

Національний лісотехнічний університет України

(повна найменування вищої навчальної установи)

Навчально-науковий інститут комп'ютерних наук та інформаційних технологій

(повна найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук

(повна назва кафедри (предметної, циклової комісії))

## Пояснювальна записка

до дипломної роботи

перший (бакалаврський)

(рівень вищої освіти)

на тему: Розроблення інформаційної системи для презентації та продажу франшизи мережі Pizza Celentano

Виконала: студентка 4 курсу групи КН-41  
спеціальності

122 «Комп'ютерні науки»

(шифр і назва напрямку підготовки, спеціальності)

Кириченко Р.Н.

(прізвище та ініціали)

Керівники Габа О.О., Дендюк М.В.

(прізвище та ініціали)

Рецензент Карашевський В.П.

(прізвище та ініціали)

Львів – 2025

ІНІ комп'ютерних наук та інформаційних технологій

Кафедра комп'ютерних наук

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 "Комп'ютерні науки"

(штамп і підпис)

**ЗАТВЕРДЖУЮ**

Завідувачка кафедри ІНІ

Борщак І. Б.

"10" червня 2025 року

## ЗАВДАННЯ НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Кириченко Роксолія Назарівна

(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення інформаційної системи для презентації та продажу франшизи мережі Pizza Selentano

керівник роботи Габа О.О., асистент кафедри комп'ютерних наук, Дендюк М.В., кандидат технічних наук, доцент кафедри комп'ютерних наук

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від 15 листопада 2024 року №С-882

2. Термін подання студентом роботи 10 червня 2025р.

3. Вихідні дані до роботи Постановка завдання та його формалізація. Алгоритм побудови інформаційних систем. Вихідні дані та прототипи схожих систем. Графічне представлення вхідних та вихідних даних. Література за тематикою роботи.

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

Вступ,

Розділ 1. Стан проблемної області,

Розділ 2. Інформаційне та математичне забезпечення,

Розділ 3. Програмне та технічне забезпечення,

Висновки,

Список використаних джерел,

Додатки.

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

Слайди для доповіді (підготовка матеріалів для доповіді загальним обсягом 10-12 слайдів)

6. Дата видачі завдання 18 листопада 2024р.

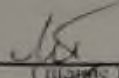
## КАЛЕНДАРНИЙ ПЛАН

Етап	Етапи бакалаврської дипломної роботи	Термін виконання етапів роботи	Примітка
1.	1. Огляд сучасного стану задачі, методів і засобів її вирішення. 2. Формування функціональних вимог та постановка задачі проекту. 3. Оформлення першого розділу пояснювальної записки.	20.11.2024р. 23.12.2024р.	виконано
2.	1. Огляд інформаційного забезпечення. 2. Огляд існуючих систем та модулів. 3. Оформлення другого розділу пояснювальної записки.	08.01.2025р. 15.03.2025р.	виконано
3.	1. Вибір програмних засобів для реалізації інформаційної системи. 2. Програмна реалізація вебзастосунку. 3. Оформлення третього розділу пояснювальної записки.	17.03.2025р. 05.06.2025р.	виконано
4.	1. Здача пояснювальної записки на кафедрі.	10.06.25	виконано

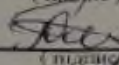
**Студент**

**Керівник проекту (роботи)**

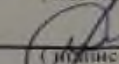
**Керівник проекту (роботи)**

  
(підпис)

**Кириченко Р.Н.**  
(прізвище та ініціали)

  
(підпис)

**Габа О.О.**  
(прізвище та ініціали)

  
(підпис)

**Дендюк М.В.**  
(прізвище та ініціали)

## ТЕХНІЧНЕ ЗАВДАННЯ

### *Розроблення інформаційної системи для презентації та продажу франшизи мережі Pizza Celentano*

1. Провести аналіз сучасних підходів до презентації та продажу франшиз, визначити переваги та недоліки платформи WordPress з використанням конструктора Beaver Builder.
2. Встановити та налаштувати CMS WordPress із застосуванням конструктора Beaver Builder та плагіна Advanced Custom Fields (ACF).
3. Розробити структуру та навігацію сайту, створити дизайн інтерфейсу з урахуванням сучасних UX/UI підходів, орієнтуючись на існуючий корпоративний стиль Pizza Celentano.
4. Реалізувати інтерактивну карту ресторанів з використанням Leaflet.js та ACF для додавання, редагування та відображення інформації про локації з можливістю фільтрації та пошуку за містами і адресами.
5. Реалізувати контактну форму для збору заявок від потенційних франчайзі з інтеграцією Google reCAPTCHA для захисту від спаму та забезпечити автоматичну відправку заявок на електронну пошту менеджерів.
6. Створити адаптивну верстку сайту, яка коректно відображається на мобільних пристроях, планшетах та десктопах.
7. Провести комплексне тестування сайту, зокрема:
  - роботу інтерактивної карти;
  - роботу контактних форм та захисту від спаму;
  - адаптивність та швидкість завантаження на різних пристроях.
8. Провести SEO-оптимізацію за допомогою плагіна SEO, забезпечити коректну генерацію метаданих та XML-карти сайту.

## АНОТАЦІЯ

Пояснювальна записка до дипломної роботи бакалавра містить 58 сторінок, 33 рисунки, 2 додатки, 18 джерел.

У дипломній роботі розроблено адаптивну інформаційну систему для презентації та продажу франшизи мережі Pizza Celentano на базі CMS WordPress із використанням конструктора Beaver Builder та плагіна Advanced Custom Fields. Проведено аналіз існуючих підходів до презентації франшиз, визначено переваги та недоліки платформи WordPress із Beaver Builder. Встановлено та налаштовано WordPress із інтеграцією Beaver Builder, ACF, Leaflet.js для інтерактивної карти ресторанів, Contact Form 7 із захистом Google reCAPTCHA для збору заявок. Розроблено структуру сайту, UX/UI-дизайн, реалізовано адаптивну верстку для мобільних пристроїв. Проведено тестування функціональності карти, форм заявок, перевірку адаптивності та швидкості завантаження.

**Ключові слова:** CMS, WordPress, Beaver Builder, Advanced Custom Fields, Leaflet.js, Contact Form 7, Google reCAPTCHA, адаптивна верстка, UX/UI.

## ABSTRACT

The explanatory note to the bachelor's thesis contains 58 pages, 33 figures, 2 appendices, and 18 sources.

In the thesis, an adaptive information system for presenting and selling the Pizza Celentano franchise was developed based on CMS WordPress, using the Beaver Builder page builder and the Advanced Custom Fields plugin. An analysis of existing approaches to franchise presentation was carried out, and the advantages and disadvantages of the WordPress and Beaver Builder platform were identified. WordPress was installed and configured with integration of Beaver Builder, ACF, Leaflet.js for an interactive restaurant map, and Contact Form 7 with Google reCAPTCHA for lead collection. The site structure and UX/UI design were developed, and a responsive layout for mobile devices was implemented. Functional testing of the map, forms, responsiveness, and loading performance was conducted.

**Keywords:** CMS, WordPress, Beaver Builder, Advanced Custom Fields, Leaflet.js, Contact Form 7, Google reCAPTCHA, responsive layout, UX/UI.

## ЗМІСТ

<b>ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧНЬ .....</b>	<b>7</b>
<b>ВСТУП.....</b>	<b>8</b>
<b>1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ .....</b>	<b>9</b>
<b>1.1. Аналіз підходів до презентації франшиз.....</b>	<b>9</b>
<b>1.2. Технологічні платформи для реалізації презентаційних сайтів.....</b>	<b>11</b>
<b>1.3. Вимоги до сучасних франшизних інформаційних систем .....</b>	<b>13</b>
<b>2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ .....</b>	<b>15</b>
<b>2.1. Загальна структура інформаційного забезпечення.....</b>	<b>15</b>
<b>2.2. Платформа WordPress та інструменти для побудови контенту.....</b>	<b>16</b>
<b>2.3. Інтеграція з зовнішніми сервісами та забезпечення безпеки.....</b>	<b>18</b>
<b>2.4. Технічні вимоги до хостингу та інфраструктури.....</b>	<b>19</b>
<b>3. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ.....</b>	<b>21</b>
<b>3.1. Розроблення та налаштування інформаційної системи.....</b>	<b>21</b>
<b>3.2. Створення інтерактивної карти ресторанів мережі.....</b>	<b>27</b>
<b>3.3. Заголовок та підвал інформаційної системи .....</b>	<b>34</b>
<b>3.3. Інтеграція контактної форми .....</b>	<b>36</b>
<b>3.4. Базова SEO оптимізація .....</b>	<b>40</b>
<b>3.5. Адаптивність інформаційної системи .....</b>	<b>43</b>
<b>ВИСНОВОК.....</b>	<b>46</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....</b>	<b>47</b>
<b>ДОДАТОК А .....</b>	<b>49</b>
<b>ДОДАТОК Б .....</b>	<b>52</b>

## **ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧНЬ**

WP – WordPress.

CMS – система управління вмістом.

ІС – інформаційна система.

ПЗ – програмне забезпечення.

БД – база даних.

ОС – операційна система.

ПК – персональний комп'ютер.

ІТ – інформаційні технології.

PHP – скриптова мова програмування

JS – динамічна, об'єктно-орієнтована прототипна мова програмування

HTML – мова розмітки гіпертексту

CSS – каскадні таблиці стилів.

## ВСТУП

*Актуальність теми* дипломної роботи полягає у тому, що сучасне бізнес-середовище стає дедалі більш конкурентним та динамічним: швидко зростає кількість франчайзингових пропозицій, особливо у сегменті громадського харчування, а потенційні партнери, обираючи мережу для інвестицій, дедалі частіше приймають рішення на основі якості онлайн-презентації. За даними досліджень, більше 70 % інвесторів ознайомлюються з умовами франшизи через вебсайт компанії, перш ніж зв'язатися з менеджером або відвідати офіс франчайзера. У такому контексті забезпечення швидкого та інтуїтивно зрозумілого доступу до всіх ключових матеріалів — від опису бізнес-моделі до деталей фінансових розрахунків — стає вирішальним фактором, що впливає на рівень конверсії зацікавлених осіб у реальних партнерів.

*Об'єкт дослідження:* процеси цифрової презентації комерційної пропозиції франшизи.

*Предмет дослідження:* методи розробки інформаційної системи для веб-презентації бізнес-моделі франшизи.

*Мета роботи* - розробити інформаційну систему для франшизи мережі Pizza Celentano на базі WordPress з використанням конструктора Beaver Builder, який дозволяє потенційним франчайзі ознайомитись із перевагами, умовами, етапами співпраці та заповнити заявку на приєднання до мережі.

*Практичне значення* роботи полягає у створенні повноцінного вебресурсу, який сприятиме розширенню партнерської мережі Pizza Celentano, автоматизує первинні етапи взаємодії з потенційними франчайзі та дозволить надалі масштабувати систему на інші регіони й проекти франчайзера.

# 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

## 1.1. Аналіз підходів до презентації франшиз

Франчайзинг є однією з найбільш динамічних бізнес-моделей сучасності, що дозволяє компаніям швидко масштабуватись із мінімальними витратами на власне розширення мережі, а підприємцям — розпочати бізнес із використанням перевіреної, відпрацьованої бізнес-моделі. В Україні за останні роки спостерігається суттєве зростання популярності франчайзингових пропозицій, особливо в таких сегментах ринку як громадське харчування (ресторани, кафе, фаст-фуд), роздрібна торгівля (ритейл) та різноманітні послуги (салони краси, медичні центри, освітні установи) [1, 2].

Успіх франчайзингової мережі безпосередньо залежить від якості презентації пропозиції франшизи потенційним франчайзі. Ефективна презентація франшизи передбачає прозорість усіх умов співпраці, чітке і зрозуміле викладення переваг та потенційних ризиків, а також якісне візуальне оформлення інформації. Особливої уваги при цьому заслуговує саме вебпрезентація, оскільки переважна більшість потенційних франчайзі в першу чергу знайомиться з умовами франшизи саме через онлайн-канали [3].

Серед сучасних вебрішень, які активно застосовують компанії для презентації своїх франшизних пропозицій, виділяють такі формати:

- **Презентаційні лендінги (односторінкові сайти).** Це найбільш простий та розповсюджений спосіб швидко і зрозуміло донести ключову інформацію про франшизу. Лендінги, як правило, включають коротку інформацію про переваги, розмір інвестицій, умови співпраці та можливості розвитку бізнесу в межах франчайзингової мережі.
- **Інтерактивні вебплатформи.** Такі сайти характеризуються розширеним набором функцій, серед яких: інтерактивна карта з географією вже відкритих точок, калькулятори інвестицій, покрокові інструкції з відкриття франчайзингового бізнесу, а також онлайн-форми для збору заявок з автоматичним перенаправленням їх до CRM-систем франчайзера [4].

Згідно з аналізом сучасних тенденцій, найбільш ефективними елементами вебпрезентацій франшиз є:

- **Інтерактивна карта існуючих франчайзингових точок.** Така карта демонструє потенційним франчайзі реальні масштаби мережі та надає можливість ознайомитися з локаціями, що підсилює довіру до бренду та заохочує нових партнерів.
- **Візуалізація фінансових результатів та успішних кейсів.** Це можуть бути графіки, діаграми, інфографіка, а також детально описані приклади успішних запусків бізнесу діючими франчайзі. Використання таких елементів дозволяє потенційним франчайзі краще уявити фінансовий потенціал і зменшує психологічні бар'єри перед інвестицією.
- **Покроковий опис процесу відкриття франшизи.** Надзвичайно важливо показати майбутнім партнерам чітку структуру і послідовність дій, які будуть здійснюватися від першого контакту до відкриття бізнесу. Це значно спрощує прийняття рішення і робить пропозицію більш зрозумілою.
- **Автоматизовані форми заявок.** Використання форм із автоматичною інтеграцією в CRM дозволяє максимально швидко реагувати на інтерес потенційного партнера і проводити оперативну роботу з лідами, підвищуючи конверсію зацікавлених у франчайзі.

Важливим аспектом при створенні вебпрезентації франшизи є професійний, продуманий дизайн, який відповідає корпоративному стилю бренду. Якісний дизайн сприяє формуванню довіри до франчайзера, позитивно впливає на сприйняття бренду потенційними франчайзі. Не менш значущими є технічні характеристики сайту, такі як швидкість завантаження сторінок, адаптивність та мобільна версія сайту, підтримка локалізації для різних регіонів та аудиторій [5].

Окрему увагу у сфері презентації франшизи необхідно приділяти розвитку омніканального підходу до роботи з потенційними клієнтами. Омніканальні стратегії включають інтеграцію вебресурсів із додатковими комунікаційними каналами, такими як email-розсилки, чат-боти, соціальні мережі та CRM-системи. Це забезпечує комплексний підхід до роботи з потенційними франчайзі та дозволяє персоналізувати

спілкування, що значно підвищує ефективність комунікацій та якість роботи з заявками потенційних партнерів.

Важливими додатковими інструментами, які здатні суттєво підвищити ефективність презентації франшизи, є:

- **Відеопрезентації успішних кейсів франчайзі.**
- **Відгуки партнерів та клієнтів, інтегровані з соціальними мережами.**
- **Статистика та аналітика прибутковості мережі.**

Таким чином, комплексне використання сучасних підходів до вебпрезентації франшизи дозволяє підвищити якість комунікації з потенційними франчайзі, сформувати довіру до бренду та значно знизити бар'єри для прийняття позитивного рішення про інвестицію у франчайзингову пропозицію.

## **1.2. Технологічні платформи для реалізації презентаційних сайтів**

Для створення ефективного презентаційного ресурсу франшизи важливим етапом є вибір технологічної платформи. Такий вибір визначає подальші можливості розвитку, масштабування, легкість підтримки та швидкість впровадження змін. На сьогодні на ринку існує значна кількість систем управління контентом (CMS), серед яких найбільш популярними та перевіреними є WordPress, Joomla, Drupal, OpenCart та Shopify [6, 7].

Серед зазначених платформ, саме **WordPress** вирізняється найбільшою популярністю і поширеністю для розробки сайтів малого та середнього бізнесу. За останніми статистичними даними, понад 40% усіх сайтів в інтернеті реалізовано саме на базі цієї CMS. Причинами такої популярності є простий та зрозумілий інтерфейс, гнучкість у налаштуванні, величезна кількість готових рішень (тем, плагінів) та велика спільнота розробників, які постійно підтримують і розвивають платформу [6]. Переваги WordPress полягають також у:

- **Низькому порозі входу:** навіть користувачі без спеціальної технічної підготовки можуть самостійно керувати контентом та базовими налаштуваннями сайту.

- **Широкий екосистемі плагінів:** завдяки використанню різноманітних розширень можна швидко налаштувати необхідні функції, такі як SEO, безпека, форми збору заявок, інтеграція з соціальними мережами та CRM-системами.
- **Великій спільноті:** це дозволяє оперативно знаходити рішення будь-яких технічних завдань та постійно оновлювати систему до сучасних стандартів безпеки й продуктивності [6].

Одним із найпопулярніших візуальних конструкторів для WordPress є **Beaver Builder**, який дозволяє реалізовувати складні сторінки з інтуїтивним drag-and-drop інтерфейсом. Beaver Builder є особливо корисним для створення лендингів та інтерактивних сторінок, оскільки дозволяє точно контролювати структуру блоків, типографіку, мультимедійні елементи та анімацію без необхідності ручного написання HTML або CSS-коду. Це значно спрощує процес розробки та дозволяє оперативно вносити зміни відповідно до потреб маркетингових кампаній або нових умов франшизи [7].

Крім того, WordPress має вбудований **REST API**, що забезпечує інтеграцію з іншими сервісами і дозволяє автоматизувати збір та обробку заявок, інтегрувати систему з CRM-системами, сервісами аналітики, поштовими розсилками та чат-ботами [8].

