

Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки

**БАКАЛАВРСЬКА ДИПЛОМНА РОБОТА**

на тему:

**Web-сайт з використанням системи керування вмістом WordPress**

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

Виконав: студент 4 курсу, групи ІКІ-186  
спеціальності 123 — Комп'ютерна інженерія

Савенко А.В.

Керівник к.т.н., доц.каф. ОТ

Кожем'яко А. В.

Рецензент д.т.н проф, каф. МБІС

Яремчук Ю.Є.

Допущено до захисту  
д.т.н., проф. Азаров О.Д.

"21" червня 2022 р.

Вінницький національний технічний університет

Факультет \_\_\_\_\_ інформаційних технологій та комп'ютерної інженерії \_\_\_\_\_

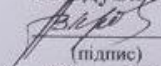
Кафедра \_\_\_\_\_ обчислювальної техніки \_\_\_\_\_

Освітньо-кваліфікаційний рівень \_\_\_\_\_ бакалавр \_\_\_\_\_

Спеціальність \_\_\_\_\_ 123 — «Комп'ютерна інженерія» \_\_\_\_\_

**ЗАТВЕРДЖУЮ**

**Завідувач кафедри ОТ**

  
(підпис) д.т.н., проф. *O. D. Trapov*

«08» 02 2022 р.

## ЗАВДАННЯ НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Савенку Андрію Валерійовичу

1 Тема проекту «Web-сайт з використанням системи керування WordPress», керівник роботи к.т.н., доц. каф. ОТ Кожем'яко А.В. затверджені наказом вищого навчального закладу від «24» березня 2022 року № 66

2 Строк подання студентом проекту — 21.06. 2022 р.

3 Вихідні дані до проекту: призначення — система керування вмістом WordPress, основні підтримувані функції — перехід за посиланнями на зовнішні ресурси, просте налаштування сторінок.



4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ, аналіз предметної області, порівняльна характеристика систем керування вмістом, розробка структури та бази даних веб-сайту.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): сторінки Web-сайту, функціональна схема веб-сторінки.

6 Консультанти розділів роботи приведені в таблиці 1.

7 Дата видачі завдання «24» березня 2022 р.

Таблиця 1 — Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видані	завдання прийняті
Спеціальна частина	Кожем'яко А. В. доцент кафедри ОТ	 24.03.22р	 16.06.22р

8 Календарний план виконання БДР приведений в таблиці 2.

Таблиця 2 — Календарний план

Ч.ч.	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Отримання завдання	11.03.22	<i>БДР</i>
2	Збір інформації	12.03-15.03.22	<i>БДР</i>
3	Характеристика предметної області	16.03-18.03.22	<i>БДР</i>
4	Аналіз особливостей систем керування вмістом	20.03-26.03.22	<i>БДР</i>
5	Розробка структури та веб-дизайну Web-сайту	6.04-10.04.22	<i>БДР</i>
6	Функціонально-вартісний аналіз продукту	15.04-20.04.22	<i>БДР</i>
7	Попередній захист БДР	17.05-18.05.22	<i>БДР</i>
8	Оформлення дипломної роботи	17.06.22	<i>БДР</i>
9	Перевірка роботи на плагіат	19.06.22	<i>БДР</i>
10	Рецензування БДР	20.06.22	<i>БДР</i>
11	Захист БДР	21.06.22	<i>БДР</i>

Студент



Савенко А.В.

Керівник роботи



Кожем'яко А.В.

## АНОТАЦІЯ

Савенко А.В. Web-сайт з використанням системи керування вмістом WordPress. Бакалаврська кваліфікаційна робота зі спеціальності 123 — Комп'ютерна інженерія, Вінниця: ВНТУ, 2022. Загальний обсяг роботи 80 сторінок, 28 рисунків, 5 таблиць.

Метою роботи є створення Web-сайту при використанні системи керування вмістом WordPress (тема корпоративний сайт кав'ярні).

В дипломній роботі проаналізовано принцип роботи систем управління вмістом, досліджено їх переваги та недоліки. Було проведено порівняльну характеристику найбільш відомих CMS, таких як Joomla!, WordPress і Drupal.

В ході виконання було розроблено структуру корпоративної Web-сторінки кав'ярні та наочно продемонстровано принцип її створення за допомогою WordPress.

Також було проведено функціонально-вартісний аналіз розробки програмного продукту.

Ключові слова: Web-сайт, система керування вмістом, WordPress, шаблон.

## **ABSTRACT**

Savenko A.V. Web-site using WordPress content management system. Bachelor's thesis in the specialty 123 — Computer Engineering, Vinnytsia: VNTU, 2022. The total volume of work is 80 pages, 28 figures, 5 tables.

The aim of the work is to create a Web-site using the WordPress content management system (the theme of the corporate website of the cafe).

In the thesis the principle of operation of content management systems is analyzed, their advantages and disadvantages are investigated. A comparative description of the most famous CMS, such as Joomla !, WordPress and Drupal.

During the implementation, the structure of the corporate Web-page of the cafe was developed and the principle of its creation with the help of WordPress was clearly demonstrated.

Functional and cost analysis of software product development was also performed.

**Keywords:** Web-site, content management system, WordPress, template.

## ЗМІСТ

<b>ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ПОЗНАЧЕНЬ.....</b>	<b>8</b>
<b>ВСТУП.....</b>	<b>9</b>
<b>1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....</b>	<b>12</b>
1.1 Основні етапи створення веб-сайтів .....	12
1.1.1 Визначення проекту.....	12
1.1.2 Інформаційне наповнення сайту .....	13
1.1.3 Розробка веб-дизайну сайту .....	14
1.1.4 Написання програмного коду та верстка сайту .....	14
1.2 Локальні сервери .....	15
1.3 Web-сервери.....	17
1.4 Використання CMS для створення веб-сайтів .....	24
<b>2 ВИЗНАЧЕННЯ ОСНОВНИХ ХАРАКТЕРИСТИК CMS ТА ВИБІР ШАБЛОНІВ ДЛЯ ПРОЕКТУ САЙТА .....</b>	<b>28</b>
2.1 Загальний огляд систем розробки.....	28
2.2 Пошук та використання шаблонів при створенні веб-сайтів .....	31
2.3 Основні критерії порівняння.....	34
Порівняльна таблиця.....	39
<b>3 ПРОЕКТУВАННЯ WEB-САЙТ З ВИКОРИСТАННЯМ СИСТЕМИ КЕРУВАННЯ ВМІСТОМ WORDPRESS .....</b>	<b>40</b>
3.1 Опис структури та бази даних WordPress .....	40
3.2 Встановлення локального серверу XAMPP.....	43
3.3 Встановлення WordPress .....	48
3.4 Встановлення плагінів .....	51
3.5 Розробка структури та наповнення сторінок.....	54

					08-23.БДР.011.00.000 ПЗ			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	Web-сайт з використанням системи керування вмісту WordPress	<i>Лім.</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Розробив</i>		Савенко А. В.						
<i>Керівник</i>		Кожем'яко А. В.					6	80
<i>Рецензент</i>		Яремчук Ю. Є.				ВНТУ, гр. 1КІ-186		
<i>Н.контр.</i>		Швець С. І.						
<i>Затвердж.</i>		Азаров О.Д						

3.5.1 Структура веб-сайту .....	54
3.5.2 Інтерфейс користувача .....	57
3.6 Постановка задачі аналізу ефективності запропонованого рішення .....	59
3.7 Обґрунтування системи параметрів ПП .....	60
3.7.1 Кількісна оцінка параметрів .....	60
3.7.2 Аналіз експертного оцінювання параметрів .....	61
<b>ВИСНОВКИ</b> .....	65
<b>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ</b> .....	67
<b>ДОДАТОК А</b> Технічне завдання .....	69
<b>ДОДАТОК Б</b> Інтерфейс web-сайту .....	72
<b>ДОДАТОК В</b> Графічна структура бази даних .....	76
<b>ДОДАТОК Г</b> Порівняльна таблиця .....	77
<b>ДОДАТОК Д</b> Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень .....	80

**ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ПОЗНАЧЕНЬ**

**СКВ** — система керування вмістом

**ПП** — програмний продукт

**ФВА** — функціонально-вартісний аналіз

**CMS** — content management system

**WP** — WordPress

**ЕОМ** — електронно-обчислювальна машина



## ВСТУП

Першим в світі сайтом, який був створений в 1991 році. Його автором був Тім Бернерс-Лі. Перша інформацією на ньому, став опис нової технології World Wide Web.

У наш час веб-сайт може бути візитною карткою підприємства, фірми, вашим портфоліо. Також ми використовуємо веб-додатки для проведення онлайн уроків або зустрічі на дистанції, що є неймовірно зручно та комфортно. За допомогою сайту можна поширювати інформацію, швидко її коригувати, таким чином люди, що відвідали сайт будуть отримувати актуальну та най достовірнішу інформацію. За допомогою веб-сайту організація чи підприємець має можливість донести до людей величезну кількість інформації. Жодне інше середовище не може дати такого ж ефекту. Газети мають обмеження щодо кількості слів і місця для друку, а радіоприймачі можуть неправильно почутись або записати/запам'ятати інформацію. Поки у організації є власний веб-сайт, організація не має обмежень і може публікувати будь-який обсяг необхідної інформації. Сайт відкритий 24 години на добу 7 днів на тиждень, тому доступ до нього може отримати кожен. Найважливіше завдання сайту — донести інформацію до користувачів і потенційних клієнтів. Сьогодні всі користуються Інтернетом, читають новини, спілкуються, грають і працюють в Інтернеті. Як наслідок, телефони, газети та телебачення занепали. Багато людей просто шукають всю необхідну інформацію в Інтернеті. Тому з власним сайтом організації будуть ближче до споживачів. І через постійно потреби та вимоги до зовнішнього вигляду та вмісту веб-сайту, організація має забезпечувати сучасність, комфортність простоту та швидкість. У цьому плані вирішальну роль відіграють системи управління контентом. Не завжди можливо швидко зв'язатися з розробником проекту, щоб переконатися, що інформація додається до ресурсу.

Крім того, самотійно компанії використовувати вміст веб-сайту та керувати ним — це забезпечення інформаційної безпеки. Використовуючи

систему управління контентом, користувач має можливість зосередитися на змісті інформації, яку він додає на сайт, а не налаштовувати програмний код.

**Актуальність теми** — використання засобів CMS останнім часом стало досить поширеним в Інтернеті. Особливістю цієї системи є те, що немає необхідності часто змінювати код сайту, всю інформацію можна налаштувати без програмування або знання HTML. Основне завдання такої системи — зібрати та об'єднати інформацію з різних джерел в єдине ціле на основі ролі і завдання. Доступ до цих джерел можна отримати як зсередини організації, так і за її межами. Крім того, система надає можливість взаємодії з різними співробітниками, проектами і робочими групами, а також створеними раніше базами даних, що робить процес пошуку та повторного використання комфортним і зручним. Ці веб-системи дозволяють наповнювати ваш сайт текстовим і мультимедійним вмістом з максимальним комфортом. І надає користувачам величезний набір інструментів для налаштування публікацій та інформації в Інтернеті. Такі системи відрізняються за складом і специфікою, але, як правило, містять дві основні частини. Першою важливою частиною є back-end, функція якого полягає в зберіганні та контролі інформації. Та не менш важлива front-end, яка полягає у візуальному сприйнятті інформації та взаємодії користувача з сайтом.

**Об'єктом досліджень** процеси застосування сучасних інтернет-технологій для накопичення інформації з різних джерел та відображення її в єдине ціле.

**Предмет дослідження** — системи керування контентом, методи та критерії їх варіантного аналізу.

**Мета дипломної роботи** — вдосконалення можливостей розробки веб-сайтів використовуючи систему керування WordPress.

Для досягнення цієї мети необхідно вирішити такі задачі:

- проаналізувати принципи розробки веб-сторінки;
- визначити CMS та їх структуру для реалізації веб-сторінок;

- розробити алгоритм використання середовища CMS при створенні web-сайту;
- розробити корпоративної інтернет сторінки кав'ярні з допомогою WordPress ;
- провести оцінку ефективності розробки програмного продукту із використанням системи керування вмістом.

Перевага використання CMS для розробки сайту полягає в тому, що вона не вимагає від користувачів знання HTML або основ програмування, а також дає можливість легко створювати нові проекти та оновлювати існуючі. Хоча існує велика різноманітність CMS, всі вони засновані на єдиному принципі роботи — шаблонах, тобто ви можете швидко змінити вигляд свого сайту, встановивши новий шаблон і використавши попередній вміст. На місце шаблону можна вставити різні матеріали та додаткові модулі, щоб зробити сайт більш зручним.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Всесвітня павутина складається з усіх загальнодоступних сайтів. Усі сторінки кожного сайту об'єднані кореневою адресою (те, що ми вводимо в адресному рядку браузера), темою, системою та дизайном. Кожна сторінка — це текстовий документ, який був написаний розробником за допомогою мов програмування. (HTML, PHP, Java тощо). Код сайту не компілюється, а інтерпретується, тобто інтерпретатор зчитує програму і, безпосередньо разом з операційною системою, виконує її. Мова програмування дозволяє редагувати текст сторінки, додавати посилання, вставляти зображення, аудіо та відео файли. Сторінки веб-сайтів можуть варіюватися від простих наборів інформації тексту та зображень до складних із великою кількістю функцій, керованих різними програмами.

### 1.1 Основні етапи створення веб-сайтів

Створенням та проектуванням Web-сайтів займається велика кількість розробників – програмістів, веб-дизайнерів, контент-мейкерів, SEO-менеджерів.

Дизайнери їх є два види, одні працюють над графічною частиною роботи втілюють в життя всі потреби замовника, а інші проводять аналіз та прототипи поведінки майбутніх користувачів, та тестують, для того, щоб досягти найбільш зрозумілого інтерфейсу та надати змогу користувачам інтуїтивно користуватись продуктом. А програмісти вже виконують весь back-end — це взаємодія сервера з клієнтом, архітектура додатку, створення платіжної системи та всього практичного функціоналу в продукті.

Також є такі люди, які просувають, оптимізують ваш сайт, розміщують його в онлайн-каталогах, розповсюджують рекламу на вашому сайті [7].

#### 1.1.1 Визначення проекту

Перший етап процесу розробки Web-сайту включає в собі збір

початкових даних. Опіраючись на опитувані замовника та будування технічного завдання потрібно виділяти:

- визначитись з контекстом, дізнатись про бізнес-цілі, сферу праці, культуру, технології, ресурси і обмеження;
- обрати типи документів і даних, інформаційне наповнення, структуру, методи управління та право власності;
- виконати аналіз цільової аудиторії, скласти основні завдання;

Базуючись на всіх даних, розробити архітектуру, концепт сайту та технічне завдання на яке будуть опиратись розробники. В собі воно повинно містити всі деталі: структуру Web-сайту, характеристики, адаптивність, гамму кольорів, додаткові модулі та бажанні нововведення. Замовник повинен узгодити завдання, тим самим структуру сайту.

Виходячи з технічного завдання виконується проектування, в яких розробляється функціональна структура сайту, навігація, визначаючи, які сторінки статичні, а які динамічні. Типи побудови сайту:

- статичні сторінки, які розробник не тільки робить, але і при необхідності сам вносить в них зміни;
- сайти, адміністратори яких, не маючи спеціальних знань, можуть самостійно вносити зміни в зміст сторінок, але не можуть змінити ні структуру, ні дизайн сайту;
- сайт, в якому замовник сам може додавати сторінки і розділи, змінювати дизайн.

### 1.1.2 Інформаційне наповнення сайту

Як тільки проект готовий і ви точно знаєте, яка навігація буде на сайті та повинно бути інформаційне наповнення сторінки, можна приступати до створення контенту. Все наповнення Web-сайту може бути представлений у вигляді тексту, таблиць, графіків, фотографій, аудіо та відеоматеріалів — все залежить від того, який матеріал надає клієнт та які в нього потреби [7].

Також, слід пам'ятати, що:

- контент без замовника створити майже нереально;
- контент повинен бути живим, його потрібно змінювати, додавати додаткові елементи, розширяти, навіть при тій умові, що діяльність залишилась незмінною.

### 1.1.3 Розробка веб-дизайну сайту

Проектування дизайну дуже клопітливий етап і разом з тим неймовірно цікавий та творчий процес. Web-дизайнери створюють візуальне оформлення — не звичайний шаблон, а повноцінний вигляд сторінок, консультуючись з замовником розробляється гарний, захоплюючий та легко запам'ятовуючий дизайн. Дизайнер не може знати продукт розробки так само, як знає її сам замовник, а веб-дизайн багато в чому відображає специфіку «фірми», для якої створюється сайт [8].

Також сайти мають тенденцію «старіти». З цього виходить, що для більш ефективної роботи потрібно змінювати візуальне оформлення сайту, не менше ніж один раз в пару років.

Переважно дизайн сайту повинен бути таким:

- стильним, відповідати іміджу «фірми»;
- гнучка навігація і керування;
- правильно скомпонованим;
- відповідним сфері «компанії»;
- не захаращеним, сторінки повинні завантажуватись швидко;
- правильно підібрана кольорова гамма та шрифти.

### 1.1.4 Написання програмного коду та верстка сайту

Написання коду — верстка сайту, розробка бази даних.

Без розробки програмного забезпечення Web-сайт більше схожий на форум, електронний довідник т.п. Розробники роблять сайт більш цікавим для клієнтів, та спрощують подальше користування ним. Макет сайту відноситься до процесу інтеграції тексту, графіки та програмних елементів в одне ціле [14].

Саме він надає сайту завершаючий вигляд, який клієнти побачать на сайті. Це створює особливу відповідальність для верстальників, оскільки потворні, неакуратні макети можуть зіпсувати зовнішній вигляд сайту і замовник поверне роботу на правки або ж повну переробку проекту.

Процес верстки один з головних етапів створення інтернет-ресурсу, так як від роботи верстальника залежить:

- швидкість завантаження веб-сайту;
- відповідність стандартам HTML;
- коректне відображення в браузері;
- відповідність вимогам пошукових систем;
- адаптивність під різні екрани користувача.

Валідна верстка — поняття верстки тісно переплітається з поняттям валідності. Дійсна верстка веб-сайту — написання HTML і CSS коду, який відповідає стандартам W3C і успішно проходить тест на валідаторі (валідний код — гарантія того, що верстальник не допустив логічних і синтаксичних помилок при програмуванні).

Необхідно враховувати нюанси, які покращують зовнішній вигляд сайту, на перший погляд дрібні деталі: заголовок під зображенням, правильне заповнення мета-тегів, встановлення фіксованого розміру для всіх елементів сторінки. Але вони грають велику роль в зручності користування сайтом.

