

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

**«Веб-застосунок "dwnclimat" для покупки та доставки
кондиціонерів по Україні»**

Виконав студент групи ПЗ-20-3
ПОМАРАНЕЦЬ Роман Миколайович

Керівник роботи:
ТУРЕНКО Єлизавета Єгорівна

Рецензент:
БАКАЛЮК Тетяна Анатоліївна

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

Зав. кафедри інженерії
програмного забезпечення

Тетяна ВАКАЛЮК

«26» лютого 2024 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **ПОМАРАНЕЦЬ Роман Миколайович**

Керівник роботи: **ТУРЕНКО Єлизавета Єгорівна**

Тема роботи: **«Веб-застосунок "dwnclimat" для покупки та доставки кондиціонерів по Україні»,**

затверджена Наказом закладу вищої освіти від «23» лютого 2024 р. №74/с

Вихідні дані для роботи: для виконання роботи використовувалися наукові статті та публікації щодо розробки веб-застосунків, огляди існуючих систем для покупки та доставки кондиціонерів, а також сучасні веб-технології та інструменти для розробки

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Туренко Є.Є,	02.03.2024р.	02.03.2024р.
2	Туренко Є.Є,	04.04.2024р.	04.04.2024р.
3	Туренко Є.Є,	02.05.2024р.	02.05.2024р.

РЕФЕРАТ

Випускна кваліфікаційна робота бакалавра складається з веб-застосунку для покупки та доставки кондиціонерів "dwnclimat" та пояснювальної записки. Пояснювальна записка до випускної роботи містить 70 сторінок, 32 ілюстрацій та 1 таблиця.

Метою роботи є створення платформи для покупки та доставки кондиціонерів з використанням сучасних технологій, що забезпечить зручність для користувачів і оптимізацію логістичних процесів.

Архітектура платформи базується на сучасних веб-технологіях, що забезпечує надійність і масштабованість системи. Веб-застосунок включає функціонал для пошуку, фільтрації та вибору товарів, а також підтримує онлайн-оплату та відстеження доставки.

Додаток має зручний інтерфейс, який дозволяє користувачам легко знаходити потрібні товари, здійснювати покупки та замовляти доставку.

КЛЮЧОВІ СЛОВА: ВЕБСЕРВІС, ЛОГІСТИКА, ОНЛАЙН-МАГАЗИН, ПЛАТФОРМА, ВЕБ-ЗАСТОСУНОК, ЕЛЕКТРОННА КОМЕРЦІЯ, ДОСТАВКА.

					ІПЗ.КР.Б-121-24-ПЗ		
Зм.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Помаранець Р.М			Веб-застосунок "dwnclimat" для покупки та доставки кондиціонерів по Україні	Літ.	Арк.
Керівник		Туренко Є.С					3
						Житомирська політехніка, група ІПЗ-20-3	
н. контр.		Тичина О.П.					
Зав. каф.		Вакалюк Т.А.					

ABSTRACT

The bachelor's final qualification consists of a web bandage for buying and delivering Dwnclimat air conditioning and an explanatory note. The explanatory note for graduation contains 70 pages, 32 illustrations, and 1 table.

The purpose of the work is to create a platform for purchasing and delivering air conditioners using modern technologies, which will provide users convenience and optimize logistics processes.

The platform architecture is based on modern web technologies, ensuring the system's reliability and scalability. The web fitting includes functionality for searching, filtering, and selecting goods, as well as supporting online payment and tracking delivery.

The application has a convenient interface that allows users to easily find the right goods, make purchases, and order delivery.

KEYWORDS: WEB SERVICE, LOGISTICS, ONLINE STORE, PLATFORM, WEB SUPPORT, E-COMMERCE, DELIVERY.

					III3.KP.Б-121-24-П3	
						4

ЗМІСТ

РЕФЕРАТ.....	3
ABSTRACT.....	4
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ РИНКУ ТА ПОТРЕБ КОРИСТУВАЧІВ.....	8
1.1. Постановка задачі.....	9
1.2. Аналіз аналогів програмного продукту.....	10
1.3. Обґрунтування вибору інструментальних засобів та вимоги до апаратного.....	14
Висновки до першого розділу.....	19
РОЗДІЛ 2. ПРОЕКТУВАННЯ ВЕБ-ЗАСТОСУНКУ.....	21
2.1. Визначення варіантів використання об'єктно-орієнтованої системи..	21
2.2. Проектування та реалізація алгоритмів роботи системи.....	29
2.3. Реалізація функціоналу системи веб-застосунку.....	31
Висновки до другого розділу.....	32
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ВЕБ-ЗАСТОСУНКОМ DNWCLIMAT.....	33
3.1 Порядок встановлення та налаштування параметрів застосунку.....	33
3.2 Структура інтерфейсу веб-застосунку.....	34
3.3 Тестування застосунку.....	44
Висновки до третього розділу.....	47
ВИСНОВКИ	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	50
ДОДАТКИ.....	52

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

API – прикладний програмний інтерфейс

БД – база даних

ПЗ – програмне забезпечення

УМ - Умовне позначення

API - Application Programming Interface (інтерфейс програмування застосунків)

UI - User Interface (користувацький інтерфейс)

UX - User Experience (користувацький досвід)

HTML - HyperText Markup Language (мова розмітки гіпертексту)

CSS - Cascading Style Sheets (каскадні таблиці стилів)

JS - JavaScript (мова програмування JavaScript)

SQL - Structured Query Language (мова структурованих запитів)

MVC - Model-View-Controller (модель-вид-контролер)

JSON - JavaScript Object Notation (формат обміну даними)

REST - Representational State Transfer (репрезентативна передача стану)

HTTP - HyperText Transfer Protocol (протокол передачі гіпертексту)

HTTPS - HyperText Transfer Protocol Secure (захищений протокол передачі гіпертексту)

URL - Uniform Resource Locator (уніфікований покажчик ресурсів)

ВСТУП

Актуальність теми полягає у тому, що сучасні технології відіграють важливу роль у покращенні якості життя людей, особливо у сфері електронної комерції. Веб-застосунки, що забезпечують можливість покупки та доставки товарів, значно спрощують процес придбання необхідних речей. В Україні ринок кондиціонерів є досить розвиненим, і створення спеціалізованого веб-застосунку для покупки та доставки цих пристроїв є актуальним та затребуваним.

Метою випускної кваліфікаційної роботи бакалавра є розробка веб-застосунку "dwnclimat", який дозволить користувачам зручно купувати кондиціонери та замовляти їх доставку по всій території України.

Для реалізації поставленої мети потрібно виконати наступні завдання:

1. Провести аналіз ринку та визначити функціональні вимоги до веб-застосунку.
2. Обрати засоби та технології для проектування і розробки веб-застосунку.
3. Спроектувати та реалізувати архітектуру веб-застосунку.
4. Провести тестування та забезпечити розгортання веб-застосунку.

Об'єктом дослідження є процес розробки веб-застосунку для електронної комерції в сфері продажу та доставки кондиціонерів..

Предметом дослідження є методи та технології розробки веб-застосунків, що використовуються для створення платформи "dwnclimat" тобто пошук кондиціонерів по Україні, замовлення, доставка, та покупка продукту.

РОЗДІЛ 1. АНАЛІЗ РИНКУ ТА ПОТРЕБ КОРИСТУВАЧІВ

1.1. Постановка задачі

Сучасний ринок кондиціонерів в Україні потребує інноваційних підходів до продажу та доставки продукції.

Основна задача розробки веб-застосунку "dwnclimat" полягає у створенні платформи, яка забезпечить зручний, швидкий та безпечний спосіб покупки кондиціонерів онлайн з можливістю їх доставки по всій території країни.

Це включає розробку зручного користувацького інтерфейсу, забезпечення надійного функціоналу для вибору та замовлення товарів, інтеграцію з платіжними системами та службами доставки.

Очікувані результати:

Веб-застосунок з інтуїтивно зрозумілим та привабливим інтерфейсом на базі WordPress.

Адмін-панель WordPress з можливістю редагування кожного елемента веб-застосунку.

База даних, що зберігає інформацію про кондиціонери, користувачів, замовлення та статуси доставки.

Надійна система обробки онлайн-платежів.

Інтеграція зі службами доставки для автоматичного відстеження та управління логістикою.

Функціонал для порівняння товарів, фільтрації та сортування, що спрощує вибір кондиціонерів.

Модуль для управління користувацькими акаунтами та історією замовлень.

Основні етапи розробки:

1. Планування та аналіз вимог:

- Збір та аналіз вимог від потенційних користувачів.
- Вивчення ринку та аналіз аналогів.
- Визначення основних функціональних вимог та технічних специфікацій.

2. Проектування:

- Розробка архітектури системи на базі WordPress.
- Проектування бази даних.
- Створення макетів та прототипів користувацького інтерфейсу.

3. Розробка:

- Встановлення та налаштування WordPress.
- Розробка та інтеграція спеціальних плагінів для реалізації необхідного функціоналу (каталог товарів, кошик, замовлення, відстеження доставки тощо).
- Розробка та налаштування теми WordPress для створення інтерфейсу користувача.
- Інтеграція з платіжними системами та службами доставки.

4. Тестування:

- Проведення модульного, інтеграційного та системного тестування.
- Тестування користувацького інтерфейсу на зручність та функціональність.
- Усунення знайдених помилок та недоліків.

5. Розгортання та запуск:

- Налаштування серверного середовища для розгортання веб-застосунку.
- Перенесення даних у продуктивне середовище.
- Запуск веб-застосунку та моніторинг його роботи в реальному часі.

6. Підтримка та вдосконалення:

- виправлення помилок та недоліків, виявлених після запуску.
- додавання нових функцій та вдосконалення існуючих на основі відгуків користувачів.
- Постійний моніторинг продуктивності та безпеки системи.

1.2. Аналіз аналогів програмного продукту

Розробка веб-застосунку для покупки та доставки кондиціонерів по Україні потребує глибокого аналізу конкурентів на ринку. Нижче наведено декілька прикладів подібних сайтів з їх основними перевагами та недоліками.

Одним з таких сайтів є Rozetka (rozetka.com.ua) (рис. 1.1).

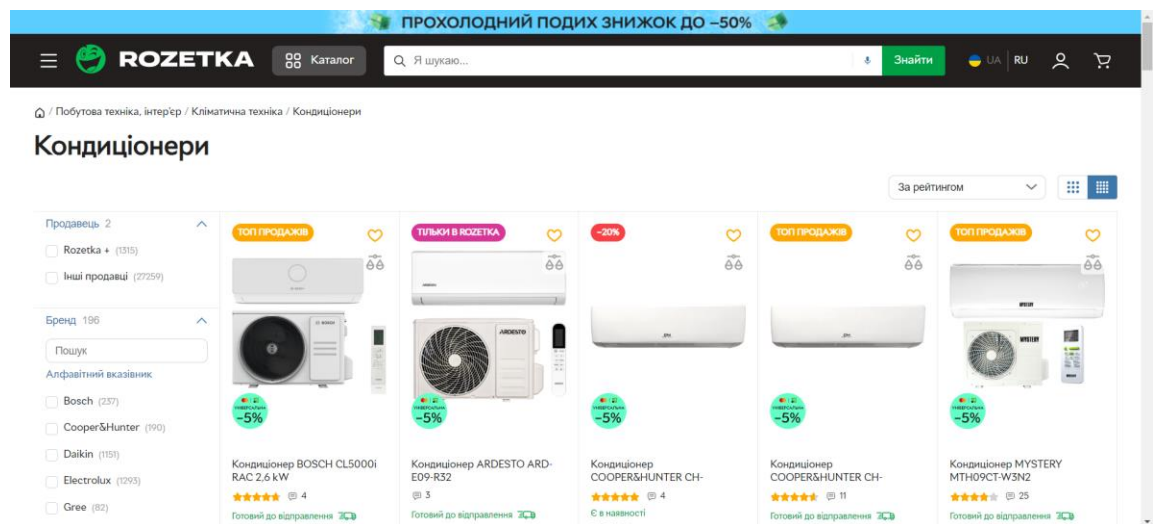


Рис. 1.1. Онлайн-магазин Rozetka [1]

Rozetka є одним з найбільших інтернет-магазинів в Україні, що пропонує широкий асортимент товарів, включаючи кондиціонери. Інтерфес інтуїтивно зрозумілий, з простими навігаційними елементами. Сучасний та професійний дизайн, з використанням нейтральних кольорів та акцентів зеленого кольору.

Переваги:

- Зручний інтерфейс суми.
- Широкий асортимент
- Велика кількість відгуків

Недоліками є:

- Складність навігації через великий обсяг інформації
- Довгий час очікування відповіді від служби підтримки

Comfy (comfy.ua) – є популярним інтернет-магазином техніки та електроніки в Україні, який також пропонує кондиціонери. Часто акції та знижки: Часті акції, знижки та бонуси для постійних клієнтів. Дизайн світлий та яскравий, з використанням червоних та білих кольорів, інтерфейс зручний та простий у використанні

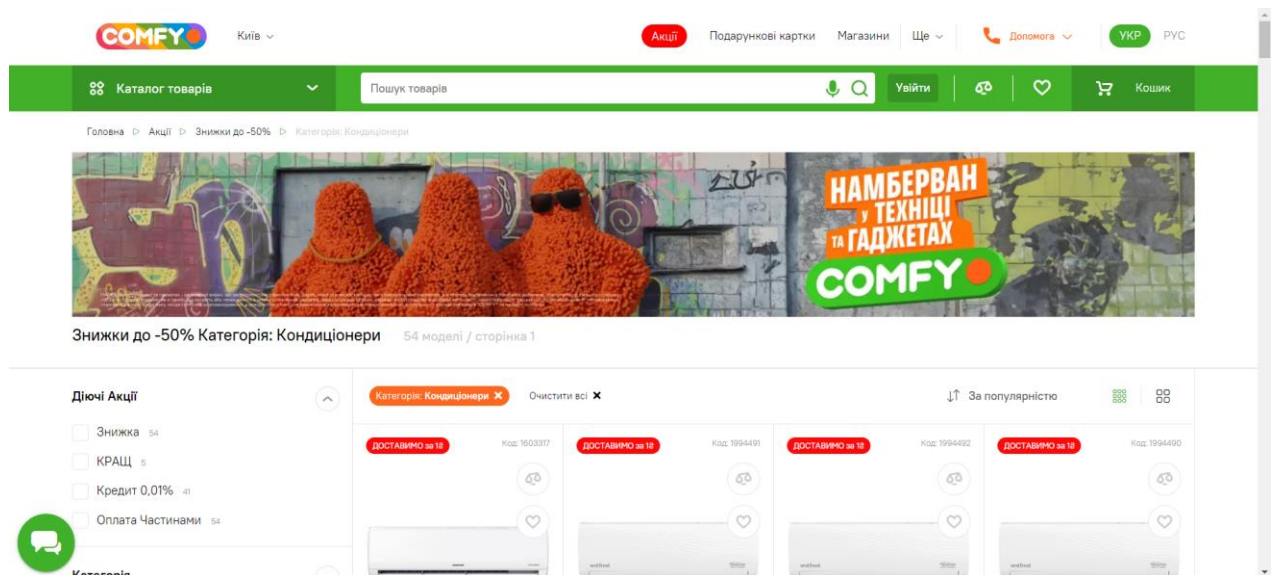


Рис. 1.2. Онлайн-магазин техніки Comfy[2]

Основними перевагами Comfy є:

- Наявність мобільного додатку
- Зручна система оплати
- Часті акції та знижки

Недоліки:

- Обмежений асортимент кондиціонерів

Foxtrot - один з лідерів на ринку електроніки та побутової техніки в Україні, що пропонує широкий вибір кондиціонерів. Інтерфейс інтуїтивно зрозумілий, з добре структурованими категоріями та фільтрами. Наявність програми лояльності для постійних клієнтів з можливістю накопичення бонусів. Можливість отримати консультацію від спеціалістів щодо вибору та установки кондиціонера.

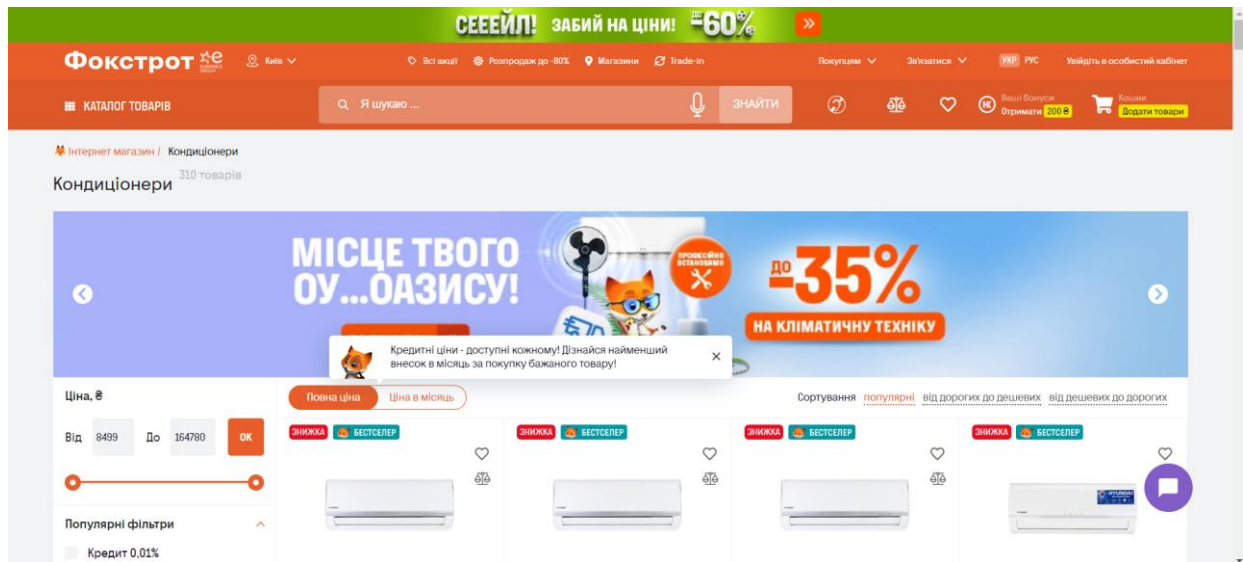


Рис. 1.3. Веб-застосунок Фокстрот [3]

Основні переваги Фокстрот:

- Розширені гарантії;
- Програма лояльності;
- Професійні консультації;

Основні недоліки Фокстрот:

- Вищі ціни порівняно з конкурентами;
- Складність використання сайту через перевантаженість інформацією;

Allo – популярний інтернет-магазин електроніки та побутової техніки в Україні, що пропонує різноманітний асортимент кондиціонерів. Проста та зручна навігація, з акцентом на пошук та категорії товарів. З великою кількістю інформаційних матеріалів та відеооглядів товарів.