Зокрема, при розробці платформи для презентації франшизи Pizza Celentano були використані такі додаткові технології та інструменти:

- **ACF (Advanced Custom Fields)** — дозволяє створювати кастомні типи контенту, такі як інтерактивні карти з ресторанами, блоки з перевагами франшизи, відгуки та фінансові показники.
- **Google Maps API** — використовується для створення інтерактивної карти з відображенням мережі ресторанів, підтримкою фільтрів за регіонами чи містами.
- **SMTP-плагіни** — забезпечують стабільну та надійну роботу контактних форм та автоматичну відправку листів на email.
- **Плагіни кешування та безпеки (WP Super Cache, Wordfence)** — забезпечують оптимальну швидкість роботи сайту та захист від злому.

- **SEO-плагіни (All in One SEO)** — дозволяють налаштувати ефективну оптимізацію контенту для пошукових систем.
- **Форми збору заявок (Contact Form 7)** — дозволяють створювати інтерактивні форми з інтеграцією у CRM-системи та автоматичною обробкою даних.

Цей комплекс технологій забезпечує надійну, ефективну та гнучку систему для презентації франчайзингових пропозицій, дозволяючи оперативно реагувати на бізнес-виклики та змінювати стратегію залежно від поточних умов ринку.

### **1.3. Вимоги до сучасних франшизних інформаційних систем**

Сучасний вебресурс, орієнтований на презентацію франшизи, має відповідати не лише інформаційним, але й маркетинговим та технічним вимогам, які допомагають ефективно залучати потенційних партнерів і конвертувати їх у реальних клієнтів. Для цього використовуються спеціалізовані підходи до UX/UI-дизайну, які враховують поведінку користувачів та сприяють комфортному використанню сайту з максимальною кількістю цільових дій (перегляд інформації, подання заявки, контактування з менеджерами) [13].

Серед основних вимог до франшизних вебресурсів слід виділити такі аспекти:

- **Зручність і простота інтерфейсу (UX/UI-дизайн):** сайт має бути інтуїтивно зрозумілим, щоб користувач міг легко знайти інформацію про умови франшизи, переваги, фінансові показники, умови співпраці. Чітко структурована інформація допомагає зменшити кількість відмов та підвищити конверсію відвідувачів у потенційних партнерів.
- **Мобільна адаптивність:** значна частка користувачів сьогодні використовує смартфони або планшети для перегляду сайтів, тому адаптивний дизайн є критично важливим фактором. Інтерфейс має залишатися зручним і функціональним на будь-якому пристрої та розмірі екрана [14].
- **Швидкість завантаження:** оптимізація швидкості завантаження сайту безпосередньо впливає на конверсію. Вебресурс повинен миттєво завантажувати контент і стабільно працювати навіть при великій кількості

одночасних запитів, особливо під час рекламних кампаній або сезонних пікових навантажень [14].

- **Безпека даних та відповідність законодавству:** важливим аспектом є захист персональних даних користувачів. Вебресурс має бути оснащений SSL-сертифікатом, що забезпечує шифрування даних, а також має підтримувати засоби захисту форм від ботів (наприклад, reCAPTCHA). Крім того, необхідна чітка і зрозуміла політика конфіденційності, яка відповідає українському та європейському законодавству [15].

Отже, сучасний франшизний вебресурс повинен враховувати бізнес-цілі, зручність користувачів, швидкість роботи, безпеку та законодавчі вимоги, що забезпечить максимальну ефективність взаємодії з потенційними франчайзі і підвищить кількість успішних звернень [14, 15].

## 2. ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

### 2.1. Загальна структура інформаційного забезпечення

Інформаційне забезпечення вебресурсу охоплює повний життєвий цикл даних: створення, зберігання, обробку, передачу зовнішнім системам та відображення користувачам. Контент формується в адміністративній панелі WordPress, зберігається в базі даних MySQL, передається до CRM і аналітичних систем, а потім виводиться на сторінках сайту з використанням кастомних полів і візуальних конструкторів [1, 2].

Текстові матеріали (опис франшизи, бізнес-модель, умови співпраці, фінансові показники) зберігаються у записах Pages та Posts і доповнюються кастомними полями ACF (Text, Textarea). Це дає можливість маркетологам вносити зміни в контент без залучення розробників [3]. Динамічні кейси та відгуки клієнтів організуються за допомогою Repeater-полів ACF і відображаються в модулях Beaver Builder, що забезпечує єдину стилістику та швидке додавання нових записів [4]. Мультимедіа-матеріали (фотогалереї, відеоогляди) зберігаються в медіатеці WordPress і виводяться адаптивно через медіа-модулі Beaver Builder [5].

Геопросторові дані для карти зберігаються в полях ACF типу Google Map та Repeater, які включають координати, адреси та категорії закладів. Ці дані у форматі JSON підвантажуються JavaScript-фреймворком Leaflet.js, що дозволяє динамічно рендерити маркери, вмикати кластеризацію та виконувати фільтрацію за регіонами без додаткових серверних запитів [6, 7].

Користувацький ввід реалізовано через Contact Form 7 із підтримкою AJAX-режиму. Форми захищені Google reCAPTCHA v3, яка аналізує поведінкові метрики для блокування бот-атак без зайвих кроків для користувача [8]. Підписка на розсилку інтегрована з MailChimp API, що дозволяє автоматично додавати контакти до списків розсилки та відстежувати показники відкриття листів.

Управління даними здійснюється через адміністративну панель з ролями (адміністратор, редактор, автор). Двофакторна аутентифікація (2FA) через Wordfence або Authy забезпечує додатковий рівень захисту при вході [9]. Резервне копіювання бази даних та файлів виконується щодня за допомогою UpdraftPlus із синхронізацією

на зовнішні сховища (Dropbox, Google Drive) для швидкого відновлення після збоїв [10].

## 2.2. Платформа WordPress та інструменти для побудови контенту

WordPress є відкритою системою управління контентом з модульною архітектурою, яка з 2003 року, відколи її створили Метт Мулленвег та Майк Літл, перетворилася на провідну CMS для вебпроектів будь-якого рівня складності. Ключовими факторами успіху WordPress стали:

- **Відкритий код та спільнота.** Велика кількість розробників та дизайнерів по всьому світу підтримують ядро та створюють тисячі плагінів і тем, що дозволяє регулярно оновлювати систему та закривати вразливості [1, 3].
- **Гнучкість та розширюваність.** Архітектура плагінів дозволяє інтегрувати додаткові функції: від SEO-оптимізації до складних електронних комерційних модулів, без зміни ядра платформи [3].
- **Зручна адмінпанель.** Інтуїтивний інтерфейс для управління контентом (Pages, Posts, Media) спрощує роботу маркетологів та редакторів, знижуючи витрати на навчання персоналу.

**Beaver Builder** (2014) — це конструктор, який пропонує:

1. **Drag-and-drop редактор:** створення макету сторінки шляхом перетягування модулів, з миттєвим попереднім переглядом результату.
2. **Модульна система:** понад 30 готових модулів для додавання тексту, заголовків, зображень, галерей, відео, кнопок, контактних форм та анімацій.
3. **Шаблони сторінок та секцій:** зберігання та повторне використання унікальних дизайнів, що пришвидшує розробку нових сторінок.
4. **Інтеграція з Beaver Themer:** можливість створювати динамічні шаблони для кастомних типів записів (Custom Post Types) та полів ACF, автоматично підставляючи дані з бази в потрібні елементи дизайну [4].

**Advanced Custom Fields (ACF)** (2015) забезпечує:

- **Групи полів:** створення логічно згрупованих полів для різних типів контенту (фінансові показники, перелік переваг, кейси партнерів).

- **Типи полів:** містить більше 30 варіантів, серед яких Text, Textarea, Image, File, Repeater, Flexible Content та Google Map для геоданих.
- **Repeater та Flexible Content:** дозволяють створювати повторювані блоки та складні макети, де кожен елемент може містити різні поля, що забезпечує максимальну гнучкість контенту [4].
- **API для розробників:** розширення функцій через власні фільтри та хуки, що спрощує інтеграцію з іншими плагінами та темами.

**All in One SEO (AIOSEO)** — популярний плагін для оптимізації під пошукові системи. Його основні можливості:

- **Генерація метаданих:** автоматичне створення title, description і Open Graph тегів.
- **Аналіз ключових слів:** вбудований інструмент підказок та оцінки щільності ключових слів.
- **XML карта сайту:** автоматичне формування карти сайту та підтримка RSS-карт для блогів [5].

**WP Mail SMTP** — плагін, що забезпечує надійну доставку пошти, налаштовуючи SMTP-з'єднання через:

- **Gmail API** чи **Mailgun** для обходу обмежень хостинга та запобігання потраплянню листів у папку "Спам".

**W3 Total Cache** та **WP Rocket** — інструменти для кешування, які:

- **HTML, CSS, JS кешування** на стороні сервера та браузера.
- **Мінімізація та об'єднання файлів** для зменшення кількості запитів.
- **Лінгвістичне відстрочене завантаження (lazy load)** для зображень та відео, що скорочує початковий час відображення сторінки [10].

**Wordfence** та **Sucuri** відповідають за безпеку:

- **Сканування на шкідливий код** та відстеження ненормальної активності.
- **Брандмауер веб-додатків (WAF)** для захисту від SQL-ін'єкцій, XSS та брутфорс-атак.

Використання WordPress разом із зазначеними плагінами забезпечує створення потужного, гнучкого та безпечного ресурсу, де контент-менеджери можуть

оперативно оновлювати інформацію без залучення розробників, а користувачі отримують стабільний та швидкий доступ до даних.

### **2.3. Інтеграція з зовнішніми сервісами та забезпечення безпеки**

Однією із ключових задач інформаційного забезпечення є налагодження надійної взаємодії інформаційної системи з зовнішніми сервісами та одночасне забезпечення високого рівня безпеки даних. Для реалізації інтеграцій застосовуються відкриті API та стандартизовані протоколи обміну, а системи захисту — багаторівневі підходи до виявлення і запобігання загрозам.

По-перше, для відображення мережі ресторанів використано Leaflet.js у комбінації з OpenStreetMap. Геопросторові дані зберігаються в ACF-полях у вигляді JSON та динамічно підвантажуються на клієнтську частину, де скрипт виконує рендеринг маркерів, їх кластеризацію та фільтрацію за параметрами (місто, формат, доступність) без додаткових запитів до сервера. Такий підхід значно покращує продуктивність і знижує навантаження на бекенд-систему [6, 7].

По-друге, інтеграція форм зворотного зв'язку здійснюється за допомогою Contact Form 7, налаштованого на відправку повідомлень через WP Mail SMTP. Використання зовнішніх SMTP-провайдерів (Gmail API чи Mailgun) гарантує високий рівень доставки листів та зменшує ризик їх потрапляння в спам-папку [8, 5].

Захист інфраструктури ґрунтується на декількох рівнях:

1. **Криптографічне захищене з'єднання.** Весь трафік шифрується за допомогою SSL/TLS сертифікатів (Let's Encrypt), що відповідає сучасним стандартам безпеки веб-комунікацій [9].
2. **Брандмауер веб-додатків (WAF).** Плагіни Wordfence або Sucuri проводять активне сканування запитів, виявляючи та блокуючи спроби SQL-ін'єкцій, XSS та інші типи атак на рівні HTTP [9].
3. **Антиспам та фільтрація.** Google reCAPTCHA v3 інтегровано не лише у форми, а й в механізми AJAX-запитів, що додатково фільтрує підозрілий трафік на рівні клієнтського скрипту [8].

4. **Моніторинг та логування.** Використання зовнішніх сервісів UptimeRobot та централізоване логування через Papertrail дозволяють своєчасно виявляти збої та підозрілу активність, а також проводити аудит подій для подальшого аналізу та реагування [10].

Багаторівневий підхід до інтеграції та безпеки дозволяє забезпечити надійність функціонування ресурсу, захист даних користувачів та бізнес-логіки франчайзингової системи Pizza Celentano.

#### **2.4. Технічні вимоги до хостингу та інфраструктури**

Для забезпечення безперебійного функціонування інформаційної системи франшизи Pizza Celentano критично важливим є вибір правильного хостинг-провайдера та налаштування інфраструктури. Сучасні вимоги включають високі показники доступності, масштабованість та можливості резервування даних.

Перш за все, рекомендується використовувати управлінський WordPress-хостинг від провайдерів із перевіреною репутацією (наприклад, SiteGround, Bluehost або локальні рішення, такі як Loccloud). Такі сервіси зазвичай надають оптимізоване середовище під PHP 8.x та MySQL 8.x, а також підтримку технологій HTTP/2 та OPcache, що прискорює обробку запитів і знижує час відповіді сервера [2, 3].

Достатній обсяг ресурсів сервера (не менш ніж 2 CPU та 4 GB оперативної пам'яті) і високошвидкісні SSD-диски гарантують мінімальну затримку при обробці складних запитів (наприклад, фільтрація даних на карті чи вивантаження великих галерей). Наявність балансувальника навантаження (load balancer) дозволяє масштабувати обчислювальні потужності горизонтально при зростанні трафіку.

Реєстрація домену у зонах .com або .ua з керуванням записами типу A, CNAME та MX забезпечує правильну маршрутизацію трафіку та коректне налаштування поштових сервісів. Важливо впевнитися, що DNS-сервери провайдера підтримують швидке оновлення записів (низький TTL) для гнучкого перенаправлення у разі проблем.[1]

Використання SSL/TLS сертифікатів є обов'язковим для шифрування всього трафіку між клієнтом та сервером. Безкоштовні сертифікати Let's Encrypt з

автоматичним оновленням через ACME-протокол дозволяють заощадити кошти й забезпечити безперервність HTTPS-з'єднання. Альтернативою є придбання комерційних сертифікатів від DigiCert чи GeoTrust, що надають розширену гарантію й довіру браузерів [9].

Для зменшення часу завантаження та оптимізації продуктивності рекомендовано підключити CDN (Content Delivery Network), наприклад, Cloudflare або AWS CloudFront. CDN-кешування статичного контенту (зображень, CSS, JavaScript) знижує навантаження на основний сервер і прискорює доставку контенту користувачам у різних регіонах [7].

Резервування даних здійснюється на двох рівнях:

- Резервні копії баз даних та файлів здійснюються щодня через UpdraftPlus або вбудовані інструменти хостингу.
- Зовнішнє зберігання копій на хмарних сервісах (Dropbox, Google Drive) або в окремому дата-центрі, що забезпечує швидке відновлення після технічних збоїв [10].

Моніторинг працездатності ресурсу рекомендовано проводити за допомогою сервісів UptimeRobot або Pingdom, які сповіщають про будь-які простоя сервера. Для централізованого логування та аналізу подій використовується Papertrail або Loggly, що дозволяє швидко виявляти аномалії й проводити аудит безпеки [10].

### 3. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

#### 3.1. Розроблення та налаштування інформаційної системи

Узгодивши та обговоривши деталі з керівником компанії Pizza Celentano, було прийняте рішення про основні розділи інформаційної системи, які мають бути реалізовані в ході розроблення даного вебсайту. Отже, було вирішено, що наша система повинна такі розділи як: «Переваги», «Умови», «Контакти» та «Карта ресторанів мережі». Також було отримано корпоративні кольори, шрифти, та логотип.

Отже, після встановлення системи управління контентом, яку було обрано для того, щоб полегшити зміни у системі, які будуть відбуватись досить часто у зв'язку з стрімким розвитком компанії. Переходимо до створення головної сторінки інформаційної системи, яка буде наповнюватись відповідним контентом за допомогою плагіну Beaver Builder.

Отже для того, що створити головну сторінку переходимо у вкладку сторінки, далі обираємо додати нову та задаємо заголовок нашої системи та опубліковуємо її, як показано на рис. 3.1.

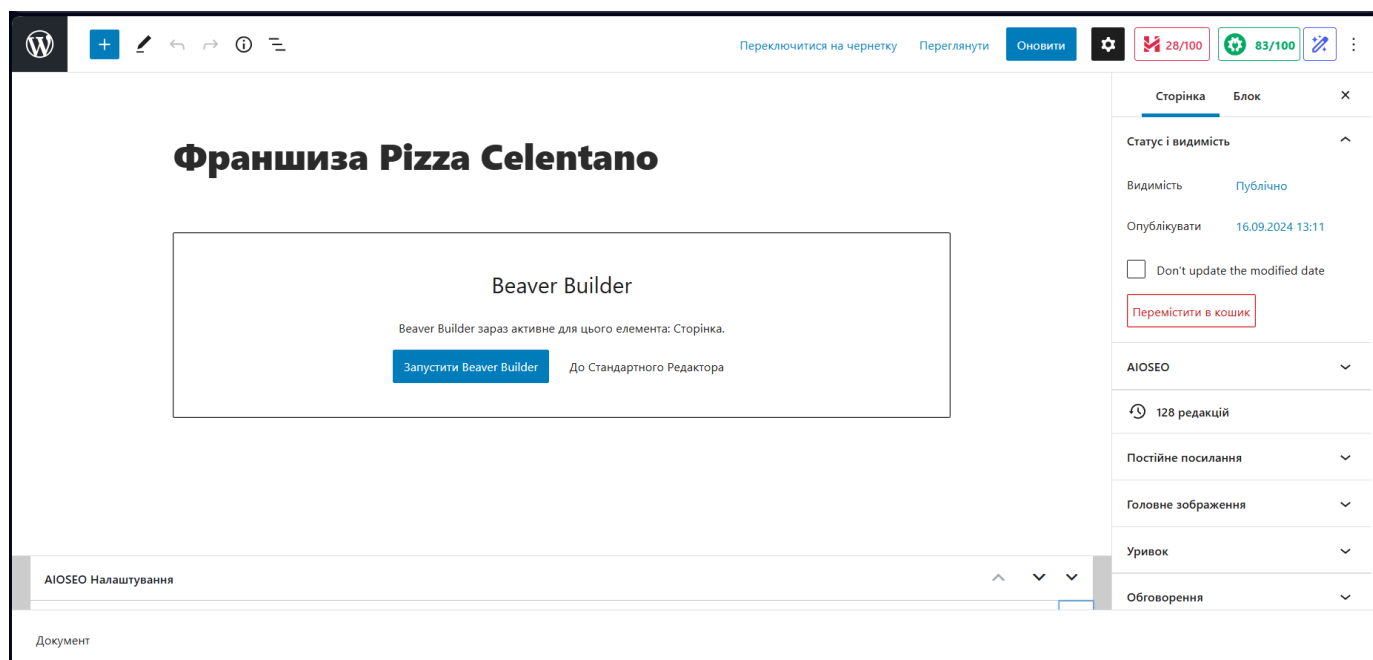


Рисунок 3.1. – Створення головної сторінки

Далі необхідно задати щойно створену сторінку, як головну. Для цього переходимо на вкладку вигляд та обираємо налаштувати, після чого потрапляємо на

сторінку для редагування теми, на якій обираємо вкладку налаштування домашньої сторінки і фінальним кроком обираємо щойно створену сторінку та зберігаємо налаштування, як показано на рис. 3.2.

