

**Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка**

**Факультет інформаційних технологій
Кафедра мережевих та інтернет технологій**

ЗАТВЕРДЖУЮ
завідувач кафедри
мережевих та інтернет технологій
_____ **Юрій КРАВЧЕНКО**
«__» _____ 2023 року

**КВАЛІФІКАЦІЙНА РОБОТА
БАКАЛАВРА**

галузі знань 17 «Електроніка та телекомунікації»
за спеціальністю 172 «Телекомунікації та радіотехніка»
освітньо-професійна програма «Мережеві та інтернет технології»
на тему:

Розробка веб-додатку з підтримкою технології CMS

Виконав: студент групи МІТ-41

Дмитро КУЛЬЧИЦЬКИЙ _____

Керівник: доцент кафедри мережевих та інтернет технологій

д.т.н., Олександр ПЛЮЩ _____

Київ 2023

Міністерство освіти і науки України
«Київський національний університет імені Тараса Шевченка»

Факультет інформаційних технологій
Кафедра мережевих та інтернет технологій

ЗАТВЕРДЖУЮ

завідувач кафедри

мережевих та інтернет технологій

Юрій КРАВЧЕНКО

«_____» _____ 2023 року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ

Здобувачу вищої освіти

Кульчицькому Дмитру Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи:

Розробка веб-додатку з підтримкою технології CMS

затверджена на засіданні кафедри МІТ «07» грудня 2022 р. протокол №5

2. Термін здачі закінченої роботи «31» травня 2023 р.

3. Вихідні дані до проекту
(роботи)

Сучасні технології створення додатків

4. Зміст пояснювальної записки (перелік питань, що їх потрібно розробити, обсяг – 35-40 сторінок)

Вступ

1. Аналіз задачі

1.1 Мета розробки

1.2 Об'єкт та предмет дослідження

1.3 Новизна роботи

2.. Аналіз методів і засобів проектування та реалізації інформаційних систем

3. Побудова Додатку

3.1 Директорія проекту

3.2 Головка сторінка

3.3 Картка товару

3.4 Сторінка відображення товару

3.5 Редагування товару

Дата видачі завдання _____

Керівник роботи _____

Олександр ПЛЮЩ

(підпис)

(посада, прізвище, ім'я, по батькові)

Завдання прийняв до виконання _____

Дмитро КУЛЬЧИЦЬКИЙ

(підпис)

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ РОБОТИ

Номер	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Підготовчий	20.01.2023	
2	Розділ 1	15.02.2023	
3	Розділ 2	15.03.2023	
4	Розділ 3	15.04.2023	
5	Розділ 4	20.04.2023	
6	Доповідь та слайди	25.05.2023	
7	Пояснювальна записка	31.05.2023	

Здобувач вищої освіти _____ Дмитро КУЛЬЧИЦЬКИЙ

(підпис)

Керівник _____ Олександр ПЛЮЩ

(підпис)

РЕФЕРАТ

Метою розробки є створення готового до використання веб-сайту з використанням технології управління вмістом (CMS). Об'єктом розробки є веб-сайт, побудований на платформі WordPress і розміщений на веб-хостингу для доступу користувачів.

У процесі виконання проекту були досягнуті такі результати:

- Розроблено дизайн веб-сайту з використанням HTML, CSS та JavaScript.
- Придбано доменне ім'я та налаштовано його на веб-хостингу.
- Веб-сайт був успішно інтегрований на хостингу, щоб його можна було відвідувати через Інтернет.
- Були підключені функціональні компоненти, надані CMS (WordPress), для розширення можливостей сайту.

Оновлення виконуються шляхом з'єднання робочого комп'ютера з віддаленим сервером за допомогою програми FileZilla. У разі критичної помилки, яка призводить до неможливості відновлення файлів сайту з веб-хостингу, був створений репозиторій на платформі GitHub, де зберігається копія проекту.

Наразі робота над веб-сайтом перебуває у стадії альфа-тестування, що означає, що він знаходиться на початковому етапі випробування перед його остаточною релізом.

ABSTRACT

The development is aimed at obtaining a ready-to-use website built on the basis of CMS technology.

The object of development is a website built on WordPress technology and placed on a web host for users to access.

During the execution of the task, the following was achieved:

- Developed website layout in HTML and CSS, JS
- A domain and a place for it on web hosting were purchased
- The layout of the site was first successfully integrated into the hosting itself
- Connection of functional parts provided by CMS

Updating the current versions of the application depends on the connection of the development computer with a common remote server through the FileZilla program.

A repository on the GitHub site was also created due to a critical error on the impossibility of restoring site files from a remote server on the hosting.

Status: The website is in alpha testing

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ I. АНАЛІЗ ЗАДАЧІ.....	9
РОЗДІЛ II. АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНИХ СИСТЕМ.....	11
РОЗДІЛ III. ПОБУДОВА ДОДАТКУ.....	25
1. Директорії проекту.....	25
2. Головна сторінка.....	30
3. Картка товару.....	35
4. Сторінка відображення товару.....	37
5. Редагування товару.....	40
ВИСНОВОК.....	43
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	44

Вступ

У сучасному цифровому світі вимагаються ефективні та гнучкі інструменти для розробки веб-додатків. WordPress, як одна з найпопулярніших платформ у сфері веб-розробки, надає зручність та гнучкість для створення веб-додатків з підтримкою системи управління контентом (CMS). Розробка веб-додатків на основі WordPress дозволяє створити потужний та функціональний інструмент для керування контентом та надання послуг користувачам. Метою цього проекту є вивчення можливостей, особливостей та переваг цієї платформи для створення міцної та легко налаштованої веб-програми з використанням CMS, яка може бути використана як комерційна структура.

Проект охоплює огляд технології CMS, включаючи основні принципи WordPress, його архітектуру, функціональність та гнучкість у керуванні контентом. Також розглядаються етапи розробки, включаючи аналіз вимог, структуру та дизайн інтерфейсу, реалізацію функціональності та тестування. Крім того, надаються настанови щодо підтримки та оптимізації веб-програми на основі WordPress. В проекті також надаються поради щодо релізу, впровадження та подальшої підтримки програми для забезпечення її надійної роботи та оптимальної продуктивності. Розробка веб-додатків на базі WordPress відкриває безліч можливостей для реалізації різноманітних проектів. Вона поєднує потужність та надійність платформи WordPress з широким набором плагінів та постійними оновленнями, що робить її сучасною системою підтримки веб-сайтів.

При розміщенні веб-програми на веб-хостингу важливим аспектом є вибір відповідного середовища для розгортання програми на основі WordPress. Веб-хостинг - це послуга, яка фізично розміщує вашу веб-програму та її файли на сервері, забезпечуючи доступ до них через Інтернет. Dedicated WordPress Server - це варіант розміщення веб-додатку на окремому фізичному сервері, спеціально налаштованому для оптимальної роботи з платформою WordPress. Цей підхід надає більшу гнучкість, контроль та продуктивність вашої веб-

програми. Для забезпечення більшої безпеки можна створити резервну копію та розмістити її на платформі GitHub.

GitHub - це платформа для розміщення та обміну кодом, яка дозволяє зберігати, керувати та спільно працювати над проектом з різних пристроїв та співавторів. Вона надає можливість відстежувати версії вашого коду, зберігати історію змін, переглядати попередні версії та легко відновлювати стан проекту. Це дозволяє систематично працювати з кодом та забезпечує безпеку в разі втрати даних або помилкових змін.

РОЗДІЛ I. АНАЛІЗ ЗАДАЧІ

1. Мета розробки

Метою розробки інтернет-магазину на платформі WordPress полягає у наступних тезах:

Створення функціонального магазину: Основною метою є створення інтернет-магазину з необхідними функціями для продажу товарів або послуг. Це включає створення каталогу товарів, можливість додавання товарів до кошика, оформлення замовлення, оплату та доставку.

Привабливий та професійний дизайн: Метою є створення зовнішнього вигляду магазину, який буде привабливим та професійним для відвідувачів. Використання відповідного теми або налаштування дизайну допоможе створити привабливу веб-сторінку, відповідну до бренду або стилю вашого бізнесу.

Легке керування та оновлення: WordPress надає зручний інтерфейс для керування контентом вашого магазину. Метою розробки є створення інтуїтивного інтерфейсу, який дозволить вам легко додавати, редагувати та оновлювати товари, описи, зображення та інші елементи вашого магазину.

Інтеграція з платіжними шлюзами: Метою може бути інтеграція вашого магазину з платіжними шлюзами, які дозволяють приймати платежі від ваших клієнтів. Це може включати інтеграцію з популярними платіжними системами, такими як PayPal, Stripe або іншими локальними платіжними провайдерами.

Покращення SEO: Метою розробки інтернет-магазину на WordPress може бути його оптимізація для пошукових систем (SEO). Це включає оптимізацію метатегів, URL-адрес, створення XML-карти сайту та використання інших SEO-

1.2 Об'єкт та предмет дослідження

Об'єктом дослідження є використання платформи WordPress для розробки інтернет-магазину. При розробці веб-магазину були задіяні та дослідженні наступні аспекти:

Функціональність магазину: Дослідження відпрацювало можливості WordPress у створенні функціонального магазину, включаючи каталог товарів,

кошик покупок, оплату, доставку, замовлення, керування запасами тощо. Аналіз може охоплювати як базові функції WordPress, так і додаткові розширення або плагіни, що забезпечують необхідну функціональність.

Дизайн та зовнішній вигляд: можливості WordPress у створенні привабливого та професійного дизайну для інтернет-магазину є майже безмежними. Це може включати вивчення доступних тем та шаблонів, налаштування зовнішнього вигляду, роботу зі стилізацією, розташуванням елементів та іншими дизайнерськими аспектами.

Розширеність та плагіни: Дослідження має на меті розвинути можливості розширення функціональності магазину за допомогою плагінів або розширень WordPress. Аналіз охоплює різноманітні плагіни для електронної комерції, які надають додаткові можливості, такі як інтеграція з платіжними системами, підвищення безпеки, управління запасами тощо.

1.3 Новизна роботи

Більш глибока розробка даного додатку може бути корисною для малих та середніх підприємств які ще не мають власного інтернет ресурсу для продажу товарів та послуг.

Однією з ключових особливостей цієї роботи є поєднання двох потужних технологій - WordPress та Woocommerce. WordPress - це найпопулярніша CMS для побудови комерційних сайтів. Woocommerce, в той час, є зручним засобом для розробки веб-магазину, він має в собі можливість згенерувати замовлення користувачів сайту.

Перевага цього додатку у його масштабованості та модульності. Завдяки використанню WordPress та Woocommerce, розробник може легко додавати нові продукти, оновлювати та модифікувати існуючі, а також змінювати дизайн сайту за власним бажанням. Це дозволяє забезпечити імплементацію нових ідей та актуальність для відвідувачів.

Таким чином, інтернет магазин, створений на основі технологій WordPress та Woocommerce, пропонує новаторський підхід до власного бізнесу. А також простоту в його організації.