## 1.2 Локальні сервери

Для розробки сайту на персональному комп'ютері необхідні локальні сервера. Найчастіше для створення сайтів використовуються готові HTMLшаблони, які можна редагувати в спеціальних програмах або за допомогою будь-якого текстового редактора. Величезна кількість сайтів створюються з використання систем керуванням контентом - CMS. Для коректної роботи необхідне середовище, в якій працюватимуть бази даних і серверна мови програмування. Локальний сервер використовується для:

- установки CMS;

- навчанню програмування і верстці сайтів;
- систем резервного копіювання;
- редагування веб-сайтів.

Для операційної системи Windows найпопулярнішими локальними серверами вважаються: Denwer, OpenServer, XAMPP.

Розглянемо плюси і мінуси кожної системи. Система Denver вийшла в 2002 році і набрала найбільшу популярність завдяки тому, що була дуже проста в установці, програма здійснювала всі настройки самостійно. У набір входили сервери: Apache, СУБД MySQL, PHP, Perl, sendmail, які постійно оновлювалися. Довгий час Denver був на першому місці серед локальних серверів і на разі теж має своїх користувачів. Для перенесення веб-сайту з одного локального сервера на іншій потрібні великі витрати ресурсів. Ще один мінус — це відсутність графічного інтерфейсу. Денвер погано сумісний з англійською ОС Windows, що вимагає ручного налаштування системи для коректної роботи.

Система OpenServer дуже проста в установці і видалення, працювати з нею досить легко за рахунок дружелюбного інтерфейсу. OpenServer включає в себе пакет веб-сервера Nginx — свого роду колега Apache. Nginx працює з запитами, які надходять від користувача і видає статичний контент. Apache в свою чергу видає динамічний контент. Nginx використовується на сторінках таких соціальних мереж як: vk.com і Facebook. Найкраще Nginx і Apache проявляють себе при одночасному використанні. OpenServer включає в себе кілька HTTP-модулів, три СУБД і велика кількість PHP-модулів. Незважаючи на це перемикання між модулями здійснюється без будь-яких складнощів у оператора.

OpenServer в разі швидше за Денвер, не дивлячись на великий розмір. Для установки системи потрібна велика кількість пам'яті і мінімум 200 мегабайт оперативної пам'яті. На відміну від попередників система не захищає системні файли. Також систему можна встановлювати на відповідному для розробника девайсі. Великим плюсом є можливість системи



самостійно встановлювати і налаштовувати додаткові елементи, що є перевагою для новачків. В системі XAMPP запропонована власна альтернатива на відміну від Денвера і OS. система є кросплатформова, без проблем працює на різних операційних системах. Замість MySQL використовується MariaDB. MariaDB — це одне з відгалужень MySQL. Підтримує мову Perl. XAMPP краще Денвера і не поступається OpenServer.

### 1.3 Web-сервери

У більшості випадків неможливо уникнути прямого контакту з сервером, навіть якщо мова йде про просту завантаження файлів. З цієї причини всі дизайнери повинні володіти базовими знаннями про серверах і їх роботі. Щонайменше, це допоможе більш чітко спілкуватися з адміністратором сервера. Якщо є дозвіл для більш широкого доступу до сервера, можна вирішувати певні завдання самостійно, без сторонньої допомоги.

Сервер — це будь-яке управляє комп'ютером програмне забезпечення, яке дає йому можливість виконувати запити на документи або інші дані. Програми, які запитують і відображають документи (такі як браузер), називаються клієнтами. Терміни "на стороні сервера" і "на стороні клієнта", що використовуються, наприклад, при роботі з картами-зображеннями, відносяться до тієї машини, яка керує процесом. Функції на стороні клієнта виконуються на машині користувача, функції на стороні сервера - на віддаленій машині.

Web-сервери відповідають на запити браузерів (клієнтських програм), знаходять файлами (або виконують сценарій CGI) і повертають документ або результати сценарію. Web-браузери і сервери спілкуються за протоколом Hypertext Transfer Protocol (HTTP, протокол передачі гіпертексту).

Більшість серверів працюють на платформі Unix. Саме тому у світі Web як і раніше використовується термінологія системи Unix. У процесі роботи знадобиться вивчити кілька Unix-команд. Однак відсоток серверів Windows

NT, Windows 95 і навіть MacOS постійно збільшується. Деякі серверні пакети пропонують графічний інтерфейс в якості альтернативи управління з командного рядка Unix.

Ось деякі відомі сервери: NCSA Server, Apache, CERN, Netscape Servers, Internet Information Server (IIS).

Сьогодні більшість серверів (приблизно 70%) працюють на Apache або його попереднику NCSA. Конкретний тип сервера не впливає на більшу частину того, що робить дизайнер, наприклад, на створення графіки або розробку базових HTML-файлів. Звичайно, він буде впливати на більш досконалі методи створення Web-сайтів, такі як Server Side Includes (серверні включення), додавання MIME та Web-сторінки, керовані базами даних.

Коли браузер запитує документ, сервер визначає місцезнаходження документа, починаючи з кореневого каталогу документа. Це каталог, який був налаштований для зберігання всіх документів, спільно використовуваних допомогою Web. Він не обов'язково з'являється в URL, який вказує на документ, тому важливо знати, який каталог є кореневим при завантаженні файлів.

Прямий слеш (/) в кінці URL означає, що URL звертається до каталогу, а не до файлу. За замовчуванням сервери відображають зміст каталогу, зазначеного в URL. Більшість серверів, однак, конфігуровано так, щоб відображати особливий файл замість списку каталогу, цей файл називається індексним. Індексні файли звичайно мають ім'я index.html, але на деяких серверах вони можуть називатися welcome.html або default.html.

Якщо сервер налаштований на знаходження індексного файлу і не виявляє його, замість нього може бути відображено зміст каталогу, але це робить файли уразливими для сторонніх. З цієї причини непогано завжди називати одну зі сторінок (зазвичай головну) у кожному каталозі index.html (або іншим умовленого ім'ям).

Як тільки сервер визначає місце розташування файлу, він посилає вміст цього файлу назад браузеру разом з деякими заголовками відгуку HTTP

(response headers). Ці заголовки забезпечують браузер інформацією про прибуває файлі, включаючи його тип даних (також відомий як "тип вмісту" або "тип MIME"). Зазвичай сервер визначає формат з розширення файлу, наприклад, файл із розширенням.gif розпізнається як файл зображення.

Браузер читає інформацію в заголовку і визначає, що робити з файлом. Він може відобразити файл у вікні або запустити відповідний допоміжне або вбудоване додаток (plug - ins).

Замість того щоб звертатися до файлів HTML, URL може зажадати запустити програму CGI. CGI розшифровується як Common Gateway Interface (загальний шлюзовий інтерфейс). Він дозволяє Web-серверу спілкуватися з іншими програмами (сценаріями CGI), які працюють на сервері. Сценарії CGI зазвичай пишуть на мовах Perl, C або C + +.

Сценарії CGI використовують для виконання різноманітних функцій, таких як пошук, управління картами-зображеннями на стороні сервера, ігри. Однак найбільш типове використання сценаріїв - обробка форм.

Більшість адміністраторів серверів дотримуються правила зберігати сценарії CGI в спеціальному каталозі, названому cgi - bin (скорочено від CGI - binaries). Коли вони зберігаються в одному каталозі, адміністраторам простіше управляти сервером і забезпечувати його безпеку. Якщо сценарій CGI запитується браузером, сервер виконує функцію і повертає браузеру динамічний вміст.

SSI - Server Side Includes або, по-російськи, - включення на стороні сервера. SSI - це директиви, що вставляються прямо в HTML-код і службовці для передачі вказівок Web - сервера. Зустрічаючи такі директиви, які називаються SSI-вставками, Web-сервер інтерпретує їх і виконує відповідні дії. Наприклад: вставка HTML-фрагмента з іншого файлу, динамічне формування сторінок в залежності від деяких змінних (наприклад, типу браузера) та інші не менш приємні речі.

Переваги SSI виявляються, коли нам потрібно підтримувати досить великий за обсягом сайт, що має певну структуру і повторювані елементи коду

на всіх сторінках. При застосуванні серверних включень сайт зручно розглядати як що складається з окремих блоків, кожен з яких відповідає за свою частину сторінки. Ці блоки практично незмінні і повторюються від сторінки до сторінки. У ці блоки можна винести такі елементи сторінки як: головне меню, рекламні вставки, що повторюються елементи оформлення сторіночок і т.д. Фізично ці блоки представляють собою просто HTML-файли, що містять частину коду, потрібну для виконання їх завдання.

Для того, щоб сервер знав, що сторінка не звичайна, а містить SSI-директиви, вона має спеціальне розширення: \*. shtml або \*. shtm, наявність якого і змушує Web-сервер попередньо обробляти сторінки. Взагалі-то, розширення може бути будь-яке в залежності від його конфігурації, але в основному застосовується саме \*. shtml.

Повна сторінка формується на льоту, збираючи код сторінки з таких ось блоків. Для того, щоб вказати серверу, який блок потрібно вставити і в якому місці сторінки, використовується спеціальна форма запису у вигляді коментаря.

Перша перевага SSI з точки зору дизайнера полягає в тому, що при такому підході Web-майстра, який займається підтримкою сайту, можна не боятися випадково зіпсувати дизайн. Елементи складної верстки приховані за рахунок використання SSI, і підтримка вмісту сторінок стає набагато більш легким і приємним справою.

Друга, не менш важлива перевага, - це можливість миттєвої заміни дизайну сайту, що не вимагає перероблення сторінок з інформаційним змістом сайту. Для зміни дизайну досить переписати SSI-вставки, що формують зовнішній вигляд сайту.

На комп'ютерах IBM PC і Macintosh більшість функцій можна виконати, використовуючи інструментарій з графічним інтерфейсом. Тим не менш, іноді неможливо знайти заміну для старої сесії Telnet. Telnet - протокол "емуляції терміналу", що дозволяє вам реєструватися у іншій комп'ютерній системі або мережі, такий як Internet. Цей термін також використовують стосовно до будь-

якого додатка, службовцю для спілкування з використанням протоколу Telnet. Програма Telnet надасть текстове термінальне вікно іншої системи, в якому можна вводити інструкції в командному рядку.

Коли за допомогою програми Telnet відкривається сеанс зв'язку з віддаленим сервером, з'являється запрошення зареєструватися в системі. Перш ніж отримати доступ до сервера, системному адміністраторові потрібно встановити для користувача обліковий запис (account) Unix.

При успішній реєстрації, ви отримаєте підказку системи Unix%, (або іноді #) залежно від різновиду Unix, на якій працює сервер. Це знак, що система готова. З цього моменту використовується програма під назвою shell. Вона інтерпретує друкуються вами команди і викликає запитувані програми. Переміщення в домашній каталог буде автоматичним.

Щоб закінчити сесію Telnet достатньо набрати «logout» або «exit».

Оскільки Web є породженням оточення Unix, він слідує багатьом його угодам. Наприклад, щоб в гіперпосиланні правильно вказати шлях, URL, потрібне розуміння організації каталогів на платформі Unix.

Каталоги ("місце зберігання файлів") організовані у вигляді ієрархічної структури. Самий верхній каталог відомий як кореневий і позначається прямим слешем (/). Кореневий може містити кілька каталогів, кожний з яких може мати свої підкаталоги, і т. д. Кажуть, підкаталог є "нащадком" каталогу, до складу якого він входить (останній називають "батьком").

Ім'я шляху (path name) - це нотація, яка використовується для вказівки конкретного файлу або каталогу. Воно вказує, який шлях треба пройти, щоб дістатися до бажаної мети. Є два типи шляхів: абсолютний (з кореневого каталогу) і відносний (з каталогу в якому знаходишся).

Найбільш частою транзакцією, з якою Web-дизайнер звертається до сервера, є завантаження на сервер HTML-документів, зображень або мультимедійних файлів. По мережі файли передаються між комп'ютерами за допомогою методу, званого FTP (File Transfer Protocol, протокол передачі файлів). При роботі на Unix в сесії Telnet можна запустити програму ftp і

передавати файли з величезним числом аргументів командного рядка.

Незалежно від того, який інструментарій використовується, базові принципи та процеси залишаються незмінними.

Двійкові файли складаються з компільованих даних (одиниць і нулів), їх прикладами є виконувани програми, зображення, фільми і т.д. Деякі програми розглядають двійковий режим як "необроблені дані" ("raw data") або "зображення" ("Image"). Всі графічні (. Gif або. Jpeg) і мультимедійні файли повинні передаватися як виконавчі або "Raw Data". У Fetch (MacOS) ви можете виявити параметр MacBinary, який передає файл повністю, разом з гілкою ресурсів (частина файлу, що містить піктограми робочого столу і інші специфічні для комп'ютера Macintosh дані). Цей варіант слід використовувати тільки при передачі з одного комп'ютера Macintosh на інший. Гілка ресурсів відділяється від мультимедійних файлів, створених на комп'ютерах Macintosh, при передачі в двійковому режимі.

Деякі FTP-програми мають також параметр Auto, який дозволяє вам передавати повністю весь каталог, що містить файли обох типів. Програма перевіряє кожний файл і визначає, чи слід передавати його в текстовому або двійковому режимі. Ця функція не у всіх програмах надійна на 100%, тому треба використовувати її з обережністю, поки не буде впевненості, що результат правильний.

Передача файлів на сервер — стандартний протокол FTP використовує терміни "помістити" ("put") для позначення передачі файлів з комп'ютера користувача на сервер і "отримати" ("get") для позначення завантаження файлу з сервера на комп'ютер, тому вони можуть використовуватися і в програмі FTP. Одночасно можна завантажувати декілька файлів.

Роз'єднання — коли передача закінчена, зв'язок з сервером завершується. Перед цим можна протестувати передані в браузер файли, щоб переконатися, що передача пройшла вдало. При передачі файлів на Web-сервер, потрібно переконатися, що всі права доступу до файлів встановлені таким чином, що кожен може їх прочитати. Право доступу контролює, хто

може читати, записувати (редагувати) або виконувати файл (якщо це програма). Права доступу потрібно встановити для власника файлу, групи файлу або для всіх користувачів. Встановлювати права доступу може тільки автор файлу.

Деякі FTP-програми дозволяють встановлювати право доступу за замовчуванням у діалоговому вікні. Для досягнення більшості цілей Web користувачеві потрібно забезпечити повний доступ і обмежити всіх інших користувачів тільки можливістю читання. Може знадобитися згода адміністратора сервера з такими установками.

Сервери додають до кожного документа заголовок, який повідомляє браузеру, який тип файлу він відправляє. Грунтуючись на цій інформації, браузер визначає, що робити з файлом: або відобразити його вміст у вікні, або запустити відповідний вбудоване або допоміжне додаток.

Система для повідомлення мультимедійних типів файлів дуже нагадує MIME (Multipurpose Internet Mail Extension, багатоцільові розширення поштової служби в Інтернеті), яка була спочатку розроблена для вкладень в листи електронної пошти. Щоб успішно повідомляти тип файлу браузеру, потрібно щоб сервер був налаштований на розпізнавання будь-якого MIME-типу. Якщо потрібно доставляти дані, що виходять за рамки стандартних HTML-файлів та зображень (наприклад, відео Shockwave Flash або аудіофайли), слід уточнити у адміністратора сервера, чи здатний сервер підтримувати цей MIME-тип [13]. Більшість поширених форматів вбудовані в поточні версії програмного забезпечення сервера. Якщо ні, то адміністратор може легко їх встановити, слід лише надати йому необхідну інформацію.

Точний синтаксис для конфігурування MIME-типів варіюється в різних серверних програмах. Тим не менш, всім потрібна одна і та ж базова інформація: тип, підтип і розширення. Типи - це найбільш загальні категорії файлів. Вони включають текст, зображення, аудіо, відео, додаток і т.д. У кожній категорії є ряд підтипів. Наприклад, файловий тип image (зображення) включає підтипи gif, jpeg і т.д. Розширення файлу використовується сервером

для визначення типу файлу і його підтипу. Не всі розширення стандартизовані.

#### 1.4 Використання CMS для створення веб-сайтів

У перекладі з англійської CMS (Content Management System) — це система керування контентом, комп'ютерна програма або інформаційна система, яка використовується для організації та підтримки процесу спільного створення, управління та редагування вмісту веб-сайту.

Особливість системи полягає в тому, що вона не потребує участі розробника проекту, а також дає змогу змінювати контент, добавляти нові блоки та модулі адміністраторам, які можуть не володіти основами програмування чи навичками в HTML.

Завдання даної системи — з'єднати інформацію з різноманітних джерел в одне ціле. Доступ можна отримати з будь-якої точки, для різних груп робітників можна надавати окремі привілеї. CMS дає унікальну можливість спільній роботі між розробниками, а також створеними раніше базами знань і даних, що робить процес пошуку та повторного використання комфортним, зручним та знайомим [1].

За складом і специфікою такі системи є різними, но схожі між собою за двома частинами. Перша складова частина — це back-end, чия функція полягає у зберіганні та контролі над функціональністю інформації. Друга — це front-end. Ця частина системи забезпечує візуальну взаємодію проекту з користувачем.

На рис. 1.1 показана типова архітектура інтернет-сайту, створеного за допомогою CMS:

Системи керування веб-сайтами як за правило зроблені для роботи в певному програмному середовищі. Наприклад, система MediaWiki, розроблена мовою програмування PHP застосовується для зберігання вмісту та налаштування в базах даних типу PostgreSQL чи MySQL. Для коректної роботи повинен бути встановлений веб-сервер (Apache, IIS або інші), підтримка систем управління базами даних і MySQL або PostgreSQL. Така



вимога дуже характерна для відкритої CMS [1].

