

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

Бакалавр

(назва освітнього ступеня)

на тему: Розробка веб-сайту ресторану "Гулівер".

Виконав: студент IV курсу, групи СН-42

спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

(підпис)

Татаревський А.С.

(прізвище та ініціали)

Керівник

(підпис)

Шимчук Г.В.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Шимчук Г.В.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Боднарчук І.О.

(прізвище та ініціали)

Рецензент

(підпис)

Золотий Р.З.

(прізвище та ініціали)

Тернопіль
2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.
(підпис) (прізвище та ініціали)

«__» червня 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

Студенту Татаревському Андрію Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка веб-сайту ресторану "Гулівер".

Керівник роботи Шимчук Григорій Валерійович, старший викладач кафедри КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «29» квітня 2024 року № 4/7-470

2. Термін подання студентом завершеної роботи 24 червня 2024р.

3. Вихідні дані до роботи Літературні та інтернет джерела щодо веб-технологій розроблення Інтерактивних засобів взаємодії з користувачами

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1 Аналіз предметної області та формування вимог до веб-сайту "Гулівер". 1.1 Аналіз предметної області веб-сайту "Гулівер". 1.2 Формування вимог до веб-сайту "Гулівер".

1.3 Опис актантів та опис їх використання для веб-сайту "Гулівер". 1.4 Варіанти використання функціональності веб-сайту "Гулівер". 1.5 Вибір інструментів для створення веб-сайту "Гулівер". 1.6 Висновок до першого розділу. 2. Проєктування основних компонентів

веб-сайту "Гулівер". 2.1 Проєктування header для веб-сайту "Гулівер". 2.2 Проєктування блогу шеф-кухаря для веб-сайта "Гулівер". 2.3 Реалізація системи входу та реєстрації користувачів для веб-сайту "Гулівер". 2.4 Архітектура взаємодії між фронтенд та бекенд

частинами веб-сайту "Гулівер". 2.5 Реалізація API для управління замовленнями на веб-сайті "Гулівер". 2.6 Висновок до другого розділу. 3 Тестування та експлуатація веб-сайту.

3.1 Реалізована головна сторінка веб-сайту "Гулівер". 3.2 Реалізований функціонал для кошика веб-сайту "Гулівер". 3.3 Тестування функціонально готового веб-сайту "Гулівер".

3.4 Ієрархія веб-сайту "Гулівер". 3.5 Висновок до третього розділу. 4. Безпека життєдіяльності, основи охорони праці. Висновки. Перелік джерел. Додатки.

3.1 Реалізована головна сторінка веб-сайту "Гулівер". 3.2 Реалізований функціонал для кошика веб-сайту "Гулівер". 3.3 Тестування функціонально готового веб-сайту "Гулівер".

3.4 Ієрархія веб-сайту "Гулівер". 3.5 Висновок до третього розділу. 4. Безпека життєдіяльності, основи охорони праці. Висновки. Перелік джерел. Додатки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1 Титульна сторінка. 2 Тема, мета, Предмет дослідження. 3 Технічні засоби. 4 Актори веб-сайту "Гулівер". 5 Діаграма використання користувача. 6 Діаграма використання адміністратора. 7 Діаграма використання шеф-кухаря. 8 Спроєктована архітектура. 9 Створений хедер та футер. 10 Головна сторінка веб-сайту. 11 Інтерактивні частини веб-сайту. 12 Реєстрація користувача. 13 Кошик та оформлення замовлення. 14 Створене замовлення в БД. 15 Спеціальний функціонал адміністратора та шеф-кухаря. 16 Блог шеф-кухаря. 17 Ієрархія проєкту. 18.Висновки.

1 Титульна сторінка. 2 Тема, мета, Предмет дослідження. 3 Технічні засоби. 4 Актори веб-сайту "Гулівер". 5 Діаграма використання користувача. 6 Діаграма використання адміністратора. 7 Діаграма використання шеф-кухаря. 8 Спроєктована архітектура. 9 Створений хедер та футер. 10 Головна сторінка веб-сайту. 11 Інтерактивні частини веб-сайту. 12 Реєстрація користувача. 13 Кошик та оформлення замовлення. 14 Створене замовлення в БД. 15 Спеціальний функціонал адміністратора та шеф-кухаря. 16 Блог шеф-кухаря. 17 Ієрархія проєкту. 18.Висновки.

1 Титульна сторінка. 2 Тема, мета, Предмет дослідження. 3 Технічні засоби. 4 Актори веб-сайту "Гулівер". 5 Діаграма використання користувача. 6 Діаграма використання адміністратора. 7 Діаграма використання шеф-кухаря. 8 Спроєктована архітектура. 9 Створений хедер та футер. 10 Головна сторінка веб-сайту. 11 Інтерактивні частини веб-сайту. 12 Реєстрація користувача. 13 Кошик та оформлення замовлення. 14 Створене замовлення в БД. 15 Спеціальний функціонал адміністратора та шеф-кухаря. 16 Блог шеф-кухаря. 17 Ієрархія проєкту. 18.Висновки.

1 Титульна сторінка. 2 Тема, мета, Предмет дослідження. 3 Технічні засоби. 4 Актори веб-сайту "Гулівер". 5 Діаграма використання користувача. 6 Діаграма використання адміністратора. 7 Діаграма використання шеф-кухаря. 8 Спроєктована архітектура. 9 Створений хедер та футер. 10 Головна сторінка веб-сайту. 11 Інтерактивні частини веб-сайту. 12 Реєстрація користувача. 13 Кошик та оформлення замовлення. 14 Створене замовлення в БД. 15 Спеціальний функціонал адміністратора та шеф-кухаря. 16 Блог шеф-кухаря. 17 Ієрархія проєкту. 18.Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Сенчишин В.С., кандидат технічних наук, доцент кафедри МТ		

7. Дата видачі завдання 29 січня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	30.01.2024	Виконано
2.	Підбір джерел про веб-технології для розробки інтерактивних веб-сайтів	31.01.2024-03.02.2024	Виконано
3.	Опрацювання джерел по темі кваліфікаційної роботи	04.02.2024-06.02.2024	Виконано
4.	Виконання дослідження щодо існуючих інструментів та технологій для розробки веб-сайтів	07.02.2024-11.02.2024	Виконано
5.	Розроблення структури та функціоналу веб-сайту "Гулівер"	30.05.2024-02.06.2024	Виконано
6.	Оформлення розділу "Аналіз предметної області та формування вимог до веб-сайту 'Гулівер'"	03.06.2024-05.06.2024	Виконано
7.	Оформлення розділу "Проектування основних компонентів веб-сайту 'Гулівер'"	06.06.2024-08.06.2024	Виконано
8.	Оформлення розділу "Тестування та експлуатація веб-сайту 'Гулівер'"	09.06.2024-11.06.2024	Виконано
9.	Виконання завдання до підрозділу "Безпека життєдіяльності"	13.06.2024-15.06.2024	Виконано
10.	Виконання завдання до підрозділу "Основи охорони Праці"	18.06.2024-20.06.2024	Виконано Виконано
11.	Оформлення кваліфікаційної роботи	21.06.2024-22.06.2024	Виконано
12.	Нормоконтроль	23.06.2024-24.06.2024	Виконано
13.	Перевірка на плагіат	25.06.2024	Виконано
14.	Попередній захист кваліфікаційної роботи	26.06.2024	Виконано
15.	Захист кваліфікаційної роботи	28.06.2024	Виконано

Студент

(підпис)

Татаревський А.С.

(прізвище та ініціали)

Керівник роботи

(підпис)

Шимчук Г.В.

(прізвище та ініціали)

АНОТАЦІЯ

Розробка веб-сайту ресторану “Гулівер” // Кваліфікаційна робота освітнього рівня “Бакалавр” // Татаревський Андрій Сергійович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп’ютерно-інформаційних систем і програмної інженерії, кафедра комп’ютерних наук, група СН-42 // Тернопіль, 2024 // С.66 , рис. – 29, табл. – 2, слайдів – 18, додат. – 10 , бібліогр. – 41.

Ключові слова: веб-сайт, ресторан, FastAPI, React, MongoDB, аутентифікація, замовлення, інтерфейс.

Кваліфікаційна робота присвячена розробці веб-сайту ресторану “Гулівер” для покращення взаємодії з клієнтами та підвищення якості обслуговування.

У першому розділі кваліфікаційної роботи розглянуто аналіз предметної області, сформовано перелік вимог та сутностей, а також розроблено діаграми варіантів використання веб-сайту “Гулівер”.

У другому розділі кваліфікаційної роботи проведено моделювання архітектури веб-сайту, проєктування та практичну реалізацію структури як окремих функціональних компонентів, так і загальної структури веб-сайту.

У третьому розділі кваліфікаційної роботи описано процес тестування та експлуатації веб-сайту “Гулівер”, проведено аналіз результатів тестування, а також розглянуто аспекти забезпечення безпеки та продуктивності веб-сайту.

ANNOTATION

Development of the Restaurant Website "Gulliver" // Qualification work of the educational level "Bachelor" // Tatarevskiy Andrii Serhiiiovych // Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SN-42 // Ternopil, 2024 // P. 66, fig. - 29, tabl. - 2, slides. - 18, annexes. – 10, references - 41.

Keywords: website, restaurant, FastAPI, React, MongoDB, authentication, orders, interface.

The qualification work is dedicated to the development of the "Gulliver" restaurant website to improve customer interaction and service quality.

The first section of the qualification work discusses the analysis of the subject area, formulates a list of requirements and entities, and develops use case diagrams for the "Gulliver" website.

The second section of the qualification work involves modeling the website's architecture, designing, and practical implementation of both individual functional components and the overall structure of the website.

The third section of the qualification work describes the process of testing and operating the "Gulliver" website, analyzes the test results, and examines aspects of ensuring the security and performance of the website.

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ТЕРМІНІВ

API (Application Programming Interface) – програмний інтерфейс додатків.

JWT (JSON Web Token) – JSON веб-токен, який використовується для безпечного обміну інформацією між сторонами.

UI (User Interface) – користувацький інтерфейс.

UX (User Experience) – користувацький досвід.

DB (Database) – база даних.

HTTP (Hypertext Transfer Protocol) – протокол передачі гіпертексту.

HTTPS (Hypertext Transfer Protocol Secure) – безпечний протокол передачі гіпертексту.

SSL (Secure Sockets Layer) – протокол захисту інформації.

REST (Representational State Transfer) – архітектурний стиль для створення веб-сервісів.

SPA (Single Page Application) – односторінковий додаток.

JSON (JavaScript Object Notation) – формат обміну даними між сервером і клієнтом.

SQL (Structured Query Language) – мова структурованих запитів для роботи з базами даних.

NoSQL – тип баз даних, що не використовують реляційну модель.

IDE (Integrated Development Environment) – інтегроване середовище розробки.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ФОРМУВАННЯ ВИМОГ ДО ВЕБ-САЙТУ “ГУЛІВЕР”	10
1.1 Аналіз предметної області веб-сайту “Гулівер”	10
1.2 Формування вимог до веб-сайту “Гулівер”	11
1.3 Опис актантів та опис їх використання для веб-сайту “Гулівер”	13
1.4 Варіанти використання функціональності веб-сайту “Гулівер”	17
1.5 Вибір інструментів для створення веб-сайту “Гулівер”	18
1.6 Висновок до першого розділу	20
РОЗДІЛ 2. ПРОЄКТУВАННЯ ОСНОВНИХ КОМПОНЕНТІВ ВЕБ-САЙТУ “ГУЛІВЕР”	22
2.1 Проєктування header для веб-сайту “Гулівер”	22
2.2 Проєктування блогу шеф-кухаря для веб-сайта “Гулівер”	24
2.3 Реалізація системи входу та реєстрації користувачів для веб-сайту “Гулівер”	26
2.4 Архітектура взаємодії між фронтенд та бекенд частинами веб- сайту “Гулівер”	28
2.5 Реалізація API для управління замовленнями на веб-сайті “Гулівер”	30
2.6 Висновок до другого розділу	31
РОЗДІЛ 3. ТЕСТУВАННЯ ТА ЕКСПЛУАТАЦІЯ ВЕБ-САЙТУ “ГУЛІВЕР”	33
3.1 Реалізована головна сторінка веб-сайту “Гулівер”	33
3.2 Реалізований функціонал для кошика веб-сайту “Гулівер”	38
3.3 Тестування функціонально готового веб-сайту “Гулівер”	42
3.4 Ієрархія веб-сайту “Гулівер”	47
3.5 Висновок до третього розділу	51
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	53

4.1	Забезпечення психоемоційного здоров'я та запобігання професійному вигоранню у користувачів ПК	53
4.2	Ергономічні вимоги до робочого місця оператора ПК	56
4.3	Висновки до четвертого розділу	59
	ВИСНОВКИ.....	60
	ПЕРЕЛІК ДЖЕРЕЛ	61
	ДОДАТКИ	

ВСТУП

Предмет дослідження. Методи та засоби розроблення інтерактивних веб-систем для обслуговування клієнтів ресторану.

Мета роботи. Підвищення якості обслуговування клієнтів ресторану шляхом створення інтерактивного та зручного веб-сайту.

Актуальність теми. Внаслідок глобалізації та зростання конкуренції у сфері ресторанного бізнесу, особливого значення набуває наявність ефективних інструментів для залучення та обслуговування клієнтів. Розробка сучасних веб-сайтів для ресторанів, що поєднують функціональність онлайн-замовлення, бронювання столиків, ведення блогу та інтерактивну взаємодію з користувачами, стає невід'ємною частиною успішного ведення бізнесу. Тому розробка веб-сайту ресторану “Гулівер” є актуальним напрямком сучасних досліджень в галузі інформаційних технологій та веб-розробки.

Мета і задачі дослідження. Метою даної кваліфікаційної роботи освітнього рівня «Бакалавр» є підвищення якості послуг ресторану “Гулівер” шляхом розробки та впровадження сучасного веб-сайту. Для досягнення поставленої мети потрібно виконати ряд завдань, зокрема:

- Проаналізувати стан досліджень в галузі веб-розробки та сучасні технології створення веб-сайтів для ресторанного бізнесу.
- Розробити архітектуру веб-сайту, що включає функціональність онлайн-замовлення, бронювання столиків, ведення блогу та управління користувачами.
- Реалізувати систему входу та реєстрації користувачів з розділенням прав доступу для адміністраторів, шеф-кухарів та звичайних користувачів.
- Розробити дизайн інтерфейсу користувача з урахуванням принципів зручності використання та адаптивного дизайну.
- Провести тестування функціональних можливостей веб-сайту та оцінити його продуктивність і безпеку.

Практичне значення одержаних результатів. Практичне значення одержаних результатів полягає у створенні сучасного, функціонального та зручного веб-сайту для ресторану “Тулівер”, який дозволить підвищити якість обслуговування клієнтів, збільшити кількість онлайн-замовлень та бронювань столиків, а також покращити комунікацію з відвідувачами через інтерактивний блог. Розроблений веб-сайт стане важливим інструментом для просування ресторану в інтернеті та підвищення його конкурентоспроможності.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ФОРМУВАННЯ ВИМОГ ДО ВЕБ-САЙТУ “ГУЛІВЕР”

1.1 Аналіз предметної області веб-сайту “Гулівер”

У сучасному світі ресторанний бізнес активно використовує веб-технології для залучення клієнтів та надання послуг онлайн. Створення ефективного веб-сайту є ключовим інструментом для досягнення цих цілей. Веб-сайти ресторанів надають можливість ознайомитися з меню, здійснити резервування столиків, замовити їжу онлайн, дізнатися про події та спеціальні пропозиції. Крім того, вони дозволяють вести блоги шеф-кухарів, спілкуватися з клієнтами через форми зворотного зв'язку, а також забезпечують доступ до соціальних мереж ресторану [1].

З точки зору користувача, добре продуманий і функціональний веб-сайт ресторану може значно покращити користувацький досвід, підвищити рівень задоволеності клієнтів та сприяти збільшенню кількості замовлень. З іншого боку, недоліки в дизайні чи функціональності сайту можуть призвести до втрати клієнтів та негативних відгуків.

Для кращого розуміння, які саме функції та особливості мають бути впроваджені на веб-сайті ресторану “Гулівер”, розглянемо аналіз трьох існуючих веб-сайтів ресторанів.

