

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет водного господарства та природокористування
Навчально-науковий інститут кібернетики, інформаційних
технологій та інженерії
Кафедра комп'ютерних технологій та економічної кібернетики

Допущено до захисту:
Завідувач кафедри
комп'ютерних технологій та
економічної кібернетики
д. е. н., проф. П. М. Грицюк

«____» _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬО-КВАЛІФІКАЦІЙНОГО РІВНЯ «БАКАЛАВР»
«Використання систем управління контентом як засобу створення
кроссплатформеного Інтернет-магазину»

Виконав:
здобувач вищої освіти за ОПП
«Інформаційні системи та технології»
спеціальності 126 «Інформаційні системи
та технології», групи ІСТ-41
Корж Денис Русланович

Керівник:
к.е.н., доцент Волошин В. С.

Рецензент:
к.т.н., доцент Василів В. Б.

РЕФЕРАТ

Кваліфікаційна робота бакалавра складається з: 65 с., 24 рис., 5 табл., 12 літературних джерел.

Актуальність теми даного дипломного проєкту зумовлена стрімким розвитком сфери електронної комерції, широким впровадженням цифрових технологій у торгівельну діяльність, а також необхідністю забезпечення підприємств сучасними засобами для онлайн-продажу продукції. Зважаючи на зміну споживацьких звичок та зростання попиту на дистанційну покупку товарів, особливо важливим стає створення ефективних, зручних і функціональних інтернет-магазинів. Серед технологій, що дозволяють реалізувати такі проєкти, особливе місце посідають CMS-системи, зокрема WordPress у поєднанні з плагіном WooCommerce, які забезпечують гнучкість, масштабованість та швидкість розгортання проєктів.

Об'єктом дослідження є інтернет-торгівля, а предметною областю – розробка Web-сайту інтернет-магазину з використанням CMS WordPress для торговельного підприємства, що реалізує універсальний асортимент товарів.

Метою дослідження є проєктування та створення функціонального інтернет-магазину на платформі WordPress із використанням WooCommerce, що охоплює аналіз вимог предметної області, формування логічної структури сайту, вибір засобів реалізації, побудову логічної моделі даних, впровадження базових функціональних модулів та оформлення кінцевого продукту.

У межах бакалаврської роботи реалізовано сайт інтернет-магазину, що включає систему управління товарами, реєстрацію користувачів, каталогізацію продукції, функціонал оформлення замовлень, інтеграцію платіжної системи та адаптивний дизайн. У процесі реалізації особливу увагу приділено структурі бази даних, безпеці користувацьких даних, масштабованості та зручності інтерфейсу як для користувачів, так і для адміністратора сайту.

Результати роботи підтверджують доцільність застосування CMS WordPress для побудови повноцінних онлайн-магазинів у сегменті малого та середнього бізнесу.

КЛЮЧОВІ СЛОВА: ІНТЕРНЕТ-МАГАЗИН, CMS, WORDPRESS, WOOCOMMERCE, ЕЛЕКТРОННА КОМЕРЦІЯ, WEB-САЙТ.

ЗМІСТ

ВСТУП	5
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ФУНКЦІОНУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ	7
1.1. Особливості кросплатформного програмування	7
1.2. Роль систем управління контентом у створенні Інтернет-магазинів.	12
1.3. Особливості забезпечення безпеки в CMS для Інтернет-магазину	24
РОЗДІЛ 2. WEB-САЙТ ІНТЕРНЕТ-МАГАЗИНУ ЯК ЕФЕКТИВНИЙ ЗАСІБ ОНЛАЙН ТОРГІВЛІ	36
2.1. Логічна модель даних	36
2.2. Розробка Інтернет-магазину типового торгівельного підприємства	38
ВИСНОВКИ	55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	57
Додатки	59

ВСТУП

Бурхливий розвиток мережі Інтернет, активне впровадження новітніх інформаційних технологій та цифрових комунікацій кардинально змінили підходи до ведення підприємницької діяльності у XXI столітті. На сучасному етапі використання онлайн-технологій стало невіддільною частиною повсякденного життя, господарської практики та бізнес-моделей, що зумовило формування нових складових електронної економіки – зокрема, електронної комерції.

Електронна комерція (e-commerce) охоплює широкий спектр економічних процесів, що здійснюються за допомогою цифрових каналів, у тому числі через веб-сайти, мобільні застосунки та платформи електронної взаємодії. Основними напрямками цієї діяльності є: організація електронних торгових майданчиків (інтернет-магазинів), проведення маркетингових кампаній у цифровому середовищі, забезпечення онлайн-платежів, а також реалізація логістичних і комунікаційних рішень у цифровій формі.

Сьогодні значна частина підприємств – як вузькоспеціалізованих, так і масштабних торгових мереж – прагнуть мати своє Web-представництво. У багатьох випадках бізнес орієнтується виключно на електронну форму торгівлі. Інтернет відкриває нові канали збуту продукції, забезпечує прямий контакт із клієнтом, а також надає можливості для проведення маркетингових досліджень і ефективної реклами.

Актуальність теми дипломної роботи зумовлена кількома чинниками:

- зростанням обсягів онлайн-продажів у всіх галузях економіки;
- потребою у сучасних цифрових платформах для ведення бізнесу;
- наявністю відкритих та гнучких інструментів для створення інтернет-магазинів;
- прагненням підприємств оптимізувати витрати, автоматизувати процеси замовлень, розширити охоплення ринку.

У сучасних умовах без використання ефективного програмного забезпечення електронної комерції неможливо досягти конкурентоспроможності. Існуючі технології, такі як PHP, JavaScript, JSP, ASP.NET, дозволяють створювати стабільні, безпечні та масштабовані Web-рішення. Сучасні веб-платформи орієнтовані як на зручність для адміністратора магазину, так і на інтуїтивність для кінцевого користувача – покупця.

Метою даної дипломної роботи є створення інтернет-магазину на базі CMS WordPress, який виступатиме повноцінним Web-представництвом торгівельного підприємства, орієнтованого на онлайн-продаж продукції. В основі реалізації покладено використання WooCommerce – плагіна, що надає розширений функціонал для електронної комерції.

Для досягнення поставленої мети у роботі визначено такі завдання:

- аналіз предметної області та ринку електронної торгівлі;
- формулювання функціональних і нефункціональних вимог до програмного забезпечення;
- вибір відповідного інструментарію розробки та технологічної платформи;
- проектування структури сайту та логічної моделі даних;
- реалізація функціональних компонентів інтернет-магазину на базі WordPress і WooCommerce.

У межах дипломного проєкту створено веб-сайт, який дозволяє підприємству публікувати інформацію про товари, приймати замовлення клієнтів, здійснювати обробку замовлень через адміністративну панель, а покупцям – ознайомлюватися з асортиментом продукції та оформлювати покупки безпосередньо через інтерфейс сайту.

Таким чином, реалізований інтернет-магазин забезпечує інтеграцію сучасних технологій у діяльність підприємства, сприяє розширенню цільової аудиторії та підвищенню ефективності комерційної взаємодії у цифровому середовищі.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ФУНКЦІОНУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ

1.1. Особливості кросплатформного програмування

Кросплатформність (або міжплатформність) – це здатність програмного забезпечення функціонувати на різних платформах чи операційних системах. У контексті веб-розробки практично всі веб-додатки вважаються кросплатформними, оскільки вони доступні через будь-який веб-браузер на різних ОС [1].

Іншими словами, інтернет-магазин або будь-який сайт, створений за допомогою стандартних веб-технологій, може однаково відображатися на настільних комп'ютерах, ноутбуках, планшетах чи смартфонах. Веб-технології (HTML, CSS, JavaScript) є стандартизованими, що дозволяє браузерам на різних пристроях інтерпретувати їх схожим чином. Проте забезпечення повної кросплатформної сумісності вимагає врахування особливостей різних браузерів і пристроїв, зокрема різних розмірів екранів та середовищ виконання.

Сучасні стандарти HTML5 і CSS3 відіграють ключову роль у кросплатформній підтримці веб-застосунків. HTML5 надає семантичні елементи та вбудовані засоби для медіаконтенту (відео, аудіо тощо) без потреби у сторонніх плагінах, що підвищує доступність сайту на різних пристроях (наприклад, відмова від Flash-технологій, не підтримуваних на мобільних платформах). CSS3, зі свого боку, вводить медіа-запити – механізм, що дозволяє застосовувати різні стилі залежно від характеристик пристрою (тип носія, роздільна здатність екрану, орієнтація тощо) [2].

Завдяки медіа-запитам дизайн сторінки може “підлаштовуватися” під ширину екрана: наприклад, багатостовпковий макет на великому моніторі може трансформуватися в одностовпковий на смартфоні. JavaScript доповнює HTML5/CSS3, забезпечуючи інтерактивність і динамічне реагування інтерфейсу на різних платформах. Сучасні JS-фреймворки (React, Angular, Vue тощо) та бібліотеки (наприклад, jQuery) враховують відмінності браузерів і

надають єдиний інтерфейс для розробника, щоб функціональність інтернет-магазину працювала узгоджено у різних середовищах. Таким чином, поєднання HTML5, CSS3 та JavaScript утворює універсальний стек технологій, який лежить в основі кросплатформного веб-програмування.

Системи управління контентом (CMS) значно спрощують створення кросплатформних інтернет-магазинів. Зокрема, WordPress – одна з найпопулярніших у світі CMS (займає понад 40% ринку) і широко використовується для різних типів сайтів, включно з інтернет-магазинами (через плагін WooCommerce) [3].

CMS типово розділяють контент і презентацію: дані про товари, тексти та зображення зберігаються у базі даних, а відображення керується шаблонами та стилями. Це означає, що один і той самий контент автоматично подається у форматі, придатному для будь-якого пристрою, без дублювання інформації для кожної платформи. WordPress та інші сучасні CMS підтримують адаптивний веб-дизайн (responsive design) через готові теми оформлення. Більшість сучасних тем WordPress є адаптивними: вони здатні динамічно змінювати макет під різні розміри екрану, забезпечуючи коректне відображення на мобільних і настільних пристроях [4].

Наприклад, офіційні теми WordPress та популярні комерційні шаблони (Divi, Avada, Astra тощо) спроектовані як мобільно-дружні (mobile-friendly) і проходять перевірку на різних пристроях. Крім того, екосистема WordPress надає спеціальні плагіни для підвищення мобільної зручності. Зокрема, розширення на кшталт WPTouch чи WP Mobile Menu можуть додавати мобільно-оптимізовані меню та інші елементи інтерфейсу, щоб покращити досвід користувачів на смартфонах [5].

Адаптивний (responsive) веб-дизайн є основним підходом, що дозволяє одному сайту ефективно працювати на множині пристроїв. Ідея полягає в створенні гнучкого макету, який автоматично перебудовується під роздільну здатність та орієнтацію екрану користувача. Для реалізації цього підходу використовуються згадані медіа-запити CSS3, гнучкі сітки (fluid grids) та

масштабовані зображення. Наприклад, замість фіксованої ширини в пікселях елементи верстаються у відсотках від ширини екрану, а зображення задаються з максимальною шириною 100% (щоб не виходити за межі контейнера). При зменшенні вікна браузера колонки можуть автоматично шикуватися одна під одною, навігаційне меню перетворюється на випадаюче “гамбургер-меню”, а нетекстові елементи (банери, таблиці) адаптуються для вертикального прокручування. У термінах програмування, адаптивність – це властивість інтерфейсу пристосовуватися без створення окремих версій сайту для кожного типу пристрою. Такий підхід кардинально відрізняється від старішої практики підтримувати окремо «мобільну версію» сайту (з окремим URL або субдоменом) або розробляти рідні додатки для кожної платформи. Адаптивний дизайн мінімізує потребу дублювання: одна кодова база обслуговує всіх користувачів, що значно спрощує підтримку і оновлення інтернет-магазину.