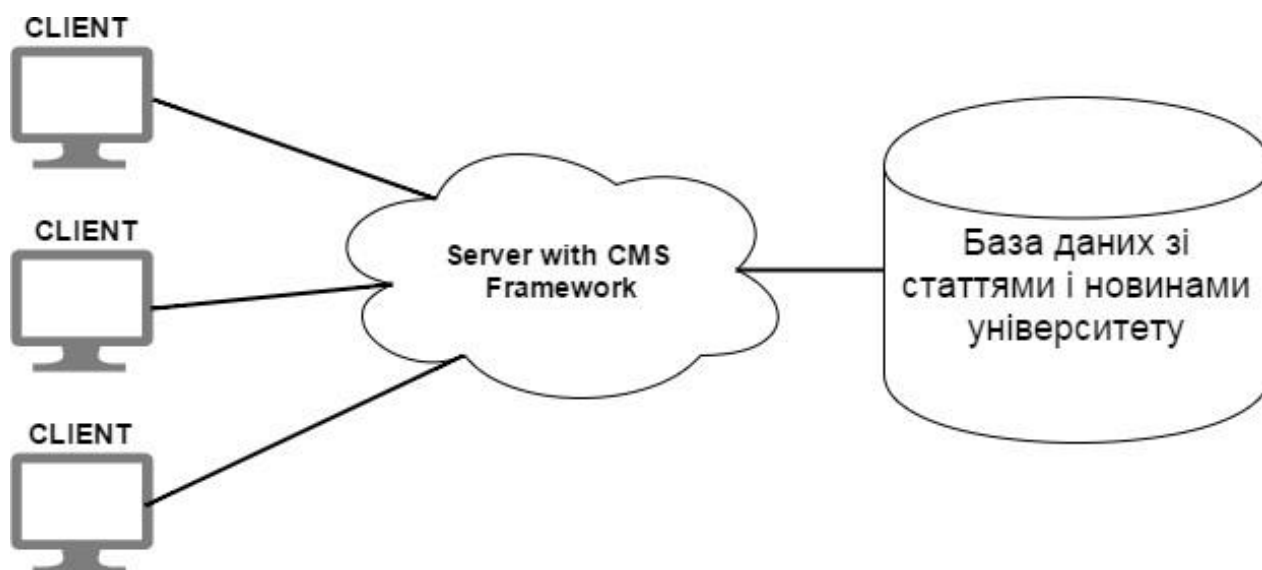


Рисунок 1.1 — Типова архітектура веб-сайту створеного за допомогою CMS

В даний час, через особливості розвитку сучасної мережі, система управління контентом є актуальною. Не завжди вдається при необхідності звернутись до автору сайту, щоб вчасно змінити наповнення або додати нову структуру. Зокрема коли корпорація може самостійно змінювати контент та здійснювати управління Web-сайтом надає гарантію безпеки, та запобігає витоку інформації. З використанням систем керування вмістом адміністратор може зосередитись на наповненні, а не налаштуванню програмного коду [1].

Ця веб-система дозволяє легко заповнювати ваш сайт текстовими, відео, звуковими, анімаційними вставками. Крім того, адміністраторам надається величезний набір інструментів для введення змін та оголошення інформації в мережі. Хоча існує велика різноманітність CMS, всі вони засновані на єдиному принципі роботи. Системи управління контентом розділяють контент і дизайн. Це необхідно для спрощення роботи з Web-сайтом, оскільки інформація, що міститься на сайті, буде періодично змінюватися залишаючи один й той самий дизайн. Система зберігає в собі створені шаблони, щоб додавати контент, в яких не складно розібратись. Шаблони допомагають обрати візуальний вигляд веб-сторінки в максимально короткий відрізок часу. Таким чином

зменшується час оновлення інформаційного наповнення, та витрати сил розробників. Система керування контентом весь час знаходиться в цілодобовому доступі на сервері, де збережена велика кількість інформації. За потреби зміни вмісту є можливість підключитись з будь-якої точки доступу до інтернету [2].

CMS добре підходить для швидкого створення різноманітних сайтів. Хоча базуючись на цьому методі розробки все одно потрібні певні знання в створенні веб-сторінок.

За допомогою модулів можна зручно налаштовувати візуальне оформлення всього web-сайту, створюючи схожі або абсолютно індивідуальні дизайни.

Системи управління можна розділити на кілька основних різновидів:

— система, яка створює сторінки безпосередньо у відгук на запит адміністратора відбувається через обмін даних між клієнтом та сервером, дана система користується попитом, но все ж таки в ній немало недоліків, найбільший це навантаження на системні ресурси;

— існує тип, що дозволяє створювати сторінки прямо під час налаштування його найлегше освоїти, він застосовує, як згадувалось вище, шаблони його інтерактивність — єдиний недолік, попри це зручність та комфортність користування перекриває його;

— гібридна система — включає в собі перший та другий метод і має широкий спектр функціональних можливостей та інтерактивності, але для освоєння займає багато часу, даний тип через свою змішаність стає найбільш вартісним.

Розробники по всьому світу намагаються створити свої власні CMS на основі фреймворкових інструментів, вдосконалюючи свої знання, але це досить трудомісткий процес, і такі проекти часто не доводять до кінця, або вже наявні системи в рази краще та підходять до різних специфіки розробки.

В CMS системах можна виділити кілька характерних особливостей:

— детермінованість внутрішньої архітектури у системі CMS

внутрішня архітектура розвиває механізм абстракції, незалежний від модуля генерації CMS — означає, що підтримувати проекти на основі CMS набагато простіше;

— шаблони CMS, котрі легко створювати та інтегрувати їх в презентації (дизайн), не впливаючи на ядро програмного забезпечення або вміст сайту розробки [9];

— CMS проста в використанні, вона може бути системою з будь-яким, досить розширюваним і настроюваним набором функцій, а також дистрибутивом, який простий у використанні непрофесійними програмістами.

Розглядаючи використання CMS на веб-сайтах, можна сказати, що характеристика системи полягає в тому, що вона не вимагає участі автора веб-сайту та не вимагає від користувача знання HTML чи оволодіння основами програмування.

Хоча існує велика різноманітність CMS, але всі вони засновані на єдиному принципі роботи. Системи чудово підходять для створення різних типів веб-сайтів, що чудово. Ви також можете використовувати наявні шаблони для графічного представлення вашого сайту, а також можете створити свій власний індивідуальний дизайн.

## 2 ВИЗНАЧЕННЯ ОСНОВНИХ ХАРАКТЕРИСТИК CMS ТА ВИБІР ШАБЛОНІВ ДЛЯ ПРОЕКТУ САЙТА

Перед розробкою сайту, Web-програміст повинен обрати відповідну CMS під яку буде створюватись проект. Наразі ринок безкоштовних систем величезний, найбільш популярні системи так як Drupal, WordPress та Joomla!.

### 2.1 Загальний огляд систем розробки

Система керування сайтом Drupal — це система управління вмістом (CMS), написана на PHP, яка використовує реляційну базу даних для зберігання даних (з підтримкою MySQL, PostgreSQL). Якщо наймати людину з великим досвідом розробки, то Drupal є чудовим варіантом. Це особливо добре, якщо ви прагнете створити сайт із індивідуальними потребами, який повинен обробляти велику кількість даних. Drupal — безкоштовне програмне забезпечення, захищене ліцензією GPL. На рис. 2.1 показано, як виглядає система [6].

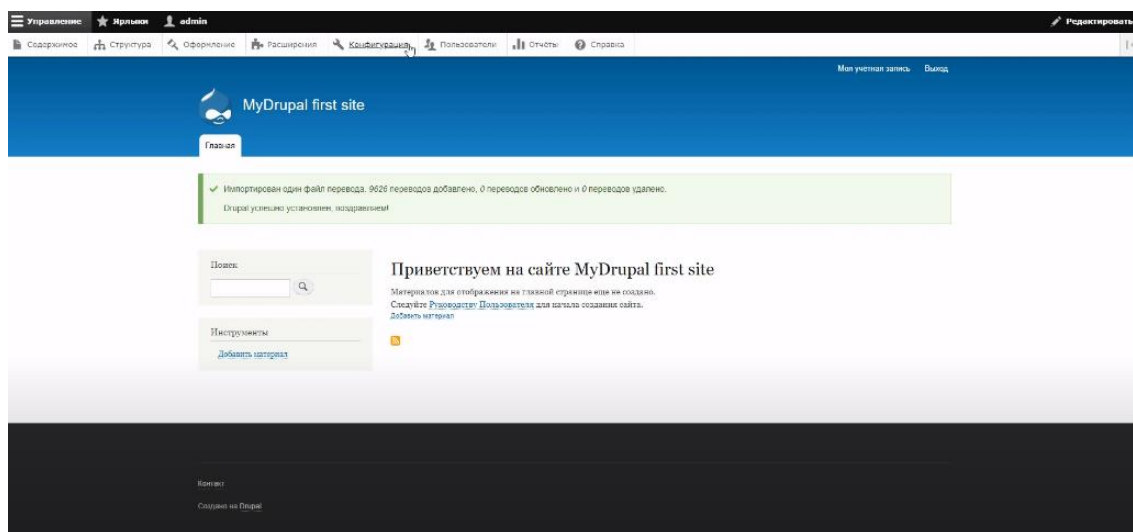


Рисунок 2.1 — Вигляд системи Drupal

Близько 7% користувачів віддають перевагу Drupal. Він має просте керування користувачами завдяки вбудованій системі, де ви можете створювати нові ролі та вказувати їхні дозволи. Розробники люблять його

всеохоплюючі функції та дуже простий інтерфейс, що дозволяє створювати веб-сайти будь-якої складності. Як і інші «чудові» інструменти, Drupal вимагає деяких технічних навичок, на відмінну від CMS Joomla або WordPress [15].

Система управління контентом WordPress — це найпопулярніша система керування вмістом із відкритим кодом (CMS), що поширюється під GNU GPL. Написано на PHP, використовує MySQL як базу даних. Діапазон використання величезний — система пропонує вам гнучкість і свободу створення веб-сайтів будь-якого типу ( інтернет-магазин , сайт аукціону , сайт для членства тощо). Легко створювати зручні для SEO URL-адреси, категорії та теги для своїх публікацій. Ви також можете вибрати один із багатьох плагінів SEO , які допоможуть зробити сайт більш популярним. Вбудована система «Тем», «Плагінів», «Рубрик», «Записів» та «Коментарів» разом з вдалою архітектурою дозволяють розробити практично будь-який проект. На рис. 2.2 показано, як виглядає система WordPress.

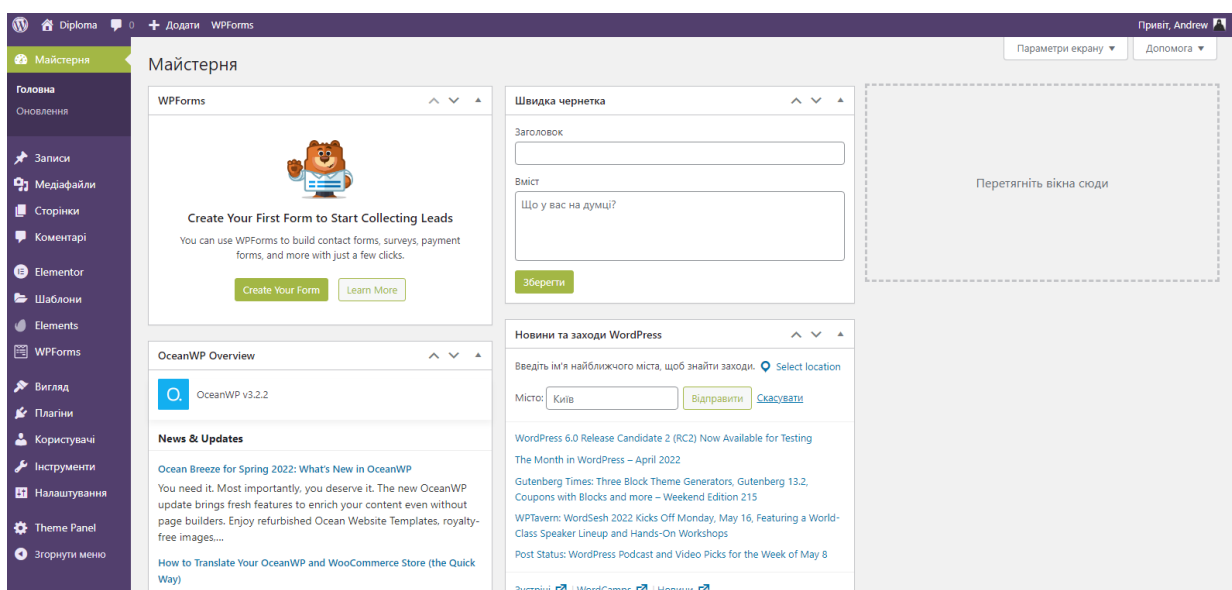


Рисунок 2.2 — Вигляд системи WordPress

Навколо цієї CMS існує величезна спільнота підтримки, оскільки ця система з відкритим кодом. Є можливість приєднатись до таких груп, як: WPBeginner, Engage у Facebook, щоб отримати відповідь на будь-яке запитання та допомогу з різноманітними проблемами.

WordPress вперше позиціонував себе як інноваційну платформу для ведення блогів з високим рівнем зручності та простоти використання. Але еволюція можливостей системи зробила її дуже популярною і серед інших форматів веб-сайтів. Послуги з розробки WordPress сьогодні користуються великим попитом в індустрії веб-дизайну. Ця система має найрізноманітніше використання, від особистих блогів до комерційних веб-сайтів [6].

Система керування Joomla! — ця система, розроблена на мові програмування PHP та JavaScript, з використанням MySQL як базу даних. Вперше він був випущений у 2005 році, тому, як і WordPress, він працює роками. Joomla! має безліч функцій, і багато веб-хостинги пропонують установку в 1 клік. Однак це дійсно хороша платформа CMS для розробників і досвідчених розробників веб-сайтів, тому для початківців це не такий добрий варіант. Система являє собою вільне програмне забезпечення, яке поширюється під ліцензією GNU GPL. На рис. 2.3 показано вигляд панелі керування Joomla!.

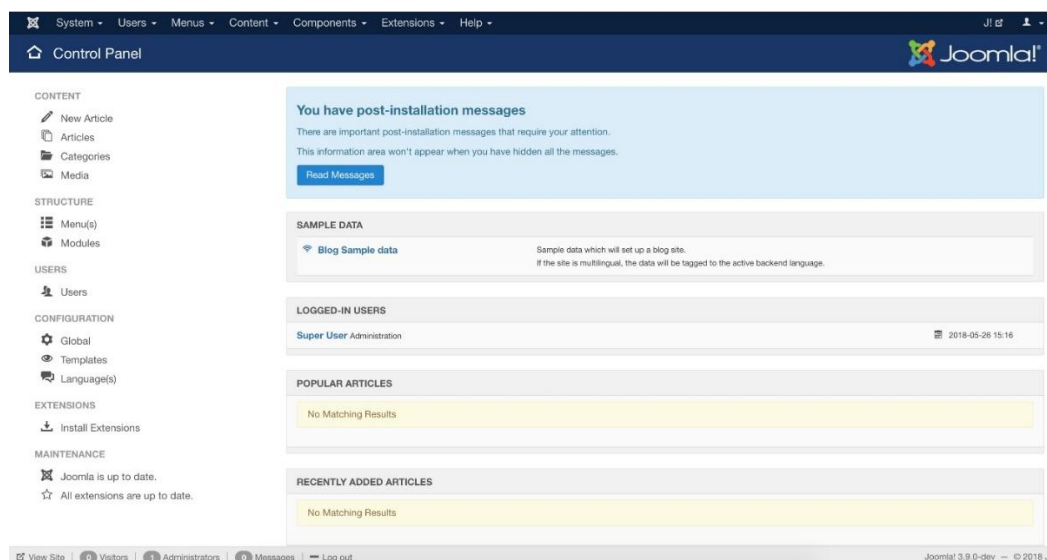


Рисунок 2.3 — Вигляд панелі керування Joomla!

9% користувачів використовують Joomla!. Він схожий по потужності на Drupal та простотою WordPress. Надає велику гнучкість і безліч опцій. Це хороший вибір для створення складного або сайту на замовлення. За

допомогою розширень має здатність запускати магазини електронної комерції.

CMS Joomla! Включає різноманітний набір інструментів для створення веб-сайтів. Особливе в цій системі, що має велику значимість є використання мінімального набору інструментів в встановленні по замовчуванню, який доповнюється за потреби. Це зменшує перевантаження непотрібними елементами в адмін-панелі та зменшує навантаження на сервер і економить місце для даних. [6].

Joomla! Інтерфейс системи, дозволяє відображати розділи інтерфейсу та адміністратора будь-якою мовою.

## 2.2 Пошук та використання шаблонів при створенні веб-сайтів

Шаблон — це вже заздалегідь, спроектовані командами розробників Joomla, WordPress, Drupal т.д . Завдання адміністратора сайту, це просто ввести текст, встановити медіа файли на сайт з використанням вже створеної адміністративної панелі.

Переваги сайту:

— швидкість створення сайту, не велика ціна.

Недоліки сайту:

— структуру такого сайту розробникам додати чи видалити елементи стає проблематично, оскільки код шаблону доволі великий і чутливий до змін [9].

В теперішній час відомо декілька типів шаблонів для сайтів:

— шаблони для CMS;

— PSD шаблони;

— HTML шаблони.

В всесвітній павутині шаблонів наразі налічує необмежену безліч, але до відбору їх, потрібно підходити обачно та з обережністю [9]. Вибір шаблону базується на трьох правилах:

— необхідно перевірити дати створення та оновлення;

- завантажені файли необхідно перевірити на наявність шкідливих файлів;
- перевірка посилань відразу після завантаження.

Використовувати шаблони має змогу будь-яка особа або організація може використовувати веб-шаблони для розробки власного Web-сайту.

Після придбання або завантаження шаблону користувач може коригувати весь контент, що наявний в шаблоні веб-сторінки, своєю особистою інформацією, інформацією про організацію чи продукт [9].

Шаблони можуть бути використані для:

- блогу, постити особисту інформацію чи транслювати свою активність;
- інтернет-магазини, дистанційний продаж товарів;
- висвітлювати всю інформацію корпорації, надавати інформацію робітникам або ж клієнтам;
- відображення фотогалереї, фото робіт, власних досягнень, музичних файлів і т.п.;
- налаштування зони входу для приватної частини веб-сайту.

Є можливість повторного використання. При потребі не кожний може собі дозволити наймати розробників для внесення невеликих правок. Для цього дизайнери створюють шаблони спеціально для повторного використання технічно не підготовлених осіб.

Крім того, вже створені веб-шаблони відіграють важливу роль навіть для висококваліфікованих і розробників з великим досвідом, для яких є важливим простота та готовність веб-рішення через які не прийдеться витратити час марно. Такі стандартні веб-шаблони іноді доступні безкоштовно і їх можна легко адаптувати до певних вимог.

Шаблони WordPress — набір файлів, які в сукупності дозволяють створювати візуальне оформлення та контент Web-сайту. Ці файли називаються шаблонами. Теми дають змогу змінити інтерфейс сайту, не викликаючи помилок в коді системи. Теми можуть містити файли шаблонів,



зображення (\*.jpg, \*.gif), каскадні таблиці стилів (\*.css), користувацькі сторінки та будь-які інші файли коду php (\*.php).

