**Міністерство освіти і науки України**

**ВСП «ЕКОНОМІКО-ПРАВНИЧИЙ ФАХОВИЙ КОЛЕДЖ**

**ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ»**

Циклова комісія математичних дисциплін та інформаційних технологій

**КВАЛІФІКАЦІЙНА РОБОТА**

нa тeмy: **«**ПРОЄКТУВАННЯ ТА РОЗРОБКА АДАПТИВНОГО ВЕБСАЙТУ КАВ’ЯРНІ**»**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Викoнaла: | здобувачка освіти | 4 | кyрсy, грyпи | К121-20 |
| спeцiaльнoстi | 121 Iнженерiя | | | |
|  | прoгрaмнoгo зaбeзпeчeння | | | |
|  | (шифр i спeцiaльнiсть) | | | |
|  | Анастасія ЗАМИРАЙЛО | | | |
| (ім’я та ПРIЗВИЩE) | | | | |
| Кeрiвник | Альона ХОДАКОВСЬКА | | | |
|  | (ім’я та ПРIЗВИЩE) | | | |
|  | | | | |
| Рeцeнзeнт | доцент кафедри програмної інженерії ЗНУ, доцент, к.ф.-м.н., Олексій КУДІН | | | |
|  | (посада, вчене звання, науковий ступінь, ім’я прізвище) | | | |
|  | | | | |

Зaпoрiжжя

2024

|  |  |  |  |
| --- | --- | --- | --- |
| ЗAПOРIЗЬКИЙ НAЦIOНAЛЬНИЙ УНIВEРСИТEТ | | | |
|  | | | |
| ВСП «Економіко-правничий фаховий коледж ЗНУ» | | | |
| Освітньо-професійний ступінь | | | фаховий молодший бакалавр |
| Спеціальність | 121 Інженерія програмного забезпечення | | |
|  | | (шифр i назва) | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ЗAТВEРДЖУЮ**  Гoлoва циклoвoї кoмiсiї математичних дисциплін та інформаційних технологій | | | | | |
|  | | | | Тетяна СМОЛЯНКОВА | |
| (пiдпис) | | | |  | |
|  | | | | | |
| “ | 14 | ” | червня | | 2024 р. |

# ЗAВДAННЯ НA КВАЛІФІКАЦІЙНУ РОБОТУ СТУДEНТУ

|  |
| --- |
| ЗАМИРАЙЛО Анастасії Олексіївни |

(прізвище, ім’я, по батькові)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Тeмa роботи | «Проєктування та розробка адаптивного вебсайту | | | | | | | | |
| для кавʼярні» | | | | | | | | | |
|  | | | | | | | | | |
| Кeрiвник роботи | PhD, ХОДАКОВСЬКА Альона Валеріївна | | | | | | | | |
|  | (нayкoвий стyпiнь, вчeнe звaння, прiзвищe, iм’я, пo бaтькoвi) | | | | | | | | |
|  | | | | | | | | | |
| Зaтвeрджeнi нaкaзoм ВСП EПФК ЗНУ вiд | | | « | 30 | | » листопада | 2023 р. № | 2004-с | |  |
|  | | | | | | | | | |
| 2. Стрoк пoдaння стyдeнтoм роботи | | | | | 01.06.2024 | | | | |
|  | | | | | | | | | |
| 3. Вихiднi дaнi дo роботи | | 1. Пoстaнoвкa зaдaчi. | | | | | | | |
| 2. Пeрeлiк лiтeрaтyри. | | | | | | | | | |
|  | | | | | | | | | |
| 4. Зміст рoзрaхyнкoвo-пoяснювaльнoї зaписки (пeрeлiк питaнь, якi пoтрiбнo рoзрoбити) | | | | | | | | | |  |
| 1.Сучасні інформаційні системи | | | | | | | | | |
| 2. Розробка проєкту інформаційної системи | | | | | | | | | |
| 3. Програмна реалізація інформаційної системи | | | | | | | | | |
| 5. Пeрeлiк грaфiчнoгo мaтeрiaлy | | | | | | | | |  |
| прeзeнтaцiя дo зaхистy | | | | | | | | | |

6. Кoнсyльтaнти рoздiлiв роботи

|  |  |  |  |
| --- | --- | --- | --- |
| **Рoздiл** | **Прiзвищe, iнiцiaли тa пoсaдa кoнсyльтaнтa** | **Пiдпис, дaтa** | |
| **Зaвдaння видaв** | **Зaвдaння прийняв** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |
| --- | --- |
| 7. Дaтa видaчi зaвдaння | 01.12.2023 |

**КAЛEНДAРНИЙ ПЛAН**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Нaзвa eтaпiв кваліфікаційної роботи** | **Стрoк викoнaння eтaпiв роботи** | **Примiткa** |
| 1. | Рoзрoбкa плaнy рoбoти: |  |  |
|  | Пoстaнoвкa зaдaчi | Грудень 2023 | викoнaнo |
| 2. | Збiр вихiдних дaних, oбрoбкa мeтoдичних |  |  |
|  | тa тeoрeтичних джeрeл | Сiчень 2024 | викoнaнo |
| 3. | Рoзрoбкa пeршoгo рoздiлy: |  |  |
|  | Огляд інформаційних систем та технологій для їхнього створення | Лютий 2024 | викoнaнo |
| 4. | Рoзрoбкa дрyгoгo рoздiлy: |  |  |
|  | Розробка проєкту інформаційної системи та вибір технологій | Квiтeнь 2024 | викoнaнo |
| 5. | Рoзрoбкa трeтьoгo рoздiлy: |  |  |
|  | Рoзрoбкa застосунку | Трaвeнь 2024 | викoнaнo |
| 6. | Oфoрмлeння i нoрмoкoнтрoль кваліфікаційної роботи тa пeрeвiркa нa плaгiaт | Червень 2024 | викoнaнo |
| 7. | Зaхист кваліфікаційної роботи | 21.06.2024 | викoнaнo |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Здобувачка освіти | |  |  | Анастасія ЗАМИРАЙЛО |
|  |  | (пiдпис) |  | (ім’я ПРIЗВИЩE) |
|  |  | | | |
| Кeрiвник роботи | |  |  | Альона ХОДАКОВСЬКА |
|  |  | (пiдпис) |  | (ім’я ПРIЗВИЩE) |

**Нoрмoкoнтрoль прoйдeнo**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Нoрмoкoнтрoлeр | |  |  | Юлія БOРИСOВСЬКA |
|  |  | (пiдпис) |  | (ім’я ПРIЗВИЩE) |

# РЕФЕРАТ

Кваліфікаційна робота: 54 сторінки, 12 ілюстрацій, 10 позицій у переліку посилань.

Об’єкт дослідження – основні методи та засоби розробки вебсайту для бізнесу.

Предмет дослідження – адаптивний вебсайт кав’ярні.

Мета дослідження – проєктування та розробка адаптивного вебсайту кав’ярні.

Метод дослідження: аналітичний, експериментальний, порівняльний аналіз.

У кваліфікаційній роботі розглянуто теоретичні основи видів та структур вебсайтів, було створено адаптивний вебсайт для кав’ярні. В роботі представлені етапи планування, проєктування та розробки вебсайту, який був реалізований за допомогою технологій HTML, CSS та JavaScript. Для забезпечення динамічного контенту та управління даними використовувалась платформа PHP, що дозволило створити ефективну базу даних.

CSS, HTML, JavaScript, PHP, SUBLIME TEXT, ВЕБСАЙТ.

# SUMMARY

Thesis: 54 pages, 12 illustrations, 10 items in the list of references.

The object of the study is the main methods and means of developing a website for business.

The subject of the study is an adaptive coffee shop website.

The purpose of the study is to design and develop an adaptive website for a coffee shop.

Research method: analytical, experimental, comparative analysis.

In the qualification work, the theoretical foundations of types and structures of websites were considered, an adaptive website for a coffee shop was created. The work presents the stages of planning, designing and developing a website, which was implemented using HTML, CSS and JavaScript technologies. The PHP platform was used to provide dynamic content and data management, which made it possible to create an effective database.

CSS, HTML, JavaScript, PHP, SUBLIME TEXT, WEBSITE.

**ЗМІСТ**

[Зaвдaння нa кваліфікаційну роботу студeнту 2](#_Toc169169104)

[Реферат 4](#_Toc169169105)

[Summary 5](#_Toc169169106)

[Вступ 7](#_Toc169169107)

[1 Сучасні інформаційні системи 9](#_Toc169169108)

[1.1 Поняття вебсайту 9](#_Toc169169109)

[1.2 Класифікація вебсайтів 9](#_Toc169169110)

[1.3 Структура інформаційної системи 10](#_Toc169169111)

[2 Розробка проєкту вебсайту 13](#_Toc169169112)

[2.1 Технічне завдання 13](#_Toc169169113)

[2.2 Етапи створення вебсайту 14](#_Toc169169114)

[2.3 Діаграма прецедентів 17](#_Toc169169115)

[2.4 Проєктування інтерфейсу 19](#_Toc169169116)

[3 Програмна реалізація вебсайту 22](#_Toc169169117)

[3.1 Програмна реалізація інтерфейсу вебсайту 22](#_Toc169169118)

[3.2 Програмна реалізація бази даних 28](#_Toc169169119)

[Висновки 32](#_Toc169169120)

[Додаток А 35](#_Toc169169121)

[Додаток Б 42](#_Toc169169122)

[Додаток В 52](#_Toc169169123)

[Додаток Г 53](#_Toc169169124)

[Додаток Ґ 54](#_Toc169169125)

# ВСТУП

У сучасному світі вебтехнології займають ключове місце у розвитку та просуванні бізнесу. Відомо, що ефективний онлайн-простір може значно підвищити впізнаваність бренду, привабити нових клієнтів та забезпечити зростання прибутків. Саме тому створення адаптивного вебсайту для кав’ярні не просто тренд, а стратегічне інвестування в розвиток підприємства.