- На сайті Пузата Хата представлено меню з фотографіями страв, інформацію про місця розташування ресторанів, а також новини та акції. Однак, сайт не надає можливості резервування столиків онлайн, а також має обмежені інтерактивні елементи. Інтерфейс також потребує покращення для мобільних пристроїв, що може створювати незручності для користувачів.

- Сайт Sushi Zoom пропонує онлайн замовлення їжі з доставкою, презентацію меню з фотографіями страв та інформацію про спеціальні пропозиції та акції. Проте, відсутність блогу шеф-кухаря та можливості резервування столиків онлайн робить сайт менш зручним для користувачів, які

хочуть отримати більш інтерактивний досвід. Навігація сайтом також може бути складною.

- Сайт Файне місто дозволяє здійснити резервування столиків, переглянути меню та дізнатися про події та новини. Однак, він не підтримує онлайн замовлення їжі, має обмежену кількість інтерактивних елементів та неадаптивний дизайн для мобільних пристроїв, що знижує його зручність.

Аналіз існуючих сайтів ресторанів показав, що для створення веб-сайту ресторану “Гулівер” необхідно врахувати кілька ключових аспектів. Перш за все, важливо забезпечити можливість онлайн замовлення їжі з доставкою, що дозволить клієнтам зручно замовляти їжу прямо з сайту. Це підвищить рівень обслуговування та задоволеність клієнтів.

Також критично важливо забезпечити адаптивний дизайн, щоб сайт був зручним для користування з мобільних пристроїв, адже все більше користувачів використовують смартфони для доступу до інтернету [2]. Впровадження блогу шеф-кухаря з рецептами та кулінарними порадами допоможе залучити більше відвідувачів і підвищити їхню зацікавленість.

Крім того, додавання функції резервування столиків онлайн значно покращить користувацький досвід, дозволяючи клієнтам зручно бронювати столики через сайт. Використання інтерактивних елементів, таких як галереї фотографій і відео, зробить сайт більш привабливим та функціональним.

Врахування цих аспектів допоможе створити веб-сайт, який відповідатиме сучасним вимогам користувачів та сприятиме залученню більшої кількості клієнтів до ресторану “Гулівер”.

1.2 Формування вимог до веб-сайту “Гулівер”

Вимоги до веб-сайту ресторану "Гулівер" визначаються з метою забезпечення високої якості обслуговування клієнтів та зручності користування. Вони поділяються на функціональні та нефункціональні аспекти, а також вимоги до інтеграції з іншими системами.

З визначеними вимогами для веб-сайту ресторану “Гулівер”, наступним кроком є створення детального плану розробки, який охоплюватиме проєктування архітектури сайту [3], визначення основних компонентів та їх взаємодії, вибір технологій для фронтенду та бекенду [4].

Проєктування архітектури сайту включає розподіл системи на модулі, визначення їх ролей і відповідальностей, а також способів їх взаємодії. Це допоможе створити узгоджений план розробки, який забезпечить ефективну реалізацію всіх функціональних та нефункціональних вимог. Основні компоненти включатимуть: систему управління контентом, модулі для обробки замовлень, систему бронювання столиків.

Нижче наведено таблицю 1.1 з основними вимогами до веб-сайту “Гулівер”.

Таблиця 1.1 – Основні вимоги до веб-сайту “Гулівер”

Категорія	Вимоги
1	2
Функціональні вимоги	Презентація меню: відображення меню з детальним описом і фотографіями страв, сортування страв за категоріями.
	Онлайн замовлення їжі: додавання страв до кошика, оформлення замовлення, підтвердження замовлення через електронну пошту або SMS.
	Резервування столиків: інтерактивна форма для бронювання столиків, підтвердження та скасування бронювання.
	Блог шеф-кухаря: створення та публікація постів, коментування та обговорення.
Нефункціональні вимоги	Продуктивність: швидке завантаження сторінок, оптимізація зображень та медіа-файлів.
	Безпека: захист персональних даних, використання SSL сертифікатів, захист від атак CSRF та XSS.

Продовження таблиці 1.1

1	2
	Зручність користування: інтуїтивно зрозумілий інтерфейс, адаптивний дизайн, легка навігація.
	Надійність: високий рівень доступності, регулярне резервне копіювання даних, стабільна робота при великих навантаженнях.
Вимоги до інтеграції	Інтеграція з платіжними системами: підтримка різних методів оплати, безпечне зберігання та обробка платіжних даних.

Потрібно створити прототипи та макети, які допоможуть візуалізувати основні сторінки сайту та забезпечити узгодженість дизайну. Після цього починається етап імплементації функціональності, який включатиме розробку та тестування всіх визначених функцій сайту. Особлива увага приділяється забезпеченню високої продуктивності, безпеки та надійності.

Це допоможе створити веб-сайт ресторану “Гулівер”, який буде відповідати очікуванням користувачів і бізнес-вимогам [5].

1.3 Опис актантів та опис їх використання для веб-сайту “Гулівер”

У цьому підрозділі розглядаються основні актори веб-сайту ресторану “Гулівер” та способи їх взаємодії з системою. Основні актори відіграють важливу роль у забезпеченні ефективного функціонування веб-сайту та взаємодії з користувачами. Нижче наведено рисунок 1.1, що ілюструє основних акторів та їхні дії [6].

Незареєстрований користувач – це відвідувач сайту, який має доступ до загальнодоступних функцій, таких як перегляд меню, ознайомлення з інформацією про ресторан, перегляд блогу шеф-кухарів. Незареєстровані користувачі можуть ознайомлюватися з останніми новинами та спеціальними

пропозиціями ресторану, переглядати фотогалерею та контакти, що дозволяє їм отримати загальне уявлення про заклад перед тим, як вирішити зареєструватися або зробити замовлення.

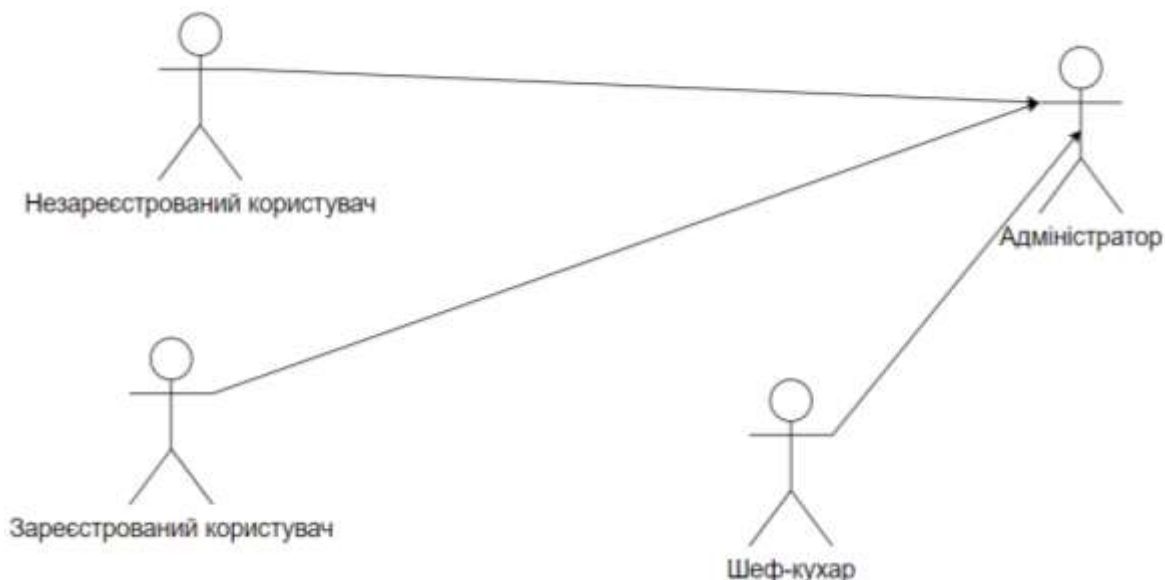


Рисунок 1.1 – Актори веб-сайту “Гулівер”

Шеф-кухар – це користувач, який має доступ до блогу та може створювати, редагувати та публікувати пости, ділитися рецептами та кулінарними порадами.

Адміністратор – це користувач, який має повний доступ до всіх функцій сайту, включаючи управління користувачами, додавання нових страв до меню, обробку замовлень та резервувань [7].

Незареєстрований користувач має доступ до обмежених функцій веб-сайту. Відвідувач сайту може переглядати меню ресторану, ознайомлюватися з інформацією про ресторан, а також читати публікації в блозі шеф-кухаря. Цей користувач має можливість робити замовлення або резервувати столики без попередньої реєстрації.

Переглядаючи меню, незареєстрований користувач може ознайомитися з різноманітними стравами, що пропонуються рестораном, і їхніми описами. Крім того, він може дізнатися про спеціальні акції та новини ресторану через публікації у блозі.

Зареєстрований користувач – це відвідувач сайту, який створив обліковий запис і увійшов у систему, використовуючи свої облікові дані. Зареєстрований користувач має доступ до всіх функцій, доступних незареєстрованим користувачам, таких як перегляд меню, ознайомлення з інформацією про ресторан та перегляд блогу шеф-кухаря.

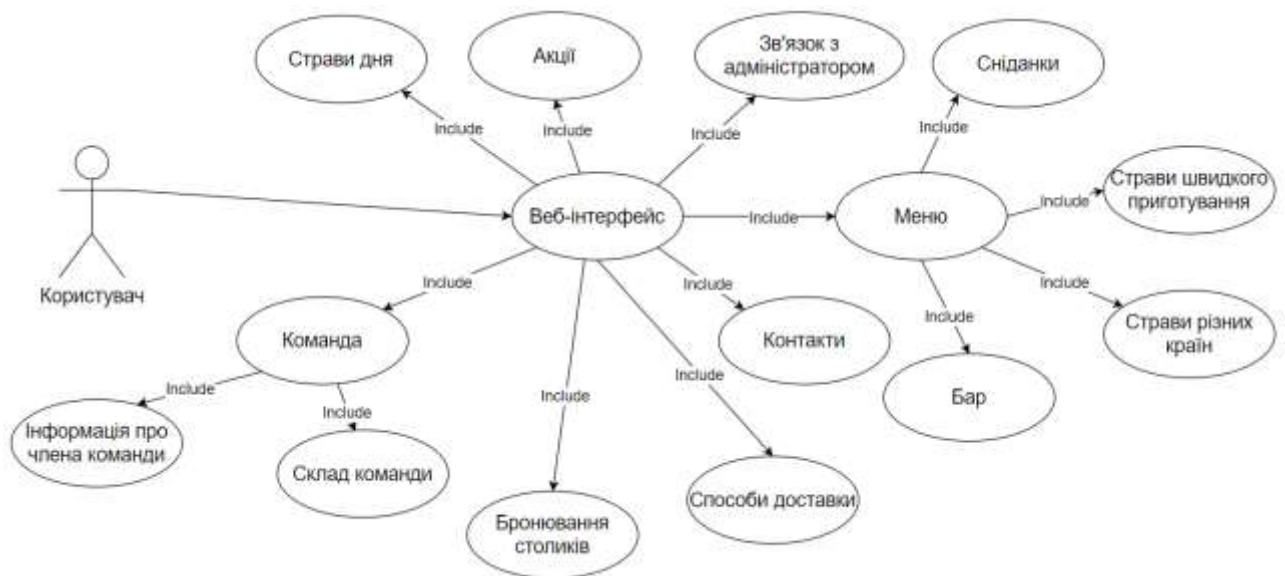


Рисунок 1.2 – Діаграма використання користувача веб-сайту “Гулівер”

На рисунку 1.2 подано, обидва, як зареєстрований, так і незареєстрований користувач можуть замовляти страви на сайті та бронювати столики. Основна відмінність полягає в тому, що для зареєстрованого користувача під час замовлення автоматично підтягується інформація про доставку, контактні дані та інші необхідні поля, що значно спрощує процес замовлення. Незареєстрований користувач буде змушений вручну вводити всі ці дані під час кожного замовлення або бронювання столика.

Перейдемо до діаграми використання адміністратора, що зображена на рисунку 1.3. Адміністратор – це користувач, який має спеціальні привілеї та доступ до адміністративної частини веб-сайту. Основні завдання адміністратора включають управління контентом сайту, модерацію користувачів, обробку замовлень та бронювань, а також забезпечення безпеки та працездатності

системи. Адміністратор може додавати, редагувати або видаляти страви з меню, створювати та публікувати нові пости в блозі шеф-кухаря, переглядати та обробляти замовлення користувачів, а також управляти бронюваннями столиків.



Рисунок 1.3 – Діаграма використання адміністратора веб-сайту “Гулівер”

Адміністратор також відповідає за підтримку актуальності інформації на сайті, проведення маркетингових акцій та спеціальних пропозицій, а також за взаємодію з постачальниками та партнерами. Крім того, адміністратор слідкує за відгуками користувачів та відповідає на їхні запитання, що допомагає підтримувати високий рівень обслуговування та задоволеності клієнтів.

Шеф-кухар має доступ до функцій блогу, де він може створювати, редагувати та публікувати пости з рецептами, кулінарними порадами та іншими матеріалами. Крім того, шеф-кухар може відповідати на коментарі відвідувачів та підтримувати активну взаємодію з клієнтами через блог [8]. Це дозволяє шеф-кухарю ділитися своїм професійним досвідом, залучати клієнтів до дискусій та підвищувати їхню зацікавленість у ресторані. Це сприяє

підвищенню довіри клієнтів та їхньої зацікавленості у відвідуванні ресторану, діаграма подана на рисунку 1.4.



Рисунок 1.4 – Діаграма використання шеф-кухаря веб-сайту “Гулівер”

Ця функціональність дозволяє шеф-кухарю ділитися своїм професійним досвідом, залучати клієнтів до дискусій та підвищувати їхню зацікавленість у ресторані.

1.4 Варіанти використання функціональності веб-сайту “Гулівер”

Для розробки ефективного веб-сайту важливо зрозуміти, як користувачі будуть взаємодіяти з різними функціями сайту. Розглянемо основні сценарії використання функціональності веб-сайту ресторану “Гулівер” [10].

Користувач заходить на веб-сайт і вибирає розділ “Резервування”. Він заповнює форму, вказуючи дату, час та кількість гостей. Після цього натискає кнопку “Забронювати”, і система обробляє запит, надсилаючи підтвердження на електронну пошту або через SMS. Користувач також має можливість скасувати або змінити бронювання через відповідну функцію на сайті.

Користувач заходить у розділ “Меню”, переглядає список страв та додає вибрані страви до кошика. Після завершення вибору користувач переходить до кошика, де перевіряє замовлення, вказує адресу доставки та обирає спосіб оплати.

Користувач заходить у розділ “Блог шеф-кухаря”, де може переглядати публікації з рецептами, кулінарними порадами та іншими матеріалами. Користувач може коментувати публікації та обговорювати їх з іншими відвідувачами сайту.

Розділ містить детальну інформацію про шеф-кухаря та інших ключових співробітників.

Користувач може взаємодіяти з різними інтерактивними елементами на сайті, такими як галереї фотографій та відео, що демонструють атмосферу ресторану, страви та події. Це дозволяє створити більш привабливий та інформативний користувацький досвід.

Для забезпечення зручності користування веб-сайтом “Гулівер” необхідно розробити детальні сценарії використання для кожної функціональності. Це допоможе визначити всі можливі варіанти взаємодії користувачів із сайтом та забезпечити їх максимально ефективно впровадження.

1.5 Вибір інструментів для створення веб-сайту “Гулівер”

Для успішної розробки веб-сайту ресторану “Гулівер” важливо вибрати правильний технологічний стек, який забезпечить високу продуктивність,

надійність та зручність в обслуговуванні. Нижче, в таблиці 1.3 наведено огляд обраних інструментів для фронтенду, бекенду та бази даних.

Таблиця 1.2 – Таблиця вибору інструментів для веб-сайту “Гулівер”

Категорія	Інструмент	Причини вибору
Бекенд	Node.js	Забезпечує високу продуктивність і ефективність завдяки неблокуючій архітектурі, підтримує величезну кількість бібліотек і модулів.
	Express.js	Забезпечує гнучкість та простоту у використанні, дозволяючи швидко створювати API та обробляти запити.
База даних	MongoDB	Забезпечує гнучкість у зберіганні даних, високу продуктивність, що особливо важливо для веб-сайтів з динамічними та великими обсягами даних.
Інтеграція	JWT	Дозволяє легко реалізувати аутентифікацію та авторизацію користувачів, забезпечуючи безпеку і зручність у використанні.