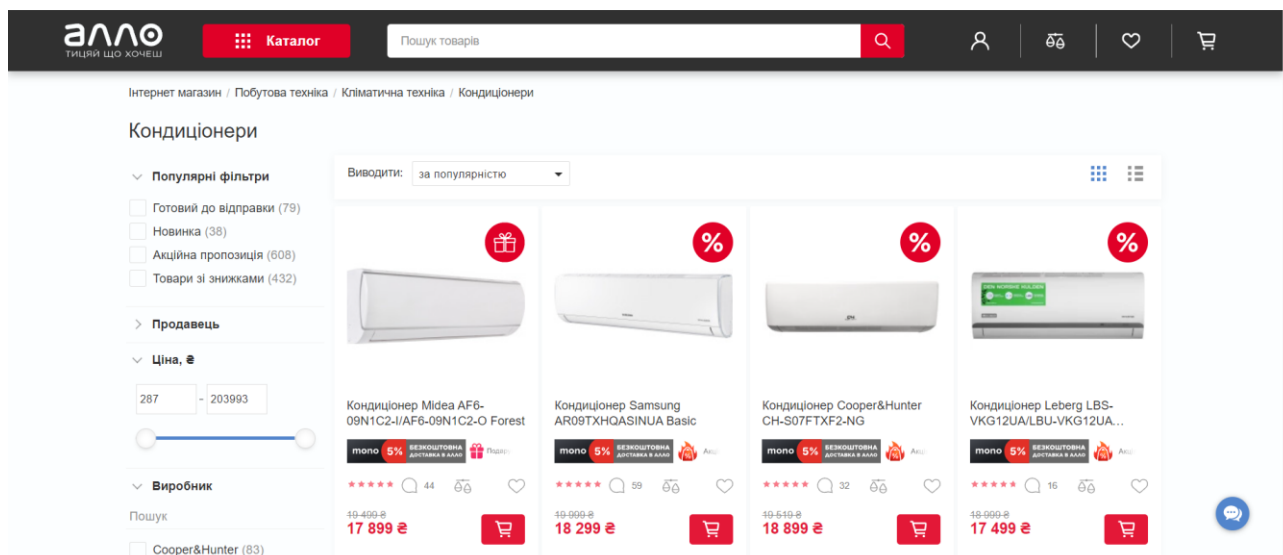


Рис. 1.4. Інтернет магазин побутової техніки Allo[4]

Основні переваги Allo:

- Швидка доставка по всій Україні;
- Зручний пошук;
- Велика кількість інформаційних матеріалів;

Основні недоліки Crowdfunder:

- Проблеми з обслуговуванням замовлень;
- Складність процесу реєстрації та оформлення замовлення;

Виходячи з наявних переваг та недоліків, складемо порівняльну таблицю вищезгаданих програмних аналогів (табл. 1.1).

Таблиця 1.1

Таблиця аналізу існуючих аналогів веб-застосунків для покупки та доставки кондиціонерів

Характеристика	Rozetka	Comfy	Foxtrot	Allo
Кросплатформенність	+	+	+	+
Доступність створення замовлення	+	+	+	+
Підтримка будь-якого типу замовлення	+	+	+	+
Зручність використання	+	+	+	+
Децентралізація	-	-	-	-
Можливість отримати товар без передоплати	+	-	+	-
Прозорість ціни	+	+	+	+

Переглянувши всі веб-застосунки, можна виділити основні відмінності: широкий асортимент товарів, зручність навігації та інтерфейсу, прозорість цінової політики, можливість отримати товар без повної передоплати, підтримка різних способів оплати, швидкість доставки, наявність відгуків та рейтингу товарів, додаткові гарантії та послуги, програми лояльності, мобільні додатки, інформаційна підтримка.

Враховуючи інформацію, отриману при дослідженні, було прийнято рішення створити наш магазин кондиціонерів із такими перевагами: широкий асортимент товарів, зручний та інтуїтивно зрозумілий інтерфейс, прозорість цінової політики, можливість отримати товар без повної передоплати, підтримка різних способів оплати, швидка доставка по Україні, наявність відгуків та рейтингу товарів, додаткові гарантії та послуги з обслуговування, програма лояльності для постійних клієнтів.

1.3 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

Вебсайт буде побудований на платформі WordPress з використанням основного плагіну WooCommerce. Серверна частина буде написана на PHP, клієнтська частина використовуватиме HTML, CSS та JavaScript. Для зберігання даних буде використовуватися база даних MySQL. Апаратне забезпечення включає сервер з мінімум 512 Мб оперативної пам'яті, процесором не менше 1.5 ГГц, та 5 ГБ вільного місця і більше. Сайт буде захищений за допомогою безкоштовного SSL сертифікату Let's Encrypt.

WordPress Проаналізуємо основні поняття, щоб у нас не виникало питання що таке cms wordpress і з чого він складається.

Отже Плагін - доповнення до програми, який допоможе створити додаткові функції для користувача і розширить можливості сайту. Всередині кожного міститься програмний код. Тема або шаблон сайту-це своєрідна сполучна ланка сайту, яка об'єднує зовнішній вигляд, дизайн сайту, і його зміст. Це ж пояснення ви отримаєте на пошуковий запит: "cms wordpress це?" Просто аббревіатура CMS розшифровується як: Content Management System - система управління контентом.

Адже все, створене на сайті- це ні що інше як CMS, двигун ресурсу. Краще сказати так WordPress CMS- двигун ресурсу. Розглянемо п'ять плюсів цієї програми і кому вона потрібна.

Переваги WordPress

Доступність. Це дуже крута система з безкоштовним доступом. Її мета допомогти новачкам в створенні свого проекту або блогу. . Розробники вклали максимум інтелекту і зусиль для того, щоб користувач нічого не зіпсував, тикаючи в різні кнопки. Більш того, він отримує різні підказки на що потрібно натискати.

Кросплатформеність. WordPress дозволяє вам використовувати ваш доступ абсолютно на будь-якому гаджеті. Якщо ви підключені до інтернету і пам'ятайте свій пароль, у вас не виникне проблем з входом в ваш кабінет. Вбудований редактор дозволяє додавати, домальовувати, ретушувати, дописувати і міняти в цілому контент парою кліків.

Система абсолютно безпечна і стійка до злому. І головне перевага-вбудована підтримка оптимізації пошуку, що дозволяє ефективно просувати свій сайт і піднімати його в рейтингу на вищій щабель. І, звичайно, постійна співпраця з RSS стрічкою новин і соціальними мережами, що дозволяє вам бути завжди в Стрімі.

					ІПЗ.КР.Б-121-24-ПЗ	
						15

PHP (Серверна частина)

PHP (Hypertext Preprocessor) є одним з найпопулярніших мов програмування для веб-розробки, що дозволяє створювати динамічні веб-сторінки та інтерактивні веб-додатки.

Переваги PHP:

Популярність та Поширеність: PHP широко використовується, що означає велику кількість ресурсів, документації та прикладів коду.

Сумісність з Базами Даних: PHP добре інтегрується з багатьма базами даних, такими як MySQL, PostgreSQL, SQLite та іншими.

Велика Спільнота та Підтримка: Існує велика кількість фреймворків (наприклад, Laravel, Symfony) та бібліотек, що полегшують розробку.

Хостинг та Інфраструктура: Багато веб-хостингів підтримують PHP за замовчуванням, що робить його зручним для розгортання.

HTML (HyperText Markup Language) є основною мовою розмітки для створення веб-сторінок.

Переваги HTML:

Основна Мова для Веб: є стандартом для створення веб-сторінок, тому його знання є необхідним для будь-якого веб-розробника.

Простота: HTML простий для вивчення та використання, що робить його доступним для новачків.

Сумісність: HTML підтримується всіма веб-браузерами, що забезпечує кросбраузерну сумісність.

Гнучкість: HTML можна використовувати разом з CSS та JavaScript для створення більш складних та інтерактивних веб-сторінок.

CSS (Cascading Style Sheets) використовується для опису зовнішнього вигляду веб-сторінки, включаючи шрифти, кольори, розміщення елементів та інші аспекти стилізації.

Переваги CSS:

Розділення Змісту та Презентації: дозволяє розділити структуру (HTML) та стиль (CSS), що робить код більш чистим та зручним для підтримки. Це розділення сприяє кращій підтримці та масштабованості веб-додатків.

Гнучкість та Можливості Стилізації: надає безліч можливостей для стилізації, включаючи шрифти, кольори, розташування елементів та анімації. Використання селекторів, класів та ID для точного налаштування стилів окремих елементів або груп елементів.

Кросбраузерна Сумісність: підтримується всіма сучасними браузерами. CSS підтримується всіма сучасними веб-браузерами, що забезпечує коректне відображення стилів на різних платформах. Використання стандартів CSS, затверджених W3C, гарантує стабільність та узгодженість у відображенні стилів.

Медіа-запити: CSS дозволяє створювати адаптивний дизайн, що коригується під різні розміри екранів та пристроїв. Використання медіа-запитів для налаштування стилів залежно від характеристик пристрою (наприклад, роздільна здатність екрану, орієнтація).

JavaScript є клієнтською мовою програмування, яка використовується для додавання інтерактивності до веб-сторінок. Він дозволяє створювати динамічні елементи, обробляти події користувача та взаємодіяти з сервером без необхідності перезавантаження сторінки.

Переваги JavaScript:

Інтерактивність: JavaScript дозволяє створювати динамічні та інтерактивні елементи на веб-сторінках, такі як анімації, спливаючі вікна, форми, слайдери та інше. Обробка подій користувача (наприклад, кліки, введення тексту, переміщення миші) в реальному часі.

Кросплатформеність: працює на всіх сучасних веб-браузерах без необхідності встановлення додаткових плагінів чи програмного забезпечення. Це забезпечує широке поширення та доступність JavaScript для всіх користувачів інтернету.

Велика Спільнота та Підтримка: JavaScript має велику та активну спільноту розробників, що постійно створює та підтримує бібліотеки, фреймворки та інструменти (наприклад, React, Angular, Vue.js). Велика кількість навчальних матеріалів, документації та онлайн-курсів для вивчення JavaScript.

Асиметричний Розподіл Даних: JavaScript дозволяє реалізувати AJAX-запити, що забезпечують асинхронний обмін даними з сервером без перезавантаження сторінки. Це покращує користувацький досвід, оскільки дозволяє оновлювати частини веб-сторінки в реальному часі.

Швидкість Виконання: Сучасні веб-браузери мають вбудовані JavaScript-движки, що забезпечують швидке виконання коду та оптимізацію роботи веб-додатків. Використання JavaScript може значно покращити продуктивність та відгук веб-сторінок.

MySQL є однією з найпопулярніших систем управління базами даних (СУБД) з відкритим кодом, яка використовується для зберігання та керування даними в багатьох веб-додатках. MySQL часто застосовується у поєднанні з PHP для створення динамічних веб-сайтів та додатків.

Переваги MySQL:

Популярність та Поширеність: MySQL використовується багатьма відомими веб-сайтами та додатками (наприклад, Facebook, Twitter, YouTube), що свідчить про її надійність та ефективність. Велика кількість ресурсів, документації та підтримки від спільноти.

Відкритий Код та Безкоштовність: MySQL є безкоштовною системою з відкритим кодом, що робить її доступною для широкого кола користувачів. Відкрита модель розробки дозволяє користувачам вносити зміни та вдосконалення у систему.

Популярність та Поширеність: MySQL використовується багатьма відомими веб-сайтами та додатками (наприклад, Facebook, Twitter, YouTube), що свідчить про її надійність та ефективність. Велика кількість ресурсів, документації та підтримки від спільноти.

Відкритий Код та Безкоштовність: MySQL є безкоштовною системою з відкритим кодом, що робить її доступною для широкого кола користувачів. Відкрита модель розробки дозволяє користувачам вносити зміни та вдосконалення у систему. MySQL забезпечує високу безпеку даних завдяки підтримці SSL для захищених з'єднань, а також функціям управління користувачами та дозволами. Регулярні оновлення та виправлення безпеки забезпечують захист від нових загроз.

Сумісність з PHP та Іншими Мовами: MySQL легко інтегрується з PHP завдяки вбудованій підтримці через розширення mysqli та PDO (PHP Data Objects). Також підтримує інші мови програмування, такі як Python, Java, Ruby, що робить її універсальним рішенням для різних проектів.

Підтримка Реплікації та Відновлення після Збоїв: MySQL підтримує реплікацію даних між серверами, що забезпечує високу доступність та надійність. Функції відновлення після збоїв дозволяють зберегти дані у випадку непередбачених обставин.

Висновки до першого розділу.

Переглянувши всі веб-застосунки, можна виділити основні відмінності: широкий асортимент товарів, зручність навігації та інтерфейсу, прозорість цінової політики, можливість отримати товар без повної передоплати, підтримка різних способів оплати, швидкість доставки, наявність відгуків та рейтингу товарів, додаткові гарантії та послуги, програми лояльності, мобільні додатки, інформаційна підтримка. Враховуючи інформацію, отриману при дослідженні, було прийнято рішення створити наш магазин кондиціонерів із такими перевагами: широкий асортимент товарів, зручний та інтуїтивно зрозумілий інтерфейс, прозорість цінової політики, можливість отримати товар без повної передоплати, підтримка різних способів оплати, швидка доставка по Україні, наявність відгуків та рейтингу товарів, додаткові гарантії та послуги з обслуговування, програма лояльності для постійних клієнтів, а також мобільний додаток для зручного замовлення товарів.

В процесі дослідження аналогічних і схожих за концепцією продуктів були проаналізовані та порівняні різні платформи з різними принципами роботи, основним функціоналом яких є створення зборів або підтримка авторів місячною підпискою. Крім цього, були сформовані основні вимоги для власної платформи.

На основі відповідних вимог та критеріїв був сформований набір основних технологій і засобів реалізації для розробки вебдодатку. Використання MySQL як системи управління базами даних у поєднанні з PHP для серверної частини та HTML, CSS і JavaScript для клієнтської частини дозволяє створювати ефективні, масштабовані та інтерактивні веб-додатки. MySQL забезпечує надійне та швидке зберігання даних, тоді як PHP, HTML, CSS та JavaScript дозволяють створювати динамічний та зручний для користувача інтерфейс.

РОЗДІЛ 2. ПРОЕКТУВАННЯ ВЕБ-ЗАСТОСУНКУ

2.1. Визначення варіантів використання та об'єктно-орієнтованої структури системи

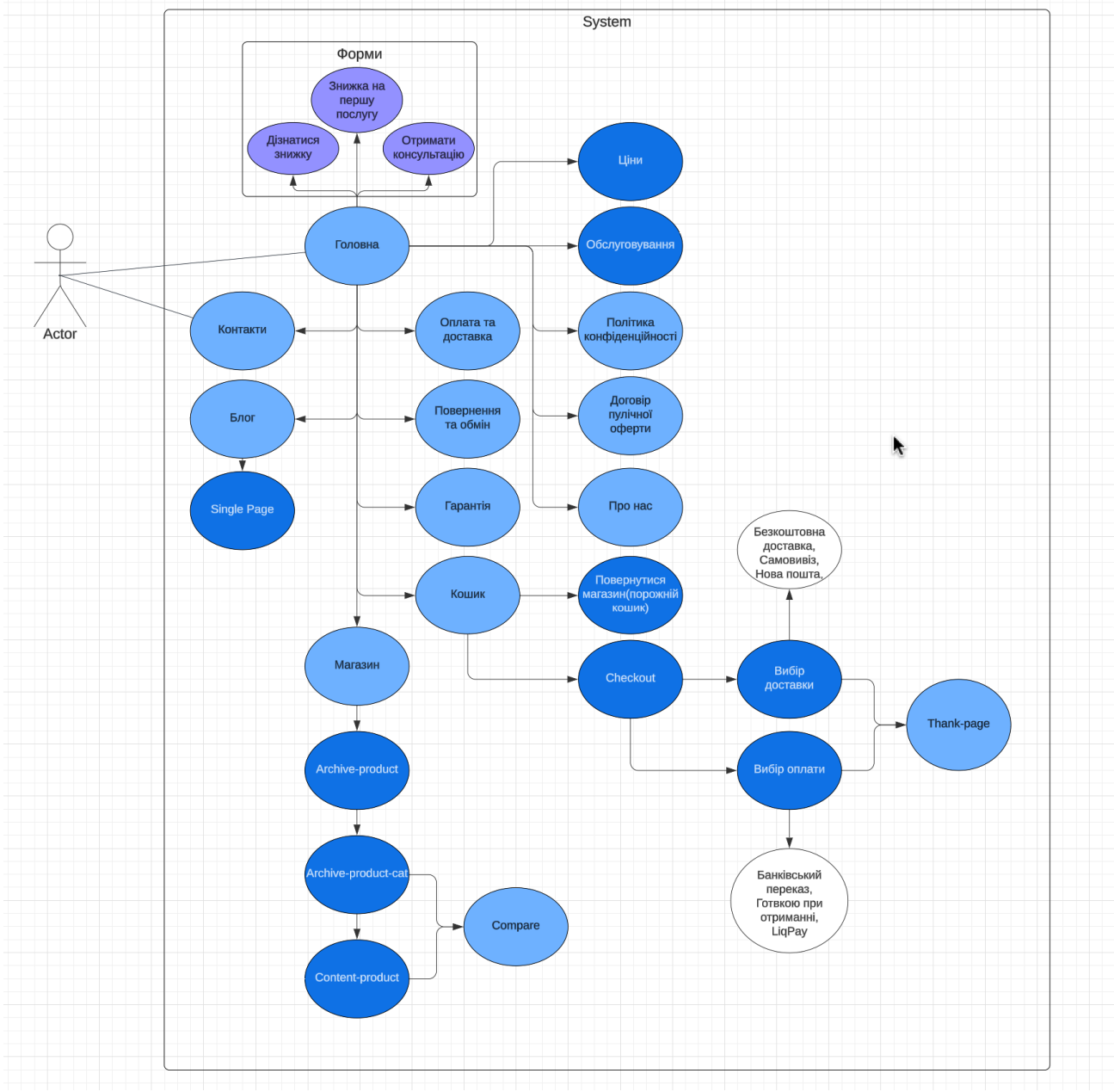


Рис. 2.1. Діаграма варіантів використання

Діаграма варіантів використання (рис.2.1) (Use Case Diagram) відображає взаємодію користувача (Actor) з різними компонентами системи. Ось детальний опис кожного елемента на діаграмі:

Актор (Actor)

- **Actor:** Представляє користувача системи, який взаємодіє з різними компонентами.

Компоненти системи

1. Головна (Main)

- Центральна сторінка, з якої користувач може переходити на інші сторінки та взаємодіяти з системою.

2. Ціни (Prices)

- Сторінка з інформацією про ціни на товари та послуги.

3. Обслуговування (Services)

- Сторінка з інформацією про обслуговування.

4. Контакти (Contacts)

- Сторінка, де користувач може знайти контактну інформацію та адресу.

5. Блог (Blog)

- Сторінка з статтями.