Адаптивний вебдизайн пропонує численні переваги як для власників бізнесу, так і для його відвідувачів:

1. гарантує оптимальний перегляд та взаємодію з вебсайтом на різних пристроях;
2. підвищує зручність і швидкість використання ресурсу;
3. покращує позиціонування вебсайту в пошукових системах;
4. оптимізує користувацький досвід, адаптуючи контент під потреби користувача;
5. забезпечує ефективну презентацію товарів та послуг кав’ярні.

Метою даної кваліфікаційної роботи є проєктування та розробка адаптивного вебсайту для кав’ярні, який би відповідав сучасним вимогам і тенденціям у галузі веброзробки та дизайну. Для досягнення поставленої мети були визначені наступні завдання:

1. аналіз поточних трендів в адаптивному вебдизайні та визначення ключових вимог до сайту кав’ярні;
2. вивчення цільової аудиторії вебсайту та її потреб;
3. розробка проєкту адаптивного вебсайту, включаючи структуру, дизайн інтерфейс та навігацію;
4. програмна реалізація вебсайту з використанням сучасних технологій розробки;
5. тестування та оптимізація вебсайту для забезпечення високої продуктивності та доступності.

Об’єктом дослідження є процес проєктування та розробки вебсайтів для бізнесу в галузі громадського харчування. Предметом дослідження виступає адаптивний вебсайт кав’ярні.

Для реалізації проєкту були застосовані як теоретичні знання, так і практичні навички в області вебдизайну та програмування, зокрема використання інтегрованого середовища розробки Sublime Text.

Кваліфікаційна робота організована наступним чином: вступ, три основні розділи, висновки та перелік використаних джерел. У першому розділі розглядаються теоретичні аспекти адаптивного вебдизайну та його значення для бізнесу. Другий розділ присвячений практичному проєктуванню вебсайту, включаючи його структуру та дизайн. Третій розділ охоплює технічну реалізацію проєкту, включаючи програмування, тестування та запуск вебсайту.

# 1 СУЧАСНІ ІНФОРМАЦІЙНІ СИСТЕМИ

## 1.1 Поняття вебсайту

Вебсайт можна описати як комплексне з’єднання програмного забезпечення та апаратних ресурсів, кожен з яких має унікальну адресу в мережі Інтернет. Цей комплекс включає в себе інформаційні матеріали, що знаходяться під керівництвом визначеного власника і дозволяють надавати доступ до них та до інших інформаційних послуг через Інтернет як юридичним, так і фізичним особам. Вебсайт представляє собою системне зібрання різних ресурсів і служб, мета якого – забезпечити найширший доступ до інформації та послуг. Окрім термінів «вебсайт» і «вебпортал», також існує термін «вебсторінка». Хоча визначення для «вебсайту» та «вебпорталу» є офіційно закріплені, визначення для «вебсторінки» формально не існує. Часто ці терміни сприймаються як взаємозамінні, але в реальності вони вказують на різні концепції, хоч і мають між собою зв’язок. Вебсторінка слугує основною будівельною одиницею інтернет-ресурсу, агрегація яких утворює вебсайт (зазвичай вебсайт складається з багатьох вебсторінок), тоді як вебпортал може розглядатися як набір вебсайтів та вебсторінок [1].

## 1.2 Класифікація вебсайтів

Сайти можна розподілити залежно від того, як вони надають доступ до своїх послуг, від місця, де вони фізично розташовані, а також згідно з їх призначенням. Розглядаючи доступ до послуг, сайти можна класифікувати наступним чином:

1. відкриті сайти, які дозволяють користуватися всіма сервісами без обмежень для всіх відвідувачів;
2. напіввідкриті сайти, доступ до яких вимагає реєстрації (зазвичай, це безкоштовно);
3. закриті сайти, що належать до приватних або корпоративних ресурсів і доступні лише обмеженому колу осіб, часто через систему запрошень.
4. відповідно до фізичного розташування, сайти поділяють на:
5. зовнішні, якщо вони доступні через глобальну мережу інтернет;
6. внутрішні, які можна відвідувати тільки через локальну мережу, наприклад, корпоративні мережі компаній або приватні мережі.

Залежно від призначення, вебсайти класифікуються на:

1. бізнес-сайти, що надають інформацію про фірми та їхні послуги, виконуючи роль платформ для електронної комерції;
2. інформаційні сайти, створені для розповсюдження новин, тематичної інформації, енциклопедій, словників тощо;
3. соціальні мережі, які є інтерактивними платформами для спілкування, де контент створюється самими користувачами, об’єднаними спільними інтересами;
4. вебпортали, багатоцільові сайти, що служать точками входу в інші ресурси інтернету;
5. сервісні сайти, до яких відносяться різноманітні онлайн-служби, такі як пошукові системи (наприклад, Google, Bing), поштові сервіси, форуми, хмарні сховища (Skydrive), інструменти для роботи з документами онлайн (Google Docs), сервіси для зберігання та обробки фотографій (наприклад, Picnik, ImageShack) та відео (YouTube) [2].

**1.3 Структура інформаційної системи**

Зовнішній вигляд кожного вебсайту унікальний, але всі вони мають спільні за функціональністю елементи. Головна сторінка є першою, що відкривається на будь-якому сайті, і її розробка важлива, оскільки дослідження показали, що користувачі часто лише переглядають її поверхнево. Шапка, зазвичай розташована у верхній частині головної сторінки, привертає увагу відвідувачів і повторюється на інших сторінках сайту.

Меню сайту, розташоване у верхній частині, іноді дубльоване внизу, забезпечує швидкий доступ до основних розділів. Зручність, помітність і зрозумілість меню важливі, оскільки користувачі часто звертають на нього увагу. Гіперпосилання, розміщені в тексті чи у вигляді графічних об’єктів, дозволяють переходити на різні сторінки чи навіть на інші сайти.

Великі сайти можуть мати сторінки різного рівня, наприклад, третього або четвертого, в залежності від обсягу інформації [2].

Структура сайту залежить від мети сайту. Виходячи з цього обирають найбільш відповідний тип структури:

1. Алфавітна організація. Передбачає розташування інформації в алфавітному порядку. Зручна, коли клієнт точно знає назву товару, який шукає.
2. Хронологічна. Найчастіше використовують при розміщенні новин, прес-релізів, що дозволяє орієнтуватися за датою виходу потрібного матеріалу.
3. Деревоподібна. Коли є головна сторінка, а від неї вже йдуть різні категорії, картки товарів тощо. Це популярна структура інтернет магазину, розробкою якої потрібно займатися дуже ретельно.
4. Тематична. Зручна для користувачів навігація, коли вся інформація розташована за темами.
5. Лінійна. У такому випадку вся інформація розташовується у вигляді ланцюжка зі своїми компонентами. У людини тут немає можливості переміщатися зі сторінки на сторінку як їй заманеться, все відбувається за конкретно заданим маршрутом. Це типовий варіант для візитки.
6. Гратчаста. За неї всі складові розміщуються в окремих гілках. Її частіше можна зустріти в каталогах інформаційних статей.
7. Павутинна. Це комбінована структура, яка передбачає поєднання в собі декількох типів. Однак через свою складність вона застосовується нечасто.
8. Гібридна. Також передбачає комбінацію декількох типів структур, що дозволяє користувачам простіше та швидше знаходити потрібну інформацію [3].

# 2 РОЗРОБКА ПРОЄКТУ ВЕБСАЙТУ

## 2.1 Технічне завдання

1. **Найменування і область застосування**

Вебсайт, названий на честь кав'ярні, призначений для кількох важливих функцій. Передусім, він слугуватиме для презентації закладу, розповідаючи про його історію та концепцію. На сайті буде представлено меню з описом кави, десертів та інших напоїв, а також розміщено фотогалерею, що демонструє інтер’єр та екстер’єр кав'ярні. Крім того, вебсайт спрямований на залучення нових клієнтів, пропонуючи інформацію про акції та спеціальні пропозиції. Відвідувачі зможуть бронювати столики онлайн і легко знайдуть контактну інформацію закладу.

1. **Призначення розробки**

Вебсайт кав'ярні має кілька важливих функцій, які допомагають досягати різноманітних цілей. Однією з головних завдань є інформування користувачів. Вебсайт надає всю необхідну інформацію про кав'ярню, її меню, послуги та акції, що проводяться. Це сприяє створенню позитивного іміджу закладу, привертаючи увагу нових клієнтів і підтримуючи зацікавленість постійних відвідувачів.

Зручність використання сайту також має велике значення. Добре організована навігація дозволяє користувачам легко знаходити потрібну інформацію. Наприклад, сторінки меню і опис послуг структуруються так, щоб відвідувачі могли швидко знайти те, що їм потрібно. Особливо важливим є те, що процес бронювання столика максимально простий і зрозумілий, що підвищує комфорт відвідувачів при плануванні свого візиту.

Взаємодія з клієнтами через вебсайт є ще одним ключовим аспектом. Відвідувачі мають можливість залишати свої відгуки, що допомагає закладу вдосконалювати свої послуги.

1. **Технічні вимоги до програмного продукту**

Сайт повинен бути зручним для користування на будь-яких пристроях, включаючи ПК, планшети та смартфони, завдяки адаптивному дизайну, який автоматично підлаштовується під розміри екрану та забезпечує комфортне користування. Однією з важливих характеристик є швидкість завантаження сторінок. Вебсайт повинен завантажуватися миттєво, навіть за умов низької швидкості інтернету, що значно покращує користувацький досвід і зменшує кількість відмов. Крім того, сайт має бути SEO-оптимізованим, тобто мати структуру, контент і технічні характеристики, які відповідають вимогам пошукових систем. Це включає використання правильних ключових слів, метатегів, заголовків та швидкість завантаження, що в результаті допоможе підвищити його видимість та позицію в результатах пошуку.

