок в обраному номері, ні в чому не відмовляючи собі!</p> <p>Готель "Вінницький" завжди готовий допомогти з організацією фуршетів, конференцій та свят (дні народження, весілля, презентації, семінари, річниця і т.д.), пропонує зал з кількістю місць по 20, 30 і 100 осіб, а також прокошені ведучі.</p> <p>Час роботи кухні з 8:00 до 23:00, ресторан працює з 10:00 до 20:00.</p> <p>Резервування столиків відбувається по телефону гарячої лінії, ласкаво просимо на сторінку "контакти" чи за допомогою форми замовлення.</p> |         |       |         |              |          |
| <div> <div>Головна</div> <div>Про нас</div> <div>Кухня</div> <div>Послуги</div> <div>Наша команда</div> <div>Контакти</div> </div> <div>             Copyright 2014. Винницький національний туристичний університет, 179-12м, Дніпропетровськ.<br/>             Всі матеріали розміщені на сайті винницького університету з використанням логотипу та назви<br/>             телефонним гарячим ліній (096) 360-546. Всі права захищено.<br/>             Розроблено: Tasha Miller           </div>                                                                                                                                                                                                                                                                                                                                                                                                                                                    |         |       |         |              |          |

Рисунок 3.16 - Сторінка «Кухня»



Рисунок 3.17 - Вихід у підменю «Кухні»



Рисунок 3.18 - Сторінка сайту

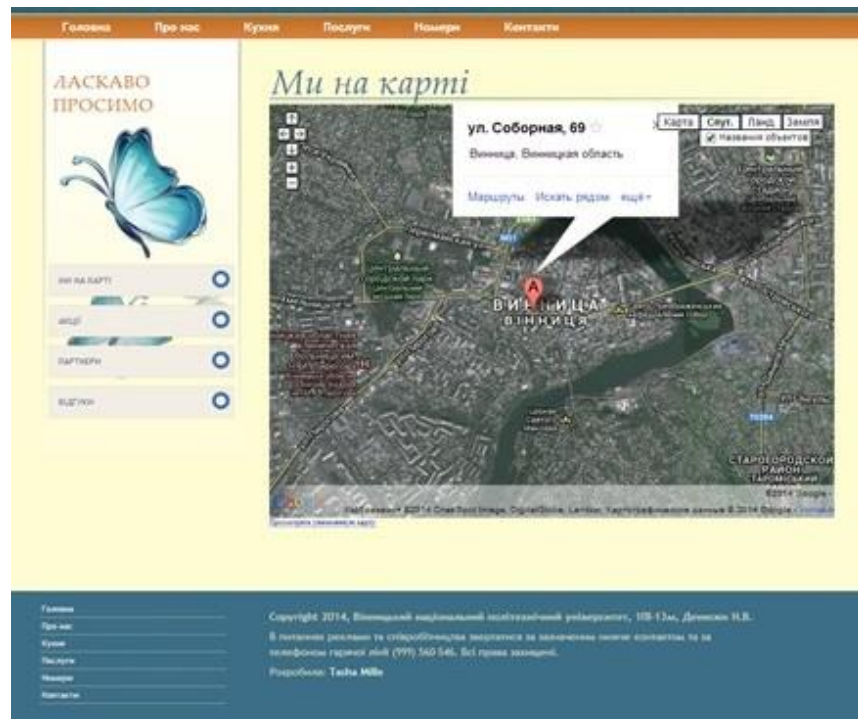


Рисунок 3.19 - Карта Google на сайті

Сайт тестувався в браузерах Firefox, Safari, Opera, Google Chrome. В них не помічено помилок, лише в старих версіях Opera можуть бути проблеми з позиціонуванням і прокруткою форми забортного зв'язку – замість того, щоб відправити форму і на цій же сторінці видати повідомлення про вдало виконану операцію, відбувається перехід на іншу вкладку де і виводиться повідомлення скрипта.

### 3.4 Висновок до розділу 3

У результаті досліджень та розробки обґрунтовано обирання мови програмування, використано сучасні технічні засоби та мови, такі як CSS3, JavaScript, HTML, PHP. Виконано ретельний підбір інформації та розроблено зображення, розроблено логотип. Обрано гармонійну кольорову гаму, яка повністю розкриває можливості та послуги готелю. Тестування підтвердило роботоспроможність WEB-ресурсу готелю у основних в браузерах Firefox, Safari, Opera, Google Chrome.

## ВИСНОВОК

Поставлені перед бакалаврською кваліфікаційною роботою задачі виконано в повному обсязі [2], а саме:

- досліджено вже наявні аналогічні рішення;
- обрано та обґрунтовано підхід до вирішення поставленого завдання;
- розроблено структури та дизайн WEB-ресурсу готелю;
- створено програму WEB-ресурсу готелю;
- протестовано та перевірено функціонування ресурсу;
- створено користувацьку інструкцію з експлуатації WEB-ресурсу.

В ході виконання бакалаврської роботи були проаналізовані найпопулярніші системи для розборки WEB-ресурсу готелю. Відповідно до завдання було проведено аналіз проблеми та обґрунтування необхідності даної розробки. Розроблено структуру та систему навігації сайту. Наступним етапом проектування було створення інформаційного забезпечення, одного із важливих етапів, який відповідає за наявність на сайті якісного контенту, що призводить до популярності сторінки серед користувачів. Відповідно до всіх вимог було проведено розробку графічного забезпечення, що включало в себе розробку графіки та анімації а саме: текстур, логотипів, «шапки» сайту, різноманітних розділювачів та інших елементів дизайну. Було проаналізовано різні типи структур вебсайтів, обрано і розроблено оптимальну структуру вебресурсу та розроблено алгоритми функціонування вебресурсу. Відповідно з завданням до бакалаврської роботи був розроблений WEB-ресурс готелю. Також було проведено тестування, яке підтвердило роботоспроможність WEB-ресурсу.