Як видно з рисунка 1.1, адаптивний інтерфейс дозволяє зберегти впізнаваний дизайн інтернет-магазину незалежно від пристрою, яким користується відвідувач. Це досягається за рахунок того, що фронтенд веб-сайту (тобто його клієнтська частина) автоматично підлаштовує розміри шрифтів, розташування блоків, зображення та навігацію під дані середовища. JavaScript може додатково використовуватися для виявлення характеристик платформи (через responsive бібліотеки або viewport API) і внесення дрібних динамічних змін, як-от активація сенсорних ефектів на екранах з підтримкою дотику або відкладене завантаження зображень для повільніших мобільних мереж. Однак базова підтримка кросплатформності у веб-додатках досягається саме за рахунок грамотно розробленого адаптивного дизайну на основі HTML5/CSS3. У підсумку, користувач отримує позитивний досвід: інтерфейс магазину лишається зручним та інтуїтивним і на 5-дюймовому телефоні, і на великому десктопному моніторі.



Рис. 1.1 – Схематична ілюстрація адаптивного веб-дизайну інтернет-магазину на різних пристроях.

У сучасному електронному бізнесі кросплатформність веб-сайту є критичною вимогою. Стрімке зростання мобільної аудиторії призвело до того, що понад половину глобальних продажів e-commerce зараз припадає на мобільні пристрої.

Якщо онлайн-магазин не оптимізований для смартфонів, він ризикує втратити значну частку потенційних клієнтів, оскільки користувачі просто підуть до конкурентів із зручнішою мобільною версією. Навпаки, забезпечення єдиного якісного досвіду на різних девайсах підвищує лояльність аудиторії: покупцеві зручно переглядати каталог на телефоні, додавати товари до кошика, а пізніше завершити замовлення на ноутбучі – при цьому інтерфейс та функціональність залишаються знайомими і послідовними. Така узгодженість інтерфейсу зміцнює довіру до бренду та збільшує ймовірність повторних покупок. SEO-фактор теж відіграє роль: пошукові системи (Google та ін.) віддають перевагу сайтам з мобільно-дружнім дизайном у результатах пошуку [6].

Google, зокрема, перейшов на індексування сайтів за принципом mobile-first, що означає використання мобільної версії контенту як основної при ранжуванні. Отже, адаптивний інтернет-магазин не лише безпосередньо

приваблює мобільних користувачів, а й опосередковано сприяє залученню трафіку через вищі позиції у пошуку. З точки зору витрат, кросплатформний підхід також вигідніший: підтримка єдиного сайту для всіх платформ обходиться дешевше і простіше, ніж розробка та оновлення окремих додатків чи версій сайту для різних пристроїв [7].

Таблиця 1.1

Переваги кросплатформності для веб-ресурсу електронної комерції

Перевага для e-business	Сутність та обґрунтування
Ширше охоплення аудиторії	Один сайт охоплює користувачів різних пристроїв і ОС, включно з мобільними – жоден сегмент потенційних клієнтів не залишається поза увагою.
Покращений досвід користувачів	Адаптивний інтерфейс забезпечує зручність і консистентність взаємодії. Користувачі легше знаходять та купують товари, що підвищує їхню задоволеність та лояльність.
Підвищення видимості та SEO	Мобільно-дружні сайти отримують вищий рейтинг у пошукових системах . Крім того, швидкість завантаження та низький показник відмов покращують SEO-показники.
Ефективність розробки та підтримки	Одна уніфікована версія сайту простіша в розробці й оновленні, ніж кілька окремих. Це знижує витрати на розробку і технічну підтримку, прискорює внесення змін.
Узгодженість бренду	Єдиний дизайн на всіх платформах підтримує цілісність іміджу бренду. Клієнти впізнають стиль магазину на будь-якому пристрої, що підсилює довіру та впізнаваність.

Отже, кросплатформне програмування в контексті веб-розробки та CMS – це не лише технічна вимога, але й стратегічна перевага для бізнесу. Використання систем управління контентом (таких як WordPress) у поєднанні з

адаптивним дизайном, стандартами HTML5/CSS3 і можливостями JavaScript дозволяє створити інтернет-магазин, доступний максимально широкій аудиторії. В результаті сучасний електронний магазин здатен забезпечити присутність на всіх популярних платформах (Windows, macOS, Android, iOS тощо) через єдиний веб-інтерфейс. Це підвищує конкурентоспроможність ресурсу, оскільки користувачі все частіше очікують безперебійного сервісу «будь-де та будь-коли». Кросплатформність стала невід'ємною характеристикою успішних веб-проектів, особливо в сфері електронної комерції, де боротьба йде за кожного клієнта на кожному пристрої.

1.2. Роль систем управління контентом у створенні Інтернет-магазинів.

Системи управління контентом (CMS) відіграють ключову роль у сучасній електронній комерції, забезпечуючи основу для швидкого створення та ефективного керування інтернет-магазинами. Завдяки CMS розробники та власники бізнесу можуть зосередитися на наповненні та функціональності магазину замість написання коду з нуля. У основі будь-якої CMS лежить спрощення процесів створення і редагування контенту, що стосується не лише текстової інформації, але й мультимедійних даних. Зокрема, власники інтернет-магазинів отримують можливість легко оновлювати описи товарів, додавати зображення чи змінювати ціни без необхідності залучення програмістів [8].

Це означає, що навіть користувач із мінімальними технічними знаннями здатний підтримувати актуальність каталогу товарів і контенту на сайті. CMS також дозволяють гнучко керувати структурою веб-ресурсу – створювати нові сторінки, додавати категорії товарів, змінювати меню навігації тощо, забезпечуючи зручну структуру й навігацію магазину для покупців. Така організація позитивно впливає на досвід користувачів, полегшуючи їм пошук потрібних товарів і інформації на сайті.

Важливою перевагою CMS для інтернет-магазинів є вбудовані засоби забезпечення безпеки та адміністрування. Сучасні платформи надають можливості розмежування прав доступу і керування користувачами (ролями). Адміністратор магазину може через панель CMS визначати, які дії дозволені менеджерам, редакторам контенту чи іншим групам користувачів, що підвищує загальну безпеку системи. Крім того, розробники CMS регулярно випускають оновлення, які усувають виявлені вразливості і покращують стабільність роботи – це особливо критично для інтернет-магазинів, де захист даних клієнтів і безпечність транзакцій мають найвищий пріоритет

Ще одним аспектом, у якому CMS відіграють значну роль, є забезпечення багатомовності та кросплатформеності. Більшість популярних платформ підтримують мультимовний контент або через вбудовані можливості, або через додаткові модулі. Це дає змогу інтернет-магазину працювати на різних ринках і мовах, розширюючи аудиторію і підвищуючи залученість користувачів. Багатомовність, у свою чергу, позитивно впливає й на SEO – пошукові системи краще індексують сайт, доступний кількома мовами, що підвищує його видимість у пошуку

Наявність адаптивних шаблонів дизайну в рамках CMS забезпечує коректне відображення інтернет-магазину на різних пристроях (ПК, планшет, смартфон), тобто реалізує кросплатформеність і зручність користування незалежно від платформи користувача.

Однією з найвагоміших причин популярності CMS у сфері електронної комерції є можливість швидкого розгортання функціонального магазину із мінімальними витратами. Використовуючи готові шаблони та плагіни, компанії можуть запустити веб-сайт магазину значно швидше і з меншими ресурсами, ніж при розробці з нуля.

Така ефективність особливо вигідна для малого і середнього бізнесу, який може не мати бюджету на тривалу індивідуальну розробку. Різноманітні плагіни розширюють базові можливості платформи: від SEO-оптимізації, інтеграції з платіжними шлюзами та службами доставки до додавання нових

маркетингових інструментів (програми лояльності, віджети відгуків, інтеграції з соцмережами тощо). Аналізуючи популярні CMS, можна помітити, що кожна з них має великий набір доступних розширень. Користувачі можуть вибирати з широкого спектру плагінів все необхідне для оптимізації та покращення роботи магазину.

Застосування таких модулів робить CMS-платформи надзвичайно гнучкими та масштабованими для бізнесу, що прагне розвитку. Масштабованість означає, що інтернет-магазин на базі CMS може рости разом із бізнесом: збільшувати каталог товарів, витримувати зростання трафіку, інтегрувати нові функції по мірі потреби без повної перебудови системи. Отже, сучасні CMS стали незамінним інструментом для створення і підтримки інтернет-магазинів – їхні можливості управління контентом, структурою сайту, інтеграції зі сторонніми сервісами, забезпечення безпеки, багатомовності та масштабування роблять їх надзвичайно привабливими для бізнесу.

Серед різних CMS особливо помітною є зв'язка WordPress + WooCommerce – одна з найпопулярніших платформ для створення інтернет-магазинів на сьогодні. WordPress, як найбільш поширена у світі система управління контентом, в поєднанні з плагіном WooCommerce фактично перетворює будь-який сайт на WordPress у повноцінний онлайн-магазин. Популярність цього тандему підтверджується статистикою: за даними останніх досліджень, WooCommerce використовується приблизно на п'ятій частині всіх веб-сайтів електронної комерції, поступаючись лише платформі Shopify [9].

Від моменту запуску WooCommerce у 2011 році цей плагін набув стрімкого поширення і вже за кілька років став одним із найпотужніших рішень для інтернет-торгівлі на базі WordPress. Компанія Automattic (розробник WordPress.com) викупила WooCommerce у 2015 році, що ще більше прискорило його розвиток і інтеграцію в екосистему WordPress [10]

Основна перевага WordPress з WooCommerce – це простота встановлення та інтуїтивно зрозумілий інтерфейс адміністратора. Адмін-панель WordPress

відома своєю зручністю, тож навіть новачок може швидко опанувати базові операції керування магазином.

Додавання товарів, керування замовленнями та публікація контенту (наприклад, оглядів, статей блогу для просування) виконується через зрозумілий веб-інтерфейс, що нагадує текстовий редактор. До того ж, WooCommerce повністю інтегрований у WordPress, тому адміністратор працює в єдиній консольній панелі для всього сайту (і контентної частини, і магазину). Це забезпечує єдність стилю керування і дозволяє використовувати всі напрацювання спільноти WordPress – тисячі тем оформлення та плагінів – безперешкодно і в магазині [11].

WooCommerce успадкував від WordPress широкі можливості кастомізації та SEO-оптимізації: доступні численні плагіни для покращення пошукової видимості магазину, генерації метаданих, інтеграції з соціальними мережами і т.д.

Для WooCommerce існує величезна екосистема доповнень: розширення для інтеграції платіжних систем (PayPal, LiqPay, Stripe тощо), модулі для доставки (наприклад, “Нова Пошта” в Україні), плагіни для аналітики продажів, CRM-інтеграції, імпорту товарів із файлів і багато інших. За потреби функціональність магазину на WooCommerce можна масштабувати – від простого сайту з кількома десятками товарів до каталогу на тисячі позицій. Звісно, при зростанні навантаження важливо забезпечити відповідний хостинг та оптимізацію (кешування сторінок, використання CDN тощо) для підтримання швидкодії. Проте сама платформа WooCommerce задовольняє потреби як невеликих інтернет-крамниць, так і достатньо великих майданчиків [12].

Повний життєвий цикл замовлення в інтернет-магазині на WooCommerce починається зі статусу "Pending" (Очікування оплати), який автоматично встановлюється системою при створенні нового замовлення. Далі, залежно від ситуації з оплатою, замовлення може перейти в стан "On-Hold" (На утриманні), якщо потрібна ручна перевірка, або "Failed" (Не вдалося), якщо платіжний

шлюз відхилив транзакцію. У разі успішної оплати, платіжний шлюз або власник магазину встановлюють статус "Processing" (В обробці), що є сигналом до початку виконання замовлення. Після відправки товару, власник вручну змінює статус на "Completed" (Виконано). Крім того, на будь-якому з початкових етапів власник може скасувати замовлення, присвоївши йому статус "Cancelled" (Скасовано), а після успішного виконання — оформити повернення коштів зі статусом "Refunded" (Повернено). Схематично функціонування плагіну Woocommerce можна розглянути на.