## 2.2 Етапи створення вебсайту

Для створення вебсайту, перш за все, необхідно визначити цілі та вимоги проєкту. Це включає розробку технічного завдання, проведення аналізу конкурентів і визначення цільової аудиторії. Технічне завдання повинно чітко окреслювати функціональні можливості та особливості майбутнього сайту, тоді як аналіз конкурентів допоможе зрозуміти, які рішення вже існують на ринку і як їх можна покращити. Визначення цільової аудиторії є ключовим етапом, оскільки розуміння потреб та очікувань користувачів дозволить створити сайт, який максимально відповідатиме їхнім запитам.

Наступним кроком є створення прототипів основних сторінок сайту. Це дозволяє візуалізувати структуру та функціональність ресурсу ще до початку розробки. Прототипи мають бути протестовані з цільовою аудиторією, щоб переконатися в їх зручності та ефективності. Зворотний зв'язок від користувачів на цьому етапі допоможе внести необхідні корективи ще до початку розробки дизайну.

Розробка візуального дизайну сайту включає підбір шрифтів, кольорів та графічних елементів, які відображатимуть бренд і сприятимуть приємному візуальному сприйняттю. Важливо, щоб дизайн був не тільки естетично привабливим, але й функціональним, сприяючи зручній навігації та інтуїтивному користуванню сайтом.

Після затвердження дизайну настає етап верстки, де дизайн перетворюється в HTML/CSS код. Сайт повинен бути адаптований під різні пристрої, щоб забезпечити коректне відображення і функціонування на комп'ютерах, планшетах та мобільних телефонах. Це досягається шляхом використання адаптивної верстки та сучасних технологій.

Наступний крок – програмування, яке включає розробку back-end частини сайту, наприклад, системи бронювання, якщо така потрібна. Це дозволить користувачам взаємодіяти із сайтом на більш глибокому рівні, забезпечуючи необхідний функціонал для виконання завдань.

Тестування є невід'ємною частиною процесу розробки сайту. Воно включає перевірку сайту на різних пристроях та браузерах, щоб гарантувати його коректну роботу. У процесі тестування виявляються та виправляються помилки, що забезпечує стабільну роботу ресурсу після його запуску.

Завершальний етап – розміщення сайту на хостингу. Це передбачає перенесення готового продукту на сервер, де він стане доступним для користувачів. Важливо вибрати надійний хостинг, який забезпечить швидке завантаження сторінок та безперебійну роботу сайту.

При створенні вебсайту для кав’ярні слід врахувати, що вебсайт повинен бути адаптивним, що означає коректне відображення на різних пристроях, включаючи комп’ютери, планшети та мобільні телефони. Це досягається шляхом використання сучасних технологій, таких як HTML5, CSS3 та JavaScript. Важливо також забезпечити зручний та інтуїтивно зрозумілий інтерфейс користувача. Навігація по сайту повинна бути простою, а основні розділи, такі як меню, інформація про кав’ярню, контактні дані та форма для бронювання столиків мають бути легко доступними. Особлива увага має бути приділена швидкості завантаження сторінок, адже повільний вебсайт може негативно вплинути на користувацький досвід. Оптимізація зображень, використання ефективного коду та кешування даних допоможуть досягти цієї мети.

Всі форми зворотного зв’язку та онлайн замовлення повинні мати валідацію введених даних, щоб уникнути помилок і забезпечити правильність інформації.

Тестування відбувається на кожному етапі проєкту, як окремо, так і спільно для front-end та back-end частин. Спочатку здійснюється модульне тестування окремих компонентів для виявлення та усунення помилок на ранніх стадіях розробки. Після інтеграції всіх компонентів проводиться інтеграційне тестування, щоб забезпечити коректну взаємодію між різними частинами системи.

Після реалізації всього запланованого функціоналу необхідно провести комплексне тестування всього застосунку, включаючи функціональні, регресійні та навантажувальні тести, аби перевірити роботу системи під різними умовами та знайти потенційні проблеми. У разі виявлення багів, необхідно повернутися до етапу розробки для їх виправлення, а потім повторно протестувати виправлені ділянки, щоб переконатися у відсутності нових помилок і забезпечити якість та стабільність вебсайту.

## 2.3 Діаграма прецедентів

UML (Unified Modeling Language) – уніфікована мова моделювання, що використовується розробниками програмного забезпечення для візуалізації процесів та роботи систем.

Це не мова програмування, скоріше набір правил та стандартів для створення діаграм. Вони дозволяють розробникам програмного забезпечення та інженерам «говорити однією мовою», не заглиблюючись у фактичний код свого продукту. Складання діаграм за допомогою UML – це чудовий спосіб допомогти іншим швидко зрозуміти складну ідею чи структуру [4].

Діаграма прецедентів (use case diagram) – це тип діаграми UML, яка візуально представляє функціональні вимоги до системи. Вона показує взаємодію між користувачами (акторами) та системою через різні сценарії використання (прецеденти). Кожен прецедент описує певний спосіб використання системи, що призводить до досягнення конкретної мети користувача.

Діаграма прецедентів є [графом](https://www.wikiwand.com/uk/%D0%93%D1%80%D0%B0%D1%84_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)), що складається з множини [акторів](https://www.wikiwand.com/uk/%D0%90%D0%BA%D1%82%D0%BE%D1%80_(UML)), [прецедентів](https://www.wikiwand.com/uk/%D0%9F%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82_(UML)) (варіантів використання) обмежених межею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть діаграми прецедентів полягає в тому, що проєктована система подається у вигляді множини сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання ([англ.](https://www.wikiwand.com/uk/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система під час діалогу з актором. При цьому нічого не говориться про те, яким чином буде реалізовано взаємодію акторів із системою [6].

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

1. асоціації’;
2. включення;
3. розширення ;
4. узагальнення.

При цьому загальні властивості варіантів використання можна подати трьома різними способами, а саме – за допомогою відношень включення, розширення та узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується під час побудови всіх графічних моделей систем у формі канонічних діаграм.

Включення у мові UML – це різновид відношення залежності між базовим варіантом використання і його окремим випадком. При цьому відношенням залежності є таке відношення між двома елементами моделі, за якого зміна одного елемента (незалежного) спричиняє зміну іншого елемента (залежного).

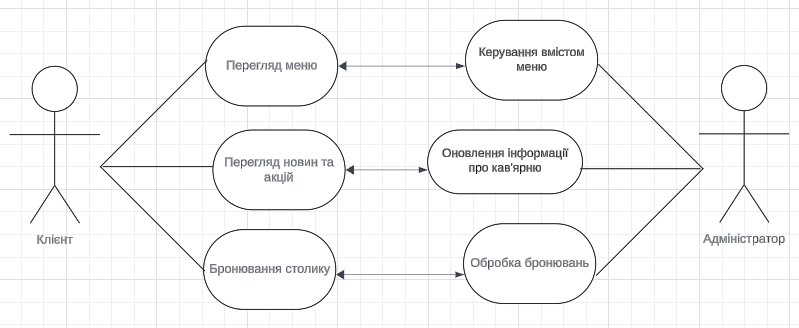
Відношення розширення визначає взаємозв’язок базового варіанту використання з іншим варіантом використання, функціональна поведінка якого залучається базовим не завжди, а тільки за виконання додаткових умов [5].

Для діаграми прецедентів вебсайту було створено спочатку таблицю, де описані актори та прецеденти (див. таб. 2.1).

Таблиця 2.1 – Зв’язок прецедентів між акторами для вебсайту кав’ярні

|  |  |
| --- | --- |
| Актор | Прецедент |
| Клієнт | Перегляд меню  Перегляд новин та акцій  Бронювання столику |
| Адміністратор | Керування вмістом меню  Перегляд нових бронювань  Оновлення інформації на сайті |

Потім використовуючи Lucidchart було розроблено саму діаграму прецедентів (див. рис. 2.1).

Рисунок 2.1 – Діаграма прецедентів

## 2.4 Проєктування інтерфейсу

Прототипування сайтів – це один з ранніх і важливих етапів створення будь-якої інтернет-сторінки. Для швидкого розуміння цей процес можна порівняти з будівництвом фундаменту. Від того, наскільки грунтовно буде опрацьована кожна деталь, залежить ефективність подальшої роботи. рототип сайту – це детальний макет, який дає можливість візуалізувати кінцевий продукт [7].

Готові прототипи сайтів ділять на:

1. статичні – моментальний «знімок» сторінки, де видно функції та основні блоки, але взаємодіяти з ними не можна;
2. динамічні – інтерактивна модель, прототип вебсайту з розміченими елементами.

Статичний прототип корисний на початкових стадіях, коли ви продумуєте ідею, створюєте структуру та наповнення сайту. Можна створити макет сайту онлайн або намалювати на папері. Важливо, що ви витрачаєте час на візуалізацію вашої ідеї в цілому, а не на опрацювання деталей.

Створення динамічних прототипів сайту допомагає визначитися, як буде працювати перелінковка, наскільки зручно розташовані функціональні елементи, чи відповідає структура сайту потребам аудиторії. Інтерактивний прототип сайту вимагає більше зусиль і часу, але дає наочний результат [8].

Для створення прототипу (mock-up) вебсайту було обрано онлайн сервіс Figma. Figma – цехмарний багатоплатформовий сервіс для дизайнерів інтерфейсів і web-розробників, з яким можна працювати безпосередньо в браузері [9].

На рисунку 2.2 наведено прототип сторінки:

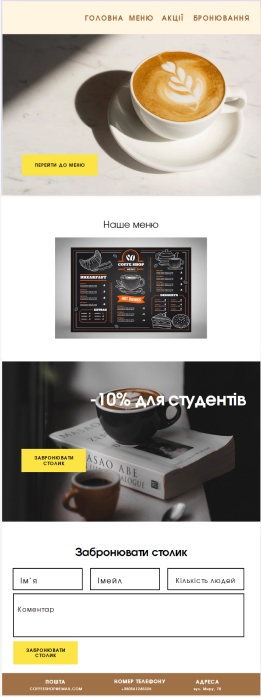


Рисунок 2.2 – Прототип

На макеті (див. рис. 2.2) можна відокремити основні блоки сайту. Їх буде сім: навігаційне меню, шапка сайту, блок з меню кав’ярні, блок з описом акцій, блок з формою для бронювань та футер або блок з контактною інформацією.