Мета роботи - розширення функціональних можливостей WEB-ресурсу готелю за рахунок використання дружнього та розвиненого інтерфейса досягнута.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Є.В. Янковий, В.О. Денисюк. Розробка WEB-ресурсу готелю. Молодь в науці: дослідження, проблеми, перспективи (МН-2025). Вінницький національний технічний університет. URL: <https://conferences.vntu.edu.ua/index.php/mn/mn2025/paper/viewFile/24676/20404> .
2. Методичні вказівки до виконання бакалаврських кваліфікаційних робіт для студентів спеціальності 122 «Комп'ютерні науки» [Електронний ресурс] / уклад.: А. А. Яровий, О. В. Сілагін, І. В. Богач. Вінниця: ВНТУ, 2023. (PDF, 54 с.). URL: [https://iq.vntu.edu.ua/method/getfile.php?fname=124285.pdf&x=1&card\\_id=46522&id=124285](https://iq.vntu.edu.ua/method/getfile.php?fname=124285.pdf&x=1&card_id=46522&id=124285) .
3. Web-аналітика. URL: <https://wizeclub.education/blog/web-analitika-shho-tse-yak-yiyi-vikoristovuvati-osnovni-instrumenti-veb-analitiki/>
4. Аналіз і оцінка сайту, для чого вона потрібна і як її провести. URL: <https://gl.ua/blog/analiz-i-otsinka-saytu-dlya-choho-vona-potribna-i-yak-yiyi?page=9>
5. Актуальні проблеми застосування теорії і практики маркетингу. URL: [https://journals.khnu.km.ua/vestnik/pdf/ekon/2011\\_2\\_1/178-182.pdf](https://journals.khnu.km.ua/vestnik/pdf/ekon/2011_2_1/178-182.pdf)
6. В Україні зростає кількість готелів та хостелів, внутрішній туризм сприяє розвитку індустрії – дослідження. URL: [https://forbes.ua/news/v-ukraini-zrostaє-kilkist-goteliv-ta-khosteliv-vnutrishniy-turizm-e-drayverom-rozvitku-industrii-doslidzhennya-03102024-23985?utm\\_source=chatgpt.com](https://forbes.ua/news/v-ukraini-zrostaє-kilkist-goteliv-ta-khosteliv-vnutrishniy-turizm-e-drayverom-rozvitku-industrii-doslidzhennya-03102024-23985?utm_source=chatgpt.com)
7. Аналіз готельного ринку України у 2024 році. URL: <https://inventure.com.ua/uk/analytics/investments/analiz-gotel'nogo-rinku-ukraini-u-2024-roci>
8. Готельний ринок України в умовах війни та перспективи його розвитку. URL: <https://economyandsociety.in.ua/index.php/journal/article/download/4816/4756/>

9. Огляд готельного ринку України 2024. URL:  
[https://ribashotelsgroup.ua/analytics/oglyad-gotel'nogo-rinku-ukrayini-2024/?utm\\_source=chatgpt.com](https://ribashotelsgroup.ua/analytics/oglyad-gotel'nogo-rinku-ukrayini-2024/?utm_source=chatgpt.com)
10. Hotel Market Review of Ukraine 2024. URL:  
[https://www.linkedin.com/pulse/hotel-market-review-ukraine-2024-mykola-siutkin-ceitf?utm\\_source=chatgpt.com](https://www.linkedin.com/pulse/hotel-market-review-ukraine-2024-mykola-siutkin-ceitf?utm_source=chatgpt.com)
11. Зростання доходів українського готельного ринку на 440% до 2024 року.  
URL: [https://agoreview.com/content/company/zrostannya-dohodiv-ukrayinskogo-gotel'nogo-rynku/?utm\\_source=chatgpt.com](https://agoreview.com/content/company/zrostannya-dohodiv-ukrayinskogo-gotel'nogo-rynku/?utm_source=chatgpt.com)
12. Готелі Вінниці і області. URL: [https://hotels24.ua/uk/Vinnytsia-Region/?utm\\_source=chatgpt.com](https://hotels24.ua/uk/Vinnytsia-Region/?utm_source=chatgpt.com)
13. 10 найпопулярніших веб-браузерів: огляд, переваги та недоліки. URL:  
<https://bizmag.com.ua/najpopulyarnishi-veb-brauzery/>
14. Browser Statistics. URL: <http://www.w3schools.com/browsers/default.asp>
15. HTML 5. URL: <https://www.w3.org/TR/2008/WD-html5-20080122/> .
16. Robin Nixon. Learning PHP, MySQL & JavaScript. O'Reilly. 2018. 799.  
URL: [https://unidel.edu.ng/focelibrary/books/Learning%20PHP,%20MySQL%20%20JavaScript%20-%20With%20jQuery,%20CSS%20%20HTML5%20by%20Robin%20Nixon%20\(z-lib.org\).pdf](https://unidel.edu.ng/focelibrary/books/Learning%20PHP,%20MySQL%20%20JavaScript%20-%20With%20jQuery,%20CSS%20%20HTML5%20by%20Robin%20Nixon%20(z-lib.org).pdf) .
17. CSS Tricks – A Complete Guide to Flexbox and Grid. URL: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> .
18. 10 найпопулярніших веб-браузерів: огляд, переваги та недоліки. URL:  
<https://bizmag.com.ua/najpopulyarnishi-veb-brauzery/>
19. Browser Statistics. URL: <http://www.w3schools.com/browsers/default.asp>
20. Website. URL: <https://en.wikipedia.org/wiki/Website>
21. Пасічник О.В., Пасічник В.В. Веб-дизайн: Підручник. - 2-ге вид., стер.  
Львів: ПП “Магнолія 2006”, 2024. 520 с.
22. Adobe Photoshop. URL: <https://www.adobe.com/ua/> .

23. Мосіюк О. О. WEB-технології. Частина 1. Верстка. Житомир: Вид-во ЖДУ ім. Івана Франка, 2020. 56 с. URL: [http://eprints.zu.edu.ua/32361/1/Web\\_ost.pdf](http://eprints.zu.edu.ua/32361/1/Web_ost.pdf)
24. Етапи розробки сайту від ідеї до реалізації. URL: <https://vinnitsa.info/article/etapy-rozrobky-saytu-vid-ideyi-do-realizatsiyi>
25. Базові правила роботи з кольором для дизайну сайтів. URL: <https://skvot.io/uk/blog/glavnoe-o-tsvetovykh-skhemakh> .
26. Романюк О.Н. Основи WEB-дизайну. URL: [https://iq.vntu.edu.ua/method/getfile.php?fname=146471.doc&x=1&card\\_id=71888&id=146471](https://iq.vntu.edu.ua/method/getfile.php?fname=146471.doc&x=1&card_id=71888&id=146471) .
27. Інструменти для тестування кросбраузерності верстки. URL: <https://training.qatestlab.com/blog/technical-articles/tools-for-testing-cross-browser-layout/> .

## ДОДАТОК А (ОБОВ'ЯЗКОВИЙ)

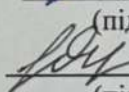
## ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Розробка WEB-ресурсу готелюТип роботи: бакалаврська кваліфікаційна робота  
(бакалаврська кваліфікаційна робота / магістерська кваліфікаційна робота)Підрозділ кафедра комп'ютерних наук  
(кафедра, факультет, навчальна група)Коефіцієнт подібності текстових запозичень, виявлених у роботі  
системою StrikePlagiarism 16,34 %

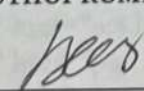
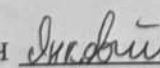
Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

☒ Запозичення, виявлені у роботі, є законними і не містять ознак  
плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту☐ У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але  
надмірна кількість текстових запозичень та/або наявність типових  
розрахунків не дозволяють прийняти рішення про оригінальність та  
самостійність її виконання. Роботу направити на доопрацювання.☐ У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як  
спроб укриття плагіату, фабрикації, фальсифікації, що суперечить  
вимогам законодавства та нормам академічної доброчесності. Робота  
до захисту не приймається.