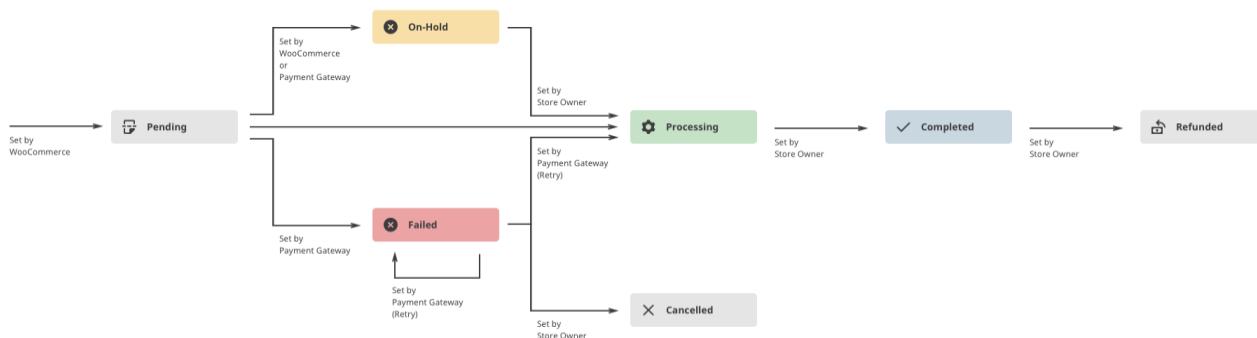


Рисунок 1.1 - Функціонування плагіну Woocommerce

Таким чином, WordPress + WooCommerce привабливий своєю комбінованою силою: WordPress дає потужний рушій для контенту і SEO, а WooCommerce додає всі необхідні інструменти для торгівлі. Ця платформа особливо підходить підприємствам, яким потрібна гнучкість у налаштуванні та контроль над сайтом – від дизайну вітрини до тонкого управління кодом при необхідності. Водночас, слід врахувати, що використання WordPress вимагає відповідального підходу до безпеки: необхідно регулярно оновлювати ядро, плагіни та теми, користуватися перевіреними розширеннями, аби уникати можливих уразливостей (оскільки популярність платформи робить її потенційною мішенню для злоумисників).

Завдяки активній спільноті та регулярним апдейтам, ці ризики можна мінімізувати. У підсумку, WordPress з WooCommerce забезпечує надзвичайно зручний баланс між простотою та розширюваністю, що робить його одним з лідерів для створення кросплатформених інтернет-магазинів.

OpenCart є прикладом спеціалізованої CMS, розробленої саме для потреб інтернет-магазинів. Це відкрите програмне забезпечення (open-source) на PHP, яке можна безкоштовно завантажити і встановити на власний сервер. На відміну від WordPress, що потребує окремого плагіну для торгівлі, OpenCart “з коробки” орієнтований на електронну комерцію і містить усі базові компоненти магазину: каталог товарів, кошик покупця, систему керування замовленнями, облікові записи клієнтів, звітність тощо

Встановлення OpenCart є доволі простим і не вимагає глибоких технічних знань: після розгортання файлів на хостингу та підключення бази даних адміністратор одержує готову панель управління магазином. Інтерфейс адміністратора OpenCart також вважається досить зручним і мінімалістичним – у ньому легко орієнтуватися навіть підприємцю-початківцю. Через панель можна додавати товари (підтримуються різні типи товарів, опції, атрибути), відстежувати запаси, обробляти замовлення, встановлювати методи оплати і доставки тощо. За потреби функціональність OpenCart розширюється за допомогою модулів та розширень. Існує офіційний маркетплейс розширень OpenCart, де представлені тисячі доповнень – від платіжних шлюзів і інтеграцій з поштовими сервісами до додаткових маркетингових інструментів і нових шаблонів дизайну

Хоча кількість доступних плагінів для OpenCart поступається екосистемі WordPress, вона все ж достатньо велика, щоб задовольнити типові потреби магазину. OpenCart підтримує мультимовність і багатовалютність: адміністратор може налаштувати кілька мов інтерфейсу для вітрини та цін у різних валютах, що важливо для міжнародної торгівлі. Система також має вбудовані засоби SEO-оптимізації – дружні URL-адреси для сторінок товарів, можливість задавати мета-теги та опис, генерація sitemap-файлів тощо.

Завдяки цьому магазин на OpenCart може бути добре проіндексований пошуковими системами. З точки зору масштабованості, OpenCart зазвичай рекомендують для малих та середніх магазинів. Він досить легковагий і швидкий, але для дуже великих каталогів (десятки тисяч товарів) або пікових

навантажень може знадобитися додаткова оптимізація і більш потужний сервер. На практиці OpenCart широко використовується у світі: особливо популярний він у тих випадках, коли потрібне просте у впровадженні і недороге рішення для e-commerce. В Україні, за деякими рейтингами, OpenCart є однією з найбільш уживаних CMS для інтернет-магазинів наряду з WooCommerce

Підсумовуючи, OpenCart приваблює користувачів поєднанням простоти і необхідного функціоналу: це безкоштовна платформа з базовим набором можливостей магазину, що не перевантажена зайвим і може бути розгорнута дуже швидко. Для підприємців, які не потребують складних інтеграцій чи унікальних рішень, OpenCart є практичним вибором, забезпечуючи основи онлайн-торгівлі «під ключ».

Joomla – це ще одна потужна відкрита CMS загального призначення, яка може бути використана для створення інтернет-магазину за допомогою спеціальних розширень. Joomla існує з 2005 року і свого часу була однією з найпопулярніших CMS поряд з WordPress. Вона відома гнучкістю, широкими можливостями налаштування та розвинутою спільнотою розробників. Базова Joomla призначена для побудови сайтів різної складності – від блогів і корпоративних порталів до онлайн-спільнот. Для додавання функціоналу інтернет-магазину в Joomla використовуються компоненти (розширення), найвідомішим з яких є VirtueMart.

VirtueMart – це безкоштовний плагін електронної комерції, який інтегрується в Joomla і надає типові можливості магазину: управління товарами, категоріями, кошиком, замовленнями, платежами тощо. Історично VirtueMart був одним із перших e-commerce-рішень для Joomla і здобув велику аудиторію завдяки багатофункціональності та стабільності роботи. За допомогою VirtueMart або аналогічних розширень (наприклад, HikaShop, J2Store) на Joomla можна розгорнути повноцінний магазин, інтегрований у контентний сайт.

Особливістю Joomla є дещо вищий поріг входження для адміністратора в порівнянні з WordPress. Інтерфейс панелі керування Joomla достатньо інтуїтивний, але має більше налаштувань і опцій, що може вимагати часу на освоєння. Водночас, ця система пропонує багатий набір вбудованих функцій, корисних для інтернет-магазину: підтримка багатомовності «з коробки» (не потрібні додаткові плагіни для створення багатомовного сайту), гнучка система керування користувачами і правами доступу, а також потужні засоби кешування і оптимізації продуктивності. У Joomla існує величезна кількість розширень – в офіційному каталозі Joomla Extensions Directory нараховується понад 8000 плагінів і модулів. Це включає і розширення для електронної комерції, і додаткові інструменти: слайдери, галереї, SEO-плагіни, засоби безпеки тощо. Така масштабна екосистема дозволяє допасувати Joomla-проект під будь-які вимоги. Наприклад, для підвищення безпеки можна встановити спеціальні firewall-плагіни, капчу, засоби захисту від DDoS; для маркетингу – модулі розсилки, відгуків, інтеграції з Facebook Pixel тощо

Багато сучасних шаблонів для Joomla є адаптивними і оптимізованими під мобільний трафік, а також пропонують широкі можливості зміни дизайну через вбудовані налаштування. Joomla підходить для проектів, де окрім магазину потрібен багатофункціональний сайт – наприклад, портал з новинами, форумом чи великою кількістю статичного контенту поряд з каталогом товарів. У таких випадках Joomla забезпечить міцну основу, а e-commerce-розширення додадуть торговельний функціонал. Хоча останнім часом частка Joomla на ринку дещо зменшилась, вона все ще використовується і підтримується спільнотою. Таким чином, Joomla з VirtueMart (або аналогом) є гарним вибором для створення інтернет-магазину, що інтегрований з контент-порталом, і надає адміністратору широкі можливості керування сайтом.

На відміну від описаних вище рішень, Shopify представляє категорію хмарних SaaS-платформ (software as a service), які пропонують повністю готове середовище для запуску інтернет-магазину. Shopify не вимагає від користувача ні оренди хостингу, ні самостійного встановлення ПЗ – достатньо

zareєструватися на сервісі, обрати тарифний план і розпочати наповнення магазину через веб-інтерфейс. Останніми роками Shopify посідає провідні позиції у світі за кількістю інтернет-магазинів: його частка оцінюється приблизно у 26% від усіх e-commerce сайтів, що навіть більше, ніж у WooCommerce [9].

Такий успіх пояснюється в першу чергу простотою використання платформи. Для підприємця, який обирає Shopify, процес запуску виглядає дуже швидко: сервіс надає бібліотеку готових шаблонів і блоків, які можна комбінувати за допомогою візуального конструктора, налаштовуючи зовнішній вигляд магазину без програмування

Базові аспекти (кошик, оформлення замовлення, робота з базою клієнтів, аналітика продажів) вже реалізовані “під капотом” – потрібно лише внести необхідні дані про компанію і товари. Інтерфейс адміністратора Shopify славиться своєю зручністю: він сучасний, drag-and-drop редактор дозволяє редагувати сторінки магазину у режимі реального часу, а всі операції (додавання товару, перегляд замовлення, зміна налаштувань) зрозумілі на інтуїтивному рівні навіть для недосвідченого користувача [12].

Великим плюсом є те, що Shopify бере на себе всю технічну сторону – підтримання інфраструктури, оновлення системи, резервне копіювання, інформаційну безпеку. Інтернет-магазин на Shopify розміщується на хмарних серверах самої компанії, тому власнику бізнесу не треба турбуватися про встановлення SSL-сертифікатів, налаштування баз даних чи захист від DDoS-атак: ці питання вирішуються платформою централізовано.

Попри те, що Shopify є закритою комерційною платформою, він забезпечує достатню гнучкість через власний магазин додатків (App Store). На сьогодні для Shopify доступно понад 4000 додатків та інтеграцій, які можна підключити у кілька кліків. Це дозволяє розширити стандартний функціонал магазину – додати нестандартні способи доставки, інтегрувати маркетингові сервіси, кастомізувати вітрину, підключити зовнішні системи обліку тощо. Багато з цих додатків створені сторонніми розробниками і покривають

практично всі потреби (наприклад, існують модулі для інтеграції з Instagram Shop, для запуску програм лояльності, для підтримки онлайн-чату з клієнтами і ін.). Таким чином, за рівнем доступних розширень Shopify наближається до відкритих CMS. Важливо, що Shopify також підтримує багатомовність і продаж у кількох валютах – це реалізовано через вбудовані функції у панелі керування. Адміністратор може додати кілька мов магазину і перемикач валют, що корисно для міжнародних продажів (хоча слід врахувати можливі комісії платіжних систем за конвертацію). Платформа добре масштабується: якщо бізнес зростає, достатньо змінити тарифний план на вищий, і Shopify забезпечить обробку більшого трафіку та замовлень без зниження продуктивності.