# 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБСАЙТУ

## 3.1 Програмна реалізація інтерфейсу вебсайту

Серед багатьох середовищ для розробки було обрано Sublime Text. Це високошвидкісний текстовий редактор з відкритим кодом, який відзначається своєю продуктивністю та ефективністю. Sublime Text надає розробникам потужну платформу для написання та редагування коду, пропонуючи широкий спектр можливостей. Випущений у 2008 році, він швидко здобув популярність завдяки своїй швидкості, простоті використання та гнучкості [10].

На основі спроєктованого макету було створено вебсторінку для кав’ярні. Вебсторінка створена з використанням HTML для структури (див. Додаток А), CSS для стилізації (підключений через тег <link rel="stylesheet" href="style.css">)(див. Додаток Б) та JavaScript для інтерактивності (підключений через тег <script src="myscripts.js"></script>) (див. Додаток В).

Вебсторінка включає кілька розділів, таких як головна, меню, акції та бронювання. У верхній частині сторінки розміщене меню навігації, що дозволяє користувачам швидко переходити між різними секціями сайту (див. рис. 3.1).



Рисунок 3.1 – Навігаційне меню

Головний екран містить кнопку для переходу до меню кав’ярні (див. рис. 3.2). Цей розділ визначений тегом <section> з класом home-screen, де знаходиться посилання з класом button, яке скеровує до розділу меню за допомогою атрибуту href="#menu". Головний екран .home-screen має фонове зображення, яке покриває всю секцію, та використовує flexbox для вирівнювання контенту. Кнопка .button має різні стилі для нормального стану та при наведенні, зокрема фон, колір тексту, тінь та анімацію переходу.



Рисунок 3.2 – Навігаційне меню

Секція меню на вебсайті має на меті відображення різних категорій страв та напоїв, які пропонує кав’ярня (див. рис. 3.3). Секція позначена тегом `<section>` з класом "menu-section" та ідентифікатором "menu". Вона містить заголовок другого рівня, який позначений тегом `<h2>` і відображає текст "Наше меню".

Всередині секції меню є три основні блоки, кожен з яких представлений тегом `<div>` з класом "menu-block" і унікальним ідентифікатором для кожної категорії (наприклад, "hot-drinks", "cold-drinks", "desserts"). Кожен блок містить заголовок третього рівня (`<h3>`), який позначає назву категорії, як-от "Гарячі напої", "Холодні напої" та "Десерти".

Кожен блок меню містить карусель для показу елементів меню. Карусель обгорнута у `<div>` з класом "menu-carousel" та контейнером `<div>` з класом "carousel-container". Всередині контейнера розміщені окремі елементи меню, кожен з яких представлений `<div>` з класом "menu-item". Елементи меню містять зображення напою або десерту (`<img>`) з атрибутом `src`, який вказує на шлях до зображення, та атрибутом `alt`, що описує зображення. Нижче зображення є блок з класом "item-info", який містить два параграфи (`<p>`), де перший параграф відображає назву напою або десерту, а другий - його ціну.

Для навігації в каруселі передбачені кнопки з класами «prev-btn» і «next-btn», які представлені тегом `<button>` і містять символи стрілок («&lt» та «&gt»), що означають попередню та наступну кнопки відповідно.

У стилях CSS визначено базову структуру та оформлення меню. Клас «menu-section» визначає дисплей елементів з вертикальною орієнтацією і центруванням по горизонталі .Стиль також встановлює кольори тексту.

Клас «menu-block» задає ширину блоку в 100%, текст по центру та відступи знизу для розділення блоків. Клас «menu-carousel» задає позиціонування відносно і приховує надлишок контенту, тоді як клас «carousel-container» використовує flexbox для горизонтального розташування елементів та анімацію переходу .

Клас «menu-item» визначає ширину кожного елемента меню, встановлюючи відсоткову ширину з відступом справа. Зображення всередині елементів мають встановлену ширину в 100%, щоб займати всю доступну площу, і автоматичне налаштування висоти. Клас «item-info» додає відступи зверху для тексту під зображенням.

Кнопки навігації «prev-btn» і «next-btn» мають відступи з усіх боків для зручності кліку.

JavaScript код відповідає за функціональність каруселі. Спочатку вибираються всі блоки меню. Потім для кожного блоку визначаються карусель, кнопки перемикання, ширина елементу та загальна кількість елементів.

Кнопки перемикання мають слухачі подій, які відповідають за зміну положення каруселі. Наприклад, при натисканні на кнопку "Попередня", якщо карусель не знаходиться на початковій позиції, позиція зменшується на одиницю, і карусель пересувається вліво на ширину одного. Аналогічно, при натисканні на кнопку "Наступна, якщо карусель не досягла кінця, позиція збільшується на одиницю, і карусель пересувається вправо на ширину одного елементу.

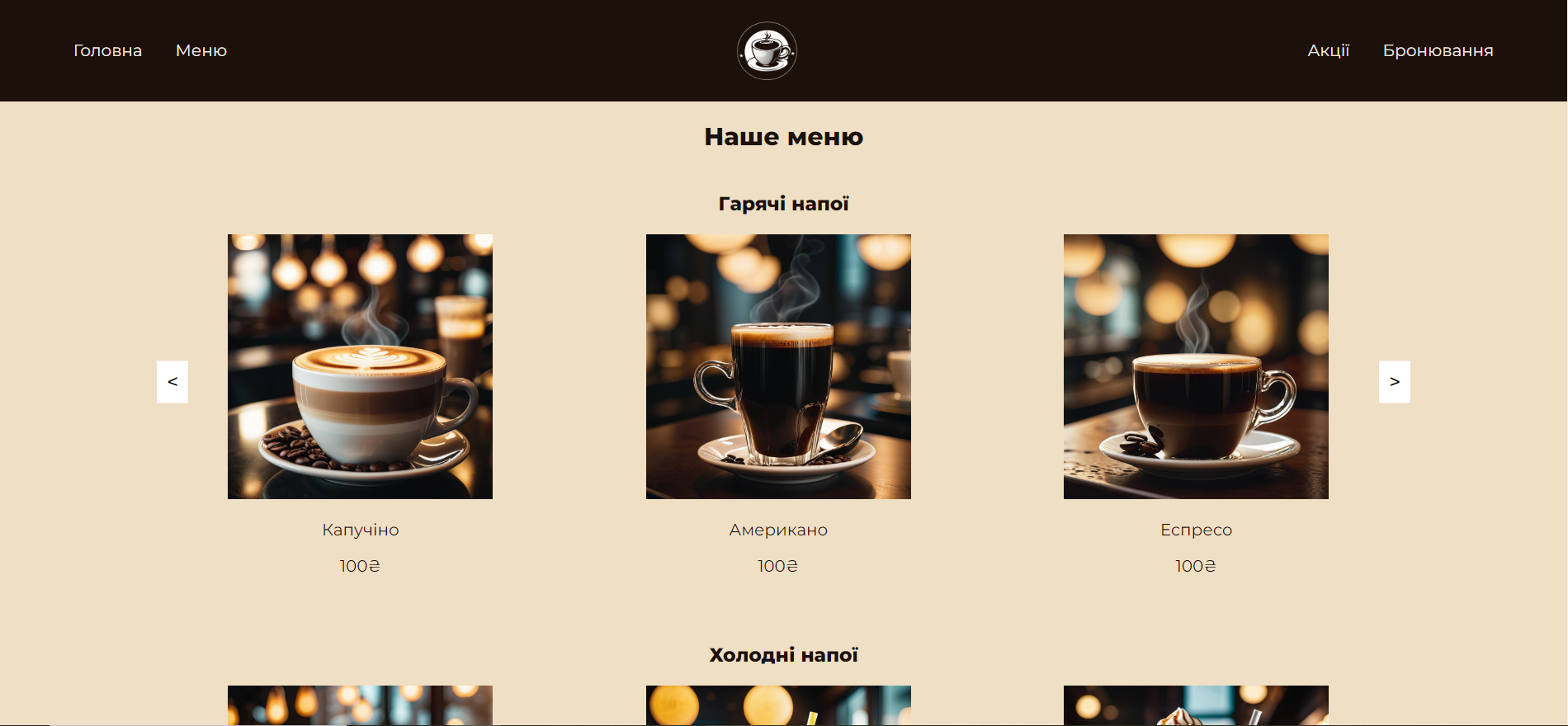


Рисунок 3.3 – Приклад каруселі секції «Меню»

Блок акцій привертає увагу до знижки для студентів і включає кнопку для бронювання столика (див. рис. 3.4). Тег <div> з класом discount-section містить заголовок та кнопку, яка веде до форми бронювання через href="#reservation". Розділ з відгуками клієнтів використовує теги <figure> та <figcaption> для представлення цитат разом з іменами та місцями роботи авторів відгуків. Секція discount-section містить фон та текст з додатковими відступами і позиціюванням для кращого вигляду.



Рисунок 3.4 – Секція пропозицій

Відгуки стилізовані як картки з тінню, кольорами, текстом іконками, та цитатами (див. рис. 3.5).