6. Single Page

- Односторінкова структура, яка відповідає за відображення шаблону для статті.

7. Магазин (Shop)

- Головна сторінка магазину, де користувач може переглядати категорії та товари.

8. Archive-product

- Архів товарів, де відображаються всі товари магазину

9. Archive-product-cat

- Архів товарів по категоріях.

10. Content-product

- Сторінка з детальною інформацією про товар.

11. Compare

- Сторінка з функціоналом для порівняння товарів.

12. Кошик (Cart)

- Кошик користувача з вибраними товарами.

13. Checkout

- Сторінка оформлення замовлення.

14. Вибір доставки (Delivery Choice)

- Сторінка вибору способу доставки.

1.Безкоштовна доставка, Самовивіз, Нова пошта

- 1.Опції для вибору способу доставки.

15. Вибір оплати (Payment Choice)

- Сторінка вибору способу оплати.

1.Банківський переказ, Готівкою при отриманні, LiqPay

- 1.Опції для вибору способу оплати.

16. Thank-page

- Сторінка подяки після успішного замовлення.

17. Форми (Forms)

- Включає в себе різні форми для взаємодії з користувачем.

1.Знижка на першу послугу

- 1.Форма для отримання знижки на першу послугу.

2.Отримати консультацію

- 1.Форма для отримання консультації.

3.Дізнатися знижку

- 1.Форма для отримання знижки на монтаж кондиціонерів.

Footer

1. **Політика конфіденційності (Privacy Policy)**
 - Сторінка з політикою конфіденційності.
2. **Договір публічної оферти (Public Offer Agreement)**
 - Сторінка з договором публічної оферти.
3. **Про нас (About Us)**
 - Сторінка з інформацією про компанію.
4. **Оплата та доставка (Payment and Delivery)**
 - Сторінка з інформацією про оплату та доставку товарів.
5. **Повернення та обміни (Returns and Exchanges)**
 - Сторінка з політикою повернення та обмінів.
6. **Гарантія (Warranty)**
 - Сторінка з інформацією про гарантії на товари.

Взаємозв'язки

- **Головна (Main):** Є центральною сторінкою, з якої користувач може перейти на інші сторінки, такі як Контакти, Блог, Магазин, Кошик та інші.
- **Форми (Forms):** Доступні з головної сторінки.
- **Checkout:** Має зв'язок з вибором доставки та вибором оплати.
- **Вибір доставки та Вибір оплати:** Впливають на завершення процесу оформлення замовлення.

Ця діаграма представляє основні варіанти використання, дозволяючи зрозуміти, як користувач взаємодіє з системою, та які основні функціональні можливості надаються йому.

Аналіз функціональних вимог дозволив виділити наступні сутності, що забезпечать програмну реалізацію веб-застосунку. На рисунку 2.2. наведено Діаграму класів, яка використовуються для його роботи.

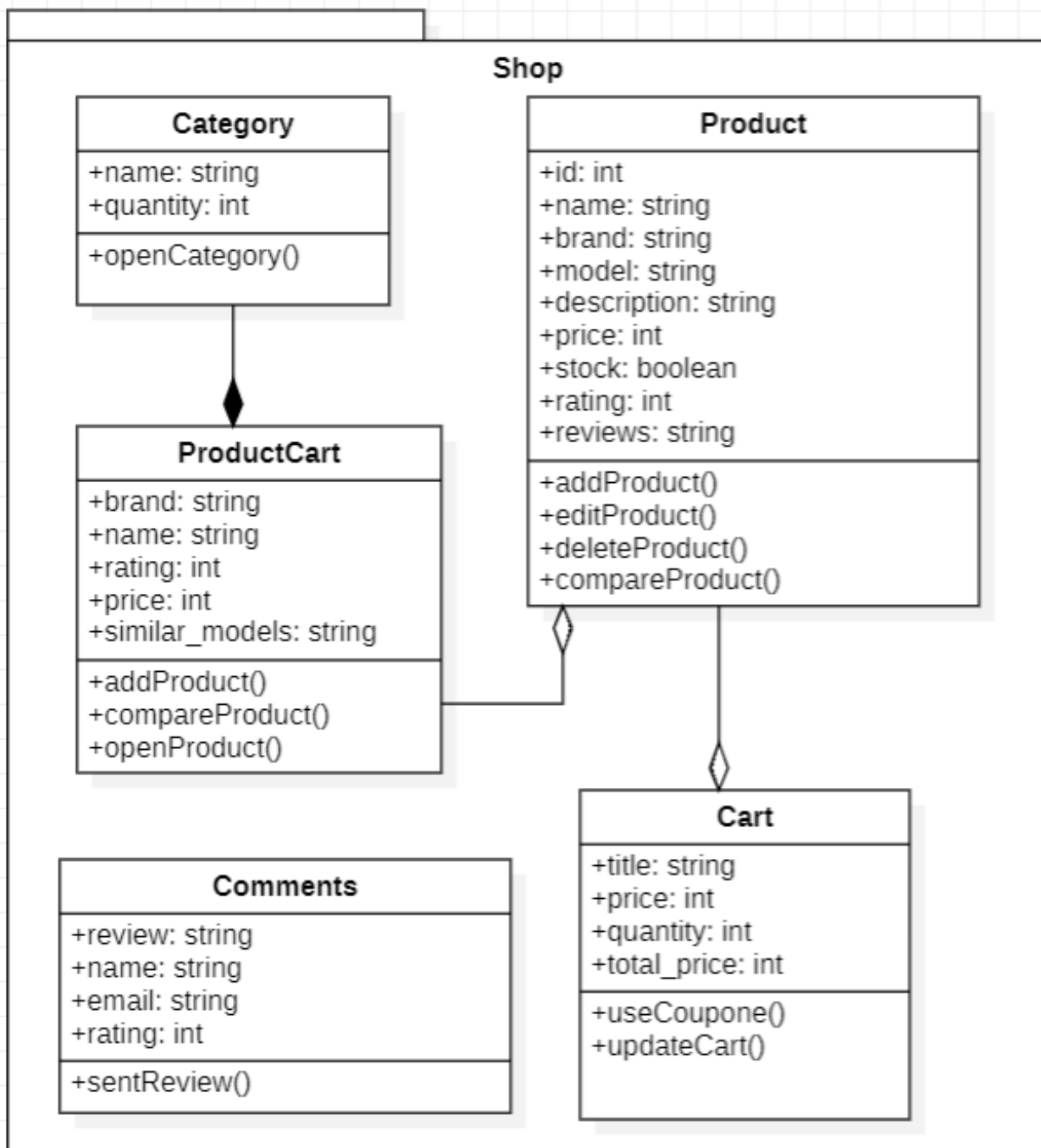


Рис. 2.2. Діаграма класів

Опис діаграми класів для веб-застосунку "dwnclimat" (рис 2.2)

Діаграма класів представляє основні класи веб-застосунку, їх атрибути, методи та взаємозв'язки між ними. Вона дозволяє зрозуміти структуру і поведінку об'єктів у системі, а також їхні взаємодії.

1. Клас Shop

Опис: Головний клас, який об'єднує всі інші класи та забезпечує їх взаємодію.

Зв'язки:

Зв'язок з класом Category: асоціація, що вказує на те, що магазин має категорії.

Зв'язок з класом Product: агрегація, яка показує, що магазин містить продукти.

Зв'язок з класом ProductCart: агрегація, що показує зв'язок між магазином та корзиною продуктів.

Зв'язок з класом Cart: агрегація, що демонструє наявність корзини для покупок у магазині.

2. Клас Category

- Атрибути:
 - name: string — назва категорії.
 - quantity: int — кількість товарів у категорії.
- Методи:
 - openCategory() — метод для відкриття категорії.
- Зв'язки:
 - Асоціація з класом ProductCart. Це свідчить про те, що категорії містять продукти, які можуть бути додані до корзини продуктів.

3. Клас Product

- Атрибути:
 - id: int — унікальний ідентифікатор продукту.
 - name: string — назва продукту.
 - brand: string — бренд продукту.
 - model: string — модель продукту.
 - description: string — опис продукту.
 - price: int — ціна продукту.
 - stock: boolean — наявність на складі.
 - rating: int — рейтинг продукту.
 - reviews: string — відгуки про продукт

- Методи:
 - addProduct() — метод для додавання продукту.
 - editProduct() — метод для редагування продукту.
 - deleteProduct() — метод для видалення продукту.
 - compareProduct() — метод для порівняння продуктів.
- Зв'язки:
 - Агрегація з класом Cart: продукти можуть бути додані до корзини.

4. Клас ProductCart

- Атрибути:
 - brand: string — бренд продукту.
 - name: string — назва продукту.
 - rating: int — рейтинг продукту.
 - price: int — ціна продукту.
 - similar_models: string — схожі моделі.
- Методи:
 - addProduct() — метод для додавання продукту до корзини.
 - compareProduct() — метод для порівняння продуктів у корзині.
 - openProduct() — метод для відкриття продукту.
- Зв'язки:
 - Асоціація з класом Category: продукти, що належать до певної категорії, можуть бути додані до корзини продуктів.

6. Клас Comments

- Атрибути:
 - review: string — текст відгуку.
 - name: string — ім'я автора відгуку.

- email: string — електронна пошта автора відгуку.
- rating: int — рейтинг, присвоєний продукту.
- Методи:
 - sendReview() — метод для відправлення відгуку.
- Зв'язки:
 - Асоціація з класом Category: продукти, що належать до певної категорії, можуть бути додані до корзини продуктів.

ProductCart включає об'єкти Product. Це означає, що кожен товар в корзині товарів може існувати окремо від неї, але об'єднаний в межах корзини для полегшення управління.

Цей зв'язок дозволяє відображати деталі продукту в контексті корзини, що допомагає користувачам швидко орієнтуватися в доданих товарах.

Пояснення зв'язків.

- Асоціація: Зв'язок між класами, де один клас використовує методи або атрибути іншого класу. Наприклад, зв'язок між Category та ProductCart показує, що категорії мають продукти, які можуть бути додані до корзини.
- Агрегація: Це часткова залежність між класами, де один клас складається з іншого класу, але об'єкти можуть існувати незалежно. Наприклад, Shop агрегує Product і Cart, що означає, що продукти та корзина можуть існувати незалежно від магазину.

Дана діаграма класів ілюструє структуру об'єктно-орієнтованого онлайн-магазину. Вона показує, як різні компоненти магазину взаємодіють один з одним через зв'язки асоціації та агрегації. Ця система дозволяє ефективно керувати категоріями товарів, корзиною покупок, відгуками та самими продуктами.

Композиція між Shop та Cart: Кошик (Cart) є частиною магазину (Shop) і не може існувати окремо.

- **Агрегація між Product та Comments:** Продукти (Product) можуть мати коментарі (Comments), але останні можуть існувати і окремо.

Діаграма класів для веб-застосунку "dwnclimat" демонструє, як різні компоненти взаємодіють один з одним і які атрибути та методи кожен з них має. Це допомагає зрозуміти структуру системи та полегшує її розробку та підтримку.

2.2. Проектування та реалізація алгоритмів роботи системи

Проектування та реалізація алгоритмів роботи системи є важливим етапом створення веб-застосунку для покупки та доставки кондиціонерів по Україні. У цьому розділі ми розглянемо основні алгоритми, які забезпечують функціональність нашого веб-застосунку, та обговоримо їх реалізацію.

Опис основних алгоритмів

Веб-застосунок "dwnclimat" включає декілька основних функціональних модулів, для кожного з яких розроблені відповідні алгоритми. Основні модулі включають:

- Алгоритми обробки запитів користувачів.
- Алгоритми керування товарами.
- Алгоритми обробки кошика покупок.
- Алгоритми обробки замовлень.

Алгоритми обробки запитів користувачів

Користувачі взаємодіють із веб-застосунком через запити, які можуть бути спрямовані на пошук товарів, перегляд деталей товару, додавання товару до кошика, обробку замовлення тощо.

Ці алгоритми відповідають за додавання, редагування, видалення та перегляд товарів у магазині. Основні етапи алгоритму керування товарами включають:

1. **Перевірка прав доступу:** Перевіряється, чи має користувач права на виконання операції (наприклад, додавання або редагування товару).

2. **Валідація даних товару:** Перевіряється коректність введених даних товару, таких як назва, ціна, опис тощо.

3. **Виконання операції:** Залежно від запиту, виконується додавання нового товару, редагування існуючого або видалення товару.

4. **Оновлення бази даних:** Зміни зберігаються у базі даних.

Ці алгоритми дозволяють користувачам додавати товари до кошика, змінювати кількість товарів та переглядати вміст кошика. Основні етапи алгоритму обробки кошика включають:

1. **Перевірка наявності товару:** Перевіряється, чи є товар у наявності.

2. **Додавання товару до кошика:** Якщо товар є в наявності, він додається до кошика користувача.

3. **Оновлення кількості товарів:** Користувач може змінювати кількість товарів у кошику.

4. **Перегляд вмісту кошика:** Користувач може переглядати товари, які знаходяться у кошику, і загальну вартість замовлення.

Алгоритми обробки замовлень

Ці алгоритми відповідають за оформлення та обробку замовлень користувачів, включаючи підтвердження замовлення, обробку оплати та організацію доставки. Основні етапи алгоритму обробки замовлень включають:

Перевірка вмісту кошика: Перевіряється, чи заповнений кошик і чи вказані всі необхідні дані для оформлення замовлення.

Підтвердження замовлення: Користувач підтверджує замовлення та надає необхідну інформацію для доставки.

2.3. Реалізація функціоналу системи веб-застосунку dnmwclimat

```
add_filter('woocommerce_single_product_image_thumbnail_html', 'remove_link_on_product_thumbnails', 10, 2);

function remove_link_on_product_thumbnails($html, $thumbnail_id) {
    $image_size = apply_filters('woocommerce_gallery_image_size', 'woocommerce_thumbnail');
    $image = wp_get_attachment_image($thumbnail_id, $image_size, false, array(
        'title'           => get_post_field('post_title', $thumbnail_id),
        'data-caption'     => get_post_field('post_excerpt', $thumbnail_id),
        'data-src'         => wp_get_attachment_image_url($thumbnail_id, 'full'),
        'data-large_image' => wp_get_attachment_image_url($thumbnail_id, 'full'),
        'data-large_image_width' => wp_get_attachment_image_src($thumbnail_id, 'full')[1],
        'data-large_image_height' => wp_get_attachment_image_src($thumbnail_id, 'full')[2],
        'class'           => 'swiper-slide-image',
    ));
    // Вертаємо лише зображення без посилання
    return $image;
}
```

Рис. 2.3 метод для зміни відображення мініатюр зображень товарів на сторінці окремого товару

Фільтр `woocommerce_single_product_image_thumbnail_html` дозволяє змінювати HTML-код, який використовується для відображення мініатюр зображень товарів на сторінці окремого товару в WooCommerce.

Функція `remove_link_on_product_thumbnails`

Ця функція використовується для заміни стандартного HTML-коду мініатюр WooCommerce. У стандартній реалізації мініатюри зображень можуть бути обгорнуті в посилання, які ведуть на великі версії зображень або на галерею. Функція `remove_link_on_product_thumbnails` видаляє ці посилання, залишаючи тільки самі зображення.

Дозволяє контролювати відображення мініатюр зображень товарів на сторінці одного товару в WooCommerce, видаляючи посилання і залишаючи тільки самі зображення. Це може бути корисно для покращення дизайну та зручності використання вашого інтернет-магазину.


```

add_action( 'wp_ajax_nopriv_add_to_cart_custom', 'add_to_cart_custom' );
add_action( 'wp_ajax_add_to_cart_custom', 'add_to_cart_custom' );

function add_to_cart_custom() {
    $product_id = apply_filters('woocommerce_add_to_cart_product_id', absint($_POST['product_id']));
    $quantity = empty($_POST['quantity']) ? 1 : wc_stock_amount($_POST['quantity']);
    $variation_id = absint($_POST['variation_id']);
    $passed_validation = apply_filters('woocommerce_add_to_cart_validation', true, $product_id, $quantity, $variation_id);

    if ($passed_validation && WC()->cart->add_to_cart($product_id, $quantity, $variation_id)) {
        do_action('woocommerce_ajax_added_to_cart', $product_id);
        echo json_encode(['success' => true]);
    } else {
        echo json_encode(['success' => false]);
    }
}

wp_die();
}

```

Рис. 2.4 Метод додавання товарів у кошик

Цей код написаний на PHP і призначений для додавання товарів у кошик у WordPress через AJAX, з використанням WooCommerce.

- 'wp_ajax_nopriv_add_to_cart_custom' — обробляє AJAX-запити від неавторизованих користувачів.
- 'wp_ajax_add_to_cart_custom' — обробляє AJAX-запити від авторизованих користувачів.

Обидва обробники вказують на одну функцію `add_to_cart_custom`.

`$_POST['product_id']` — отримує ідентифікатор товару з POST-запиту, застосовуючи фільтр `woocommerce_add_to_cart_product_id`.

`$_POST['quantity']` — отримує кількість товару. Якщо значення відсутнє або порожнє, використовується значення 1. Функція `wc_stock_amount` приводить значення до відповідного формату.

`$_POST['variation_id']` — отримує ідентифікатор варіації товару.

- Використовує фільтр `woocommerce_add_to_cart_validation` для перевірки валідності даних (ідентифікатор товару, кількість, ідентифікатор варіації).