Звичайно, використання Shopify має і інший бік – це вартість та контроль. Платформа працює за передплатною моделлю: користувач щомісяця сплачує фіксовану суму згідно обраного тарифу. При невеликому обороті базовий тариф є доступним, але для успішного масштабного магазину витрати можуть стати значними. Окрім абонплати, Shopify стягує комісію з транзакцій (від 0,5% до 2% з кожного продажу залежно від плану), якщо не використовується фірмовий платіжний шлюз Shopify Payments. Також чимало додатків у Shopify App Store є платними або передбачають окрему підписку, що збільшує сукупні витрати на підтримку магазину.

Ще один нюанс – менша гнучкість у порівнянні з open-source CMS. Код сайту, розгорнутого на Shopify, не належить власнику повністю (фактично це оренда програмної платформи). Можливості глибокого втручання в логіку роботи обмежені: користувач може додавати HTML/JavaScript у визначені області, налаштовувати тему через мову Liquid, але не має прямого доступу до серверного коду. Для більшості бізнесів це не є проблемою, адже на практиці надається широкий набір налаштувань, проте для дуже специфічних вимог Shopify може виявитись невідповідним. Незважаючи на ці обмеження, для переважної більшості підприємців переваги (швидкий старт, простота, безпека, підтримка) переважають. Shopify став популярним серед малого і середнього

бізнесу, творчих підприємців, стартапів, оскільки дозволяє запустити якісний інтернет-магазин буквально за кілька днів і одразу продавати товари по всьому світу.

На Рис. 1.1. діаграма демонструє розподіл популярності систем управління контентом (CMS) для створення інтернет-магазинів на українському ринку. Лідером є зв'язка WordPress + WooCommerce, яка займає 30% ринку, що свідчить про її високу популярність завдяки гнучкості та великій екосистемі. На другому місці знаходиться OpenCart з часткою 25%, що підкреслює його затребуваність як спеціалізованого та простого рішення для електронної комерції. Такі платформи, як Magento (15%) та PrestaShop (8%), також займають значні позиції, орієнтуючись на середній та великий бізнес.

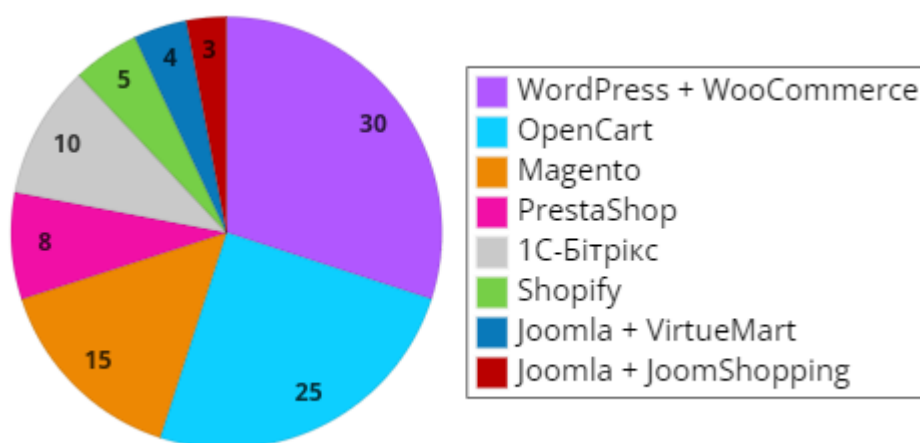


Рисунок 1.2 - Популярність CMS для розробки інтернет-магазинів в Україні

Наведені вище платформи – не вичерпний перелік рішень для інтернет-магазинів. Існують також інші CMS та фреймворки, які широко застосовуються в e-commerce: наприклад, PrestaShop (відкрита CMS, спеціально орієнтована на онлайн-торгівлю, зі схожою філософією до OpenCart) чи Magento (потужна open-source платформа від Adobe, розрахована на великі бізнеси з високими вимогами до масштабованості). Вибір конкретної системи залежить від багатьох факторів: розміру та бюджету бізнесу, вимог до функціоналу, наявності технічних знань у команди, планів щодо розвитку тощо. Щоб узагальнити відмінності між найбільш популярними підходами (самостійно

хостованими CMS і SaaS-платформами), доцільно порівняти їх за основними параметрами [12].

Від правильного вибору CMS залежить, наскільки швидко можна запустити проект, як легко адмініструвати сайт та масштабувати бізнес в майбутньому, і які витрати (часові та фінансові) будуть супроводжувати підтримку магазину. Використання CMS дозволяє підприємствам будь-якого розміру вийти на онлайн-ринок з мінімальними бар'єрами: готові платформи беруть на себе значну частину роботи, надаючи перевірені рішення для типових задач інтернет-комерції. Надалі, у міру зростання магазину, саме можливості обраної CMS визначатимуть простоту впровадження нових функцій – чи то запуск програми лояльності, чи інтеграція з маркетплейсами, чи впровадження мобільного застосунку. Таким чином, роль CMS у створенні інтернет-магазинів полягає у наданні базису, який робить процес розробки швидшим і доступнішим, а подальше управління – гнучким і ефективним. Завдяки системам управління контентом власники можуть зосередитися на бізнес-логіці та контентній частині свого інтернет-магазину, залишаючи технічні нюанси платформи, що, врешті-решт, сприяє успішному розвитку онлайн-бізнесу в кросплатформеному середовищі.

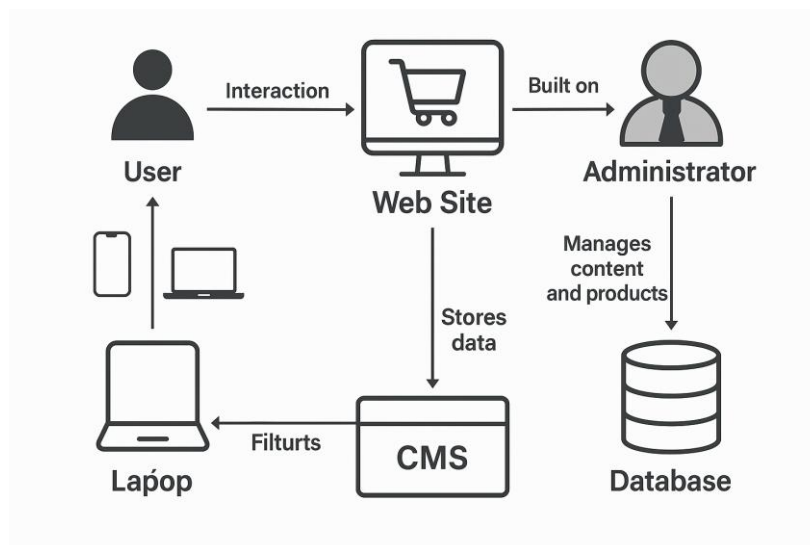


Рисунок 1.3 - Схема взаємодії користувача, адміністратора та системи керування Інтернет-магазином (CMS)

1.3. Особливості забезпечення безпеки в CMS для Інтернет-магазину

Безпека є критичною складовою успішної роботи Інтернет-магазину на базі систем управління контентом (CMS). Онлайн-магазини обробляють конфіденційні дані користувачів (особисті дані, облікові записи, платіжну інформацію), тому вони стають привабливою цілью для кіберзлочинців. Вразливий веб-сайт може призвести до витоку даних клієнтів, фінансових збитків та втрати репутації бізнесу. Тому сучасні CMS для e-commerce реалізують комплекс механізмів захисту, а розробники та адміністратори мають приділяти особливу увагу питанням безпеки. У цьому підрозділі розглянуто типові загрози для безпеки CMS інтернет-магазинів та способи захисту від них на прикладі зв'язки WordPress + WooCommerce, а також виконується порівняння із платформами OpenCart, Joomla та Shopify щодо механізмів забезпечення безпеки.

Веб-сайти на основі CMS піддаються широкому спектру атак. Серед найбільш поширених загроз – SQL-ін'єкції, міжсайтовий скриптинг (XSS), фішингові атаки, а також експлуатування вразливостей плагінів і тем оформлення [13].

SQL-ін'єкція – це атака, за якої зловмисник «вставляє» шкідливі SQL-команди у запити до бази даних через веб-форми або URL, отримуючи несанкціонований доступ до даних. Наприклад, якщо розробник не використав належної екранування введених даних, хакер може дописати фрагмент 'OR '1'='1 до параметра в URL і отримати доступ до усієї таблиці користувачів. Атаки XSS полягають у впровадженні зловмисного скрипту на веб-сторінку (через поля вводу, коментарі тощо), який у браузері жертви викрадатиме її дані або виконуватиме небажані дії від її імені. Обидва ці типи атак є прикладами ін'єкцій коду – коли хакер навмисно підсовує шкідливий код у динамічний контент сайту [14].

Фішингові атаки націлені на обман власників сайтів або їх клієнтів з метою виманити у них паролі чи дані кредитних карток. Зловмисники можуть надсилати, наприклад, підроблені електронні листи від імені популярних сервісів (банку або платформи електронної комерції) із вимогою «підтвердити обліковий запис», сподіваючись, що отримувач перейде за фальшивим посиланням і введе свої облікові дані. Окрім цього, поширеною загрозою є атаки грубої сили – автоматизований підбір паролів до адміністративної панелі. Якщо власник сайту використовує простий пароль і не обмежує кількість спроб входу, бот може за кілька годин перебрати тисячі варіантів та зламати обліковий запис. Відомий випадок – у 2023 році хакери отримали доступ до облікових записів деяких магазинів на Shopify саме через слабкі паролі адміністраторів [15].

Важливо згадати уразливості самих CMS- платформ та їхніх розширень: навіть без зусиль з боку атакуючого, сайт може бути зламано через вже відому помилку в коді плагіна чи модуля. Якщо адміністратор вчасно не оновлює систему, зловмисники можуть скористатися оприлюдненою вразливістю. Приміром, у 2021 році масово зламують WordPress-сайти через уразливий плагін WooCommerce – ця атака стала можливою через використання застарілої версії плагіна з давно виправленою помилкою безпеки

WordPress є найпопулярнішою CMS у світі (станом на 2024 рік на WordPress працює біля 45% всіх сайтів), а плагін WooCommerce перетворює її на повноцінну платформу електронної комерції. Висока популярність означає, що ця платформа постійно перебуває в полі зору хакерів та дослідників безпеки. З одного боку, це призводить до швидкого виявлення нових вразливостей і оперативного випуску оновлень. З іншого боку – WordPress стає пріоритетною мішенню для зловмисників.

Атакувальні скрипти постійно сканують Інтернет у пошуках сайтів на WordPress із відомими «дірами» в безпеці. Основні джерела ризику для WooCommerce-магазину на WordPress такі: по-перше, велика кількість додаткових плагінів і тем від сторонніх розробників. Ця модульність – перевага

WordPress, але якщо встановлено багато розширень, зростає ймовірність наявності серед них уразливого компонента. Як зазначають фахівці, велика кількість сторонніх плагінів може вводити в систему діри безпеки, особливо якщо їх не оновлювати регулярно.

По-друге, людський фактор: адміністративна панель WordPress (/wp-admin) відома, тому боти активно підбирають до неї паролі; якщо власник не ввімкнув двофакторну автентифікацію і використовує слабкий пароль, ризик злому зростає. Крім того, поширені випадки фішингу, коли власнику сайту надсилають нібито від служби підтримки WordPress чи хостингу повідомлення з вимогою негайно «оновити безпеку» за посиланням – і необережний адміністратор може повідомити свої дані зловмисникам. По-третє, як і будь-яке програмне забезпечення, самі WordPress і WooCommerce періодично містять програмні помилки. Розробники регулярно публікують патчі безпеки: наприклад, у версіях WooCommerce 8.8.0+ було виявлено XSS-вразливість, про що випущено відповідне попередження та оновлення [16].