РОЗДІЛ II. АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНИХ СИСТЕМ

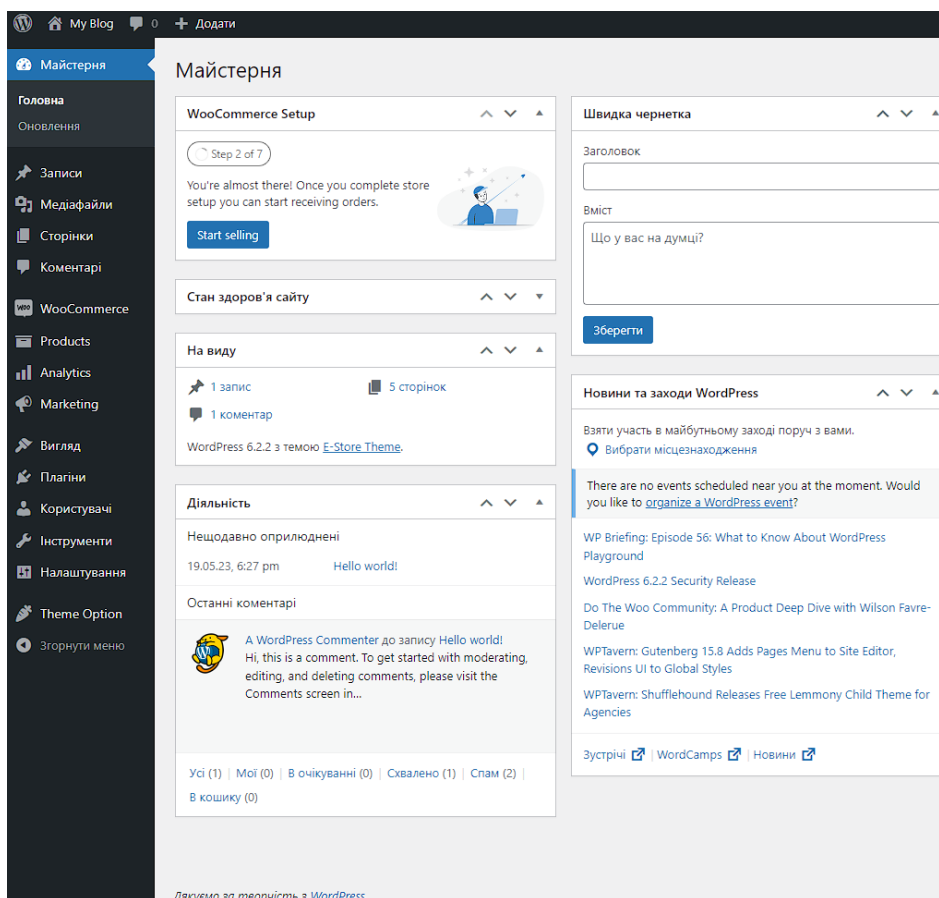
В розробці веб-програми за допомогою WordPress існують певні переваги. Простота використання: WordPress відомий своїм зручним інтерфейсом та інтуїтивно зрозумілою системою керування вмістом (CMS). Він надає графічний інтерфейс користувача, який спрощує процес створення вмісту веб-сайту та керування ним. Це робить його доступним для користувачів із різним рівнем технічного досвіду, дозволяючи їм легко оновлювати та змінювати додаток. Широкий вибір тем і плагінів доступний на WordPress через величезну бібліотеку тем і плагінів, які можуть покращити функціональність і дизайн вашої веб-програми. Тем дозволяють налаштувати зовнішній вигляд вашого сайту, а плагіни додають певні функції та функції. Ця широка екосистема економить час і зусилля на розробку, надаючи готові рішення для різних вимог. Можливості керування вмістом дозволяє WordPress чудово керувати вмістом, що робить його придатним для веб-додатків, які потребують частого оновлення вмісту. Він надає такі функції, як створення дописів у блозі, керування медіафайлами, ролі користувачів і дозволи, а також оптимізація SEO.

Ці можливості дозволяють ефективно керувати вмістом вашої програми та впорядковувати її. Гнучкість і масштабованість CMS надає гнучку структуру, яка може адаптуватися до різних типів веб-додатків. Незалежно від того, чи створюєте ви веб-сайт для малого бізнесу, платформу електронної комерції, сайт-портфолію чи форум спільноти, WordPress може задовольнити ваші потреби. Він також пропонує масштабованість, що дозволяє вашій програмі розвиватися та обробляти збільшений трафік у міру розширення вашого бізнесу. Потужна підтримка спільноти надає WordPress велику та активну спільноту розробників і користувачів. Це означає, що ви можете знайти розширену документацію, навчальні посібники, форуми та онлайн-ресурси, які допоможуть вам у вирішенні будь-яких проблем розробки, з якими ви можете

зіткнутися. Спільнота також сприяє постійному вдосконаленню WordPress, забезпечуючи регулярні оновлення та виправлення безпеки.

Зручність для SEO: WordPress розроблено з урахуванням оптимізації пошукових систем (SEO). Він генерує чистий і семантичний код, надає настроювані URL-адреси, дозволяє легко інтегрувати плагіни SEO та пропонує інструменти для оптимізації мета-тегів, заголовків і вмісту. Ці вбудовані функції сприяють кращій видимості пошукової системи та органічному трафіку для вашої веб-програми. Економічне рішення: у порівнянні з індивідуальною розробкою з нуля, створення веб-додатку за допомогою CMS може бути економічно ефективним рішенням. Наявність безкоштовних тем і плагінів, а також легкість розробки та обслуговування можуть допомогти зменшити витрати на розробку та час виходу на ринок.

На Рисунку 2.1 наданий вигляд головної сторінки панелі управління WordPress



Дякуємо за творчість з WordPress.

Рисунок 2.1 – Вигляд головної сторінки панелі управління WordPress

Розробляючи веб-програму за допомогою WordPress, ви можете розглянути кілька методів і підходів Налаштування теми: цей метод включає в себе початок із наявної теми WordPress і налаштування її відповідно до вимог вашої програми. Ви можете змінити дизайн, макет і функціональність теми за допомогою настрювача WordPress або безпосередньо відредагувавши код теми. Цей підхід підходить, коли потреби вашої програми тісно пов'язані з функціями та структурою, наданими вибраною темою.

Розробка дочірньої теми: дочірня тема — це тема, яка успадковує функціональність і стиль батьківської теми. За допомогою дочірньої теми ви можете робити налаштування, не змінюючи безпосередньо код батьківської теми. Цей метод корисний, коли ви хочете підтримувати оновлення та сумісність батьківської теми, одночасно додаючи спеціальні функції, специфічні для вашої веб-програми. Розробка плагінів на WordPress забезпечує надійну архітектуру плагінів, яка дозволяє розширити основні функції CMS. Розробка користувальницьких плагінів дає змогу додавати нові функції, змінювати наявні функції або інтегрувати служби сторонніх розробників у веб-програму. Цей метод забезпечує гнучкість і дозволяє модульну розробку, зберігаючи ваш код окремо від теми та полегшуючи оновлення та обслуговування. Розробка спеціальної теми: якщо вашій веб-програмі потрібен унікальний дизайн або складна функціональність, розробка спеціальної теми може бути найкращим підходом. Це передбачає створення теми з нуля за допомогою HTML, CSS, JavaScript і PHP. Користувальницькі теми пропонують повний контроль над дизайном і функціональністю програми, дозволяючи налаштувати кожен аспект відповідно до ваших конкретних вимог. Інтеграція REST API: WordPress представив потужний REST API, який забезпечує зв'язок між WordPress і зовнішніми програмами. Цей метод передбачає розробку окремого зовнішнього додатку з використанням таких фреймворків, як React, Angular або Vue.js, і використання WordPress як безголової CMS. Такий підхід

забезпечує більшу гнучкість щодо дизайну, інтерактивності та інтеграції із зовнішніми системами. Розробка електронної комерції: якщо ваша веб-програма передбачає продаж продуктів або послуг онлайн, ви можете використовувати плагіни WordPress, як-от WooCommerce, для розробки платформи електронної комерції. WooCommerce надає низку функцій для керування продуктами, запасами, платежами та обробкою замовлень. Налаштувавши та розширивши WooCommerce, ви можете створити індивідуальне рішення для електронної комерції, яке відповідає вимогам вашого бізнесу. WordPress пропонує функцію мультисайту, яка дозволяє створювати та керувати кількома веб-сайтами з однієї інсталяції WordPress. Цей метод підходить, коли вам потрібно розробити мережу пов'язаних сайтів або надати користувачам платформу для створення власних веб-сайтів. Кожен сайт у мережі може мати власну тему, плагіни та вміст, користуючись однаковою основною функціональністю.