Якщо валідація пройшла успішно (`$passed_validation` повертає `true`) і товар успішно доданий у кошик (`WC()->cart->add_to_cart`), то виконується дія `woocommerce_ajax_added_to_cart` та повертається JSON відповідь `{'success': true}`.

```

Redux::set_section( $opt_name, array(
    'title'      => __( 'Перша секція', 'mdv' ),
    'desc'       => __( ' ', 'mdv' ),
    'id'         => 'home-first-screen-body',
    'subsection' => true,
    'customizer_width' => '400px',
    'fields'     => array (
        array (
            'id'           => 'mdv-first-screen-slider-block',
            'type'         => 'slides',
            'title'        => __( 'Налаштування слайдеру (комп'ютерна/планшетна версія)', 'mdv' ),
            'subtitle'     => __( ' ', 'mdv' ),
            'desc'         => __( 'Рекомендовані пропорції зображення 5:1. Рекомендовані розміри зображення: 1440px-2880px', 'mdv' ),
            'placeholder' => array(
                'title'      => __( 'Заголовок', 'mdv' ),
                'description' => __( 'Опис', 'mdv' ),
                'url'        => __( 'Посилання', 'mdv' ),
            ),
            'settings' => array(
                'editor'     => array(
                    'media_buttons' => true, // Включення кнопок медіа
                    'tinymce'    => array(
                        'wpautop' => true, // Включення автоматичного форматування
                        'teeny'   => false, // Відключення teeny-версії редактора
                    ),
                    'quicktags' => true, // Включення швидких тегів
                ),
            ),
        ),
    ),
),
),
);

```

Рис. 2.5 Redux options

```

        array (
            'id'           => 'mdv-first-screen-slider-block-mobile',
            'type'         => 'slides',
            'title'        => __( 'Налаштування слайдеру (мобільна версія)', 'mdv' ),
            'subtitle'     => __( ' ', 'mdv' ),
            'desc'         => __( 'Рекомендовані пропорції зображення 3:4. Рекомендовані розміри зображення: 350px-480px', 'mdv', 'mdv' ),
        ),
        'placeholder' => array(
            'title'      => __( 'Заголовок', 'mdv' ),
            'description' => __( 'Опис', 'mdv' ),
            'url'        => __( 'Посилання', 'mdv' ),
        ),
        'settings' => array(
            'editor'     => array(
                'media_buttons' => true, // Включення кнопок медіа
                'tinymce'    => array(
                    'wpautop' => true, // Включення автоматичного форматування
                    'teeny'   => false, // Відключення teeny-версії редактора
                ),
                'quicktags' => true, // Включення швидких тегів
            ),
        ),
    ),
),
);

        array (
            'id'           => 'mdv-conact-us-button',
            'type'         => 'text',
            'title'        => __( 'Кнопка', 'mdv' ),
            'subtitle'     => __( 'Текст для кнопки', 'mdv' ),
            'desc'         => __( 'Введіть текст для кнопки блоку "Зв'язатися з нами"', 'mdv' ),
            'default'      => '',
        ),
    ),
),
);

```

Рис. 2.6 Redux options продовження

Приклад коду (рис.2.5-2.6) використовує фреймворк Redux для створення розділів налаштувань у панелі адміністратора WordPress. Зокрема, код створює секції для налаштування різних слайдерів та кнопок. Давайте детально розглянемо кожну частину коду. Код складається з кількох викликів функції Redux::setSection, яка використовується для додавання секцій у панель налаштувань. Кожна секція містить різні поля для введення та налаштувань.

Цей код створює три секції налаштувань у панелі адміністратора WordPress:

1. Слайдер для комп'ютерної/планшетної версії.
2. Слайдер для мобільної версії.
3. Кнопка для блоку "Зв'яжіться з нами".

Ці налаштування дозволяють адміністратору легко налаштувати відображення слайдерів і кнопки на різних пристроях.

Висновки до другого розділу.

У другому розділі «Проектування веб-застосунку» була розроблена та описана концепція і архітектура веб-застосунку для продажу та доставки кондиціонерів. Розробка включала визначення варіантів використання системи, проектування об'єктно-орієнтованої структури та аналіз основних вимог до функціональності веб-додатку.

Проектування веб-застосунку для продажу та доставки кондиціонерів завершено з урахуванням сучасних технологічних вимог та стандартів. Розроблена архітектура і об'єктно-орієнтована структура забезпечують гнучкість, масштабованість і надійність системи. Сформовані вимоги до інтерфейсу користувача та безпеки даних створюють передумови для успішної реалізації проекту, що забезпечить високий рівень задоволеності користувачів і ефективне управління бізнес-процесами.

**РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ ВЕБ-ЗАСТОСУНКУ
DNWCLIMAT**

3.1 Порядок встановлення та налаштування параметрів платформи

Розробка та впровадження веб-застосунку "dwnclimat" для покупки та доставки кондиціонерів по Україні вимагає ретельного підходу до його встановлення та налаштування. Даний розділ описує процес інсталяції та конфігурації застосунку для забезпечення його стабільної роботи та легкого використання як з боку користувачів, так і адміністраторів. Для опису архітектури веб-застосунку було створено діаграму розгортання (рис. 3.1).

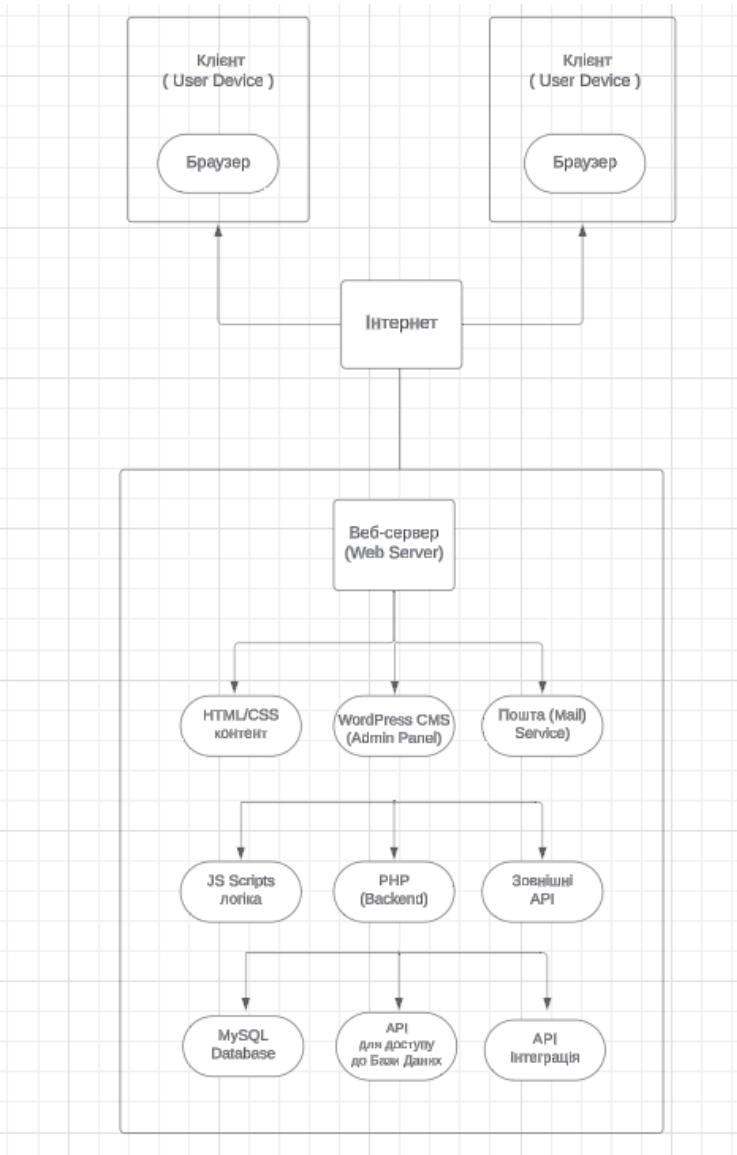


Рис. 3.1. Діаграма розгортання веб-застосунку dwnclimat

Опис процесу розгортання (рис 3.1)

Клієнт (User Device): Комп'ютери або мобільні пристрої користувачів, які використовують веб-браузер для доступу до веб-застосунку.

Веб-сервер (Web Server): Сервер, на якому розгорнуто веб-застосунок. Він відповідає за обробку запитів від користувачів, передачу веб-сторінок і взаємодію з базою даних.

База даних (Database Server): Сервер, що зберігає дані веб-застосунку, такі як інформація про користувачів, товари, замовлення тощо.

Адмін-панель (Admin Panel): Система керування контентом на WordPress, яка дозволяє адміністраторам додавати, редагувати та видаляти контент.

Служба пошти (Mail Service): Служба для відправки електронних повідомлень користувачам, наприклад, підтверджень замовлень, новин тощо.

Зовнішні API (External APIs): Сторонні служби, які використовуються для інтеграції додаткових функцій, таких як платіжні системи або карти для вибору адрес доставки.

Діаграма розгортання веб-застосунку "dwnclimat" показує, як різні компоненти взаємодіють один з одним і де вони розміщені. Це допомагає забезпечити належну організацію розгортання і підтримки веб-застосунку, зокрема забезпечення високої продуктивності, безпеки і надійності системи.

3.2 Структура інтерфейсу веб-застосунку

Структура інтерфейсу веб-застосунку є важливою складовою успіху користувацького досвіду (UX). Веб-застосунок "dwnclimat" призначений для полегшення процесу покупки та доставки кондиціонерів по Україні. Інтерфейс має бути інтуїтивно зрозумілим, функціональним і естетично привабливим, що забезпечить зручність користувачам у виконанні необхідних дій.

Основні компоненти інтерфейсу

Інтерфейс веб-застосунку складається з декількох основних компонентів:

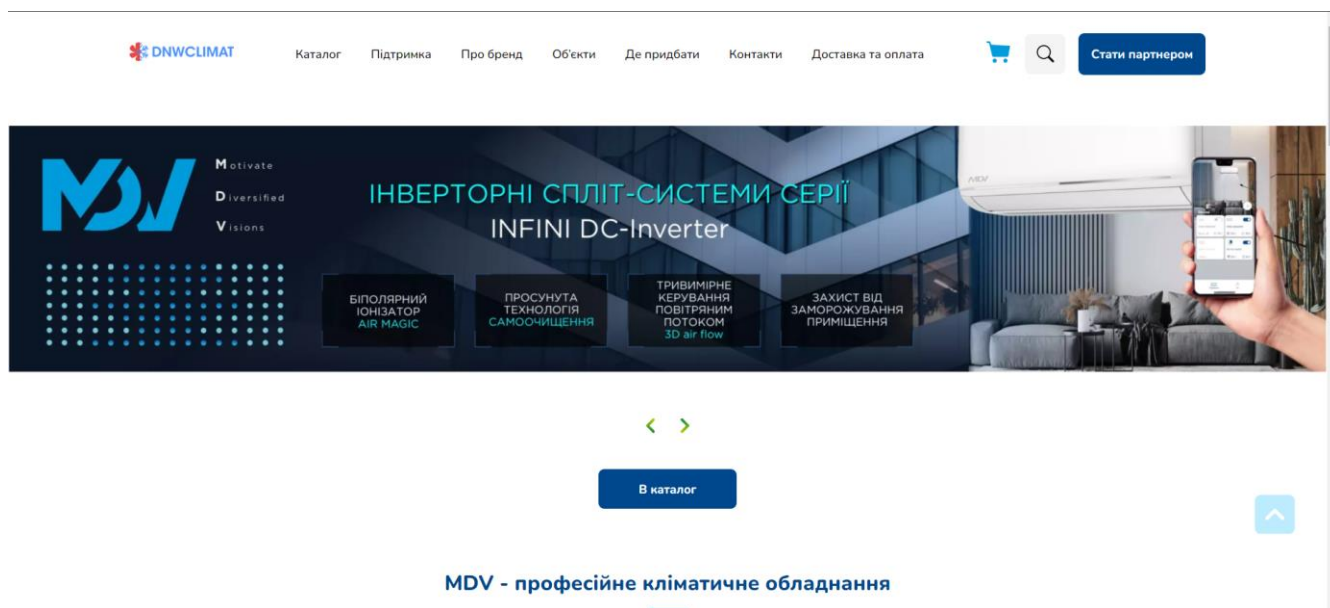


Рис. 3.2 Головна сторінка

Відразу на головній сторінці користувач бачить красивий інтерфейс, головне меню сайту та відразу може перейти в каталог товарів

Привітання користувачів та представлення основної інформації про сайт (рис.3.2)



Рис. 3.3 Інформація про компанію та каталог товарів

Також на головній сторінці є найпопулярніші товари з інформацією, де можна перейти відразу до покупки або обрати за параметрами які потрібні користувачу (рис. 3.4)

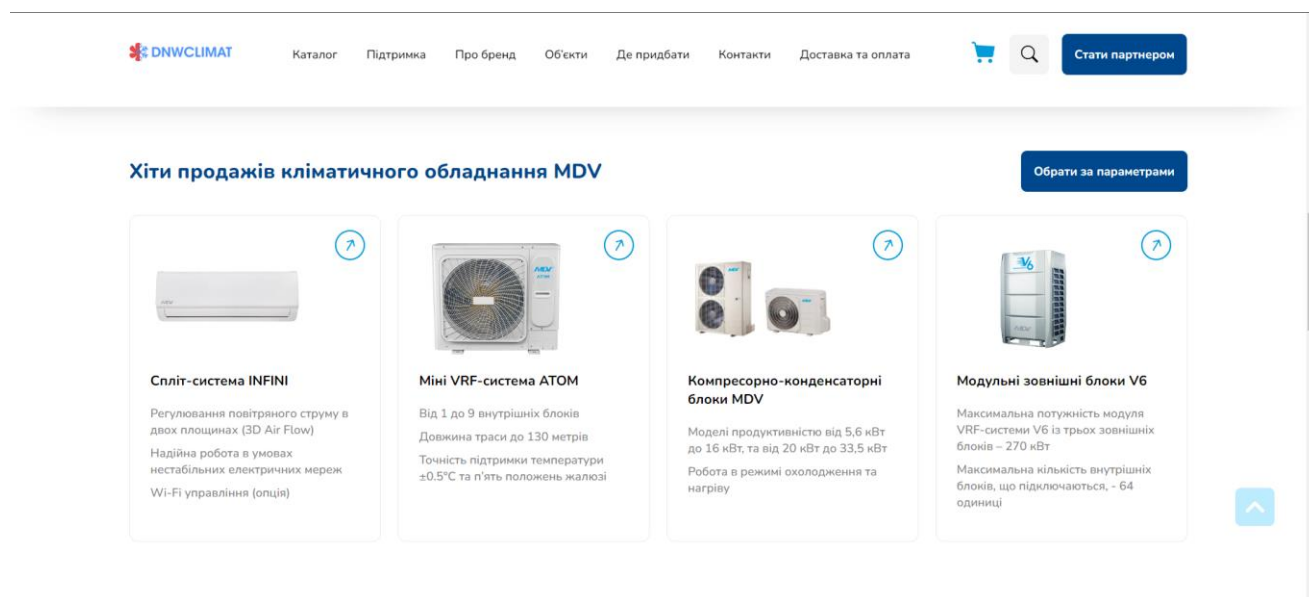


Рис. 3.4 Хіти продажів кліматичного обладнання

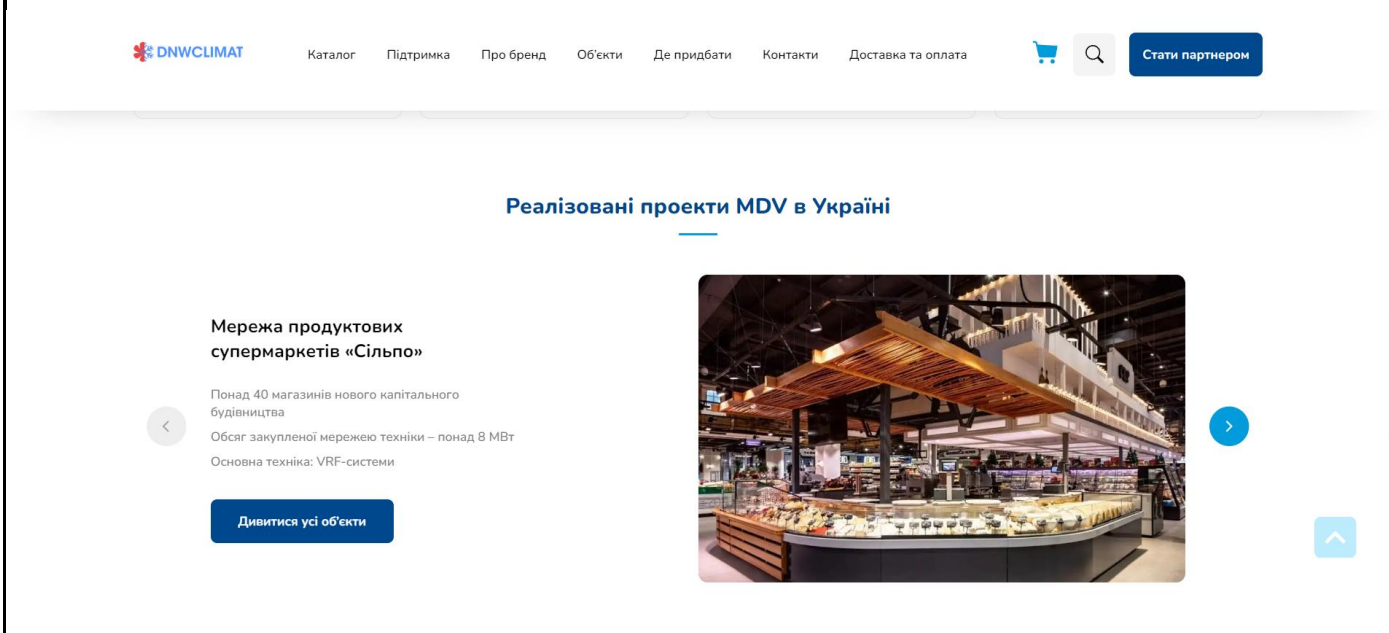


Рис. 3.5 Реалізовані проекти

Можна стати партнером якщо є бажання, потрібно всього заповнити форму в кінці сторінки партнерства. За багаторічний досвід на кліматичному ринку насамперед важливо вам, партнерам, щоб вони могли вигравати тендери, реалізовувати прибуткові об'єкти, успішно працювати з роздрібним покупцем. Та інформація послуг що надає компанія партнерам, це в основному Інженери та проєктувальники, корпоративним клієнтам та кліматичним компаніям

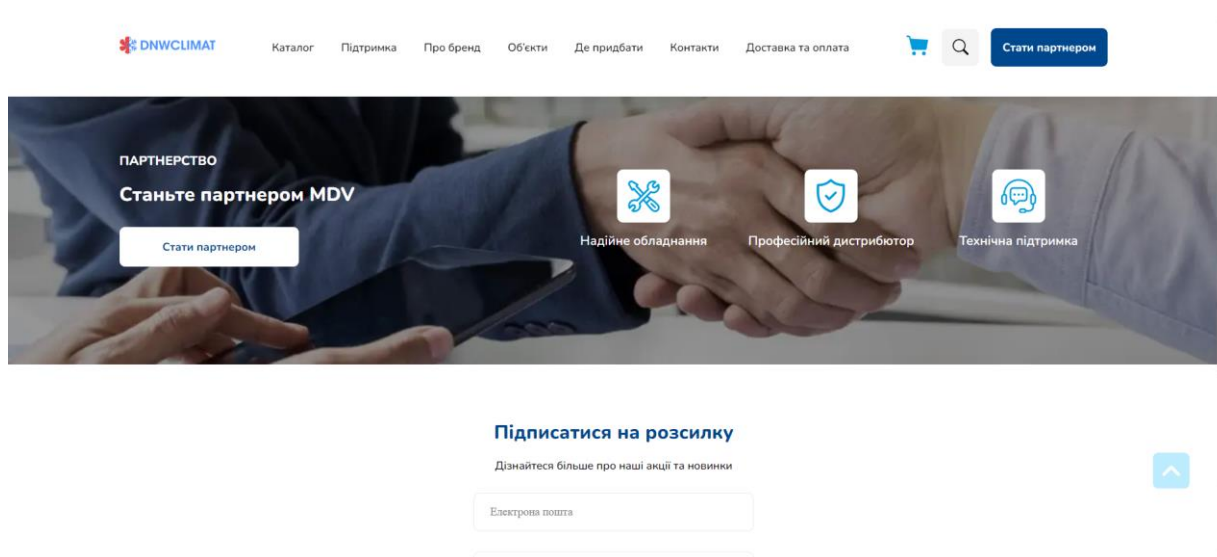


Рис. 3.6 Партнерство

Останнім елементом на головній сторінці можна підписатися на розсилку, де будуть приходити на пошту сповіщення та новини стосовно компанії, обладнання в обраному місті