Експертна комісія:

Яровий А.А., зав. каф. КН  
(прізвище, ініціали, посада)Колесницький О.К., проф. каф КН  
(прізвище, ініціали, посада)  
(підпис)  
  
(підпис)Особа, відповідальна за перевірку   
(підпис)Озеранський В.С.  
(прізвище, ініціали)

З висновком експертної комісії ознайомлений(-на)

Керівник   
(підпис)Денисюк В.О.  
(прізвище, ініціали, посада)Здобувач   
(підпис)Янковий Є.В.  
(прізвище, ініціали)

## ДОДАТОК Б (ОБОВ'ЯЗКОВИЙ)

### ЛІСТИНГ ПРОГРАМИ

#### Лістинг сайту

##### index.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 5.0 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Готель "Готель"</title>
<link href="css/main.css" rel="stylesheet" type="text/css">
</head>
<body class="secondary" id="main">
<div id="banner">

<ul id="siteTools">
<li><a href="form.html">Написати нам</a></li>
<li><a href="map.html">Ми на карті</a></li>
<li>
<form name="form1" method="post" action="">
<div><input type="text" name="search" id="search"></div>
</form>
</li>
</ul>
</div>
<div id="navWrap">
<ul id="nav">
<li><a href="index.html" class="link active">Головна</a></li>
<li><a href="about.html" class="link">Про нас</a></li>
<li><a href="kitchen.html" class="link">Кухня</a></li>
<li><a href="other.html" class="link">Послуги</a></li>
<li><a href="table.html" class="link">Номери</a></li>
<li><a href="vk.html" class="link">Контакти</a></li>
</ul>
</div>
<div id="mainWrap">
<div id="main">
<div id="sidebar">
<h2>Ласкаво просимо</h2>
<div class="actions">
<a class="callToAction" href="map.html">Ми на карті</a>
<a class="callToAction" href="act.html">Акції</a>
<a class="callToAction" href="part.html">Партнери</a>
<a class="callToAction" href="guest.php">Відгуки</a> </div>
</div>
<div id="content">
<h1>Ласкаво просимо до Вінниччини!</h1>
```

<p>Вінниця відома своїм призовим місцем у рейтингу найкращих міст для проживання в Україні, своєю архітектурою у стилі бароко і найбільшим плаваючим фонтаном у Європі. Це надзвичайне місто з задоволенням приймає гостей і пропонує до Вашої уваги також місця відпочинку і ночівлі.<br><br>



Наш готель є одним з найбільших у місті, розташований близько до вокзалів (ж/д і автомобільного) і до відомого фонтану «Roshen». <br><br>

Запрошуємо всіх бажаючих зупинитися у номерах готелю «Готель» і скористатися можливістю дослідити місто!</p>

<p>&nbsp;</p>

</div>

</div>

</div>

<div id="footer">

<div id="col1">

<ul>

<li><a href="index.html" class="link active">Головна</a></li>

<li><a href="about.html" class="linkAbout">Про нас</a></li>

<li><a href="kitchen.html" class="linkLorem">Кухня</a></li>

<li><a href="other.html" class="linkMattis">Послуги</a></li>

<li><a href="table.html" class="linkEuismod">Номери</a></li>

<li><a href="vk.html" class="linkVenenatis">Контакти</a></li>

</ul>

</div>

<div id="col2">

<p><strong>Copyright 2025, Вінницький національний технічний університет, 4КН-216, Янковий Є.В.</strong></p>

<p>В питаннях реклами та співробітництва звертатися за зазначеним нижче контактом та за телефоном гарячої лінії +38-093-05951-53. Всі права захищені.</p>

<p>Розробила: <a href="mailto:zarina92@i.ua">Tasha Mille</a></p>

</div>

</div>

</body>

</html>

## guest.php

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 5.0 Transitional//EN">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Готель "Вінничина"</title>

<link href="css/main.css" rel="stylesheet" type="text/css">

<style>

.prokrutka {

height: 490px; /\* висота нашого блока \*/

width: 600px; /\* ширина нашого блока \*/

overflow: auto; /\* прокрутка по вертикалі \*/

}

```

</style>

</head>
<body class="secondary" id="main">
<div id="banner">

<ul id="siteTools">
<li><a href="form.html">Написати нам</a></li>
<li><a href="map.html">Ми на карті</a></li>
<li>
  <form name="form1" method="post" action="">
    <div><input type="text" name="search" id="search"></div>
  </form>
</li>
</ul>
</div>

<div id="navWrap">
<ul id="nav">
  <li><a href="index.html" class="link">Головна</a></li>
  <li><a href="about.html" class="link">Про нас</a></li>
  <li><a href="kitchen.html" class="link">Кухня</a></li>
  <li><a href="other.html" class="link">Послуги</a></li>
  <li><a href="table.html" class="link">Номери</a></li>
  <li><a href="vk.html" class="link">Контакти</a></li>
</ul>
</div>

<div id="mainWrap">
<div id="main">
<div id="sidebar">
<h2>Ласкаво просимо</h2>
<div class="actions">
<a class="callToAction" href="map.html">Ми на карті</a>
<a class="callToAction" href="act.html">Акції</a>
<a class="callToAction" href="part.html">Партнери</a>
<a class="callToAction" href="guest.php">Відгуки</a> </div>
</div>
<div id="content">
<? // Створюємо Скрипт, який записує дані форми та переадресовує запит
header("Content-Type: text/html; charset=UTF-8");
function check($str){
  $str=ereg_replace("<([>])*>|<|>","",$str); //Витягнути текст із тегів
  $str=str_replace("|","|",$str); //Замінити вертикальну межу (символ 124) кодом,
характеристика буде використовуватися як роздільник полів
  $str=str_replace(""","`",$str); // Замінити оператор виконання (символ 96) кодом
  $str=str_replace("'",'&quot;',$str); // Замінити подвійні лапки їх кодами (тепер ніякі лапки
не потраплять до сторінки)
  $str=str_replace("\r","",$str); // Видаляємо переклад каретки (інакше порушиться структура
бази даних)

```