Можна також розглянути додаткові інструменти для покращення роботи веб-сайту “Гулівер”. Наприклад, використання Redis для кешування даних може значно прискорити обробку запитів і знизити навантаження на базу даних. Для управління станом на фронтенді можна розглянути використання Redux або Context API в поєднанні з React, що допоможе краще організувати дані та забезпечити їх ефективне використання [10]. Також варто звернути увагу на тестування: інструменти на зразок Jest і Cypress можуть бути дуже

корисними для забезпечення високої якості коду і стабільності роботи застосунку.

Для покращення SEO варто використовувати Next.js, який дозволяє реалізувати серверний рендеринг з React, що позитивно впливає на індексацію сайту пошуковими системами. Щодо підвищення безпеки, крім використання JWT, важливо налаштувати правильну обробку заголовків HTTP за допомогою Helmet.js і забезпечити регулярне оновлення всіх залежностей проєкту для запобігання вразливостям.

Ці кроки допоможуть створити сучасний, продуктивний і безпечний веб-сайту, що задовольнить потреби клієнтів ресторану “Гулівер” та забезпечить високу якість обслуговування.

1.6 Висновок до першого розділу

У першому розділі було проведено глибокий аналіз предметної області та визначено основні вимоги до веб-сайту ресторану “Гулівер”. Аналіз конкурентів на ринку виявив декілька ключових аспектів, які потребують уваги під час розробки сайту, такі як можливість онлайн замовлення їжі, адаптивний дизайн для мобільних пристроїв, блог шеф-кухаря, функція резервування столиків онлайн та інтерактивні елементи.

Формування вимог до веб-сайту включало визначення функціональних та нефункціональних аспектів, а також вимог до інтеграції. Це допоможе створити зручний, продуктивний та безпечний веб-сайт, що відповідатиме потребам користувачів.

Огляд технологічного стеку та вибір інструментів для розробки проєкту продемонстрував, що використання сучасних технологій, таких як React.js, Node.js, Express.js та MongoDB, дозволить створити продуктивний і надійний веб-сайт. Додатково, використання JWT для аутентифікації та додаткових інструментів для оптимізації роботи веб-сайту допоможуть досягти високих стандартів якості та задовольнити потреби клієнтів.

Подальші кроки включатимуть детальне планування розробки, створення прототипів, імплементацію функціональності та проведення тестування для забезпечення стабільності та безпеки веб-сайту. Це допоможе забезпечити успішну реалізацію проєкту та створити сучасний і функціональний веб-сайт для ресторану “Гулівер”, який сприятиме залученню нових клієнтів та покращенню якості обслуговування.

Додатково, буде розроблено систему аналітики для моніторингу активності користувачів та ефективності роботи сайту, що дозволить постійно вдосконалювати і оптимізувати ресурс відповідно до потреб користувачів. Важливою частиною розробки буде інтеграція системи лояльності для постійних клієнтів, яка включатиме спеціальні пропозиції, знижки та бонуси, що стимулюватиме повторні візити. Також можна додати розробка мобільного додатку як доповнення до веб-сайту для забезпечення доступу до сервісів ресторану з будь-яких пристроїв. Це забезпечить зручність для користувачів та збільшить їх задоволення від взаємодії з рестораном. Впровадження системи зворотного зв'язку дозволить отримувати цінні відгуки від клієнтів, що допоможе оперативно реагувати на їхні потреби та побажання, покращуючи таким чином якість обслуговування.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ОСНОВНИХ КОМПОНЕНТІВ ВЕБ-САЙТУ “ГУЛІВЕР”

2.1 Проєктування header для веб-сайту “Гулівер”

Проєктування хедера для веб-сайту ресторану “Гулівер” є важливим етапом у розробці загальної структури сайту, він представлений на рисунку 2.1. Хедер є ключовим елементом, який забезпечує користувачам зручну навігацію та доступ до основних розділів сайту. Важливо забезпечити простоту навігації та швидкий доступ до основних розділів сайту (див. додаток А). Адаптивний дизайн дозволяє автоматично змінювати розмір і розташування елементів залежно від розміру екрану користувача [11].

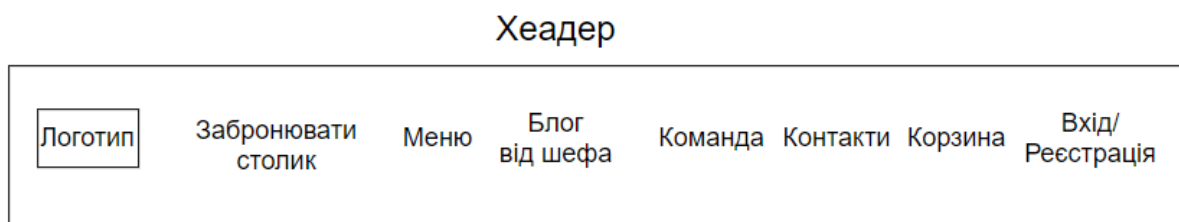


Рисунок 2.1 – Спроєктований header для веб-сайту “Гулівер”

Хедер включає логотип, який розміщений зліва і виконує роль посилання на головну сторінку сайту, що дозволяє користувачам легко повернутися на головну сторінку з будь-якого розділу сайту. Навігаційне меню містить основні розділи сайту: “Забронювати столик”, “Меню”, “Блог від шефа”, “Команда”, “Контакти”, “Кошик”. Ці розділи забезпечують швидкий доступ до ключової інформації та функцій сайту.

Іконка пошуку, яка відкриває поле для введення запиту, розташована праворуч від навігаційного меню. Це дозволяє користувачам швидко знайти необхідну інформацію на сайті. Перемикач мови, що дозволяє користувачам вибрати між різними мовами інтерфейсу, зазвичай розташовується у верхньому правому куті хедера. Додатково можна розмістити іконки

соціальних мереж (Facebook, Instagram, Twitter тощо) для швидкого переходу до офіційних сторінок ресторану, що дозволяє користувачам стежити за новинами та акціями ресторану в соціальних мережах.

Естетичний вигляд хедера має відповідати загальному дизайну сайту та відображати бренд ресторану (див. додаток Л). Використання корпоративних кольорів, шрифтів та логотипу допомагає створити цілісний та професійний вигляд. Важливо також враховувати зручність користувачів, забезпечуючи достатній контраст та читабельність тексту. Хедер повинен бути адаптивним та зручним у використанні на різних пристроях, включаючи мобільні телефони та планшети.

Проектування хедера є важливим етапом, який впливає на загальне враження користувачів від сайту та їх зручність користування. Ретельно продуманий хедер сприятиме підвищенню ефективності навігації та покращенню користувацького досвіду.

Проектування футера для веб-сайту ресторану “Гулівер” є важливим етапом, що забезпечує користувачів необхідною інформацією та додатковими можливостями для взаємодії з сайтом, що показний на рисунку 2.2. Футер розташований в нижній частині сторінки та містить важливі посилання і контактну інформацію [12].

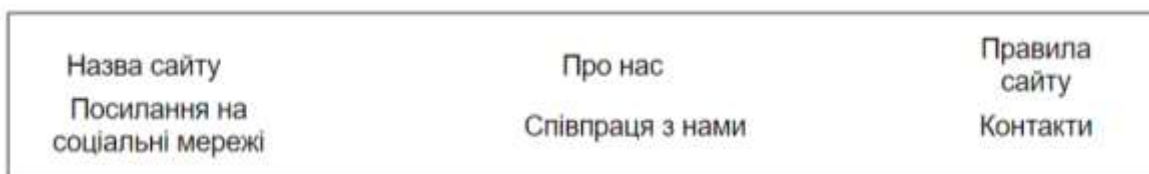


Рисунок 2.2 – Спроектований footer для веб-сайту “Гулівер”

Назва сайту відображається у футері для швидкої ідентифікації та повернення користувачів на головну сторінку. Клік по назві сайту перенаправляє користувача на головну сторінку.

Футер містить іконки соціальних мереж (Facebook, Instagram, Twitter тощо). Клік по іконці перенаправляє користувача на відповідну сторінку

соціальної мережі ресторану, де можна стежити за новинами, акціями та іншими оновленнями.

Цей розділ містить посилання на сторінку з детальною інформацією про ресторан. Клік по цьому посиланню перенаправляє користувача на сторінку “Про нас”, де описана історія ресторану, його концепція та цінності.

Посилання на сторінку співпраці, де користувачі можуть дізнатися про можливості співпраці з рестораном, наприклад, щодо проведення заходів або партнерських програм. Клік по цьому посиланню перенаправляє на відповідну сторінку з формою для зв'язку [13].

Цей розділ містить посилання на сторінку з правилами користування сайтом та політикою конфіденційності. Клік по цьому посиланню відкриває сторінку з детальною інформацією про правила та умови використання сайту.

Футер також містить розділ з контактною інформацією, включаючи адресу, телефон та електронну пошту ресторану. Клік по електронній пошті відкриває поштовий клієнт з попередньо заповненою адресою для швидкого зв'язку.

Дизайн футера повинен бути гармонійним з загальним стилем веб-сайту (див. додаток Б). Використання корпоративних кольорів, шрифтів та логотипу забезпечить цілісний вигляд сайту. Важливо забезпечити достатній контраст тексту для легкої читабельності, а також зручність користування на різних пристроях.

2.2 Проєктування блогу шеф-кухаря для веб-сайта “Гулівер”

Проєктування блогу шеф-кухаря є важливим аспектом веб-сайту ресторану “Гулівер”. Блог дозволяє шеф-кухарю ділитися рецептами, порадами, новинами та взаємодіяти з відвідувачами сайту. Створення блогу зручним та привабливим для користувачів допоможе підвищити інтерес до ресторану та збільшити кількість постійних клієнтів.

Блог шеф-кухаря складається з серії публікацій, кожна з яких містить зображення та заголовок [14]. Зображення привертає увагу користувачів та надає візуальне уявлення про тему статті, представлений на рисунку 2.3. Клік по зображенню перенаправляє користувача на сторінку з повним текстом публікації. Під зображенням розташовується заголовок публікації, який коротко описує її зміст і допомагає користувачам швидко зрозуміти, про що йдеться у статті. Клік по заголовку також перенаправляє на сторінку з повним текстом публікації.



Рисунок 2.3 – Спроектований блог для шеф-кухаря веб-сайту “Гулівер”

Публікації у блозі представлені у вигляді сітки, що дозволяє користувачам легко переглядати кілька статей одночасно. Це забезпечує зручність навігації та швидкий доступ до різних публікацій. Важливо додати функцію пошуку, що дозволяє користувачам знаходити конкретні статті за ключовими словами. Це покращить зручність використання блогу та дозволить користувачам швидко знаходити необхідну інформацію.

Додавання категорій та тегів до публікацій допоможе організувати контент та зробити його більш доступним для користувачів. Користувачі зможуть переглядати статті за категоріями або шукати публікації за конкретними тегами. Можливість залишати коментарі під публікаціями дозволить користувачам взаємодіяти з шеф-кухарем та іншими відвідувачами сайту. Це створить спільноту навколо блогу та допоможе підвищити залученість користувачів.

Додавання кнопок для поділу статтями у соціальних мережах дозволить користувачам легко ділитися цікавими публікаціями зі своїми друзями та підписниками. Це сприятиме підвищенню популярності блогу та ресторану в цілому.

Проектування блогу шеф-кухаря має бути спрямоване на створення зручного та інтерактивного інтерфейсу, що сприятиме залученню користувачів та підвищенню їх інтересу до контенту. Використання візуально привабливих елементів, зручної навігації та додаткових функцій дозволить створити успішний блог, який стане важливою частиною веб-сайту ресторану “Гулівер”.

2.3 Реалізація системи входу та реєстрації користувачів для веб-сайту “Гулівер”

Система входу дозволяє зареєстрованим користувачам вводити свої облікові дані для доступу до свого профілю. Форма входу містить поля для введення імені користувача та пароля, а також опцію “Remember Me”, яка зберігає сесію користувача для подальших відвідувань сайту без необхідності повторного входу [15].

Процес реєстрації користувачів передбачає заповнення форми з обов’язковими полями для введення імені, прізвища, електронної пошти, номера телефону, пароля та підтвердження пароля. Користувач також повинен надати згоду на обробку особистих даних для завершення реєстрації. Після заповнення всіх полів і натискання кнопки “Submit”, дані користувача

зберігаються в базі даних, і користувач отримує доступ до свого особистого кабінету, ця візуальна форма зображена на рисунку 2.4.

Зареєстровані користувачі мають переваги, такі як автоматичне заповнення контактних даних та адреси доставки під час оформлення замовлення, можливість переглядати історію своїх замовлень, отримувати спеціальні пропозиції та знижки.

Для забезпечення безпеки та захисту даних використовуються сучасні методи шифрування та захисту інформації. Паролі зберігаються у зашифрованому вигляді, а всі дані передаються через захищені з'єднання з використанням SSL сертифікатів.

Рисунку 2.4 показує спроектований блог для шеф-кухаря веб-сайту "Гулівер". Блог складається з трьох основних елементів, розташованих вертикально в межах великого прямокутника:

- Верхній елемент: горизонтальний прямокутник з текстом "Логін".
- Середній елемент: горизонтальний прямокутник з текстом "Пароль".
- Нижній елемент: менший горизонтальний прямокутник з текстом "Відправити".

Рисунок 2.4 – Спроектований блог для шеф-кухаря веб-сайту "Гулівер"

Для забезпечення ефективного управління контентом і зручності використання системи було прийнято рішення надати адміністраторам і шеф-кухарям окремі логіни і паролі для входу на веб-сайт "Гулівер". Це рішення було обрано з кількох причин.

По-перше, різні ролі в системі вимагають різних рівнів доступу та функціональних можливостей. Адміністратори мають доступ до широкого спектру функцій, включаючи додавання нових страв, редагування меню, перегляд та обробку замовлень, а також управління контентом сайту. Ці функції вимагають високого рівня доступу і відповідальності, тому адміністратори отримують унікальні облікові записи для забезпечення контролю та безпеки.

По-друге, шеф-кухарі мають свої унікальні завдання та функції, які включають створення і публікацію постів у блозі, додавання рецептів та кулінарних порад, а також взаємодію з клієнтами через коментарі. Надання шеф-кухарям окремих облікових записів дозволяє їм зосередитися на їхніх спеціалізованих завданнях, не зміщуючи їх з адміністративними функціями.

Нарешті, розділення доступу між адміністраторами та шеф-кухарями підвищує загальну безпеку системи. Кожен користувач має обмежений доступ до тих функцій, які йому необхідні для виконання своїх обов'язків. Це зменшує ризик випадкового або навмисного втручання у функціонування сайту, оскільки кожна роль має чітко визначені права та обов'язки [16].

Таким чином, надання окремих логінів і паролів для адміністраторів та шеф-кухарів дозволяє забезпечити ефективне управління веб-сайтом “Гулівер”, підвищує безпеку та забезпечує зручність користування для всіх залучених осіб.

2.4 Архітектура взаємодії між фронтенд та бекенд частинами веб-сайту “Гулівер”

Архітектура взаємодії між фронтенд та бекенд частинами веб-сайту “Гулівер” є ключовим елементом, що забезпечує ефективну роботу всієї системи. Веб-сайт побудовано з використанням сучасних технологій, таких як React для фронтенду та FastAPI для бекенду, що дозволяє створювати динамічні, масштабовані та безпечні веб-сайтів [17].

Фронтенд частина застосунку реалізована на основі бібліотеки React, яка дозволяє створювати інтерактивний та швидкий користувацький інтерфейс, що відповідно зображено на рисунку 2.5. React компоненти забезпечують відображення сторінок сайту, обробку подій користувача та взаємодію з бекендом через API запити. Використання React забезпечує можливість створення односторінкових застосунків (SPA), що покращує користувацький досвід завдяки швидкому завантаженню та оновленню контенту без перезавантаження сторінки.