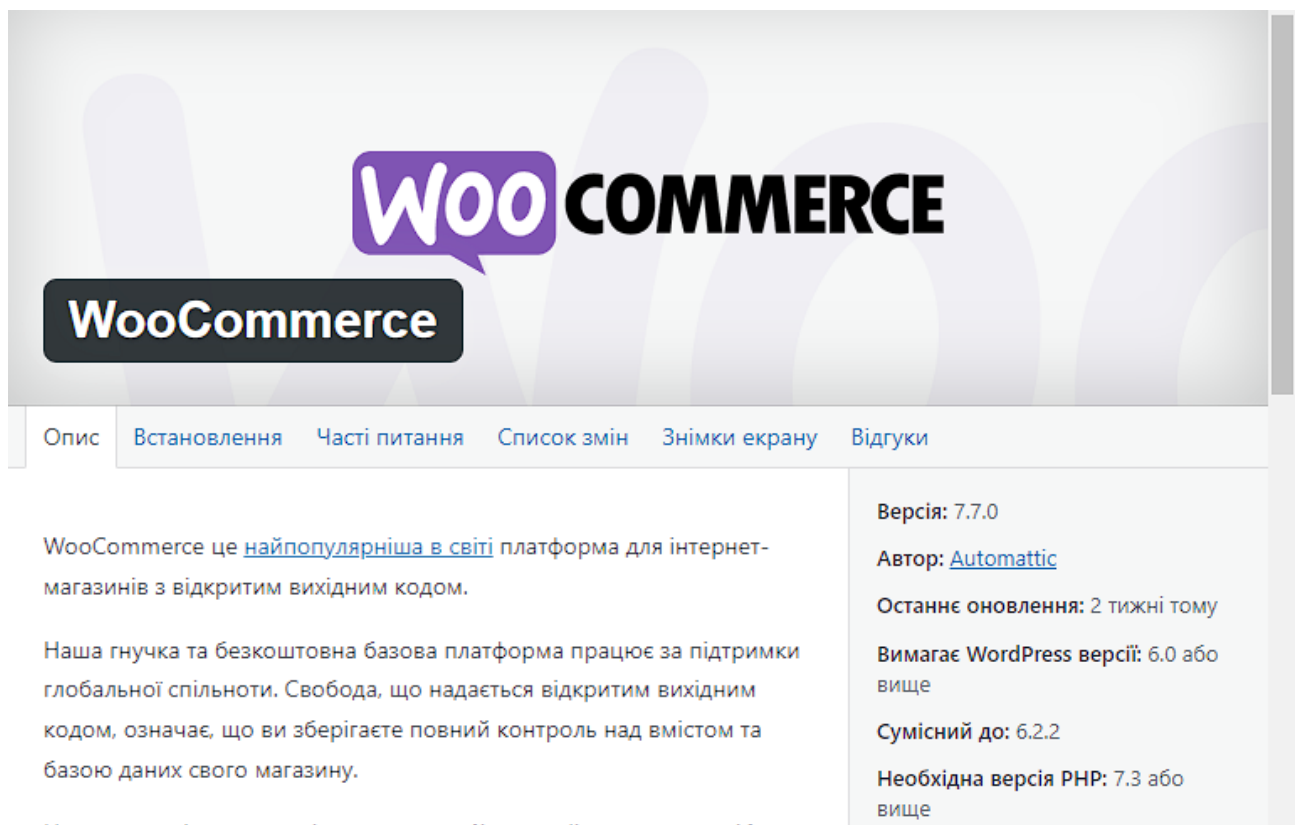


Рисунок 2.2 – Вигляд плагіна WooCommerce

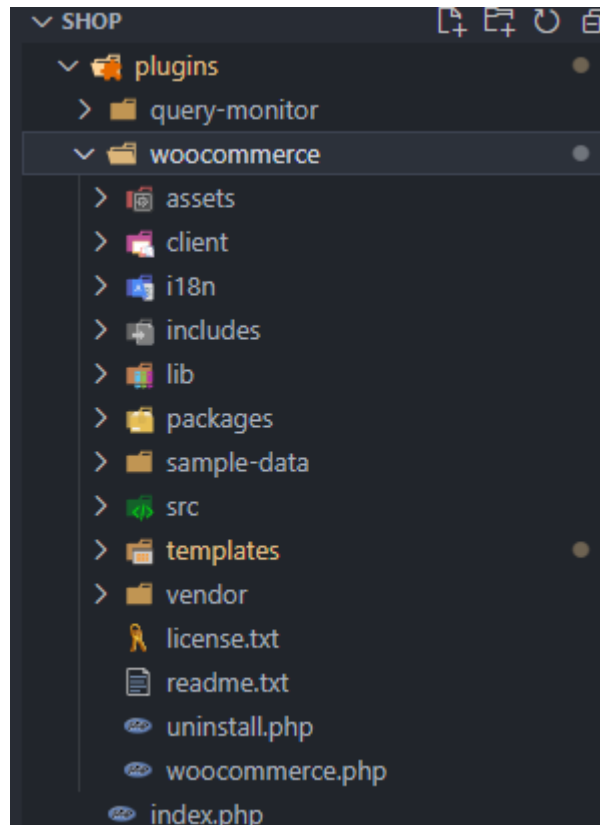


Рисунок 2.3 – Вигляд завантажуванного контенту з плагіна

В цій папці ми можемо побачити вже написані функції які звертаються до WordPress. Рисунок 2.3

```

if ( ! defined( 'ABSPATH' ) ) {
    exit; // Exit if accessed directly
}

get_header( 'shop' ); >>

<?php
/**
 * woocommerce_before_main_content hook.
 *
 * @hooked woocommerce_output_content_wrapper - 10 (outputs opening divs for the content)
 * @hooked woocommerce_breadcrumb - 20
 */
do_action( 'woocommerce_before_main_content' );
?>

<?php while ( have_posts() ) : the_post(); ?>

    <?php wc_get_template_part( 'content', 'single-product' ); ?>

<?php endwhile; // end of the loop. ?>

<?php
/**
 * woocommerce_after_main_content hook.
 *
 * @hooked woocommerce_output_content_wrapper_end - 10 (outputs closing divs for the content)
 */
do_action( 'woocommerce_after_main_content' );
?>

<!-- removed in wc-functions-remove -->
<?php
/**
 * woocommerce_sidebar hook.
 *
 * @hooked woocommerce_get_sidebar - 10
 */
do_action( 'woocommerce_sidebar' );
?>

<?php get_footer( 'shop' );

/* Omit closing PHP tag at the end of PHP files to avoid "headers already sent" issues. */

```

Рисунок 2.4 Файл single-product.php

Файл single-product.php у WooCommerсе є шаблоном, який відповідає за відображення окремої сторінки продукту у вашому інтернет-магазині. Він контролює вигляд і розташування інформації про продукт, зображень, цін та пов'язаних елементів на цій сторінці. Цей файл є частиною ієрархії шаблонів WooCommerсе, яка використовується для відтворення різних типів сторінок. Якщо файл single-product.php відсутній, WooCommerсе використовує стандартний файл single.php з вашої теми.

У файлі single-product.php ви можете налаштувати макет і структуру сторінки окремого продукту. Він містить HTML-розмітку, CSS-стилі і PHP-код для отримання та відображення відповідної інформації про продукт. Цей файл дозволяє вам впливати на дизайн і вигляд сторінки, враховуючи ваші вимоги.

У файлі `single-product.php` ви можете отримати доступ до різних відомостей про продукт, таких як назва, опис, SKU, наявність, категорії, теги, атрибути і будь-які користувацькі поля або метадані, пов'язані з продуктом.

Цей файл також відповідає за відображення зображень продукту. Ви можете використовувати функції і зачіпки, щоб вивести основне зображення продукту, додаткову галерею зображень і функцію збільшення, якщо вона активована. Ви можете керувати розмірами зображень і поведінкою галереї в цьому файлі шаблону.

Файл `single-product.php` містить необхідний код для відображення ціни продукту, знижкової ціни (якщо вона є) і будь-яких варіацій цін. Він також підтримує функцію "Додати до кошика", яка дозволяє клієнтам додавати продукт до кошика з вказаною кількістю та варіантами.

Залежно від вашої теми і налаштувань, файл `single-product.php` може також містити секції для відображення пов'язаних продуктів або додаткових пропозицій. Ці розділи демонструють інші продукти, які доповнюють або схожі на поточний продукт, спонукаючи клієнтів розглянути додаткові варіанти.

WooCommerce надає різні хуки та фільтри в файлі `single-product.php`, які дозволяють розробникам налаштовувати і розширювати його функціональність. Ви можете змінювати вихідні дані, додавати спеціальні елементи або інтегрувати сторонні плагіни за допомогою цих хуків, щоб налаштувати

сторінку окремого продукту згідно з вашими потребами.

```

<div id="product-<?php the_ID(); ?>" <?php post_class(); ?>>

    <?php
        /**
         * woocommerce_before_single_product_summary hook.
         *
         * @hooked woocommerce_show_product_sale_flash - 10
         * @hooked woocommerce_show_product_images - 20
         */
        do_action( 'woocommerce_before_single_product_summary' );
    ?><!-- .beforesummary -->
    <div class="content__buy">
        <?php
            /**
             * woocommerce_single_product_summary hook.
             *
             * @hooked woocommerce_template_single_title - 5
             * @hooked woocommerce_template_single_rating - 10
             * @hooked woocommerce_template_single_price - 10
             * @hooked woocommerce_template_single_excerpt - 20
             * @hooked woocommerce_template_single_add_to_cart - 30
             * @hooked woocommerce_template_single_meta - 40
             * @hooked woocommerce_template_single_sharing - 50
             * @hooked WC_Structured_Data::generate_product_data() - 60
             */
            do_action( 'woocommerce_single_product_summary' );
        ?>
    </div>
    <!-- .summary -->

    <?php
        /**
         * woocommerce_after_single_product_summary hook.
         *
         * @hooked woocommerce_output_product_data_tabs - 10
         * @hooked woocommerce_upsell_display - 15
         * @hooked woocommerce_output_related_products - 20
         */
        do_action( 'woocommerce_after_single_product_summary' );
    ?>

</div><!-- #product-<?php the_ID(); ?> -->

<?php do_action( 'woocommerce_after_single_product' ); ?>

```

Рисунок 2.5 Файл content-single-product.php

У WooCommerce файл content-single-product.php Рисунок 2.5 це інший файл шаблону, який використовується для відображення вмісту одного

продукту в циклі на сторінках архіву, наприклад на сторінках категорій або тегів. Він відповідає за відтворення інформації про продукт і витягу, коли кілька продуктів відображаються разом. Подібно до інших файлів шаблонів WooCommerce, `content-single-product.php` дотримується системи ієрархії шаблонів. Він використовується для відображення вмісту одного продукту в циклі на сторінках архіву. Якщо цього файлу немає, WooCommerce повертається до стандартного файлу `content.php` у вашій темі. Інформація про продукт: файл `content-single-product.php` містить код для відображення різних деталей продукту в циклі. Це може включати назву продукту, зображення, ціну, розпродажну ціну, короткий опис та будь-яку іншу відповідну інформацію, яку ви вирішите включити. Він надає гачки та функції для динамічного виведення цих деталей для кожного продукту. Якщо ви налаштували витяги продукту або резюме, файл `content-single-product.php` обробляє відображення витягу продукту в циклі. Це дозволяє показувати короткий для кожного продукту, надаючи огляд його функцій або спонукаючи клієнтів переглянути повну інформацію про продукт.

```
<?php
/**
 * The main template file
 *
 * This is the most generic template file in a WordPress theme
 * and one of the two required files for a theme (the other being style.css).
 * It is used to display a page when nothing more specific matches a query.
 * E.g., it puts together the home page when no home.php file exists.
 *
 * @link https://developer.wordpress.org/themes/basics/template-hierarchy/
 *
 * @package estore
 */

get_header(); ?>

<main class="main">...
<?php
get_sidebar();
get_footer();
```

Рисунок 2.6 - Файл index.php

У WordPress існує основний файл шаблону "Рисунок 2.6", який використовується за замовчуванням як резервний шаблон для відображення вмісту домашньої сторінки веб-сайту та інших сторінок, коли немає доступних більш спеціалізованих шаблонів. Цей файл, `index.php`, є частиною шаблонної ієрархії в WordPress. Він є найбільш загальним шаблонним файлом і використовується як альтернативний варіант, якщо не існує більш конкретних шаблонів. У випадку відсутності спеціального файлу шаблону для конкретної сторінки або типу публікації, WordPress автоматично використовує `index.php` для відображення вмісту. Цикл вмісту: зазвичай, у файлі `index.php` міститься код основного циклу вмісту, який отримує та відображає дописи або інший вміст відповідно до запиту за замовчуванням. Цей цикл ітерується через кожен елемент і генерує відповідну розмітку HTML для відображення вмісту на сторінці.

Відображення дописів у блозі: якщо на вашому веб-сайті є блоговий розділ, файл `index.php` відповідає за відображення дописів у цьому розділі. Він отримує повідомлення блогу з бази даних і форматує їх відповідно до структури циклу, яка визначена у файлі. Ви можете налаштувати цей цикл для контролю кількості відображених публікацій, порядку їх виведення та додавання будь-якого додаткового стилю або вмісту за вашим бажанням.

Розбиття на сторінки: якщо на вашій домашній або інших сторінках присутня велика кількість публікацій або елементів вмісту, файл `index.php` може містити функціонал розбиття на сторінки. Це дає користувачам змогу навігувати між кількома сторінками вмісту та забезпечує зручне прокручування сторінок.

Налаштування та хуки: у файлі `index.php` також містяться різні хуки та фільтри, які дозволяють налаштовувати його вихідні дані та функціонал. Ви можете змінювати параметри циклу, додавати користувацькі елементи або інтегрувати плагіни та віджети, щоб поліпшити вміст, який відображається на домашній або інших сторінках, за допомогою цих хуків.

Умовна логіка: Ви також маєте можливість використовувати умовну логіку в файлі `index.php` для обробки специфічних сценаріїв або відображення вмісту залежно від певних умов. Наприклад, ви можете перевірити, чи поточна сторінка є домашньою сторінкою або сторінкою певної категорії і змінити результат відповідно до цього.

Включення частини шаблону: у багатьох випадках файл `index.php` включає інші частини шаблону або файли для відображення конкретних розділів сторінки. Цей модульний підхід дозволяє розбити різні розділи сторінки на окремі файли, спрощуючи процес обслуговування та налаштування.

Шорткоди в WooCommerce: шорткоди є спеціальними кодовими фрагментами, які дозволяють додавати динамічний вміст або функціональність до вашого онлайн-магазину без необхідності глибоких знань в програмуванні. Шорткоди, зазвичай записані в квадратних дужках, можуть розміщуватись в різних місцях, таких як сторінки, публікації, віджети або навіть в тематичних файлах. Шорткоди діють як заповнювачі, які замінюються динамічним вмістом при відображенні сторінки або публікації на передній частині вашого веб-сайту. Вони дозволяють відображати та контролювати конкретні елементи або функції, надані WooCommerce або сторонніми плагінами розробників. WooCommerce надає кілька вбудованих шорткодів, які дозволяють відображати списки продуктів, категорії продуктів, вміст кошика і т.д.

Ось кілька прикладів часто використовуваних шорткодів в WooCommerce:

Відображення продукту: Шорткод `[products]` дозволяє відобразити сітку продуктів. Ви можете налаштувати його, використовуючи різні атрибути, такі як ліміт, кількість стовпців, порядок тощо. Наприклад, `[products limit="4" columns="2" orderby="popularity"]` відобразить чотири популярних продукти в сітці з двома стовпцями.