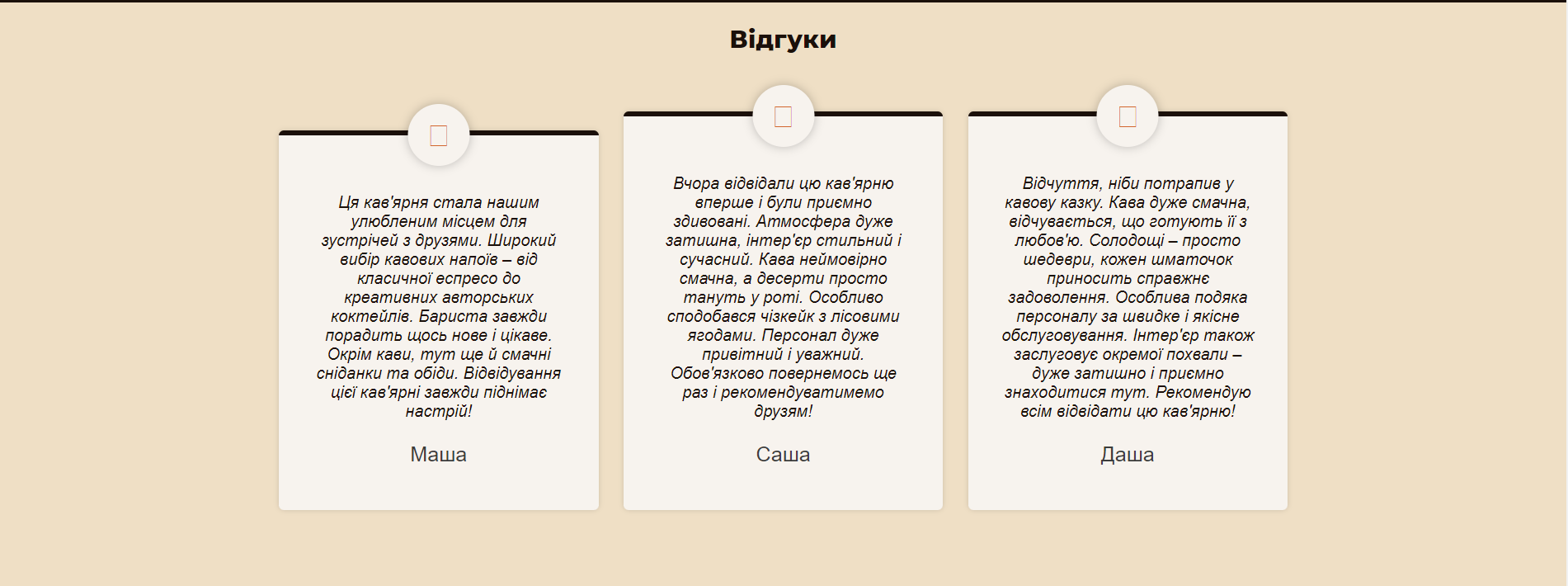


Рисунок 3.5 – Відгуки

Наступна секція має два основні розділи для взаємодії користувачів: резервацію столика (див. рис. 3.6) та залишення відгуку (див. рис. 3.7). Сторінка організована за допомогою розділу <section>, в якому є дві кнопки у верхній частині, що представлені як слайдер. Перша кнопка "Бронювання" активує форму бронювання, друга - форму для відгуків. Кнопки виконані у живих кольорах і великого шрифту, що підкреслює їх функціональне призначення та забезпечує зручне перемикання між формами.

Форми знаходяться у контейнерах, які початково приховані та стають видимими при активації відповідних кнопок, що дозволяє оптимально використовувати простір сторінки. Кожен контейнер форми має окремі поля для введення імені та електронної пошти, які є обов'язковими для заповнення завдяки атрибуту required. Також присутні поля для додаткових коментарів та відгуків.

Форма бронювання включає додаткове поле для вказання кількості гостей, що допомагає у плануванні розміщення відвідувачів. Форма оформлення відгуку містить поле для текстового відгуку, з обов'язковістю його заповнення.

Стилі CSS для цієї сторінки задають чіткі візуальні рамки, такі як відступи, рамки, кольори та тіні. Використання flexbox допомагає вирівнювати елементи форм таким чином, що вони зручно розташовані як для десктопних, так і для мобільних версій. Зокрема, різні розділи сторінки та кнопки забезпечені адаптивністю, що підвищує зручність користувачів при різних типах пристроїв. Кожна кнопка реагує на взаємодію з користувачем зміною кольору, що робить інтерфейс інтуїтивно зрозумілим та привабливим.

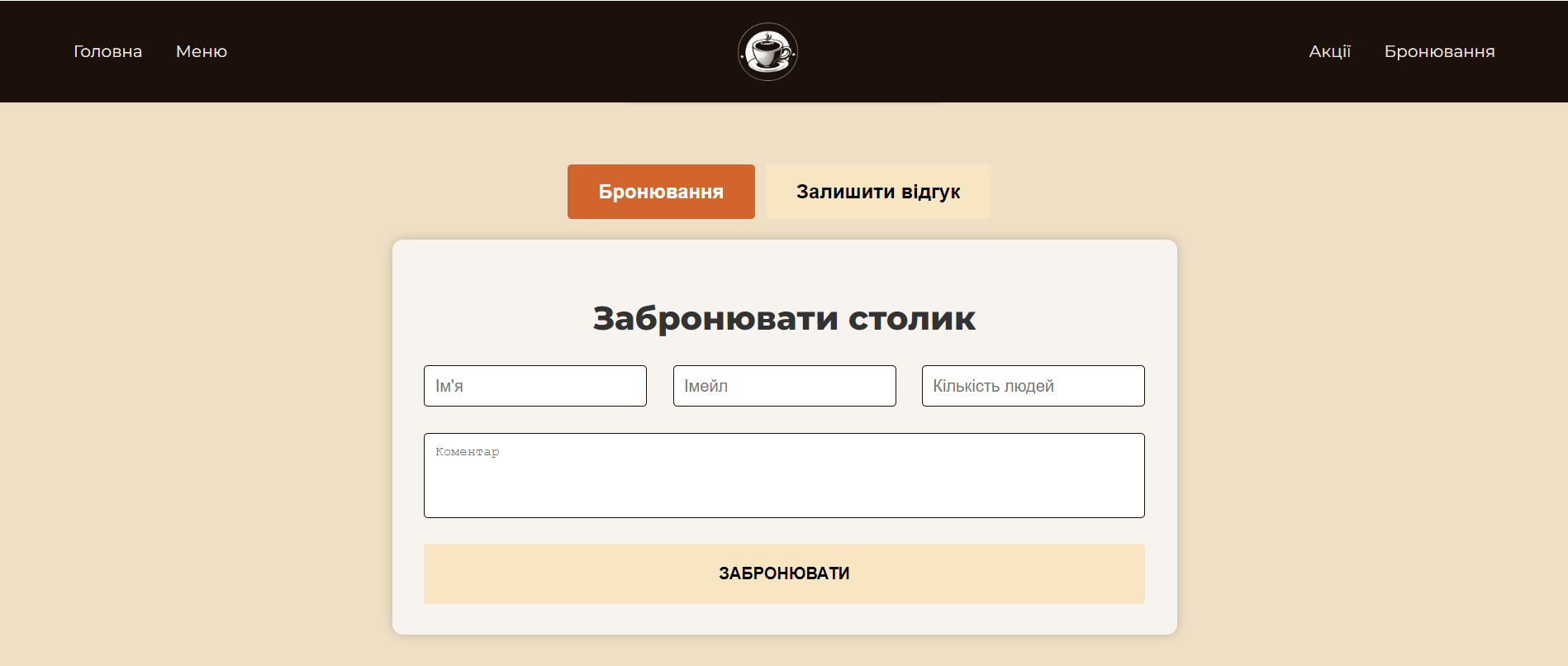


Рисунок 3.6 – Форма для резервацій

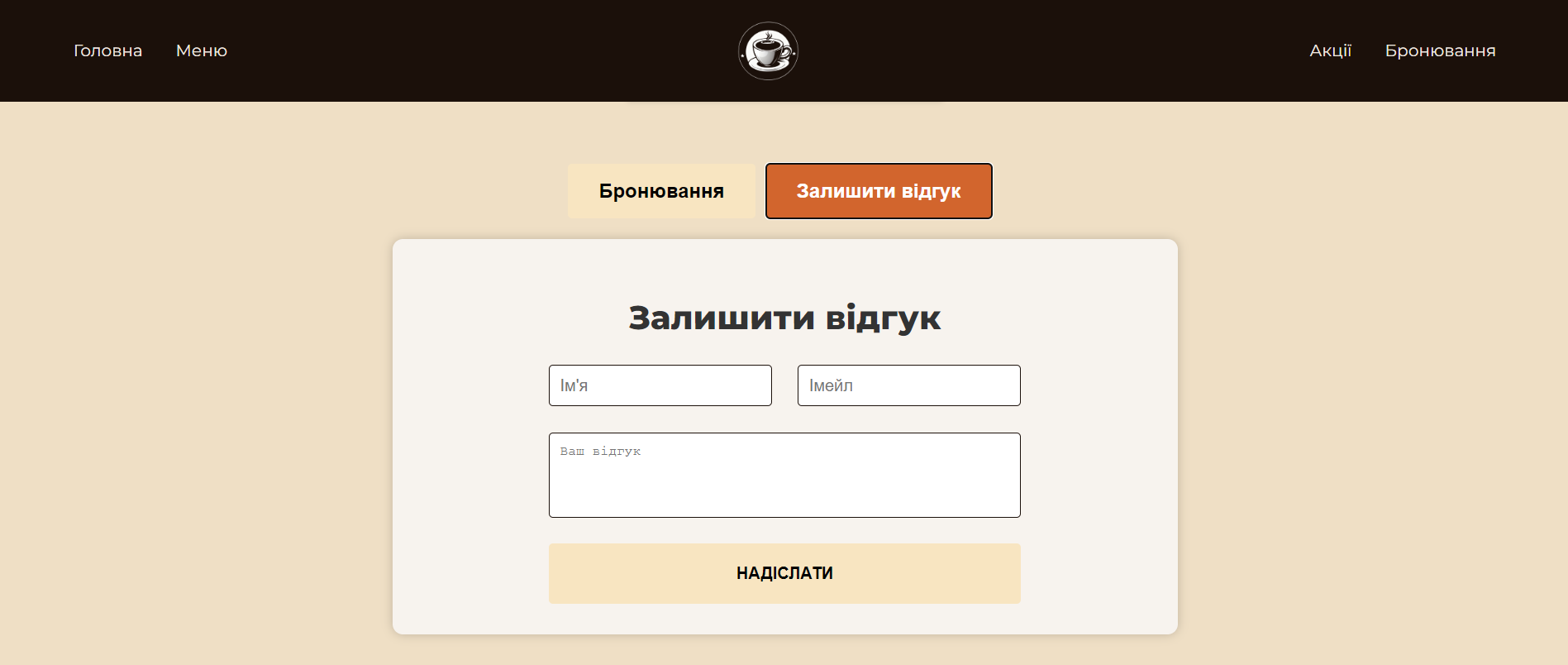


Рисунок 3.7 – Форма для відгуків

Внизу сторінки знаходиться футер з контактною інформацією кав’ярні, включаючи електронну пошту, номер телефону та адресу, визначеними тегами <p> та <a>. (див. рис. 3.8).