Важливо розуміти, що переважна більшість інцидентів стається не через злом сучасного, оновленого ядра WordPress, а саме через слабкі місця на рівні плагінів або паролів. Отже, стратегія захисту WordPress-магазину зводиться до «випередження» потенційних атак – регулярного закриття відомих уразливостей та використання превентивних засобів безпеки.

Для забезпечення безпеки WordPress+WooCommerce сайтів застосовується цілий комплекс заходів, що охоплюють різні рівні системи – від мережевого трафіку до політик доступу. Насамперед, своєчасне оновлення ядра CMS, плагінів та тем – базова вимога безпеки. Необхідно відразу встановлювати нові версії, де усунуто виявлені вразливості. Адміністратор WooCommerce-магазину повинен слідкувати за анонсами оновлень WordPress і самого плагіна WooCommerce: розробники регулярно випускають «латки» для критичних багів, і зволікання з їх установкою може відкрити шлях зловмисникам. Відомо, що не виправлені уразливості в CMS чи модулях слід закривати якомога швидше. Другий ключовий напрям – посилення

автентифікації та управління доступом. Рекомендується запровадити політику складних, унікальних паролів для адміністраторів і користувачів, а також вмикати багатофакторну автентифікацію (MFA) для доступу до адмін-панелі.

Існують спеціалізовані плагіни, що додають двофакторний вхід до WordPress (наприклад, через одноразові коди або додатки типу Google Authenticator). Двофакторна автентифікація значно ускладнює злом, оскільки навіть у разі компрометації пароля зловмисник не зможе зайти без другого фактору. Окрім того, WordPress має гнучку рольову модель: варто призначати користувачам лише ті права, які потрібні для виконання їх завдань, дотримуючись принципу найменших привілеїв. Наприклад, менеджеру контенту необов'язково бути адміністратором сайту – достатньо ролі редактора. Обмеження прав доступу мінімізує потенційну шкоду у разі компрометації окремого облікового запису. Третій важливий шар – захист мережевого периметру та трафіку. Обов'язковим є використання шифрування TLS/SSL для всіх передач чутливих даних між клієнтом і сервером.

Сертифікат SSL забезпечує захищений протокол HTTPS, шифруючи введені користувачем паролі, номери карток тощо, що унеможливорює їх перехоплення зловмисником на проміжних вузлах мережі. Як свідчить практика, наявність SSL є надзвичайно важливою для довіри користувачів і захисту їх інформації

На додаток до шифрування, бажано застосовувати веб-фаєрвол (WAF) – програмний або хмарний. У контексті WordPress це може бути плагін безпеки (наприклад, Wordfence, All In One WP Security тощо) або сервіс на зразок Cloudflare WAF. Веб-фаєрвол фільтрує шкідливі HTTP-запити ще до їх виконання на сайті, блокуючи типові сигнатури SQL-ін'єкцій, XSS та інших атак на веб-додатки.

Таким чином, навіть якщо в коді сайту існує теоретична вразливість, WAF може не допустити її експлуатації. Четвертий аспект – моніторинг і аудит безпеки. Рекомендується налаштувати журнали активності WordPress та

відстежувати підозрілі дії (великі обсяги запитів, численні невдалі спроби входу тощо).

Існують плагіни-сканери, які регулярно перевіряють файли сайту на наявність відомих шкідливих програм або сторонніх вставок коду. Регулярний аудит безпеки – наприклад, перевірка сайту на типові вразливості OWASP Top 10 – допомагає своєчасно виявити та усунути проблеми

Не менш важливий рубіж безпеки – резервне копіювання даних. Навіть за максимальних превентивних заходів, не можна гарантувати стовідсоткову невразливість; тому необхідно мати план реагування. Регулярне створення резервних копій файлів сайту та бази даних, що зберігаються на окремому носії чи у хмарному сховищі, гарантує можливість відновлення роботи після потенційного інциденту. Бекапи дозволять швидко відновити магазин до працездатного стану, якщо він буде зламаний або пошкоджений. Рекомендовано автоматизувати резервне копіювання (наприклад, за розкладом раз на добу) та періодично перевіряти цілісність і працездатність резервів.

Для наочності на рис. 1.3 зображено спрощену схему багаторівневої архітектури захисту WordPress + WooCommerce сайту. Впровадження такої багаторівневої моделі – ефективний спосіб мінімізувати ризики: навіть якщо один рубіж буде подолано, наступні зупинять напад. Зокрема, користувач проходить автентифікацію з двома факторами (пароль + OTP), усі його дії йдуть через зашифроване з'єднання HTTPS (SSL). На стороні сервера працює плагін безпеки, який виступає застосунковим фаєрволом і відсіює шкідливі запити (напр., спроби SQL-ін'єкцій) до того, як вони досягнуть ядра WordPress.

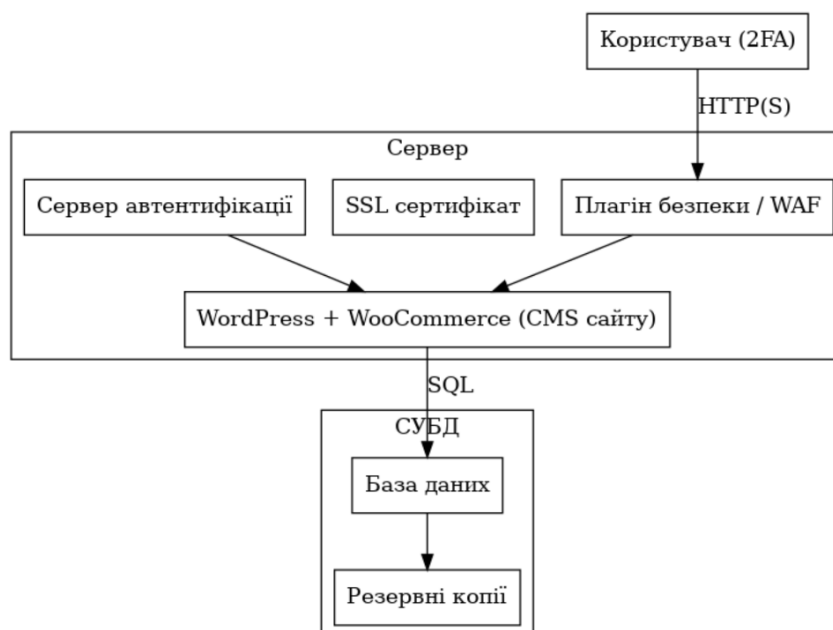


Рис. 1.4 – Схема архітектури захисту WordPress + WooCommerce

Таким чином, захист інтернет-магазину на WordPress є багатогранним процесом, що вимагає впровадження цілого комплексу заходів на різних рівнях. Ці заходи можна умовно поділити на кілька ключових напрямків: регулярне оновлення компонентів системи, посилення контролю доступу та автентифікації, захист на рівні сервера та мережі, а також розробка плану реагування на інциденти. Для наочного узагальнення цих численних рекомендацій, на рисунку 1.5 у вигляді інтелект-карти зображено основні вектори та конкретні дії для забезпечення безпеки WordPress.

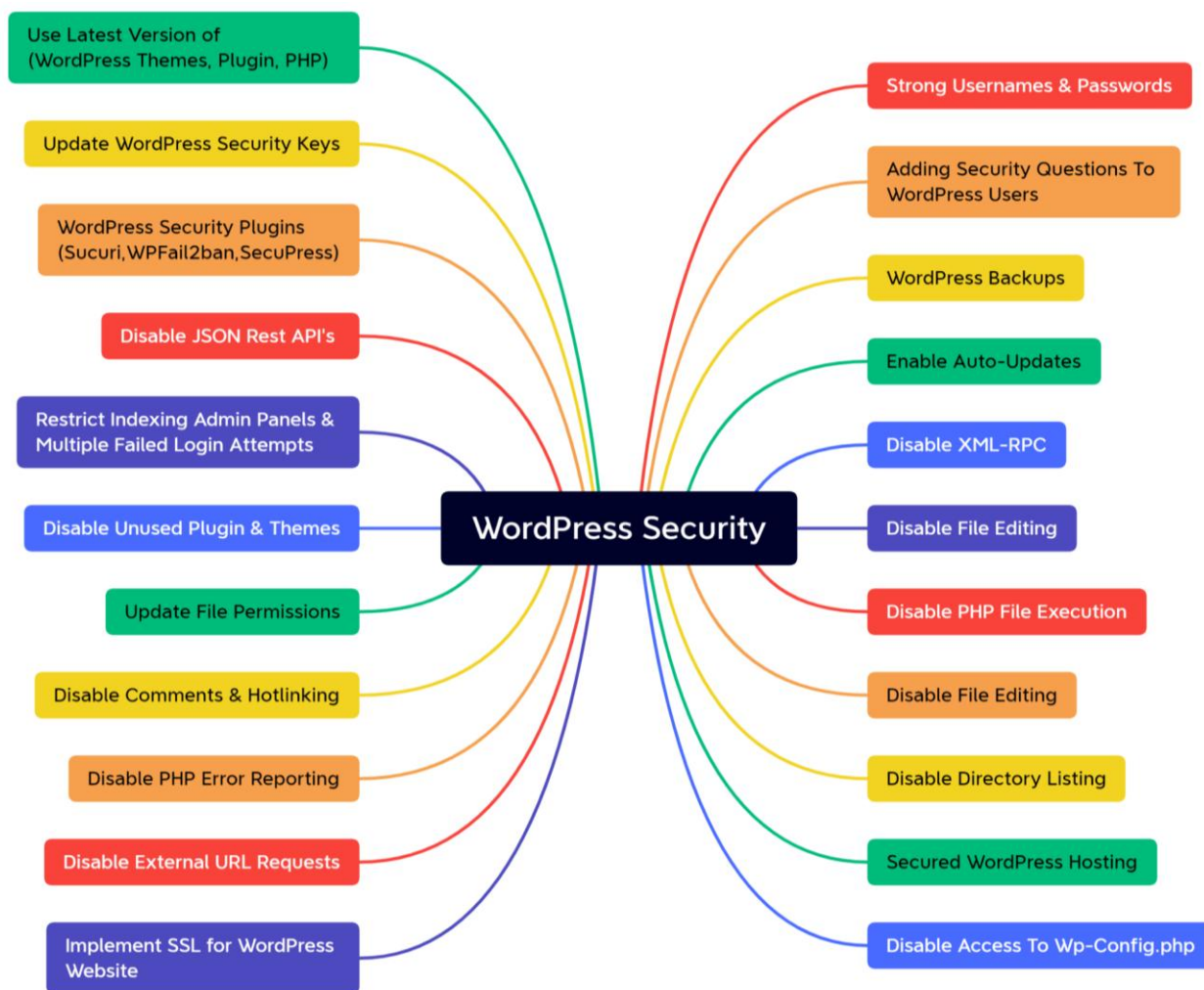


Рис. 1.5 – Основні вектори для забезпечення безпеки WordPress

Сервер автентифікації може бути задіяний для обробки 2FA або OAuth-авторизації, ізольовано від основного додатка. Дані інтернет-магазину зберігаються у СУБД (системі управління базою даних), яка розміщена окремо і доступ до якої здійснюється лише з веб-сервера. База даних зберігає інформацію про користувачів, замовлення тощо; до неї періодично робляться резервні копії, що зберігаються в безпечному сховищі. Така багатошарова архітектура – з мережевим захистом, захистом застосунку та захистом даних – дозволяє суттєво підвищити стійкість WordPress-магазину до кіберзагроз.