Категорії продуктів: Шорткод `[product_categories]` містить список категорій продуктів у вашому магазині. Ви можете використовувати атрибути, такі як `number`, `parent` або `hide_empty`, щоб налаштувати відображення.

Сторінка кошика: Шорткод [woocommmerce_cart] створює сторінку кошика, на якій клієнти можуть переглядати та змінювати вміст свого кошика перед покупкою.

Сторінка оформлення замовлення: Шорткод [woocommmerce_checkout] створює сторінку оформлення замовлення, де клієнти можуть ввести свою платіжну інформацію та інформацію про доставку для завершення покупки.

Ці шорткоди дозволяють вам контролювати відображення та функціональність окремих елементів вашого онлайн-магазину, спрощуючи процес створення та налаштування магазину.

Крім цього, сторонні плагіни можуть використовувати власні короткі коди для додаткових функцій та інтеграцій.

Шорткоди дозволяють легко додавати динамічні елементи до вашого магазину WooCommerce та налаштовувати представлення продуктів і вмісту без потреби внесення змін у файли теми або написання складного коду.

Варто відзначити, що у цьому додатку були створені додаткові шорткоди для розширення кількості сторінок. Наприклад, на Рисунку 2.7 показано шорткод для сторінки "Контакти".

```
no usages
function woocommerce_map_shortcode() {
    // Your delivery-related logic goes here
    // Generate and return the delivery information
    ?>
    <!-- HTML markup for delivery information -->
    <?php
    wc_get_template( template_name: 'map/map.php' );
    ?>
    <?php
}
add_shortcode('woocommerce_map', 'woocommerce_map_shortcode');
```

Рисунок 2.8 Створення функції і додавання неї в створений нижче шорткод

Контакти

[/] Shortcode

+

Рисунок 2.8 - Виклик шорткоду

```
<?php
/** The template for displaying all pages ...*/

get_header();

do_action( hook_name: 'woocommerce_before_main_content' );

?>

<!-- <main class="site-main">-->
<!-- <div class="container">-->

    <?php
    while ( have_posts() ) : the_post();

        get_template_part( slug: 'template-parts/content', name: 'page' );

        // If comments are open or we have at least one comment, load up the comment template.
        if ( comments_open() || get_comments_number() ) :
            comments_template();
        endif;

    endwhile; // End of the loop.
    ?>
<!-- </div-->
<!-- </main-->

<?php

do_action( hook_name: 'woocommerce_after_main_content' );

get_sidebar();
get_footer();
```

Рисунок 2.9 вигляд файлу page.php

Окремої уваги потребує файл зазначений на Рисунку 2.9. Він є шаблоном для усіх сторінок які будуть створюватися додатково. На Рисунку 2.9 рядок `get_template_part('template-parts/content', 'page');`. Визиває файл `content-page.php` який знаходиться в папці `template-parts`. Таким чином ми отримуємо все що знаходиться в тому файлі й записуємо його на місце функції.

```
<?php
/** Template part for displaying page content in page.php ...*/
?>

<div class="page">
    <div class="page__container">
        <div class="page__body">

            <?php
                the_content();
            ?>

        </div>
    </div>
</div>
```

Рисунок 2.10 вигляд файлу content-page.php

На Рисунку 2.10 відображений вигляд шаблону по якому будуть створюватися сторінки не передбачені за замовчанням. Функція `the_content()`; викликає шорткод який прописаний як це зроблено на Рисунку 2.8 за цим іде виклик функції яка приписана до цього шорткоду як це показано на Рисунку 2.7

РОЗДІЛ III. ПОБУДОВА ДОДАТКУ

1. Директорії проекту

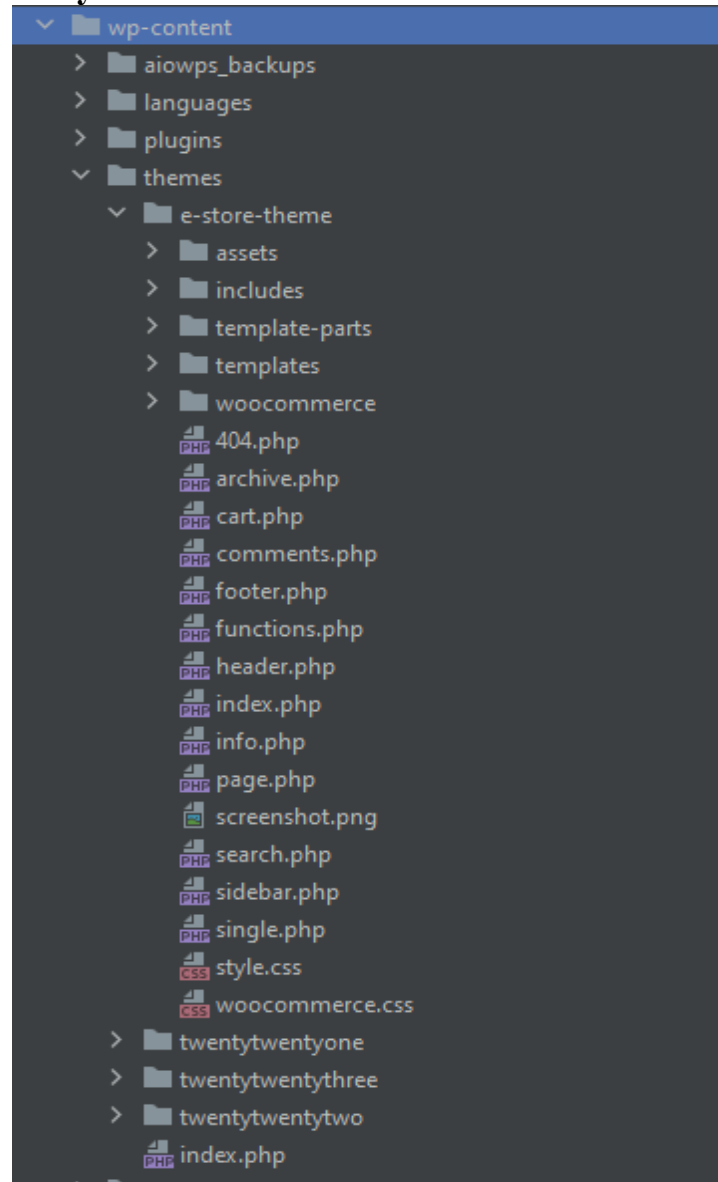


Рисунок 3.1 - Вигляд робочих директорій

На Рисунку 3.1 Зображено простір в якому буде виконуватися робота. Основні робочі папки це plugins та themes.

1.1 Директорія plugins

WooCommerce створює спеціальну папку плагінів з назвою "woocommerce" у каталозі "wp-content/plugins". В цій папці знаходяться основні файли плагіна WooCommerce, як показано на Рисунок 3.1.1. Однак, ви можете додавати додаткові функції та можливості, встановлюючи інші плагіни, які є сумісними з WooCommerce.

Для забезпечення сумісності та безпеки важливо регулярно оновлювати плагіни. Ви можете керувати та оновлювати плагіни з розділу "Плагіни" на панелі адміністратора WordPress.

Додаткові плагіни можна завантажити з сховища плагінів WordPress.org або отримати від сторонніх розробників. Після завантаження плагіна ви можете встановити його, виконавши наступні кроки:

1. Увійдіть до панелі адміністрування WordPress.
2. Перейдіть до розділу "Плагіни" і клацніть на "Додати новий".
3. Натисніть кнопку "Завантажити плагін".
4. Виберіть файл плагіна (зазвичай це файл .zip) на своєму комп'ютері і клацніть "Установити зараз".
5. Після встановлення клацніть кнопку "Активувати", щоб активувати плагін.

Після активації функції плагіна стануть доступними в налаштуваннях WooCommerce. Залежно від конкретного плагіна, вам може знадобитись налаштувати його параметри або виконати додаткові кроки налаштування.

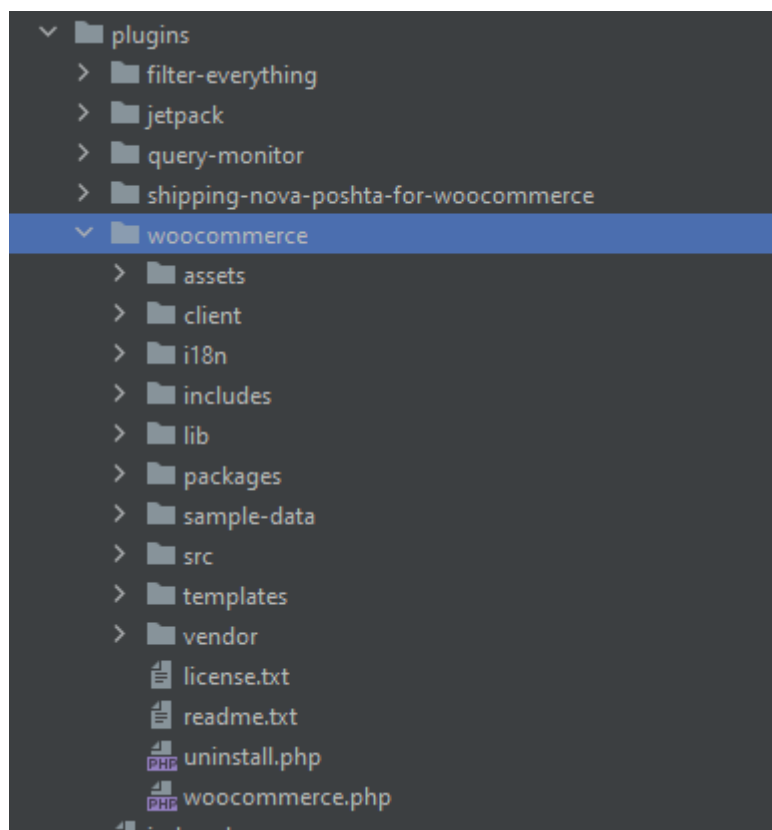


Рисунок 3.1.1 Видяд папки woocommerce

1.2. Директорія themes

У WordPress теми зберігаються в папці «wp-content/themes». Після встановлення WordPress у вас вже буде наявна тема за замовчуванням, наприклад «Twenty Twenty-One» або «Twenty Twenty-Two», яку можна знайти у папці тем, як показано на Рисунку 3.1.2. Щоб встановити нову тему або змінити поточну, вам потрібно виконати такі кроки:

1. Увійдіть до панелі адміністрування WordPress.
2. Перейдіть до розділу "Вигляд" і клацніть на "Теми".
3. У розділі "Теми" буде відображений список встановлених тем.
4. Щоб додати нову тему, клацніть кнопку "Додати нову".

Якщо вам потрібна тема, яка доступна у каталозі тем WordPress, ви можете здійснити пошук за назвою або переглянути доступні теми. Після знаходження потрібної теми, ви можете натиснути кнопку "Встановити", а потім кнопку "Активувати", щоб активувати її. Якщо у вас є файл теми у форматі .zip, ви можете клацнути кнопку "Завантажити тему". Потім виберіть файл теми на своєму комп'ютері і клацніть "Установити зараз". Після встановлення ви можете активувати тему.

Після активації теми зовнішній вигляд вашого веб-сайту WordPress буде оновлено відповідно до обраної теми. Ви також можете налаштувати параметри, макет і дизайн теми за допомогою настроювача WordPress або параметрів конкретної теми. Важливо регулярно оновлювати теми для забезпечення сумісності, безпеки та доступу до останніх функцій. Ви можете керувати своїми темами і оновлювати їх у розділі "Теми" на панелі адміністратора WordPress.

У цій роботі була створена власна тема під назвою "e-store-theme".