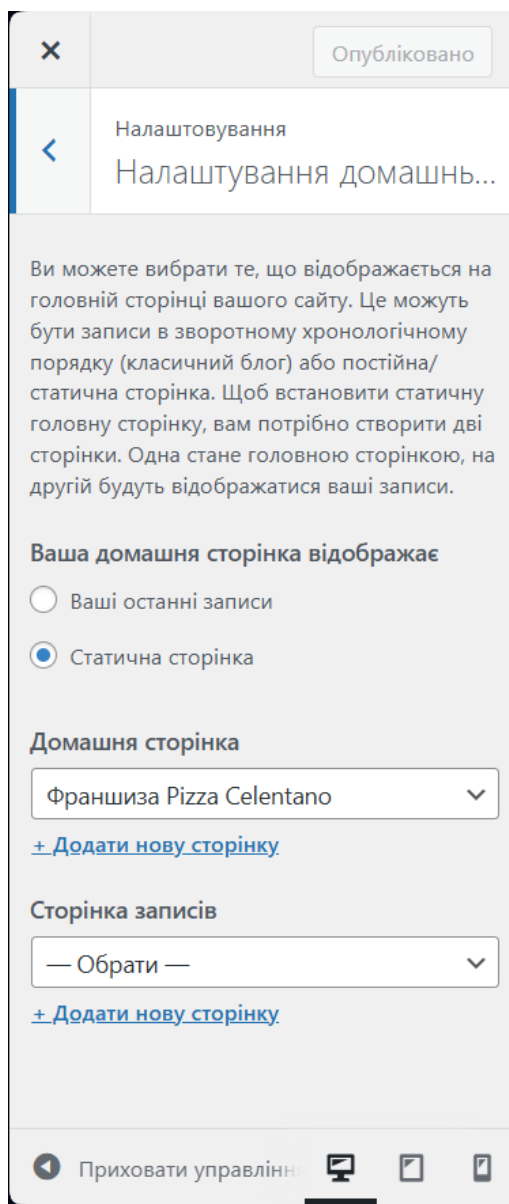


Рисунок 3.2. – Встановлення домашньої сторінки

Wordpress дає можливість користуватись своїм редактором, який дозволяє наповнювати сторінки відповідним вмістом, проте цей редактор не дає гнучкості, та досить обмежений по можливостям, так як першочергово він був створений для потреб ведення блогу. Тому було прийнято рішення встановити плагін, який зможе надати нам цю гнучкість, та дозволить швидше та якісніше додавати, змінювати, редагувати контент на сторінці. І цим плагіном є Beaver Builder.

Для того, щоб його встановити потрібно обрати пункт меню плагіни, натиснути додати новий, та в пошуку знайти наш плагін та встановити його, як показано на рис. 3.3.

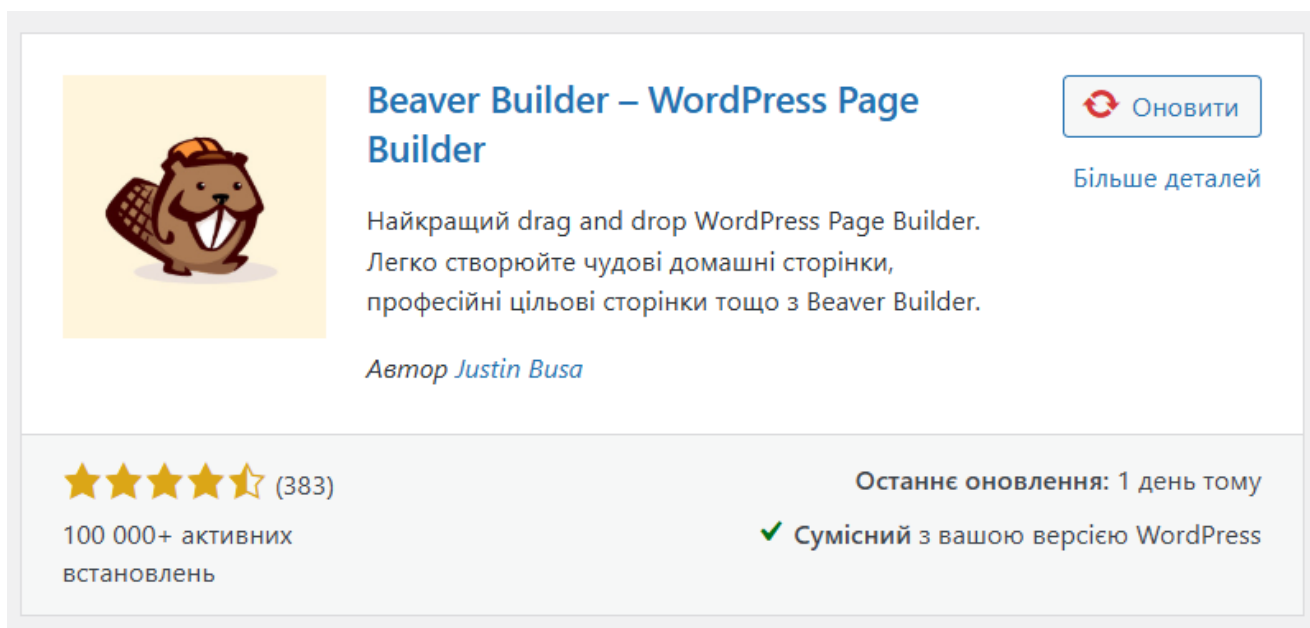


Рисунок 3.3. Встановлення плагіну Beaver Builder.

Наступним кроком переходимо до наповнення нашої головної сторінки за допомогою встановленого плагіну. Для цього відкриваємо головну сторінку та обираємо запустити Beaver Builder, як показано на рис. 3.4.

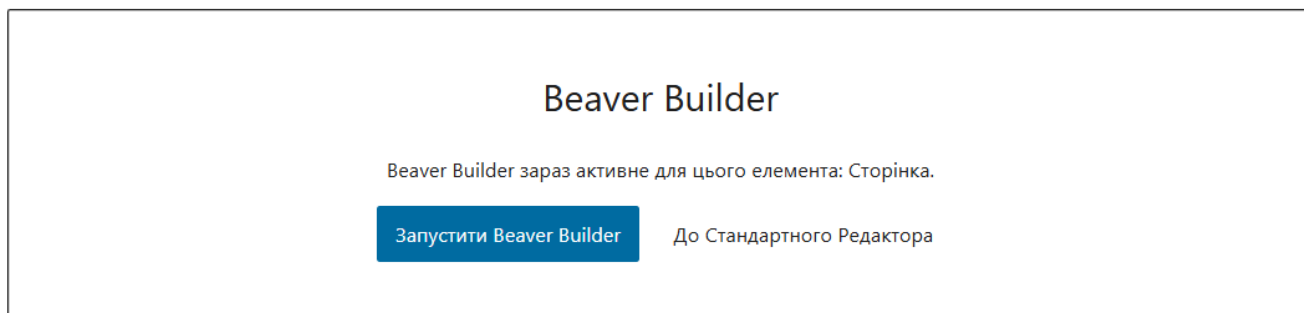


Рисунок 3.4. Запуск плагіну Beaver Builder.

Після цього потрапляємо на сторінку, де відкривається весь функціонал плагіну, для візуального редагування макетів, що дозволяє «перетягувати» елементи напряму на сторінку, забезпечуючи миттєвий перегляд результатів, модульну систему контенту, яка містить понад 30 вбудованих модулів (текст, зображення, відео, кнопки, галереї, форми, акордеони, іконки, лічильники, анімації), які можна поєднувати у довільні секції, можливість управління шаблонами, які зберігають власні секції та сторінки у бібліотеці для

багатократного використання, що пришвидшує розробку нових сторінок та зберігає єдині стилістичні рішення. Забезпечує реактивність та адаптивність, за допомогою вбудованих налаштування для мобільних, планшетних і десктопних версій сторінки, з можливістю точного налаштування вигляду на кожному пристрої, також має розширюваність через API і хуки, які надають можливість додавати власні модулі або підключати сторонні розширення за допомогою PHP-фільтрів та JavaScript-хуків для тонкого налаштування функціоналу та зображений на рис. 3.5.

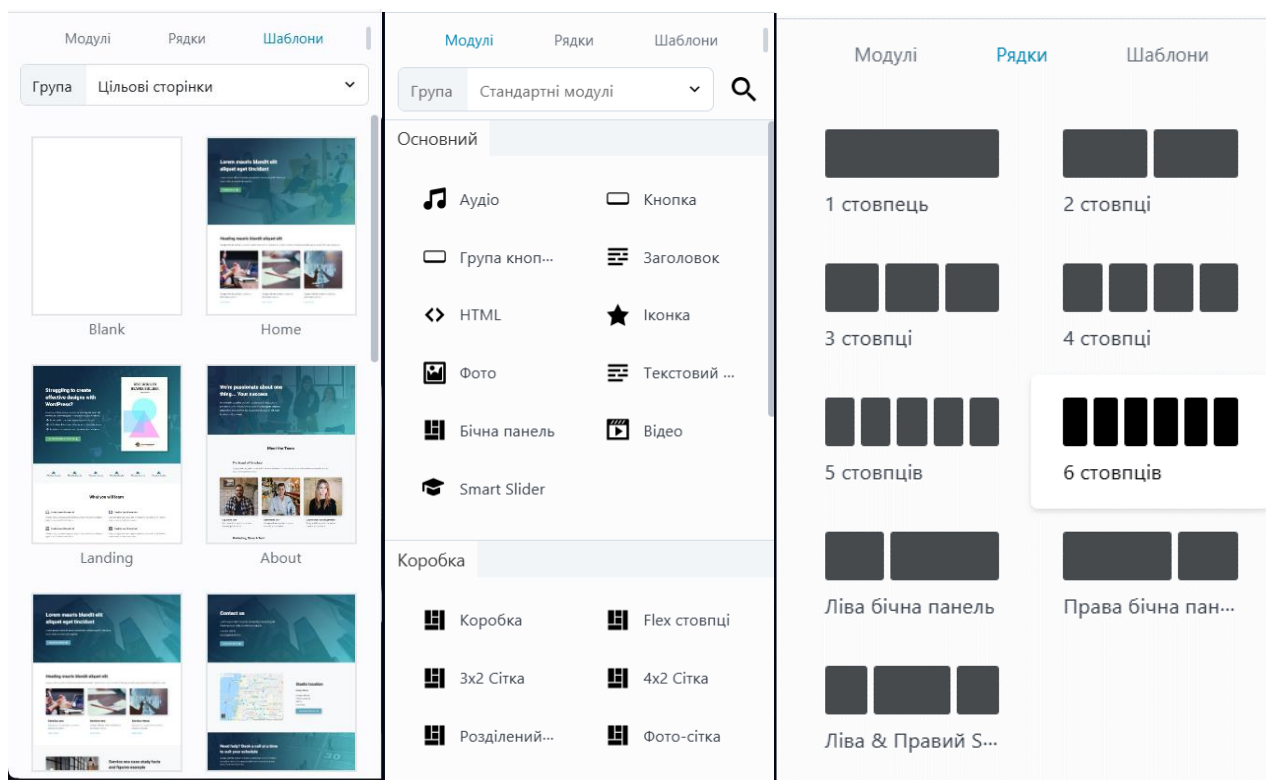


Рисунок 3.5. Функціонал Beaver Builder

Розробивши дизайн головної сторінки інформаційної системи відповідно до брендбуку мережі, та отримавши текстові та графічні матеріали переходимо до наповнення головної сторінки контентом.

Отже для першої секції було додано фонове відео, блоки з текстом та блоки з кнопками для форми зворотного зв'язку та отримання презентації, також було передбачено кнопку для переходу до карти ресторанів мережі, що показано на рис. 3.6:

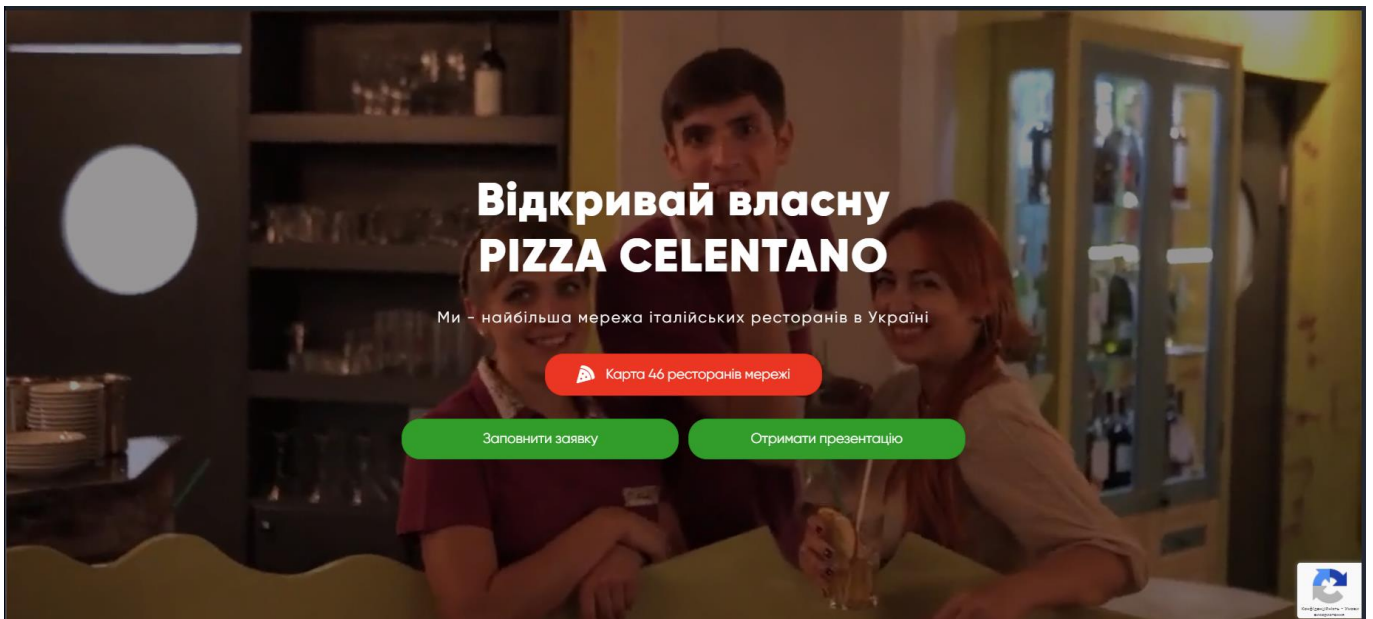


Рисунок 3.6. Перша секція

Наступна секція включає в себе переваги франшизи Pizza Celentano, яку було реалізовано за допомогою рядків плагіну та блоку html, в якому можна задати окремі стилі та скрипти, отримуємо наступну структуру секції та фінальний результат на рис. 3.7:

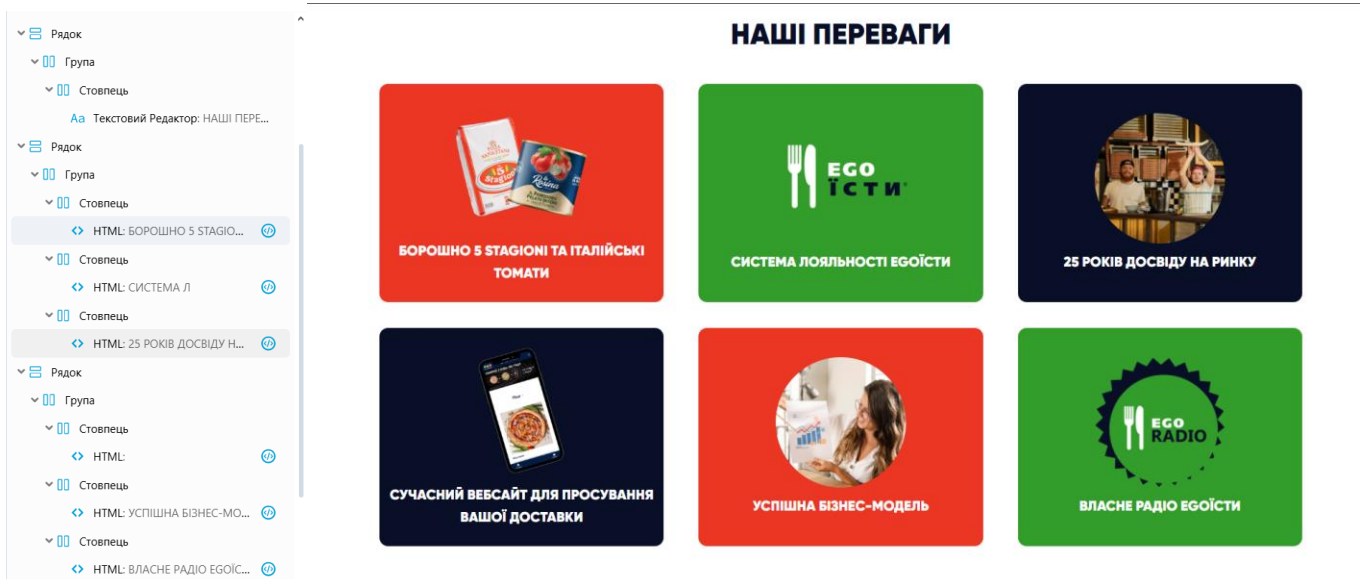


Рисунок 3.7. Структура та результат 2-гої секції

Наступним етапом ми створили адаптивну галерею, яка презентує різні заклади мережі Pizza Celentano. Кожне зображення демонструє потенційним франчайзі не лише багатство дизайнерських рішень і варіативність інтер'єрів, але й єдність стилю, що підкреслює впізнаваність бренду, що показано на рис. 3.8:



Рисунок 3.8. Галерея

Наступний кроком йде секція умови франшизи, яка також була реалізована за допомогою html та css блоку, що показано на рис. 3.9 та отримано результат який зображено на рис 3.10:

```

HTML
    Загальні
    Додатково
    1 <div class="circles">
    2   <div class="circle green">
    3     10-12ТИС$
    4     Паушальний внесок
    5   </div>
    6   <div class="circle blue">
    7     850$/M2
    8     Інвестиції
    9   </div>
    10   <div class="circle red">
    11     3%
    12     Рента
    13   </div>
    14   <div class="info-boxes">
    15     <div class="info-box">
    16       
    17       ЧИСТИЙ ПРИБУТОК
    18       Від 5тис. $
    19     </div>
    20     <div class="info-box">
    21       
    22       ТЕРМІН ОКУПНОСТІ
    23       24+ місяців
    24   </div>
    25 </div>
    26 </div>
    27 </div>
    28 </div>
    29 </div>
    CSS
    1 .circles {
    2   display: flex;
    3   justify-content: center;
    4   align-items: center;
    5   margin-bottom: 40px;
    6 }
    7 .circle {
    8   border-radius: 50%;
    9   display: flex;
    10  justify-content: center;
    11  align-items: center;
    12  color: white;
    13  font-size: 40px;
    14  margin: 0 -20px;
    15  position: relative;
    16 }
    17 .circle.green {
    18   font-size: 35px;
  
```

Рисунок 3.9. HTML блок

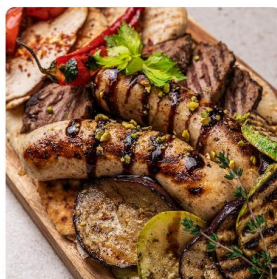


Рисунок 3.10. Секція умови франшизи

Наступним кроком було розроблено секцію для додаткових опцій, які доступні при купівлі франшизи. Дана секція була реалізована за допомогою html - структури, css - стилів та js – інтерактивності. Отримуємо наступний результат показаний на рис. 3.11:

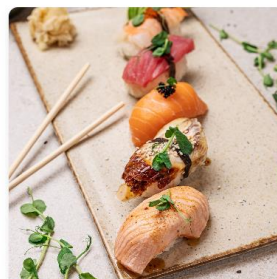
## ДОДАТКОВІ ОПЦІЇ

Франшиза Pizza Celentano – це не лише про піцу. Ми пропонуємо цілий ряд додаткових функцій, які підвищують комфорт та залучення гостей. Онлайн-замовлення, технічне забезпечення заходів, спеціалізовані меню – усе це створює унікальний досвід для кожного відвідувача.



### БЕНКЕТНЕ МЕНЮ

Готове рішення для обслуговування свят, корпоративів і подій. Додатковий прибуток та ефективне використання залу у непіковій годині.



### СУШІ

Суші – популярний формат для доставки та вечірніх замовлень. Розширення аудиторії та конкурентна перевага для вашого закладу.



### ДИТЯЧЕ МЕНЮ

Привертає сімейних гостей і відкриває можливості для дитячих. Стабільний трафік і лояльна аудиторія.



Рисунок 3.11. Секція додаткові опції

Дана секція виконує стратегічну функцію демонстрації повного спектру сервісів і функціональних можливостей мережі. Вона складається з інтерактивного слайдера, у якому поєднано яскраві фото та лаконічні текстові описи, що разом створюють цілісне уявлення про бізнес-моделі закладів франчайзингової мережі.

Таким чином, секція «Додаткові опції» відіграє ключову роль у просуванні франшизи Pizza Celentano, ефективно поєднуючи візуальну привабливість із предметним обґрунтуванням бізнес-вигод і технічної підтримки.

## 3.2. Створення інтерактивної карти ресторанів мережі

Інтерактивна карта реалізована як окремий фронтенд-блок на головній сторінці сайту мережі Pizza Celentano. Її основне призначення — візуалізація географічного розташування ресторанів по Україні для користувачів та потенційних франчайзі, забезпечення легкого пошуку закладів і доступу до ключової інформації.

Карта адаптивна, інтерактивна, містить розширений функціонал: фільтрацію за містами, пошук за адресами, поп-ап з контактами та соціальними мережами, автоматичне масштабування та адаптацію для мобільних пристроїв.

Перш за все за допомогою плагіну Advance Custom Fields було створено групу полів для карти, в якій передбачено два текстових поля місто та адреса, два числових поля для координат, поле для зображення та дві url адреси для посилання на соціальні мережі, що показано на рис. 3.12.

Порядок	Ярлик	Назва	Тип
1	Місто	city	Текст
2	Адреса	address	Текст
3	Координати (широта)	coords_lat	Число
4	Координати (довгота)	coords_lng	Число
5	Фото	image	Зображення
6	Instagram	instagram	Url
7	Facebook	facebook	Url

Рисунок 3.12. Група полів restaurant

Наступним кроком було розроблено функцію `create_restaurants_post_type()`, яка створює кастомний тип запису в WordPress для зберігання даних про ресторани мережі Pizza Selentano. Це дозволяє розділити дані ресторанів від стандартних записів і сторінок, а також працювати з ними окремо в адмін-панелі.

```
Customify: functions.php Оберіть тему для редагування: C  
Вибраний вміст файлу:  
91 }  
92  
93 function create_restaurants_post_type() {  
94     register_post_type('restaurant',  
95         array(  
96             'labels' => array(  
97                 'name' => __('Ресторани', 'textdomain'),  
98                 'singular_name' => __('Ресторан', 'textdomain'),  
99                 'menu_name' => __('Ресторани', 'textdomain'),  
100                'add_new' => __('Додати новий', 'textdomain'),  
101                'add_new_item' => __('Додати новий ресторан', 'textdomain'),  
102                'edit_item' => __('Редагувати ресторан', 'textdomain'),  
103            ),  
104            'public' => true,  
105            'has_archive' => true,  
106            'menu_icon' => 'dashicons-store',  
107            'supports' => array('title', 'editor', 'thumbnail'),  
108            'show_in_rest' => true,  
109        )  
110    );  
111 }  
112 add_action('init', 'create_restaurants_post_type');  
113
```

Рисунок 3.13. Створена функцію create\_restaurants\_post\_type().

Після запуску функції в меню з'являється окрема вкладка Ресторини, на який можна додавати, видаляти та редагувати дані про ресторани, що показано на рис. 3.14:

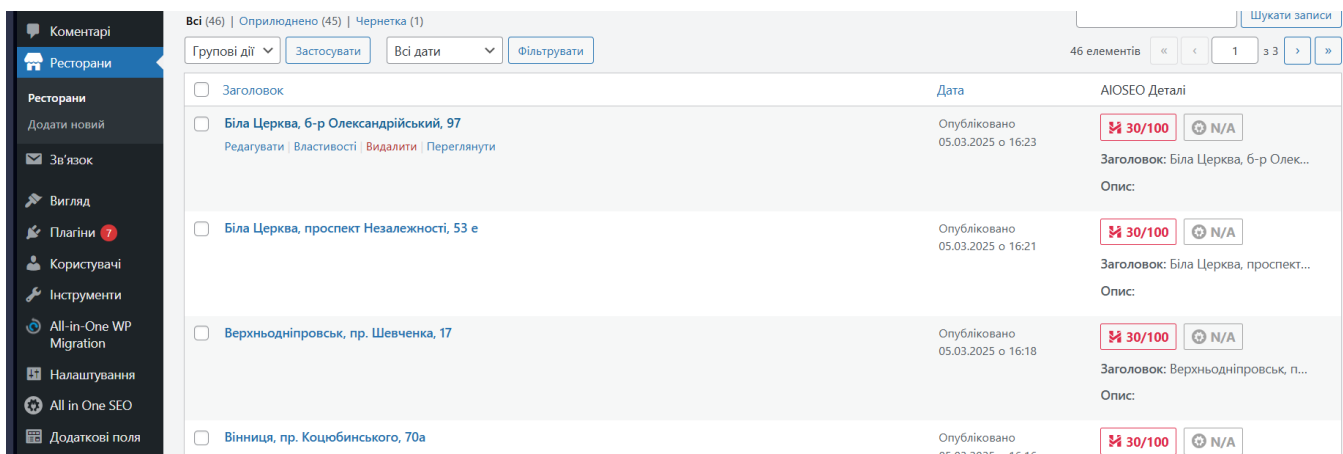


Рисунок 3.14. Вкладка ресторани.

Далі, в ACF зв'язали створені поля та створений тип запису Ресторан, тобто поля будуть відображатись тільки коли тип матеріалу буде Ресторан, як показано на рис. 3.15:

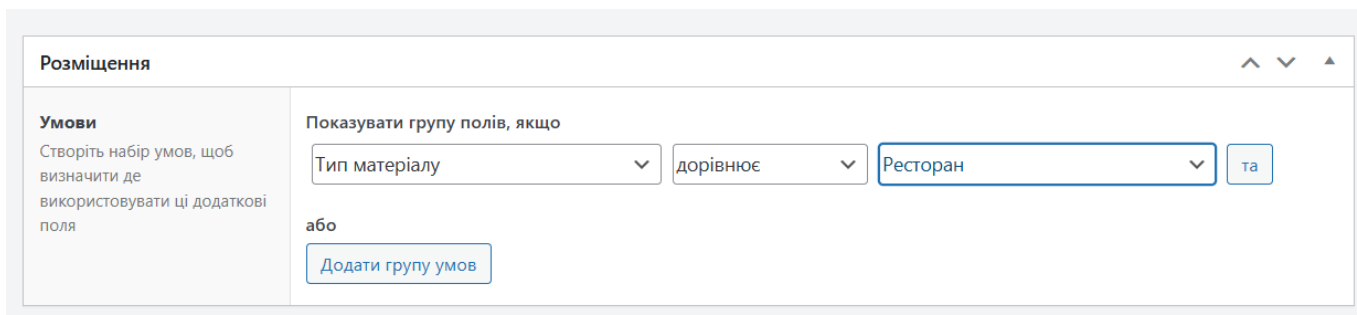


Рисунок 3.15. Зв'язок ACF та типу запису Ресторан.

Після чого на вкладці ресторану і при спробі додати новий ресторан зможемо побачити нашу створену групу полів та заповнити її відповідними даними, як показано на рис. 3.16:


restaurant	^
<b>Місто</b>	
<input type="text" value="Вінниця"/>	
<b>Адреса</b>	
<input type="text" value="вул. М. Литвиненко, 1"/>	
<b>Координати (широта)</b>	
<input type="text" value="49,227751879338875"/>	
<b>Координати (довгота)</b>	
<input type="text" value="28,450265371163674"/>	
	
<b>Instagram</b>	
<input type="text" value=""/>	
<b>Facebook</b>	
<input type="text" value="https://www.facebook.com/pizzacelentanoVinnitsaLitvinenko"/>	


Рисунок 3.16. Заповнення запису Ресторан.

Для реалізації інтерактивної карти на головній сторінці франшизи Pizza Celentano був створений окремий шаблон сторінки в WordPress з назвою Restaurants Map, який показано рис. 3.17. Його мета — зібрати всі дані про ресторани, збережені через адміністративну панель WordPress, та передати їх у JavaScript для відображення маркерів на карті за допомогою Leaflet.js. Для кожного ресторану:

- отримується його назва через `get_the_title($restaurant->ID)`;
- отримуються значення з ACF-полів: місто, адреса, координати широти і довготи;
- також додається посилання на зображення (якщо воно завантажене);
- збираються соціальні мережі (facebook, instagram) та сайт.

Всі ці дані зберігаються в асоціативному масиві `$data[]`, який згодом буде передано у форматі JSON до JavaScript.

Цей шаблон виконує роль мосту між WordPress та Leaflet.js, надаючи структуровану інформацію про всі заклади, які адміністратор вводить у зручному інтерфейсі. Всі дані динамічні, не хардкодяться в коді, що дозволяє гнучко керувати контентом без залучення розробника.



```
7 ));
8
9 $data = [];
10
11 if ($restaurants) {
12     foreach ($restaurants as $restaurant) {
13         $image = get_field('фото_ресторану', $restaurant->ID);
14         $facebook = get_field('facebook', $restaurant->ID);
15         $instagram = get_field('instagram', $restaurant->ID);
16         $website = get_field('website', $restaurant->ID);
17
18         $data[] = [
19             'name' => get_the_title($restaurant->ID),
20             'city' => get_field('місто', $restaurant->ID),
21             'address' => get_field('адреса', $restaurant->ID),
22             'coords_lat' => get_field('координати_широта', $restaurant->ID),
23             'coords_lng' => get_field('координати_довгота', $restaurant->ID),
24             'image' => $image ? $image['url'] : '',
25             'facebook' => $facebook,
26             'instagram' => $instagram,
27             'website' => $website
28         ];
29     }
30 }
31 if ($restaurants) {
32     foreach ($restaurants as $restaurant) {
33         echo "<div class='restaurant'><h3>"; get_the_title($restaurant->ID); "</h3><div class='";
```

Рисунок 3.17. Шаблон сторінки Restaurant Map.

Для забезпечення повної інтерактивності карти було реалізовано фронтенд-функціонал на JavaScript із використанням бібліотеки Leaflet.js, що показано на рис. 3.18. Цей функціонал відповідає за завантаження маркерів ресторанів, керування масштабуванням, фільтрацію за містами, пошук за адресами, а також за зручність використання на мобільних пристроях. При завантаженні сторінки карта створюється без фіксованого центру. За допомогою методу Leaflet `fitBounds()` автоматично масштабується так, щоб показати всі координати ресторанів, якщо такі є. Якщо ж даних немає — задається стандартний центр (м. Львів). Далі для кожного ресторану створюється маркер із координатами. До кожного маркера прикріплюється кастомізований поп-ап із HTML-розміткою, яка включає зображення, адресу, місто, а також іконки соціальних мереж. Усі елементи мають стилізацію через CSS. Реалізовано фільтрації по містах. Міста автоматично збираються із поля ACF `місто`, сортуються по алфавіту та додаються до `<select>`-елемента. Після вибору міста всі



Після формування даних, функція підключає зовнішню бібліотеку Leaflet (CSS та JS) через `wp_enqueue_style()` і `wp_enqueue_script()`. Також підключаються локальні файли `map.css` та `map.js` з теми за допомогою `get_stylesheet_directory_uri()`. До назви файлів додається параметр версії на основі часу останньої модифікації (`filemtime()`), щоб уникнути кешування старої версії. Масив `$data`, який містить усі необхідні поля ресторану, передається у JavaScript через `wp_localize_script()`. У результаті, у фронтенді буде доступна змінна `restaurantsData`, яка використовується в JS для побудови маркерів. HTML-структура виводиться через `ob_start()` / `ob_get_clean()` — це буферизація виводу. Карта складається з контейнера з класом `restaurants-map-container`, який включає блок фільтрів (input пошуку та select для міст) і саму карту з id `map`. Цей HTML повертається як результат шорткоду. За допомогою функції `add_shortcode('restaurants_map', 'restaurants_map_shortcode')` цей шорткод стає доступним у WordPress. Щоб вивести карту, достатньо вставити `[restaurants_map]` у текстовому редакторі будь-якої сторінки чи блоку.

Цей підхід на рис 3.19 дозволяє легко вставити карту з актуальними даними, які редагуються через адмінку WordPress.

```

Customify: functions.php
Оберіть тему для редагування: Custc
Вибраний вміст файлу:
117
118
119
120 $data = [];
121
122 foreach ($restaurants as $restaurant) {
123     $image = get_field('image', $restaurant->ID);
124     $image_url = $image ? $image['url'] : '';
125     $data[] = [
126         'name' => get_the_title($restaurant->ID),
127         'city' => get_field('city', $restaurant->ID),
128         'address' => get_field('address', $restaurant->ID),
129         'coords_lat' => get_field('coords_lat', $restaurant->ID),
130         'coords_lng' => get_field('coords_lng', $restaurant->ID),
131         'image' => $image_url,
132         'facebook' => get_field('facebook', $restaurant->ID),
133         'instagram' => get_field('instagram', $restaurant->ID)
134     ];
135 }
136
137 wp_enqueue_style('leaflet-css', 'https://unpkg.com/leaflet/dist/leaflet.css');
138 wp_enqueue_script('leaflet-js', 'https://unpkg.com/leaflet/dist/leaflet.js', [], null, true);
139
140 // Унікальні версії для уникнення кешу
141 $version = filemtime(get_stylesheet_directory() . '/js/map.js');
142
143 // Підключаємо власні CSS і JS
144 wp_enqueue_style('map-styles', get_stylesheet_directory_uri() . '/css/map.css', [], $version);
145 wp_enqueue_script('map-script', get_stylesheet_directory_uri() . '/js/map.js', ['leaflet-js'], $version, true);
146
147 // Передаємо дані в JS
148 wp_localize_script('map-script', 'restaurantsData', $data);

```

Рис. 3.19. Створення шорткоду.