Розглянуті вище принципи безпеки багато в чому універсальні і застосовні до різних CMS. Втім, кожна платформа має свої особливості реалізації механізмів захисту, різний рівень відповідальності адміністратора за

підтримку безпеки, а також власні типові вразливості. OpenCart – популярна CMS із відкритим вихідним кодом, спеціально розроблена для створення інтернет-магазинів. Подібно до WordPress, OpenCart є самостійно хостинговим рішенням, тобто власник магазину повністю контролює файли програми і сервер, але й несе основну відповідальність за їх захист. OpenCart, як і WooCommerce, має відкритий код, що робить її потенційно вразливішою для атак, адже зловмисники можуть аналізувати публічний код у пошуках слабких місць.

За замовчуванням базова установка OpenCart забезпечує лише базові засоби безпеки; розширені функції (наприклад, додаткові методи автентифікації, капча на форму входу, тощо) можуть вимагати встановлення окремих модулів. Експерти відзначають, що OpenCart характеризується обмеженими можливостями безпеки «з коробки» та значною залежністю від сторонніх модулів, які, у свою чергу, можуть містити уразливості.

Таким чином, ситуація подібна до WordPress: необхідно уважно обирати та перевіряти додаткові розширення, встановлювати оновлення OpenCart і модулів одразу після виходу, а також подбати про зовнішні засоби захисту (фаєрвол на рівні сервера, регулярні бекапи тощо). Варто сказати, що через меншу частку ринку OpenCart менше привертає увагу автоматизованих ботів, аніж WordPress. Однак недооцінювати ризики не можна – великі інтернет-магазини на OpenCart також ставали жертвами хакерів через використання застарілих версій. Отже, адміністраторам OpenCart слід так само впроваджувати політики складних паролів, двофакторну автентифікацію (через додаткові модулі, оскільки штатно 2FA не передбачено), шифрувати трафік HTTPS та здійснювати аудит безпеки. Загалом, відповідальність за захист OpenCart-магазину повністю лежить на власнику сайту: CMS надає інструменти, але їх належне використання й підтримка визначають рівень безпеки.

Joomla посідає проміжне місце між WordPress та спеціалізованими e-commerce движками. Це потужна CMS загального призначення з відкритим

кодом, яка також може бути використана для інтернет-магазину (шляхом додавання компонентів електронної комерції, напр. VirtueMart або JoomShopping). Розробники Joomla традиційно приділяють значну увагу питанням безпеки: платформа побудована з урахуванням принципів захищеного програмування, регулярно виходять оновлення, що підвищують рівень безпеки [17].

На відміну від WordPress, починаючи з Joomla 3.x у систему вбудовано двофакторну автентифікацію – адміністратор може ввімкнути перевірку одноразовим кодом при вході до панелі без встановлення додаткових розширень. Також із «коробки» Joomla пропонує гнучку модель керування правами користувачів (ACL), що дозволяє тонко налаштувати, хто які дії може виконувати на сайті. Ці особливості роблять Joomla досить надійною платформою для бізнес-сайтів. Водночас, є і специфічні недоліки. По-перше, архітектура Joomla складніша, ніж у WordPress, що потенційно може призвести до помилок у налаштуванні безпеки недосвідченими адміністраторами.

Велика кількість параметрів і конфігурацій іноді ускладнює підтримку – напрочуд легко пропустити, наприклад, ввімкнення SSL для всіх сторінок або правильно не налаштувати доступ до адміністративної директорії. По-друге, порівняно менша спільнота розробників та експертів з безпеки означає, що виявлення та виправлення уразливостей може відбуватися повільніше.

Якщо WordPress-спільнота негайно реагує на нові загрози (через величезну кількість задіяних фахівців по всьому світу), то у випадку Joomla іноді мають місце затримки з випуском патчів. По-третє, Joomla, як і інші CMS, підтримує встановлення розширень (модулів, плагінів) для додавання функціональності. Надмірна кількість встановлених додатків або використання сумнівних джерел може негативно вплинути на безпеку сайту. Самі розробники рекомендують завантажувати розширення лише з офіційного каталогу та перевірених джерел, аби уникнути присутності шкідливого коду. Безкоштовні плагіни інколи містять помилки або бекдори, тому ставитися до них слід

обережно. Щоб збої не стали критичними, радять постійно робити резервні копії Joomla-сайту і перевіряти роботу встановлених розширень.

В цілому ж, безпечність Joomla-сайту багато в чому залежить від професіоналізму розробника та адміністратора, який його налаштовує. При належній підтримці (оперативні оновлення, правильна конфігурація прав, використання вбудованих засобів безпеки) Joomla здатна забезпечити високий рівень захисту для інтернет-магазину [18].

Shopify кардинально відрізняється від вищезгаданих платформ, оскільки це пропріетарне SaaS-рішення (software-as-a-service) для електронної комерції. У випадку Shopify магазин працює на хмарній платформі – користувач отримує готовий хостинг із передвстановленою системою, а усі аспекти безпеки інфраструктури забезпечує провайдер. Інакше кажучи, відповідальність за оновлення серверного ПЗ, виправлення вразливостей движка та захист бази даних лежить на команді Shopify, а не на власнику магазину. Це дає суттєві переваги з огляду на безпеку: Shopify із самого початку пропонує вбудовані засоби захисту та відповідність стандартам PCI DSS (Payment Card Industry Data Security Standard) для роботи з платіжною інформацією [19].

Кожен магазин на Shopify автоматично обслуговується через HTTPS – платформа видає SSL-сертифікат для домену магазину за замовчуванням. Таким чином, власнику не потрібно турбуватися про налаштування шифрування – усі дані клієнтів і транзакцій шифруються в дорозі. Також Shopify забезпечує безпечне зберігання чутливих даних: номери кредитних карток клієнтів не зберігаються на рівні магазину, а обробляються через захищені шлюзи платежів, сертифіковані за PCI DSS. Завдяки закритому коду цієї платформи, ризик типових веб-уразливостей (SQL-ін'єкцій, XSS) для власника магазину мінімізований – адміністратор не має прямого доступу до коду бекенду і не може випадково залишити «дірку». Всі оновлення системи відбуваються централізовано і непомітно для користувача, тобто немає поняття «застарілої версії Shopify» – користувач завжди працює з актуальною версією, яку підтримує провайдер. Усі *функції безпеки* (брандмауери, системи виявлення

атак тощо) реалізовані на рівні хмарної інфраструктури та прозорі для власника магазину. Разом з тим, слід враховувати і специфічні моменти. Хоча Shopify значно спрощує технічну сторону безпеки, людський фактор все одно залишається релевантним. Адміністратору магазину на Shopify так само потрібно дбати про надійність своїх облікових записів – використовувати складні паролі, увімкнути двофакторну автентифікацію для доступу до акаунту Shopify, обережно надавати доступ співробітникам. Якщо власник магазину нехтує цими правилами, навіть найбезпечніша платформа не вбереже від злому – як було згадано, були випадки компрометації акаунтів Shopify через слабкі паролі. Крім того, фішингові атаки можуть бути спрямовані і на користувачів магазинів на Shopify: наприклад, зловмисники можуть створити підроблений лист нібито від магазину і виманити у клієнта дані його картки. Такі атаки не пов'язані з уразливістю платформи, але їх ймовірність треба мати на увазі, проводячи навчання персоналу і попереджаючи клієнтів. Ще один момент – розширення функціональності. На Shopify, як і на інших SaaS-платформах, користувачі можуть встановлювати сторонні додатки (Apps) із офіційного магазину додатків Shopify. Хоча ці додатки проходять перевірку, теоретично вони можуть стати джерелом проблем, якщо запитують надмірні доступи до даних магазину. Адміністратор повинен переглядати дозволи, які запитує додаток, і встановлювати лише ті, яким довіряє. Проте загалом, безпекові ризики при використанні Shopify значно нижчі для кінцевого користувача, ніж у випадку самостійного розгортання CMS. Це «готове рішення», де питаннями кіберзахисту опікується висококваліфікована команда, а від власника вимагається лише дотримання базових рекомендацій (захист акаунтів, уникання підозрілих інтеграцій). Такий підхід особливо привабливий для малого та середнього бізнесу, який може не мати ресурсів на власну команду з кібербезпеки – замість цього використання платформи з вбудованими інструментами безпеки (на кшталт Shopify або інших SaaS-рішень) дозволяє знизити ризики.

РОЗДІЛ 2. WEB-САЙТ ІНТЕРНЕТ-МАГАЗИНУ ЯК ЕФЕКТИВНИЙ ЗАСІБ ОНЛАЙН ТОРГІВЛІ

2.1. Логічна модель даних

Логічна модель даних – це структурований опис предметної області, реалізованої у рамках інформаційної системи, що дозволяє формалізувати її основні об'єкти, їхні атрибути, а також взаємозв'язки між ними. У контексті інтернет-магазину логічна модель служить основою для побудови бази даних, організації навігаційної структури сайту, забезпечення коректного зберігання та обробки інформації про товари, користувачів, замовлення й платежі. Важливо, що логічна модель є незалежною від фізичних характеристик зберігання, що дозволяє її адаптацію до різних платформ та СУБД.

У процесі аналізу предметної області інтернет-торгівлі, а також структури CMS WordPress з плагіном WooCommerce, було виокремлено низку ключових сутностей, які складають основу логіки функціонування магазину. Серед них: «Товар», «Категорія товару», «Користувач», «Замовлення», «Деталі замовлення», «Платіж», «Виробник», а також додаткові сутності – «Огляд», «Кошик», «Адреса доставки».

Сутність «Товар» описує одиницю продукції, представлену в каталозі магазину, та включає такі атрибути: унікальний ідентифікатор (ID), ідентифікатор категорії, назву, короткий і повний опис, модель, артикул (SKU), ціну, статус наявності, знижку, кількість переглядів, кількість у кошику, кількість голосів користувачів, рейтинг, дату додавання в каталог, а також шляхи до зображень. Ця інформація зберігається у таблицях WordPress з типом запису `product` та у пов'язаних мета-таблицях, таких як `wp_postmeta`.

Сутність «Категорія товару» представлена у вигляді таксономії та містить атрибути: унікальний ідентифікатор категорії, назву, опис, зображення категорії, ідентифікатор батьківської категорії (для побудови ієрархії), дату створення, порядок сортування. Структура категорій дозволяє організовувати

багаторівневу навігацію та логічне групування товарів. У WooCommerce для цього використовуються таблиці `wp_terms`, `wp_term_taxonomy`, `wp_term_relationships`.

Сутність «Користувач» включає ідентифікатор користувача, логін, електронну пошту, зашифрований пароль, роль (покупець, адміністратор, менеджер), а також додаткові дані: номер телефону, адреса доставки, історія замовлень, дата реєстрації, аватар або фото профілю. Структурно ці дані зберігаються у `wp_users` та `wp_usermeta`.

Сутність «Замовлення» – одна з ключових у логіці інтернет-магазину. Вона об'єднує інформацію про процес покупки: ідентифікатор замовлення, користувача, дату створення, статус (у обробці, оплачено, доставлено тощо), метод доставки, спосіб оплати, загальну вартість, валюту, IP-адресу користувача, купони, нотатки. WooCommerce зберігає замовлення як тип запису `shop_order` у таблиці `wp_posts`, а пов'язані атрибути – у `wp_postmeta`.

Сутність «Інформація про замовлення» (деталі або позиції замовлення) відображає кожную товарну позицію в замовленні окремо, включаючи ідентифікатор товару, його назву, модель, кількість одиниць, ціну за одиницю, суму з урахуванням податків та знижок. Ці дані знаходяться у таблицях `wp_woocommerce_order_items` та `wp_woocommerce_order_itemmeta`.

Сутність «Платіж» містить інформацію про транзакцію: тип платежу (LiqPay, Stripe, готівка при отриманні тощо), статус (успішний, помилка), дату, ідентифікатор транзакції, пов'язане замовлення, суму, валюту. WooCommerce інтегрує платіжні шлюзи через API та зберігає метадані у полях замовлення.