Сайт «WordPress Theme Directory» — найбільший каталог тем в якому можна обрати шаблон для будь-якої теми сайт. На сайті є можливість пошуку за параметрами теми, а також можна заздалегідь подивитись на те як вона буде виглядати в кінцевому варіанті [6].

Використання теми WordPress включає в себе стандартну тему (WordPress Twenty Sixteen theme), та ще три різних але схожих варіанти теми, як відразу є можливість обрати після встановлення. Обирати теми потрібно в консолі CMS. Теми, які додаються в відповідну директорію на сервері, будуть доступні в розділі Консоль> Зовнішній вигляд> Теми в списку "Доступні теми".

Для додавання обраної теми в WordPress, треба виконати наступні основні кроки:

- скачати архів «Тема» і розпакувати файли, які він містить. Дотримуватись рекомендацій щодо встановлення та зберегти всю структуру в архіві при розпакуванні.

- для отримання доступу до серверу потрібно застосовувати FTP-клієнт, щоб отримати доступ до веб-сервера, та створити каталог задля збереження теми в каталозі wp-content/themes, представленої WordPress.

Для вибору застосованої теми web-сайту:

- увійти в адмін-панель WordPress;
- вибрати зовнішній вигляд панелі, а потім теми;
- у розділі «Доступні теми», натиснути на назву теми;
- попередній перегляд теми буде показано;
- для активації теми необхідно натиснути «Активувати "Theme Name"» посилання у правому верхньому куті.

Шаблон в Joomla! — це основна ознака дизайну майбутнього сайту. В порівнянні з іншими CMS ця система, проста за своєю структурою, що полегшує процес їх розробки і редагування.

В структуру шаблонів Joomla! входить:

- файл `index.php`, який позначає базову структуру сторінки веб-сайту;
- файл CSS-стилів, який дозволяє регулювати оформлення блоків сторінки Joomla! (Кольору, розміри шрифтів тощо);
- файл для завантаження шаблону через адмін-панель

Наразі є 2 основних види шаблонів Joomla!, варіанти дизайну (Site Template) та шаблон адмін-панелі (Backend Template). Ця CMS має змогу встановлювати окремі шаблони для потрібних сторінок. Наприклад, маючи на сайті блог і магазин, можна оформити їх як два абсолютно різні сайти, всього лише оперуючи шаблонами Joomla! [10].

Весь вигляд Web-сайту на Joomla! залежить суто від встановленого шаблону. В інтернеті можна знайти велику кількість сайтів для безкоштовного або платного встановлення шаблону для будь-якої тематики. До певних тем можна придбати додатковий доступ на певний період на місяць чи то на рік [10].

### 2.3 Основні критерії порівняння

При порівнянні систем керування вмістом слід зосереджуватись на головних аспектах, які мають значення в плані зручності та області застосування.

Вигідно інсталиувати CMS, зі встановлення по замовчуванню мають змогу налаштовувати веб-інтерфейс при потребі. При завантаженні визначаються параметри з'єднання з базою даних, дані для входу в адмін-панель, ім'я сайту. Також можна обрати мову панелі керування та вказати електронну пошту розробника для зворотного зв'язку.

Безпека Drupal вважається одним з найбезпечніших безкоштовних движків. Модулі пишуться тільки самими розробниками движка, або висококваліфікованими програмістами, через це всі модулі стають ефективними, але їхня кількість в рази менше ніж в інших системах керування

вмістом.

Для системи WordPress існує величезна кількість модулів, так як створювати і додавати їх може будь-який розробник. В цьому є плюси та мінуси, перевага полягає в тому, що можна знайти модуль для вирішення всіх задач, но недоліком стає, що якість наявного плагіну може призвести до злому і весь сайт «впаде». Проте розробники потурбувались про такі проблеми і можна з неймовірною швидкістю виправляти діри, якими можуть скористатись зловмисники.

Сам по собі движок Joomla! захищений добре. Однак велика кількість модулів написано не практично в певній мірі не правильно. В результаті через ці проблеми може статись виток інформації та злом всього web-сайту.

Початкові можливості офіційної збірки містить чималу кількість плагінів для створення сайту будь-якої спрямованості. На «пустому» Drupal виконуються форуми, що розраховані на велику кількість користувачів, блоги і т.д.

Можливості WordPress без використання плагінів досить невеликі після встановлення за замовчуванням. Максимум це сайт-портфоліо, власний блог.

В момент завантаження Joomla!, відразу можна без використання модулів створити звичайний сайт, який буде функціонувати та коректно працювати. Багатомовність CMS навіть Drupal без нововведень, можна створити багатомовний сайт. Однак управління в цій системі доволі важке та незрозуміле.

У WordPress кількість плагінів величезна. З їх допомогою створюється максимально унікальні сайти. Але якщо ними перевантажити Web-сайт, то він буде працювати значно повільніше, та займати багато місця.

У Joomla! серед компонентів варто відзначити Joom!Fish. Компонент безкоштовний, і може перекладати статті, метатеги і компоненти на безліч мов.

Керування контентом в CMS Drupal досить зручне управління розміщенням матеріалів на сайті. Но в порівнянні програє іншим движкам. Drupal має таксономію — система термінології для управління класами та

візуальним зображенням матеріалів.

Можливостей для управління контентом у WordPress не багато. Хороших повністю безкоштовних модулів практично не знайти, но за невелику плату відкривається маса можливостей для управління всією CMS, можна змінювати буквально все.

В Joomla є вставлені засоби для створення категорій і контент-розділів. Можливо публікувати інформацію лише для зареєстрованих або для всіх клієнтів.

Розміщення блоків в Drupal не прив'язані їх можна перетягувати в різні позиції, при доступу теми, не більше. Однак є можливість стилізації їх виводу для різних типів блоків.

В WP виведення інформації за замочуванням закріплено на певних позиціях без доступу до змін. Тільки при встановленні певних модулів ця здатність з'являється.

Joomla в цій системі окрім заданих позицій, можна самостійно позиціонувати блоки за допомогою спец-тега в матеріалах та інших блоках [11].

Редактор тексту у системі Drupal візуально оформленого текстового редактора. Для редагування інформації необхідно використовувати мову гіпертекстової розмітки та мову стилю сторінок, що потребує вже певного досвіду та навичок. Редактор тексту в Drupal показано на рис. 2.4.

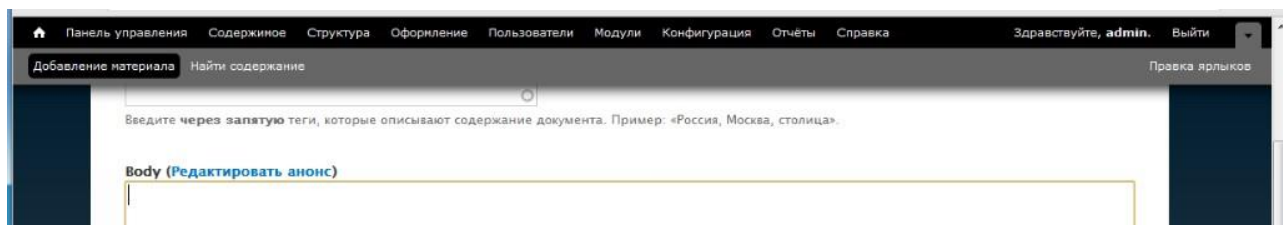


Рисунок 2.4 — Редактор текста в Drupal

Стандартний візуальний редактор в WordPress (рис. 2.5). В собі містить всі головні функції для зручного налаштування тексту та медіа файлів

Найкращий вибір для мало знайомих з HTML і CSS.

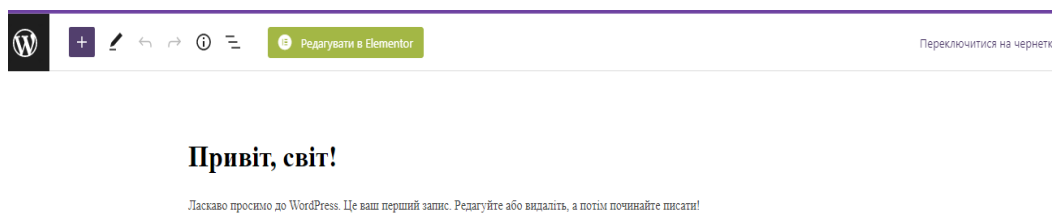


Рисунок 2.5 — Редактор тексту в Wordpress

Найпопулярніші редактори Joomla! — це TinyMCE, Code Mirror включені в стандартну збірку та редактор JCE, який є розширенням інших виконавців. Всі три текстові редактори повністю безкоштовні, що забезпечує комфортну роботу зі всім інформаційним контентом.

Додаткові можливості Drupal, відома завдяки власній таксономічній складової і зручним налаштування розмітки та категорій.

Простота є головним плюсом як для новачків, так і для професіоналів. Розробивши сайт на WordPress, в руках клієнта, можна не турбуватися про правки або доопрацювання продукту, власних зможе самостійно наповнити контентом web-сайт як йому заманеться. Кількість довідкової та іншої інформації в мережі та простота редагування, чинять те, що CMS стає ідеальною для розробників з невеликим досвідом та для замовників які хочуть самостійно налаштовувати власний сайт.

Система Joomla! створена для основної цілі розробки платформ мас-медіа.

Розробка дизайну Drupal починається із звичайного шаблону, в свою чергу складається з основних файлів: сторінка макета, сторінка виведення матеріалу, сторінка виведення блоку, і сторінка виведення меню. Для кожного типу і блоку можна застосовувати індивідуальний шаблон.

WordPress не до кінця виконана структура: для «header», для «sidebar» і для «footer» свої файли шаблону. В переробленні якогось шаблону під свій

проект можуть виникати помилки навіть при створенні не складного інтернет-магазину.

Joomla для досвідченого розробника найзручніший варіант, так як все редагування відбувається за допомогою навичок в мові гіпертекстової розмітки та спеціальної мови стилів сторінок.

Розміщення меню у Drupal можна вибрати потрібні позиції (рис. 2.6). Однак для правильного оформлення потребується знання HTML і CSS.

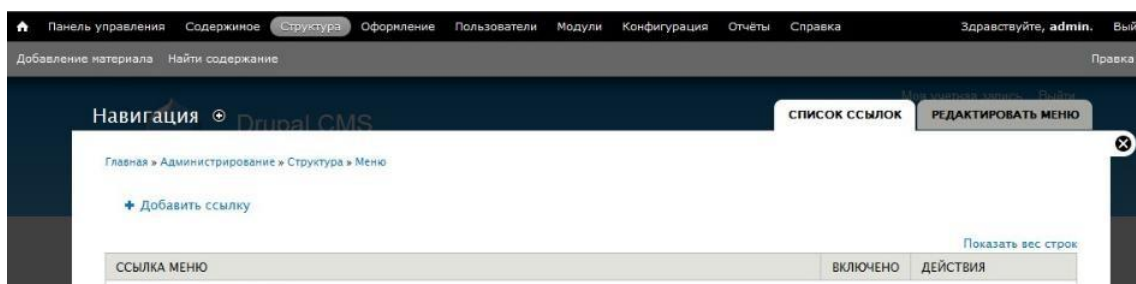


Рисунок 2.6 — Налаштування навігації Drupal

WordPress: Місце і стиль меню суворо закріплені в шаблоні (рис. 2.7). Для редагування потребує певного досвіду та навичок програмування PHP, HTML і CSS для зміни меню, або встановлення додаткового модулю який дозволить корегувати блоки.

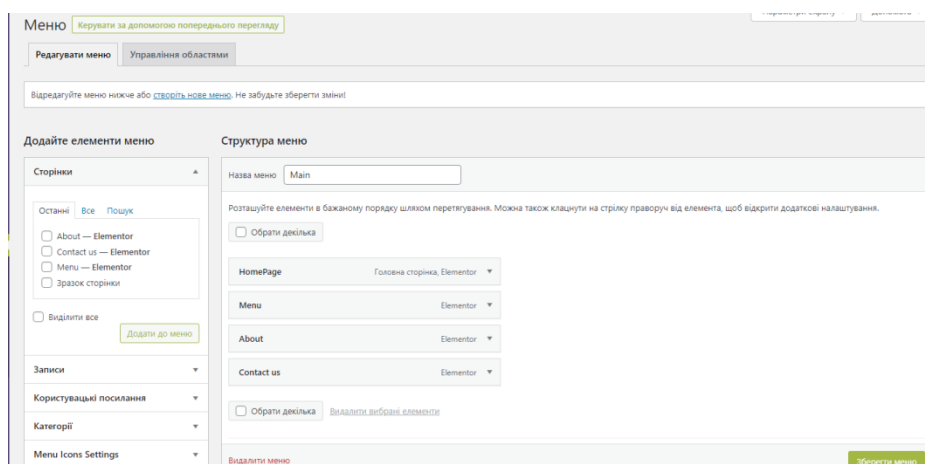


Рисунок 2.7 — Налаштування меню WordPress

Joomla Можливість змінити позицію в шаблоні на ту яку потребує

замовники дає унікальні можливості для візуального оформлення, також можна змінювати відображення меню (горизонтально або вертикально). Менеджер створення меню представлений на рис. 2.8.

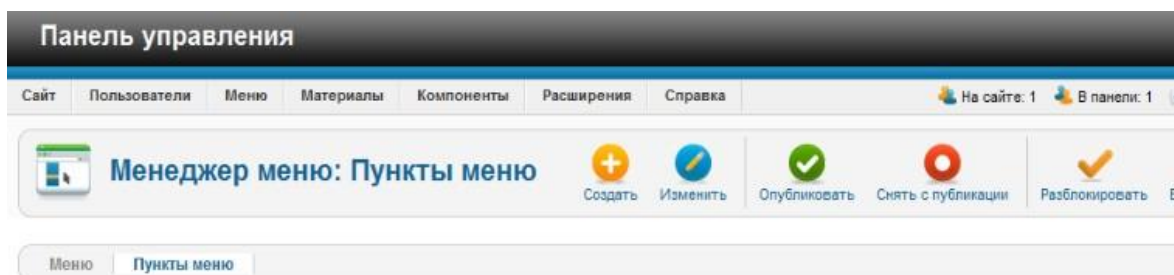


Рисунок 2.8 — Менеджер меню Joomla

### Порівняльна таблиця

В додатку Г (таблиця Г.1) наведено порівняльну характеристику найбільш популярних CMS Joomla!, Drupal та WordPress.

## 3 ПРОЕКТУВАННЯ WEB-САЙТ З ВИКОРИСТАННЯМ СИСТЕМИ КЕРУВАННЯ ВМІСТОМ WORDPRESS

### 3.1 Опис структури та бази даних WordPress

CMS WordPress використовує кілька взаємопов'язаних таблиць. На новому веб-сайті WordPress за замовчуванням є 11 таблиць. Кожна таблиця може зберігати лише певні дані. Наприклад, таблиця коментарів «WP» фіксує всю інформацію, залишену людиною, яка коментує публікацію, як-от IP-адреса, код автора коментаря тощо. Зберігання даних у певній таблиці робить їх швидшим і легшим пошуком. З'єднання встановлюються за принципом "один до багатьох". Наприклад, на одній сторінці може бути багато коментарів. Якщо створюється кілька сайтів із загальною базою даних, то потрібно вказати інший префікс для кожної встановленої таблиці. Детальніше в таблиці 3.1.

Таблиця 3.1 — Структура WordPress

Таблиця	Дані	Зв'язок з іншими таблицями
wp_posts	Публікації, сторінки, вкладення, редакції та пункти меню навігації	wp_postmeta через post_id wp_term_relationships через post_id
wp_postmeta	Метадані для кожної публікації	wp_posts через post_id
wp_comments	Коментарі	wp_posts через post_id
wp_commentmeta	Метадані для кожного коментаря	wp_comments через comment_id
wp_term_taxonomy	Таксономії (включаючи категорії та теги)	wp_term_relationships через term_taxonomy_id



Продовження таблиці 3.1

wp_terms	Ваші категорії та теги, а також терміни, призначені для спеціальних таксономій	wp_term_taxonomy через term_id
wp_links	Посилання у вашому списку блогів (якщо вони є)	wp_term_relationships через link_id
wp_users	Користувачі	wp_posts через post_author
wp_user_meta	Метадані для кожного користувача	wp_users через user_id
wp_options	Налаштування та параметри сайту (встановлюються за допомогою екранів налаштувань і за допомогою плагінів і тем)	Відсутні
wp_term_relationships	Зв'язки між посадами та таксономіями	wp_posts через post_id wp_term_taxonomy через term_taxonomy_id

Більшість таблиць пов'язані одна з одною. Одна частина даних може бути пов'язана з іншими даними, наприклад, публікація може бути пов'язана з категоріями та тегами. Таким чином, таблиця, в якій зберігаються публікації блогу, матиме взаємозв'язок із таблицями, де зберігаються категорії та теги.

Перевірити таблиці можна увійшовши в обліковий запис хостингу WordPress і вибравши phpMyAdmin зі власної cPanel.

В додатку В (таблиця В.1) показано ці таблиці та їх зв'язки. Майже всі таблиці використовують єдине поле для посилання на одну або кілька інших таблиць. Це поле, безумовно, є унікальним ідентифікатором (наприклад, `post_id`) у кожному записі. Стандартний префікс `wp_` у WordPress може не змінюватися.

— `wp_post` — найважливіша таблиця, оскільки типи вмісту з ваших публікацій і сторінок зберігаються саме в ній, типи вмісту включають текст, редакції, пункти меню, медіа-вкладення та всі користувацькі елементи.

— `wp_postmeta` — це розширення попередньої таблиці, вона зберігає додаткову інформацію з публікацій, спеціальні плагіни зберігають дані в цій таблиці, наприклад плагін для соціальних мереж MashShare зберігає кількість певних публікацій у цій таблиці.

— `wp_option` — таблиця, у якій зберігаються конфігурації веб-сайту, такі як назва сайту, слоган і часовий пояс, також він також зберігає налаштування плагінів і тем. На відміну від інших таблиць, вона не має жодного спільного зв'язку.

— `wp_users` зберігає список усіх зареєстрованих користувачів сайту WordPress, також зберігає основну інформацію, таку як ім'я користувача, паролі, ідентифікатор електронної пошти, відображуване ім'я, час реєстрації тощо.

— `wp_usermeta` зберігає додаткову інформацію про користувачів у таблиці `wp_usermeta` наприклад, прізвища клієнтів.

— `wp_terms` зберігає категорії як для сторінок так і для публікацій, а також теги для них і посилання, що пов'язані з категоріями.