Після цього вставляємо наш шорткод у відповідну секцію та отримуємо наступний результат, який показано на рис. 3.20:

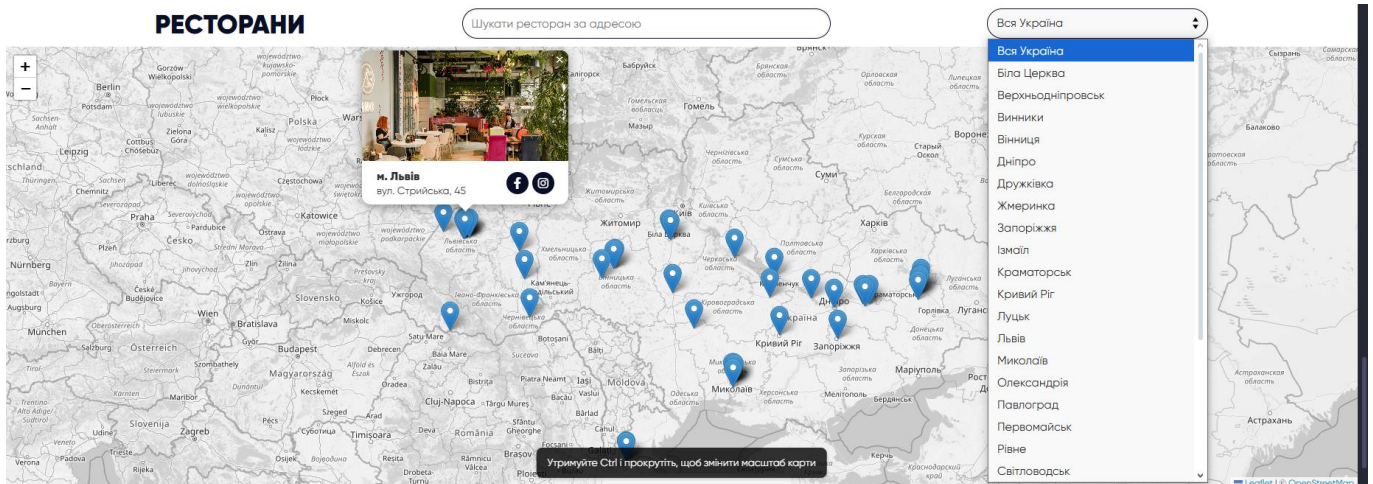


Рисунок 3.20. Відображення карти.

### 3.3. Заголовок та підвал інформаційної системи

Наступним кроком переходимо до налаштування header та footer нашої інформаційної системи франшизи, перши кроком створюємо меню, в якому будуть посиління на секції переваги, умови та контакти. Переходимо на вкладку вигляд та обираємо меню, створюємо з користувацьких посилання на id відповідних розділів, як показано на рис. 3.21.

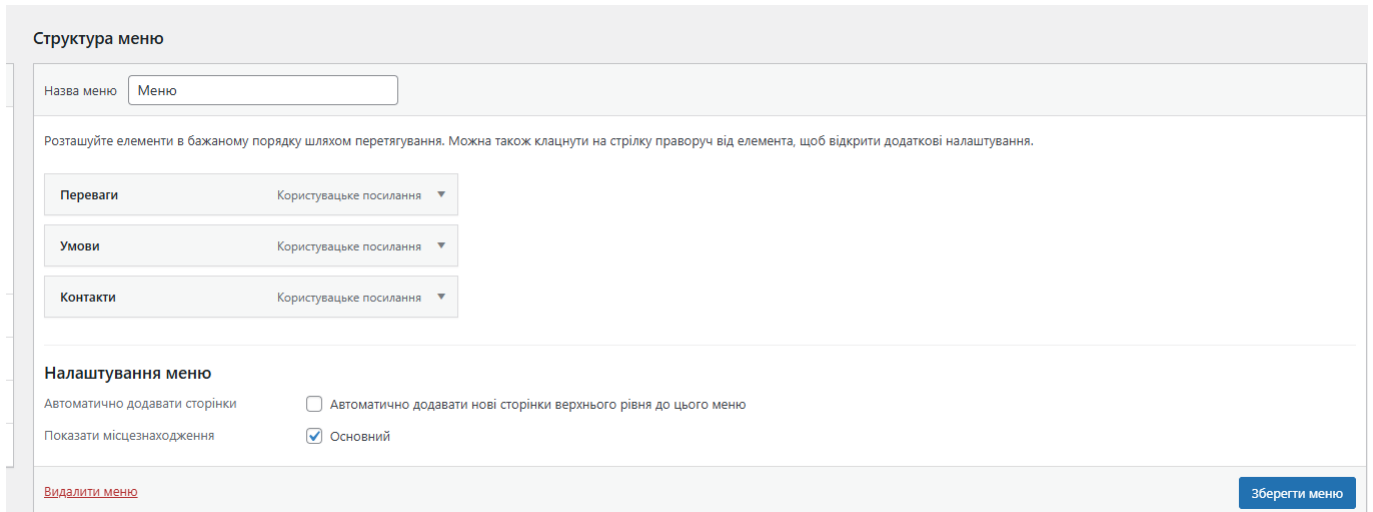


Рисунок 3.21 Пункти меню

Коли створили меню, переходимо до його дизайну та відображення, для цього на вкладці вигляд обираємо налаштувати, і потрапляємо на сторінку налаштувань теми, на якій обираємо пункт хедер. Додаємо основне меню, логотип та робимо адаптивність на цій сторінці, яка показана на рис. 3.22:

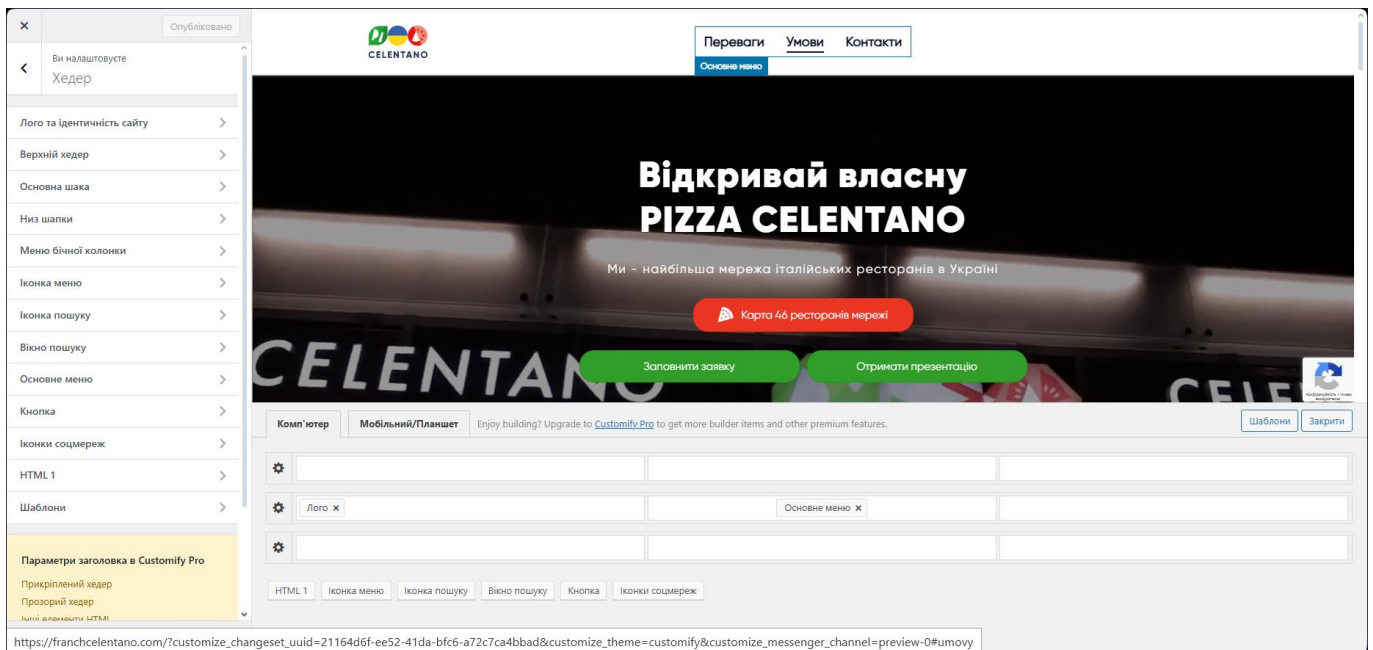


Рисунок 3.22 Конструктор шапки

Збергіємо налаштування та переходимо до налаштувань футеру, додаємо заголовок, телефон, адресу, логотип, та соціальні мережі, як показано на рис. 3.23. Також, на цій сторінці передбачаємо мобільну версію підвалу нашої інформаційної системи.

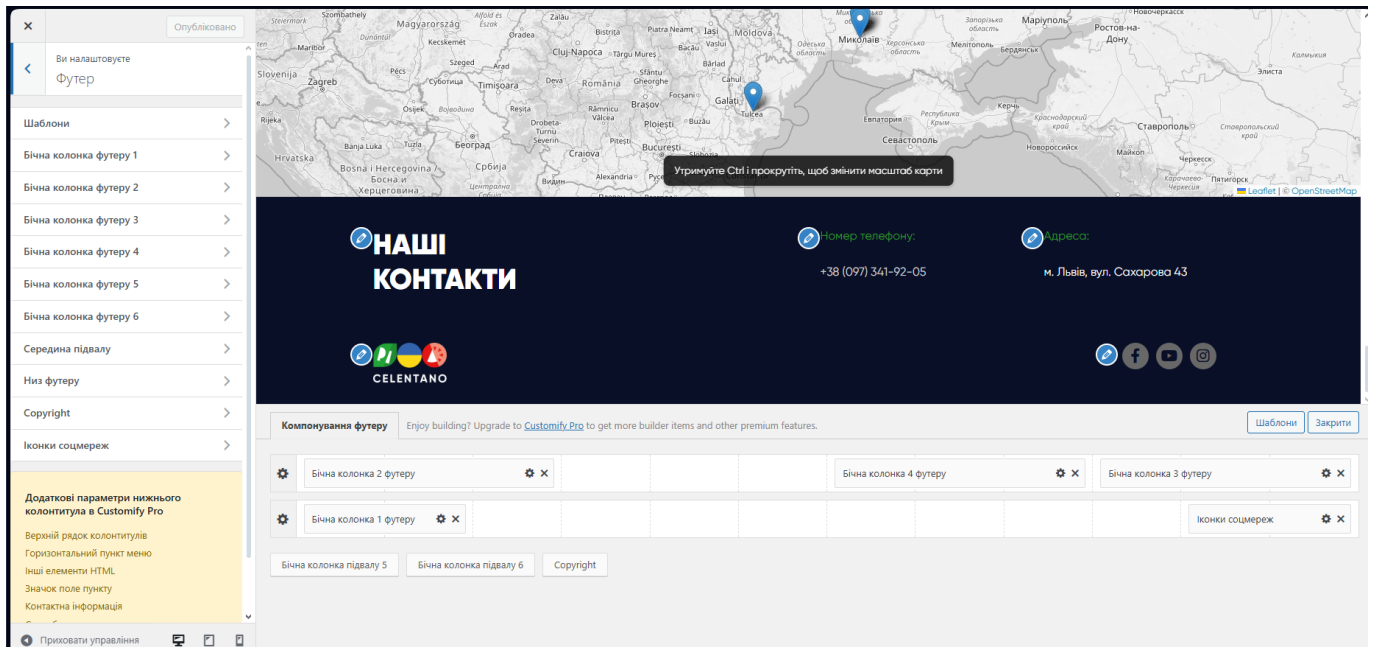


Рисунок 3.23 Конструктор підвалу.

### 3.3. Інтеграція контактної форми

Для забезпечення зворотного зв'язку з користувачами на сайті реалізовано контактну форму за допомогою плагіна Contact Form 7. Цей плагін дозволяє швидко створювати форми зворотного зв'язку та інші інтерактивні форми без потреби програмування. Перш за все починаємо з встановлення Contact Form 7, для цього через адмін-панель WordPress переходимо у розділ «Плагіни > Додати новий». У пошуку вводимо 'Contact Form 7'. Натискаємо кнопку «Встановити» та далі — «Активувати», що показано на рис. 3.24.

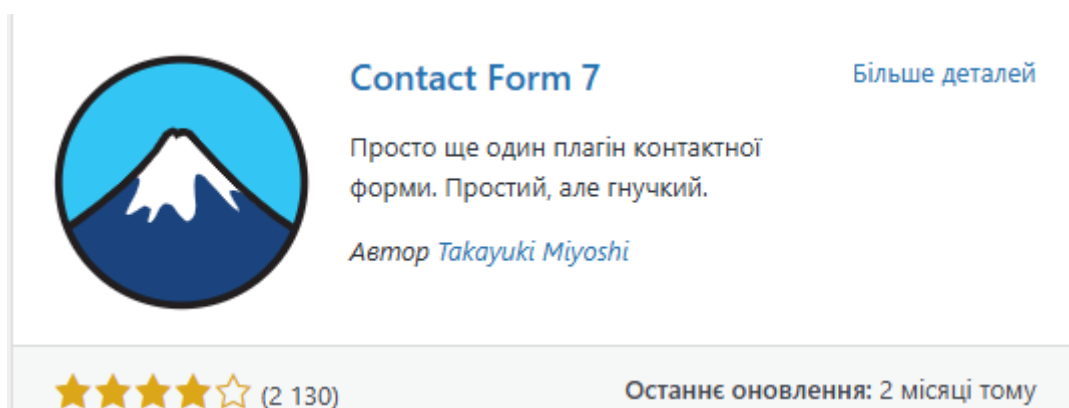


Рисунок 3.24 Встановлення плагіну.

Наступним кроком переходимо до створення самої контактної форми. Після встановлення з'являється розділ «Зв'язок -> Контактні форми». Створюємо нову форму, як показано на рис. 3.25.

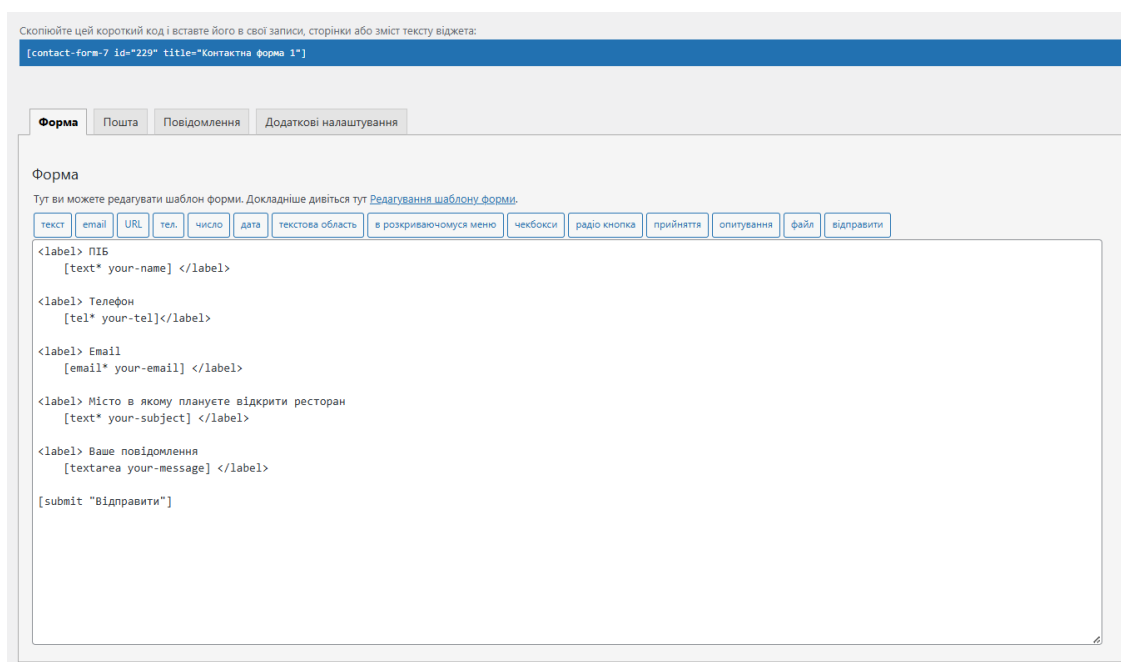


Рисунок 3.25 Створення форми.