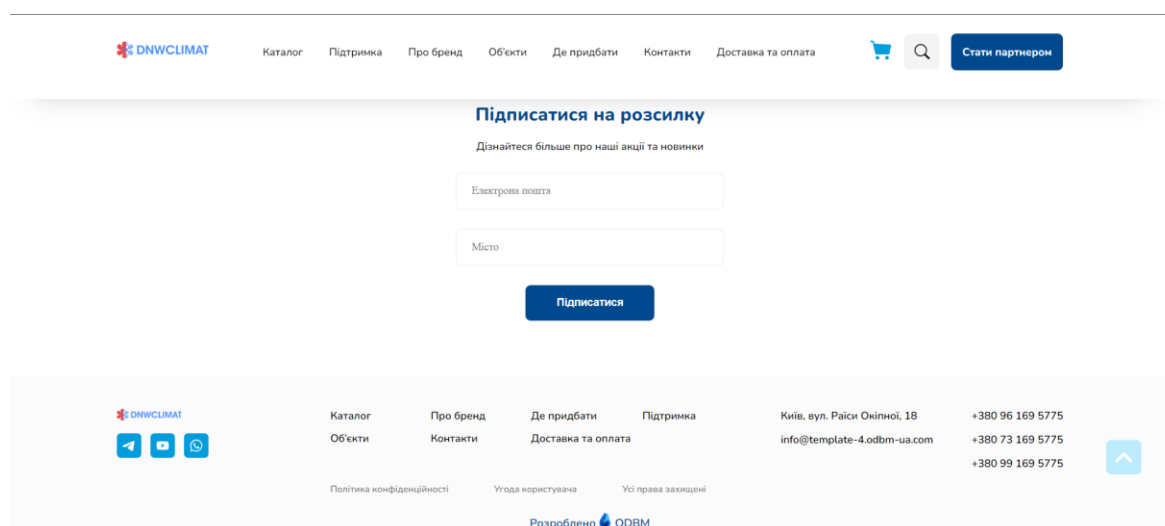


Рис. 3.7 Форма розсилки та футер

В футері можна побачити всю деталбну інформацію, контакти, адреса, емейл та меню з якого можна перейти на будь яку сторінку сайту, також політика конфіденціальності, угода користувача, та всі права захищені

Розглянемо процес створення замовлення одно з кондиціонерів. Для цього потрібно в головному меню перейти в каталог (рис 3.8-3.9)

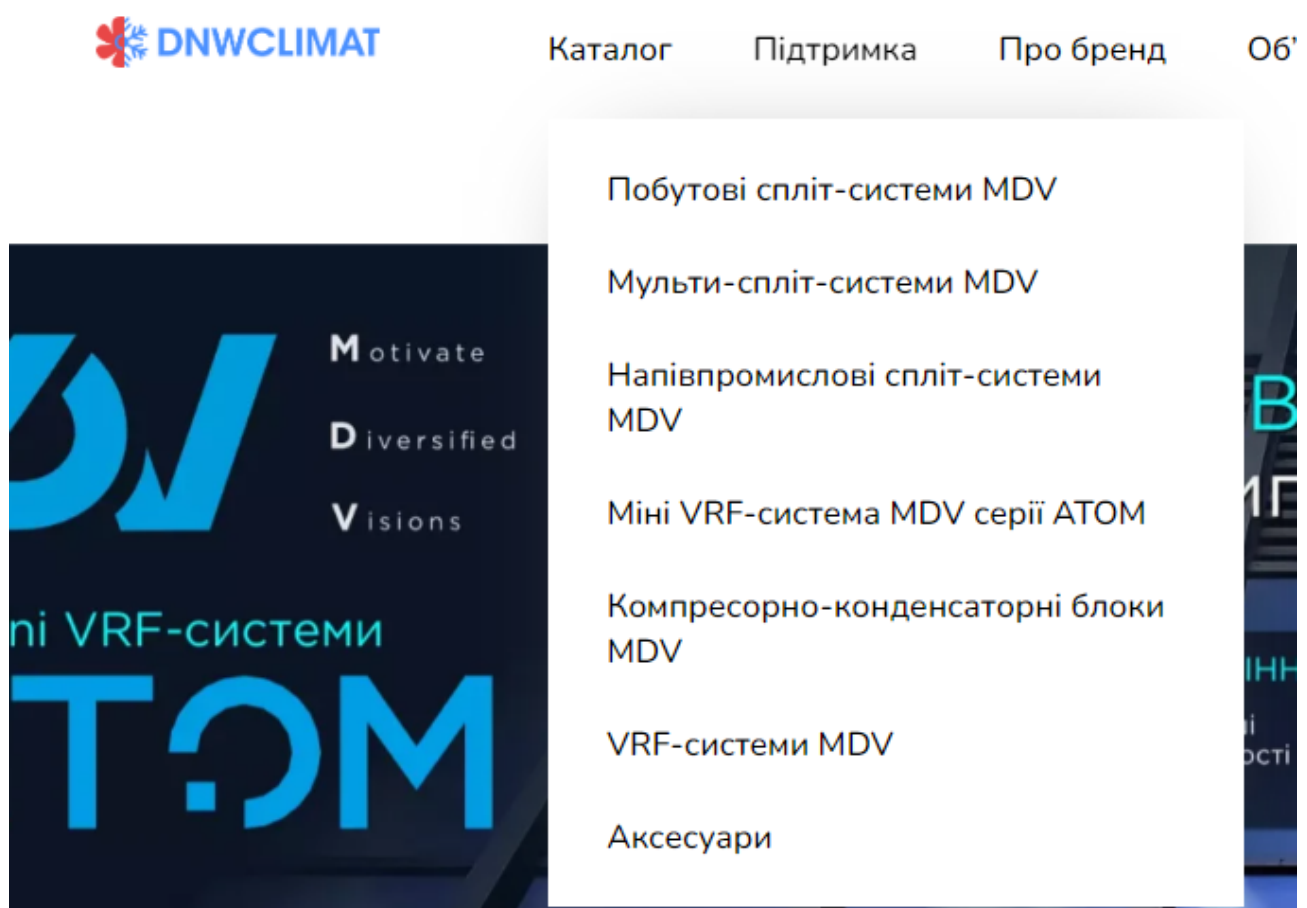


Рис. 3.8 Каталог

Після наведення на каталог обрати в бургер меню системи обладнання які нам потрібні, наприклад, обираємо Побутові спліт-системи MDV і бачимо декілька видів даних кондиціонерів, обираємо «Настінні внутршні блоки»

Побутові спліт-системи MDV

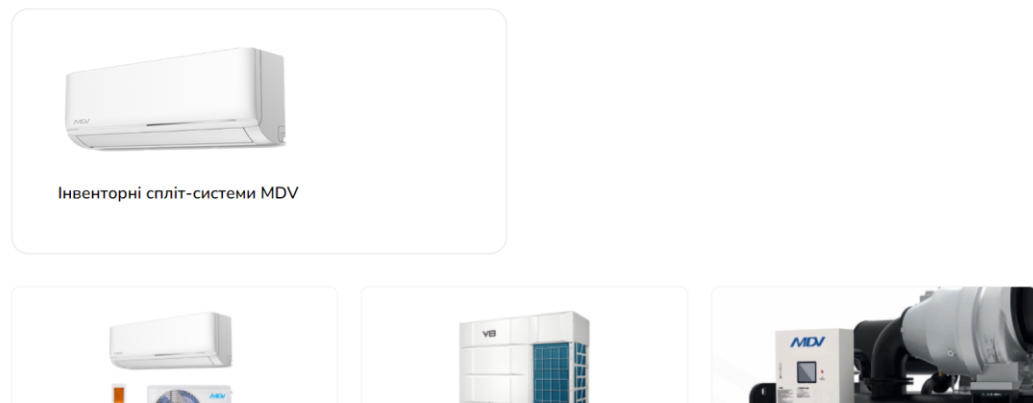


Рис. 3.10 Побутові спліт-системи

Після чого обираємо види (рис. 3.10) Інверторні спліти-системи MDV які є тільки одні на даний момент. Далі обираємо серію кондиціонера, візьмемо перший варіант і перекидає на сторінку товару (рис. 3.11) з повною інформацією про нього, документами та гарантією.

[Каталог](#)
[Підтримка](#)
[Про бренд](#)
[Об'єкти](#)
[Де придбати](#)
[Контакти](#)
[Доставка та оплата](#)

Стати партнером

Головна / Каталог обладнання MDV / Побутові спліт-системи MDV / Інверторні спліт-системи MDV / [INFINI UVpro Inverter](#)

INFINI UVpro Inverter

Заявка на розрахунок

Отримати прайс-лист

INFINI UVpro 3D DC-Inverter – це нова серія спліт-систем MDV із ультрафіолетовою лампою. Технологія UVpro (ультрафіолетова лампа) забезпечує ефективний захист від бактерій та вірусів. УФ-випромінювання знезаражує не тільки повітря, що проходить через внутрішній блок кондиціонера, але й теплообмінник. Повністю інверторна технологія (DC-інверторний компресор, DC-інверторні вентилятори зовнішнього та внутрішнього блоків) забезпечує високий рівень енергоефективності, надійність та низький рівень шуму. Серія представлена моделями продуктивністю 9-24 кВт. У спліт-системах серії INFINI UVpro поєднуються функції та опції, які забезпечують комфорт, зручність управління, турботу про здоров'я, надійну роботу системи, зручний монтаж та сервісне обслуговування.

A++

Клас енергоефективності A++

ERP

ERP 3D DC-Inverter

Бездротовий ПДК в комплекті

4

Гарантія 4 роки

Голосове керування (опція)

Рис. 3.11 Вибрана серія кондиціонера

Щоб замовити кондиціонер необхідно вибрати модель або натиснути детальніше і побачимо сторінку з обраним точно товаром де зможемо його додати в кошик та оформити замовлення (рис. 3.12)

Назва моделі	MDSAL-18HRFN8 / MDOAG-18HRFN8	MDSAL-24HRFN8 / MDOAG-24HRFN8
Модель внутрішнього блоку	MDSAL-18HRFN8	MDSAL-24HRFN8
Модель зовнішнього блоку	MDOAG-18HRFN8	MDOAG-24HRFN8
Номинальна теплопродуктивність, кВт	5,57 (3,10 - 5,85)	7,03 (2,11 - 8,21)
Номинальна холодопродуктивність, кВт	5,28 (3,39 - 5,90)	7,33 (1,55 - 8,21)
Електроживлення, В/Гц/Ф	220-240/50/1	220-240/50/1
	Детальніше	Детальніше

Рис. 3.12 Усі моделі вибраного кондеціонера

Після того як перейшли на вибрану модель бачимо ціну, фото, та детальний опис про обраний нами товар. Також нижче всі характеристики про обладнання.

На цій сторінці представлена більш повна інформація:

- Назва і бренд: Назва моделі і виробник кондиціонера.
- Модель: Номер моделі кондиціонера.
- Опис: Повний опис товару, включаючи його основні характеристики та переваги.
- Ціна: Актуальна ціна товару.
- Наявність на складі: Інформація про наявність товару на складі.
- Рейтинг: Рейтинг товару на основі відгуків користувачів.
- Відгуки: Список відгуків користувачів про товар.

Кожна картка товару містить кнопку "Додати в кошик". При натисканні на цю кнопку обраний товар додається до кошика покупок користувача, що дозволяє йому згодом оформити замовлення.

Взаємодія з користувачем: Після додавання товару в кошик користувачеві відображається повідомлення про успішне додавання, а також з'являється можливість перейти до кошика для перегляду або оформлення замовлення.

Завдяки кнопкам можна отримати прайс-лист або заявку на розрахунок якщо заповнити форму після натискання. В нашому випадку треба додати в кошик товар для замовлення де в нас відразу буде можливість перейти до кошика щоб продовжити замовлення або продовжити покупки і додати ще кілька товарів.

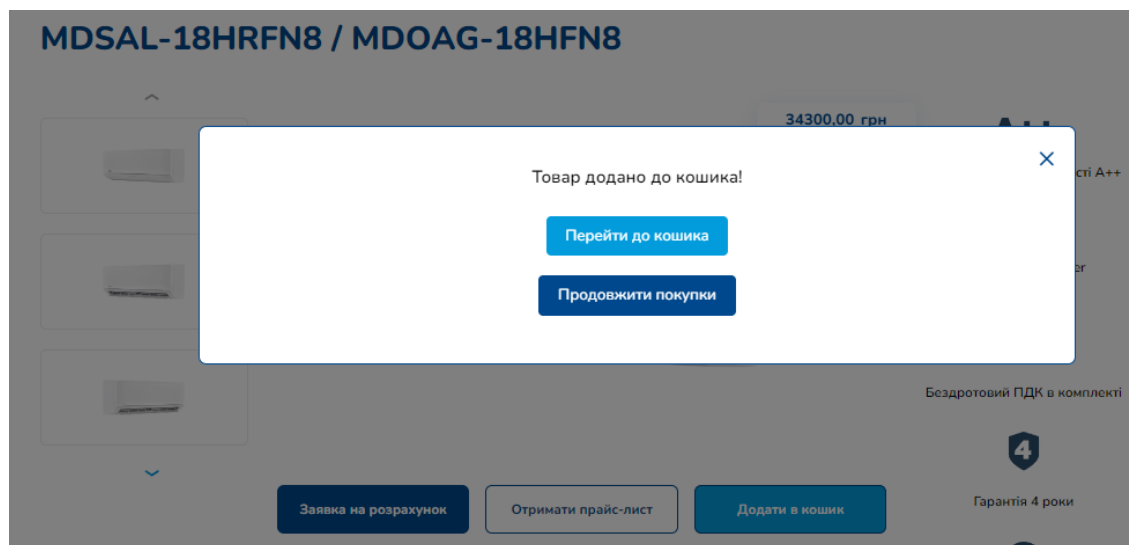


Рис. 3.13 Товар додано до кошика

На сторінці кошик товарів які ми обрали до замовлення, та переходим до оформлення замовлення. Тут можна видаляти товари які нам не потрібні або додали випадково ти обрати кількість.

Кошик товарів

		Товар	Ціна	Кількість	Проміжний підсумок
X		MDSAL-18HREFN8 / MDOAG-18HFN8	34300,00 грн	1	34300,00 грн
Оновити кошик					

Підсумки кошика

Проміжний підсумок	34300,00 грн
Загалом	34300,00 грн

Перейти до оформлення

Рис. 3.14 Кошик товарів

На сторінці оформлення товару ми бачимо інформацію про товар який вибрали, його модель, назва та ціна. Чітко перевіряємо та переходимо до заповнювання форми, треба записати всі свої платіжні дані обрати місто та спосіб доставки (рис. 3.15-3.16

Ваше замовлення

Товар	Проміжний підсумок
MDSAL-18HREFN8 / MDOAG-18HFN8 x 1	34300,00 грн
Проміжний підсумок	34300,00 грн
Загалом	34300,00 грн

Платіжні дані

Ім'я *	Прізвище *
Роман	Помаранець
Телефон *	Е-mail адреса *
+380680459785	romapomaranec@gmail.com
Область / Округ *	Місто / Село *
місто Київ	Київ
Спосіб доставки *	
Самовивіз	

Рис. 3.15 Оформлення замовлення

DNWCLIMAT Каталог Підтримка Про бренд Об'єкти Де придбати Контакти Доставка та оплата

місто Київ

Спосіб доставки.*
Виберіть спосіб доставки...

☒ Банківський переказ

Проведіть платіж безпосередньо на наш банківський рахунок. Будь ласка, вкажіть номер Вашого замовлення в описі переказу. Ваше замовлення не буде виконано, доки кошти не будуть зараховані на наш рахунок.

☐ Готівка при отриманні

Ваші персональні дані будуть використовуватися для обробки вашого замовлення, підтримки вашого досвіду на цьому сайті та для інших цілей, описаних у нашій політиці конфіденційності.

Підтвердити замовлення

DNWCLIMAT Каталог Про бренд Де придбати Підтримка Київ, вул. Раїси Окіпної, 18 +380 96 169 5775
Об'єкти Контакти Доставка та оплата info@template-4.odbm-ua.com +380 73 169 5775
+380 99 169 5775

Політика конфіденційності Угода користувача Усі права захищені

Розроблено ODBM

Рис. 3.16 Вибір платежу

Після заповнення всіх полів платіжних даних, та вибору оплати підтверджуємо замовлення і чекаємо поки зв'яжиться менеджер

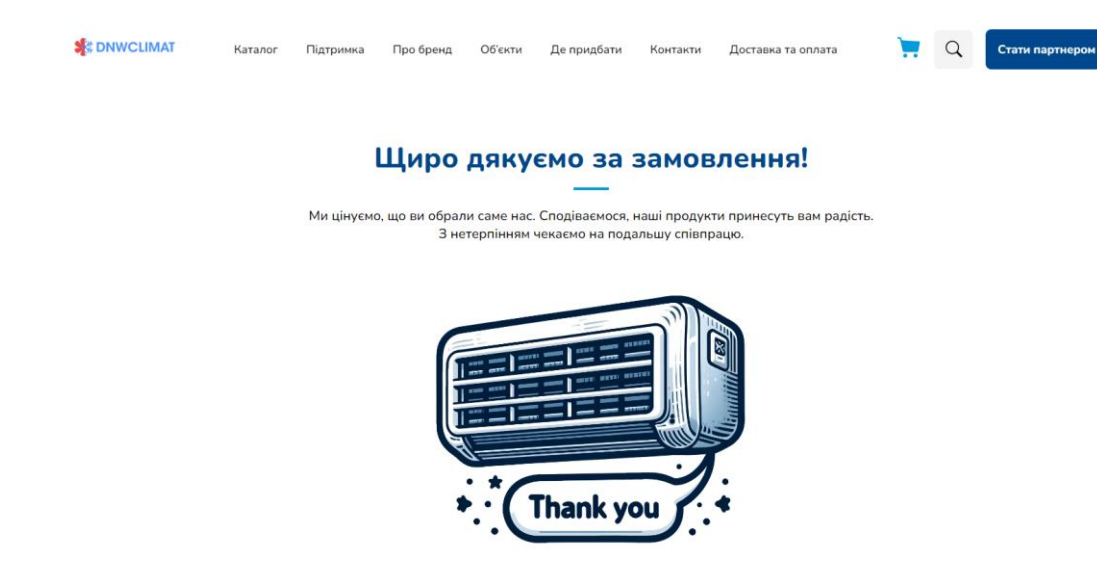


Рис.3.17 Замовлення виконано

3.2 Тестування веб-застосунку

За допомогою онлайн сервісу було проведено тестування веб-застосунку. Для тестування було використано GTMetrix та Pagespeed.