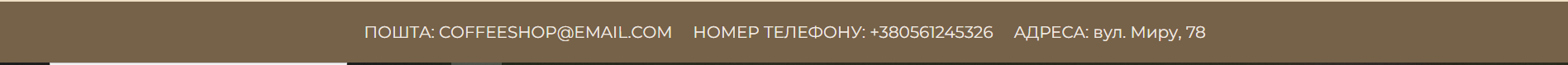


Рисунок 3.8. – Футер

## 3.2 Програмна реалізація бази даних

**3.2.1 Програмна реалізація таблиці reservations**

Для коректної та повноцінної роботи форми було створено базу даних. Для цього використовувались наступні технології та інструменти: MySQL як система управління базами даних, SQLyog як інтерфейс для роботи з базою даних, PHP для взаємодії між фронтендом і базою даних, Open Server як локальний сервер для розробки. Для коректної роботи було запущено Open Server і поміщено HTML, CSS, JavaScript та PHP-файли у відповідну директорію, щоб він був доступний через локальний сервер.

Процес створення таблиці та бази даних розпочинається з налаштування необхідного середовища. Спочатку встановлюється Open Server, який забезпечує роботу локального сервера та бази даних. В Open Server включено MySQL, що дозволяє створювати і керувати базами даних.

Далі, за допомогою SQLyog, інтерфейсу для роботи з MySQL, створюється нова база даних та таблиця «reservations» через SQL-запити:

|  |
| --- |
| CREATE TABLE reservations  ( id INT AUTO\_INCREMENT PRIMARY KEY,  NAME VARCHAR(255) NOT NULL,  email VARCHAR(255) NOT NULL,  people INT NOT NULL,  comment TEXT,  created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP); |

Він визначає структуру таблиці, включаючи поля для ідентифікатора резервації, імені клієнта, електронної пошти, кількості людей, коментаря та дати створення. Поле id є первинним ключем і автоматично збільшується з кожним новим записом. Рядок «id INT AUTO\_INCREMENT PRIMARY KEY» створює стовпець id, який є цілочисельним (INT), автоматично збільшується (AUTO\_INCREMENT) і є первинним ключем (PRIMARY KEY). Це означає, що кожен запис у таблиці буде мати унікальний ідентифікатор. Рядок «name VARCHAR(255) NOT NULL» створює стовпець name, який зберігає рядки символів довжиною до 255 символів (VARCHAR(255). Параметр NOT NULL означає, що це поле обов’язково повинно бути заповнене. Рядок «email VARCHAR(255) NOT NULL» аналогічно створює стовпець email, який також обов’язково заповнюється і може містити до 255 символів.

Рядок «people INT NOT NULL» створює стовпець people, що зберігає цілочисельне значення кількості людей і також є обов’язковим для заповнення.

Рядок «comment TEXT» створює стовпець comment, який може містити довільний текст.

Рядок «created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP» створює стовпець created\_at, який автоматично заповнюється поточним часом створення запису (DEFAULT CURRENT\_TIMESTAMP).

Після успішного виконання команди, таблиця reservations буде створена у базі даних (див. рис 3.9).

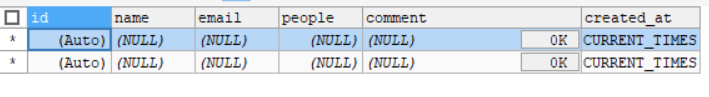


Рисунок 3.9 – Таблиця БД

Після створення таблиці бази даних було написано PHP код для обробки даних, що вноситимуть користувачі сайту (див. Додаток Г).

Код PHP обробляє бронювання столиків у кав’ярні. Він отримує дані з форми, підключається до бази даних MySQL, вставляє дані у відповідну таблицю і повертає повідомлення про успішність або помилку операції.

На початку коду задаються параметри підключення до бази даних: servername, username, password, dbname. Вони використовуються для встановлення зв’язку з базою даних.

Якщо дані були надіслані методом POST (тобто, якщо користувач натиснув кнопку на формі), код починає обробляти ці дані. Форма містить поля для введення імені, електронної пошти, кількості людей та коментарів. Дані з цих полів отримуються за допомогою $\_POST

Після цього здійснюється підключення до бази даних за допомогою new mysqli ($servername, $username, $password, $dbname). Якщо підключення не вдається, програма виведе повідомлення про помилку і завершить роботу.

Якщо підключення успішне, готується SQL-запит для вставки даних у таблицю reservations. Запит використовує підготовлені оператори для забезпечення безпеки від SQL-ін’єкцій.

Потім, за допомогою prepare та bind\_param, дані з форми зв’язуються із запитом, а execute виконує запит. Якщо вставка даних у таблицю успішна, виводиться повідомлення про успішне бронювання, інакше – повідомлення про помилку з детальною інформацією.

Наприкінці, підключення до бази даних закривається за допомогою методів close.

**3.2.2 Програмна реалізація таблиці reviews**

Далі так само як і для іншої форми, була створена таблиця reviews у базі даних diplom:

|  |
| --- |
| CREATE TABLE IF NOT EXISTS reviews (  id INT AUTO\_INCREMENT PRIMARY KEY,  name VARCHAR(255) NOT NULL,  email VARCHAR(255) NOT NULL,  comment TEXT NOT NULL,  created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP )  ENGINE=InnoDB; |

Після успішного виконання команди, таблиця reservations буде створена у базі даних (див. рис. 3.10).

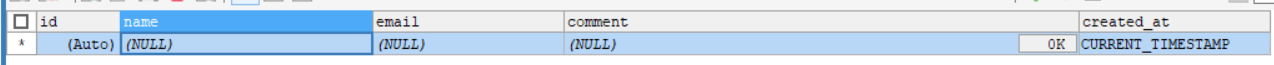
****

Рисунок 3.10 – Таблиця БД

PHP-код розміщений всередині HTML-розділу з класом review-section (див. Додаток Ґ). Він запуститься тільки в разі використання форми з відгуками. Спершу увімкнено відображення помилок для налагодження. Встановлено параметри для підключення до бази даних: сервер, ім'я користувача, пароль та назва бази даних. Після цього виконується підключення до бази даних за допомогою об'єкта mysqli. Якщо з'єднання не вдається, виводиться повідомлення про помилку. Далі SQL-запитом вибираються ім'я та коментарі з таблиці reviews. Якщо є результати, в циклі while формуються HTML-структури для кожного відгуку, де відображаються коментар і ім'я автора. Якщо відгуків немає, виводиться повідомлення про їх відсутність. Наприкінці з'єднання з базою даних закривається.

# ВИСНОВКИ

У кваліфікацйній роботі було розглянуто створення вебсайту для кав’ярні. Було визначено вимоги до сучасних вебсайтів та засоби для їх створення. Сформовано технічне завдання та технічні вимоги до розроблюваного сайту. Було створено програмну реалізацію вебсайту кав’ярні з різноманітними функціональними можливостями.

У ході роботи були виконані наступні завдання:

1. визначено поняття вебсайту, його класифікації та структуру
2. описано інструменти, за допомогою яких може бути реалізовано вебсайт;
3. розроблено прототип вебсайту для кав'ярні;
4. розроблено вебсайт кав’ярні;
5. реалізовано бекенд частину вебсайту;
6. проведено тестування вебсайту.

Розроблений вебсайт відповідає сформованому технічному завданню та технічним вимогам.

**ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Фелечко О.С. Вебсайт: від поняття до створення та функціонування. ІТ-право: проблеми і перспективи розвитку в Україні (друга міжнародна щорічна конференція). URL: http://aphd.ua/ publication-363 (дата звернення: 25.05.2024).
2. Поняття, структура та різновиди вебсайтів. (дата звернення: 25.05.2024).
3. Структура сайту: основні види та правила їх розробки URL: https://webtune.com.ua/statti/web-rozrobka/struktura-sajtu/ (дата звернення: 05.06.2024).
4. Автоматизоване розроблення вебсайтів. URL: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://www.ndu.edu.ua/liceum/web.pdf (дата звернення: 25.05.2024).
5. Юлія Каграманова. Як будувати UML-діаграми. Розбираємо три найпопулярніші варіанти. URL: https://dou.ua/forums/topic/40575/ (дата звернення: 25.05.2024).
6. Діаграма прецедентів. URL: https://www.wikiwand.com/uk/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0\_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D1%96%D0%B2 (дата звернення: 25.05.2024).
7. Прототипування сайту. URL: https://webtune.com.ua/statti/dyzajn/prototypuvannya-sajtu/ (дата звернення: 05.06.2024).
8. Як створити прототип сайту. URL: https://hostiq.ua/blog/ukr/site-prototype/(дата звернення: 05.06.2024).
9. Що таке Figma: функції, інструменти та переваги URL: https://wezom.academy/ua/chto-takoe-figma-funktsii-instrumenty-ipreimuschestva/(дата звернення: 05.06.2024).
10. 10 кращих редакторів коду для програмістів URL: https://mate.academy/blog/front-end-and-js/top-10-text-editors/(дата звернення: 05.06.2024).