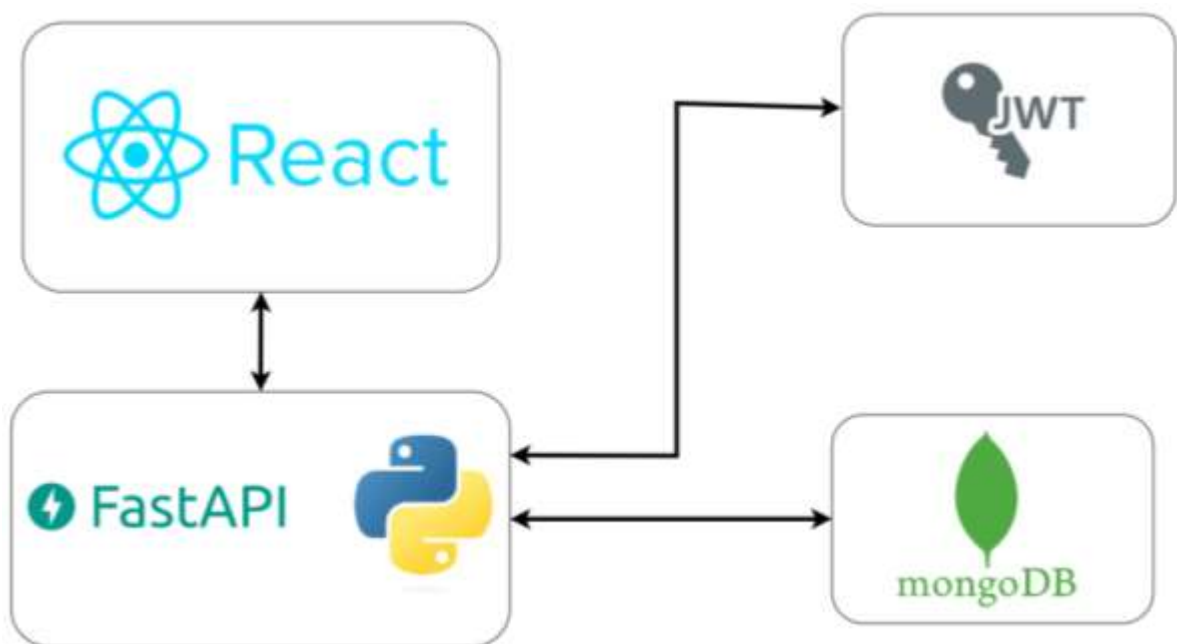


Рисунок 2.5 – Спроектвана архітектура веб-сайту “Гулівер”

Бекенд частина веб-сайту розроблена з використанням FastAPI, високопродуктивного фреймворку для створення веб-API з Python. FastAPI забезпечує обробку HTTP-запитів від фронтенду, виконує бізнес-логіку та взаємодіє з базою даних.

Одним із важливих аспектів архітектури є використання JSON Web Token (JWT) для аутентифікації та авторизації користувачів. JWT дозволяє створювати безпечні токени, які зберігаються на клієнтській стороні та

надсилаються з кожним запитом до бекенду для перевірки прав доступу. Це забезпечує безпеку даних та захист від несанкціонованого доступу.

База даних MongoDB використовується для зберігання інформації про користувачів, замовлення, меню та інші важливі дані. MongoDB обрана через її гнучкість у роботі з неструктурованими даними та високу продуктивність при роботі з великими обсягами даних [18].

Основні компоненти архітектури взаємодії включають фронтенд, бекенд, JWT та MongoDB. Комунікація між фронтендом та бекендом здійснюється через REST API, що дозволяє легко розширювати та підтримувати систему. React надсилає запити до FastAPI, який обробляє їх, взаємодіє з MongoDB та повертає відповідні дані або результати операцій назад до фронтенду. Ця архітектура забезпечує високу продуктивність, масштабованість та безпеку веб-сайту “Гулівер”, дозволяючи легко додавати нові функції та підтримувати існуючі.

2.5 Реалізація API для управління замовленнями на веб-сайті “Гулівер”

Реалізація API для управління замовленнями на веб-сайті “Гулівер” є важливим аспектом для забезпечення зручності та ефективності обробки замовлень. Для цієї задачі було обрано FastAPI завдяки його високій продуктивності та простоті використання.

API дозволяє виконувати операції створення, читання, оновлення та видалення для замовлень, що забезпечує легку інтеграцію з фронтендом, побудованим на React. Всі запити та відповіді між фронтендом і бекендом передаються у форматі JSON, що забезпечує стандартизовану та легку для обробки структуру даних.

Процес створення API за допомогою FastAPI включає кілька основних кроків. Спершу потрібно встановити FastAPI та Uvicorn, що є сервером для запуску FastAPI застосунків. Це можна зробити за допомогою `pip`. Після цього

створюється основний файл застосунку, наприклад, 'main.py', який буде містити головний код для API.

У кореневій директорії створюється файл main.py, який буде основним файлом застосунку FastAPI.

У цьому файлі визначено модель замовлення за допомогою Pydantic. Модель включає поля для назви страви, дати замовлення, дати доставки, ціни, імені та адреси клієнта. Для простоти, замовлення зберігаються у списку.

Основні маршрути API включають створення нового замовлення, отримання всіх замовлень, отримання конкретного замовлення за ідентифікатором, оновлення замовлення та видалення замовлення. Кожен з цих маршрутів обробляє відповідний HTTP-запит та взаємодіє зі списком замовлень.

Запуск FastAPI застосунку здійснюється за допомогою Uvicorn, який дозволяє розробникам тестувати та використовувати API в реальному часі. Це забезпечує зручний інтерфейс для обробки запитів і відповідей між фронтендом і бекендом [19].

Використання FastAPI дозволяє швидко створювати високопродуктивні та легко масштабовані веб-API. Цей приклад демонструє базову реалізацію API для управління замовленнями.

2.6 Висновок до другого розділу

Проектування основних компонентів веб-сайту "Гулівер" є важливим етапом у розробці функціонального та зручного веб-сайту для ресторану.

Проектування header для веб-сайту "Гулівер". Header є ключовим елементом, який забезпечує зручну навігацію та доступ до основних розділів сайту. Він включає логотип, навігаційне меню з основними розділами сайту та іконки соціальних мереж. Простота навігації, естетичний вигляд та адаптивний дизайн забезпечують позитивний користувацький досвід.

Проектування блогу шеф-кухаря для веб-сайта “Гулівер”. Блог шеф-кухаря є важливим аспектом для взаємодії з відвідувачами, дозволяючи ділитися рецептами, порадами та новинами. Використання візуально привабливих елементів, зручної навігації та додаткових функцій, таких як категорії, теги та коментарі, створює інтерактивний і цікавий інтерфейс для користувачів.

Реалізація системи входу та реєстрації користувачів для веб-сайту “Гулівер”. Система входу та реєстрації користувачів дозволяє забезпечити безпечний доступ до особистих кабінетів та додаткових функцій сайту. Зареєстровані користувачі отримують можливість автоматичного заповнення контактних даних під час замовлення, перегляду історії замовлень та отримання спеціальних пропозицій.

Архітектура взаємодії між фронтенд та бекенд частинами веб-сайту “Гулівер”. Використання React для фронтенду та FastAPI для бекенду забезпечує динамічний, масштабований та безпечний веб-сайт. JSON Web Token використовується для аутентифікації та авторизації користувачів, забезпечуючи безпеку даних. Комунікація між фронтендом та бекендом здійснюється через REST API, що дозволяє легко розширювати та підтримувати систему.

Реалізація API для управління замовленнями на веб-сайті “Гулівер”. Створення API для управління замовленнями за допомогою FastAPI забезпечує ефективність обробки замовлень та інтеграцію з фронтендом. Всі запити та відповіді передаються у форматі JSON, що забезпечує стандартизовану та легку для обробки структуру даних.

Цей розділ демонструє важливість детального проектування та реалізації основних компонентів веб-сайту для забезпечення високої якості обслуговування клієнтів та зручності користування веб-сайтом ресторану “Гулівер”.

РОЗДІЛ 3. ТЕСТУВАННЯ ТА ЕКСПЛУАТАЦІЯ ВЕБ-САЙТУ “ГУЛІВЕР”

3.1 Реалізована головна сторінка веб-сайту “Гулівер”

Як можна помітити на рисунку 3.1, головна сторінка веб-сайту “Гулівер” є ключовим елементом сайту, що забезпечує перше враження відвідувачів та надає основну інформацію про ресторан. Вона виконує функцію вітальної сторінки, де користувачі можуть ознайомитися з основними розділами сайту та отримати швидкий доступ до потрібної інформації.



Рисунок 3.1 – Створений header веб-сайту “Гулівер”

Реалізований хедер веб-сайту “Гулівер” є центральним елементом інтерфейсу користувача, який забезпечує легку навігацію та доступ до ключових розділів сайту (див. додаток М). Він містить логотип ресторану, основне навігаційне меню, контактну інформацію та форму для входу [20].

Логотип ресторану “Guliver” розташований у верхньому лівому куті новоствореного хедера. Він забезпечує ідентифікацію бренду і служить візуальною точкою відліку для користувачів. Основне навігаційне меню включає такі пункти: Головна, Меню, Команда, Блог Шефа, Кошик, що дозволяє користувачам швидко переходити до відповідних розділів сайту.

Також відображається номер телефону, що дозволяє користувачам швидко знайти контактний номер для дзвінків або запитів. Праворуч розташована форма для входу, яка включає поля для введення імені користувача і пароля. Є опція “Remember Me”, яка дозволяє користувачам

залишатися в системі навіть після закриття браузера, а кнопка “Log In” підтверджує введені дані і здійснює вхід.

Поруч із формою входу розташовані іконки для соціальних мереж, таких як Telegram, Instagram та Google, що дозволяє користувачам легко знайти та слідкувати за рестораном у соціальних мережах.

На рисунку 3.2 представлений реалізований хедер, який забезпечує користувачам легкий доступ до основних розділів сайту, надає можливість швидкого зв'язку з рестораном та здійснення авторизації. Для подальших покращень можна забезпечити адаптивний дизайн, щоб хедер коректно відображався на різних пристроях, включаючи мобільні телефони та планшети, а також додати можливість перемикання між різними мовами для зручності користувачів, що не розмовляють українською.

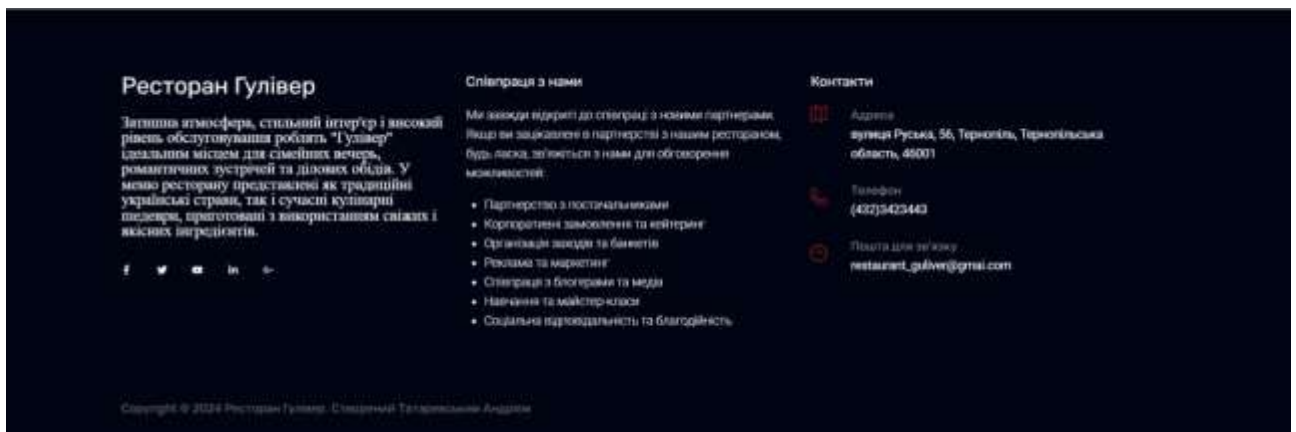


Рисунок 3.2 – Створений footer для веб-сайту “Гулівер”

Футер веб-сайту “Гулівер” є завершальним елементом інтерфейсу, який забезпечує користувачів важливою інформацією про ресторан, можливості співпраці та контактні дані (див. додаток К).

У лівій частині розташований блок з назвою ресторану “Гулівер” та коротким описом. Опис акцентує увагу на затишній атмосфері, стильному інтер'єрі та високому рівні обслуговування, що робить “Гулівер” ідеальним місцем для сімейних вечерів, романтичних зустрічей та ділових обідів. Меню

ресторану включає як традиційні українські страви, так і сучасні кулінарні шедеври, приготовані зі свіжих і якісних інгредієнтів.

В центральній частині зазначено, що ресторан завжди відкритий до співпраці з новими партнерами. Перераховані можливості співпраці включають партнерство з постачальниками, корпоративні замовлення та кейтеринг, організацію заходів та банкетів, рекламу та маркетинг, співпрацю з блогерами та медіа, навчання та майстер-класи, а також соціальну відповідальність та благодійність [21].

У правій частині футера розташовані контактні дані ресторану. Тут вказана адреса: вулиця Руська, 56, Тернопіль, Тернопільська область, 46001. Також зазначено телефонний номер (432)3423443 та електронну пошту restaurant_guliver@gmail.com, що дозволяє користувачам легко зв'язатися з рестораном для замовлень, запитів або будь-якої іншої інформації (див. додаток Н).

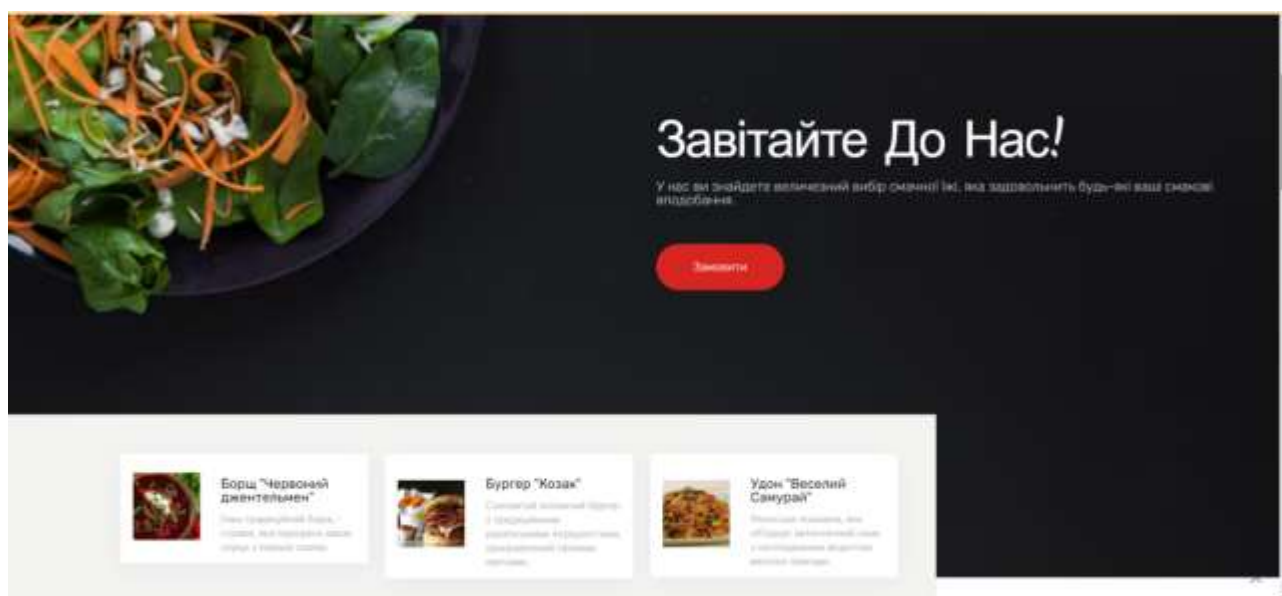


Рисунок 3.3 – Головна сторінка веб-сайту “Гулівер”

На рисунку 3.3 подана основна сторінка веб-сайту ресторану “Гулівер” включає кілька ключових елементів для створення привабливого і зручного інтерфейсу. Представлений великий банер із закликом “Завітайте до нас!”, який

привертає увагу відвідувачів. Тут же розміщені фотографії деяких популярних страв, таких як борщ “Червоний джентльмен”, бургер “Козак” та удон “Веселий самурай”, із короткими описами та можливістю замовлення. Цей розділ дозволяє відвідувачам швидко ознайомитися з популярними позиціями меню та зробити замовлення.