```

    $str=str_replace("\n",$str); //Замінити початок рядка HTML-кодом (інакше порушиться
структура бази даних)
    return ($str);
}
if (@$warn) {unset($warn);}
$maxlines=10; // повідомлень на сторінці
$maxlen=35; // довжина полів
$maxlenMes=1000;// довжина повідомлення
$filename = "guest.csv"; // Занесемо ім'я файлу до змінної
if (@$_POST['add']) {
    if (empty($_POST['name']) or empty($_POST['mess'])) {// перевірка наявності обов'язкових
полів (саме у методі ПОСТ)
        $warn="Не заповнені всі поля!";
    }else{
        $warn="";
        if(strlen($_POST['name']) > $maxlen)
        {
            $name=substr(check($_POST['name']),0,$maxlen);
            $warn="Занадто довге ім'я. Зменшіть до $maxlen символів! ";
        }else{
            $name=check($_POST['name']);
        }
        if(strlen($_POST['mail']) > $maxlen)
        {
            $mail=substr(check($_POST['mail']),0,$maxlen);
            $warn="Занадто довге ім'я. Зменшіть до $maxlen символів! ";
        }elseif($_POST['mail'] and ! eregi("([0-9a-z\.-_]+)@([0-9a-z\.-_]+\.[a-z]{2,})
*",$_POST['mail'])){
            $warn.="Таких електронних адрес не буває! ";
        }else{
            $mail=check($_POST['mail']);
        }

        if(strlen($_POST['url']) > $maxlen)
        {
            $url=substr(check($_POST['url']),0,$maxlen);
            $warn.="Надто довга адреса. Зменшіть до $maxlen символів!";
        }else{
            $url=check($_POST['url']);
        }
        if(strlen($_POST['city']) > $maxlen)
        {
            $city=substr(check($_POST['city']),0,$maxlen);
            $warn.="Надто довга назва міста. Зменшіть до $maxlen символів!";
        }else{
            $city="";
        }
        if(strlen($_POST['mess']) > $maxlenMes)
        {
            $mess=substr(check($_POST['mess']),0,$maxlenMes);
            $warn.="Занадто довге повідомлення. Зменшіть до $maxlenMes символів!";
        }
    }
}

```

```

}else{
    $mess=check($_POST['mess']);
}
}
    if (empty($warn)) {
        $f=fopen($filename,"a") or die("Не можу відкрити файл!");
        flock($f,LOCK_EX); // чекаємо, поки не станемо єдиними та блокуємо
        fwrite($f, "$name|$mail|$url|$city|.time()."|"$mess"\n"); // записуємо дані у файл
        fflush($f); // записуємо всі зміни на диск
        flock($f,LOCK_UN); // говоримо що, більше не працюватимемо з файлом і розблокуємо
        fclose($f); // Закриваємо файл
        Header("Location:                                     http://".$_SERVER["SERVER_NAME"].
$_SERVER["SCRIPT_NAME"].'?adet=1');
        exit();
    }
}
?>
<?
if (@$adet==1):
    echo "<P style='color:green'>Ваш відгук доданий</p>";
elseif (@$warn):
    echo "<P style='color:red'>$warn</p>";
else:
    echo "<P style='color:navy'><h1>Залиште свій відгук</h1></p>";
endif;
// Створюємо Скрипт, який відображає дані з файлу
function datestring($date){

    $monstring=array("січня","лютого","березня","квітня","травня","червня","липня","серпня","
вересеня","жовтня","листопада","січня");
    $m=$monstring[$date['mon']-1];
    $daystring=array("у неділю","у понеділок","у вівторок","у середу","у четвер","у
п'ятницю","у суботу");
    $d=$daystring[$date['wday']];
    if(strlen($date['minutes'])==1)
    {
        $min="0".$date['minutes'];
    }else{
        $min=$date['minutes'];
    }
    return "Повідомлення написано $d, ".$date['mday']." $m ".$date['year']." року, в
".$date['hours']." год. $min хв.";
}
if (is_readable($filename)) { // якщо файл читається
    $data=file($filename); // весь файл у масив рядків
    $totlines=count($data); // скільки записів
    if (!isset($totlines)) { $totlines=0; }
    $int=(int)($totlines/$maxlines); // обчислюємо кількість сторінок
    if ($int*$maxlines < $totlines):
        $pages=$int+1;
    else:

```

```

    $pages=$int;
endif;
if(@$page < 1 || @$page > $pages) {$page=1;} // поточна сторінка задається через GET.
Якщо не задана, то перша

for ($link=1;$link<=$pages;$link++) // визначаємо посилання на інші сторінки
{
    if ($link!=$page)
    {
        @$line.="<a href='$_SERVER["SCRIPT_NAME"]."?page=$link">[link]</a>";
    }
    if ($link==$page)
    {
        @$line=" $link ";
    }
}
if (!isset($line)) {$line=1;}
echo "<p>Відгуків $totlines, Сторінок <b>$line</b></P>";
}
?>
<div class="prokrutka">
<!-- Створюємо форму, визначивши, які дані нас цікавлять -->
<form action=<?echo $_SERVER["SCRIPT_NAME"];?> name="Guestbook" method=post>
<table width=250px>
<tr><td align=right class=p>*Ваше ім'я:<input class=p type=text SIZE=39
MAXLENGTH=<?=@$maxlen;?> name=name value="<?=@$name;?>"></td></tr>
<tr><td align=right class=p>E-mail:<input class=p type=text SIZE=39
MAXLENGTH=<?=@$maxlen;?> name=mail value="<?=@$mail;?>"></td></tr>
<tr><td align=right class=p>Home Page:<input class=p type=text SIZE=39
MAXLENGTH=<?=@$maxlen;?> name=url value="<?=@$url;?>"></td></tr>
<tr><td align=right class=p>Micro:<input class=p type=text SIZE=39
MAXLENGTH=<?=@$maxlen;?> name=city value="<?=@$city;?>"></td></tr>
<tr><td align=left class=p valign=top>*Повідомлення (Не більше <?=@$maxlenMes;?>
символів):<br><textarea class=p name=mess cols=35><?=@$mess;?></textarea></td></tr>
<tr><td class=p colspan=2><input class=p type=submit value=Отправить name=add id=b> * –
поля обов'язкові для заповнення.</td></tr>
</table>
</form>
<?
// Запис строк
if (!isset($totlines)) {$totlines=0;}
if (!isset($page)) {$page=1;}
if (!isset($line)) {$line=1;}
$to=$totlines+(1-$page)*$maxlines; // $page*$maxlines-1; // від(оскільки це масив і
його можна отримати за ключами, визначимо точне положення найсвіжішої з належних
відобразитися на сторінці)
$from=$to-$maxlines; // ($page-1)*$maxlines-1; // до (найстаріша на сторінці)
if ($from < 0) {$from=0;}

for($i=$to-1; $i >= $from; $i--) // від останньої з необхідних, до найстарішої, що має
відобразитися (оскільки масив нумерується з нуля, відніmemo одиницю з to)

```