GTMetrix – це інструмент для тестування продуктивності та швидкості завантаження вебсторінок. Він надає звіти та аналіз продуктивності вашого вебсайту на основі різних метрик, таких як час завантаження сторінки, розмір сторінки, кількість запитів до сервера і багато іншого [25].

Після завершення тестування було отримано звіт, який містить різноманітну інформацію про продуктивність сайту (рис. 3.18).

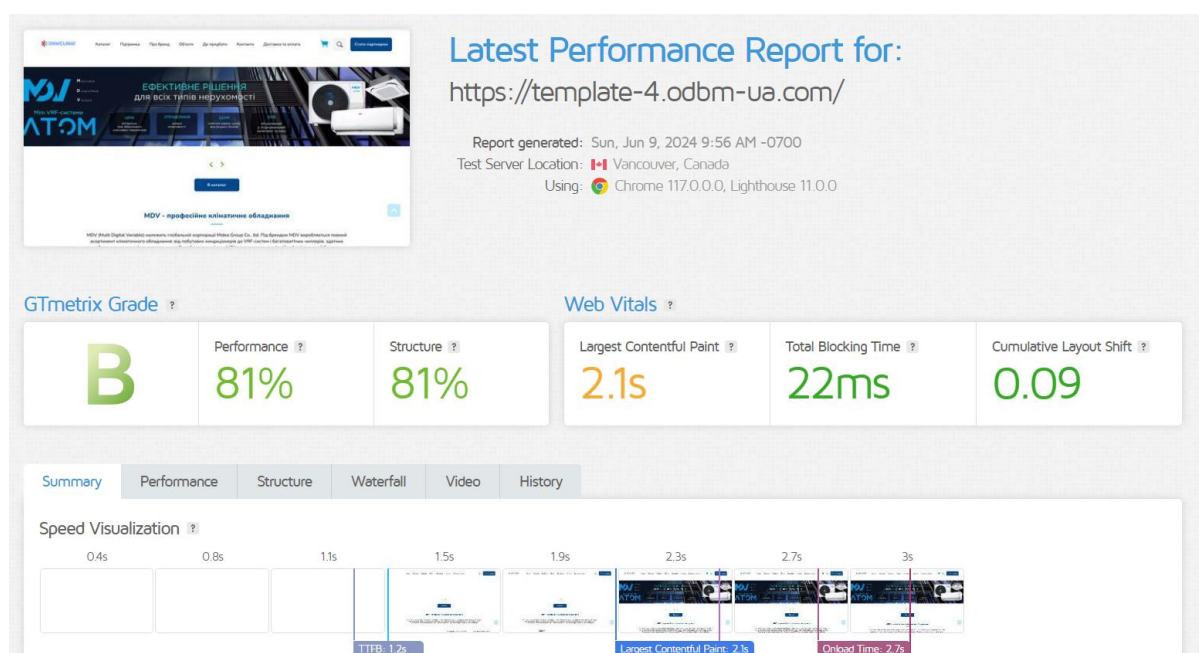


Рис. 3.18. Тестування вебдодатку за допомогою GTMetrix

Звіти GTMetrix містять такі дані, як загальний час завантаження сторінки, час завантаження вмісту, кількість запитів до сервера, розмір сторінки, швидкість завантаження на різних типах з'єднань і багато іншого. Ця інформація допомагає вам зрозуміти, як ваш вебсайт працює з точки зору продуктивності та швидкості, і дозволяє вам вжити заходів для поліпшення цих показників (рис.3.19).

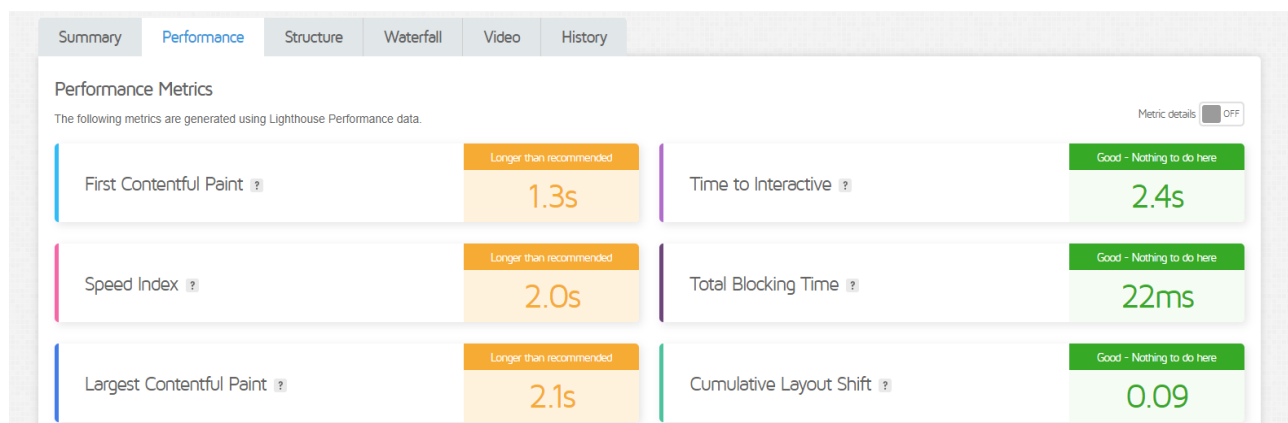


Рис. 3.19. Тестування швидкості вебдодатку за допомогою GTMetrix

GTMetrix також надає ряд інструментів та функцій для оптимізації продуктивності вебсайту, включаючи можливість зберігання та порівняння звітів, моніторинг продуктивності, сповіщення про проблеми продуктивності та багато іншого.

Pagespeed (рис. 3.20) вебсайт, який надає інструменти та рекомендації щодо оптимізації швидкості завантаження сторінок. Цей вебсайт створений командою Google з метою допомогти веброзробникам покращити продуктивність своїх вебсайтів.

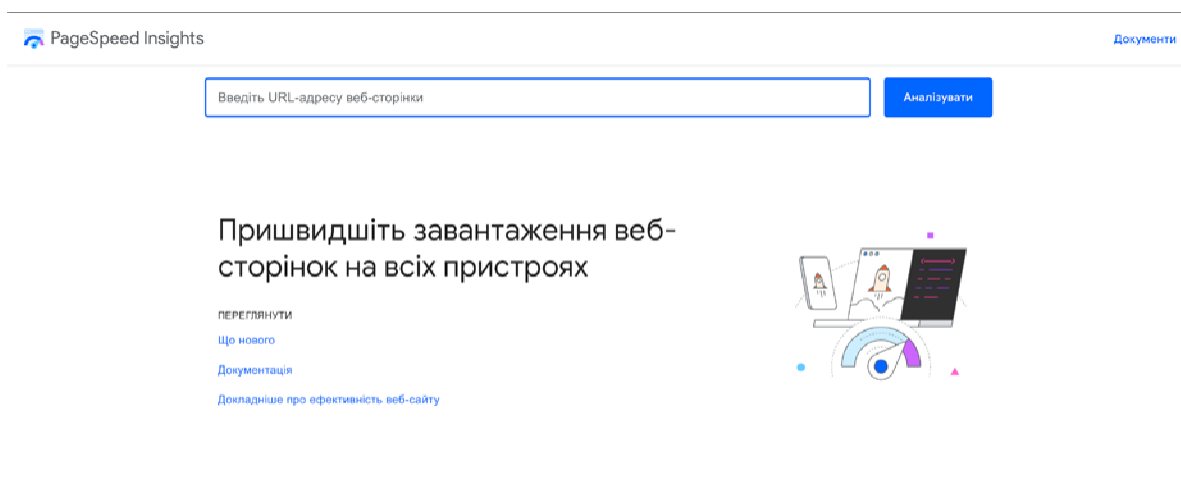


Рис. 3.20. Веб сайт Pagespeed

Аналіз швидкості сторінок. Вебсайт надає можливість ввести URL-адресу свого вебсайту і отримати докладний аналіз швидкості його завантаження. Інструмент оцінює різні аспекти, включаючи час завантаження, розмір сторінки, кількість запитів до сервера та інші параметри (рис. 3.20).

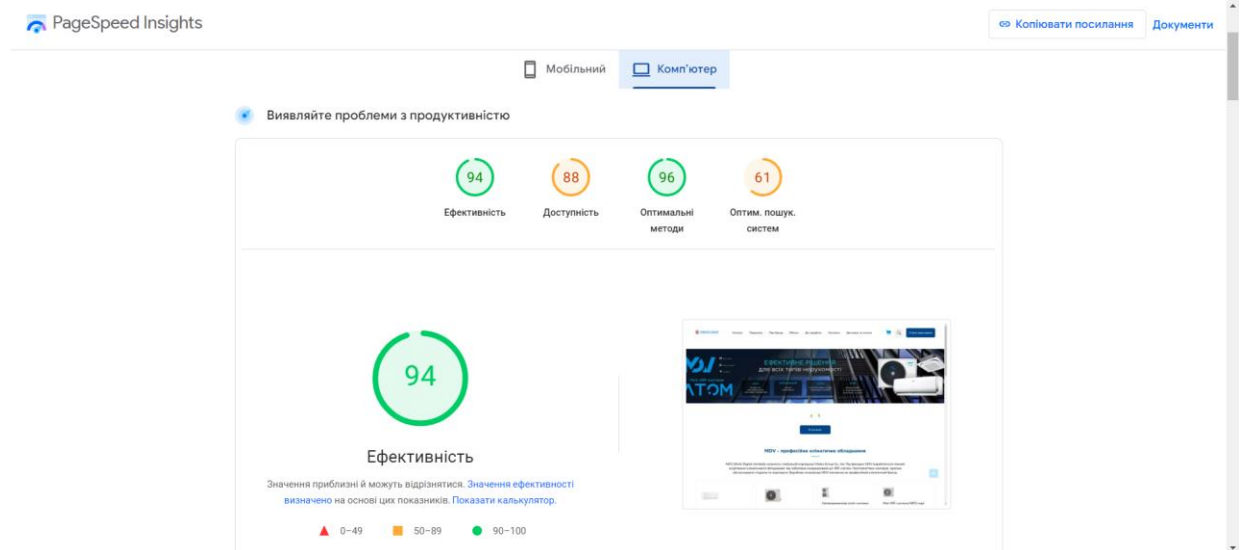


Рис. 3.20. Тестування вебдодатку за допомогою Pagespeed

Рекомендації щодо оптимізації. Вебсайт надає докладний список рекомендацій та практичні поради щодо того, як покращити швидкість завантаження сторінок. Ці рекомендації базуються на кращих практиках та рекомендаціях Google для оптимізації вебсайтів

Перевірка на мобільну готовність. Окрім аналізу швидкості сторінок, вебсайт також перевіряє готовність вебсайту до мобільних пристроїв. Він надає рекомендації щодо адаптивного дизайну, розміщення контенту та інших аспектів, які можуть впливати на використання вебсайту на мобільних пристроях (рис. 3.21).

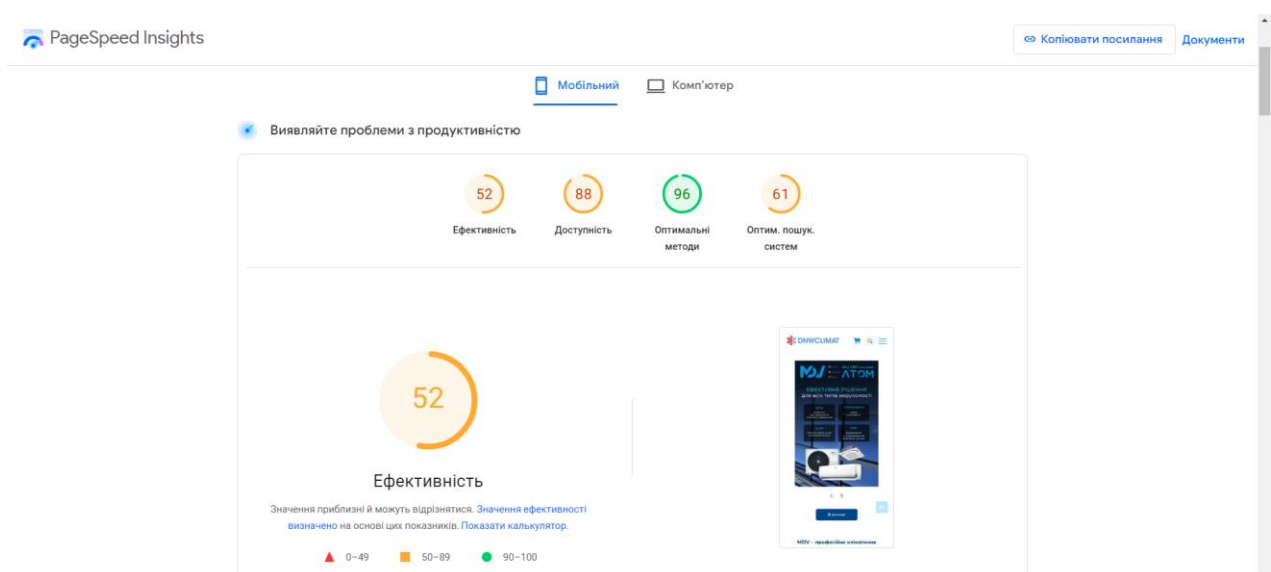


Рис. 3.21 Тестування веб додатку за допомогою Pagespeed

Перегляд різних метрик. Вебсайт надає можливість переглядати різні метрики швидкості, такі як Largest Contentful Paint (LCP), Cumulative Layout Shift (CLS), Total Blocking Time (TBT) та інші. Це допомагає зосередитися на конкретних показниках і виправити проблеми, що впливають на продуктивність (рис. 3.22).

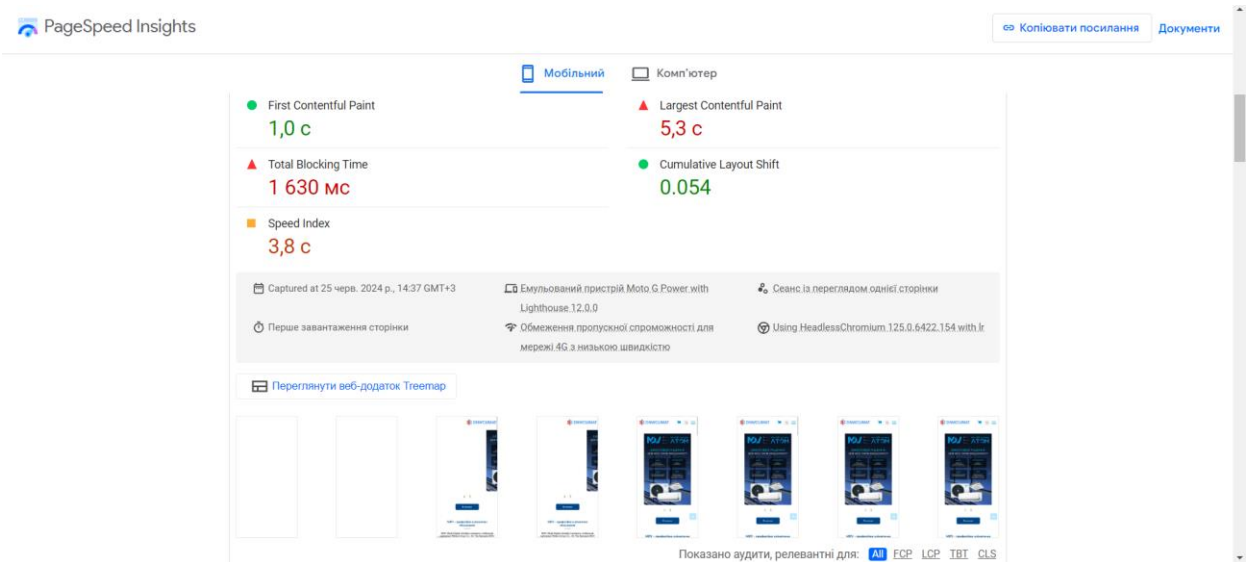


Рис. 3.30. Тестування вебдодатку за допомогою Pagespeed

За результатами тестування отримано метрики, такі як час завантаження сторінки, розмір сторінки, кількість запитів до сервера, Largest Contentful Paint (LCP), Cumulative Layout Shift (CLS), Total Blocking Time (TBT) та інші показники.

Висновки до третього розділу.

У третьому розділі кваліфікаційної роботи було докладно описано процес створення інтерфейсу та взаємодії користувачів з веб-застосунком для продажу та доставки кондиціонерів. Основні аспекти, що були висвітлені, включають структуру інтерфейсу, порядок встановлення, налаштування параметрів застосунку, а також основні функції, що забезпечують зручність та ефективність використання системи.

Було розроблено та впроваджено інтуїтивний і зручний інтерфейс, який полегшує користувачам взаємодію з системою. Було проведено всебічне тестування функціональності веб-застосунку.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи бакалавра було спроектовано та реалізовано веб-застосунок для компанії "dwnclimat", яка спеціалізується на продажу та встановленні кондиціонерів в Україні. Було проведено детальний аналіз предметної області та вивчено основні функціональні вимоги до системи на основі конкурентних рішень у цій галузі. Це дозволило визначити ключові характеристики та функціональність, яка необхідна для забезпечення ефективної роботи платформи. Під час розробки було створено архітектуру веб-застосунку, обрано підходи та технології, що найкраще підходять для реалізації проекту. Проектування системи включало аналіз варіантів використання програмного продукту та детальне моделювання взаємодії між основними компонентами. Було розроблено структуру бази даних, яка забезпечує ефективне зберігання та обробку даних про товари, користувачів і замовлення. Основні етапи проектування включали

Аналіз вимог: Визначення функціональних та нефункціональних вимог до системи на основі аналізу існуючих рішень та потреб користувачів.

- Моделювання архітектури: Створення архітектурної моделі веб-застосунку, що включає серверну частину на основі NestJS, клієнтську частину на основі NextJS та смарт-контракти на основі Solidity.
- Проектування бази даних: Розробка структури бази даних PostgreSQL для зберігання інформації про товари, замовлення, користувачів та інші необхідні дані.

Було також розроблено алгоритми роботи з додатком, включаючи опис послідовності дій для користувачів під час взаємодії з платформою. Описано основні функціональні можливості, такі як перегляд та порівняння товарів, додавання товарів до кошика, оформлення замовлення та взаємодія з адміністрацією сайту.

Детально описана схема розгортання веб-застосунку, що включає взаємодію основних компонентів системи, таких як фронтенд, бекенд, база даних та смарт-контракти. Описано інтерфейс користувача, який забезпечує зручну навігацію та доступ до всіх основних функцій платформи.