Рисунок 3.4 – Реклама на головній сторінці веб-сайту “Гулівер”

Також на рисунку 3.4 можна побачити, що на головній сторінці веб-сайту ресторану “Гулівер” представлена яскрава реклама страви “Бургер Козак” [22]. На зображенні показаний соковитий бургер, який містить яловичу котлету, салат, мариновані огірки, помідори, смажену цибулю, сир чеддер і спеціальний соус, розміщений у булочці з кунжутом. На передньому плані також можна побачити приправи та картоплю фрі. Вся композиція виглядає апетитно та привабливо. Поруч із бургером розміщена кнопка “Замовити”, яка закликає відвідувачів сайту зробити замовлення цієї страви. Це рекламне зображення ефективно привертає увагу та стимулює бажання скуштувати запропоновану страву.

В наступному розділі головної сторінки веб-сайту “Гулівер”, що зображено на рисунку 3.5 представлено девіз та відзнака. Розміщене

зображення суші, що символізує різноманітність страв у меню ресторану. Поруч розташований девіз, що підкреслює якість та натуральність інгредієнтів, використовуваних у стравах ресторану.



Рисунок 3.5 – Девіз та відзнака на головній сторінці веб-сайту “Гулівер”

Також можна помітити текст, який акцентує увагу на свіжості, якості та автентичності страв, приготованих за оригінальними рецептами. Видно відзнаку, що свідчить про високу оцінку ресторану авторитетним джерелом.

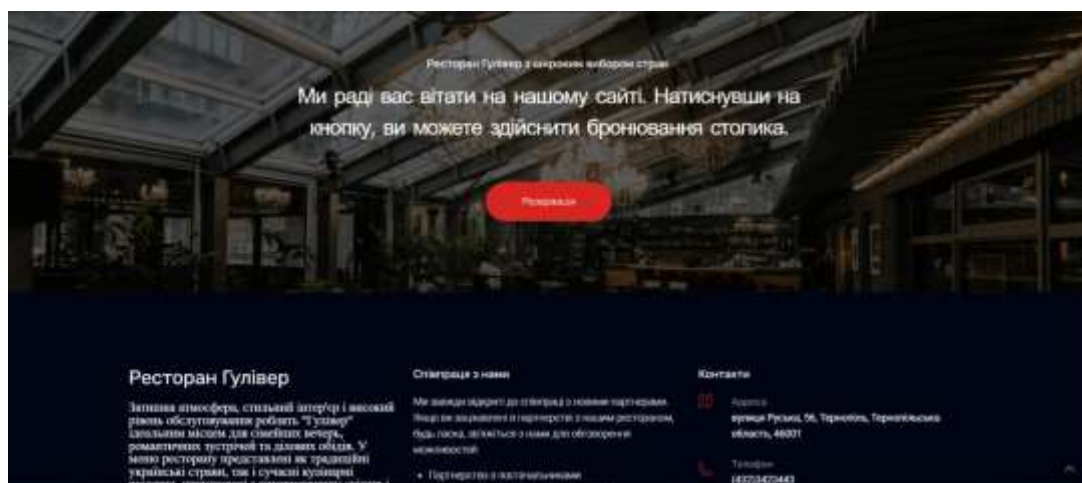


Рисунок 3.6 – Можливість зарезервувати столик на веб-сайті “Гулівер”

А ось на рисунку 3.6 відображена можливість зарезервувати столик на веб-сайті “Гулівер”. У верхній частині сторінки розташований заклик до дії, який спонукає користувачів до бронювання столика.

Центральне місце займає велика червона кнопка “Резервація”, яка привертає увагу та веде до форми бронювання столика. Такий дизайн інтерфейсу забезпечує інтуїтивно зрозумілу та зручну навігацію для користувачів, дозволяючи їм легко та швидко здійснювати бронювання.

Основна сторінка розроблена з урахуванням сучасних тенденцій веб-дизайну, забезпечуючи користувачам зручність навігації та приємний візуальний досвід. Всі елементи сторінки працюють разом, щоб створити позитивне враження про ресторан та заохотити відвідувачів до замовлення.

3.2 Реалізований функціонал для кошика веб-сайту “Гулівер”

Цей розділ присвячений реалізації кошика на веб-сайті ресторану “Гулівер” (див. додаток Д). Кошик є важливим елементом для будь-якого сайту, що надає послуги онлайн-замовлення, оскільки він дозволяє користувачам переглядати та керувати своїми замовленнями перед їх підтвердженням [23].

У кошику користувачі можуть бачити всі страви, які вони додали до свого замовлення, а також загальну суму замовлення. На рисунку 3.7 видно, що користувач додав до кошика бургер “Козак” вартістю 150,00 грн. Це дозволяє користувачам легко перевірити свої замовлення перед їх остаточним оформленням.



Рисунок 3.7 – Обраний товар в кошику веб-сайту “Гулівер”

Інтерфейс кошика включає кнопки для оформлення замовлення та повернення до меню, що візуально представлено на рисунку 3.7. Кнопка

“Оформити замовлення” дозволяє користувачам перейти до наступного кроку, де вони можуть підтвердити своє замовлення та обрати спосіб оплати. Кнопка “Повернутись в меню” дозволяє користувачам повернутись до перегляду меню, щоб додати або видалити страви зі свого замовлення.

Таким чином, реалізація кошика на веб-сайті ресторану “Гулівер” забезпечує зручність користувачів при здійсненні замовлень, дозволяючи їм легко керувати своїми замовленнями та контролювати загальну вартість перед оформленням.

ВАШЕ ЗАМОВЛЕННЯ	
СТРАВИ	ЦІНА
Motorcycle: Бургер "Козак" × 1	
Дата Замовлення: 19:00:50 10 Червня, 2024	150,00 ₴
Дата Та Час Доставки: 11 Червня, 2024 О 10:00	
Замовлення На	150,00 ₴
Доставка	Безкоштовна Доставка
Всього	150,00 ₴

Рисунок 3.8 – Короткі відомості замовлення в кошику веб-сайту “Гулівер”

На рисунку 3.8 подано таблицю з обраними стравами, їх кількістю та ціною. Відображається загальна сума замовлення, що дозволяє клієнту зрозуміти, скільки він витратить [24]. Крім того, клієнт може легко редагувати своє замовлення, змінюючи кількість страв або видаляючи непотрібні позиції. Інтерфейс також надає можливість переглянути деталі кожної страви, включаючи опис та склад. Це сприяє більшому інформуванню клієнтів та знижує ймовірність помилок під час замовлення.

Крім того, система автоматично обчислює знижки та акції, якщо вони доступні, і відображає їх у підсумковій сумі. Це допомагає клієнтам зекономити кошти та робить процес замовлення привабливішим. Користувачі можуть

зберегти своє замовлення на пізніше, що дозволяє їм повернутися до оформлення замовлення у зручний для них час.

Рисунок 3.9 – Форма для введення платіжних даних на веб-сайті “Гулівер”

Показано форму, яка представлена на рисунку 3.9, для введення платіжних даних клієнта. Ця форма включає поля для імені, прізвища, адреси, контактного телефону та електронної пошти [25]. Всі ці дані є обов'язковими для забезпечення коректної доставки замовлення.

Рисунок 3.10 – Вибір оплати та оформлення замовлення на веб-сайті “Гулівер”

На останньому етапі оформлення замовлення знаходиться кнопка вибору способу оплати та кнопка для завершення оформлення замовлення, наведено на рисунку 3.10. Клієнт може обрати оплату при доставці. Крім того, присутнє повідомлення про використання особистих даних відповідно до політики конфіденційності.

Перевіримо, чи насправді сформоване замовлення відправилось, для цього використаємо створене АРІ використовуючи Python, яке представлено на рисунку 3.11 [26]. Для цього виконаємо відповідний запит до нашого АРІ, що дозволить отримати інформацію про замовлення з бази даних та перевірити його коректність.



Рисунок 3.11 – Отримання всіх замовлень за допомогою FastAPI веб-сайту “Гулівер”

Створено маршрути для отримання і збереження даних замовлень, наприклад, POST для збереження нового замовлення і GET для отримання даних про існуючі замовлення, яке було протестоване на рисунку 3.12.



Рисунок 3.12 – Отримане замовлення створене за допомогою форми на веб-сайті “Гулівер”

Цей функціонал забезпечує зручність для адміністратора у перегляді та моніторингу замовлень, що дозволяє краще організовувати процес доставки та взаємодії з клієнтами.

Загалом, ця реалізація забезпечує зручний процес оформлення замовлення для клієнтів ресторану, дозволяючи їм легко переглянути обрані страви, ввести необхідну інформацію для доставки та вибрати спосіб оплати.

3.3 Тестування функціонально готового веб-сайту “Гулівер”

На головній сторінці, що представлена на рисунку 3.13, було реалізовано функцію входу для адміністратора. Це дозволяє адміністраторам сайту мати доступ до додаткових функцій, таких як управління стравами, перегляд замовлень та управління вмістом сайту.



Рисунок 3.13 – Вхід до веб-сайту “Гулівер” за адміністратора

Форма для входу містить поля для введення імені користувача та пароля, а також опцію “Remember Me”, яка дозволяє зберегти сесію для зручності. Після успішного входу адміністратор отримує доступ до спеціальних можливостей, які дозволяють ефективно керувати веб-сайтом ресторану.

Для адміністратора було реалізовано можливість додавати нові страви через спеціальну форму на сайті. Форма містить поля для введення назви страви, категорії, опису, складу, ціни та завантаження зображення [28]. Після заповнення всіх полів і натискання кнопки “Додати”, страву зберігається в базі даних і стає доступною в меню для користувачів. Цей функціонал забезпечує адміністраторам зручний інтерфейс для оновлення меню ресторану та додавання нових позицій, що сприяє підтриманню актуальності асортименту страв і підвищенню задоволеності клієнтів. Крім того на рисунку 3.14

представлено можливість адміністратором редагувати вже існуючі страви, оновлюючи їхні деталі або видаляючи їх з меню в разі необхідності (див. додаток Е). Це дозволяє підтримувати високий рівень обслуговування та оперативно реагувати на зміну потреб і уподобань клієнтів.


Додавання нової страви

Назва
Суп "Нова історія"

Категорія
Супи

Опис
Це ідеальне поєднання свіжих інгредієнтів і традиційної японської техніки, що створює новий смак кожного шматка

Складові
Лосось, тунець, авокадо, огірок, ікра, рис для супу, нері

Картинка

 Выберите файл | sushi021-700x700.jpg
 Maximum file size: 300 KB

Ціна
032

ДОБАВИ

Рисунок 3.14 – Форма для створення страви на веб-сайті “Гулівер”

На рисунку 3.15 було реалізовано спеціальний вхід для шеф-кухаря, що дозволяє йому отримати доступ до функцій блогу.

Username
chef

Password
.....

Remember Me

LOG IN

Рисунок 3.15 – Вхід за шеф-кухаря на веб-сайті “Гулівер”

Після авторизації, шеф-кухар може створювати, редагувати та публікувати пости з рецептами, кулінарними порадами та іншими матеріалами,

що представлено на рисунку 3.16. Крім того, він може відповідати на коментарі відвідувачів і підтримувати активну взаємодію з клієнтами через блог.

Рисунок 3.16 – Форма для створення нового поста на веб-сайт “Гулівер”

Це дозволяє шеф-кухарю ділитися своїм професійним досвідом, залучати клієнтів до дискусій і підвищувати їхню зацікавленість у ресторані.

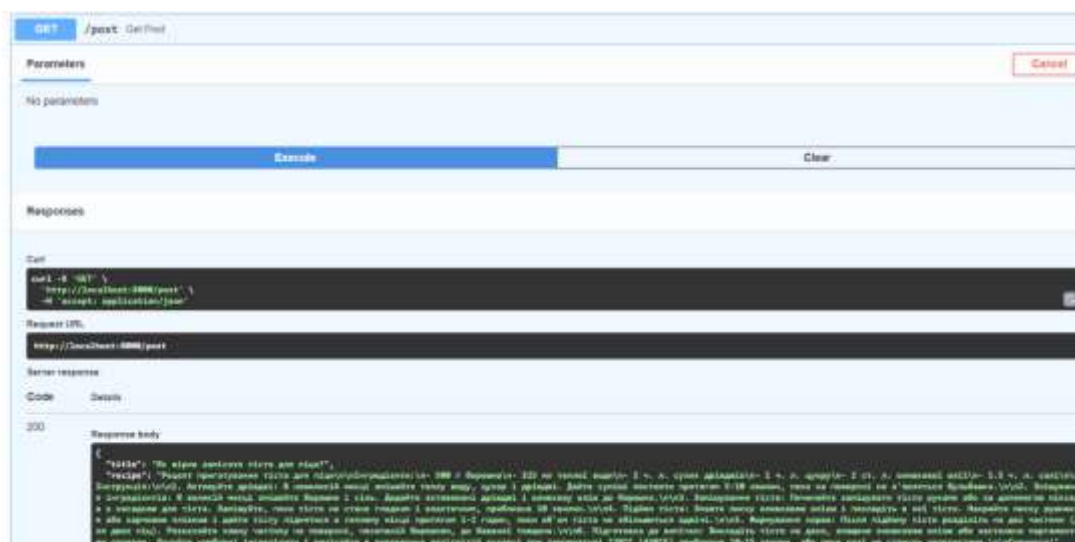


Рисунок 3.17 – Успішне створення поста шеф-кухарем на веб-сайті “Гулівер”

За допомогою FastAPI було реалізовано можливість перегляду результатів через зручний інтерфейс документації. Цей інтерфейс, що

зображений на рисунку 3.17, дозволяє отримувати всі пости, додані шеф-кухарем, використовуючи метод GET до кінцевої точки “/post” [28]. На зображенні можна побачити приклад успішного запиту, де у відповідь повертається інформація про пост, включаючи назву, рецепт та інгредієнти. Це забезпечує зручний спосіб взаємодії з API для отримання необхідних даних та перевірки їхньої коректності.

Різотто з грибами

Різотто з грибами - це ароматна і кремозна страва, що поєднує в собі ніжний рис арборіо та смачні лісові гриби.

300 г рису арборіо 200 г лісових грибів 1 цибулина 2 зубчики часнику 1 л овочевого бульйону 100 мл білого вина 50 г пармезану Оливкова олія, вершкове масло, сіль, перець



Наріжте цибулю та часник дрібними кубиками. Очистіть гриби і наріжте їх скибочками. У великій сковороді нагрійте оливкову олію та вершкове масло

Салат "Цезар" з куркою

Салат "Цезар" з куркою - це популярна страва з ніжною куркою, хрусткими крутонами та ароматним пармезаном, заправлена спеціальним соусом.

1 куряча грудка 1 романо салат 100 г крутони 50 г пармезану 2 яйця 1 ч.л. гірчиці 2 зубчики часнику 2 анчоуса Оливкова олія, лимонний сік, сіль, перець



Відваріть курячу грудку до готовності і наріжте скибочками. Відваріть яйця і наріжте їх на

Лазанья з м'ясом

Лазанья з м'ясом - це класична італійська страва, яка складається з шарів пасти, м'ясного соусу та сиру. Ідеально підходить для родинних вечерь.