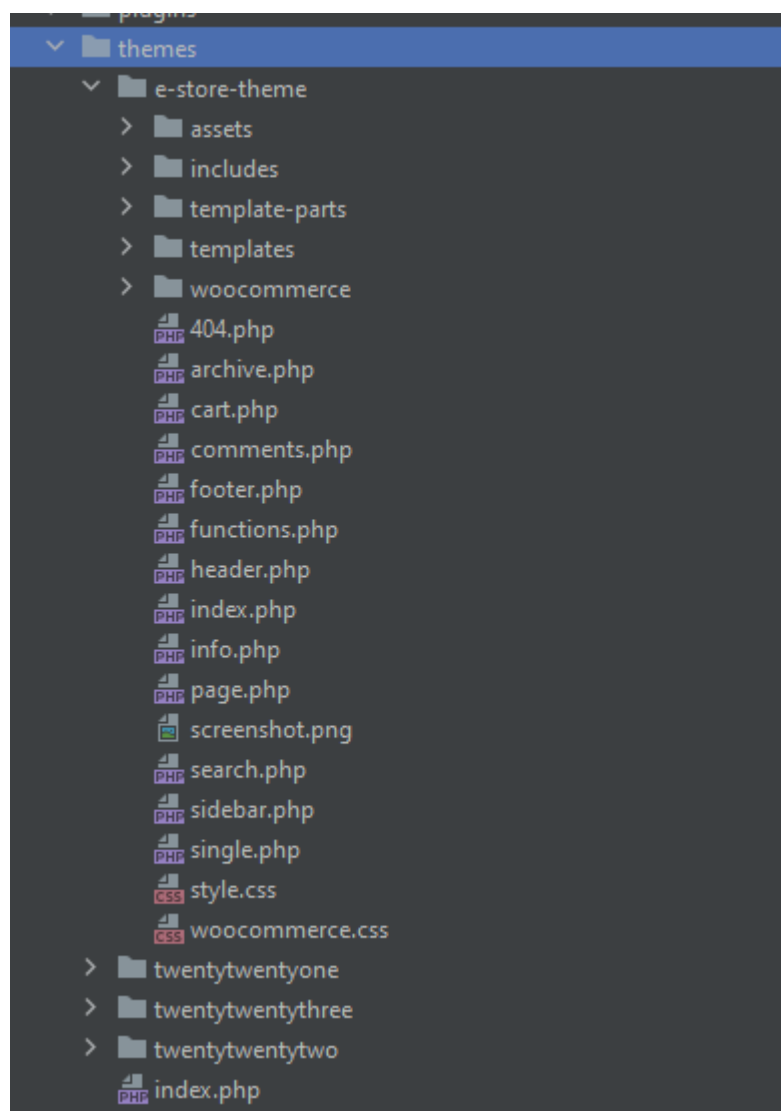


Рисунок 3.1.2 Видяд папки themes

2. Головна сторінка

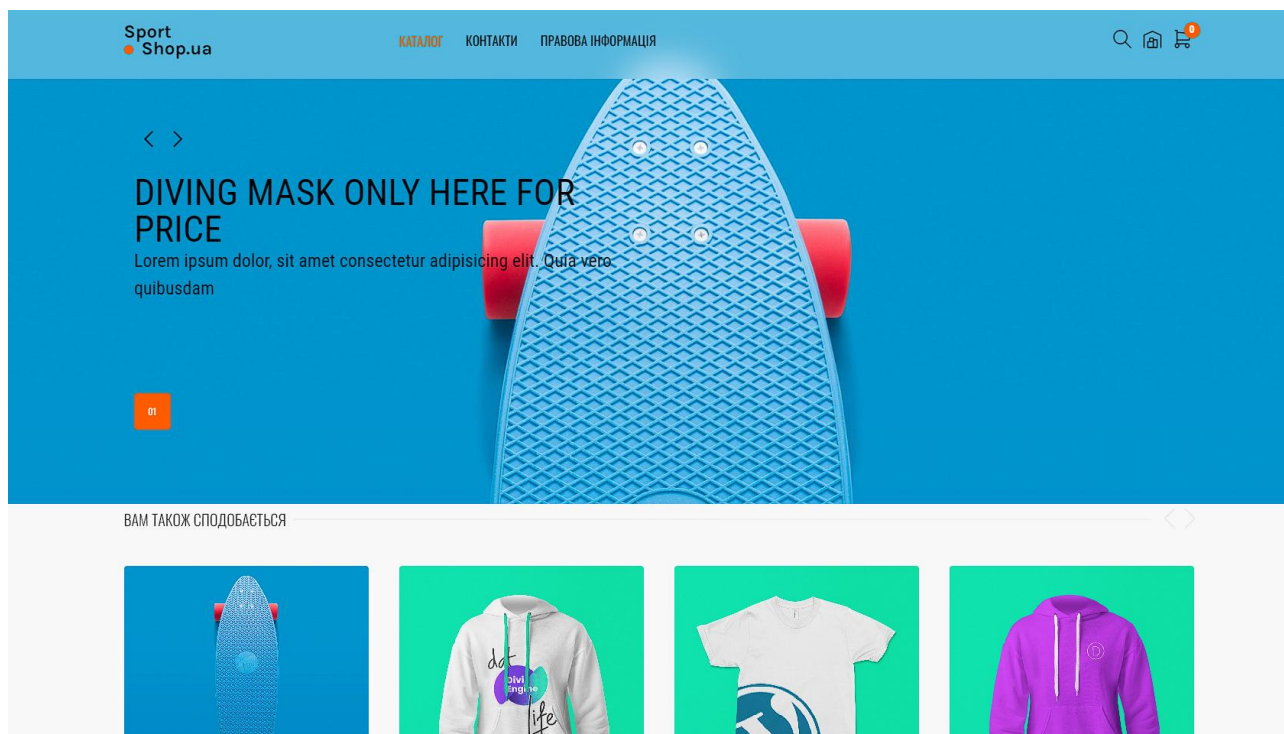


Рисунок 3.2.1 - Загальний вигляд головної сторінки веб-додатку

Перейшовши по адресі <http://shopdip.sbs/> користувач потрапляє на



Рисунок 3.2.2 вигляд “шапки” веб-додатка

Рисунок 3.2.2 можна поділити на 2 частини а саме на Рисунок 3.2.3 та

Рисунок 3.2.4



Рисунок 3.2.3 головне меню

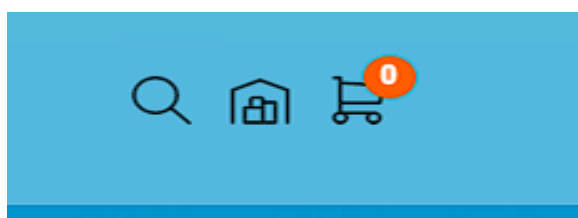


Рисунок 3.2.4 панель пошуку, кабінет покупця та кошик

На Рисунку 3.2.3 показано меню сайту в якому є наступні:

- Каталог
- Контакти
- Правова Інформація

Елемент Каталог відображає категорії товарів та їх підкатегорії Рисунок 3.2.5



Рисунок 3.2.5 – вигляд елемента каталогу

Елемент Контакти відправляє користувача на Рисунок 3.2.6

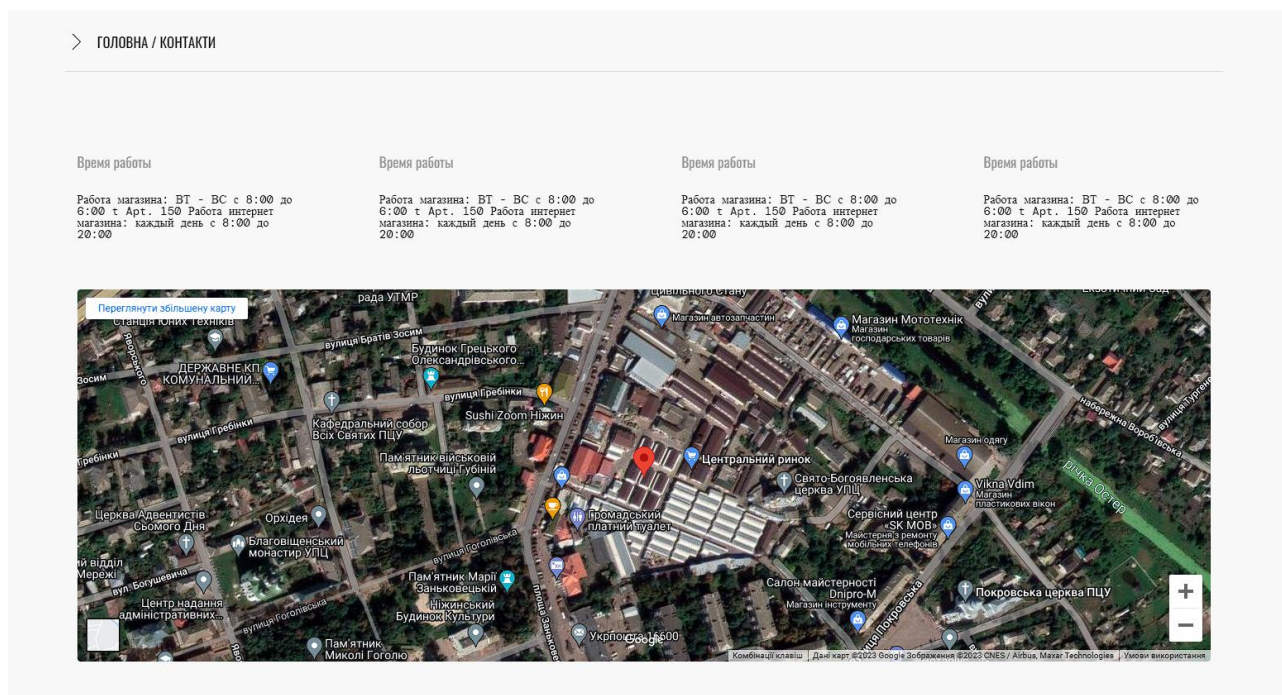


Рисунок 3.2.6 – сторінка Контакти

Елемент Права Інформація відправляє користувача на Рисунок 3.2.7

Sport Shop.ua КАТАЛОГ КОНТАКТИ ПРАВОВА ІНФОРМАЦІЯ

ГОЛОВНА / ПРАВОВА ІНФОРМАЦІЯ

ДОГОВОР ПУБЛИЧНОЇ ОФЕРТИ
Договор публичной оферты

Общие положения

1.1. Настоящая оферта, является официальным предложением ТМ «Planet-shop.com.ua», далее по тексту – «Продавец», заключить Договор купли-продажи товаров дистанционным способом, то есть через Интернет-магазин, далее по тексту – «Договор», и размещает Публичную оферту (предложение) на официальном интернет-сайте Продавца «https://planet-shop.com.ua/» (далее - «Интернет-сайт»).

1.2. Моментом полного и безусловного принятия Покупателем предложения Продавца (акцептом) заключить электронный договор купли-продажи товаров, считается факт оплаты Покупателем заказа на условиях настоящего Договора, в сроки и по ценам, указанным на Интернет-сайте Продавца.
Понятия и определения

2.1. В настоящей оферте, если контекст не требует иного, нижеприведенные термины имеют следующие значения: * «товары» - модели, аксессуары, комплектующие и сопроводительные предметы; * «Интернет-магазин» - в соответствии с Законом Украины «об электронной коммерции», средство для представления или реализации товара, работы или услуги путем совершения электронной сделки; * «Продавец» - компания, реализующая товары, представленные на Интернет-сайте; * «Покупатель» - физическое лицо, заключающее с Продавцом Договор на условиях, изложенных ниже; * «Заказ» - выбор отдельных позиций из перечня товаров, указанных Покупателем при размещении заказа и проведении оплаты. Предмет Договора

3.1. Продавец обязуется передать в собственность Покупателя Товар, а Покупатель обязуется оплатить и принять Товар на условиях данного Договора. Настоящий Договор регулирует куплю-продажу товаров в Интернет-магазине, в том числе «Заказ» - выбор отдельных позиций из перечня товаров, указанных Покупателем при размещении заказа и проведении оплаты. Предмет Договора

Настоящий Договор регулирует куплю-продажу товаров в Интернет-магазине, в том числе «Заказ» - выбор отдельных позиций из перечня товаров, указанных Покупателем при размещении заказа и проведении оплаты.