— `wp_term_taxonomy` вміщує в собі описи категорій, теги та певні посилання, пов'язані з категоріями

— таблиця `wp_term_relationships` — допомагає підтримувати відносини, наприклад, у цій публікації та, яку ви читаете, пов'язана з кількома тегами і категорією та допомагає підтримувати цю асоціацію;

— `wp_comments` — таблиця, у якій зберігаються як схвалені, так і

неприйняті коментарі, залишені до ваших публікацій і сторінок, конкретні дані про автора, такі як ім'я автора, адреса електронної пошти, тип коментаря (будь то простий коментар, (pingback чи трекбек), також зберігаються в цій таблиці;

- `wp_links` — таблиця, що зберігає інформацію, пов'язану з блогами, її в основному використовують розробники, які працюють з більш старими версіями системи;

- `wp_commentmeta` зберігає додаткові дані про коментарі, залишені на веб-сайті, наприклад, з якою публікацією пов'язаний коментар.

База даних є невід'ємною частиною веб-сайту WordPress. Тому що він зберігає всю інформацію, необхідну для роботи сайту. Для розробників плагінів або WordPress необхідно знати, як працює база даних.

### 3.2 Встановлення локального серверу ХАМРР

ХАМРР — безкоштовна кросплатформова збірка веб сервера з відкритим початковим кодом в якій міститься набір програм для функціонування сайту. При цьому не потрібне налаштування кожного компонента окремо. Складання дозволяє вести розробку на окремому комп'ютері без підключення до мережі. Відповідно, немає необхідності у високошвидкісному доступі до Інтернету та наявності хостингу. Проект кросплатформенний, а це означає, що він однаково добре працює на популярних платформах: Windows, Linux, Mac OS.

Сама назва ХАМРР — аббревіатура, яка розшифровується наступним чином:

- `X` — означає, що програма працюватиме на всіх операційних системах;

- `Apache` — сервер, який підтримує найуживаніші операційні системи, зручний в налаштуванні;

- `MySQL` — система керування базами даних, використовує переважна більшість;

- `PHP` — мова програмування, підтримувана майже всіма хостинг-

провайдерами та лідер серед мов, що використовуються для створення веб-сайтів;

— Perl — легка в освоєнні мова програмування, яка, до того ж, підтримує модулі сторонніх розробників.

Завантажити XAMPP можна з будь-яких ресурсів, як з офіційних так і з інших.

Но краще не завантажувати зі сторонніх неперевічених сайтів, тому що в подальшому можливі такі проблеми в вигляді вірусів, він-локерів і додатково встановленими програмами.

Заходимо на офіційний сайт збірки [www.apachefriends.org](http://www.apachefriends.org). Після чого потрібно вибрати операційну систему, виходячи з того в якій працюємо та завантажити (рис. 3.1). Останні версії XAMPP працюють лише з новими операційними системами – XP, Vista, програма просто не запуститься.



Рисунок 3.1 — Вибір версії XAMPP

Далі буде запропоновано директорію для встановлення файлу завантаження. На жорсткому диску інсталятор займе 164 мегабайт, час завантаження залежить від швидкості інтернет з'єднання після встановлення XAMPP буде займати до 1 гігабайту пам'яті .

Наступним кроком потрібно обрати основні компоненти які знадобляться, а саме: Apache, MySQL, FileZilla FTP Server, PHP, phpMyAdmin, а інші встановлюємо додатково, щоб полегшити роботу в майбутньому та зекономити час на встановленні потрібного забезпечення. Після чого натиснути “Next”, та дочекатись встановлення локального серверу XAMPP, відкрити його з правами адміністратора та надати дозвіл в мережу через брандмауер. Обрати потрібні встановлювальні компоненти (рис. 3.2).

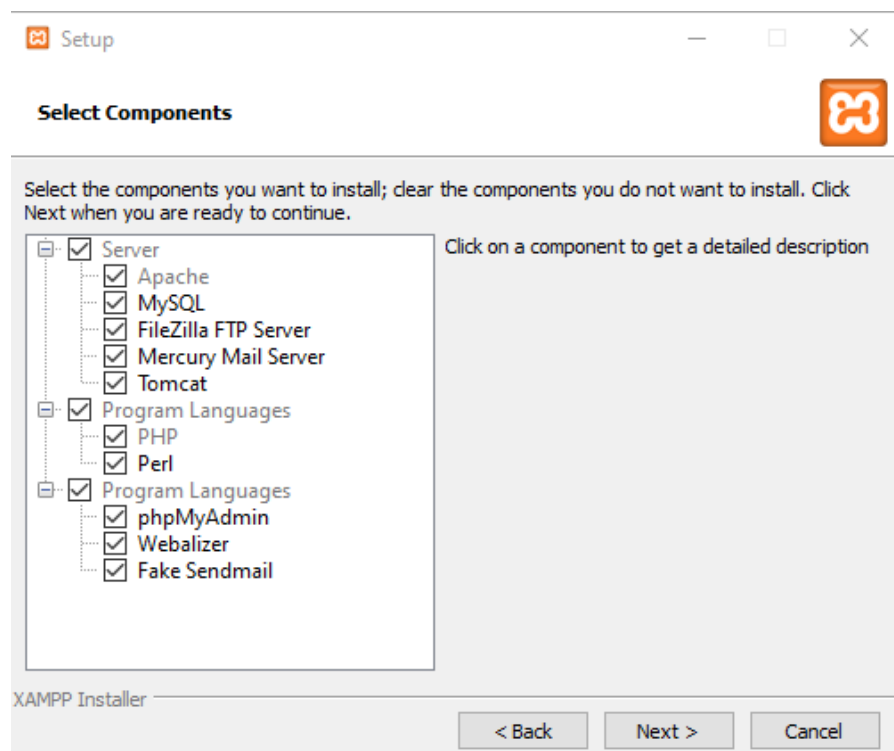


Рисунок 3.2 — Компоненти XAMPP

Обов’язково потрібно перевірити працездатність встановлених модулів та компонентів. Запустити сервер та після відкриття панелі керування XAMPP послідовно запустити MySQL та Apache (рис. 3.3).

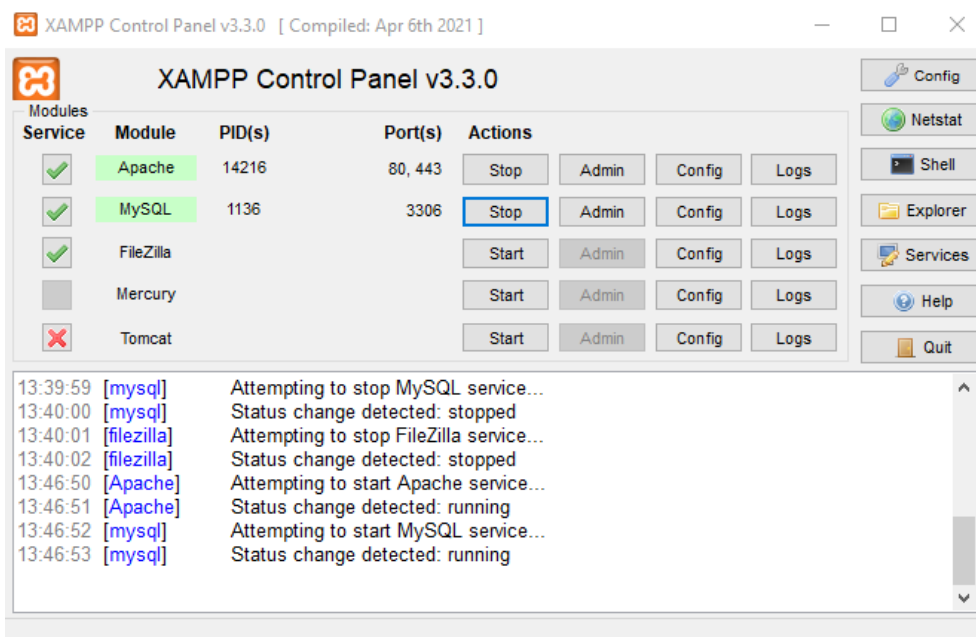


Рисунок 3.3 — Панель керування XAMPP

Через кнопку “Admin” навпроти MySQL переходим в скрипт для праці з базами даних PhpMyAdmin. На рис. 3.4 зображено головну сторінку для роботи з PhpMyAdmin. Відразу створимо базу даних для майбутнього сайту.

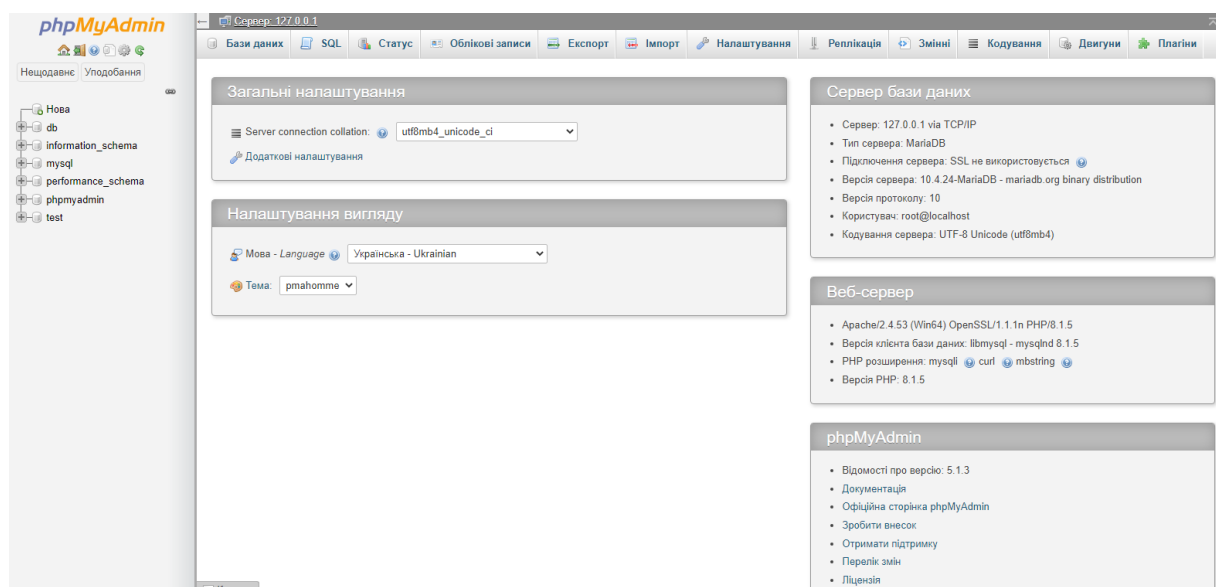


Рисунок 3.4 — Головна сторінка PhpMyAdmin

Переходимо в вкладку «Нова», задаємо ім'я «db» та вибираємо алгоритм сортування `utf8mb4_unicode_general_ci` — який представляє собою спрощений набір правил сортування ціллю якого — робити все що він може,

використовуючи множинне скорочення, задля збільшення швидкості та натискаємо «Створити» (рис. 3.5).

## Бази даних

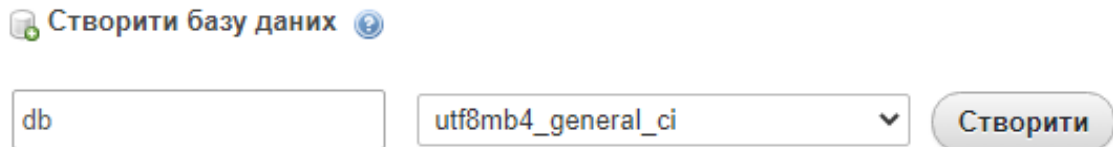


Рисунок 3.5 — Форма створення бази даних

Також потрібно перевірити працездатність служб, тому створюємо в директорії де був встановлений ХАМРР в папці «htdocs» створюємо ще одну папку з назвою «test», та пишем простий скрипт на РНР і зберігаємо його під назвою «index.php». На рис. 3.6 зображений скрипт для перевірки працездатності служби.

```
index.php x
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Test</title>
5 </head>
6 <body>
7   <?php echo '<p>Script Work</p>';?>
8
9 </body>
10 </html>
```

Рисунок 3.6 — РНР скрипт

Якщо в будь-якому браузері по адресі: `http://localhost/test` і все виконано правильно, то на екрані виведеться надпис «Script work». На цьому налаштування локального серверу закінчено. На рис. 3.7 зображено результат виконання скрипту.

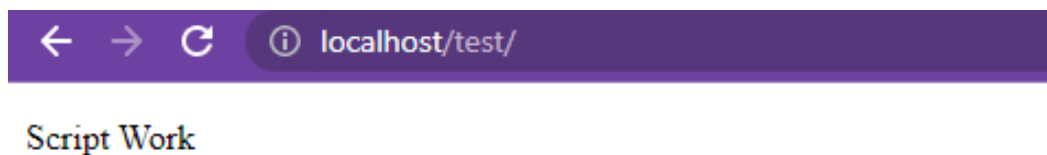


Рисунок 3.7 — Результат роботи виконання скрипту

### 3.3 Встановлення WordPress

WordPress — кросплатформова система керування вмістом написана на мові програмування PHP. Вбудована система тем і плагінів у поєднанні з чудовою архітектурою дозволяє проектувати на основі WordPress майже будь-які веб-сайти.

Щоб встановити дану CMS, потрібно перейти на офіційний сайт та завантажити останню версію системи. Після встановлення, попередньо створити папку в директорії XAMPP в папці «htdocs» з назвою «Diploma» та розпакувати архів в нею. Наступним кроком в будь-якому браузері переходимо по адресі «localhost/test/», де «test» — це назва нашого проекту який був створений в директорії XAMPP. На екрані з'явиться меню для вибору мови WordPress обираємо зручну мову на якій буде весь інтерфейс та натискаємо «continue». Далі потрібно вказати назву бази даних, яка вже була створена попередньо, ім'я користувача за замовчуванням це «root», хост бази даних залишаємо «localhost» та табличний префікс «wp\_», натискаємо «відправити». На рис. 3.8 зображено базові дані для WordPress.

Потім потрібно вказати необхідну інформацію для встановлення WordPress, таку як: назва сайту буде «Diploma», ім'я користувача «Andrew», пароль та e-mail за якими будемо авторизовуватись в подальшому, натиснути «Встановити WordPress» та авторизуватись використовуючи раніше задані дані. На рисунку 3.9 зображено початкові дані WordPress.

Потім потрібно вказати необхідну інформацію для встановлення WordPress, таку як: назва сайту буде «Diploma», ім'я користувача «Andrew», пароль та e-mail за якими будемо авторизовуватись в подальшому.



Below you should enter your database connection details. If you are not sure about these, contact your host.

Назва бази даних	<input type="text" value="db"/>	Назва бази даних, яку ви хочете використовувати з WordPress.
Ім'я користувача	<input type="text" value="root"/>	Ваше ім'я користувача бази даних.
Пароль	<input type="password"/>	Ваш пароль бази даних.
Хост бази даних	<input type="text" value="localhost"/>	You should be able to get this info from your web host, if localhost does not work.
Табличний префікс	<input type="text" value="wp_"/>	Якщо ви хочете мати одну базу даних для кількох інсталяцій WordPress, змініть це.

Рисунок 3.8 — Базові дані WordPress

Натиснути «Встановити WordPress» та авторизуватись використовуючи раніше задані дані. На рисунку 3.9 зображено початкові дані WordPress.

Ласкаво просимо

Ласкаво просимо до відомого п'ятихвилинного процесу встановлення WordPress! Просто заповніть інформацію нижче, та скоро ви будете користуватись найбільш розширеною та потужною платформою персональної публікації в світі.

Необхідна інформація

Please provide the following information. Do not worry, you can always change these settings later.

Назва сайту

Ім'я користувача

Імена користувачів можуть містити тільки букви, цифри, пробіли, нижні лінії, дефіси, крапки, та символ @.

Пароль

**Важливо:** Вам буде потрібен цей пароль, щоб увійти. Будь ласка, зберігайте його в безпечному місці.

Ваш e-mail

Ділічі перевірте свою e-mail адресу перед тим, як продовжити.

Видимість для пошукових систем  Запропонувати пошуковим системам не індексувати цей сайт

Пошукові системи можуть ігнорувати цей запит.

Рисунок 3.9 — Початкові дані WordPress

Якщо після авторизації виконався вхід в майтерню системи, то це значить, що завантаження пройшло успішно. Використовуючи панель

управління можна ввійти на сирий сайт та переконатись, що все працює правильно (рис 3.10).



Рисунок 3.10 — Стартова сторінка

Та відразу перейдемо до встановлення теми з якою буде працювати наш сайт. Переходим в консоль на вкладку «Тема», натискаєм на кнопку «Додати» і в пошуку шукаємо тему «OceanWP» встановлюємо та активуємо.

OceanWP — ідеальна тема для проекту. Легка і дуже розширювана, вона дозволить створити майже будь-який тип веб-сайту, такий як блог, портфолію, веб-сайт для бізнесу або магазин WooCommerce з гарним та професійним дизайном. Дуже швидка, адаптивна, RTL яка готова до перекладу, кращі практики SEO, унікальні функції WooCommerce для збільшення конверсії та багато іншого.

Можна редагувати налаштування на планшеті та мобільному пристрої, щоб сайт виглядав добре на будь-якому пристрої. Працювати з найпопулярнішими конструкторами сторінок, такими як Elementor, Beaver Builder, Visual Composer, Divi, SiteOrigin та інші. Розробники люблять її гнучкий код, що робить її простою у налаштуванні та розширюванні. Найкращий друг Elementor та WooCommerce [7].

### 3.4 Встановлення плагінів

За допомогою різноманітних плагінів, яких WordPress налічує більше 54 000 одиниць, можна втілити на власному веб-сайті будь-яку ідею чи задумку не витрачаючи багато часу для рішення задачі або ж завантажити свої власні плагіни за потреби. Встановлення відбувається так само як з темами, через консоль в вкладці плагіни натискаємо «Додати» та в пошуку шукаємо потрібні нам плагіни.

Першим модулем використовуємо Elementor — це найкращий конструктор веб-сайтів WordPress. В ньому є живий редактор перетягування, який дає змогу створювати та налаштовувати кожен частину сайту інтуїтивно за допомогою візуального конструктора. Не потрібні навички кодування, що дасть змогу замовнику самостійно змінювати вміст, та розташування віджетів, яких плагін налічує більше 100, можна досягти своїх цілей дизайну для будь-яких потреб. На рис. 3.11 зображений візуальний конструктор Elementor.

Також завантажуюмо плагіни WPForms та Ocean Extra. Ocean Extra додає до встановленої теми «OceanWP» додаткові функції, такі як віджети, метабокси, активація/деактивація розділів налаштування, увімкнення/вимкнення скриптів і стилів теми, імпорт/експорт і панель для активації преміальних розширень.