500 г фаршу (яловичина або свинина) 1 цибулина 2 зубчики часнику 400 г томатного соусу 250 г пасти для лазаньї 200 г моцарели 100 г пармезану Оливкова олія, сіль, перець, орегано

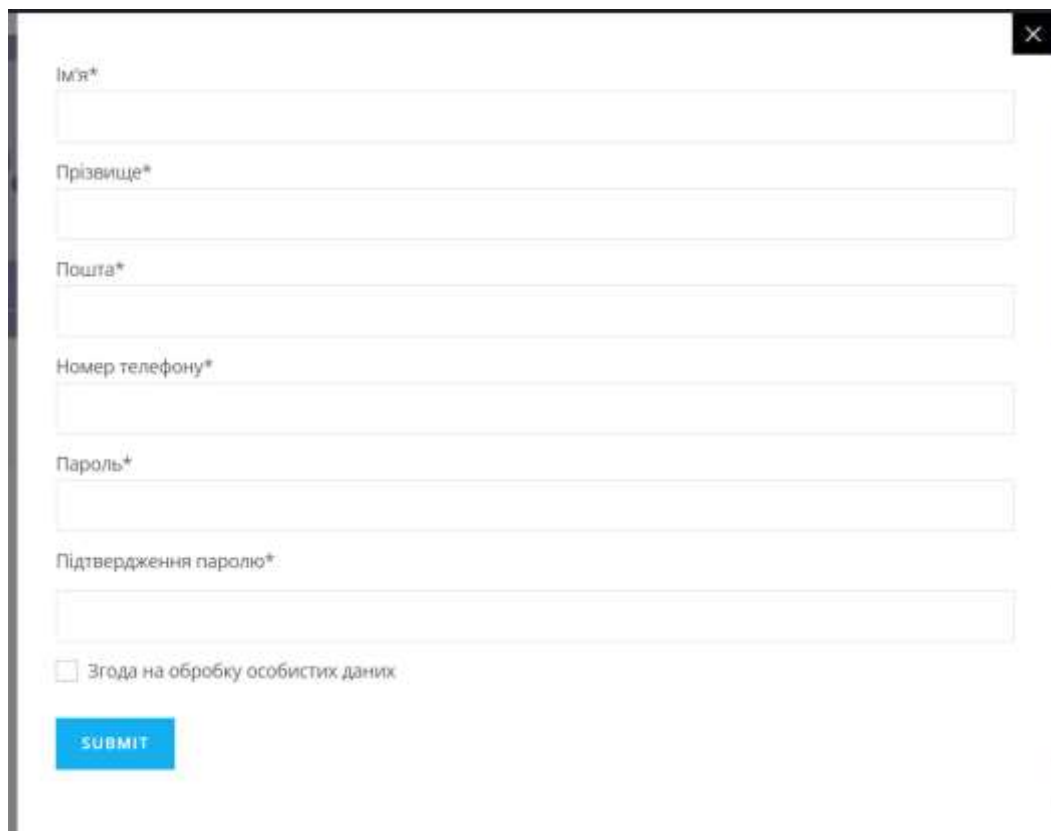


Рисунок 3.18 – Пости створені шеф-кухарем на веб-сайті “Гулівер”

Ця сторінка зображена на рисунку 3.18 демонструє інформацію про страви з рецептами та зображеннями. Було реалізовано інтеграцію з бекендом, що дозволяє отримувати дані про страви та відображати їх на фронтенді. Для кожної страви представлено її назву, опис, список інгредієнтів та зображення. Це робить сторінку привабливою для відвідувачів і допомагає їм вибрати бажану страву.

Реалізована архітектура дозволяє користувачам переглядати меню, додавати страви до кошика, оформляти замовлення, а також залишати відгуки та коментарі (див. додаток Ж). Це створює інтерактивний досвід для відвідувачів і сприяє підвищенню їхньої задоволеності сервісом.

Перейдемо до реєстрації користувача. Для цього на сайті реалізована спеціальна форма, яка дозволяє новим користувачам створювати обліковий запис.



The image shows a registration form with the following fields and elements:

- Ім'я* (Name) - text input field
- Прізвище* (Surname) - text input field
- Пошта* (Email) - text input field
- Номер телефону* (Phone number) - text input field
- Пароль* (Password) - text input field
- Підтвердження паролю* (Confirm password) - text input field
- Згода на обробку особистих даних (Agree to terms of use)
- SUBMIT button

Рисунок 3.19 – Процес реєстрації користувача на веб-сайті “Гулівер”

Подана форма на рисунку 3.19 містить обов'язкові поля для введення імені, прізвища, електронної пошти, номера телефону, пароля та підтвердження пароля [29]. Також користувач повинен надати згоду на обробку особистих даних, що є обов'язковою умовою для завершення реєстрації. Після заповнення всіх полів і натискання кнопки “Submit”, дані користувача зберігаються в базі даних, і користувач отримує доступ до свого особистого кабінету, де може керувати своїми замовленнями, бронюваннями та іншою інформацією.

Реєстрація користувача забезпечує зручність при оформленні замовлень і бронювань, оскільки вся необхідна інформація автоматично підтягується під час цих процесів (див. додаток В).

3.4 Ієрархія веб-сайту “Гулівер”

У процесі розробки веб-сайту “Гулівер” була створена комплексна ієрархія компонентів для забезпечення ефективної роботи та взаємодії між фронтенд і бекенд частинами [31].

Фронтенд частина веб-сайту побудована на бібліотеці React і включає різноманітні компоненти для основних сторінок, таких як головна, меню, контакти, команда, кошик, додавання страв, вхід та реєстрація користувачів .



Рисунок 3.20 – Ієрархія веб-сайту “Гулівер”

Як можна помітити на рисунку 3.20, фронтенд частина веб-сайту ресторану “Гулівер” побудована з використанням бібліотеки React, що дозволяє створювати динамічні та інтерактивні веб-сторінки. Структура файлів

фронтенду організована таким чином, щоб забезпечити зручність розробки, масштабованість і легкість у підтримці.

У кореневій папці фронтенд проєкту знаходиться кілька ключових папок та файлів:

- `index.html`: Основний HTML файл, який слугує точкою входу для React-застосунку. Він містить кореневий елемент, в який буде вбудовано React-компоненти.

- `favicon.ico` та інші файли логотипів: Використовуються для відображення іконки сайту в браузері.

- `App.js`: Головний компонент React-застосунку, який відповідає за маршрутизацію та структуру основних сторінок. Він містить маршрути для різних компонентів, таких як головна сторінка, меню, контакти, команда, кошик, додавання страв, вхід та реєстрація.

- `App.css`: Файл стилів для головного компонента. Він містить CSS правила, які визначають зовнішній вигляд застосунку.

- `index.js`: Точка входу для React-застосунку. Цей файл рендерить головний компонент `App.js` у кореневий елемент HTML файлу.

- `index.css`: Загальний файл стилів для всього застосунку.

- `AuthContext.js`: Файл контексту, який забезпечує управління станом авторизації користувача. Він дозволяє зберігати інформацію про аутентифікацію та забезпечує захищений доступ до приватних маршрутів.

- `CartContext.js`: Файл контексту для управління станом кошика. Він дозволяє зберігати інформацію про додані до кошика страви та керувати ними.

- `Header.js`: Компонент, що відповідає за відображення заголовку сайту, включаючи логотип, навігаційне меню та форму входу.

- `Footer.js`: Компонент для відображення нижнього колонтитула сайту, який містить інформацію про ресторан, контактні дані та посилання на сторінки соціальних мереж [31].

- MainPage.js: Компонент, що відповідає за відображення головної сторінки сайту. Він містить вітальний текст та основну інформацію про ресторан.

- ContactPage.js: Компонент для сторінки контактів, де відображається контактна інформація ресторану.

- TeamPage.js: Компонент для сторінки команди, де відображаються профілі команди ресторану.

- MenuPage.js: Компонент для сторінки меню, який відображає перелік страв, доступних для замовлення.

- CartPage.js: Компонент для сторінки кошика, де користувач може переглядати додані до кошика страви, змінювати їх кількість та оформляти замовлення.

- AddDish.js: Компонент для сторінки додавання нових страв, доступний лише для адміністратора. Він містить форму для введення інформації про нову страву.

- Login.js: Компонент для сторінки входу користувачів.

- Register.js: Компонент для сторінки реєстрації нових користувачів.

Ця структура файлів забезпечує чітку організацію коду, роблячи його легким для розуміння та підтримки. Кожен компонент відповідає за окрему частину функціоналу, що дозволяє розробникам легко додавати нові функції або змінювати існуючі без порушення роботи всього застосунку. Інтеграція з бекендом через API дозволяє отримувати та відправляти дані, забезпечуючи динамічну взаємодію користувачів із сайтом.

Бекенд частина веб-сайту ресторану “Гулівер” побудована з використанням FastAPI – сучасного, високопродуктивного веб-фреймворку для створення API з Python. Бекенд забезпечує обробку запитів від фронтенду, взаємодію з базою даних та виконання бізнес-логіки застосунку. Ось структура файлів бекенд частини:

- `routes.py`: Містить маршрути (ендпоінти) API, визначені за допомогою FastAPI. Тут обробляються HTTP-запити, такі як GET, POST, PUT, DELETE, та взаємодія з моделями бази даних.

- `server.js`: Основний файл конфігурації та запуску серверу. Він налаштовує підключення до бази даних, ініціалізує FastAPI та реєструє маршрути API.

- `node_modules`: Директорія, яка містить встановлені пакети Node.js, необхідні для роботи бекенду. Вона включає бібліотеки для обробки запитів, підключення до бази даних та інші утиліти.

- `package.json`: Файл, що містить метадані проєкту Node.js. Він визначає залежності проєкту, скрипти для запуску та інші налаштування. З його допомогою керуються всіма пакетами та бібліотеками, необхідними для роботи бекенду.

- `package-lock.json`: Файл, який автоматично генерується та містить точну версію всіх встановлених пакетів, забезпечуючи стабільність залежностей проєкту.

- `main.py`: Основний файл для запуску FastAPI сервера. Він ініціалізує FastAPI додаток, налаштовує маршрути та підключення до бази даних.

Кожен файл та директорія мають свої чітко визначені ролі, що сприяє організації коду та його підтримці. Ось як взаємодіють основні файли:

- `main.py` ініціалізує сервер FastAPI та налаштовує підключення до бази даних.

- `models.py` містить визначення моделей бази даних, що представляють структуру даних.

- `routes.py` містить маршрути API, що обробляють запити від клієнтів (фронтенду), використовуючи моделі для взаємодії з базою даних.

- `package.json` та `package-lock.json` визначають та фіксують залежності проєкту, забезпечуючи стабільність та відтворюваність середовища розробки.

Для забезпечення безпеки та авторизації було реалізовано систему входу та реєстрації користувачів, що дозволяє розділяти ролі адміністратора та шеф-

кухаря. Адміністратор має можливість додавати нові страви, оновлювати існуючі дані та керувати замовленнями. Шеф-кухар може створювати та редагувати пости у блозі, ділитися рецептами та кулінарними порадами.

Таким чином, розроблена ієрархія та взаємодія компонентів забезпечують оптимальну роботу веб-сайту ресторану “Гулівер”, роблячи його зручним для користувачів та легким у підтримці та розвитку для розробників. Впровадження FastAPI для створення API забезпечує швидку обробку запитів та легку інтеграцію з базою даних, що робить систему ефективною та масштабованою [32].

3.5 Висновок до третього розділу

У цьому розділі було детально розглянуто процес тестування та експлуатації веб-сайту ресторану “Гулівер”. Розроблена головна сторінка веб-сайту є важливим елементом, що забезпечує позитивне перше враження відвідувачів та надає швидкий доступ до основних розділів сайту. Реалізовані header та footer включають необхідні елементи для зручної навігації, контактної інформації та соціальної взаємодії.

Функціонал кошика дозволяє користувачам зручно переглядати та керувати своїми замовленнями, що підвищує зручність користування сайтом. Завдяки формі для введення платіжних даних, клієнти можуть швидко і безпечно завершувати свої замовлення. Реалізація перевірки замовлень за допомогою API на Python забезпечує ефективний контроль та моніторинг даних.

Для адміністратора були створені зручні інтерфейси для додавання нових страв та управління меню, що дозволяє підтримувати актуальність асортименту ресторану. Шеф-кухар має можливість вести блог, ділитися рецептами та взаємодіяти з клієнтами, що сприяє залученню нових відвідувачів.

Розроблена ієрархія компонентів веб-сайту та інтеграція з бекендом забезпечують надійну та ефективну роботу системи. Використання сучасних

технологій, таких як React для фронтенду та FastAPI для бекенду, дозволило створити динамічний та масштабований веб-сайт, який відповідає високим вимогам до продуктивності, безпеки та зручності користування.

Загалом, реалізація веб-сайту ресторану “Гулівер” показала високу ефективність використаних рішень та технологій, забезпечивши інтуїтивно зрозумілий інтерфейс для користувачів та гнучкість у керуванні контентом для адміністрації ресторану.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Забезпечення психоемоційного здоров'я та запобігання професійному вигоранню у користувачів ПК

У процесі своєї діяльності людина використовує не тільки свої фізичні можливості, а й витрачає значні психологічні зусилля, такі, як особливості характеру, воля, розумові здібності тощо.

Людина в діяльності виступає як особистість, що має певні мотиви і намічені цілі. Мотивами можуть виступати потреби, почуття тощо. Для здійснення діяльності необхідно мати об'єкт діяльності, внутрішні спонуки, а також співвідношення спонук і цілей людини, які вона хоче досягнути в результаті своєї діяльності. Наприклад, людину до діяльності спонукає або особисте збагачення (задоволення особистих потреб), або неможливість фізичного існування без діяльності.

На відміну від фізичної, розумова діяльність супроводжується меншими витратами енергетичних запасів, але це не значить, що вона є легкою. Основним робочим органом під час такого виду діяльності виступає мозок. Під час розумової діяльності значно активізуються аналітичні та синтетичні функції ЦНС, ускладнюється прийом і переробка інформації, виникають функціональні зв'язки, нові комплекси умовних рефлексів, зростає роль функцій уваги, пам'яті, напруження зорового та слухового аналізаторів і навантаження на них. Для розумової діяльності характерні напруження уваги, сприйняття, пам'яті, велика кількість стресів, малорухливість, вимушена поза. Все це зумовлює застійні явища у м'язах ніг, органах черевної порожнини і малого тазу, погіршується постачання мозку киснем, зростає потреба в глюкозі. Погіршуються також функції зорового аналізатора: стійкість ясного бачення, гострота зору, зорова працездатність, збільшується час зорово-моторної реакції.

Розумовій праці притаманний найбільший ступінь напруження уваги — в середньому у 5-10 разів вищий, ніж при фізичній праці. Завершення робочого

дня зовсім не перериває процесу розумової діяльності. Розвивається особливий стан організму - втома, що з часом може перетворитися на перевтому. Все це призводить до порушення нормального фізіологічного функціонування організму. Люди, що займаються розумовою діяльністю, навіть у стані перевтоми здатні довгий час виконувати свої обов'язки без особливого зниження рівня працездатності і продуктивності. Переважно люди розумової діяльності не здатні вимкнути механізм переробки інформації на ніч; вони працюють не лише 8-12 годин на добу, а майже постійно з короткими переключеннями. Це і є підтвердженням так званої інформативної теорії, згідно з якою людина під час сну перероблює інформацію, отриману в період активної бадьорості. Фізичний і розумовий види діяльності вимагають різного напруження певних функціональних систем організму, тому навантаження необхідно класифікувати відповідно до важкості і напруженості. Важкість праці – це напруження функціональних систем, які зумовлені фізичним навантаженням. Напруженість характеризує рівень напруження центральної нервової системи.

Психофізіологічні небезпеки у сучасному світі є чинниками цілісності чи розладу, стійкості чи дисгармонії, спокою чи тривоги, успіху чи невдач, фізичного та морального благополуччя. На сьогодні не існує жодного фактора психофізіологічних небезпек, що не впливав би на людину. Кожен з цих факторів залежно від тривалості дії можна віднести до постійних чи тимчасових. Психофізіологічними факторами потенційної небезпеки постійної дії слід вважати:

- недоліки органів відчуття (дефекти зору, слуху тощо);
- порушення зв'язків між сенсорними та моторними центрами, внаслідок чого людина не здатна реагувати адекватно на ті чи інші зміни, що сприймаються органами відчуття;
- дефекти координації рухів (особливо складних рухів та операцій, прийомів тощо);
- підвищена емоційність;

- відсутність мотивації до трудової діяльності (незацікавленість в досягненні цілей, невдоволення оплатою праці, монотонність праці, відсутність пізнавального моменту, тобто нецікава робота, тощо).

Психофізіологічними факторами потенційної небезпеки тимчасової дії є:

- недостатність досвіду (поява імовірної помилки, невірні дії, напруження нервово-психічної системи, побоювання припуститися помилки);
- необережність (може призвести до ураження не лише окремої людини, а й всього колективу);
- втома (розрізняють фізіологічне та психологічне втомлення);
- емоційні явища (особливо конфліктні ситуації, душевні стреси, пов'язані з побутом, сім'єю, друзями, керівництвом). На успіх діяльності особливо впливає стан людини. Будь-який вид діяльності викликає втому.