```

{
    list($name, $mail, $url, $city, $time, $mess) = explode("|",trim($data[$i])); // перекинемо
масив у змінні
echo
"
<table width=90% cellpadding=2 cellspacing=1 bgcolor=#E7E3CB class=p>
<tr><td align=center bgcolor=#D67B31><b>$name</b> ($city)</td></tr>
<tr><td align=center bgcolor=#FDFCD2>E-mail: $mail<br>Home page: $url</td></tr>
<tr><td bgcolor=#FDFCD2>$mess</td></tr>
}

```

## ДОДАТОК В (ОБОВ'ЯЗКОВИЙ)

## ІЛЮСТРАТИВНА ЧАСТИНА

## РОЗРОБКА WEB-РЕСУРСУ ГОТЕЛЮ

Виконав: студент 4-го курсу,  
групи 4КН-226  
спеціальності 122 «Комп'ютерні науки»

(шифр і назва напрямку підготовки, спеціальності)

Янковий Є.В.  
(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН

Денисюк В.О.  
(прізвище та ініціали)

«03 червня» 2025 р.

Вінниця ВНТУ - 2025 рік

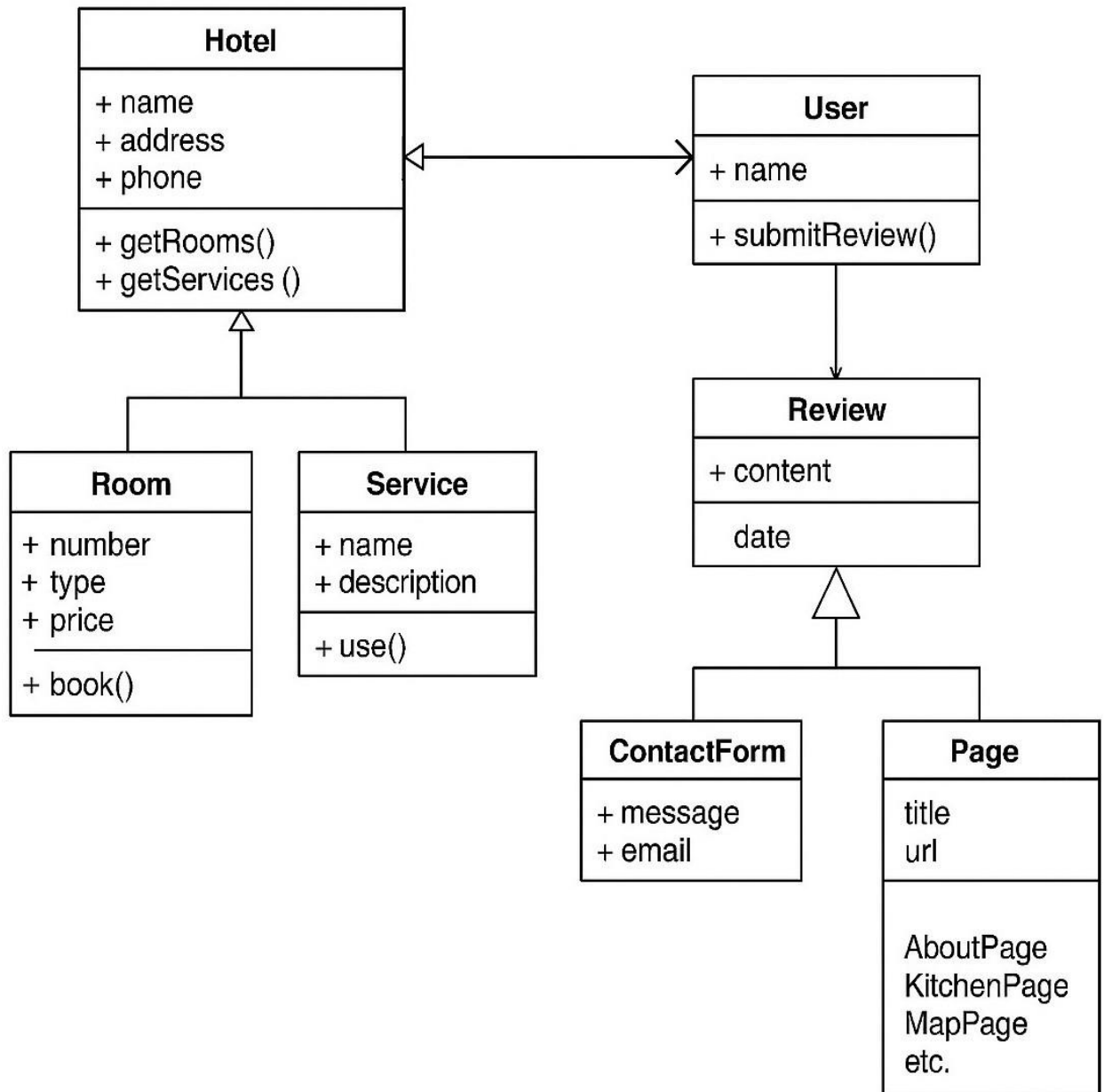


Рисунок В.1 - UML-діаграма класів WEB- ресурсу готелю

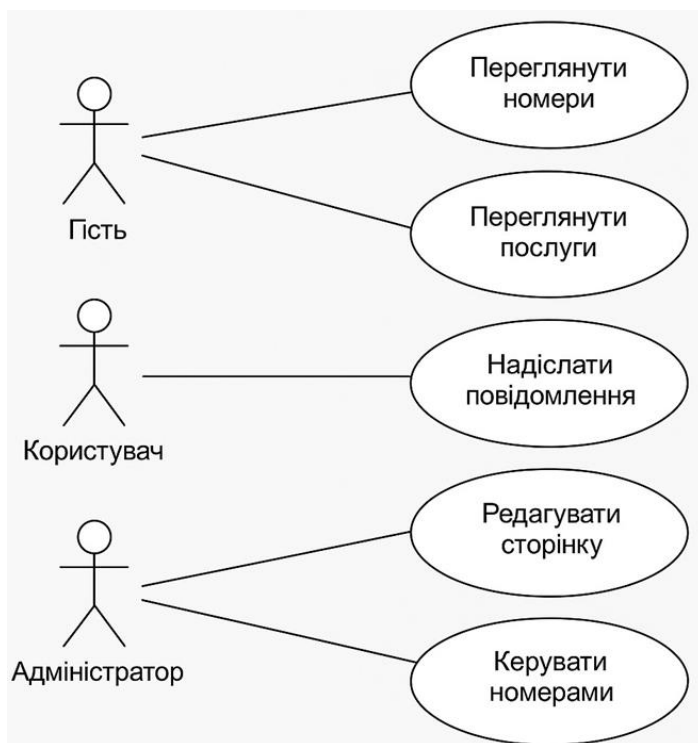


Рисунок В.2 - UML-діаграма варіантів використання WEB- ресурсу готелю

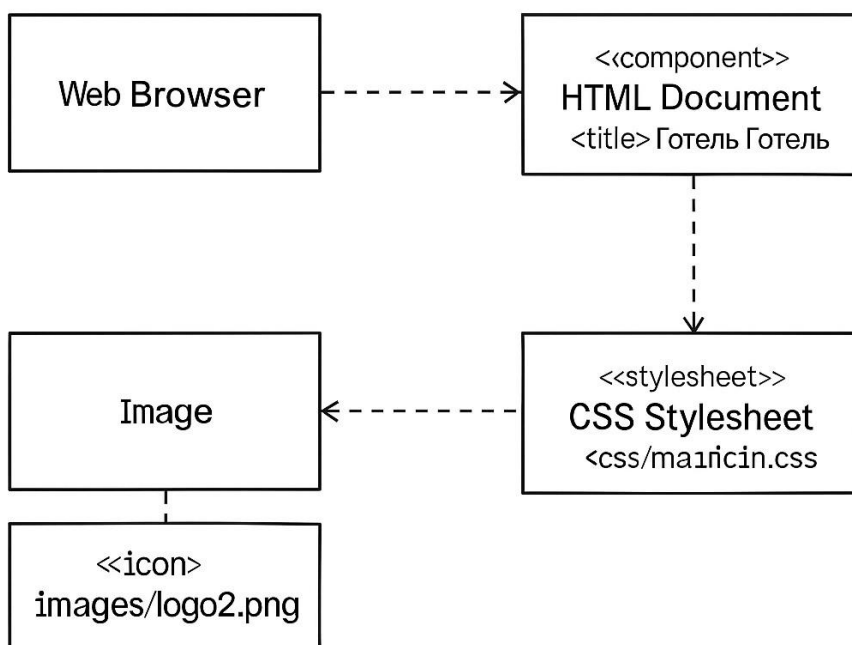


Рисунок В.3 - UML-діаграма компонентів WEB- ресурсу готелю

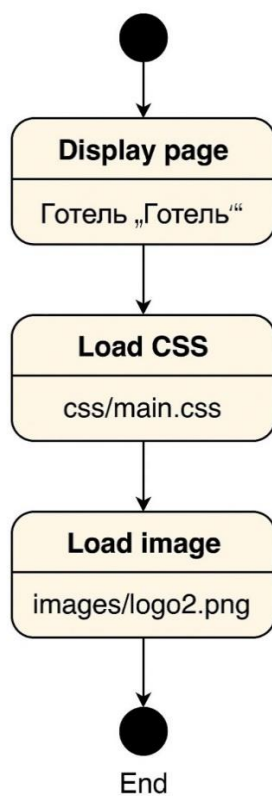


Рисунок В.4 - UML-діаграма активності WEB- ресурсу готелю

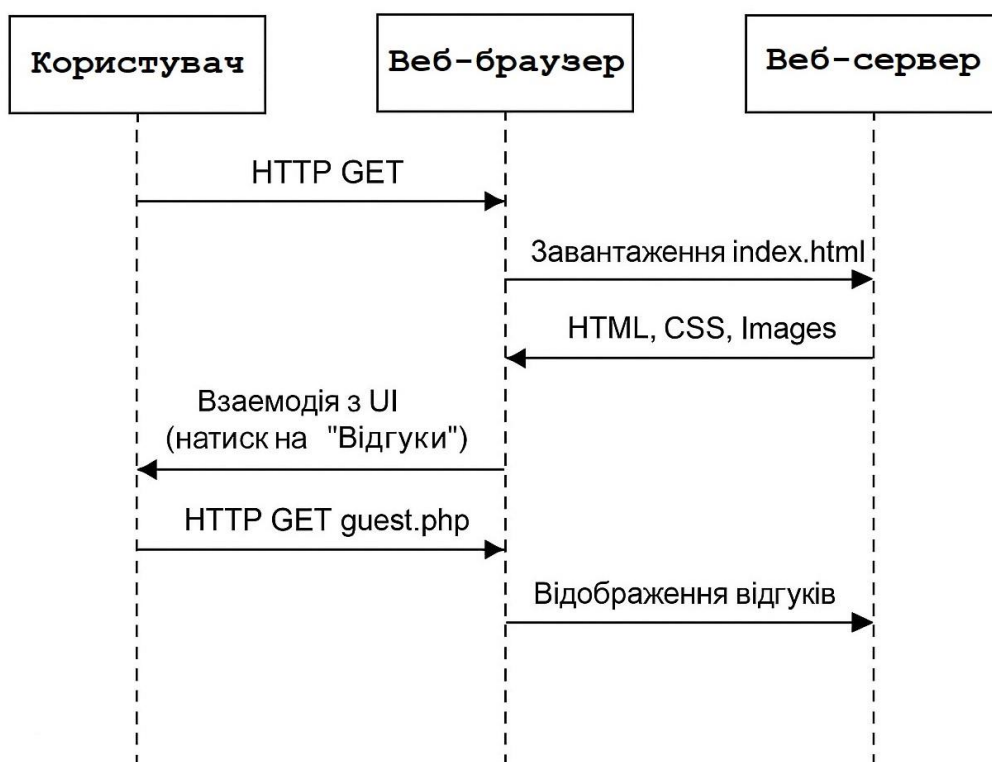


Рисунок В.5 - UML-діаграма послідовності WEB-ресурсу готелю



Рисунок В.6 – Сторінка з HTML5 і елементом canvas



Рисунок В.7 – Сторінка з HTML5 в старішій версії браузера



Рисунок В.8 – Лінійна структура сайту

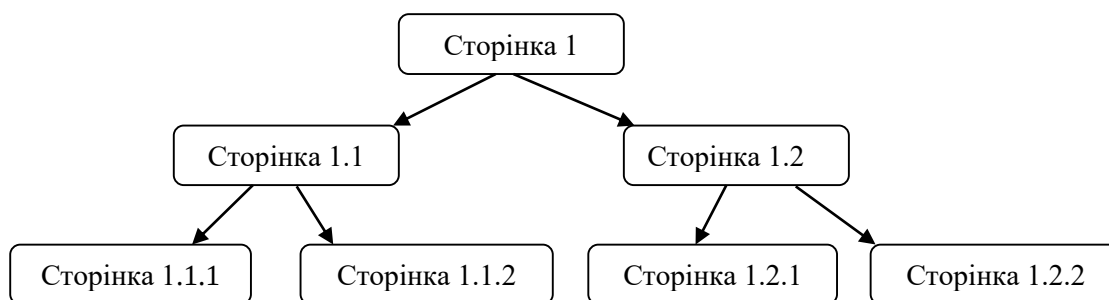


Рисунок В.9 – Ієрархічна структура сайту

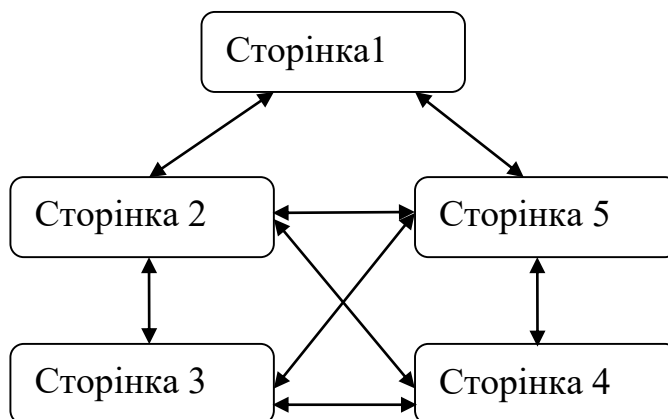


Рисунок В.10 – Мережна структура сайту