WPForms — це найбільш зручний для початківців плагін контактної форми WordPress. Його використовують понад 5 000 000 сайтів. Його онлайн конструктор форм, дозволяє легко створювати красиву контактну форму, форму підписки на електронну пошту, форму оплати або будь-який інший тип онлайн-форми лише кількома кліками. І цілком адаптивний для гарного вигляду на мобільних телефонах, планшетах і настільних комп'ютерах . WPForms також можна використовувати для проведення опитувань. Він поставляється з красивими графіками та діаграмами звітів про ваші результати.

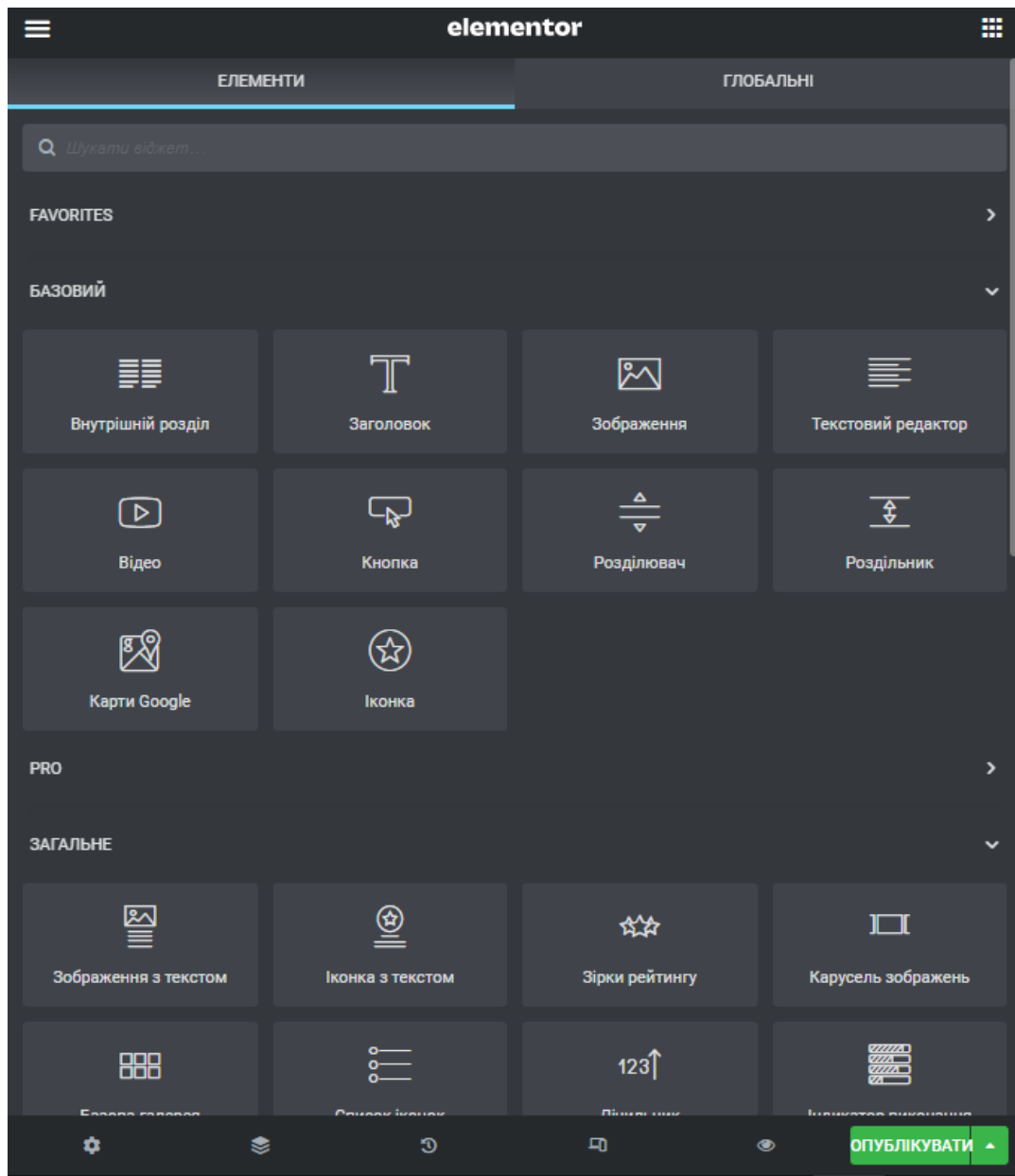


Рисунок 3.11 — Візуальний конструктор Elementor

Переваги використання плагіну:

— адаптивні форми, бо WPForms зручний для мобільних пристроїв, а це означає, форми завжди виглядатимуть чудово, незалежно від пристрою, яким користується відвідувач;

— конструктор форм перетягування — зручний та зрозумілий конструктор онлайн-форм перетягуванням дозволяє легко створювати контактні форми WordPress та інші онлайн-форми всього за кілька хвилин без написання коду;

— шаблони форм — можливість використовувати більше 150 готових шаблонів форм, щоб заощадити час, та не починати з нуля кожного разу;

— захист від спаму, так як WPForms використовує розумну CAPTCHA та метод Honeypot, щоб зупинити надсилання спам-форм.

— Швидкі сповіщення про форму ви будете отримувати електронною поштою, щойно форму буде надіслано, це важливо, якщо вам потрібно відразу дізнатись про замовлення, наприклад, якщо ви керуєте рестораном із замовленням їжі в Інтернеті, крім того, у WPForms легко створювати сповіщення для кількох одержувачів — це може означати умовне надсилання деяких або всіх деталей форми іншій людині у вашій команді або надсилання автоматичної квитанції форми особі, яка заповнила форму;

— розумне підтвердження форми, так як тільки хтось надішле форму, ви можете скористатися функціями підтвердження форми, щоб показати їм власне повідомлення про успіх або перенаправити його на спеціальну сторінку подяки.

— завантаження файлів — легко (і безпечно) збирати файли та медіа за допомогою онлайн-форм, використанням параметра завантаження файлів WPForms чудовий спосіб додати на ваш сайт відгуки, створений користувачами вміст або навіть гостьові дописи.

— багатосторінкові форми, бо за допомогою WPForms можна легко розділити довгі форми на кілька сторінок, щоб покращити роботу користувача, особливо корисно для опитувань або детальних форм замовлення;

— інтеграція платежів, бо за допомогою WPForms можна легко інтегрувати свої форми з PayPal, Stripe та/або Authorize.Net, щоб приймати платежі, пожертвування та онлайн-замовлення, полегшує створення індивідуальних форм замовлення;

— розумна умовна логіка — здатність показати або приховати вказані поля та розділи форм на основі поведінки користувача,

використовуючи умовну логіку. Наприклад, якщо є форма зворотного зв'язку, ви можете показати різні варіанти залежно від оцінки, яку користувач дає вашій службі, наприклад, якщо ставиться оцінка 4 або 5 зірок, можна попросити відгук, а якщо вони ставлять оцінку 1 або 2 зірки, то можна заохочувати їх повідомити вам, що пішло не так, щоб ви могли допомогти виправити це;

— легке керування записами WPForms зберігає записи форм у вашій базі даних WordPress — це означає, що можна керувати своїми записами форми прямо з інформаційної панелі WordPress, таким чином можна сортувати записи у формі та помічати їх як прочитані або навіть видалити непотрібні, приємним бонусом стає експорт всіх записів у файл «.csv», щоб легко переглядати їх у Excel;

— розмовні форми WPForms — це інтерактивні форми, розроблені виключно для WordPress, вони підвищують швидкість заповнення форм і загальну кількість конверсії форми замість того, щоб одночасно показувати цілу купу запитань, вони використовують підхід «по одному питанню», як справжня розмова.

Вміння використовувати плагіни при створенні web-сайту за допомогою CMS WordPress важливо, так як це зекономить купу часу для простих рішень, що дає змогу замовнику в подальшому інтуїтивно налаштовувати свій власний сайт без допомоги розробника, що надасть більшої автономності роботі веб-сайту.

### 3.5 Розробка структури та наповнення сторінок

#### 3.5.1 Структура веб-сайту

Структура сайту визначає, наскільки легко користувачам орієнтуватися на web-сайту та знаходити потрібну інформацію. Структура сайту — це те, як різні сторінки вашого сайту пов'язуються одна з одною за допомогою внутрішніх посилань та їхньої ієрархії. Як інформація на сайті організована та

представлена так, щоб алгоритм добре читав її контекст. Хороша структура сайту полегшує навігацію як для користувачів, так і для сканерів, що покращує рейтинг SEO вашого веб-сайту в пошукових системах, тому важливо обрати найоптимальнішу структуру.

Так як за тематику сайту взята тема кав'ярні, то найкраще підходить лінійна, або ж послідовна структура. В якій сторінки з'єднані лінійно, і всі вони виконують логічний шлях дій, наприклад, від головної сторінки до сторінки сервісу та до сторінки запити/цитати. Він добре працює для невеликих веб-сайтів, які мають на меті представити продукт, послугу чи бренд. На рис. 3.12 графічно зображена структура інтернет сторінки кав'ярні.

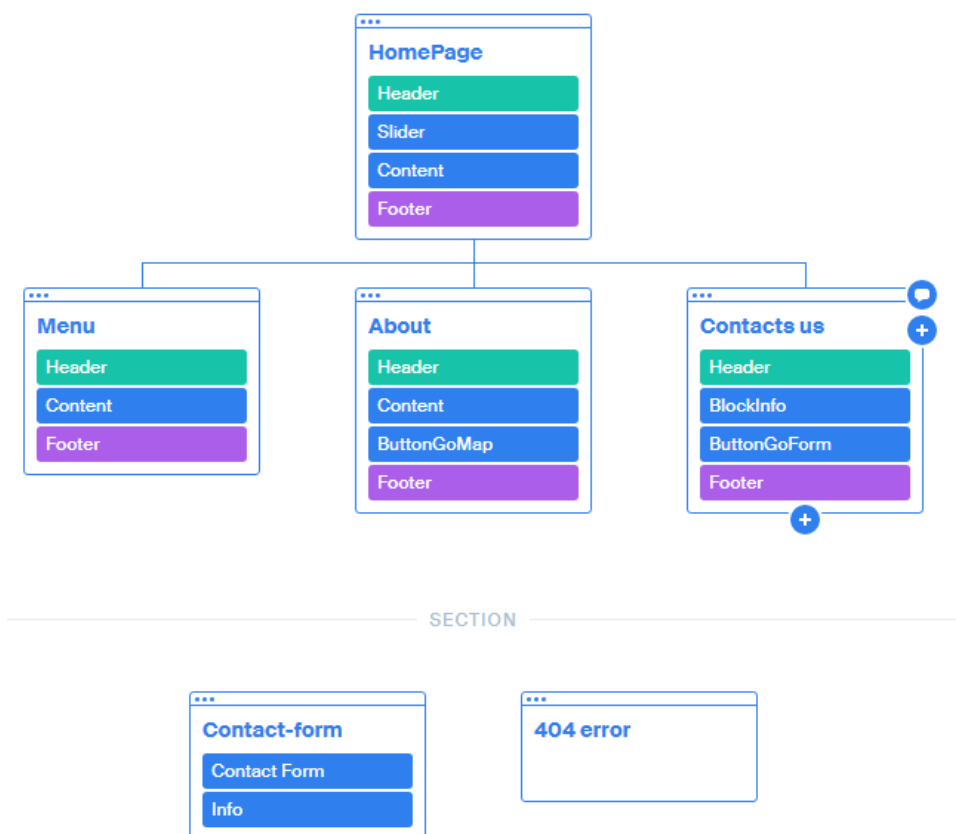


Рисунок 3.12 — Графічна структура веб-сайту

Основним засобом переходу по цій структурі є меню сайту, через яке будуть відбуватись основні переходи між сторінками, тому воно має містити всі важливі розділи. Але спочатку їх потрібно створити через майстерню WordPress в вкладці «сторінки» натиснути на «Додати нову» та в рядку

«додати заголовок» ввести назву сторінки «Homepage» і за допомогою встановленого плагіна OceanWP задати потрібні параметри, а саме вимкнути відображення заголовку, макет вмісту 100% по ширині, вимкнути верхню панель.

Ще створити сторінки можна, використовуючи читабельні слова та робити URL-адресу якомога коротшою : menu, about, contact us, contact-form. Під час введення назви використовувати дефіси (-) замість символів підкреслення (\_), оскільки пошукові системи сприймають перший як засіб поділу слів, а другий як частину слова. Та виконати для них такі ж самі дії.

В консолі системи в пункті «вигляд → меню» розташовуємо елементи в бажаному порядку шляхом перетягування. На рис. 3.13, 3.14 зображена структура меню інтернет сторінки.

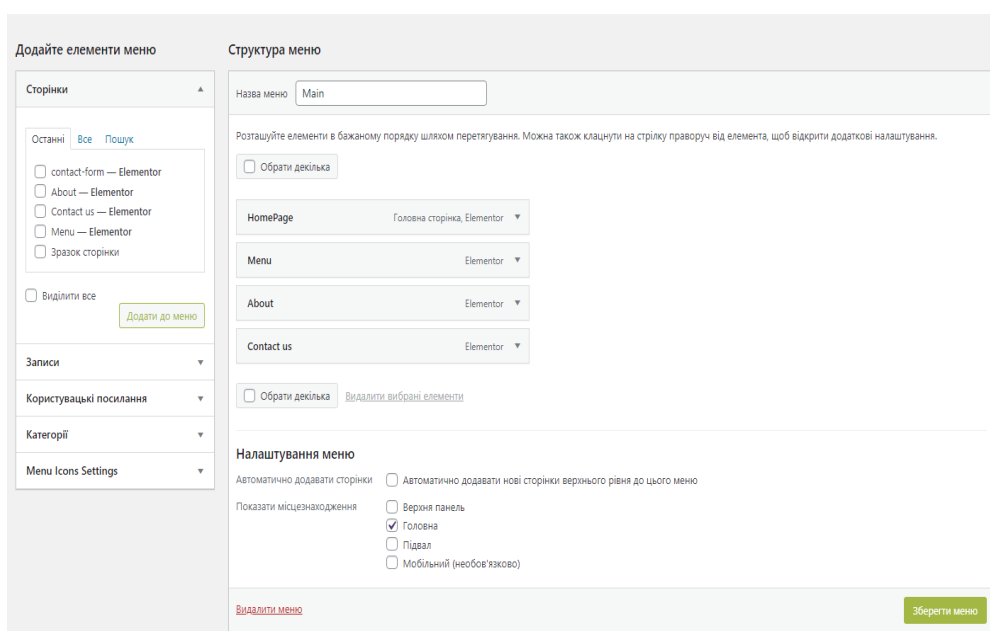


Рисунок 3.13 — Структура меню інтернет сторінки

DeKava

HOME PAGE

MENU

ABOUT

CONTACT US

Рисунок 3.14 — Структура меню інтернет сторінки



### 3.5.2 Інтерфейс користувача

Дизайн інтерфейсу користувача (UI) зосереджується на передбаченні того, що може знадобитися зробити користувачам, і забезпечення того, щоб інтерфейс мав елементи, до яких легко отримати доступ, зрозуміти та використовувати для полегшення цих дій. Інтерфейс користувача об'єднує концепції дизайну взаємодії, візуального дизайну та інформаційної архітектури.

Користувачі вже знайомі з елементами інтерфейсу, які діють певним чином, тому потрібно використовувати прості та звичні елементи у своєму виборі та їхньому розміщенні. Це сприятиме зручному та ефективному використанні сайту.

Логотип або назва сайту, та головне меню розміщені в заголовку веб-сайту — це блок, розміщений у верхній частині сторінки, який містить найважливіші категорії або сторінки (наприклад розділ «About», «Contact us» і т.п.).

Футер сайту (Підвал або нижній колонтитул) — це блок розміщений у нижній частині сторінки, який може дублювати те. Що включено в головне меню, а також містить посилання на політику конфіденційності та іншу документацію, поширені запитання, контакти тощо. Також в ньому можна виконати фільтрування та сортування – якщо на сайті ж багато послуг або продуктів, то слід розробити систему фільтрів і параметрів сортування, щоб допомогти користувачам знайти те, що вони шукають. Фільтри можуть відрізнятися в залежності від категорії і повинні охоплювати всі можливі характеристики продукту.

Інформаційне наповнення — повинно бути не перевантаженим, зрозумілим для всіх користувачів та приємним для ока, включає в собі основну інформацію про продукт, опис або ж історію компанії чи підприємства, наглядно показує тематику сайту. На рис. 3.15 зображено структурний інтерфейс веб-сайту. В додатку Б (рис. Б.1) наведено функціональну схему web-сторінки.

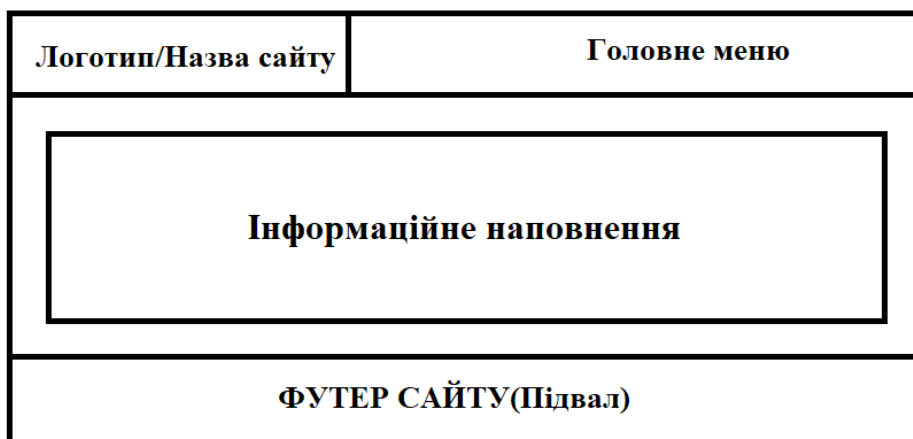


Рисунок 3.15 — Інтерфейс Web-сайту

Для редагування сторінок, потрібно перейти на сторінку та в верхній панелі натиснути «Редагувати в Elementor» або ввійти в майстерню через вкладку «сторінки», та навести курсор на потрібну сторінку, та натиснути на таку ж кнопку, щоб відкрити конструктор налаштування сторінок цього плагіна. Далі Інформаційно заповнити блоками та модулями сайт, на головній сторінці додати фотогалерею кав'ярні та блок інформації з графіком роботи, та кнопки для зв'язку, створити нижній колонтитул на всіх сторінках задати в ньому меню, контакти, та посилання на E-mail, Instagram, YouTube канал та Facebook. На рис. 3.16 зображено фотогалерею на головній сторінці.