Вивчення умов праці показало, що існує багато факторів, які впливають на продуктивність праці.

Вони формують загальний настрій на працю та полегшують або ускладнюють появу трудових зусиль. Розробляючи критерії професіографічної оцінки (опис характеристик праці) ступеня нервового напруження в процесі праці, користуються характеристиками, які відображають напруження сенсорного апарату, вищих нервових центрів, що забезпечують функції уваги, мислення, регуляції рухів. На сьогодні складено спеціальні таблиці класифікації праці за ступенем нервово-емоційного напруження, в основу яких покладено такі показники:

- інтелектуальне та емоційне навантаження;
- тривалість зосередженого спостереження;
- кількість об'єктів одночасного спостереження, що є виробничо-важливими;
- кількість сигналів на годину;
- темп;
- час активних дій;
- необхідність самостійного пошуку пошкоджень;

- монотонність праці;
- напруження зору;
- точність виконання роботи;
- змінність;

Ступінь працездатності визначається також типом нервової системи. Сильний тип має найбільшу працездатність, слабкий — незначну. Працездатність залежить від таких факторів, як вік, здоров'я людини, стать, навички у роботі, санітарно-гігієнічні умови тощо. Певною мірою на неї впливають і мотивація, і моральні та матеріальні стимули.

Подовжити стійку працездатність можна за рахунок: оптимального рівня напруги психофізіологічних функцій, комфортними умовами праці, правильним поєднанням режимів праці та відпочинку, проведенням фізкультурних пауз та емоційного розвантаження, використання спеціальних психофармакологічних засобів чи тонізуючих напоїв, спеціальним інформуванням людини про результати її діяльності, наглядом і контролем за її роботою. Комфортні умови забезпечують, крім високої працездатності, добре самопочуття; при цьому не виникають небезпечні напруги компенсаторних систем організму; здоров'я людини не погіршується тривалий час. Відомо, що на працездатність, продуктивність праці, на життєдіяльність загалом впливає відпочинок. Відпочинок може бути двох типів — активний і пасивний. Відомо, що неспокійні ночі, неприємні сновидіння з переживанням жахів, небезпек, страху, коли життя ніби висить на волоссю, а сили немає, щоб дати відпір, — все це негативно позначається на стані людини, а відповідно на активності її життєдіяльності. [33]

4.2 Ергономічні вимоги до робочого місця оператора ПК

Обладнання та організація робочих місць користувачів ПК мають забезпечувати відповідність конструкцій усіх елементів робочого місця та їх

взаємного розташування ергономічним вимогам з урахуванням характеру й особливостей трудової діяльності відповідно до ДСанПіН 3.3.2.007-98.

При розташуванні елементів робочого місця користувача ПК слід ураховувати: робочу позу користувача, простір для розміщення користувача, можливість огляду елементів робочого місця, можливість ведення записів, розміщення документації і матеріалів, які використовуються користувачем. Конструкція робочого місця користувача ПК має забезпечити підтримання оптимальної робочої пози.

Робочі місця з ПК слід так розташовувати відносно вікон, щоб природне світло падало збоку переважно зліва. Робочі місця з ПК повинні бути розташовані від стіни з вікнами на відстані не менш ніж 1,5 м, від інших стін – на відстані не менше ніж 1 м. При розміщенні робочого місця поряд з вікном кут між екраном монітора і площиною вікна повинен складати не менше 90° (для виключення відблисків), частину вікна, що прилягає, бажано зашторити. Недопустиме розташування ПК, при якому працюючий повернений обличчям або спиною до вікон кімнати або до задньої частини ПК, у яку монтуються вентилятори. При розміщенні робочих столів з ПК слід дотримуватись таких відстаней: між бічними поверхнями ПК – 1,2 м, від тильної поверхні одного ПК до екрана іншого ПК – 2,5 м.

Монітор повинен бути встановлений таким чином, щоб верхній край екрана знаходився на рівні очей. Екран монітора ПК має розташовуватися на оптимальній відстані від очей користувача, що становить 600 – 700 мм, але не ближче ніж 600 мм з урахуванням розміру літерноцифрових знаків і символів. Для забезпечення точного та швидкого зчитування інформації в зоні найкращого бачення площа екрана монітора повинна бути перпендикулярною нормальній лінії зору. Клавіатура повинна бути розташована так, щоб на ній було зручно працювати двома руками. Клавіатуру слід розміщувати на поверхні столу на відстані 100 – 300 мм від краю. Кут нахилу клавіатури до столу повинен бути в межах від 5 до 15°, зап'ястя та долоні рук мають розташовуватися горизонтально до площини столу.

Принтер має бути розміщений у зручному для користувача положенні, так, щоб максимальна відстань від користувача до клавіш управління принтером не перевищувала довжину витягнутої руки користувача.

Конструкція робочого стола повинна забезпечувати можливість оптимального розміщення на робочій поверхні обладнання, що використовується, з урахуванням його кількості та конструктивних особливостей (розмір монітора, клавіатури, принтера, ПК та ін.) і документів, а також урахувати характер роботи, що виконується.

Висота робочої поверхні столу з ПК має регулюватися в межах 680 – 800 мм, а ширина і глибина – забезпечувати можливість виконання операцій у зоні досяжності моторного поля (рекомендовані розміри: 600 – 1400 мм, глибина – 800 – 1000 мм). Робочий стіл повинен мати простір для ніг висотою не менше ніж 600 мм, шириною – не менше ніж 500 мм, глибиною (на рівні колін) – не менше ніж 450 мм, на рівні простягнутої ноги – не менше ніж 650 мм. Ноги не повинні бути витягнені при сидінні далеко вперед, тому що в такому разі м'язи будуть надто напружені; положення “нога на ногу” не рекомендується, тому що підвищується тиск на сідничний нерв і порушується кровообіг ніг.

Робочий стілець має бути підйомно-поворотним, регульованим за висотою, з кутом нахилу сидіння та спинки, поверхня сидіння має бути плоскою, передній край – заокругленим. Регулювання за кожним із параметрів має здійснюватися незалежно, легко і надійно фіксуватися. Висота поверхні сидіння має регулюватися в межах 400 – 500 мм, а ширина і глибина становити не менше ніж 400 мм. Кут нахилу сидіння – до 15° вперед і до 5° – назад. Висота спинки стільця має становити 300 ± 20 мм, ширина – не менше ніж 380 мм. Кут нахилу спинки має регулюватися в межах 1 – 30° від вертикального положення. Відстань від спинки до переднього краю сидіння має регулюватися в межах 260 – 400 мм. Для зниження статичного напруження м'язів верхніх кінцівок слід використовувати стаціонарні або змінні підлокітники завдовжки не менше ніж 250 мм, завширшки – 50 – 70 мм, що регулюються за висотою над сидінням у межах 230 – 260 мм і відстанню між підлокітниками у межах 350 –

500 мм. Поверхня сидіння і спинки стільця має бути напівм'якою з нековзним повітронепроникним покриттям, що легко чиститься і не електризується. Робоче місце має бути обладнане підставкою для ніг шириною не менше ніж 300 мм, глибиною – не менше ніж 400 мм, що регулюється за висотою в межах до 150 мм і за кутом нахилу опорної поверхні підставки – до 20°. Підставка повинна мати рифлену поверхню і бортик по передньому краю висотою 10 мм.[34-35]

4.3 Висновки до четвертого розділу

Забезпечення психоемоційного здоров'я та запобігання професійному вигоранню, а також створення безпечних умов праці для розробників за ПК є надзвичайно важливими аспектами у сучасному робочому середовищі. Розумова діяльність, яка супроводжує роботу за комп'ютером, потребує значного психофізіологічного напруження, що може призвести до стресу та втоми. Для мінімізації негативного впливу на здоров'я працівників необхідно створювати комфортні умови праці, дотримуватися ергономічних вимог до робочого місця та обладнання.

Оптимальне розміщення елементів робочого місця, правильне освітлення, регулювання висоти столів і стільців, забезпечення підтримки правильної робочої пози – все це сприяє зниженню рівня стресу та запобігає виникненню професійного вигорання. Крім того, важливо забезпечити можливість фізичних пауз і відпочинку для підтримки високого рівня працездатності та загального фізичного і психічного благополуччя працівників.

Таким чином, комплексний підхід до організації робочих місць користувачів ПК, що включає як ергономічні, так і психофізіологічні аспекти, є ключовим для створення безпечного та здорового робочого середовища.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи освітнього рівня “Бакалавр” був спроектований та практично реалізований веб-сайт ресторану “Гулівер”, призначений для покращення взаємодії з клієнтами та підвищення якості обслуговування. Веб-сайт реалізовано засобами FastAPI, React та MongoDB.

У першому розділі кваліфікаційної роботи:

- проведено аналіз предметної області;
- сформовано перелік вимог до веб-сайту ресторану “Гулівер”;
- визначено перелік зовнішніх сутностей та описано варіанти використання веб-сайту “Гулівер”;
- визначено середовище та засоби розробки;
- сформовано архітектуру веб-сайту “Гулівер”;
- запропоновано структуру веб-сайту “Гулівер”.

У другому розділі кваліфікаційної роботи:

- спроектовано структуру функціональних компонентів;
- сформовано структури елементів веб-сайту “Гулівер”;
- описано елементи інтерфейсу в різних режимах експлуатації веб-сайту “Гулівер”.

У третьому розділі кваліфікаційної роботи:

- розроблено процес тестування та експлуатації веб-сайту “Гулівер”;
- проведено аналіз результатів тестування;
- запропоновано заходи для забезпечення безпеки та продуктивності веб-сайту;
- спроектовано інтеграцію системи аутентифікації та авторизації.

У четвертому розділі кваліфікаційної роботи “Безпека життєдіяльності, основи охорони праці” розглянуто питання забезпечення психоемоційного здоров'я та запобігання професійному вигоранню у користувачів ПК, а також ергономічні вимоги до організації їх робочих місць.

ПЕРЕЛІК ДЖЕРЕЛ

1 10 Top Restaurant Website Designs To Analyze Or Why Your Restaurant Needs A Website [Електронний ресурс] // Cadabra. – 2020. – Режим доступу до ресурсу: <https://cadabra.studio/best-restaurant-websites-designs/>.

2 Responsive vs Adaptive Web Design: Comparing Guide [Електронний ресурс] // Medium. – 2023. – Режим доступу до ресурсу: <https://medium.com/theymakedesign/responsive-vs-adaptive-web-design-1da9e5398669>.

3 Best front-end technologies for web development to Use in 2024 [Електронний ресурс] // Canadian Agency. – 2023. – Режим доступу до ресурсу: <https://canadian.agency/best-front-end-technologies-for-web-development/>.

4 How to Choose the Right Backend Technology? [Електронний ресурс] // Geeksforgeeks. – 2023. – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/how-to-choose-the-right-backend-technology/>.

5 How to Make a Website For Your Restaurant: A Complete Guide [Електронний ресурс] // Medium. – 2023. – Режим доступу до ресурсу: https://medium.com/@elijah_williams_agc/how-to-make-a-website-for-your-restaurant-a-complete-guide-c93b9b93ac74.

6 User Management in 2024 and Beyond: A Complete Guide [Електронний ресурс] // Frontegg. – 2024. – Режим доступу до ресурсу: <https://frontegg.com/guides/user-management>.

7 Creating multi user role based admin using PHP Mysql and bootstrap [Електронний ресурс] // Medium. – 2021. – Режим доступу до ресурсу: <https://webscodex.medium.com/creating-multi-user-role-based-admin-using-php-mysql-and-bootstrap-dbebf2740411>.

8 How to create a Food ordering website [Електронний ресурс] // CleverRoad. – 2023. – Режим доступу до ресурсу: <https://www.cleveroad.com/blog/how-to-make-a-food-ordering-website/>.

9 What it takes to create a successful restaurant website [Электронный ресурс] // restarantify. – 2023. – Режим доступа до ресурсу: <https://restaurantify.com/what-it-takes-to-create-a-successful-restaurant-website-budget-guide/>.

10 Building a website with React.js: Everything you should know [Электронный ресурс] // Coax. – 2024. – Режим доступа до ресурсу: <https://coaxsoft.com/blog/building-a-website-with-react-js>.

11 How to create Header in React JS ? [Электронный ресурс] // Geeksforgeeks. – 2023. – Режим доступа до ресурсу: <https://www.geeksforgeeks.org/how-to-create-header-in-react-js/>.

12 Creating a Responsive Sticky Footer in React. [Электронный ресурс] // DopeBase. – 2023. – Режим доступа до ресурсу: <https://dopebase.com/creating-responsive-sticky-footer-react>.

13 How to Create Pages and Set Up Routes in a React App [Электронный ресурс] // CBT nuggets. – 2023. – Режим доступа до ресурсу: <https://www.cbtnuggets.com/blog/technology/programming/how-to-create-pages-and-set-up-routes-in-a-react-app>.

14 Blog app using ReactJS [Электронный ресурс] // Geeksforgeeks. – 2023. – Режим доступа до ресурсу: <https://www.geeksforgeeks.org/blog-app-using-reactjs/>.

15 How to Develop User Registration Form in React JS ? [Электронный ресурс] // Geeksforgeeks. – 2024. – Режим доступа до ресурсу: <https://www.geeksforgeeks.org/how-to-develop-user-registration-form-in-reactjs/>.

16 React 30-Project 10: Building User Authentication Forms using React.js and React Router [Электронный ресурс] // Medium. – 2023. – Режим доступа до ресурсу: <https://saurabhnativeblog.medium.com/react-30-project-10-building-user-authentication-forms-using-react-js-and-react-router-1e3f2efa0c3a>.

17 Demystifying Backend for Frontend Architecture: Building Robust and Scalable Web Applications [Электронный ресурс] // Medium. – 2023. – Режим доступа до ресурсу: <https://mobilelive.medium.com/demystifying-backend-for-frontend-architecture-building-robust-and-scalable-web-applications-22c169a81918>.

18 Exploring Backend and Frontend Development in Technical Architecture [Электронный ресурс] // MoldStud. – 2024. – Режим доступа до ресурсу: <https://moldstud.com/articles/p-exploring-backend-and-frontend-development-in-technical-architecture>.

19 FastAPI — Really Fast and Modern [Электронный ресурс] // Medium. – 2024. – Режим доступа до ресурсу: <https://medium.com/codenx/fastapi-really-fast-and-modern-7a1796597b60>.

20 How to avoid redundancy through the use of Layout Components in React [Электронный ресурс] // Chisom Nhenna Chukwuma. – 2023. – Режим доступа до ресурсу: <https://devzibah.hashnode.dev/how-to-avoid-redundancy-through-the-use-of-layout-components-in-react>.

21 Website footer design — 10 BEST FOOTER EXAMPLES FOR INSPIRATION [Электронный ресурс] // Halo Lab. – 2023. – Режим доступа до ресурсу: <https://www.halo-lab.com/blog/website-footer-design-examples>.

22 How to create a restaurant website: the complete guide [Электронный ресурс] // WIXBlog. – 2023. – Режим доступа до ресурсу: <https://www.wix.com/blog/how-to-create-restaurant-website#viewer-fe0mt>.

23 How to Build a Shopping Cart with React and TypeScript [Электронный ресурс] // freecodecamp. – 2023. – Режим доступа до ресурсу: <https://www.freecodecamp.org/news/how-to-build-a-shopping-cart-with-react-and-typescript/>.

24 How To Create An Advanced Shopping Cart With React and TypeScript [Электронный ресурс] // YouTube | Web dev Simplified. – 2023. – Режим доступа до ресурсу: https://youtu.be/lATafp15HWA?si=_pYCSBIQU4xfPXW9m.

25 How to create a website with an online payment system? [Электронный ресурс] // UPTECH. – 2024. – Режим доступа до ресурсу: <https://www.uptech.team/blog/how-to-create-online-payment-website>.

26 Create API for a database using FastApi [Електронний ресурс] // Medium. – 2023. – Режим доступу до ресурсу: <https://medium.com/@Lijitha/connect-to-the-database-using-fastapi-443878f6a87e>.

27 How to Create Forms in React using react-hook-form [Електронний ресурс] // freecodecamp. – 2022. – Режим доступу до ресурсу: <https://www.freecodecamp.org/news/how-to-create-forms-in-react-using-react-hook-form/>.