Додатково логічна модель передбачає обробку таких сутностей, як «Огляд товару», де зберігаються текстові та числові відгуки користувачів, і «Кошик», який відображає проміжний стан замовлення до його оформлення. Дані кошика можуть бути тимчасово збережені у cookie або сесії, або синхронізовані з сервером у випадку авторизованих користувачів.

2.2. Розробка Інтернет-магазину типового торгівельного підприємства

WordPress - це система керування контентом з відкритим програмним кодом, яка відзначається простотою встановлення, широкими можливостями налаштування, підтримкою великої кількості плагінів і тем. Для реалізації електронної торгівлі в екосистемі WordPress застосовується спеціалізований плагін WooCommerce, що забезпечує усю необхідну функціональність для онлайн-продажу товарів. Використання цієї зв'язки дозволяє створити

повноцінний інтернет-магазин без глибоких знань у програмуванні, але з можливістю розширення за потребою.

Для реалізації інтернет-магазину торгівельного підприємства в середовищі WordPress було обрано безкоштовне рішення, що відповідає сучасним стандартам – CMS WordPress у поєднанні з WooCommerce. Розгортання проекту здійснювалося у локальному середовищі з використанням програмного пакету XAMPP, який містить усі необхідні компоненти: веб-сервер Apache, інтерпретатор PHP та СУБД MySQL. Цей вибір обґрунтований зручністю налаштування, багатоплатформністю, підтримкою української локалізації та відповідністю технічним вимогам WordPress.

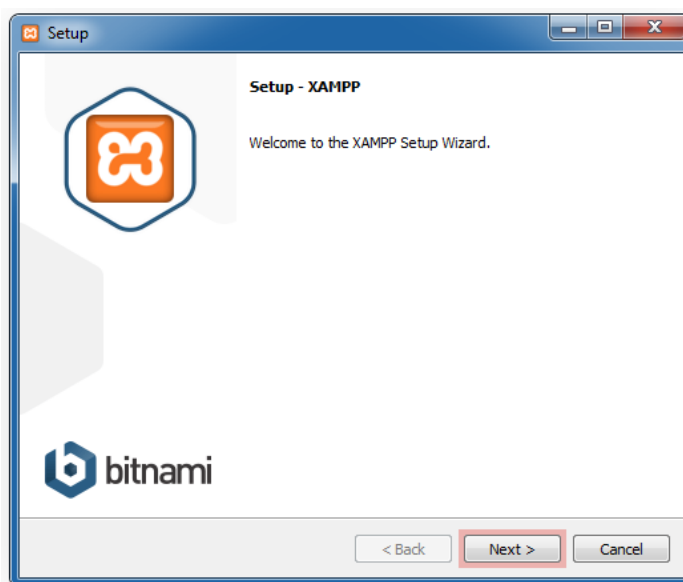


Рис. 2.2 - Установка XAMPP на комп'ютері

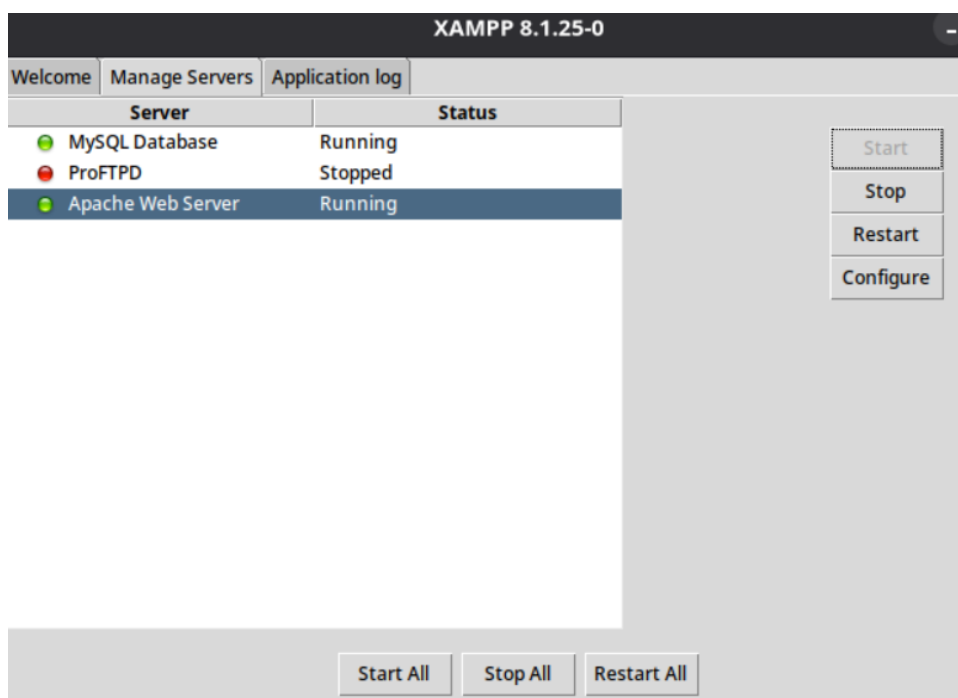


Рис. 2.3 - Запуск локального сервера Apache та MySQL

Після встановлення середовища розробки було завантажено дистрибутив WordPress з офіційного сайту, а також створено нову базу даних для майбутнього сайту в phpMyAdmin. Після цього CMS було інстальовано у локальну директорію, де здійснюється перше налаштування – вказується ім'я бази даних, логін користувача та пароль. Завдяки автоматичному інсталятору WordPress виконує первинне налаштування системи та створює таблиці бази даних.

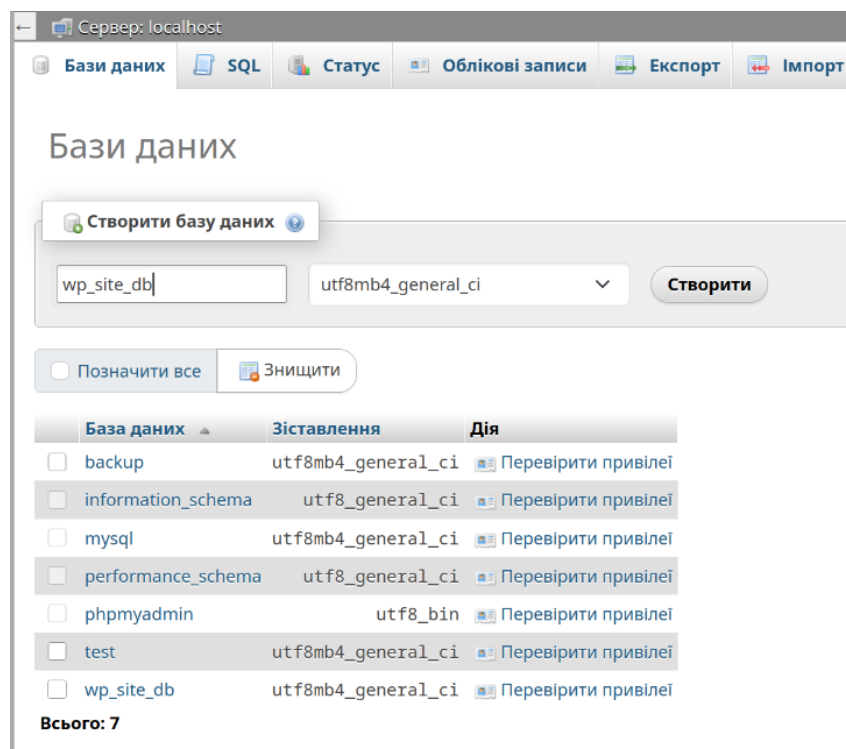


Рис. 2.4 - Створення бази даних у phpMyAdmin

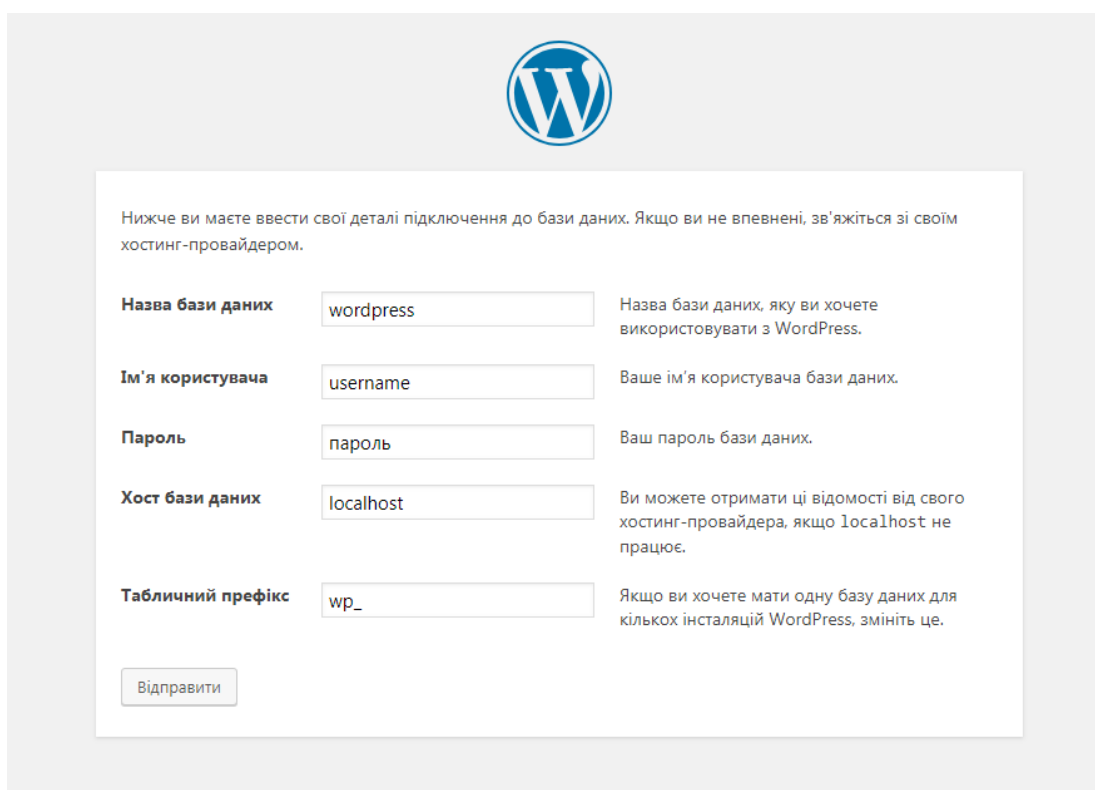


Рис. 2.5 - Початкове встановлення WordPress у браузері

Після встановлення CMS наступним кроком стало встановлення плагіна WooCommerce, що додає інтерфейс для створення товарів, оформлення замовлень, керування клієнтами та обробки оплат. Одночасно система

автоматично створює базові сторінки: каталог товарів, кошик, сторінку оформлення замовлення та особистий кабінет користувача.