Далі налаштуємо на яку адресу буде відправлятися форма, та передбачаємо повідомлення після відправки, чи з невірною інформацією. Тоді можемо додавати форму на сторінку, для того щоб відобразити контактну форму на сайті, достатньо скопіювати автоматично згенерований шорткод [contact-form-7 id="123" title="Контактна форма"] і вставити його на головну секцію сторінку через редактор Beaver Builder і отримуємо наступний результат, який показано на рис. 3.26:

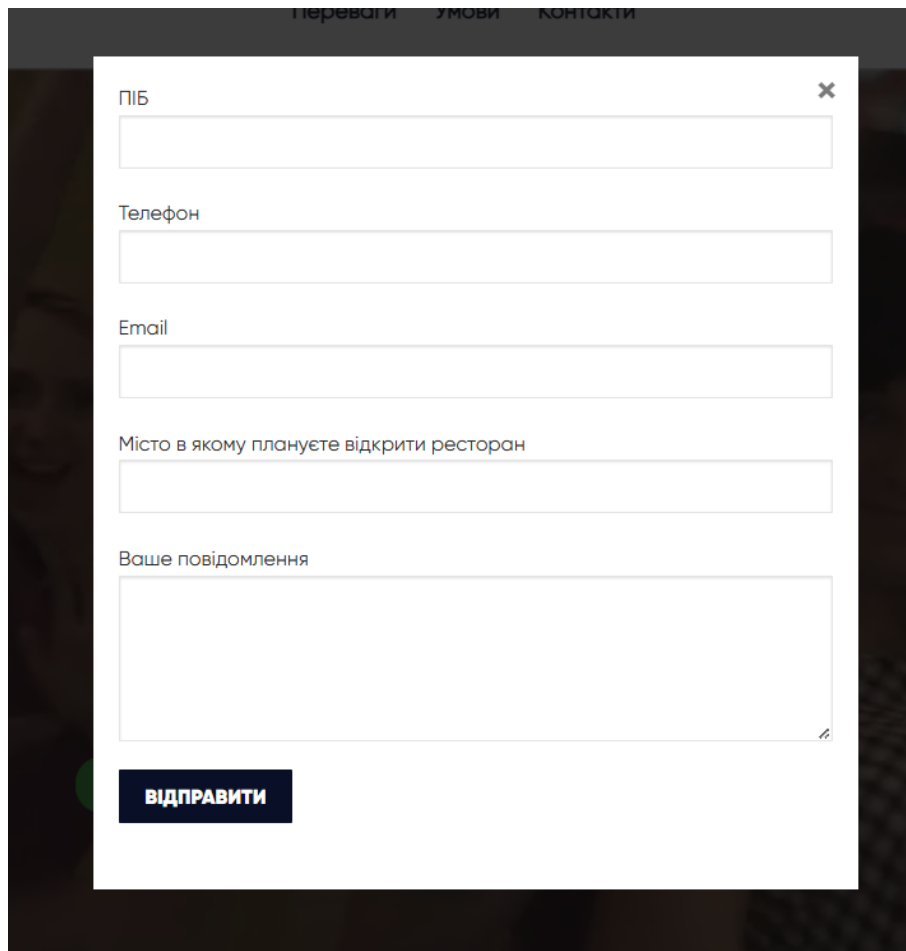
The image shows a contact form with the following fields: "ПІБ" (Name), "Телефон" (Phone), "Email", "Місто в якому плануєте відкрити ресторан" (City where you plan to open a restaurant), and "Ваше повідомлення" (Your message). There is a dark blue button labeled "ВІДПРАВИТИ" (SEND) at the bottom. The form is set against a dark background with some navigation links visible at the top: "переваги", "умови", and "контакти".

Рисунок 3.26 Створена форма.

Для того, щоб захисту форми від спаму використовується Google reCAPTCHA v3. Щоб активувати її потрібно зареєструватися сайті на сайті <https://www.google.com/recaptcha/admin/create>. Ввести дані про інформаційну систему, а саме задати ярлик, обрати версію, ввести домен, та обрати google console проект, як показано на рис. 3.27. Після чого отримуємо Site Key і Secret Key.

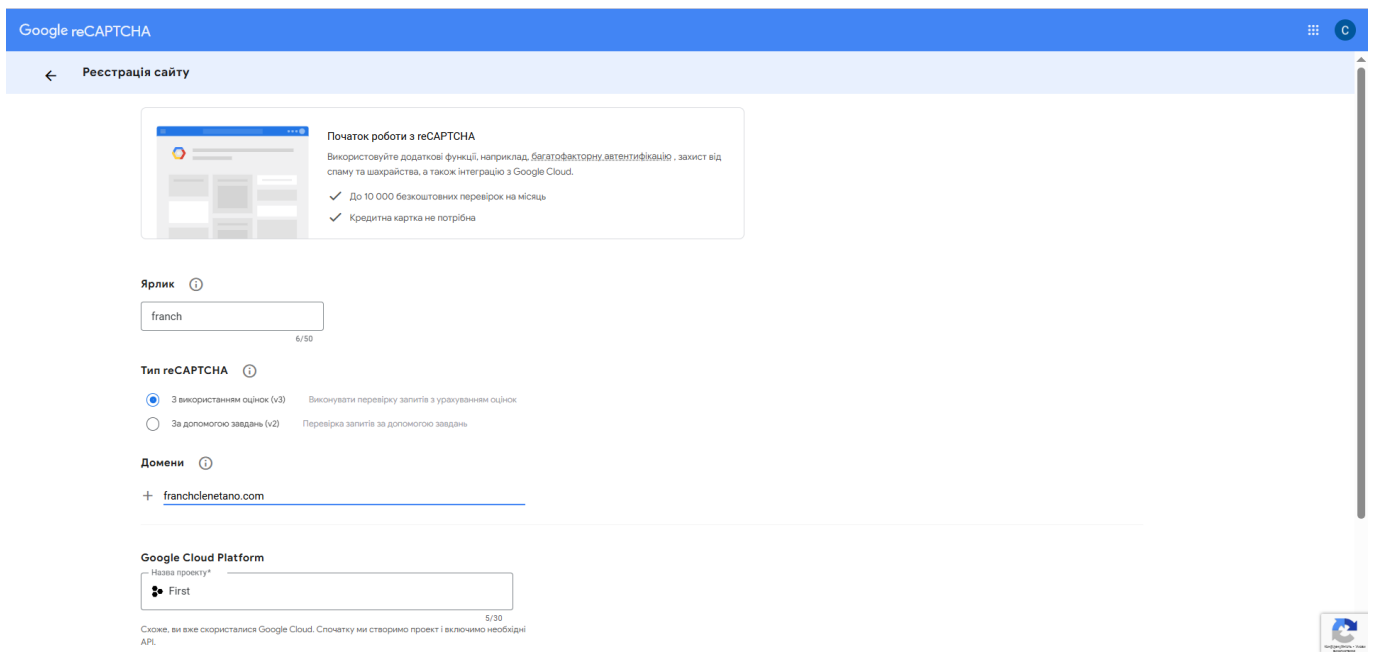


Рисунок 3.27. Створення reCAPTCHA.

Далі потрібно перейти в WordPress у «Зв'язок > Integration», Натиснути «Setup Integration» і ввести ключі. Після цього reCAPTCHA автоматично підключається до всіх форм на сайті.

На сторінці з формою з'явиться значок reCAPTCHA в нижньому правому куті. Коли користувач заповнює форму, система аналізує його поведінку та визначає, чи є це ботом. Це забезпечує надійний захист без потреби вводити капчу вручну.

Далі для гарантованої доставки листів, зокрема при використанні зовнішніх поштових сервісів (наприклад, Gmail або SMTP-серверу хостингу), встановлено плагін Post SMTP Mailer/Email Log. Встановлюємо та активуємо плагіна через розділ Плагіни (рис 3.28).

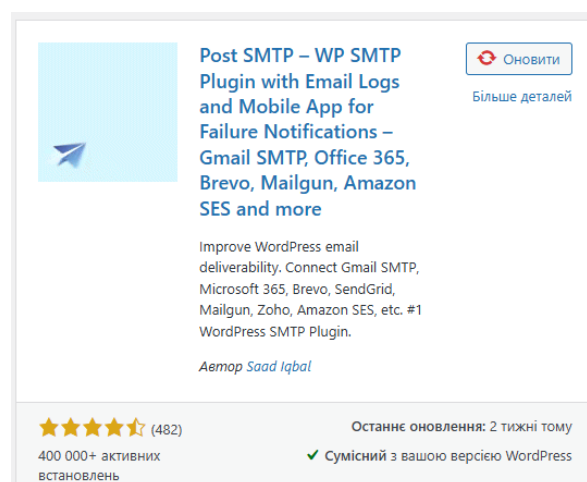


Рисунок 3.28 Встановлення Post SMTP.

Переходимо до меню Post SMTP > Налаштування, запускаємо майстра конфігурації, де вказуємо: email-адреса відправника, SMTP-сервер smtp.gmail.com, порт 587, тип шифрування (SSL/TLS), дані авторизації OAuth2, які потрібно отримати зареєструвавшись у google cloud console, та створивши Web Application рис. 3.29 згідно з нашим доменом та редіректором з налаштувань post smtp, як показано на рис. 3.30. Тоді проведено тестування надсилення листа та перевірка логів.

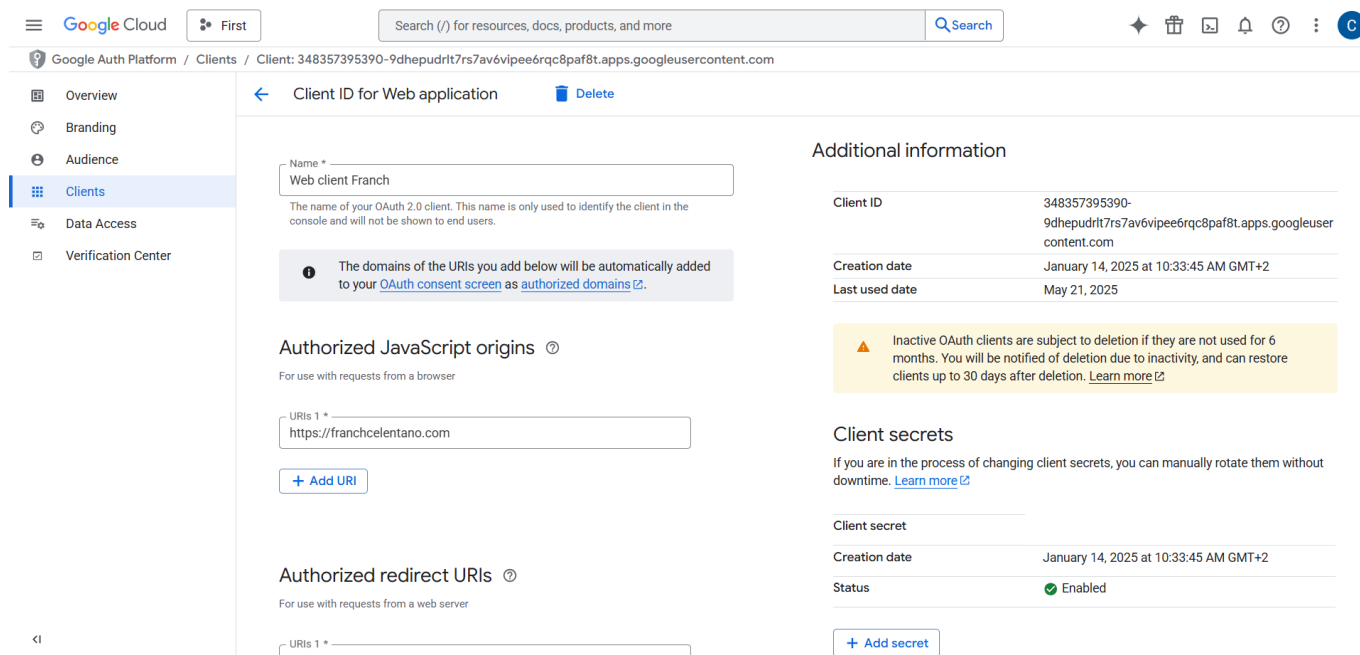


Рисунок 3.29 Web Application в Cloud Console.

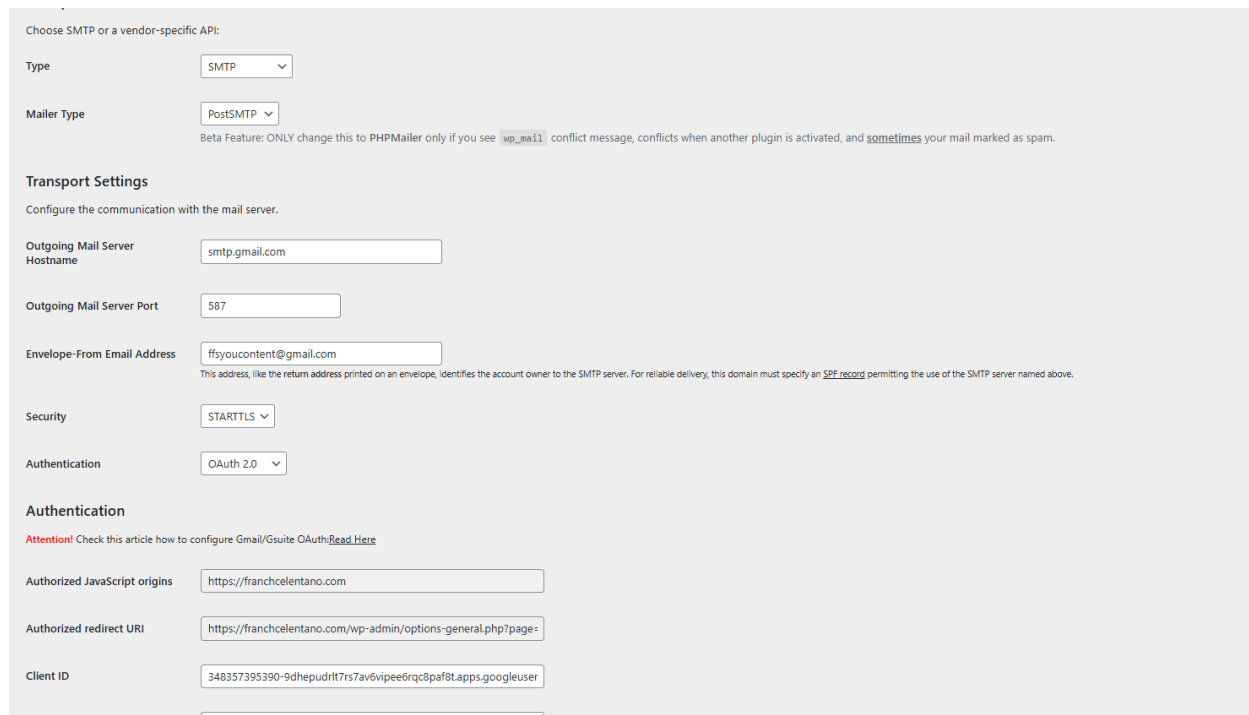


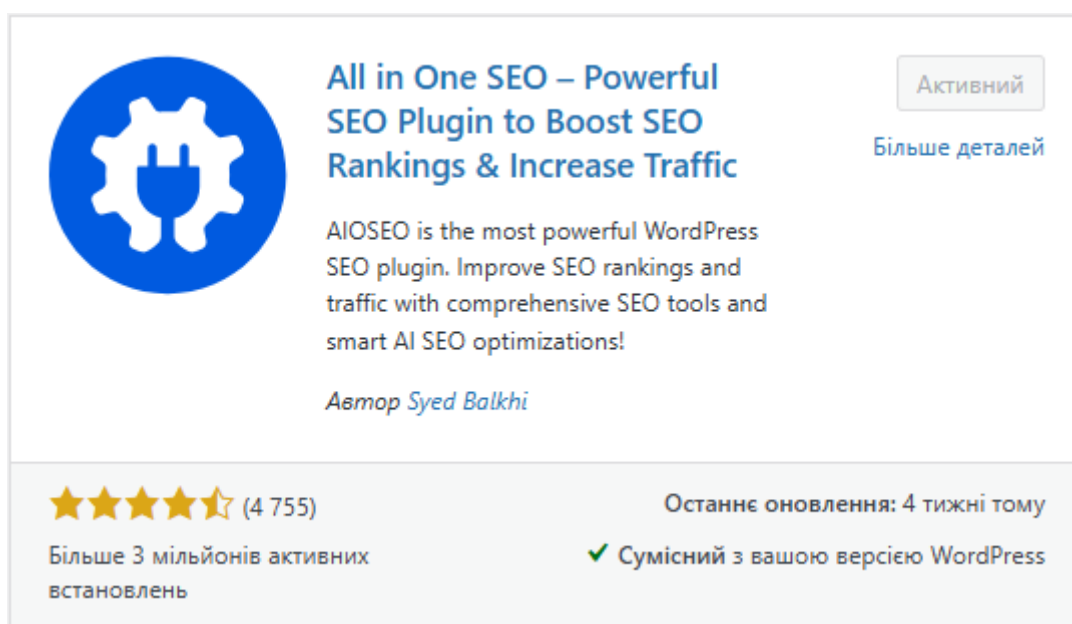
Рисунок 3.30 Налаштування Post SMTP.

Це рішення забезпечує стабільну доставку повідомлень із інформаційної системи, навіть якщо стандартна функція `wp_mail()` не працює через обмеження хостингу.

### 3.4. Базова SEO оптимізація

Для забезпечення видимості сайту у пошукових системах, таких як Google чи Bing, необхідно реалізувати базову SEO-оптимізацію (Search Engine Optimization — пошукова оптимізація). В межах даної інформаційної системи для франшизи Pizza Celentano була використана популярна SEO-платформа All in One SEO (AIOSEO), яка інтегрується в WordPress та надає гнучкі інструменти для покращення органічного трафіку. Плагін All in One SEO автоматизує процеси оптимізації сторінок та записів, надаючи зручний інтерфейс для налаштування мета-тегів (title, description) для кожної сторінки; генерації карти сайту (sitemap.xml); оптимізації відкритих графів для Facebook та Twitter (Open Graph, Twitter Cards); керування індексацією контенту (noindex, canonical); перевірки читабельності та SEO-рейтингів для кожного запису.

Для того щоб встановити та активувати плагін переходимо у панель керування WordPress, у розділі Плагіни > Додати новий та вводимо “All in One SEO”, після встановлення плагіна тиснемо кнопку «Активувати»(рис 3.31).