# ДОДАТОК А

**HTML код сторінки**

<!DOCTYPE html>

<html lang="uk">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Кав'ярня</title>

<link rel="stylesheet" href="style.css">

<script src="myscripts.js" defer></script>

</head>

<body>

<header class="header-container">

<ul class="nav-list nav-left">

<li class="nav-item"><a href="#main">Головна</a></li>

<li class="nav-item"><a href="#menu">Меню</a></li>

</ul>

<img src="images/logo.png" alt="Логотип" class="logo">

<ul class="nav-list nav-right">

<li class="nav-item"><a href="#offers">Акції</a></li>

<li class="nav-item"><a href="#reservation">Бронювання</a></li>

</ul>

</header>

<section class="home-screen" id="home">

<div>

<a href="#menu" class="button">ПЕРЕЙТИ ДО МЕНЮ</a>

</div>

</section>

<section class="menu-section" id="menu">

<h2>Наше меню</h2>

<div class="menu-block" id="hot-drinks">

<h3>Гарячі напої</h3>

<button class="prev-btn">&lt;</button>

<div class="menu-carousel">

<div class="carousel-container">

<div class="menu-item">

<img src="images/cappuccino.jpg" alt="Cappuccino">

<div class="item-info">

<p>Капучіно</p>

<p>100₴</p>

</div>

</div>

<div class="menu-item">

<img src="images/americano.jpg" alt="Americano">

<div class="item-info">

<p>Американо</p>

<p>100₴</p>

</div>

</div>

<div class="menu-item">

<img src="images/espresso.jpg" alt="Espresso">

<div class="item-info">

<p>Еспресо</p>

<p>100₴</p>

</div>

</div>

<div class="menu-item">

<img src="images/latte.jpg" alt="Latte">

<div class="item-info">

<p>Лате</p>

<p>100₴</p>

</div>

</div>

</div>

</div>

<button class="next-btn">&gt;</button>

</div>

<div class="menu-block" id="cold-drinks"> <h3>Холодні напої</h3> <button

class="prev-btn">&lt;</button> <div class="menu-carousel"> <div

class="carousel-container"> <div class="menu-item"> <img

src="images/iced-coffee.jpg" alt="Iced Latte"> <div class="item-info">

<p>Айс лате</p> <p>100₴</p> </div> </div> <div class="menu-item"> <img

src="images/smoothie.jpg" alt="Smoothie"> <div class="item-info">

<p>Смузі</p> <p>100₴</p> </div> </div> <div class="menu-item"> <img

src="images/frappe.jpg" alt="Frappe"> <div class="item-info"> <p>Фрапе</p>

<p>100₴</p> </div> </div> </div> </div> <button

class="next-btn">&gt;</button> </div>

<div class="menu-block" id="desserts">

<h3>Десерти</h3>

<button class="prev-btn">&lt;</button>

<div class="menu-carousel">

<div class="carousel-container">

<div class="menu-item">

<img src="images/cake.jpg" alt="Cake">

<div class="item-info">

<p>Тортик</p>

<p>100₴</p>

</div>

</div>

<div class="menu-item">

<img src="images/croissant.jpg" alt="Croissant">

<div class="item-info">

<p>Круасан</p>

<p>100₴</p>

</div>

</div>

<div class="menu-item">

<img src="images/brownie.jpg" alt="Brownie">

<div class="item-info">

<p>Брауні</p>

<p>100₴</p>

</div>

</div>

<div class="menu-item">

<img src="images/muffin.jpg" alt="Muffin">

<div class="item-info">

<p>Мафін</p>

<p>100₴</p>

</div>

</div>

</div>

</div>

<button class="next-btn">&gt;</button>

</div>

</section>

<section>

<div class="discount-section" id="offers">

<h3 class="discount-text">-10% для студентів</h3>

<a href="#reservation" class="button">ЗАБРОНЮВАТИ СТОЛИК</a>

</div>

</section>

<section class="review-section">

<h2>Відгуки</h2>

<figure class="review">

<figcaption>

<blockquote>

<p>Ця кав'ярня стала нашим улюбленим місцем для зустрічей з друзями. Широкий вибір кавових напоїв – від класичної еспресо до креативних авторських коктейлів. Бариста завжди порадить щось нове і цікаве. Окрім кави, тут ще й смачні сніданки та обіди. Відвідування цієї кав'ярні завжди піднімає настрій!</p>

</blockquote>

<h3>Маша</h3>

</figcaption>

</figure>

<figure class="review">

<figcaption>

<blockquote>

<p>Вчора відвідали цю кав'ярню вперше і були приємно здивовані. Атмосфера дуже затишна, інтер'єр стильний і сучасний. Кава неймовірно смачна, а десерти просто тануть у роті. Особливо сподобався чізкейк з лісовими ягодами. Персонал дуже привітний і уважний. Обов'язково повернемось ще раз і рекомендуватимемо друзям!</p>

</blockquote>

<h3>Саша</h3>

</figcaption>

</figure>

<figure class="review">

<figcaption>

<blockquote>

<p>Відчуття, ніби потрапив у кавову казку. Кава дуже смачна, відчувається, що готують її з любов'ю. Солодощі – просто шедеври, кожен шматочок приносить справжнє задоволення. Особлива подяка персоналу за швидке і якісне обслуговування. Інтер'єр також заслуговує окремої похвали – дуже затишно і приємно знаходитися тут. Рекомендую всім відвідати цю кав'ярню!

</p>

</blockquote>

<h3>Даша</h3>

</figcaption>

</figure>

</section>

<section class="review-section">

<?php

// Enable error reporting

ini\_set('display\_errors', 1);

ini\_set('display\_startup\_errors', 1);

error\_reporting(E\_ALL);

// Database connection parameters

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "diplom";

// Connect to the database

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

// SQL query to select reviews

$sql = "SELECT name, comment FROM reviews";

$result = $conn->query($sql);

// Check if there are any reviews

if ($result->num\_rows > 0) {

// Output data of each row

while($row = $result->fetch\_assoc()) {

// Output HTML structure for each review

echo '<figure class="review">';

echo '<figcaption>';

echo '<blockquote>';

echo '<p>' . $row["comment"] . '</p>';

echo '</blockquote>';

echo '<h3>' . $row["name"] . '</h3>';

echo '</figcaption>';

echo '</figure>';

}

} else {

echo "No reviews yet.";

}

// Close connection

$conn->close();

?>

</section>

<section id="reservation-section">

<div class="slider">

<button id="reservation-btn" class="active">Бронювання</button>

<button id="review-btn">Залишити відгук</button>

</div>

<div id="reservation-form" class="form-container active">

<div class="reservation-form">

<h2>Забронювати столик</h2>

<form action="reserv.php" method="POST">

<div class="form-group">

<input type="text" id="name" name="name" placeholder="Ім'я" required>

<input type="email" id="email" name="email" placeholder="Імейл" required>

<input type="number" id="people" name="people" placeholder="Кількість людей" required>

</div>

<textarea id="comment" name="comment" placeholder="Коментар" rows="4"></textarea>

<button type="submit">ЗАБРОНЮВАТИ</button>

</form>

</div>

</div>

<div id="review-form" class="form-container">

<div class="review-form">

<h2>Залишити відгук</h2>

<form action="submit\_review.php" method="POST">

<div class="form-group">

<input type="text" id="review-name" name="name" placeholder="Ім'я" required>

<input type="email" id="review-email" name="email" placeholder="Імейл" required>

</div>

<textarea id="review-comment" name="comment" placeholder="Ваш відгук" rows="4" required></textarea>

<button type="submit">НАДІСЛАТИ</button>

</form>

</div>

</div>

</section>

<footer class="footer">

<p>ПОШТА: <a href="mailto:CONFISERIE@EMAIL.COM">COFFEESHOP@EMAIL.COM</a></p>

<div class="line"></div>

<p>НОМЕР ТЕЛЕФОНУ: <a href="tel:+380561245326">+380561245326</a></p>

<div class="line"></div>

<p>АДРЕСА: вул. Миру, 78</p>

</footer>

</body>

</html>

# ДОДАТОК Б

**CSS код сторінки**

@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@400;600;800&display=swap');

@font-face {

font-family: 'Roboto';

src: url('./Roboto/Roboto-Medium.ttf') format("truetype");

}

html,body {

font-family: "Montserrat";

font-size: 1em;

background-color: #efdfc5;

color: #f1eae2;

margin: 0;

padding: 0;

overflow-x: hidden;

}

a {

color: #f1eae2;

text-decoration: none;

}

h2 {

color: #1b100a;

}

/\* Навігаційне меню \*/

.header-container {

display: flex;

align-items: center;

padding: 1.28em 3.44em;

justify-content: space-between;

position: fixed;

z-index: 2;

top: 0;

left: 0;

right: 0;

background-color: #1b100a;

color: #f1eae2;

}

.logo {

width: 60px;

height: auto;

margin: 0 10px;

}

.nav-list {

display: flex;

align-items: center;

list-style: none;

padding: 0;

}

.nav-item {

margin: 0 1em;

}

.nav-item a:hover {

text-decoration: underline;

}

/\* Головний екран \*/

.home-screen {

background-image: url('images/coffee.jpg');

background-repeat: no-repeat;

background-size: cover;

background-position: bottom;

width: 100%;

height: 100vh;

display: flex;

flex-direction: row;

align-items: center;

margin-top: 2.8vh;

padding-left: 20px;