Рисунок 3.16 — Фотогалерея

Приклад головної сторінки у додатку Б (рис Б.2).

На сторінці «Menu» додати блок для заголовку, та окремі блоки для меню які мають містити: назву напою, його опис та ціну. Приклад сторінки з меню наведений у додатку Б (рис Б.3).

Сторінка «About» виконана схожим чином, додана кнопка переходу на карту з місцезнаходженням кав'ярні, та запрошенням перейти на сторінку «Contact us». Вигляд сторінки опису кав'ярні наведений у додатку Б (рис. Б.4).

«Contact us» виконана як і всі попередні сторінки мінімалістично, та наведена основна інформація з кнопкою для переходу на контактну форму. Вигляд інформаційної сторінки зв'язку наведено в додатку Б (рис. Б.5).

«Contact-form» містить в собі форму створену за допомогою плагіну WPForms, Google maps на якій зображено місце знаходження кав'ярні. Вигляд сторінки контактної форми наведено в додатку Б (рис. Б.6).

Додаткову навігацію та посилання на інші інформаційні ресурси створити в footer. На рис. 3.17 наведений приклад Footer.

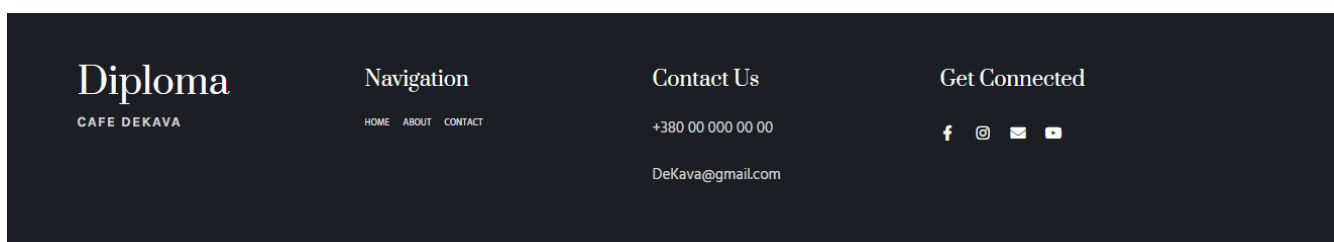


Рисунок 3.17 — Footer

### 3.6 Постановка задачі аналізу ефективності запропонованого рішення

Для виконання техніко-економічний аналізу розробки програмних продуктів слід вибрати систему індексів якості.

Технічні вимоги до продукту наступні:

- адаптивність — однаково показуватись на різних пристроях;
- надавати цілодобовий безперервний доступ;
- виконати інтуїтивну взаємодію між клієнтом і сайтом та адміністратором і програмним забезпеченням;
- вирахувати витрати на створення проекту.

### 3.7 Обґрунтування системи параметрів ПП

#### 3.7.1 Кількісна оцінка параметрів

Для того, щоб охарактеризувати програмний продукт, будемо використовувати наступні параметри:

- Y1 — затримка відповіді на запит клієнта;
- Y2 — кількість модулів на сайті;
- Y3 — обсяг даних, що передається через Інтернет;
- Y4 — обсяг оперативної пам'яті, що використовується.

Параметри від найгіршого до найкращого обираються базуючись на вимогах замовника, що позначають ПП як показано у таблиці 4.1.

Таблиця 3.1 — Основні параметри ПП

Назва Параметра	Умовні позначення	Одиниці виміру	Значення параметра		
			найгірші	оптимал ьні	найкращі
Час відгуку на запит клієнта	Y1	с	11	5	3
Кількість модулів на сайті	Y2	Кількість модулів	4	6	11
Обсяг даних, що надсилається	Y3	Кб	3000	900	600
Обсяг оперативної пам'яті	Y4	Мб	250	200	150

За даними таблиці 3.18 будуються графічні характеристики параметрів (рис 3.18а — час відгуку на запит клієнта, 3.18б — Кількість модулів на сайті, 3.18в — Обсяг даних, що надсилається через Інтернет, 3.18г — Обсяг оперативної пам'яті, який використовується).

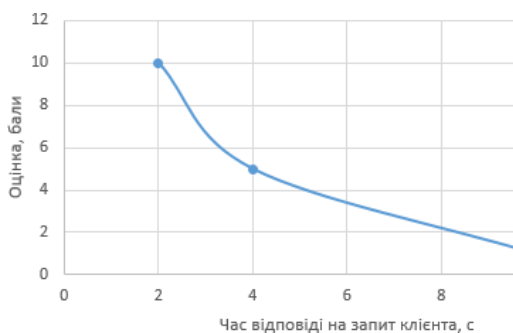


Рисунок 3.18а — Час відгуку на запит клієнта

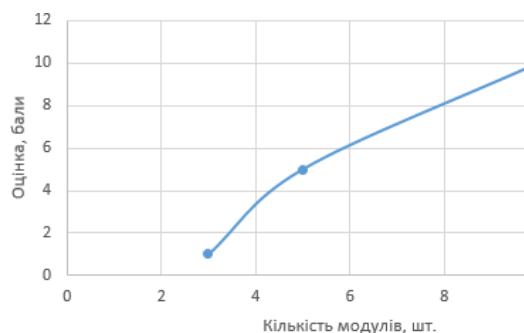


Рисунок 3.18б — Кількість модулів на сайті



Рисунок 3.18в — Обсяг даних, що надсилається через Інтернет

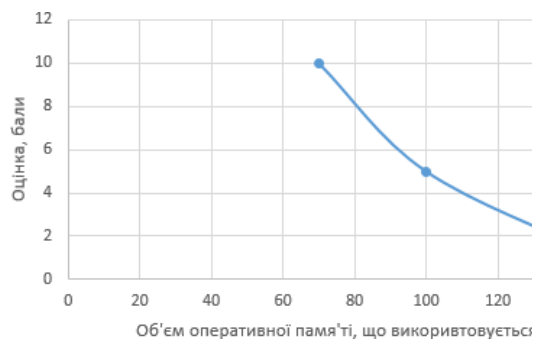


Рисунок 3.18г — Обсяг оперативної пам'яті, який використовується

### 3.7.2 Аналіз експертного оцінювання параметрів

Глибоко проаналізувавши, експерти визначають ступінь важливості всіх параметрів для конкретних цілей — вибір ПП, в якому є найкращий варіант продукту з відмінним та корисним набором можливостей при невеликих витратах.

Важливість параметрів визначається методом попарного порівняння. Коефіцієнт важливості містить в собі:

- визначення рівня важливості параметра відбувається внаслідок присвоєння різних ступенів;
- визначити оцінку пріоритету параметрів;

— опрацювати результати та визначити коефіцієнт значущості.

Результати експертного ранжування наведені у таблиці 3.2.

Таблиця 3.2 — Результати ранжування параметрів

Позначення параметра	Назва параметра	Одиниці виміру	Рівень параметра за оцінкою експерта							Сума рівнів $R_i$	Відхилення $\Delta_i$	$\Delta_i^2$
			1	2	3	4	5	6	7			
X1	Час відповіді на запит клієнта	с	1	2	1	2	1	1	1	9	-8,5	72,25
X2	Кількість модулів на сайті	К-сть модулів	4	3	4	3	4	4	3	25	7,5	56,25
X3	Обсяг даних, при надсиланні	Кб	2	1	2	1	2	3	2	13	-4,5	20,25
X4	Залучений обсяг оперативної пам'яті	МБ	3	4	3	4	3	2	4	23	5,5	30,25
Разом			10	10	10	10	10	10	10	70	0	179

Щоб перевірити всі рівні вірогідності оцінок експертів, потрібно виконати певні обчислення. сума рівнів всіх параметрів і суцільна сума рівнів Розглянуто докладніше структуру виразу (3.1)

$$R_i = \sum_{j=1}^N r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 70, \quad (3.1)$$

де  $N$  — к-сть експертів;

$n$  — число параметрів.

Проміжна сума рівнів:

$$T = \frac{1}{n} R_{ij} = 17,5. \quad (3.2)$$

Формулою (3.3) визначаємо відхилення суми рівнів параметрів від проміжної суми рівнів:

$$\Delta_i = R_i - T. \quad (3.3)$$

Сума відхилень для всіх параметрів дорівнює 0.

Суцільна сума квадратів відхилення:

$$S = \sum_{i=1}^N \Delta_i^2 = 179. \quad (3.4)$$

Обчислення коефіцієнта відповідності, формула:

$$W = \frac{12S}{N^2(n^3-n)} = \frac{12 \times 179}{7^2(4^3-4)} = 0,73 > W_k = 0,67 \quad (3.5)$$

Ранжування проведено правильно, оскільки коефіцієнт який дорівнює 0,67, перевищує нормативний.

Застосуємо попарне порівняння використовуючи результати ранжирування, наслідок впишемо в таблицю 3.2.

Таблиця 3.2 — Попарне порівняння параметрів

Параметри	Експерти							Кінцева оцінка	Числове значення
	1	2	3	4	5	6	7		
Y1 і Y2	>	>	>	>	>	>	>	>	1,5
Y1 і Y3	>	<	>	<	>	>	>	>	1,5
Y1 і Y4	>	>	>	>	>	>	>	>	1,5
Y2 і Y3	<	<	<	<	<	<	<	<	0,5
Y2 і Y4	<	>	<	>	<	<	>	<	0,5
Y3 і Y4	>	>	>	>	>	<	>	>	1,5

Формулою (4.6) з'ясуємо значення  $a_{ij}$ , котре формулює ступінь  $i$ -го

параметра над  $j$ -тим:

$$a_{ij} = \begin{cases} 1.5 & \text{при } Y_i > Y_j \\ 1.0 & \text{при } Y_i = Y_j \\ 0.5 & \text{при } Y_i < Y_j \end{cases} \quad (3.6)$$

Складаємо матрицю  $A = \| a_{ij} \|$ , з привілейованих оцінок. Для кожного параметра проводимо розрахунок вагомості  $K_{Bi}$  завдяки формулі (3.7)

$$K_{Bi} = \frac{b_i}{\sum_{i=1}^n b_i}, \text{ де } b_i = \sum_{j=1}^N a_{ij} \quad (3.7)$$

Розрахунок відносних оцінок відбувається до того моменту, коли всі наступні числа почнуть відрізнятися від попередніх (менш ніж на 2%). На послідуєчих кроках відносні оцінки розраховуються виразом (3.8).

$$K_{Bi} = \frac{b'_i}{\sum_{i=1}^n b'_i}, \text{ де } b'_i = \sum_{j=1}^N a_{ij} b_j \quad (3.8)$$

Таблиця 4.3 – Розрахунок вагомості параметрів

Параметри $x_i$	Параметри $x_j$				Перша ітер.		Друга ітер.		Третя ітер.	
	Y1	Y2	Y3	Y4	$b_i$	$K_{Bi}$	$b_i^1$	$K_{Bi}^1$	$b_i^2$	$K_{Bi}^2$
Y1	1,0	1,5	1,5	1,5	5,5	0,344	21,25	0,360	77,875	0,361
Y2	0,5	1,0	0,5	0,5	2,5	0,156	9,25	0,157	34,125	0,158
Y3	0,5	1,5	1,0	1,5	4,5	0,281	16,25	0,275	59,125	0,274
Y4	0,5	1,5	0,5	1,0	3,5	0,219	12,25	0,208	44,875	0,207
Всього:					16	1	59	1	216	1

Як видно з таблиці 4.3, різниця відмінність коефіцієнтів не більша за 2%, тому повторно застосовувати математичну задачу не потрібно.



## ВИСНОВКИ

У даній роботі було створено корпоративний веб-сайт для найзручнішого надання потрібної інформації між клієнтом та представником структури з використанням системи керування вмістом WordPress, проаналізовані аналоги створення Web-сторінки за допомогою інших систем.

Одною з переваг CMS є те, що вони чудово підходять для створення різних типів веб-сторінок. Також можна створювати графічні зображення свого сайту за допомогою наявних шаблонів і створювати власні унікальні конструкції.

Порівнявши популярні безкоштовні CMS можна зробити висновки, що:

— Drupal — досить зручна система з добре продуманою панеллю адміністратора в front-end частині. Має безліч плагінів. Дасть бажаний результат після налаштування. Потрібно лише добре вивчити можливості всіх необхідних модулів. Система підходить для складних інформаційних сайтів, блогів, форумів і порталів.

— WordPress — платформа номер один для найкращої платформи CMS. Це найпопулярніше у світі програмне забезпечення CMS, яке обслуговує близько 43% усіх веб-сайтів в Інтернеті. З доступом до тисяч тем і плагінів, як платних так і безкоштовних, що дозволяються додавати корисні доповнення по типу контактних форм, фотогалереї і т.п. Вона безкоштовна та має відкритий код, спочатку розроблена для введення блогів, а тепер використовують для розробки сайтів будь-якої

— Joomla! — Загалом, досить проста і гнучка система створення сайтів компаній. Ви можете створити робочий сайт без єдиного плагіна. Має зручний редактор для роботи з текстом. Видалені об'єкти можна відновити. Недоліками цієї системи є: шаблони які погано працюють, занадто багато файлів, структура трохи захаращена, а сайт може займати багато місця.

Побудова міцної та простої структури сайту — це не лише розділи та посилання — це також користувацький досвід та дизайн. Перш ніж

створювати сайт, потрібно переконатись, є достатньо часу і ресурсів у створенні зручної для SEO структури. Вивчити структури конкурентів, проведіть поглиблене дослідження ключових слів і класифікувати свої веб-сторінки на основі отриманих даних. Підтримувати чітку структуру URL-адрес, підключати сторінки внутрішніми посиланнями та створювати елементи навігації, які дозволять користувачам швидко знайти все, що їм потрібно.

В ході виконання функціонально-вартісного аналізу програмного продукту обрався найоптимальніший варіант розробки який надає найбільшу ефективність через використання найновіших версій MySQL та PHP, та швидкість створення за рахунок відмінного адміністративного меню з великою кількістю доступних тем та плагінів.

В роботі було обрано тематикою сайту кав'ярню, призначеного для надання інформації та зв'язку з клієнтами В результаті виконання практичної частини було розроблено структуру меню сайту, завантажено необхідні плагіни для сайту та інстальовано тему для зручнішого налаштування, реалізованого для CMS WordPress, додано пошукову систему Google maps, на сторінках, практичну фотогалерею та форму для зв'язку, також виконане гарне інформаційне та візуальне наповнення яке не захаращує простір та дає змогу користувачам легко та інтуїтивно використовувати Web-сайт.

Після повного створення даного сайту було протестована його робота. Сайт адаптивно відображається на всіх браузерах, та інших пристроях таких як планшети, телефони, тощо. Всі посилання та сторінки працюють.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Горнаков, С.Г. Осваиваем популярные системы управления сайтом / Горнаков С.Г. – ДМК Пресс, 2009. — С. 336
2. Lisa Sabin—Wilson. Wordpress All—In—One for Dummies – John Willey & Sons Publisinginf, 2019. 864/852 p.
3. Офіційний сайт Wordpress [Електронний ресурс] — режим доступу: <http://wordpress.org/>.
4. Tessa Blakeley Silver WordPress Theme Design б.м.: PACT Publishing, 2008. — 224 p.
5. Elias Ericson Learn WordPress. From beginner to advanced user — Elias Ericson — 152 p.
6. Chris Coyier, Jeff Starr Digging Into WordPress v3. б.м.: WordPress, 2012. — 442 p.
7. Итан Маркотт. Отзывчивый веб—сайт / Итан Маркотт – М:Манн, Иванов иФербер, 2012. — С. 176
8. Савельева Н. Системы управления контентом / Савельева Н. // Открытые системы, 2004. — С. 41—47
9. Прохоров Н. Системы управления контентом / Прохоров Н. — КомпьютерПресс, 2007. – С. 134—137
10. Титенко, С. В. FreshKnowledge — система управління навчальним Веб-контентом на семантичному рівні / С. В. Титенко, О. О. Гагарін // VII международная конференция «Интеллектуальный анализ информации ИАИ—2007», Киев, 15—18 мая 2007г. : Сб. тр. / Ред. кол. : С. В. Сирота (гл.ред.) и др. — К.: Просвіта, 2007. – С. 342—352.
11. Киссейн, Э. Основы контентной стратегии / Эрин Киссейн – М:Манн, Иванов и Фербер. 2012 – С. 128
12. Стратегія і основні кроки при розробці web—сайту. [Електронний ресурс] — режим доступу: <http://ruszura.in.ua/neobhidno—znaty/stratehiya—i—osnovni—kroky—pry—rozrobsi—web—sajta.html>.

13. Martin Brampton. PHP5 CMS Framework Development. — PACKT Publishing, June 2008. — 328 p.
14. Tris Hussey. Using WordPress. б.м.: Que, 2012, — 425p.
15. Володимир Преловаць. WordPress Plugin Development: PACKT Publishing February 16, 2009 – 296 p.
16. Веллінг, Томсон // Розробка веб-додатків за допомогою PHP і MySQL, 2017р. - 768 с.

## ДОДАТОК А

Технічне завдання

Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ

проф., д.т.н.. Азаров О.Д..

" " 2022 р.

### ТЕХНІЧНЕ ЗАВДАННЯ

на виконання бакалаврської дипломної роботи

“Web-сайт з використанням системи керування WordPress.”

08-23.БДР.011.00.000 ТЗ

Науковий керівник: доцент к.т.н.

\_\_\_\_\_ Кожем'яко А. В.

Студент групи 1КІ-186

\_\_\_\_\_ Савенко А. В.

## 1 Підстава для використання бакалаврської кваліфікаційної роботи (БДР)

1.1 Актуальність розробки полягає у вирішенні потреб у створенні корпоративних Web-сайтів, а також через збільшення використання у корпоративному секторі дистанційної форми праці

1.2 Наказ про затвердження теми бакалаврської дипломної роботи.

## 2 Мета і призначення БДР

2.1 Мета проекту — розробка Web-сайту, який буде надавати доступ до інформації цілодобово.

2.2 Призначення розробки — виконання бакалаврського дипломного проекту із майбутнім впровадженням та розвитком продукту.

## 3 Вихідні дані для виконання БДР

3.1 Характеристика предметної області

3.2 Аналіз та порівняння популярних CMS

3.2 Розробка інтерфейсу та структури Web-сайту.

## 4 Вимоги до виконання БДР

Головна вимога — розробити зручний інтерфейс до Web-сайту, використовуючи усі здобуті навчки та знання.