Архітектура веб-застосунку побудована з використанням сучасних технологій, що забезпечують високу продуктивність, масштабованість та безпеку. Впровадження смарт-контрактів забезпечує надійну та прозору роботу системи, дозволяючи автоматизувати процеси та забезпечити високий рівень довіри серед користувачів.

Інтуїтивний інтерфейс веб-застосунку дозволяє користувачам легко знаходити потрібні товари, порівнювати їх, додавати до кошика та оформляти замовлення. Адміністрація сайту має доступ до детальної статистики продажів, що дозволяє відстежувати активність користувачів та ефективність різних категорій товарів.

Розроблений веб-застосунок забезпечує компанію "dwnclimat" ефективним інструментом для продажу та підтримки клієнтів, що дозволяє збільшити обсяги продажів та підвищити задоволеність клієнтів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. **Apollo Client** [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.apollographql.com/docs/react/>
2. **CSS3** [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.w3schools.com/css/>
3. **ESLint** [Електронний ресурс] – Режим доступу до ресурсу:
<https://eslint.org/docs/latest/use/configure/>
4. **GraphQL** [Електронний ресурс] – Режим доступу до ресурсу:
<https://graphql.org/learn/>
5. **HTML5** [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.w3schools.com/html/>
6. **JavaScript** [Електронний ресурс] – Режим доступу до ресурсу:
<https://developer.mozilla.org/en-US/docs/Web/JavaScript>
7. **MongoDB** [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.mongodb.com/docs/manual/>
8. **React Router** [Електронний ресурс] – Режим доступу до ресурсу:
<https://reactrouter.com/en/main/start/tutorial/>
9. **React.js** [Електронний ресурс] – Режим доступу до ресурсу:
<https://react.dev/learn/>
10. **Redux** [Електронний ресурс] – Режим доступу до ресурсу:
<https://redux.js.org/introduction/getting-started/>
11. **REST API Design** [Електронний ресурс] – Режим доступу до ресурсу:
<https://restfulapi.net/rest-architectural-constraints/>
12. **SQLite** [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.sqlite.org/index.html/>
13. **TypeScript** [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.typescriptlang.org/docs/handbook/intro.html/>
14. **Webpack** [Електронний ресурс] – Режим доступу до ресурсу:
<https://webpack.js.org/guides/getting-started/>

15. E. McKinney, "Programming JavaScript Applications: Robust Web Architecture with Node, HTML5, and Modern JS Libraries," O'Reilly Media, Inc., Sebastopol, CA, USA, 2017. 450 p. ISBN: 978-1491950296.

16. B. Stevens, "Full-Stack Vue.js 2 and Laravel 5: Bring the Frontend and Backend Together with Vue, Vuex, and Laravel," Packt Publishing Ltd., Birmingham, UK, 2018, 400 p. ISBN: 978-1788839464

17. W. Griffiths, "Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages," O'Reilly Media, Inc., 2016. 750 p. ISBN: 978-0596159900.

18. M. Elliott, "Learning React: Functional Web Development with React and Redux," O'Reilly Media, Inc., 2018. ISBN: 978-1491954621.

19. D. Duckett, "HTML and CSS: Design and Build Websites," John Wiley & Sons, Indianapolis, IN, USA, 2014, 512 p. ISBN: 978-1118008188

ДОДАТКИ

					ІІЗ.КР.Б-121-24-ІІЗ	54

Функції теми (functions.php)

```
<?php
/**
 * dnwclimat functions and definitions
 *
 * @package mdv
 */

/**
 * Enqueue scripts and styles.
 */

add_action( 'wp_enqueue_scripts', function(){
    wp_enqueue_style( 'mdv-css', get_template_directory_uri() . '/assets/css/style.css');
    wp_enqueue_style( 'font-awesome-style', 'https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css');
    wp_enqueue_style( 'swiper-css', 'https://cdn.jsdelivr.net/npm/swiper@11/swiper-bundle.min.css');

    wp_enqueue_script( 'swiper', 'https://cdn.jsdelivr.net/npm/swiper@11/swiper-bundle.min.js');
    wp_enqueue_script( 'mdv-scripts', get_template_directory_uri() . '/assets/js/script.js', array(), true);
    wp_enqueue_script( 'mdv-swiper', get_template_directory_uri() . '/assets/js/swiper.js', array(), true);
    wp_enqueue_script( 'font-awesome', 'https://kit.fontawesome.com/1249968a46.js');
} );

// Додавання атрибуту defer до скриптів
add_filter( 'script_loader_tag', function( $tag, $handle, $src ) {
    // Перелік скриптів, до яких потрібно додати defer або async
    $scripts_to_defer = array( 'mdv-scripts', 'font-awesome' );

    if ( in_array( $handle, $scripts_to_defer ) ) {
        return "<script src='{$src}' defer></script>" . "\n";
    }

    return $tag;
}, 10, 3);

add_theme_support( 'post-thumbnails' );
add_theme_support( 'custom-logo' );
add_theme_support( 'menus' );

// init TGM plugins installer
require get_template_directory() . '/inc/tgm-list.php';

// Init Metaboxes Options.
require get_template_directory() . '/inc/metaboxes.php';

// Init Redux Theme Options Sections.
require get_template_directory() . '/inc/redux-options.php';

if ( class_exists( 'WooCommerce' ) ) {
    require_once( get_template_directory() . '/woo-c.php' );
}

function mytheme_header_cart() {
    $cart_count = WC()->cart->get_cart_contents_count();
    ?>
    <a class="cart-contents" href="<?php echo wc_get_cart_url(); ?>" title="<?php _e( 'View your shopping cart' ); ?>">
        
        <?php if ( $cart_count > 0 ): ?>
            <span class="cart-count-badge"><?php echo $cart_count; ?></span>
        <?php endif; ?>
    </a>
```

```

background-color: red;
color: white;
border: 2px solid white;
color: white;
font-size: 11px;
font-weight: bold;
display: flex;
justify-content: center;
align-items: center;
border-radius: 50%;
}
</style>
<?php
}
add_action( 'mini-cart', 'mytheme_header_cart' );

register_nav_menus(array(
    'header' => 'Верхнє меню',
    'footer' => 'Нижнє меню',
    'technical-library-menu' => 'Технічна бібліотека',    ));

function mdv_metaboxes($meta_boxes) {

    $meta_boxes = array();

    wp_reset_postdata();

    $prefix = "mdv_";

    $meta_boxes[] = array(
        'id'      => 'page_metabox',
        'title'   => 'Example Options',
        'pages'   => array( 'page', ), // Post type
        'context' => 'normal',
        'priority' => 'high',
        'show_names' => true, // Show field names on the left
        // 'show_on' => array( 'key' => 'page-template', 'value' => array('template-home.php'), ), // Specific post tem-
plates to display this metabox
        'fields' => array(
            array(
                'name' => __('About Photo','mdv'),
                'desc' => __('Upload a photo. Recommended size: 280-194px','mdv'),
                'id'   => $prefix . 'about_photo',
                'std'  => "",
                'type' => 'file',
            ),
            array(
                'name' => __('About title','mdv'),
                'desc' => __('The title','mdv'),
                'id'   => $prefix . 'about_title',
                'std'  => 'About Us',
                'type' => 'text',
            ),
            array(
                'name' => __('Description About Box','mdv'),
                'desc' => __('Type the description','mdv'),
                'id'   => $prefix . 'about_desc',
                'std'  => "",
                'type' => 'wysiwyg',
            ),
            array(

```

```

'name' => __('About Link','mdv'),
        'desc' => __('The Link','mdv'),
        'id' => $prefix . 'about_link',
        'std' => "",
        'type' => 'text',
    ),

    array(
        'name' => __('Info Title 1','mdv'),
        'desc' => __('Type here the Info Title 1','mdv'),
        'id' => $prefix . 'info_title_1',
        'std' => "",
        'type' => 'text',
    ),
    array(
        'name' => __('Info Description 1','mdv'),
        'desc' => __('Type here the Info Description 1','mdv'),
        'id' => $prefix . 'info_description_1',
        'std' => "",
        'type' => 'text',
    ),
    array(
        'name' => __('Info Title 2','mdv'),
        'desc' => __('Type here the Info Title 2','mdv'),
        'id' => $prefix . 'info_title_2',
        'std' => "",
        'type' => 'text',
    ),
    array(
        'name' => __('Info Description 2','mdv'),
        'desc' => __('Type here the Info Description 2','mdv'),
        'id' => $prefix . 'info_description_2',
        'std' => "",
        'type' => 'text',
    ),
    array(
        'name' => __('Info Title 3','mdv'),
        'desc' => __('Type here the Info Title 3','mdv'),
        'id' => $prefix . 'info_title_3',
        'std' => "",
        'type' => 'text',
    ),
    array(
        'name' => __('Info Description 3','mdv'),
        'desc' => __('Type here the Info Description 3','mdv'),
        'id' => $prefix . 'info_description_3',
        'std' => "",
        'type' => 'text',
    ),
    ),
);

return $meta_boxes;
}

function register_header_menu() {
    register_nav_menu('header-menu', __('Header Menu'));
}

add_action('after_setup_theme', 'register_header_menu');

//Woocommerce
require get_template_directory() . '/woocommerce/woocommerce.php';

```

```

add_filter( 'woocommerce_get_image_size_thumbnail', 'custom_woocommerce_thumbnail_size', 10, 1 );

function custom_woocommerce_thumbnail_size( $size ) {
    $size['width'] = 500;
    $size['height'] = 300;
    $size['crop'] = 0;
    return $size;
}

add_filter( 'woocommerce_single_product_image_thumbnail_html', 'remove_link_on_product_thumbnails', 10, 2 );

function remove_link_on_product_thumbnails( $html, $thumbnail_id ) {
    $image_size = apply_filters( 'woocommerce_gallery_image_size', 'woocommerce_thumbnail' );
    $image = wp_get_attachment_image( $thumbnail_id, $image_size, false, array(
        'title' => get_post_field( 'post_title', $thumbnail_id ),
        'data-caption' => get_post_field( 'post_excerpt', $thumbnail_id ),
        'data-src' => wp_get_attachment_image_url( $thumbnail_id, 'full' ),
        'data-large_image' => wp_get_attachment_image_url( $thumbnail_id, 'full' ),
        'data-large_image_width' => wp_get_attachment_image_src( $thumbnail_id, 'full' )[1],
        'data-large_image_height' => wp_get_attachment_image_src( $thumbnail_id, 'full' )[2],
        'class' => 'swiper-slide-image',
    ));
    // Вертаємо лише зображення без посилання
    return $image;
}

//Вигляд валюти на сайті
add_filter( 'woocommerce_currency_symbol', 'change_currency_symbol', 10, 2 );

function change_currency_symbol( $currency_symbol, $currency ) {
    switch( $currency ) {
        case 'UAH': $currency_symbol = 'грн'; break;
    }
    return $currency_symbol;
}

add_action( 'woocommerce_single_product_summary', 'woocommerce_template_single_add_to_cart', 30 );

// Видалення вибору кількості товару всюди окрім кошика
function remove_quantity_fields() {
    if ( ! is_cart() ) { // Если это не страница товара
        add_filter( 'woocommerce_is_sold_individually', '__return_true' );
    }
}
add_action( 'wp', 'remove_quantity_fields' );

add_action( 'wp_ajax_nopriv_add_to_cart_custom', 'add_to_cart_custom' );
add_action( 'wp_ajax_add_to_cart_custom', 'add_to_cart_custom' );

function add_to_cart_custom() {
    $product_id = apply_filters( 'woocommerce_add_to_cart_product_id', absint( $_POST['product_id'] ) );
    $quantity = empty( $_POST['quantity'] ) ? 1 : wc_stock_amount( $_POST['quantity'] );
    $variation_id = absint( $_POST['variation_id'] );
    $passed_validation = apply_filters( 'woocommerce_add_to_cart_validation', true, $product_id, $quantity, $variation_id );

    if ( $passed_validation && WC()->cart->add_to_cart( $product_id, $quantity, $variation_id ) ) {
        do_action( 'woocommerce_ajax_added_to_cart', $product_id );
        echo json_encode( [ 'success' => true ] );
    } else {
        echo json_encode( [ 'success' => false ] );
    }

    wp_die();
}

```

```
//
add_filter( 'woocommerce_order_formatted_billing_address', 'custom_woocommerce_order_formatted_billing_address', 10, 2 );

function custom_woocommerce_order_formatted_billing_address( $address, $order ) {
    $states = array(
        'UA05' => 'Вінницька область',
        'UA07' => 'Волинська область',
        'UA12' => 'Дніпропетровська область',
        'UA18' => 'Житомирська область',
        'UA21' => 'Закарпатська область',
        'UA23' => 'Запорізька область',
        'UA26' => 'Івано-Франківська область',
        'UA30' => 'місто Київ',
        'UA32' => 'Київська область',
        'UA35' => 'Кіровоградська область',
        'UA46' => 'Львівська область',
        'UA48' => 'Миколаївська область',
        'UA51' => 'Одеська область',
        'UA53' => 'Полтавська область',
        'UA56' => 'Рівненська область',
        'UA59' => 'Сумська область',
        'UA61' => 'Тернопільська область',
        'UA63' => 'Харківська область',
        'UA65' => 'Херсонська область',
        'UA68' => 'Хмельницька область',
        'UA71' => 'Черкаська область',
        'UA74' => 'Чернігівська область',
        'UA77' => 'Чернівецька область'
    );

    // Заміна коду області на назву, якщо вона існує в масиві
    if ( array_key_exists( $order->get_billing_state(), $states ) ) {
        $address['state'] = $states[ $order->get_billing_state() ];
    }

    return $address;
}

add_filter( 'woocommerce_checkout_fields', 'custom_add_checkout_fields' );
function custom_add_checkout_fields( $fields ) {
    $fields['billing']['new_field'] = array(
        'label' => __( 'Номер відділення', 'woocommerce' ),
        'placeholder' => _x( 'Введіть номер відділення', 'placeholder', 'woocommerce' ),
        'required' => true,
        'class' => array( 'form-row-wide' ),
        'clear' => true
    );
    return $fields;
}
add_action( 'woocommerce_checkout_update_order_meta', 'custom_checkout_field_update_order_meta' );

function custom_checkout_field_update_order_meta( $order_id ) {
    if ( !empty( $_POST['billing_delivery_method'] ) ) {
        update_post_meta( $order_id, 'billing_delivery_method', sanitize_text_field( $_POST['billing_delivery_method'] ) );
    }
    if ( !empty( $_POST['new_field'] ) ) {
        update_post_meta( $order_id, 'new_field', sanitize_text_field( $_POST['new_field'] ) );
    }

    function validate_phone_field() {
        if ( isset( $_POST['billing_phone'] ) && ! empty( $_POST['billing_phone'] ) ) {
            // Видаляємо всі символи, крім цифр, з номера телефону
            $phone_number = preg_replace( '/\D/', '', $_POST['billing_phone'] );
        }
    }
}
```

```

function custom_add_select2_field( $fields ) {
    $fields['billing']['billing_delivery_method'] = array(
        'type'      => 'select',
        'label'     => __( 'Спосіб доставки', 'woocommerce' ),
        'required'  => true,
        'class'     => array('form-row-wide'),
        'options'   => array(
            ''       => __( 'Виберіть спосіб доставки...', 'woocommerce' ),
            'nova_poshta' => __( 'Відділення Нової Пошти', 'woocommerce' ),
            'pickup'  => __( 'Самовивіз', 'woocommerce' ),
        ),
        'priority'  => 80,
    );
    return $fields;
}
add_filter( 'woocommerce_checkout_fields', 'custom_add_select2_field' );

add_action( 'woocommerce_checkout_update_order_meta', 'custom_checkout_field_update_order_meta' );

function custom_checkout_field_update_order_meta( $order_id ) {
    if ( !empty( $_POST['billing_delivery_method'] ) ) {
        update_post_meta( $order_id, 'billing_delivery_method', sanitize_text_field( $_POST['billing_delivery_method'] ) );
    }
    if ( !empty( $_POST['new_field'] ) ) {
        update_post_meta( $order_id, 'new_field', sanitize_text_field( $_POST['new_field'] ) );
    }
}

add_action( 'woocommerce_admin_order_data_after_shipping_address', 'custom_display_delivery_info_in_order_shipping_section',
10, 1 );

function custom_display_delivery_info_in_order_shipping_section( $order ) {
    $delivery_method_code = get_post_meta( $order->get_id(), 'billing_delivery_method', true );
    $delivery_methods_names = array(
        'nova_poshta' => __( 'Відділення Нової Пошти', 'woocommerce' ),
        'pickup'      => __( 'Самовивіз', 'woocommerce' ),
    );

    $delivery_method_name = isset( $delivery_methods_names[ $delivery_method_code ] ) ? $delivery_methods_names[ $delivery_method_code ] : __( 'Не визначено', 'woocommerce' );
    echo '<p><strong>'.__( 'Спосіб доставки' ).'</strong> '. $delivery_method_name . '</p>';

    $department_number = get_post_meta( $order->get_id(), 'new_field', true );
    echo '<p><strong>'.__( 'Номер відділення' ).'</strong> '. ( !empty( $department_number ) ? $department_number : __( 'Не вказано', 'woocommerce' ) ) . '</p>';
}

add_filter( 'woocommerce_checkout_fields', 'custom_order_checkout_fields' );

function custom_order_checkout_fields( $fields ) {
    // Встановлення пріоритетів для існуючих полів
    $fields['billing']['billing_first_name']['priority'] = 10;
    $fields['billing']['billing_last_name']['priority'] = 20;
    $fields['billing']['billing_phone']['priority'] = 30;
    $fields['billing']['billing_email']['priority'] = 40;
    $fields['billing']['billing_state']['priority'] = 50;
    $fields['billing']['billing_city']['priority'] = 60;
    $fields['billing']['billing_delivery_method']['priority'] = 70;
    $fields['billing']['new_field']['priority'] = 90;
    $fields['billing']['new_field']['required'] = false;
    return $fields;
}
add_filter( 'woocommerce_checkout_fields', 'custom_order_checkout_fields' );

add_filter( 'woocommerce_get_checkout_order_received_url', 'custom_order_received_url', 10, 2 );
function custom_order_received_url( $order_received_url, $order ) {