}

.button {

position: relative;

overflow: hidden;

background-color: #f8e5c1;

color: #000000;

display: inline-block;

padding: 15px 45px;

line-height: 1.85;

min-width: 230px;

font-size: 16px;

letter-spacing: 2px !important;

font-family: "Roboto", -apple-system, BlinkMacSystemFont, "Segoe UI", "Helvetica Neue", Arial, sans-serif;

font-weight: 900;

text-transform: uppercase;

white-space: nowrap;

text-align: center;

cursor: pointer;

vertical-align: middle;

user-select: none;

transition: all .25s ease-in-out;

box-shadow: 7px 7px 13px 0 rgba(0, 0, 0, 0.13);

}

.button:hover {

background-color: #bc9060;

}

/\* Меню \*/

.menu-section {

display: flex;

flex-direction: column;

align-items: center;

text-align: center;

color: #1b100a;

}

.menu-block {

width: 100%;

max-width: 80%;

text-align: center;

margin-bottom: 30px;

position: relative;

}

.menu-carousel {

position: relative;

overflow: hidden;

display: flex;

justify-content: center;

}

.carousel-container {

display: flex;

transition: transform 0.3s ease;

}

.menu-item {

flex: 0 0 calc(33.3333% - 10px); /\* 3 items per view \*/

margin-right: 10px;

text-align: center;

}

.menu-item img {

width: 65%;

height: auto;

}

.item-info {

margin-top: 5px;

}

.prev-btn,

.next-btn {

position: absolute;

top: 50%;

transform: translateY(-50%);

background-color: #fff;

border: none;

padding: 10px;

cursor: pointer;

font-size: 18px;

z-index: 1;

}

.prev-btn {

left: 0;

}

.next-btn {

right: 0;

}

/\* Акції \*/

.discount-section {

/\*розташування елементів\*/

display: flex;

justify-content: flex-start;

flex-direction: column;

align-items: flex-end;

/\*фон секції\*/

background-repeat: no-repeat;

background-size: cover;

background-position: bottom;

width: 100%;

height: 80vh;

padding-left: 20px;

background-position: 50%;

background-image: url('images/sale.jpg');

padding-right: 20px; /\* Додає відступ праворуч \*/

box-sizing: border-box;

}

.discount-section .discount-text {

text-align: right;

font-size: 3em;

color: white;

line-height: 1.22;

letter-spacing: .05em;

font-weight: 400;

text-transform: none;

margin-left: 0;

margin-top: 20vh;

margin-bottom: 20px;

max-width: 100%;

word-wrap: break-word;

padding-right: 20px;

}

/\*Відгуки\*/

.review-section {

padding: 30px 10px;

text-align: center;

}

.review {

box-shadow: 0 0 5px rgba(0, 0, 0, 0.15);

display: inline-block;

font-family: 'Roboto', Arial, sans-serif;

font-size: 16px;

margin: 35px 10px 10px;

max-width: 310px;

min-width: 250px;

position: relative;

text-align: center;

width: 100%;

background-color: #f7f3ee;

border-radius: 5px;

border-top: 5px solid #1b100a;

color: #1b100a;

}

.review \*,

.review \*:before {

-webkit-box-sizing: border-box;

box-sizing: border-box;

-webkit-transition: all 0.1s ease-out;

transition: all 0.1s ease-out;

}

.review figcaption {

padding: 13% 10% 12%;

}

.review figcaption:before {

-webkit-transform: translateX(-50%);

transform: translateX(-50%);

background-color: #f7f3ee;

border-radius: 50%;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.25);

color: #d2652d;

content: "\f10e";

font-family: 'FontAwesome';

font-size: 32px;

font-style: normal;

left: 50%;

line-height: 60px;

position: absolute;

top: -30px;

width: 60px;

}

.review h3 {

color: #3c3c3c;

font-size: 20px;

font-weight: 300;

line-height: 24px;

margin: 10px 0 5px;

}

.review blockquote {

font-style: italic;

font-weight: 300;

margin: 0 0 20px;

}

/\*Форма\*/

#reservation-section {

display: flex;

flex-direction: column;

justify-content: center;

align-items: center;

margin-bottom: 10px;

padding: 20px;

}

.slider {

display: flex;

margin-bottom: 20px;

}

.slider button {

padding: 15px 30px;

background-color: #f8e5c1;

border: none;

border-radius: 4px;

font-size: 1.2em; /\* Increased font size \*/

font-weight: bold;

cursor: pointer;

margin-right: 10px;

}

.slider button.active {

background-color: #d2652d;

color: white;

}

.form-container {

display: none;

width: 100%;

max-width: 700px; /\* Increased max-width \*/

background-color: #f7f3ee;

border-radius: 10px;

padding: 30px; /\* Increased padding \*/

box-shadow: 0 0 10px rgba(0, 0, 0, 0.2); /\* Increased box-shadow \*/

}

.form-container.active {

display: flex;

flex-direction: column;

align-items: center;

justify-content: center;

}

.reservation-form h2,

.review-form h2 {

font-size: 2em; /\* Increased font size \*/

color: #333;

display: flex;

justify-content: center;

}

.reservation-form form,

.review-form form {

max-width: 700px; /\* Increased max-width \*/

margin: 0 auto;

display: flex;

flex-direction: column;

gap: 25px; /\* Increased gap \*/

}

.reservation-form .form-group,

.review-form .form-group {

display: flex;

gap: 25px; /\* Increased gap \*/

}

.reservation-form input,

.reservation-form textarea,

.review-form input,

.review-form textarea {

padding: 10px; /\* Increased padding \*/

border: 1px solid #1b100a;

border-radius: 4px;

box-sizing: border-box;

font-size: 1em; /\* Increased font size \*/

}

.reservation-form input,

.review-form input {

flex: 1;

}

.reservation-form textarea,

.review-form textarea {

flex: 1;

resize: none;

}

.reservation-form button,

.review-form button {

padding: 20px; /\* Increased padding \*/

background-color: #f8e5c1;

border: none;

border-radius: 4px;

font-size: 1em; /\* Increased font size \*/

font-weight: bold;

cursor: pointer;

}

.footer {

display: flex;

justify-content: center;

align-items: center;

background-color: #766149;

text-align: center;

padding: 20px;

margin-top:5px;

}

.footer p {

margin: 0 10px;

.line {

height: 20px;

border-left: 1px solid white;

margin: 0 20px;

}

# ДОДАТОК В

**JavaScript код сторінки**

const menuBlocks = document.querySelectorAll('.menu-block');

menuBlocks.forEach(block => {

const carousel = block.querySelector('.carousel-container');

const prevBtn = block.querySelector('.prev-btn');

const nextBtn = block.querySelector('.next-btn');

const itemWidth = carousel.querySelector('img').offsetWidth + 10;

const totalItems = carousel.querySelectorAll('img').length;

let currentPosition = 0;

prevBtn.addEventListener('click', () => {

if (currentPosition > 0) {

currentPosition--;

carousel.style.transform = `translateX(-${currentPosition \* itemWidth}px)`;

}

});

nextBtn.addEventListener('click', () => {

if (currentPosition < totalItems - 3) {

currentPosition++;

carousel.style.transform = `translateX(-${currentPosition \* itemWidth}px)`;

}

});

# ДОДАТОК Г

**PHP код для обробки бронювань**

<?php

// Параметри підключення до бази даних

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "diplom";

// Перевірка, чи дані були надіслані через POST

if ($\_SERVER["REQUEST\_METHOD"]. == "POST") {

// Отримання даних з форми

$name = $\_POST['name'].;

$email = $\_POST['email'].;

$people = $\_POST['people'].;

$comment = $\_POST['comment'].;

// Підключення до бази даних

$conn = new mysqli($servername, $username, $password, $dbname);

// Перевірка підключення

if ($conn->connect\_error) {

die("Підключення не вдалося: " . $conn->connect\_error);

}

// SQL-запит для вставки даних

$sql = "INSERT INTO reservations (name, email, people, comment) VALUES (?, ?, ?, ?)";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("ssis", $name, $email, $people, $comment);

// Виконання запиту та перевірка успіху

if ($stmt->execute()) {

echo "Резервацію успішно створено!";

} else {

echo "Помилка: " . $stmt->error;

}

// Закриття підключення

$stmt->close();

$conn->close();

}

?>

# ДОДАТОК Ґ

**PHP код для обробки нових відгуків**

<?php

ini\_set('display\_errors', 1);

ini\_set('display\_startup\_errors', 1);

error\_reporting(E\_ALL);

$servername = "localhost";

$username = "root";

$password = "";

$dbname = "diplom";

$conn = new mysqli($servername, $username, $password, $dbname);

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$sql = "SELECT name, comment FROM reviews";

$result = $conn->query($sql);

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

echo '<figure class="review">';

echo '<figcaption>';

echo '<blockquote>';

echo '<p>' . $row["comment"] . '</p>';

echo '</blockquote>';

echo '<h3>' . $row["name"] . '</h3>';

echo '</figcaption>';

echo '</figure>';

}

} else {

echo "No reviews yet.";

}

$conn->close();

?>

**Декларація**

**академічної доброчесності**

**здобувача освіти ВСП «Економіко-правничого фахового коледжу ЗНУ»**

Я, Замирайло Анастасія Олексіївна здобувачка освіти 4 курсу, спеціальності/освітньо-професійної програми інженерія програмного забезпечення/розробка програмного забезпечення групи К 121-20, адреса електронної пошти epfk121.20zamiraylo@gmail.com.

- підтверджую, що написана мною кваліфікаційна робота на тему «Проєктування та розробка адаптивного вебсайту для кав’ярні» відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст. 42 Закону України «Про освіту», зі змістом яких ознайомлений/ознайомлена;

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

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

Дата 01.06.2024 Підпис \_\_\_\_\_\_ Анастасія ЗАМИРАЙЛО

Дата \_\_\_\_\_\_\_\_\_ Підпис \_\_\_\_\_\_\_\_\_ Альона ХОДАКОВСЬКА