**All in One SEO – Powerful SEO Plugin to Boost SEO Rankings & Increase Traffic**

Активний

Більше деталей

AIOSEO is the most powerful WordPress SEO plugin. Improve SEO rankings and traffic with comprehensive SEO tools and smart AI SEO optimizations!

Автор Syed Balkhi

★★★★★ (4 755)

Більше 3 мільйонів активних встановлень

Останнє оновлення: 4 тижні тому

✓ Сумісний з вашою версією WordPress

Рисунок 3.31 Плагін All in One SEO.

Далі відкривається покроковий майстер налаштування, в якому вказуємо тип сайту, назва компанії, соціальні мережі та інші базові параметри. Після активації AIOSEO додає панель налаштувань до кожної сторінки або запису. Для кожного матеріалу можна вказати:

- SEO-назву (Title Tag) — відображається у результатах пошуку;
- Мета-опис (Meta Description) — короткий опис сторінки (до 160 символів);
- Ключові слова (в окремих режимах) — для аналізу релевантності.

Що і було зроблено для основної сторінки інформаційної ситсеми, як показано на рис. 3.32.

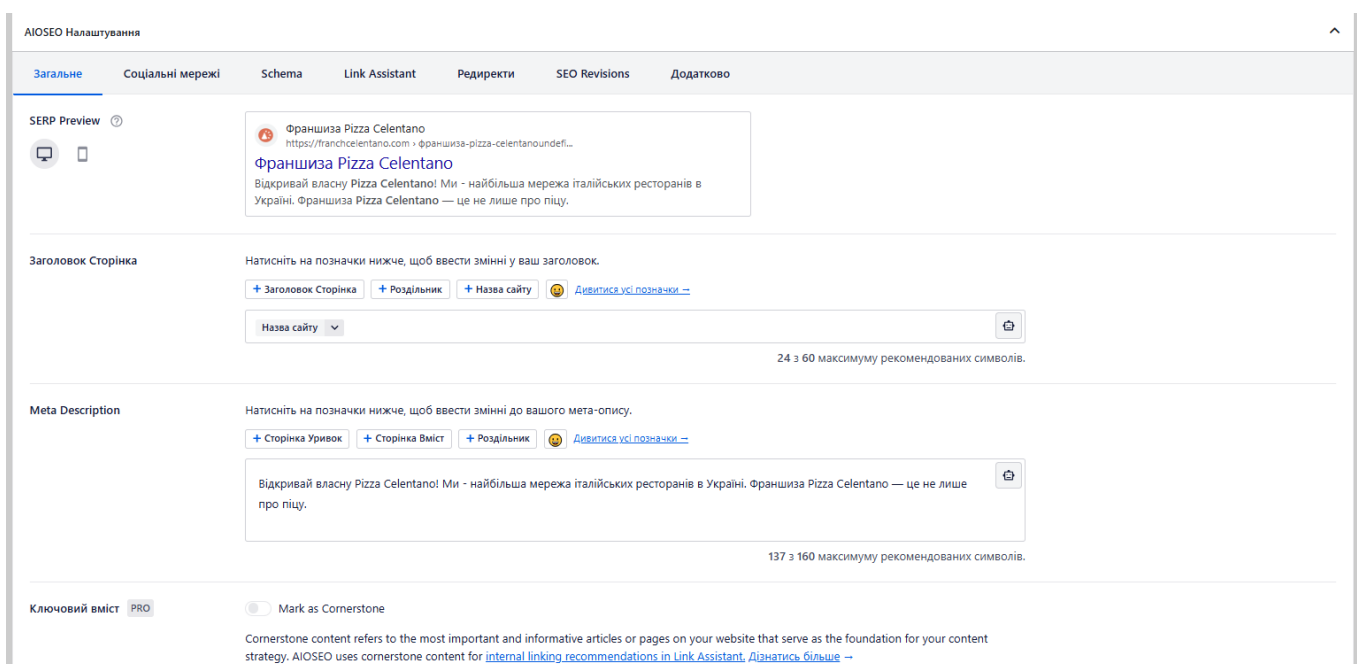


Рисунок 3.32 Налаштування All in One SEO для сторінки.

Карта сайту — це XML-файл, що містить усі важливі сторінки сайту. All in One SEO генерує її автоматично. Шлях до мапи: <https://franchcelentano.com/sitemap.xml> Карта автоматично оновлюється після створення нового контенту. Вона використовується для індексації Google Search Console. Також у вкладці соціальні мережі налаштовано параметри:

- зовнішній вигляд посилань при поширенні у Facebook та Instagram;
- стандартні зображення для попереднього перегляду;
- прив'язка до облікових записів у соцмережах.

Встановлений плагін має перевірку SEO-рейтингу сторінки у вигляді шкали (від 0 до 100), контроль індексації: можливість заборонити певні сторінки до індексації, локальне SEO налаштування — якщо активувати відповідне розширення, можна налаштувати відображення бізнесу у локальних пошуках Google.

Застосування базової SEO-оптимізації є невід’ємним етапом у розробці сучасної інформаційної системи, особливо коли йдеться про комерційний ресурс, що має на меті залучення нових користувачів і партнерів — таким як франшизи Pizza Celentano.

Використання плагіна All in One SEO (AIOSEO) дало змогу:

- централізовано керувати метаданими сайту (title, description, canonical);
- забезпечити формування автоматичної XML-карти сайту, що пришвидшує індексацію в Google;
- налаштувати підтримку соціальних мереж (Open Graph, Twitter Cards), що особливо важливо для маркетингових кампаній у Facebook та Instagram;
- реалізувати контроль над індексацією контенту (наприклад, приховати технічні сторінки або чернетки від сканування роботами);
- перевіряти рівень SEO-оптимізації кожної сторінки завдяки інтегрованому аналізу й отримувати практичні поради щодо покращення.

Особливо цінною є можливість встановити динамічні шаблони для заголовків та описів, що дозволяє значно зекономити час при масштабуванні сайту. Автоматична генерація цих елементів на основі назви сторінки чи категорії дозволяє зберегти логічну структуру мета-даних та уникнути дублювання.

Крім того, в комбінації з Google Search Console та Google Analytics, плагін AIOSEO дозволяє глибше аналізувати ефективність контенту та покращувати видимість сайту в результатах пошуку.

Загалом, впровадження SEO-механізмів за допомогою All in One SEO стало ефективним рішенням для підвищення як органічного трафіку, так і загальної привабливості сайту для потенційних франчайзі, інвесторів і відвідувачів. Це підтверджує важливість пошукової оптимізації навіть на ранніх етапах розвитку цифрового продукту.

### 3.5. Адаптивність інформаційної системи

У сучасних умовах, коли більшість користувачів заходять на сайти зі смартфонів і планшетів, адаптивність інформаційної системи є не просто бажаною, а критично необхідною. За статистикою, понад 65% трафіку у сфері ресторанного бізнесу генерується саме з мобільних пристроїв. Тому для сайту франшизи Pizza Selentano була реалізована повноцінна адаптивна верстка. Під час розробки було дотримано ключових принципів адаптивного дизайну(рис. 3.33):

- використання відсоткових ширин (`width: 100%, max-width`) замість фіксованих пікселів;
- застосування медіа-запитів (`media queries`) для зміни стилів залежно від ширини екрану;
- організація гнучких сіток (`flexbox, grid`), які забезпечують рівномірне розташування елементів на різних розмірах дисплея;
- зменшення або приховання другорядного контенту на мобільних пристроях (наприклад, анімацій або великих банерів).

Окрема увага приділена інтерактивній карті ресторанів, яка:

- при завантаженні на мобільному автоматично масштабується (`setView`) та деактивує скрол (`scrollWheelZoom.disable()`), щоб не заважати користувачу перегортати сторінку;
- підтримує жестовий масштаб (`touchZoom.enable()`);
- відкриває попередження про масштаб лише у десктопній версії.

Також передбачено оптимальне відображення елементів керування (пошук, фільтр) у мобільному форматі, через вертикальне групування в межах одного блоку, зі збереженням UX.

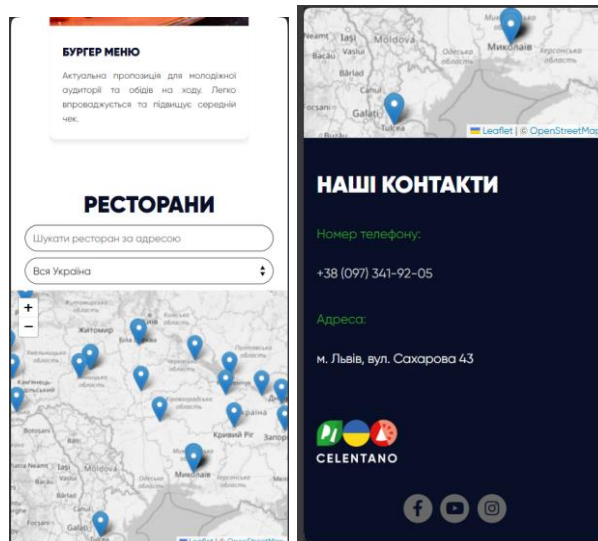
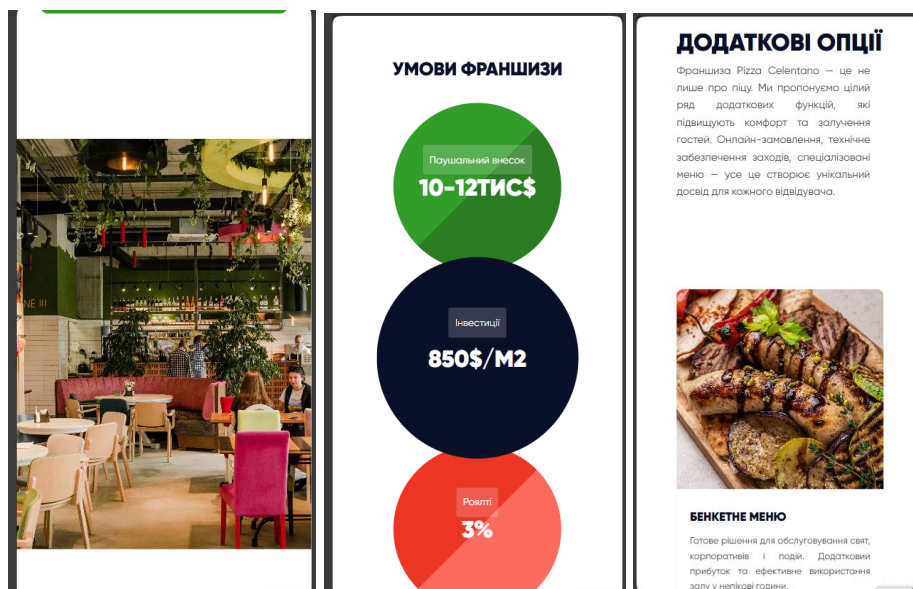
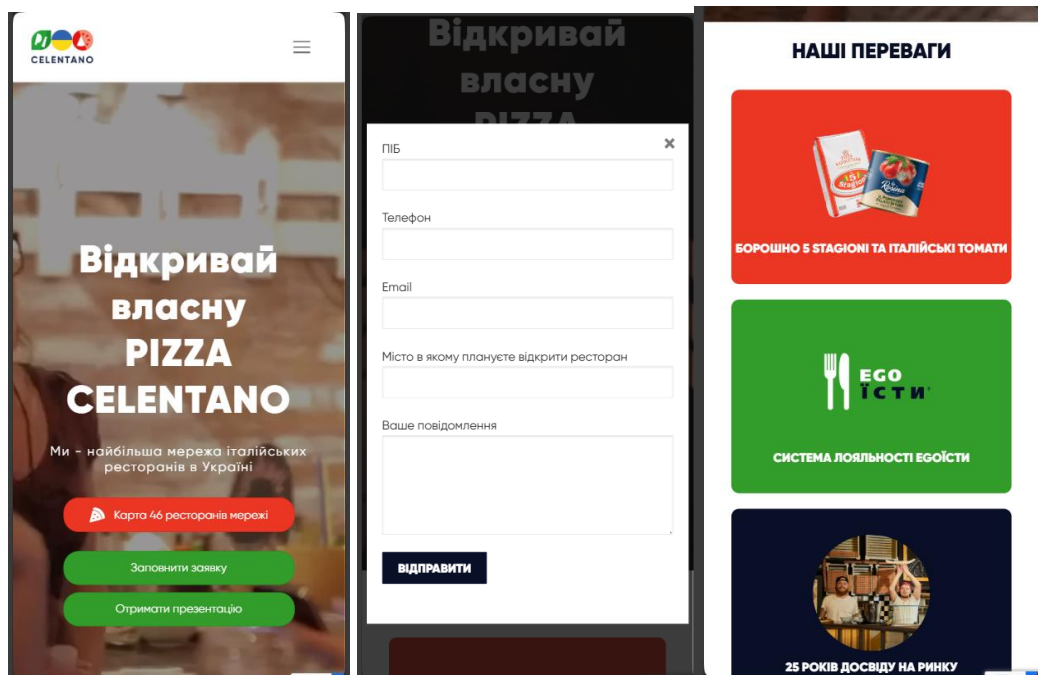


Рисунок 3.33. Адаптивний дизайн

Завдяки реалізації адаптивного дизайну сайт отримав такі переваги:

- покращення юзабіліті на різних пристроях;
- зниження показника відмов (bounce rate);
- підвищення часу перебування користувача на сайті;
- відповідність рекомендаціям Google для мобільних версій (Mobile-Friendly Test).

Таким чином, адаптивна верстка стала ключовим фактором зручності користування сайтом франшизи та позитивно вплинула на його технічні метрики та SEO-показники.

## ВИСНОВОК

У дипломній роботі було розроблено адаптивну інформаційну систему для презентації та продажу франшизи мережі Pizza Celentano на базі CMS WordPress з використанням конструктора Beaver Builder та плагіна Advanced Custom Fields.

Система забезпечує такі функціональні можливості як, структурування контенту: за допомогою ACF створено групи полів для текстових блоків, фінансових показників, кейсів успішних франчайзі та мультимедійних матеріалів, що дозволяє маркетологам швидко оновлювати інформацію без втручання розробників.

Розроблено інтерактивну карту ресторанів за допомогою інтеграція Leaflet.js та OpenStreetMap з даними з ACF-полів надає користувачам можливість фільтрувати локації за містами та форматом закладів, переглядати деталі кожного об'єкта через спливаючі вікна. Реалізовано збір заявок через контактні форми з використання Contact Form 7 із захистом Google reCAPTCHA v3 та надійною доставкою листів через WP Mail SMTP, що гарантує отримання якісних лідів.

Забезпечено захист даних та безпека, що реалізується за допомогою впровадження SSL/TLS шифрування, двофакторної аутентифікації, веб-фаєрволи Wordfence/Sucuri та централізованого логування, що забезпечує відповідність сучасним стандартам безпеки.

Проведене тестування підтвердило коректність роботи всіх модулів системи, адаптивність інтерфейсу на різних пристроях та стійкість до навантажень.

Розроблена система готова, розміщена на хостингу компанії за адресою <https://franchcelentano.com>, впроваджена у промислову експлуатацію та відповідає всім вимогам технічного завдання.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Дослідження ринку онлайн-замовлень їжі 2023. – Київ, 2023. – 54 с.
2. Аналітика українського ринку доставки їжі 2022. – Львів: Аналітичний центр "Інфомедіа", 2022. – 48 с.
3. Customer Behavior Trends in Online Food Ordering / Джонсон А., Сміт Д. – Нью-Йорк: Springer, 2022. – 234 с.
4. Challenges in Online Food Delivery Systems / Міллер К. – Лондон: Routledge, 2021. – 189 с.
5. Benefits of Creating Own Food Delivery Platform – Київ: Видавничий дім "Бізнес-аналітика", 2022. – 65 с.
6. Офіційний сайт WordPress [Електронний ресурс]. – Режим доступу: <https://wordpress.org/> (дата звернення: 25.03.2025).
7. Офіційний сайт WooCommerce [Електронний ресурс]. – Режим доступу: <https://woocommerce.com/> (дата звернення: 25.03.2025).
8. WooCommerce Optimization Guide / Вільямс П. – Сан-Франциско: TechPress, 2022. – 148 с.
9. Основи інформаційних систем та інформаційних технологій / Савчук І.Г., Мельник О.В. – Київ: Видавничий дім "Кондор", 2021. – 312 с.
10. Закон України "Про захист персональних даних" від 01.06.2010 №2297-VI [Електронний ресурс]. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/2297-17> (дата звернення: 25.03.2025).
11. HTML5 and CSS3 All-in-One For Dummies / Harris A. – Нью-Йорк: Wiley Publishing, 2014. – 1080 с.
12. SEO 2023: Learn Search Engine Optimization with Smart Internet Marketing Strategies / Джеймс К. – Лондон: Independently published, 2023. – 224 с.
13. Domain Name System Basics / Албертсон Дж. – Бостон: O'Reilly Media, 2016. – 206 с.
14. Вибір хостинг-провайдера: практичні рекомендації / Гнатюк І.О. – Київ: "ІТ-консалтинг", 2022. – 76 с.
15. WordPress Hosting Guide / Тейлор М. – Нью-Йорк: DigitalPress, 2023. – 142 с.

16. Плагіни для WordPress: Посібник з налаштування та використання / Романенко С.П. – Київ: Видавничий дім "Веб-технології", 2022. – 188 с.
17. Contact Form 7 Documentation [Електронний ресурс]. – Режим доступу: <https://contactform7.com/docs/> (дата звернення: 25.03.2025).
18. Google Maps Platform Documentation [Електронний ресурс]. – Режим доступу: <https://developers.google.com/maps/documentation> (дата звернення: 25.03.2025).