Предмет Договора
ДОГОВОР ПУБЛИЧНОЇ ОФЕРТИ
Договор публичной оферты

Общие положения

1.1. Настоящая оферта, является официальным предложением ТМ «Planet-shop.com.ua», далее по тексту – «Продавец», заключить Договор купли-продажи товаров дистанционным способом, то есть через Интернет-магазин, далее по тексту – «Договор», и размещает Публичную оферту (предложение) на официальном интернет-сайте Продавца «https://planet-shop.com.ua/» (далее - «Интернет-сайт»).

1.2. Моментом полного и безусловного принятия Покупателем предложения Продавца (акцептом) заключить электронный договор купли-продажи товаров, считается факт оплаты Покупателем заказа на условиях настоящего Договора, в сроки и по ценам, указанным на Интернет-сайте Продавца.
Понятия и определения

2.1. В настоящей оферте, если контекст не требует иного, нижеприведенные термины имеют следующие значения: * «товары» - модели, аксессуары, комплектующие и сопроводительные предметы; * «Интернет-магазин» - в соответствии с Законом Украины «об электронной коммерции», средство для представления или реализации товара, работы или услуги путем совершения электронной сделки; * «Продавец» - компания, реализующая товары...

Рисунок 3.2.7 - вигляд сторінки Права Інформація

ВАМ ТАКОЖ СПОДОБАЄТЬСЯ

Test	Dat Divi Engine Life Hoodie - Limited Edition ₴20.00	WordPress Tee ₴134.00	Mens Divi Hoodie ₴354.00
------	---	--------------------------	-----------------------------

Рисунок 3.2.8 - слайдер товарів

На Рисунку 3.2.8 відображаються товари які пропонуються користувачу, Також кількість відображаємих товарів регульована, в донному випадку вона встановлена за замовчуванням, а саме на значення 10 карток Рисунок 3.2.8.

```
<?php
$products = wc_get_products(array(
    'status' => 'publish', // Retrieve only published products
    'limit' => 10, // Adjust the number of products to display
));
?>
<div class="recommend-slider">
```

Рисунок 3.2.9 - налаштування відображення товарів

3. Картка товару

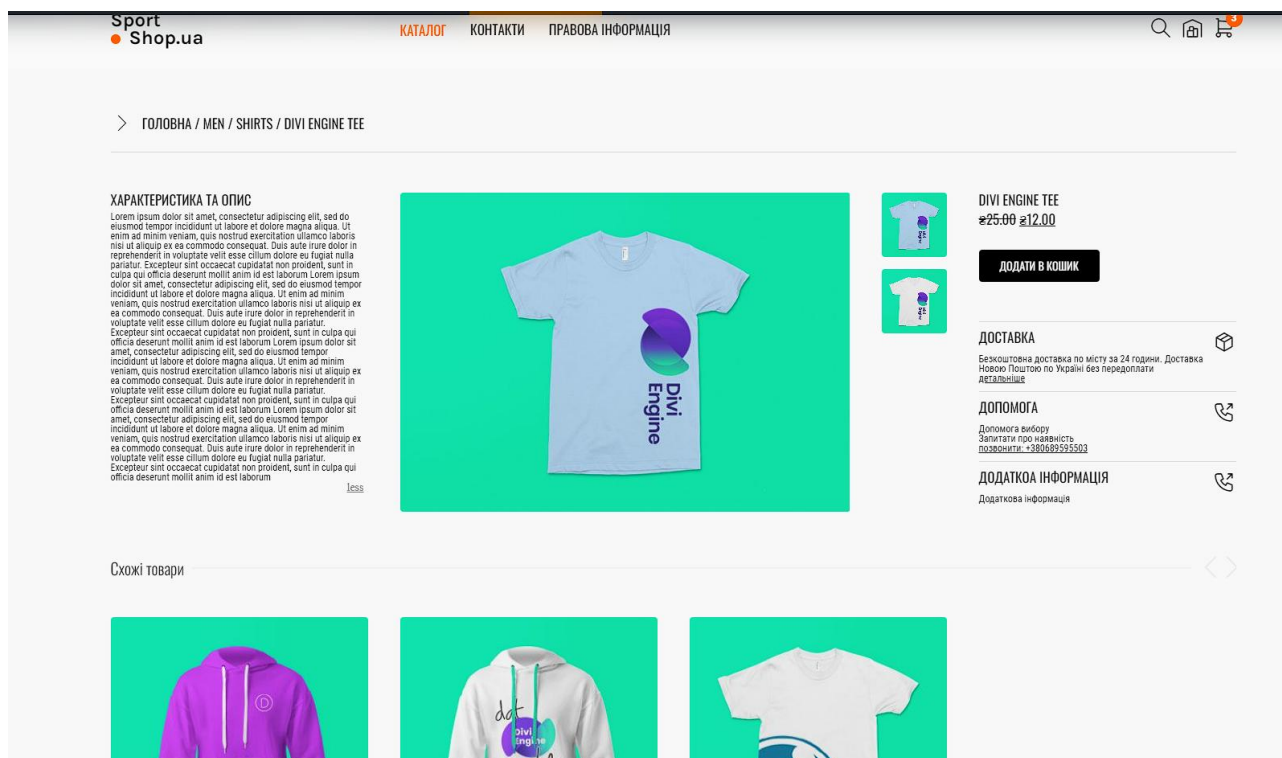


Рисунок 3.3.1 - Загальний вигляд картки товару

Рисунок 3.3.1 зображує кнопку "ДОДАТИ В КОШИК", яка дозволяє додати один товар до кошика. Після натискання на цю кнопку, поряд з іконкою кошика з'являється число, яке відображає кількість елементів, що знаходяться в кошику. Рисунки 3.3.2-3 демонструють цей процес.