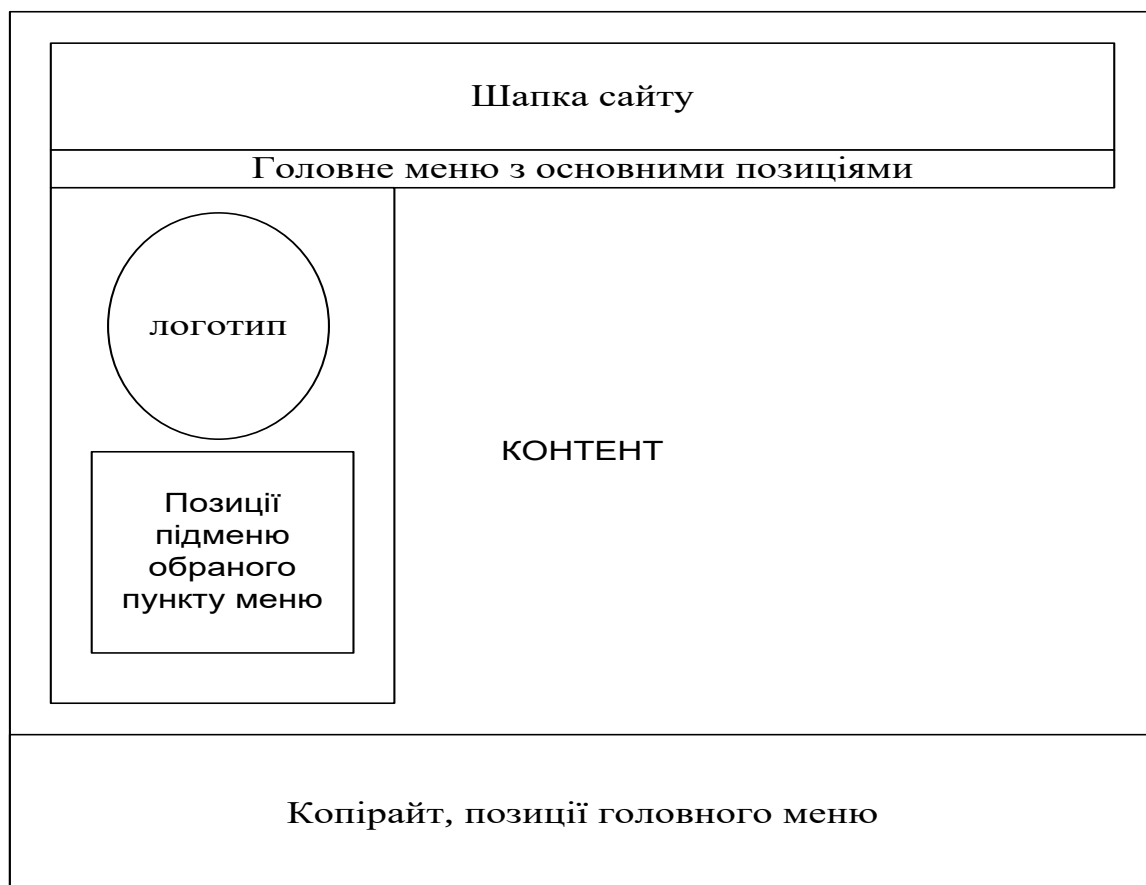


Рисунок В.11 – Схематичне представлення інтерфейсу сайту



Рисунок В.12 – Шапка сайту.



Рисунок В.13 – Коло кольорів Іттена



Рисунок В.14 – Обрані основні кольори за принципом «Тріади»



Рисунок В.15 – Додаткові кольори і напівтони



Рисунок В.16 - Головна сторінка сайту

Головна Про нас Кухня Послуги Номери Контакти

**ЛАСКАВО ПРОСИМО**

*Залиште свій відгук*

Відгуків 4, Сторінок 1

\*Ваше ім'я:

Е-пошта:

Home Page:

Місто:

\*Повідомлення (Не більше 1000 символів):

\* – поля обов'язкові для заповнення.

админ ()

Е-пошта:

Home page:

test

Повідомлення написано у неділю, 2 березня 2014 року, в 14 год. 52 хв.

Головна  
Про нас  
Кухня  
Послуги  
Номери  
Контакти

Copyright 2014, Вінницький національний політехнічний університет, 1П-13м, Денисенко Н.В.  
В питаннях реклами та співробітництва звертатися за зазначеним нижче контактом та за телефоном гарячої лінії (999) 560-546. Всі права захищені.  
Розробила: Tasha Mills

Рисунок В.17 - Гостьова книга сайту

Головна Про нас Кухня Послуги Номери Контакти

**ЛАСКАВО ПРОСИМО**

*Відправити повідомлення*

ВАШЕ ПОВІДОМЛЕННЯ :

ТЕКСТ ТЕКСТ ТЕКСТ

ІМ'Я:

Админ

Е-ПОШТА:

**Готово. Відправити!**

Головна  
Про нас  
Кухня  
Послуги  
Номери  
Контакти

Copyright 2014, Вінницький національний політехнічний університет, 1П-13м, Денисенко Н.В.  
В питаннях реклами та співробітництва звертатися за зазначеним нижче контактом та за телефоном гарячої лінії (999) 560-546. Всі права захищені.  
Розробила: Tasha Mills

Рисунок В.18 - Форма зворотного зв'язку на сайті



Рисунок В.19 - Сторінка «Кухня»