```

```

if ( is_a( $order, 'WC_Order' ) ) {
    // Додаємо ідентифікатор замовлення як параметр URL
    return home_url( 'thank-you/?order_id=' . $order->get_id() );
}
<h2><?php esc_html_e( 'Order Details', 'woocommerce' ); ?></h2>
<ul>
    <li><strong><?php echo __( 'Ім'я/фамілія:', 'woocommerce' ); ?></strong> <?php echo $order->get_billing_first_name() . ' '.
$order->get_billing_last_name(); ?></li>
    <li><strong><?php echo __( 'Номер телефону:', 'woocommerce' ); ?></strong> <?php echo $order->get_billing_phone(); ?></li>
    <li><strong><?php echo __( 'Email-адреса:', 'woocommerce' ); ?></strong> <?php echo $order->get_billing_email(); ?></li>

    <li><strong><?php echo __( 'Область:', 'woocommerce' ); ?></strong> <?php echo isset( $states[ $order->get_billing_state() ] ) ?
$states[ $order->get_billing_state() ] : __( 'Не визначено', 'woocommerce' ); ?></li>

    <li><strong><?php echo __( 'Місто/село:', 'woocommerce' ); ?></strong> <?php echo $order->get_billing_city(); ?></li>

<?php
// Відображення кастомних полів
$delivery_method = get_post_meta( $order->get_id(), 'billing_delivery_method', true );
$department_number = get_post_meta( $order->get_id(), 'new_field', true );

if ( !empty( $delivery_method ) ) {
    $delivery_method_name = isset( $delivery_methods[ $delivery_method ] ) ? $delivery_methods[ $delivery_method ] :
__( 'Невідомий метод', 'woocommerce' );
    echo '<li><strong>' . __( 'Спосіб доставки:', 'woocommerce' ) . '</strong> ' . esc_html( $delivery_method_name ) . '</li>';
}

if ( !empty( $department_number ) ) {
    echo '<li><strong>' . __( 'Номер відділення:', 'woocommerce' ) . '</strong> ' . esc_html( $department_number ) . '</li>';
}
?>
</ul>
<?php
}
add_action( 'woocommerce_email_order_details', 'custom_email_order_details', 10, 4 );

add_filter( 'woocommerce_checkout_fields', 'custom_remove_checkout_fields' );
function custom_remove_checkout_fields( $fields ) {
    unset( $fields['billing']['billing_company'] ); // Видаляє поле "Назва компанії"
    unset( $fields['billing']['billing_country'] ); // Видаляє поле "Країна / Region"
    unset( $fields['order']['order_comments'] ); // Видаляє поле "Нотатки до замовлення"
    unset( $fields['billing']['billing_address_2'] ); // Видаляє поле "Квартира, офіс, блок, тощо (опціонально)"
    unset( $fields['billing']['billing_address_1'] ); // Видаляє поле "Назва вулиці"
    unset( $fields['billing']['billing_postcode'] ); // Видаляє поле "Поштовий індекс"

    return $fields;
}

function validate_phone_field() {
    if ( isset( $_POST['billing_phone'] ) && ! empty( $_POST['billing_phone'] ) ) {
        // Видаляємо всі символи, крім цифр, з номера телефону
        $phone_number = preg_replace( '/\D/', '', $_POST['billing_phone'] );

        // Перевіряємо, чи номер починається з "+380" і містить загалом 12 цифр
        if ( ! preg_match( '/^380\d{9}$/ ', $phone_number ) ) {
            wc_add_notice( __( 'Будь ласка, введіть коректний номер телефону у форматі +380XXXXXXXXXX', 'woocommerce' ),
            'error' );
        }
    }
}
add_action( 'woocommerce_checkout_process', 'validate_phone_field' );

```

Результати пошуку (search.php)

```

<?php
/**
 * The template for displaying search results pages
 *
 * @link https://developer.wordpress.org/themes/basics/template-hierarchy/#search-result
 *
 * @package mdv
 */

get_header(); `
?>
<main>
<div class="search-page-body">
<div class="search-page-title">
<div class="container">
<h1 class="page-title">
<?php printf( esc_html__( 'Результат пошуку: %s', 'mdv' ), '<span class="similar-product-brand-orange-text">' . get_search_query() . '</span>' );?>
</h1>
</div>
</div>

<?php if ( have_posts() ) : ?>
<div class="container">
<?php while ( have_posts() ) : the_post(); ?>
<div class="search-page-body_products">
<a href="<?php the_permalink(); ?>" class="search-page-body_products__item-link">
<div class="search-page-body_products__item">
<div class="search-page_item__img">
<?php if ( get_the_post_thumbnail( get_the_ID() ) ) :
    echo get_the_post_thumbnail( get_the_ID() );
else: ?>
    
<?php endif; ?>
</div>

<div class="search-page_item__title">
<h2><?php the_title(); ?></h2>
<?php
// Виведення категорій поста
$categories = get_the_category();
if ( ! empty( $categories ) ) {
    echo '<div class="search-page_item__categories">';
    foreach ( $categories as $category ) {
        echo '<span>' . esc_html( $category->name ) . '</span> ';
    }
    echo '</div>';
    }
    ?>
</div>
</div>
</a>
</div>
<?php endwhile; ?>

</div>

```

```

        <?php else : ?>
        <section class="not-found">
        <div class="container">
        <h3><?php _e( 'За вашим запитом нічого не знайдено.', 'textdomain' ); ?></h3>
        </div>
        </section>
        <?php endif; ?>

    </div>
</main>
<?php
get_footer();

```

Додаток С

Результати сторінки підтримки (search.php)

```

<?php
/*
    Template Name: support-service
*/
get_header();
?>
<main class="support-service">
    <section class="support-service-title">
        <div class="container">
            <div class="breadcrumbs">
                <a href="<?php echo esc_url(home_url('/')); ?>">Головна</a> /
                <a href="<?php echo esc_url( home_url( '/support' ) ); ?>">Підтримка</a> /
                <span class="breadcrumbs-active-page"><?php the_title(); ?></span>
            </div>
            <div class="service-title_h1">
                <h1><?php echo $mdv_options['support-service-title']; ?></h1>
            </div>

            <div class="service-title_subtitle">
                <p><?php echo $mdv_options['support-service-subtitle']; ?></p>
            </div>
        </div>
    </section>

    <section class="support-service-list">
        <div class="container">
            <div class="service-list_title">
                <h2><?php echo $mdv_options['support-service-favours-title']; ?></h2>
            </div>
            <div class="service-list_body">
                <div class="service-list_body__block">
                    <div class="service-list_body__block-number">
                        <span>1</span>
                    </div>

                    <div class="service-list_body__block-title">
                        <p><?php echo $mdv_options['support-service-favours-title1']; ?></p>
                    </div>

                    <div class="service-list_body__block-subtitle">
                        <p><?php echo $mdv_options['support-service-favours-subtitle1']; ?></p>
                    </div>
                </div>
            </div>
        </div>
    </section>

```

```
<div class="servise-list_body__block">
  <div class="servise-list_body__block-number">
    <span>2</span>
  </div>

  <div class="servise-list_body__block-title">
    <p><?php echo $mdv_options['support-service-favours-title2']; ?></p>
  </div>

  <div class="servise-list_body__block-subtitle">
    <p><?php echo $mdv_options['support-service-favours-subtitle2']; ?></p>
  </div>
</div>

<div class="servise-list_body__block">
  <div class="servise-list_body__block-number">
    <span>3</span>
  </div>

  <div class="servise-list_body__block-title">
    <p><?php echo $mdv_options['support-service-favours-title3']; ?></p>
  </div>

  <div class="servise-list_body__block-subtitle">
    <p><?php echo $mdv_options['support-service-favours-subtitle3']; ?></p>
  </div>
</div>
</div>
</div>
</div>
</section>

<section class="servise-branded">
  <div class="container">
    <div class="servise-branded_title">
      <h2><?php echo $mdv_options['support-service-branded-title']; ?></h2>
    </div>

    <div class="servise-branded-body">
      <div class="servise-branded-body_block">
        <div class="servise-branded-body_block-img">
          
        </div>

        <div class="servise-branded-body_block-title">
          <p><?php echo $mdv_options['support-service-branded-title1']; ?></p>
        </div>

        <div class="servise-branded-body_block-subtitle">
          <p><?php echo $mdv_options['support-service-branded-subtitle1']; ?></p>
        </div>
      </div>

      <div class="servise-branded-body_block">
        <div class="servise-branded-body_block-img">
          
        </div>
      </div>
    </div>
  </div>
</section>
```

```
<div class="servise-branded-body_block-title">
  <p><?php echo $mdv_options['support-service-branded-title2']; ?></p>
</div>

<div class="servise-branded-body_block-subtitle">
  <p><?php echo $mdv_options['support-service-branded-subtitle2']; ?></p>
</div>
</div>

<div class="servise-branded-body_block">
  <div class="servise-branded-body_block-img">
    
  </div>

  <div class="servise-branded-body_block-title">
    <p><?php echo $mdv_options['support-service-branded-title3']; ?></p>
  </div>

  <div class="servise-branded-body_block-subtitle">
    <p><?php echo $mdv_options['support-service-branded-subtitle3']; ?></p>
  </div>
</div>
</div>
</div>
</section>

<section class="servise-guarantee">
  <div class="container">
    <div class="servise-guarantee-title">
      <h2><?php echo $mdv_options['support-service-garanty-title']; ?></h2>
    </div>

    <div class="servise-guarantee-body">
      <div class="servise-guarantee-body_block">
        <div class="servise-guarantee-body_block__title">
          <p><?php echo $mdv_options['support-service-garanty-title1']; ?></p>
        </div>

        <div class="servise-guarantee-body_block__arrow">
          <svg xmlns="http://www.w3.org/2000/svg" width="15" height="21" viewBox="0 0 15 21" fill="none">
            <path d="M6.79289 20.7071C7.18342 21.0976 7.81658 21.0976 8.20711 20.7071L14.5711 14.3431C14.9616
13.9526 14.9616 13.3195 14.5711 12.9289C14.1805 12.5384 13.5474 12.5384 13.1569 12.9289L7.5 18.5858L1.84315
12.9289C1.45262 12.5384 0.819456 12.5384 0.428932 12.9289C0.0384072 13.3195 0.0384072 13.9526 0.428932
14.3431L6.79289 20.7071ZM6.5 -4.37119e-08L6.5 20L8.5 20L8.5 4.37119e-08L6.5 -4.37119e-08Z" fill="#029CDC"/>
          </svg>
        </div>

        <div class="servise-guarantee-body_block__subtitle">
          <p><?php echo $mdv_options['support-service-garanty-subtitle1']; ?></p>
        </div>
      </div>

      <div class="servise-guarantee-body_block">
        <div class="servise-guarantee-body_block__title">
          <p><?php echo $mdv_options['support-service-garanty-title2']; ?></p>
        </div>

        <div class="servise-guarantee-body_block__arrow">
```



```

<a href="<?php echo esc_url(home_url('/support')); ?>">Підтримка</a> /
<a href="<?php echo esc_url(home_url('/support-technical')); ?>">Технічна підтримка</a> /
<span class="breadcrumbs-active-page">
    <?php the_title(); ?>
</span>
</div>

```

Додаток D

Результати сторінки бібліотеки(technical-library.php)

```

<div class="technical-library-title">
    <h1>
        <?php echo $mdv_options['technical-library-title']; ?>
    </h1>
</div>

<div class="technical-library-subtitle">
    <p>
        <?php echo $mdv_options['technical-library-subtitle']; ?>
    </p>
</div>

<div class="technical-library-body">
    <div class="technical-library_list">
        <?php
            $GLOBALS['slugCategory'] = array();

            function print_product_categories($parent_id = 0, $taxonomy = 'product_cat', $level = 0)
            {
                $args = array(
                    'taxonomy' => $taxonomy,
                    'parent' => $parent_id,
                    'hide_empty' => true,
                );
                $categories = get_terms($args);

                if ($categories) {
                    echo $level == 0 ? '<ul id="menu-technical-library-menu" class="library-menu">' : '<ul class="sub-menu">';

                    foreach ($categories as $category) {
                        $child_categories = get_terms(['taxonomy' => $taxonomy, 'parent' => $category->term_id, 'hide_empty' =>
false]);
                        $has_children_class = !empty($child_categories) ? ' menu-item-has-children' : '';

                        $term_link = get_term_link($category);
                        if (is_wp_error($term_link)) {
                            continue;
                        }
                        echo '<li class="menu-item' . $has_children_class . '"><a href="' . esc_url($term_link) . '" class="' .
esc_html($category->slug) . '">' . esc_html($category->name) . '</a>';

                        if (!empty($has_children_class)) {
                            print_product_categories($category->term_id, $taxonomy, $level + 1);
                        } else {

                            $GLOBALS['slugCategory'][] = $category->slug;
                        }
                    }
                }
            }

```

```

        echo '</li>';
    }
    echo '</ul>';
}

print_product_categories();
?>
</div>

```

```

<div class="series-documents-body infini-inverter">
    <?php $queried_object = get_term_by('slug', 'infini-inverter', 'product_cat');
    if ($queried_object) {
        $category_id = $queried_object->term_id; // ID категорії
        $slug = $queried_object->slug; // Слаг категорії
        $series_documents_body_block_1 = get_field('series_documents_body_block_1', 'product_cat_' . $category_id);
        $series_documents_body_block_2 = get_field('series_documents_body_block_2', 'product_cat_' . $category_id);
        $series_documents_body_block_3 = get_field('series_documents_body_block_3', 'product_cat_' . $category_id);
        $series_documents_body_block_4 = get_field('series_documents_body_block_4', 'product_cat_' . $category_id);
        $series_documents_body_block_5 = get_field('series_documents_body_block_5', 'product_cat_' . $category_id);
        $series_documents_body_block_6 = get_field('series_documents_body_block_6', 'product_cat_' . $category_id);

        //Перевіряємо чи є дані в одному із полів
        if ($series_documents_body_block_1 || $series_documents_body_block_2 ||
        $series_documents_body_block_3) {
            // Представленням блоки з інформацією
            if ($series_documents_body_block_1 &&
            !empty($series_documents_body_block_1['nazva_dokumentu_1'])) {
                // Представлення блока для series_documents_body_block_1
                echo '<div class="series-documents-body_block series-documents-' . esc_attr($slug) . '">';

                $nazva_dokumentu_1 = $series_documents_body_block_1['nazva_dokumentu_1'];
                $ico_preview_1 = $series_documents_body_block_1['ico_preview'];
                $ico_copy_1 = $series_documents_body_block_1['ico_copy'];
                $ico_download_1 = $series_documents_body_block_1['ico_download'];
                $download_file_1 = $series_documents_body_block_1['download_file_1'];
                ?>
                <div class="series-documents-body_block__info">
                    <p>
                        <?php echo esc_html($nazva_dokumentu_1); ?>
                    </p>
                </div>

                <div class="series-documents-body_block__icons">
                    <a href="<?php echo esc_html($download_file_1); ?>">
                        <?php echo '<img src="" . esc_url($ico_preview_1) . '" alt="Серія INFINI Inverter">'; ?>
                    </a>
                    <a href="#">
                        <?php echo '<img src="" . esc_url($ico_copy_1) . '" alt="Серія INFINI Inverter">'; ?>
                    </a>
                    <a href="#">
                        <?php echo '<img src="" . esc_url($ico_download_1) . '" alt="Серія INFINI Inverter">'; ?>
                    </a>
                </div>

                <?php
                echo '</div>';
            }
        }
    }
}

```

```

if ($series_documents_body_block_2 && !empty($series_documents_body_block_2['nazva_dokumentu_2'])) {
    // Представлення блока для series_documents_body_block_2

    echo '<div class="series-documents-body_block series-documents-' . esc_attr($slug) . '">';

    $nazva_dokumentu_2 = $series_documents_body_block_2['nazva_dokumentu_2'];
    $ico_preview_2 = $series_documents_body_block_2['ico_preview'];
    $ico_copy_2 = $series_documents_body_block_2['ico_copy'];
    $ico_download_2 = $series_documents_body_block_2['ico_download'];
    $download_file_2 = $series_documents_body_block_2['download_file_2'];
    ?>

    <div class="series-documents-body_block__info">
        <p>
            <?php echo esc_html($nazva_dokumentu_2); ?>
        </p>
    </div>

    <div class="series-documents-body_block__icons">
        <a href="#">
            <?php echo ''; ?>
        </a>
        <a href="#">
            <?php echo ''; ?>
        </a>
        <a href="#">
            <?php echo ''; ?>
        </a>
    </div>
    <?php
    echo '</div>';
}

if ($series_documents_body_block_3 && !empty($series_documents_body_block_3['nazva_dokumentu_3'])) {
    // Представлення блока для series_documents_body_block_3
    echo '<div class="series-documents-body_block series-documents-' . esc_attr($slug) . '">';
    $nazva_dokumentu_3 = $series_documents_body_block_3['nazva_dokumentu_3'];
    $ico_preview_3 = $series_documents_body_block_3['ico_preview'];
    $ico_copy_3 = $series_documents_body_block_3['ico_copy'];
    $ico_download_3 = $series_documents_body_block_3['ico_download'];
    $download_file_3 = $series_documents_body_block_3['download_file_3'];
    ?>

    <div class="series-documents-body_block__info">
        <p>
            <?php echo esc_html($nazva_dokumentu_3); ?>
        </p>
    </div>

    <div class="series-documents-body_block__icons">
        <a href="#">
            <?php echo ''; ?>
        </a>
        <a href="#">
            <?php echo ''; ?>
        </a>
    </div>
}

```

```

        <a href="#">
            <?php echo ''; ?>
        </a>
    </div>

    <?php
    echo '</div>';
    }
} if ($series_documents_body_block_4 && !empty($series_documents_body_block_4['nazva_dokumentu_4'])) {
    // Представлення блока для series_documents_body_block_4
    echo '<div class="series-documents-body_block series-documents-' . esc_attr($slug) . '">';
    $nazva_dokumentu_4 = $series_documents_body_block_4['nazva_dokumentu_4'];
    $ico_preview_4 = $series_documents_body_block_4['ico_preview'];
    $ico_copy_4 = $series_documents_body_block_4['ico_copy'];
    $ico_download_4 = $series_documents_body_block_4['ico_download'];
    $download_file_4 = $series_documents_body_block_4['download_file_4'];
    ?>
    <div class="series-documents-body_block__info">
        <p>
            <?php echo esc_html($nazva_dokumentu_4); ?>
        </p>
    </div>

    <div class="series-documents-body_block__icons">
        <a href="#">
            <?php echo ''; ?>
        </a>
        <a href="#">
            <?php echo ''; ?>
        </a>
        <a href="#">
            <?php echo ''; ?>
        </a>
    </div>

    <?php
    echo '</div>';
}

if ($series_documents_body_block_5 && !empty($series_documents_body_block_5['nazva_dokumentu_5'])) {
    // Представлення блока для series_documents_body_block_5
    echo '<div class="series-documents-body_block series-documents-' . esc_attr($slug) . '">';
    $nazva_dokumentu_5 = $series_documents_body_block_5['nazva_dokumentu_5'];
    $ico_preview_5 = $series_documents_body_block_5['ico_preview'];
    $ico_copy_5 = $series_documents_body_block_5['ico_copy'];
    $ico_download_5 = $series_documents_body_block_5['ico_download'];
    $download_file_5 = $series_documents_body_block_5['download_file_5'];
    ?>
    <div class="series-documents-body_block__info">
        <p>
            <?php echo esc_html($nazva_dokumentu_5); ?>
        </p>
    </div>
}

```