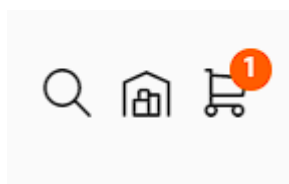


Рисунок 3.3.2 - Вигляд кошика після покупки

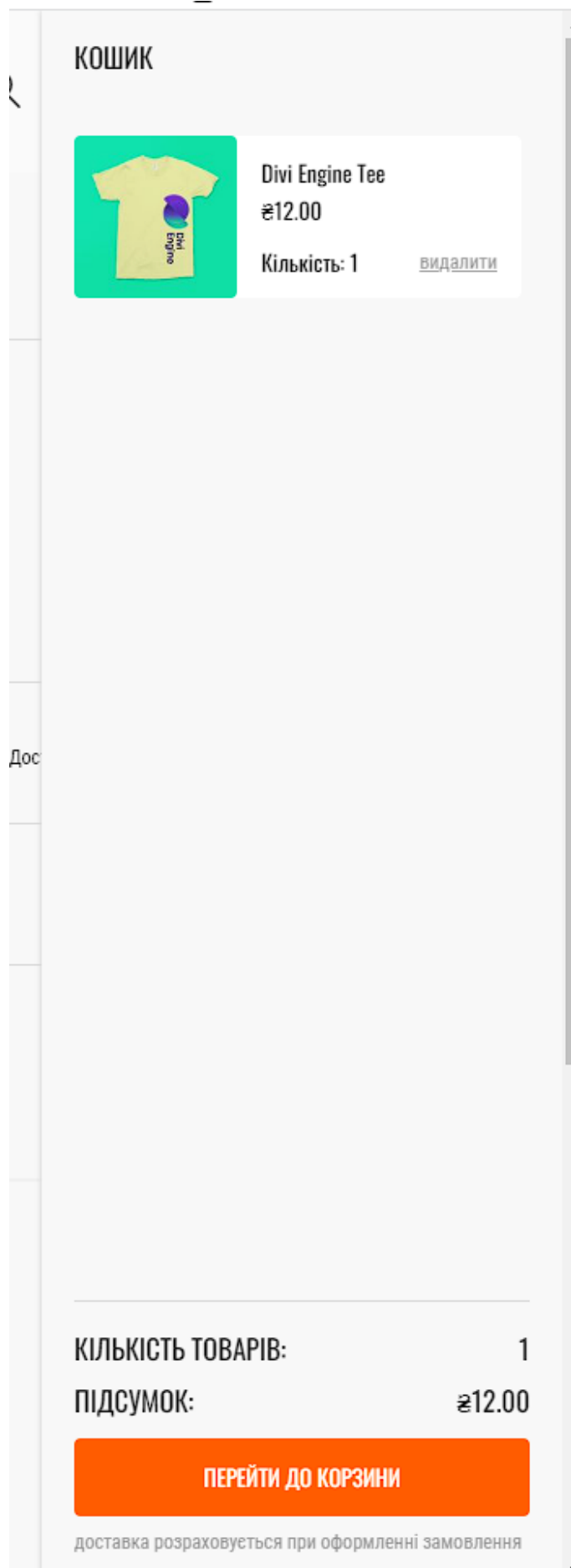


Рисунок 3.3.3 - Вигляд кошику

4. Сторінка відображення категорій товару

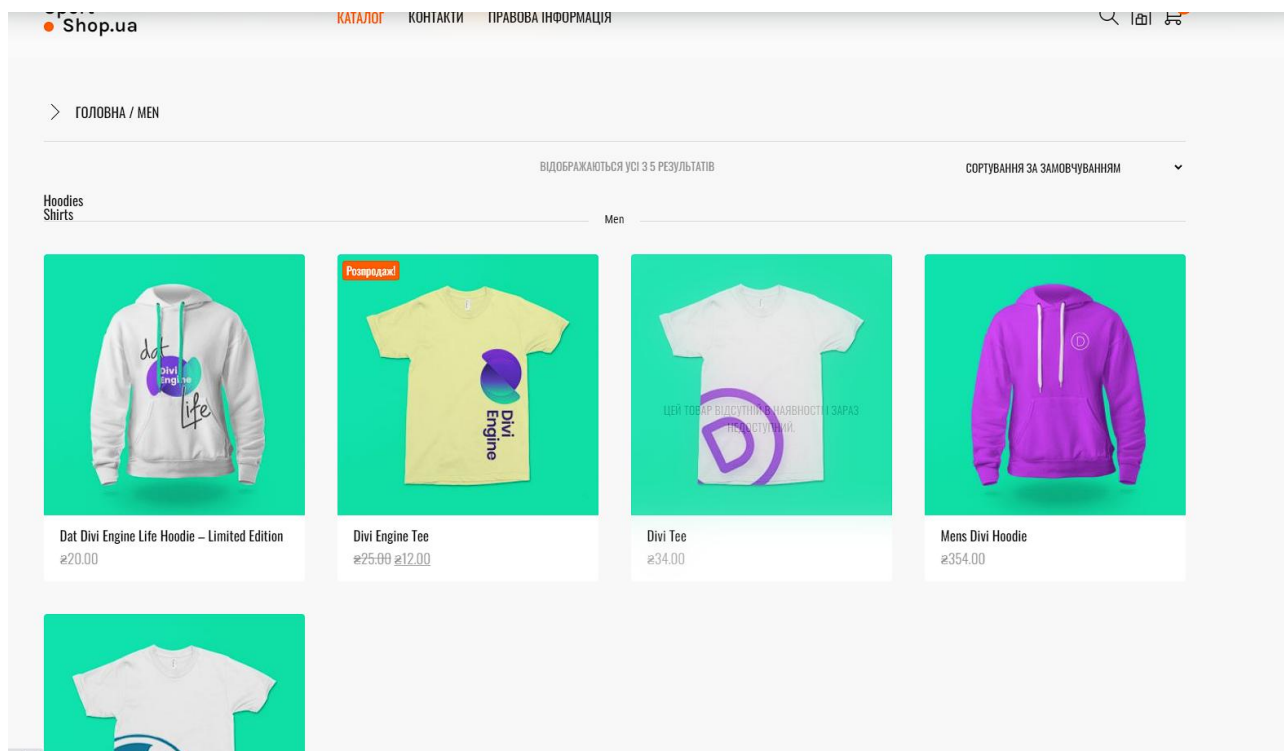


Рисунок 3.4.1 - Вигляд сторінки категорії товару

Задля поліпшення орієнтованості користувача в наявних на сайті товарах необхідно зробити ділення товарів на категорії Рисунок 3.4.1. Завдяки плагіну Woocommerce нам не потрібно власноруч прописувати відображення по категоріям Рисунок 3.4.2.

```

$categories = get_categories($args);

if ($categories) {
    echo '<div class="catalogMain">
        <div class="catalogMain__container container">
            <div class="catalogMain__body">';

    echo '<div class="catalogMain__recommend-header">
        <div class="catalogMain-header">
            <div class="recommend-header__title title">Каталог</div>
            <div class="recommend-header__line"></div>';

    echo '</div>
</div>';

    echo '
        <div class="content">
            <div class="catalogMain__slider">
                <div class="catalogMain__wrapper">';

    foreach ($categories as $category) {
        $category_link = get_term_link($category);
        $image_cat = wp_get_attachment_image_src(get_term_meta($category->term_id, 'thumbnail_id', true), 'full');
        echo '<div class="catalogMain__slide"><div data-prodid="1" class="product-card"> <div class="product-card__wrapper">';
        echo '<a href="' . esc_url($category_link) . '" class="product-card__image IImg"></a>';
        echo '<a href="' . esc_url($category_link) . '" class="product-card__title subtitle_hover"> . $category->name . '</a>';
        echo '</div></div></div>';
    }
    echo '</div></div></div>';

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

```

Рисунок 3.4.2 - вигляд коду сторінки відображення категорій

Також було реалізовано зміна відображення товару якщо його немає в наявності на сайті. Зміну можна побачити при наведенні курсора миші на картку товару Рисунок 3.4.3.

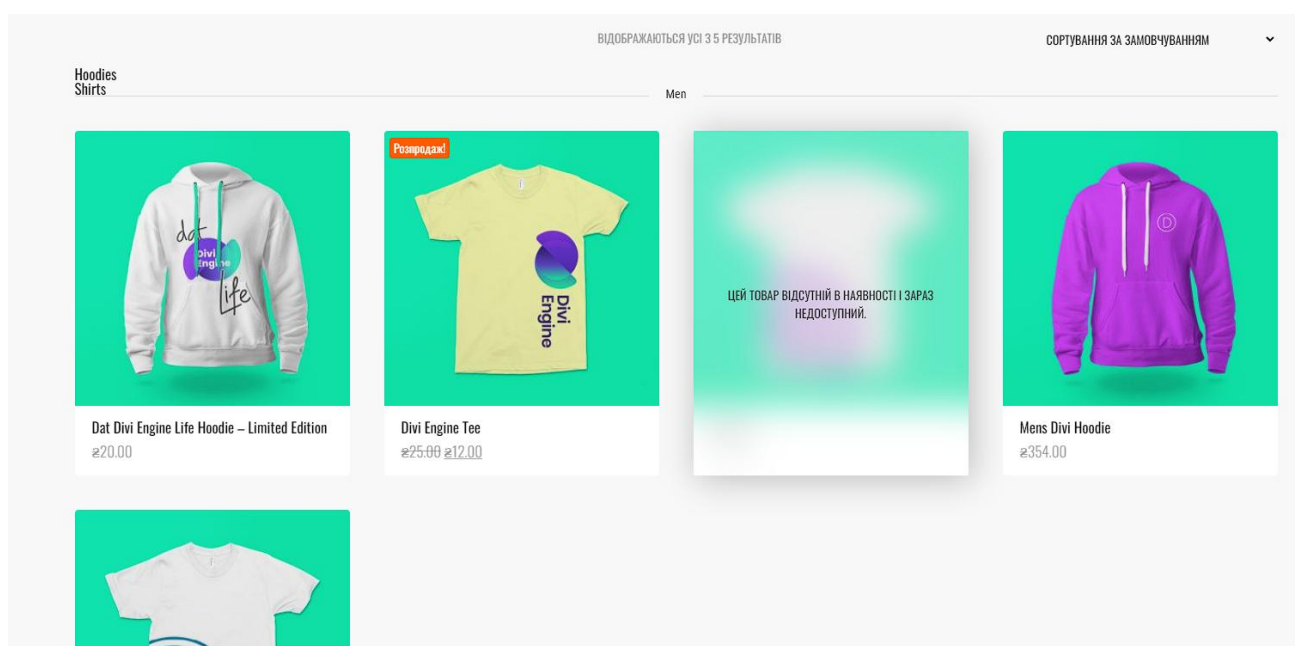


Рисунок 3.4.3 – вигляд сторінки переліку товарів

Було також реалізовано сортування товарів Рисунок 3.4.4

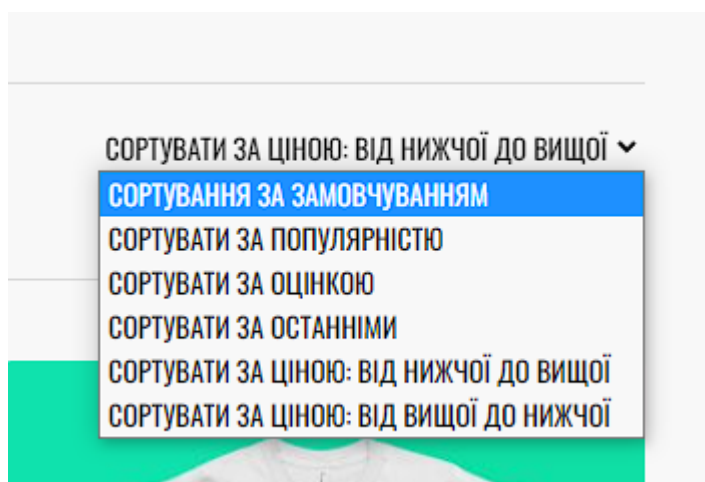


Рисунок 3.4.4 – вигляд сортувального елемента
 При перевірці роботоздатності сортувального метода було виявлено його роботоздатність Рисунок 3.4.5.

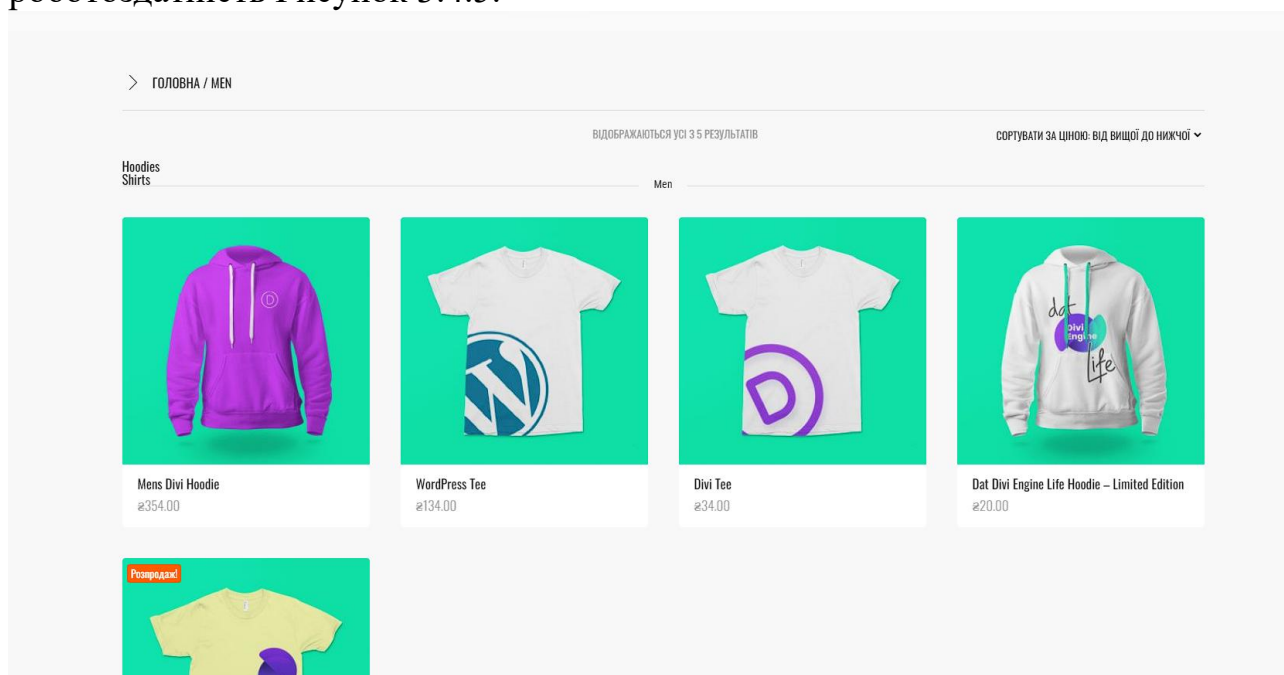


Рисунок 3.4.5 – Сортування товарів за ціною

5. Редагування товару

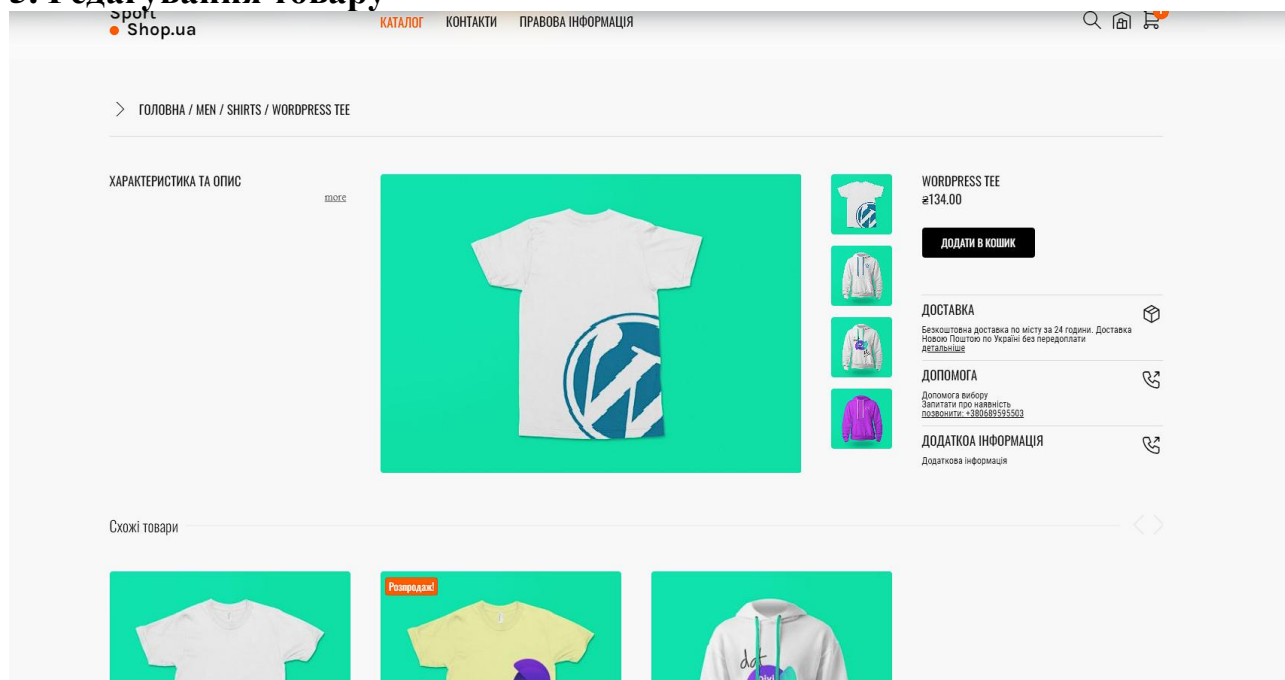


Рисунок 3.5.1 - Вигляд картки товару

Перевагою наявності картки товару Рисунок 3.5.1, на сайті є те що її наявність унеможливорює покупцям потребу відвідувати фізичний магазин або зв'язуватися з продавцем для отримання інформації. Вони можуть переглянути сторінку товару в будь-який зручний для них час та місце, використовуючи свій комп'ютер або мобільний пристрій. Це збільшує зручність та доступність процесу пошуку та покупки товару.