Рисунок В.20 - Вихід у підменю «Кухні»



Рисунок В.21 - Сторінка сайту

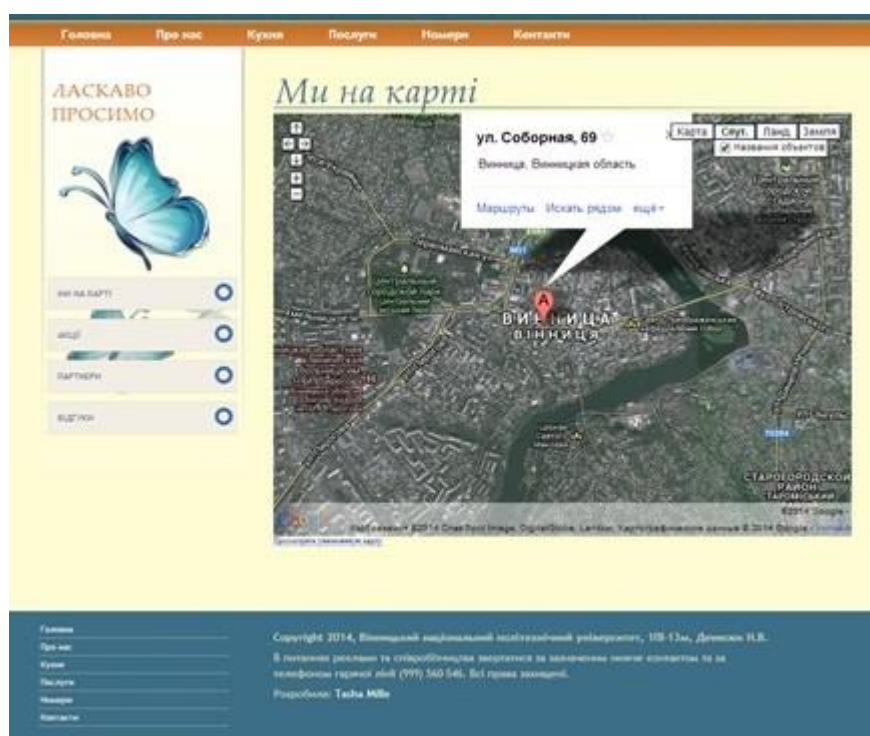


Рисунок В.22 - Карта Google на сайті

## ДОДАТОК Г (ДОВІДНИКОВИЙ) ІНСТРУКЦІЯ КОРИСТУВАЧА

Для доступу на головну сторінку, достатньо через браузер відкрити файл `index.html`. Результати відкриття сторінок сайту наведені в додатку Г.

На сайті представлено 6 головних сторінок (див. меню у верхній частині сайту), які мають посилання на дочірні сторінки та сторінки загальної важливості (такі як «Ми на карті», «Написати нам»). Всі сторінки пов'язані одна з одною і можна зручно переміщуватися між ними. На сторінці «Написати нам» є форма зворотного зв'язку, яка створена на CSS3. На сторінці «Відгуки» розміщена гостьова книга, створена за допомогою скриптів.

На таких сторінках, як «Кухня» і «Послуги» є додаткова панель меню, яка доступна лише в обраній вами під категорії. Тобто, обравши «Кухню» ви зможете роздивитися меню тої чи іншої кухні, але перейшовши на головну кнопку меню «Головна» ви втратите доступ до цього підменю.

У результаті досліджень та розробки реалізовано коди `index.html` та `guest.php` (додаток В).

*Код `index.html`.*

Створює головну сторінку веб-сайту готелю:

- відображає головну сторінку готелю;
- дає змогу переглядати інші сторінки через меню;
- містить навігацію, коротку інформацію про готель, стилізований зовнішній вигляд (через CSS);
- надає перехід на `guest.php` (PHP-код для відгуків).

Ось що він робить поетапно.

### 1. Оголошення HTML-документа

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 5.0  
Transitional//EN">
```