## ДОДАТОК А Function.php

```
<?php
/**
 * Customify functions and definitions
 *
 * @link https://developer.wordpress.org/themes/basics/theme-functions/
 *
 * @package customify
 */

/**
 * Same the hook `the_content`
 *
 * @TODO: do not effect content by plugins
 *
 * 8 WP_Embed:run_shortcode
 * 8 WP_Embed:autoembed
 * 10 wptexturize
 * 10 wpautop
 * 10 shortcode_unautop
 * 10 prepend_attachment
 * 10 wp_filter_content_tags || wp_make_content_images_responsive
 * 11 capital_P_dangit
 * 11 do_shortcode
 * 20 convert_smilies
 */
global $wp_embed;
add_filter( 'customize_the_content', array( $wp_embed, 'run_shortcode' ), 8 );
add_filter( 'customize_the_content', array( $wp_embed, 'autoembed' ), 8 );
add_filter( 'customize_the_content', 'wptexturize' );
add_filter( 'customize_the_content', 'wpautop' );
add_filter( 'customize_the_content', 'shortcode_unautop' );
if ( function_exists( 'wp_filter_content_tags' ) ) {
    add_filter( 'customize_the_content', 'wp_filter_content_tags' );
} else {
    add_filter( 'customize_the_content', 'wp_make_content_images_responsive' );
}

add_filter( 'customize_the_content', 'capital_P_dangit' );
add_filter( 'customize_the_content', 'do_shortcode' );
add_filter( 'customize_the_content', 'convert_smilies' );
/**
 * Same the hook `the_content` but not auto P
 *
 * @TODO: do not effect content by plugins
 *
 * 8 WP_Embed:run_shortcode
 * 8 WP_Embed:autoembed
 * 10 wptexturize
 * 10 shortcode_unautop
 * 10 prepend_attachment
 * 10 wp_filter_content_tags || wp_make_content_images_responsive
 * 11 capital_P_dangit
 * 11 do_shortcode
 * 20 convert_smilies
 */
add_filter( 'customize_the_title', array( $wp_embed, 'run_shortcode' ), 8 );
add_filter( 'customize_the_title', array( $wp_embed, 'autoembed' ), 8 );
add_filter( 'customize_the_title', 'wptexturize' );
add_filter( 'customize_the_title', 'shortcode_unautop' );
```

```

if ( function_exists( 'wp_filter_content_tags' ) ) {
    add_filter( 'customize_the_title', 'wp_filter_content_tags' );
} else {
    add_filter( 'customize_the_title', 'wp_make_content_images_responsive' );
}
add_filter( 'customize_the_title', 'capital_P_dangit' );
add_filter( 'customize_the_title', 'do_shortcode' );
add_filter( 'customize_the_title', 'convert_smilies' );

// Include the main Customize class.
require_once get_template_directory() . '/inc/class-customize.php';

/**
 * Main instance of Customize.
 *
 * Returns the main instance of Customize.
 *
 * @return Customize
 */
function customize() {
    // phpcs:ignore WordPress.NamingConventions.ValidFunctionName.
    return Customize::get_instance();
}

customize();

function my_upload_size_limit( $limit ) {
    add_filter( 'upload_size_limit', 'my_upload_size_limit' );
    return wp_convert_hr_to_bytes( '50M' );
}

function create_restaurants_post_type() {
    register_post_type( 'restaurant',
        array(
            'labels' => array(
                'name' => __( 'Ресторани', 'textdomain' ),
                'singular_name' => __( 'Ресторан', 'textdomain' ),
                'menu_name' => __( 'Ресторани', 'textdomain' ),
                'add_new' => __( 'Додати новий', 'textdomain' ),
                'add_new_item' => __( 'Додати новий ресторан', 'textdomain' ),
                'edit_item' => __( 'Редагувати ресторан', 'textdomain' ),
            ),
            'public' => true,
            'has_archive' => true,
            'menu_icon' => 'dashicons-store',
            'supports' => array( 'title', 'editor', 'thumbnail' ),
            'show_in_rest' => true,
        )
    );
}
add_action( 'init', 'create_restaurants_post_type' );

function restaurants_map_shortcode() {
    $restaurants = get_posts([
        'post_type' => 'restaurant',
        'numberposts' => -1
    ]);

    $data = [];

    foreach ( $restaurants as $restaurant ) {
        $image = get_field( 'image', $restaurant->ID );
    }
}

```

```

        $image_url = $image ? $image['url'] : '';
    $data[] = [
        'name' => get_the_title($restaurant->ID),
        'city' => get_field('city', $restaurant->ID),
        'address' => get_field('address', $restaurant->ID),
        'coords_lat' => get_field('coords_lat', $restaurant->ID),
        'coords_lng' => get_field('coords_lng', $restaurant->ID),
        'image' => $image_url,
        'facebook' => get_field('facebook', $restaurant->ID),
        'instagram' => get_field('instagram', $restaurant->ID)
    ];
}

wp_enqueue_style('leaflet-css', 'https://unpkg.com/leaflet/dist/leaflet.css');
wp_enqueue_script('leaflet-js', 'https://unpkg.com/leaflet/dist/leaflet.js', [], null, true);

// Унікальні версії для уникнення кешу
$version = filemtime(get_stylesheet_directory() . '/js/map.js');

// Підключаємо власні CSS і JS
wp_enqueue_style('map-styles', get_stylesheet_directory_uri() . '/css/map.css', [], $version);
wp_enqueue_script('map-script', get_stylesheet_directory_uri() . '/js/map.js', ['leaflet-js'], $version, true);

// Передаємо дані в JS
wp_localize_script('map-script', 'restaurantsData', $data);

ob_start(); ?>

<div class="restaurants-map-container">
    <div class="filters">
        <h2 class="title">Ресторани</h2>
        <input type="text" id="search-address" placeholder="Шукати ресторан за адресою">
        <select id="filter-city">
            <option value="">Вся Україна</option>
        </select>
    </div>
    <div id="map"></div>
</div>

<?php
return ob_get_clean();
}
add_shortcode('restaurants_map', 'restaurants_map_shortcode');

```

## ДОДАТОК Б

### map.js та css

```
document.addEventListener("DOMContentLoaded", function () {

    // Створюємо карту, але не задаємо фіксований центр
    let map = L.map('map', { scrollWheelZoom: false });

    // Отримуємо bounds (межі) всіх точок
    let bounds = L.latLngBounds();

    restaurantsData.forEach((restaurant) => {
        let lat = parseFloat(restaurant.coords_lat);
        let lng = parseFloat(restaurant.coords_lng);

        if (!isNaN(lat) && !isNaN(lng)) {
            bounds.extend([lat, lng]); // Додаємо координати ресторанів у bounds
        }
    });

    // Якщо є хоча б один ресторан, центруємо карту
    if (bounds.isValid()) {
        map.fitBounds(bounds, { padding: [20, 20] }); // Масштабуємо карту так, щоб показати всі точки
    } else {
        map.setView([49.8397, 24.0297], 6); // Якщо ресторанів немає, використовуємо стандартний центр
    }

    // Використовуємо сіру карту Carto
    L.tileLayer('https://tile.openstreetmap.org/{z}/{x}/{y}.png', {
        attribution: '&copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a>',
        detectRetina: true,
        language: 'uk' // Додаємо підтримку української мови
    }).addTo(map);

    let mapContainer = document.getElementById("map"); // Отримуємо блок карти

    if (!mapContainer) {
        console.error(" Блок #map не знайдено!");
        return;
    }

    // Створюємо підказку про Ctrl + колесо миші
    setTimeout(() => {
        const zoomHint = document.createElement("div");
        zoomHint.id = "zoomHint";
        zoomHint.innerHTML = "Утримуйте Ctrl і прокрутіть, щоб змінити масштаб карти";
        mapContainer.appendChild(zoomHint);
        console.log(" Підказка про Ctrl додана у DOM:", zoomHint);
    }, 1000); // Затримка 1 сек, щоб впевнитись, що DOM повністю завантажений

    // Вимикаємо зум колесом миші за замовчуванням
    map.scrollWheelZoom.disable();

    // Вмикаємо зум тільки при натисканні Ctrl
    document.addEventListener("keydown", function (e) {
        if (e.key === "Control") {
            console.log(" Натиснуто Ctrl – увімкнено зум");
            map.scrollWheelZoom.enable(); // Увімкнути зум тільки якщо натиснуто Ctrl
        }
    });
});
```

```

document.addEventListener("keyup", function (e) {
  if (e.key === "Control") {
    console.log(" Відпущено Ctrl – вимкнено зум");
    map.scrollWheelZoom.disable(); // Вимикаємо зум при відпусканні Ctrl
  }
});

// Запобігаємо зміні масштабу сторінки при натисканні Ctrl + колесо миші
window.addEventListener("wheel", function (e) {
  if (e.ctrlKey) {
    e.preventDefault(); // Вимикаємо зміну масштабу всієї сторінки
  }
}, { passive: false });

if (window.innerWidth < 768) {
map.setView([49.8397, 24.0297], 5); // Менший масштаб для мобільних
map.touchZoom.enable(); // Увімкнення жестового зуму
map.scrollWheelZoom.disable(); // Вимкнення скролу на мобільних
}

let markers = [];
let cities = new Set();

restaurantsData.forEach((restaurant) => {

  let lat = parseFloat(restaurant.coords_lat);
  let lng = parseFloat(restaurant.coords_lng);

  if (!isNaN(lat) && !isNaN(lng)) {
    let popupContent = `
      <div class="popup-container">
        
        <div class="popup-content">
          <div class="popup-text">
            <strong> м. ${restaurant.city}</strong><br>
            ${restaurant.address}
          </div>
          <div class="popup-icons">
            ${restaurant.facebook ? `<a href="${restaurant.facebook}" target="_blank"><i class="fa fa-facebook" aria-
hidden="true"></i></a>:`}
            ${restaurant.instagram ? `<a href="${restaurant.instagram}"
target="_blank"><i class="fa fa-instagram" aria-hidden="true"></i></a>:`}
          </div>
        </div>
      </div>
    `;

    let marker = L.marker([lat, lng]).addTo(map).bindPopup(popupContent);
    markers.push({ marker, city: restaurant.city, address: restaurant.address });

    if (restaurant.city) {
      cities.add(restaurant.city);
    }
  } else {
    console.warn("Помилка: некоректні координати для", restaurant.name);
  }
});

```

```

// Додаємо міста у фільтр
// Додаємо міста у фільтр (сортуємо їх по алфавіту)
const citySelect = document.getElementById("filter-city");

// Отримуємо масив міст, сортуємо і додаємо в select
const sortedCities = Array.from(cities).sort((a, b) => a.localeCompare(b, 'uk'));

sortedCities.forEach(city => {
  let option = document.createElement("option");
  option.value = city;
  option.textContent = city;
  citySelect.appendChild(option);
});

// Фільтр по містах
citySelect.addEventListener("change", function () {
  let selectedCity = this.value;
  let bounds = L.latLngBounds();

  // Очищаємо пошук
  searchInput.value = "";
  searchDropdown.style.display = "none";

  markers.forEach(({ marker, city }) => {
    if (selectedCity === "" || city === selectedCity) {
      marker.addTo(map);
      bounds.extend(marker.getLatLng());
    } else {
      marker.remove();
    }
  });

  if (bounds.isValid()) {
    map.fitBounds(bounds, { padding: [20, 20] });
  } else {
    map.setView([49.8397, 24.0297], 6);
  }
});

// Пошук за адресою
// Масив для autocomplete
let searchSuggestions = [];

restaurantsData.forEach(({ name, city, address, coords_lat, coords_lng }) => {
  if (city && address) {
    searchSuggestions.push({
      label: `${city}, ${address}`,
      value: { name, city, address, coords_lat, coords_lng }
    });
  }
});

// Додаємо autocomplete до пошуку
const searchInput = document.getElementById("search-address");
const searchDropdown = document.createElement("ul");
searchDropdown.id = "search-suggestions";
searchDropdown.style.position = "absolute";
searchDropdown.style.top = "100%";
searchDropdown.style.left = "29%";
searchDropdown.style.width = "35%";
searchDropdown.style.background = "white";

```

```

//searchDropdown.style.border = "1px solid #ccc";
searchDropdown.style.listStyle = "none";
searchDropdown.style.padding = "0";
searchDropdown.style.margin = "0";
searchDropdown.style.display = "none";
searchDropdown.style.zIndex = "1000";
//searchDropdown.style.borderRadius = "30px";
searchDropdown.style.maxHeight = "300px";
searchDropdown.style.overflow = "scroll";
searchInput.parentNode.appendChild(searchDropdown);

//document.querySelector('.search-wrapper').appendChild(searchDropdown);

// Оновлення підказок при введенні
searchInput.addEventListener("input", function () {
  let searchValue = this.value.toLowerCase();
  searchDropdown.innerHTML = "";
  searchDropdown.style.display = "none";

  if (searchValue.length < 2) return; // Не показуємо підказки для коротких запитів

  let filteredResults = searchSuggestions.filter(item =>
    item.label.toLowerCase().includes(searchValue)
  );

  if (filteredResults.length > 0) {
    searchDropdown.style.display = "block";
    filteredResults.forEach(item => {
      let listItem = document.createElement("li");
      listItem.textContent = item.label;
      listItem.style.padding = "8px";
      listItem.style.cursor = "pointer";
      listItem.style.borderBottom = "1px solid #eee";
      listItem.addEventListener("click", function () {
        searchInput.value = item.label; // Заповнюємо поле вибраним варіантом
        searchDropdown.style.display = "none";
        let selectedRestaurant = item.value;
        let lat = parseFloat(selectedRestaurant.coords_lat);
        let lng = parseFloat(selectedRestaurant.coords_lng);
        if (!isNaN(lat) && !isNaN(lng)) {
          map.panTo([lat, lng]); // Переміщуємо карту
          markers.forEach(({ marker }) => {
            let markerPos = marker.getLatLng();
            if (markerPos.lat === lat && markerPos.lng === lng) {
              marker.openPopup(); // Відкриваємо pop-up
            }
          });
        }
      });
    });
  }
});
searchDropdown.appendChild(listItem);
});
}
});
// Закриваємо підказки при кліку поза ними
document.addEventListener("click", function (e) {
  if (!searchInput.contains(e.target) && !searchDropdown.contains(e.target)) {
    searchDropdown.style.display = "none";
  }
});
});

```

```

.restaurants-map-container {
  width: 100%;
  max-width: 100%;
  margin: auto;
  text-align: center;
}
.title{
  font-size: 36px;
  font-weight: bold;
  color: #090F27;
  text-transform: uppercase;
  font-family: 'Gilroy-Black';
  margin:0;
}

.filters {
  display: flex;
  justify-content: space-evenly;
  align-items: center;
  gap: 10px;
  margin-bottom: 10px;
  position: relative;
}

#search-address, #filter-city {
  border-radius: 5px;
  border: 1px solid #090F27;
  border-radius:30px;
  font-family: 'Gilroy-Regular';
}
#search-address {
  width:500px;
}
#filter-city {
  width:300px;
}

#map {
  display:block;
  position: relative;
  height: 600px;
  width: 100%;
  background-color: #f5f5f5;
  font-family: 'Gilroy-Regular';
}
@media (max-width: 768px) {
  #map {
    height: 400px !important; /* Менший розмір карти для мобільних */
  }

  .filters {
    flex-direction: column; /* Фільтри в колонку */
    align-items: center;
  }

  #search-address, #filter-city {
    width: 90%;
  }
  #zoomHint{

```

```

        display:none!important;
    }
}
#zoomHint {
    position: absolute;
    bottom: 15px;
    left: 50%;
    transform: translateX(-50%);
    background: rgba(0, 0, 0, 0.8);
    color: white;
    padding: 10px 15px;
    border-radius: 8px;
    box-shadow: 0 2px 4px rgba(0, 0, 0, 0.2);
    font-size: 14px;
    z-index: 1000;
    text-align: center;
    display: block; /* Гарантуємо, що елемент видимий */
    font-family: 'Gilroy-Regular';
}

.leaflet-container {
    background-color: #f8f9fa;
}

.leaflet-tile {
    filter: grayscale(1);
}

.leaflet-popup-content {
    width: 280px !important; /* Збільшений розмір pop-up'у */
    display: flex;
    flex-direction: column;
    align-items: center;
    padding: 0;
    margin: 0;
}

.leaflet-container a.leaflet-popup-close-button{
    color:white;
}

.leaflet-popup-content-wrapper {
    padding: 0px;
}

.popup-container {
    display: flex;
    flex-direction: column;
    width: 100%;
    box-sizing: border-box;
}

.popup-image {
    width: 100%;
    height: 150px;
    object-fit: cover;
    border-top-left-radius: 10px;
    border-top-right-radius: 10px;
}

.popup-content {
    display: flex;
    flex-direction: row;
    justify-content: space-between;
}

```

```

width: 100%;
padding: 10px;
box-sizing: border-box;
}

.popup-text {
flex: 2;
padding-left: 10px;
text-align: left;
font-size: 14px;
font-family: 'Gilroy-Regular';
}

.popup-text strong {
font-size: 16px;
color: #333;
font-family: 'Gilroy-Black';
}

.popup-icons {
flex: 1;
display: flex;
justify-content:center;
align-items: center;
gap: 5px;
}

.popup-icons a {
display: flex;
align-items: center;
justify-content: center;
width: 30px;
height: 30px;
background-color: #090F27;
color: white;
border-radius: 50%;
text-decoration: none;
font-size: 16px;
}

.popup-icons a:hover {
background-color: #090F20;
}
#search-suggestions{
font-family: 'Gilroy-Regular';
}
@media (max-width: 768px) {
#search-suggestions{
width: 90%!important;
left:5%!important;
}
}

```