28 How To Make Sign In & Sign Up Form Using React JS | ReactJS Login & Registration Form [Електронний ресурс] // YouTube | GreatStack. – 2023. – Режим доступу до ресурсу: <https://youtu.be/8QgQKRcAUvM?si=d5wgXdNnZSWzeExV>.

29 How to Create a Registration Form using React.js and Connect it to Firebase [Електронний ресурс] // WEBSCALE. – 2022. – Режим доступу до ресурсу: <https://www.webscale.com/engineering-education/how-to-create-a-registration-form-using-react-js-and-connect-it-to-firebase/>.

30 Стандарт розробки React для початківців [Електронний ресурс] // DOU. – 2023. – Режим доступу до ресурсу: <https://dou.ua/forums/topic/44659/>.

31 React Architecture: How to Structure and Organize a React Application [Електронний ресурс] // Taniarascia. – 2021. – Режим доступу до ресурсу: <https://www.taniarascia.com/react-architecture-directory-structure/>.

32 Developing a Single Page App with FastAPI and React [Електронний ресурс] // testdriven.io. – 2022. – Режим доступу до ресурсу: <https://testdriven.io/blog/fastapi-react/>.

33 Желібо Є. П. Безпека Життєдіяльності / Є. П. Желібо, Н. М. Заверуха, В. В. Задарний. – Київ: Каравела, 2008. – 344 с. – (Вища освіта в Україні; кн. 6).

34 Кобзін В. Г. Основи Охорони Праці [Електронний ресурс] / В. Г. Кобзін, О. В. Северинов // ХНЕУ. – 2012. – Режим доступу до ресурсу: http://www.repository.hneu.edu.ua/jspui/bitstream/123456789/7542/1/Кобзин_1%20%28Конспект%20лекцій%29%20ООП.pdf.

35 ДСТУ 7950:2015 Дизайн і ергономіка. Робоче місце під час виконання робіт стоячи. Загальні ергономічні вимоги [Текст]. - Чинний від 2016-09-01. - Київ : УкрНДНЦ, 2016. - III, 7 с. : рис. - (Національний стандарт України).

36 V. Kozlovskiy, Y. Balanyuk, H. Martyniuk, O. Nazarevych, L. Scherbak and G. Shymchuk, «Information Technology for Estimating City Gas Consumption During the Year,» 2022 International Conference on Smart Information Systems and Technologies (SIST), Nur-Sultan, Kazakhstan, 2022, pp. 1-4, doi: 10.1109/SIST54437.2022.9945786.

37 Approach to gas consumption process forecasting on the basis of a mathematical model in the form of a random cyclic process / Serhii Lupenko, Iaroslav Lytvynenko, Oleg Nazarevych, Grigorii Shymchuk, Volodymyr Hotovych // ICAAEIT 2021, 15-17 December 2021. – Tern. : TNTU, Zhytomyr «Publishing house „Book-Druk“» LLC, 2021. – P. 213–219. – (Mathematical modeling in power engineering and information technologies).

38 Lytvynenko, S. Lupenko, O. Nazarevych, G. Shymchuk and V. Hotovych, «Mathematical model of gas consumption process in the form of cyclic random process,» 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), LVIV, Ukraine, 2021, pp. 232-235, doi: 10.1109/CSIT52700.2021.9648621.

39 Additive mathematical model of gas consumption process / Iaroslav Lytvynenko, Serhii Lupenko, Oleh Nazarevych, Hryhorii Shymchuk, Volodymyr Hotovych // Scientific Journal of TNTU. – Tern. : TNTU, 2021. – Vol 104. – No 4. – P. 87–97.

40 O. Nazarevych, Y. Leshchyshyn, S. Lupenko, V. Hotovych, G. Shymchuk and N. Shabliy, «Method of Gas Consumption Change-point Detection Based on Seasonally Multicomponent Model,» 2020 10th International Conference on Advanced Computer Information Technologies (ACIT), Deggendorf, Germany, 2020, pp. 152-155, doi: 10.1109/ACIT49673.2020.9208924.

41 Y. Leshchyshyn, L. Scherbak, O. Nazarevych, V. Gotovych, P. Tymkiv and G. Shymchuk, «Multicomponent Model of the Heart Rate Variability Change-point,»

2019 IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH), Polyana, Ukraine, 2019, pp. 110-113, doi: 10.1109/MEMSTECH.2019.8817379.

ДОДАТКИ

Програмний код індексного файлу веб-сайту “Гулівер”

```
import React from 'react';
import { BrowserRouter as Router, Route, Routes } from 'react-
router-dom';
import Header from './components/Header';
import Footer from './components/Footer';
import MainPage from './components/MainPage';
import ContactPage from './components/ContactPage';
import TeamPage from './components/TeamPage';
import MenuPage from './components/MenuPage';
import CartPage from './components/CartPage';
import AddDish from './components/AddDish';
import Login from './components/Login';
import Register from './components/Register';
import AuthProvider from './AuthContext';
import CartProvider from './CartContext';
function App() {
  return (
    <AuthProvider>
      <CartProvider>
        <Router>
          <div className="App">
            <Header />
            <div className="content">
              <Routes>
                <Route exact path="/"
element={<MainPage />} />
                <Route path="/contacts"
element={<ContactPage />} />
                <Route path="/team"
element={<TeamPage />} />
                <Route path="/menu"
element={<MenuPage />} />
                <Route path="/cart"
element={<CartPage />} />
                <Route path="/add-dish"
element={<AddDish />} />
                <Route path="/login"
element={<Login />} />
                <Route path="/register"
element={<Register />} />
              </Routes>
            </div>
            <Footer />
          </div>
        </Router>
      </CartProvider>
    </AuthProvider>)}
export default App;
```

Програмний код стилів для головного компонента веб-сайту “Гулівер”

```
.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}
```

Програмний код контексту аутентифікації для веб-сайту “Гулівер”

```
import React, { createContext, useState } from 'react';

export const AuthContext = createContext();

const AuthProvider = ({ children }) => {
  const [token, setToken] =
    useState(localStorage.getItem('token'));

  const logout = () => {
    setToken(null);
    localStorage.removeItem('token');
  };

  return (
    <AuthContext.Provider value={{ token, setToken, logout }}>
      {children}
    </AuthContext.Provider>
  );
};

export default AuthProvider;
```

Програмний код контексту кошика для веб-сайту “Гулівер”

```
import React, { createContext, useState, useEffect } from 'react';

export const CartContext = createContext();

const CartProvider = ({ children }) => {
  const [cart, setCart] = useState(() => {
    const savedCart = localStorage.getItem('cart');
    return savedCart ? JSON.parse(savedCart) : [];
  });

  useEffect(() => {
    localStorage.setItem('cart', JSON.stringify(cart));
  }, [cart]);

  const addToCart = (dish) => {
    setCart([...cart, dish]);
  };

  const removeFromCart = (dishId) => {
    setCart(cart.filter(dish => dish._id !== dishId));
  };

  return (
    <CartContext.Provider value={{ cart, addToCart,
removeFromCart }}>
      {children}
    </CartContext.Provider>
  );
};

export default CartProvider;
```


Програмний код форми для додавання страви на веб-сайт “Гулівер”

```
import React, { useState, useContext } from 'react';
import axios from 'axios';
import { AuthContext } from '../AuthContext';
const AddDish = () => {
  const [price, setPrice] = useState('');
  const [description, setDescription] = useState('');
  const { token } = useContext(AuthContext);

  const handleSubmit = async (e) => {
    e.preventDefault();
    const newDish = { name, category, price, description };
    await axios.post('http://localhost:5000/api/dishes',
newDish, {
      headers: { Authorization: `Bearer ${token}` },
    });
    setName('');
    setCategory('');
    setPrice('');
    setDescription('');
  };
  return (
    <form onSubmit={handleSubmit}>
      <div>
        <label>Назва:</label>
        <input type="text" value={name} onChange={(e) =>
setName(e.target.value)} />
      </div>
      <div>
        <label>Категорія:</label>
        <input type="text" value={category} onChange={(e)
=> setCategory(e.target.value)} />
      </div>
      <div>
        <label>Ціна:</label>
        <input type="number" value={price} onChange={(e)
=> setPrice(e.target.value)} />
      </div>
      <div>
        <label>Опис:</label>
        <textarea value={description} onChange={(e) =>
setDescription(e.target.value)}></textarea>
      </div>
      <button type="submit">Додати страву</button>
    </form>
  );
};
export default AddDish;
```

Програмний код сторінки кошика для веб-сайту “Гулівер”

```
import React, { useContext } from 'react';
import { CartContext } from '../CartContext';

const CartPage = () => {
  const { cart, removeFromCart } = useContext(CartContext);

  return (
    <div>
      <h1>Кошик</h1>
      {cart.map(dish => (
        <div key={dish._id} className="cart-item">
          <h3>{dish.name}</h3>
          <p>Ціна: {dish.price} грн</p>
          <button onClick={() =>
removeFromCart(dish._id)}>Видалити</button>
          </div>
        )))}
      <button>Оформити замовлення</button>
    </div>
  );
}

export default CartPage;
```

Програмний код футера для веб-сайту “Гулівер”

```
import React from 'react';
import './Footer.css';

const Footer = () => {
  return (
    <footer className="footer">
      <div className="footer-section">
        <h3>Гулівер</h3>
        <p><a href="https://www.facebook.com" target="_blank"
rel="noopener noreferrer">Facebook</a></p>
        <p><a href="https://www.instagram.com" target="_blank"
rel="noopener noreferrer">Instagram</a></p>
        <p><a href="https://www.twitter.com" target="_blank"
rel="noopener noreferrer">Twitter</a></p>
      </div>
      <div className="footer-section">
        <h3>Про нас</h3>
        <p><a href="/about">Співпраця з нами</a></p>
      </div>
      <div className="footer-section">
        <h3>Правила сайту</h3>
        <p><a href="/rules">Контакти</a></p>
      </div>
    </footer>
  );
}

export default Footer;
```

**Програмний код файл стилів для верхнього колонтитулу веб-сайту
“Гулівер”**

```
.header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 10px 20px;
  background-color: #fff;
  border-bottom: 1px solid #ccc;
}

.logo img {
  height: 50px;
}

.navigation {
  flex-grow: 1;
  display: flex;
  justify-content: center;
}

.navigation ul {
  list-style: none;
  display: flex;
  gap: 20px;
  margin: 0;
  padding: 0;
  align-items: center;
}

.navigation li {
  position: relative;
}

.navigation a {
  text-decoration: none;
  color: #000;
  font-weight: bold;
  padding: 10px 0;
}

.navigation a:hover {
  color: #007BFF;
}

.menu-button {
  background-color: #fff;
  border: none;
}
```

```
    cursor: pointer;
    padding: 10px 0;
    font-size: 16px;
    font-weight: bold;
}

.dropdown {
    display: none;
    flex-direction: column;
    position: absolute;
    top: 100%;
    left: 0;
    background-color: #fff;
    border: 1px solid #ccc;
    padding: 10px;
    z-index: 1;
    white-space: nowrap;
}

.menu:hover .dropdown {
    display: flex;
}

.dropdown a {
    text-decoration: none;
    color: #000;
    padding: 5px 10px;
}

.dropdown a:hover {
    background-color: #f0f0f0;
}
```

Програмний код компоненту header для веб-сайту “Гулівер”

```

import React, { useContext } from 'react';
import { Link } from 'react-router-dom';
import './Header.css';
import logo from '../assets/Screenshot_5.jpg';
import Menu from './Menu';
import { AuthContext } from '../AuthContext';

const Header = () => {
  const { token, logout } = useContext(AuthContext);

  return (
    <header className="header">
      <div className="logo">
        <Link to="/">
          <img src={logo} alt="Logo" />
        </Link>
      </div>
      <nav className="navigation">
        <ul>
          <li><Link to="/reservation">Забронювати
столик</Link></li>
          <li><Menu /></li>
          <li><Link to="/blog">Блог від шефа</Link></li>
          <li><Link to="/team">Команда</Link></li>
          <li><Link to="/contacts">Контакти</Link></li>
          <li><Link to="/cart">Кошик</Link></li>
          {token ? (
            <>
              <li><Link to="/add-dish">Додати
страву</Link></li>
              <li><button
onClick={logout}>Вийти</button></li>
            </>
          ) : (
            <>
              <li><Link
to="/login">Увійти</Link></li>
              <li><Link
to="/register">Реєстрація</Link></li>
            </>
          )}
        </ul>
      </nav>
    </header>
  );
};

```

Програмний код компоненту сторінки меню для веб-сайту “Гулівер”

Лістинг Н.1 – Код файлу “Menu.php”

```
export default Header;
import React, { useEffect, useState, useContext } from 'react';
import axios from 'axios';
import { CartContext } from '../CartContext';

const MenuPage = () => {
  const [dishes, setDishes] = useState([]);
  const { addToCart } = useContext(CartContext);

  useEffect(() => {
    const fetchDishes = async () => {
      const response = await
axios.get('http://localhost:5000/api/dishes');
      setDishes(response.data);
    };
    fetchDishes();
  }, []);

  const renderDishesByCategory = (category) => {
    return dishes
      .filter(dish => dish.category === category)
      .map(dish => (
        <div key={dish._id} className="menu-item">
          <h3>{dish.name}</h3>
          <p>{dish.description}</p>
          <p>Ціна: {dish.price} грн</p>
          <button onClick={() => addToCart(dish)}>Додати
до кошика</button>
        </div>
      ));
  };

  return (
    <div>
      <h1>Меню</h1>
      <div id="special-dishes" className="menu-section">
        <h2>Спеціальні страви від шефа</h2>
        {renderDishesByCategory('special')}
      </div>
      <div id="breakfast" className="menu-section">
        <h2>Сніданок</h2>
        {renderDishesByCategory('breakfast')}
      </div>
      <div id="main-dishes" className="menu-section">
        <h2>Основні страви</h2>
        {renderDishesByCategory('main')}
      </div>
    </div>
  );
};
```

```

        </div>
        <div id="asian-cuisine" className="menu-section">
          <h2>Азійська кухня</h2>
          {renderDishesByCategory('asian')}
        </div>
        <div id="fast-food" className="menu-section">
          <h2>Кухня швидкого приготування</h2>
          {renderDishesByCategory('fast')}
        </div>
      </div>
    );
  }

export default MenuPage;

```

Лістинг Н.2 – Код файлу “Registration.php”

```

import React, { useState } from 'react';
import axios from 'axios';

const Register = () => {
  const [username, setUsername] = useState('');
  const [password, setPassword] = useState('');
  const [role, setRole] = useState('');

  const handleSubmit = async (e) => {
    e.preventDefault();
    await axios.post('http://localhost:5000/api/register', {
username, password, role });
    setUsername('');
    setPassword('');
    setRole('');
  };

  return (
    <form onSubmit={handleSubmit}>
      <div>
        <label>Username:</label>
        <input type="text" value={username} onChange={(e)
=> setUsername(e.target.value)} />
      </div>
      <div>
        <label>Password:</label>
        <input type="password" value={password}
onChange={(e) => setPassword(e.target.value)} />
      </div>
      <div>
        <label>Role:</label>
        <select value={role} onChange={(e) =>
setRole(e.target.value)}>
          <option value="">Select role</option>
          <option value="admin">Admin</option>

```



```

        <option value="chef">Chef</option>
    </select>
</div>
<button type="submit">Register</button>
</form>
);
};

export default Register;

```

Лістинг Н.3 – Код файлу “Menu.css”

```

.menu {
    position: relative;
    display: inline-block;
}

.menu-button {
    background-color: #fff;
    border: none;
    cursor: pointer;
    padding: 10px 20px;
    font-size: 16px;
    font-weight: bold;
}

.dropdown {
    display: none;
    flex-direction: column;
    position: absolute;
    top: 100%;
    left: 0;
    background-color: #fff;
    border: 1px solid #ccc;
    padding: 10px;
    z-index: 1;
    white-space: nowrap;
}

.menu:hover .dropdown, .menu .menu-button:focus + .dropdown, .menu
.menu-button:active + .dropdown {
    display: flex;
}

.dropdown a {
    text-decoration: none;
    color: #000;
    padding: 5px 10px;
}

.dropdown a:hover {
    background-color: #f0f0f0;
}

```