2. <head> — Встановлення метаданих сторінки.

```
<meta charset="UTF-8">
<title>Готель "Готель"</title>
<link href="css/main.css" rel="stylesheet">
```

Детально:

- встановлює кодування UTF-8;
- вказує заголовок у вкладці браузера;
- підключає CSS-файл зі стилями.

3. <body> — опис основної структури сторінки.

Верхній банер:

```
<div id="banner">
  
  <ul id="siteTools">...</ul>
</div>
```

Детально:

- виводить логотип готелю;
- показує інструменти: посилання на форму зворотного зв'язку, карту, поле пошуку.

Навігаційне меню:

```
<ul id="nav">...</ul>
```

Меню з переходами по сторінках: Головна, Про нас, Кухня, Послуги, Номери, Контакти.

Основний контент сторінки:

```
<div id="mainWrap">
  <div id="sidebar">...</div>
  <div id="content">...</div>
</div>
```

Ліва панель (sidebar) містить привітання і посилання на:

- карту,
- акції,
- партнерів,
- відгуки (guest.php).

Центральна частина (content) забезпечує:

- вивід текст-привітання з інформацією про місто Вінницю;
- фото міста (home\_photo.png);
- рекламний текст про готель та його зручності.

Футер (нижня частина сайту):

```
<div id="footer">
  <div id="col1">...</div>
  <div id="col2">...</div>
</div>
```

- колонка 1 - повторює меню навігації;
- колонка 2 – містить інформацію про авторські права, контактні дані, ім'я розробника.

*Код guest.php.*

Цей код - це HTML-сторінка готелю, яка поєднує в собі інтерфейс і PHP-логіку для гостьової книги (відгуків). Розберемося по частинах, що робить цей код:

#### 1. Заголовок сторінки (HTML + CSS)

```
<!DOCTYPE html>
<head>
<meta charset="UTF-8">
<title>Готель "Вінничина"</title>
<link href="css/main.css" rel="stylesheet">
```

```
<style> .prokrutka { height: 490px; width: 600px;
overflow: auto; } </style>
</head>
```

Визначає мета-дані, стилі, зовнішній CSS, та інлайн-стиль для блоку з прокруткою (відгуки).

## 2. Верхній банер та навігація

```

<ul>
    <li><a href="form.html">Написати нам</a></li>
    ...
</ul>
```

Відображає логотип і посилання: «написати», «карта», пошук.

## 3. Меню сайту

```
<ul id="nav">
    <li><a href="index.html">Головна</a></li>
    <li><a href="about.html">Про нас</a></li>
    ...
</ul>
```

Горизонтальне меню з навігацією по сайту.

## 4. Бічна панель

```
<div id="sidebar">
    <h2>Ласкаво просимо</h2>
    <a href="map.html">Ми на карті</a>
    ...
</div>
```

Кнопки переходу до розділів: карта, акції, партнери, відгуки.

## 5. PHP: Обробка форми гостьової книги.

Головні дії:

- перевірка та очищення введених даних (функція `check()`);

- валідація: ім'я, email, місто, повідомлення;
- збереження даних у файл `guest.csv`;
- переадресація після збереження.

```
if ($_POST['add']) {
    // перевірка обов'язкових полів
    // обрізання довгих полів
    // запис у guest.csv
}
```

## 6. HTML-форма для введення відгуків.

```
<form action=<?echo $_SERVER["SCRIPT_NAME"];?>
method="post">
    <input type="text" name="name">
    <input type="text" name="mail">
    <textarea name="mess"></textarea>
    <input type="submit" value="Відправити" name="add">
</form>
```

Форма для залишення відгуку: ім'я, e-mail, сторінка, місто, повідомлення.

## 7. PHP: Вивід збережених відгуків:

- читання з файлу `guest.csv`;
- підрахунок кількості записів і сторінок;
- формування сторінок і пагінація;
- виведення записів у таблиці.

```
$data = file($filename);
...
list($name, $mail, $url, $city, $time, $mess) =
explode("|", ...);
```

## 8. Дата написання відгуку

```
function datestring($date) {
```

формує повідомлення: "написано у понеділок, 1 травня 2025 року, в 14:23"