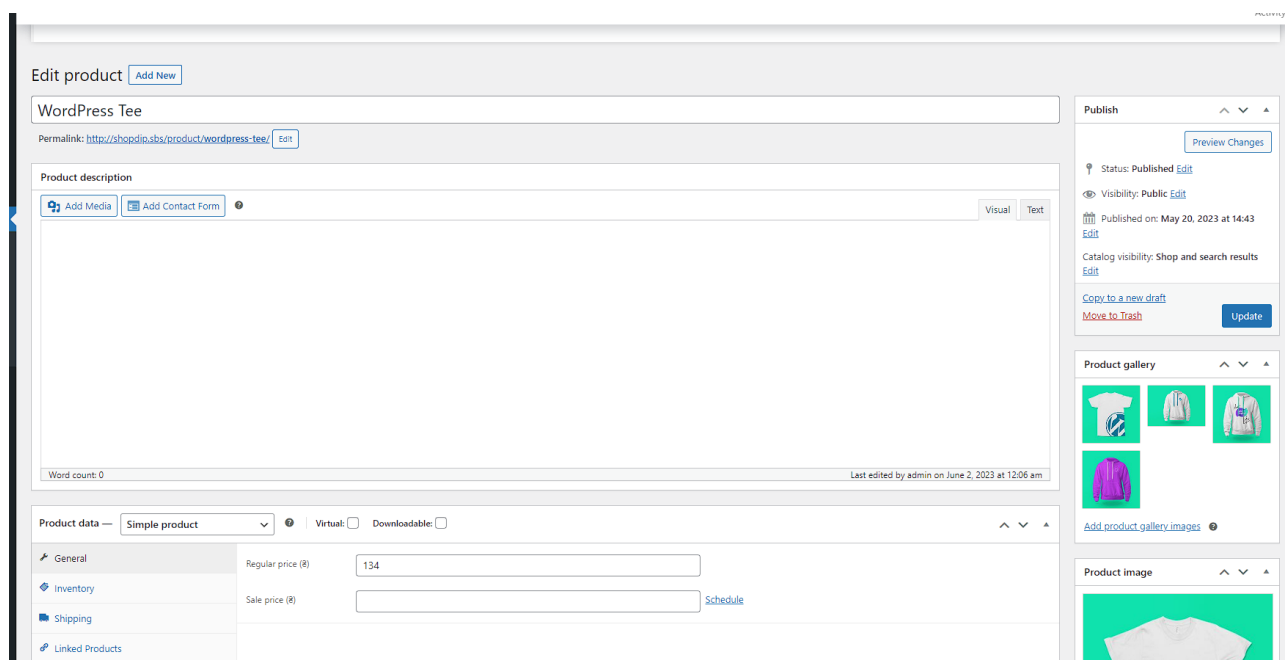


Рисунок 3.5.2 - Вигляд картки товару з панелі адміністратор

У адмін-панелі WordPress зазвичай для управління картками товарів використовуються плагіни або розширення для електронної комерції, наприклад, WooCommerce (Рисунок 3.5.2). Керування картками товарів в адмін-панелі WordPress має декілька переваг. По-перше, воно надає простоту використання завдяки інтуїтивному та користувачу-дружньому інтерфейсу. Користувач може легко створювати, редагувати та оновлювати інформацію про товари без необхідності в програмуванні або складних технічних навичках. Крім того, керування картками товарів дозволяє повністю контролювати деталі товару, такі як опис, характеристики, ціна, наявність і т. д. Користувач також може легко додавати фотографії, відео та відгуки, що допомагає покупцям отримати повну інформацію про товар перед покупкою. Загалом, це зручний і ефективний спосіб управління товарами в вашому інтернет-магазині, що допоможе привернути увагу покупців і збільшити продажі.

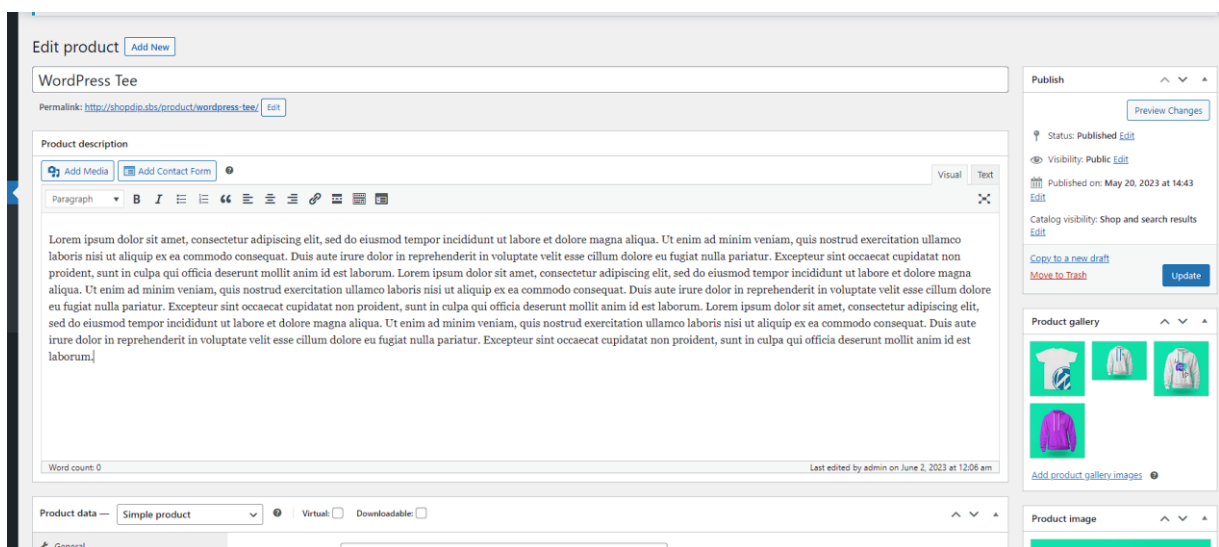


Рисунок 3.5.3 – Додавання опису до товару
Адмін панель дозволяє легко та швидко змінювати інформацію про товар чим і треба скористуватися аби це перевірити Рисунок 3.5.3.

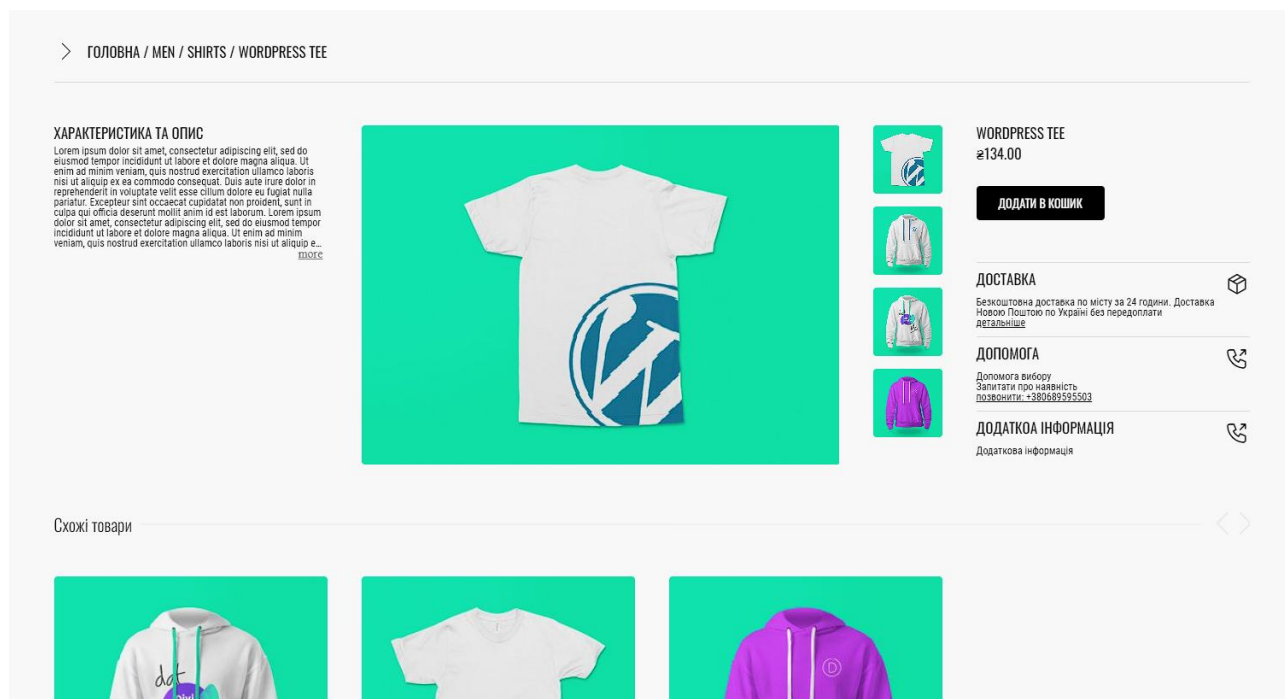


Рисунок 3.5.4 – Оновлена картка товару

Перейшовши назад на сторінку цього товару то маємо оновлений вигляд сторінки товару Рисунок 3.5.4.

ВИСНОВКИ

У межах даної дипломної роботи був розроблений інтернет-магазин, який готовий до використання. Використовуючи технології CMS, WordPress, HTML, CSS та JavaScript, був створений функціональний та естетично привабливий додаток, який може бути налаштований під вимоги власника та має дизайн, придатний для користувачів.

У розділі "АНАЛІЗ ЗАДАЧІ" було визначено об'єкт дослідження (інтернет-магазин) та сформульовано мету, завдання та методи дослідження, які були використані для реалізації додатку. Розділ "АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНИХ СИСТЕМ" надав загальний огляд використаних технологій, їх особливостей та можливостей.

Дослідження цільової аудиторії визначило, що додаток призначений для широкого спектру користувачів різних вікових груп з різними музичними уподобаннями та без географічних обмежень. Це дало змогу побудувати додаток, який надає доступ до музики різних смаків та вподобань.

У розділі "ПОБУДОВА ДОДАТКУ" було придбано хостинг та встановлено FTP-з'єднання з ним, налаштовано середовище розробки та створено резервну копію. Крім того, був реалізований веб-додаток, використовуючи необхідні технології, такі як HTML, CSS та JavaScript, що забезпечують ефективну взаємодію з користувачем та забезпечують зручний інтерфейс.

Були проаналізовані існуючі рішення щодо розробки інтернет-магазину, виявлено їх недоліки та можливості для покращення. Глибша розробка даного додатку може бути корисною для малих та середніх підприємств, які ще не мають власного інтернет-ресурсу для продажу товарів та послуг.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

<https://woocommerce.com/>

(електронний ресурс)

<https://www.php.net/docs.php>

(електронний ресурс)

<https://github.com/artikus11/E-Store-Theme/tree/master> - посилання на репозиторій до курсу по Woocommerce

(електронний ресурс)

<https://uk.wikipedia.org/wiki/HTTP>

(електронний ресурс)

<https://uk.wikipedia.org/wiki/HTTPS>

(електронний ресурс)

<http://shopdip.sbs/>

(адреса додатку)