## 5 Етапи БДР та очікувані результати

Етапи роботи та очікувані результати приведено в Таблиці А.1.

## 6 Матеріали, що подаються до захисту БДР

До захисту подаються: пояснювальна записка БДР, ілюстративні матеріали, протокол попереднього захисту БДР на кафедрі, відгук наукового керівника, анотації до БДР українською та іноземною мовами, довідка про відповідність оформлення БДР діючим вимогам.

Таблиця А.1 — Етапи БДР

№ з/п	Назва етапів виконання комплексної бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Отримання завдання	11.03.22	
2	Збір інформації	12.03-15.03.22	
3	Характеристика предметної області	16.03-18.03.22	розділ 1
4	Порівняльна характеристика найпопулярніших CMS	20.03-26.03.22	розділ 2
5	Розробка бази даних систем	28.03-03.04.22	розділ 3
6	Розробка структури та веб-дизайну Web-сайту	6.04-10.04.22	розділ 3
7	Функціонально-вартісний аналіз продукту	15.04-20.04.22	розділ 4
8	Попередній захист БДР	17.05-18.05.22	
9	Перевірка роботи на плагіат	19.06.22	
10	Нормоконтроль БДР	20.06.22	
11	Рецензування БДР	20.06.22	

## 7 Порядок контролю виконання та захисту БДР

Виконання етапів графічної та розрахункової документації БДР контролюється науковим керівником згідно зі встановленими термінами. Захист БДР відбувається на засіданні Екзаменаційної комісії, затвердженої наказом ректора.

## 8 Вимоги до оформлювання та порядок виконання БДР

При оформлюванні БДР використовуються:

— ДСТУ 3008 : 2015 «Звіти в сфері науки і техніки. Структура та правила оформлювання»;

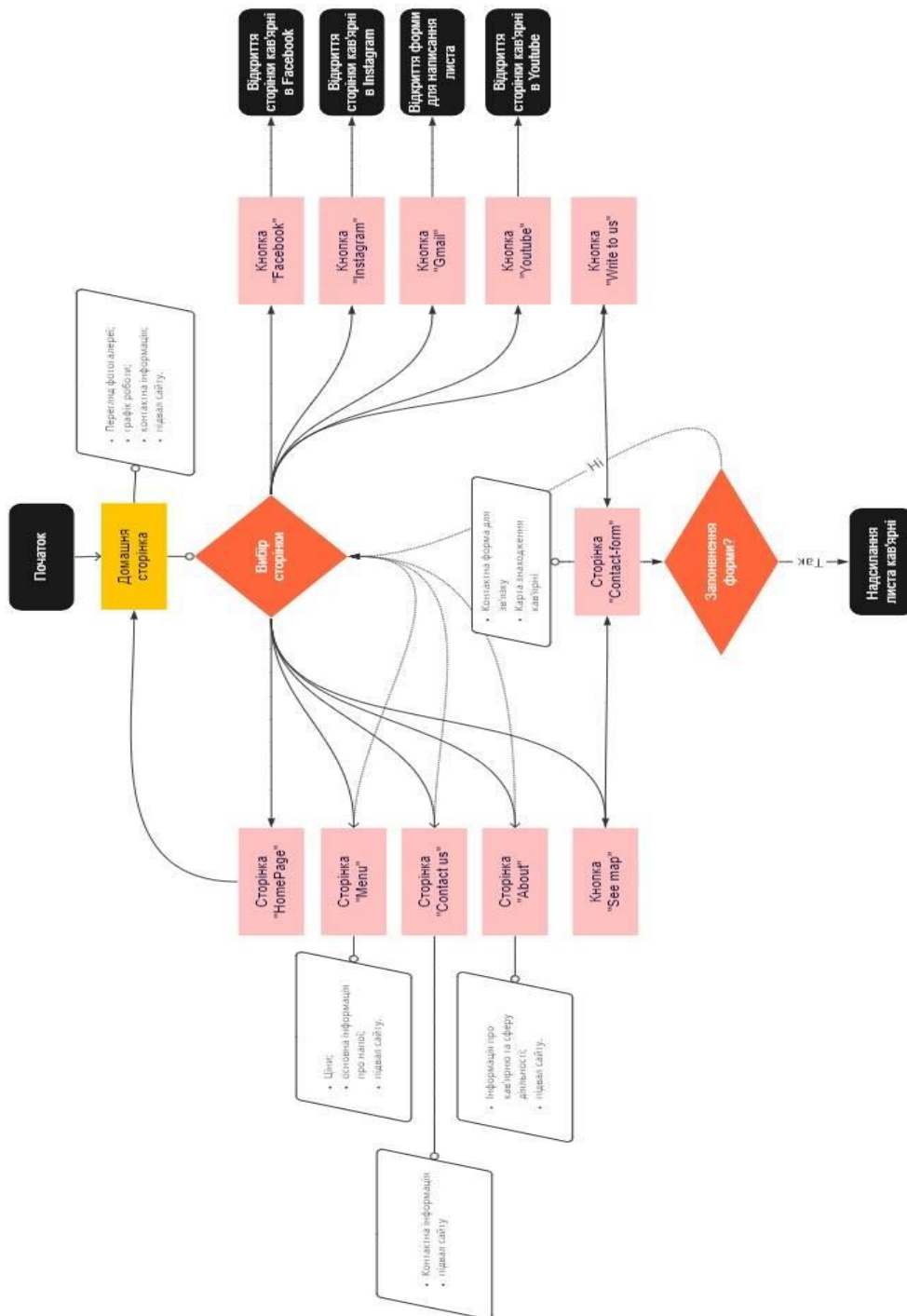
— ДСТУ 8302 : 2015 «Бібліографічні посилання. Загальні положення та правила складання»;

— ГОСТ 2.104-2006 «Єдина система конструкторської документації. Основні написи»;

— документами на які посилаються у вище вказаних.

## ДОДАТОК Б

### Інтерфейс Web-сайту



miro

Рис Б.1 — Функціональну схему web-сторінки



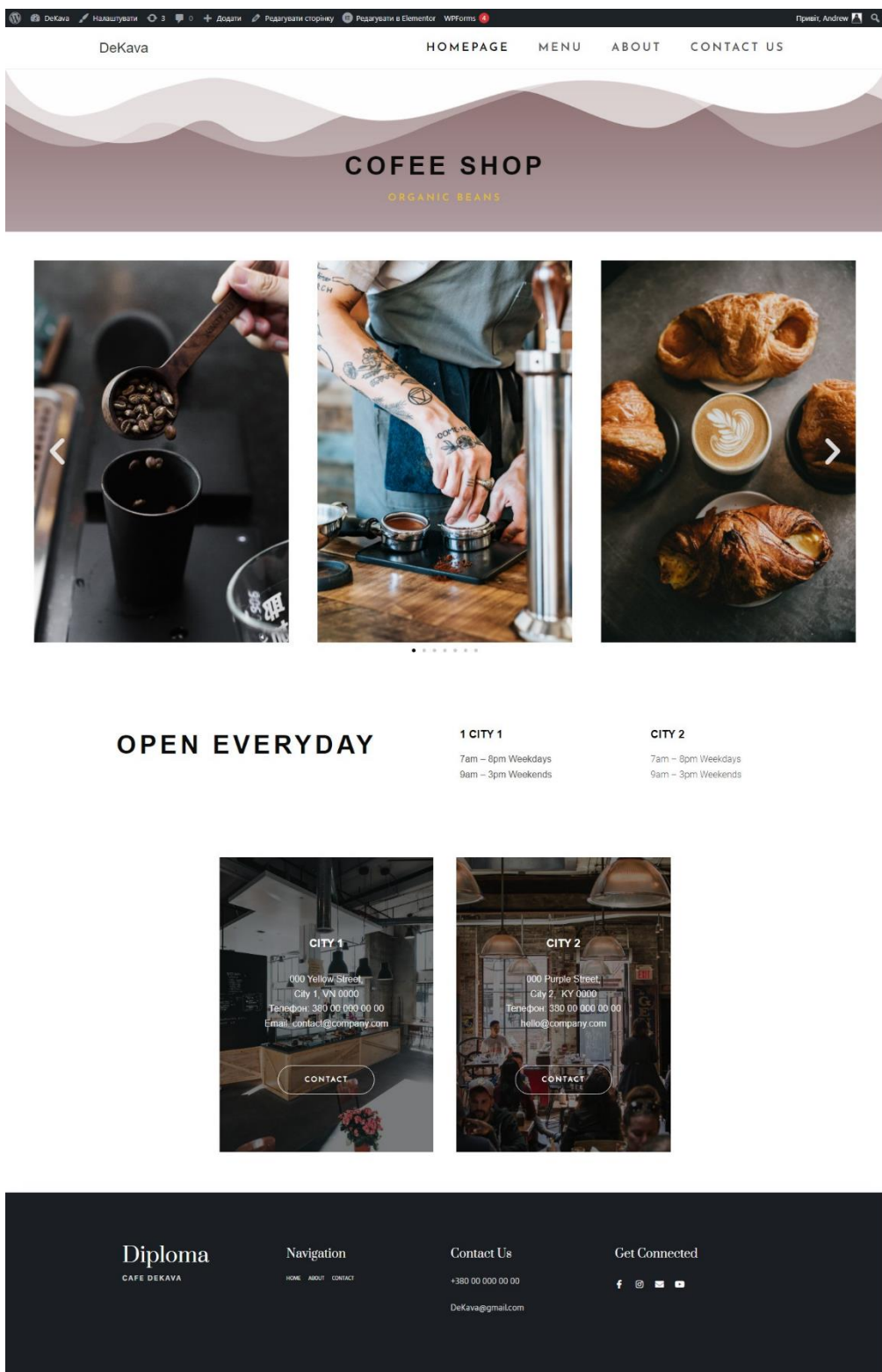


Рисунок Б.2 – Головна сторінка Web-сайту

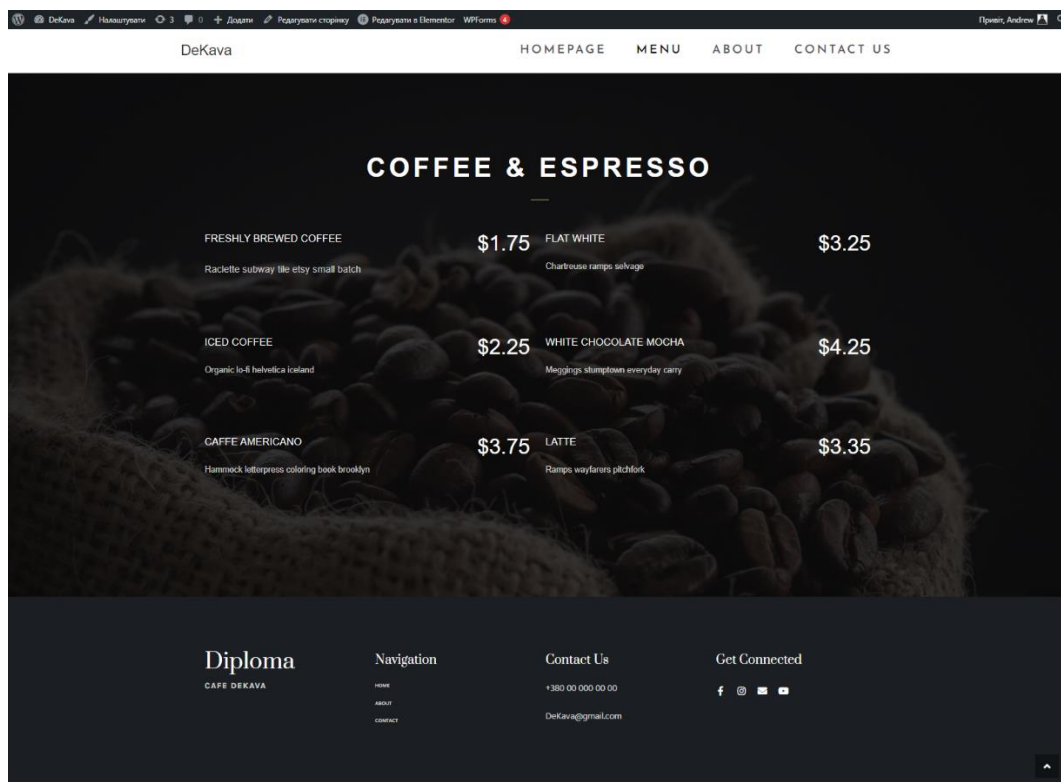


Рисунок Б.3 — Сторінка з меню

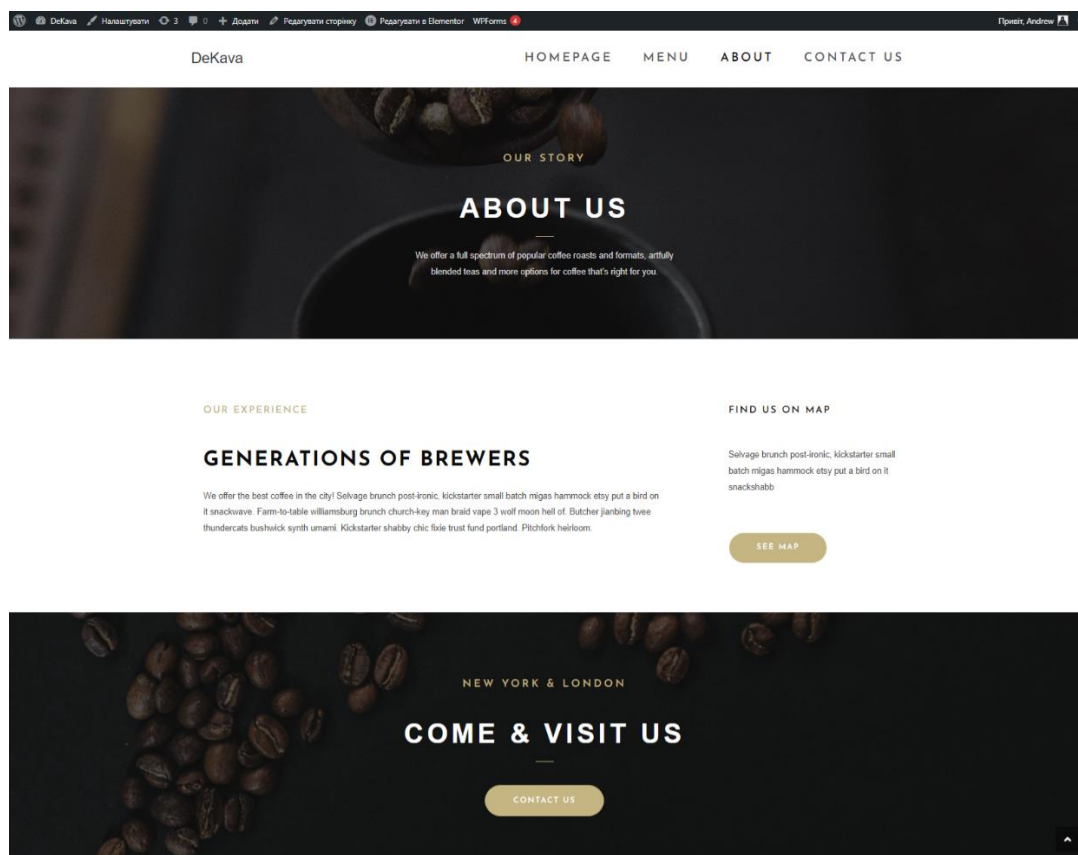


Рисунок Б.4 — Сторінка з описом





## ДОДАТОК Г

## Порівняльна таблиця

Таблиця Г.1 — порівняльна характеристика систем керування вмістом

Критерії	Drupal	Joomla!	WordPress
<b>База даних</b>	MySQL, Postgres SQL	MySQL, MS SQL, Postgres SQL, InnerDB	MySQL
<b>Веб—сервер</b>	Apache	Apache, IIS, nginx	Apache
<b>Застосування</b>	Блоги, портали, складні інформаційні сайти, форуми	Електронні магазини, фотогалереї, основа соціальної мережі, корпоративні сайти	Від блогів до інтернет- магазинів
<b>Установка</b>	Первісно вбудований jQuery, можливість відключення за необхідністю	Для установки будь- якої JavaScript бібліотеки, досить встановити відповідний модуль	Потребує установки JS бібліотек, які необхідні деяким модулям
<b>Безпека</b>	Один з найбезпечніших безкоштовних движків	Чудовий захист але багато модульність може стати загрозою для роботи.	За допомогою плагінів є можливість максимального захисту, з малим ризиком взлому.
<b>Можливості за замовчуванням</b>	Велика кількість плагінів, можна створити і форум	Без єдиного плагіна можна створити працюючий сайт	На такій збірці можна побудувати будь-який сайт з невеликим навантаженням

## Продовження таблиці Г.1

<b>Критерії</b>	<b>Drupal</b>	<b>Joomla!</b>	<b>WordPress</b>
<b>Розміщення блоків</b>	Можна виводити блоки в будь-які позиції, які дозволені шаблоном	Крім заданих шаблоном позицій, можна створювати свої позиції для блоків	Суворо закріплено в шаблоні, но можна змінювати за допомогою модулів.
<b>Багатомовність</b>	Присутня, але є складність в управлінні сайтом	Присутня, компонент JoomFish! статті, мета теги і компоненти на безліч мов	Присутня, але переклад не завжди читабельний і професійний
<b>Відновлення об'єктів</b>	+	+	-
<b>Редактортексту</b>	Відсутність візуального редактору	Редактори TinyMCE, Code Mirror и JCE гарантують зручну роботу з текстом, таблицями і зображеннями	Редактор містить всі необхідні функції для простого оформлення тексту і зображень на сайті
<b>Розміри сайту</b>	Розміри сайту маленькі	Сайту може вирости до великих розмірів	Розміри сайту невеликі

Продовження таблиці Г.1

<b>Критерії</b>	<b>Drupal</b>	<b>Joomla!</b>	<b>WordPress</b>
<b>Відновлення об'єктів</b>	+	+	-
<b>Система документообігу</b>	+	+	-
<b>Чат</b>	+	+	-
<b>Кешування сторінок</b>	+	+	+
<b>Особливості</b>	Потужня таксономія, можливість по розмітці, категоризації	Розроблялася в якості платформи для спільнот з акцентом на соціальні функції	Ідеально підходить для технічно не підготовлених користувачів
<b>Розміщення меню</b>	Будь-яку позиція для відображення меню	Крім вибору будь-якої позиції в шаблоні, можна змінювати його відображення в налаштуваннях	Місце і стиль меню суворозакріплені в шаблоні
<b>Пошук</b>	+	+	+
<b>Платіжні системи</b>	+	+	+