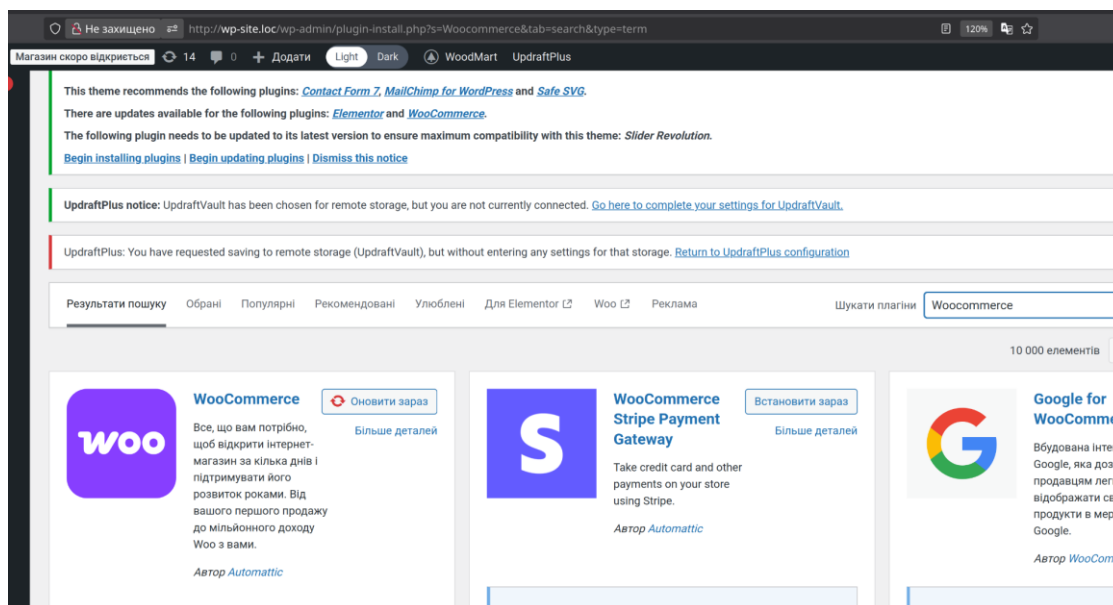


Рис. 2.6 - Встановлення та активація WooCommerce

Для візуального оформлення було обрано адаптивну тему WoodMart, яка добре сумісна з WooCommerce. Тема підтримує сучасні UI/UX-рішення, адаптується під мобільні пристрої, дозволяє змінювати колірні схеми, типографіку, структуру шаблонів без втручання в код.

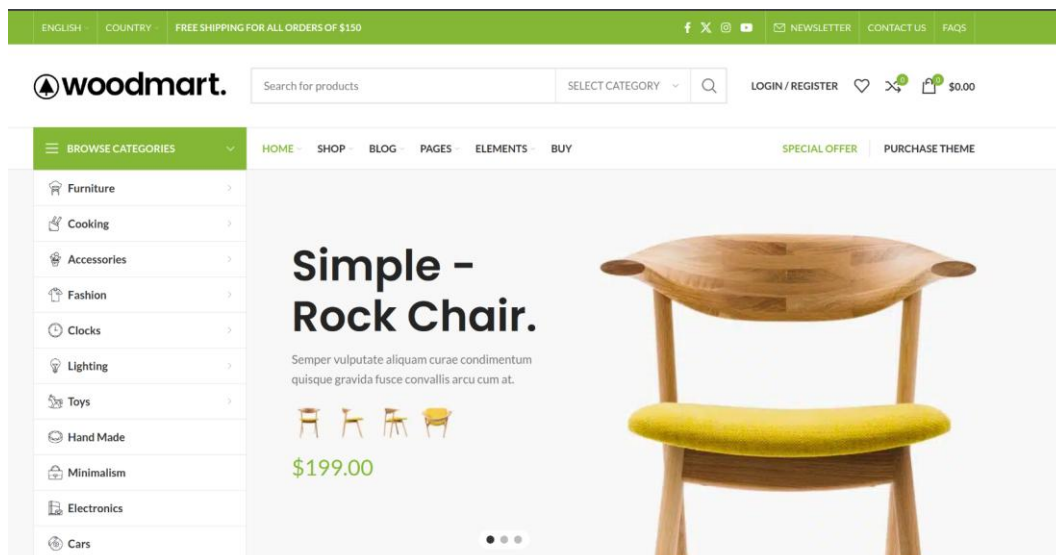


Рис. 2.7 - Головна сторінка сайту після активації теми

Хоча базовий функціонал дозволяє швидко запустити проєкт, справжня ефективність досягається шляхом кастомізації та розширення можливостей платформи відповідно до унікальних бізнес-вимог. WordPress, завдяки своїй архітектурі, надає потужний механізм для таких модифікацій через систему хуків (hooks) – дій (actions) та фільтрів (filters). Це дозволяє розробникам додавати новий або змінювати існуючий функціонал, не втручаючись у ядро CMS чи плагінів, що є ключовою вимогою для забезпечення стабільності та можливості подальших оновлень. Весь користувацький код для модифікації поведінки WooCommerce слід розміщувати у файлі `functions.php` дочірньої теми або у спеціально створеному для цього плагіні, що вважається кращою практикою, оскільки це робить код незалежним від теми оформлення та більш організованим.

Одним із поширених завдань є отримання додаткової інформації від клієнта в контексті конкретного товару, як-от текст для гравіювання або примітка щодо пакування. Для реалізації цього функціоналу створюється додаткове поле на сторінці товару. Процес включає чотири логічні кроки, кожен з яких реалізується за допомогою відповідного хука WooCommerce. Спочатку, для візуального відображення поля вводу перед кнопкою додавання до кошика використовується хук `woocommerce_before_add_to_cart_button`. Далі, для збереження введених користувачем даних у сесії кошика застосовується фільтр `woocommerce_add_cart_item_data`. Наступним етапом є відображення цієї інформації на сторінках кошика та оформлення замовлення, що досягається за допомогою фільтра `woocommerce_get_item_data`, забезпечуючи користувачу підтвердження, що його дані враховано. Фінальним і найважливішим для бізнес-процесу кроком є збереження цих даних як мета-інформації до конкретної позиції в замовленні, що реалізується через хук `woocommerce_checkout_create_order_line_item`. Це гарантує, що адміністратор магазину побачить примітку клієнта в панелі управління замовленнями.

```
<?php
add_action('woocommerce_before_add_to_cart_button',
'add_custom_product_field');
function add_custom_product_field() {
    echo '<div class="custom-note-field" style="margin-bottom: 15px;">
        <label for="custom_note">Примітка до замовлення: </label>
        <input      type="text"      id="custom_note"      name="custom_note"
placeholder="Наприклад, текст для гравіювання" />
        </div>';
}
```

```
add_filter('woocommerce_add_cart_item_data', 'save_custom_field_to_cart_item',
10, 2);
```

```
function save_custom_field_to_cart_item($cart_item_data, $product_id) {
    if (isset($_POST['custom_note']) && !empty($_POST['custom_note'])) {
        $cart_item_data['custom_note'] = sanitize_text_field($_POST['custom_note']);
    }
    return $cart_item_data;
}
```

```
add_filter('woocommerce_get_item_data', 'display_custom_field_in_cart', 10, 2);
```

```
function display_custom_field_in_cart($item_data, $cart_item) {
    if (isset($cart_item['custom_note'])) {
        $item_data[] = array(
            'key' => __('Примітка', 'woocommerce'),
            'value' => $cart_item['custom_note'],
            'display' => $cart_item['custom_note']
        );
    }
    return $item_data;
}
```

```
add_action('woocommerce_checkout_create_order_line_item',
'add_custom_field_to_order_items', 10, 4);
```

```
function add_custom_field_to_order_items($item, $cart_item_key, $values, $order) {
    if (isset($values['custom_note'])) {
        $item->add_meta_data(__('Примітка', 'woocommerce'),
        $values['custom_note']);
    }
}
```

?>

Для покращення структури сторінки товару та надання клієнтам уніфікованої інформації, наприклад, про умови доставки або гарантії, доцільно створювати додаткові вкладки. Ця задача вирішується за допомогою фільтра `woocommerce_product_tabs`. Розробник може модифікувати масив існуючих вкладок, додаючи новий елемент. Цей елемент повинен містити заголовок вкладки, її пріоритет (що впливає на позицію), та, що найважливіше, назву `callback`-функції. Ця функція відповідає за генерацію та виведення HTML-вмісту нової вкладки. Такий підхід дозволяє централізовано керувати інформаційними блоками, які є спільними для багатьох товарів, та підтримувати їхню актуальність.

```
<?php
```

```
add_filter('woocommerce_product_tabs', 'add_custom_product_tab');
```

```
function add_custom_product_tab($tabs) {
```

```
    $tabs['shipping_info_tab'] = array(
```

```
        'title' => __('Доставка та Оплата', 'woocommerce'),
```

```
        'priority' => 50,
```

```
        'callback' => 'render_shipping_info_tab_content'
```

```
    );
```

```
    return $tabs;
```

```
}
```

```
function render_shipping_info_tab_content() {
```

```
    echo '<h2>Інформація про доставку та оплату</h2>';
```

```
    echo '<p>Ми доставляємо товари по всій Україні службою "Нова Пошта".
```

```
Вартість доставки розраховується за тарифами перевізника.</p>';
```

```
    echo '<p>Можливі способи оплати: Онлайн-оплата карткою Visa/MasterCard  
через LiqPay або післяплата при отриманні у відділенні "Нова Пошта".</p>';
```

```
}
```

```
?>
```

Ще одним важливим аспектом кастомізації є програмне керування цінами, що є незамінним інструментом для проведення маркетингових акцій. Замість ручної зміни цін для великої кількості товарів, можна динамічно застосовувати знижки на основі певних умов, наприклад, належності товару до конкретної категорії. Це реалізується за допомогою фільтра `woocommerce_product_get_price`, який перехоплює значення ціни "на льоту". Функція, прикріплена до цього фільтра, перевіряє умову (в даному випадку, приналежність товару до категорії 'accessories' за допомогою функції `has_term`) і повертає або модифіковану, або оригінальну ціну. Такий метод не змінює дані в базі, а лише фільтрує їх при відображенні та розрахунках. Для коректного візуального представлення акційної пропозиції необхідно також використати фільтр `woocommerce_get_price_html`, який дозволяє сформувати HTML-код із закресленою старою ціною та новою акційною, що є усталеним стандартом у електронній комерції.

```
<?php
```

```
add_filter('woocommerce_product_get_price', 'apply_category_discount', 10, 2);
```

```
function apply_category_discount($price, $product) {
    if (has_term('accessories', 'product_cat', $product->get_id())) {
        $discounted_price = $price * 0.90; // Знижка 10%
        return $discounted_price;
    }
    return $price;
}
```

```
add_filter('woocommerce_get_price_html', 'display_discounted_price_html', 100, 2);
```

```
function display_discounted_price_html($price_html, $product) {
    if (has_term('accessories', 'product_cat', $product->get_id())) {
        $original_price = $product->get_regular_price();
        $sale_price = $product->get_price();
```

```

    $price_html    =    wc_format_sale_price(wc_get_price_to_display($product,
array('price' => $original_price)), wc_get_price_to_display($product, array('price' =>
$sale_price))) . $product->get_price_suffix();
    return $price_html;
}
return $price_html;
}
?>

```

Завдяки відкритому коду та архітектурі, що базується на хуках, розробник отримує можливість глибокої адаптації функціоналу інтернет-магазину до специфічних бізнес-процесів будь-якого торговельного підприємства. Це може включати інтеграцію зі сторонніми системами, як-от CRM чи програми складського обліку, а також реалізацію складних цінових політик і програм лояльності. Саме ця гнучкість у поєднанні з потужністю WordPress робить дану платформу ефективним інструментом для успішного ведення комерційної діяльності в мережі Інтернет

2.3. Функціональні можливості Web-сайту

Функціональні можливості веб-сайту відіграють ключову роль у забезпеченні ефективності онлайн-торгівлі та задоволенні потреб як бізнесу, так і кінцевих користувачів. Реалізований інтернет-магазин на базі CMS WordPress із використанням плагіна WooCommerce охоплює всі основні елементи електронної комерції – від демонстрації товарів до обробки замовлень, керування клієнтською базою, адміністрування асортименту та комунікації з покупцями.

Перш за все, сайт підтримує адаптивне відображення контенту, що забезпечує його коректну роботу на будь-яких пристроях – настільних комп'ютерах, ноутбуках, планшетах та смартфонах. Завдяки темі з підтримкою адаптивного дизайну інтерфейс автоматично підлаштовується під роздільну

здатність екрана, зберігаючи зручність навігації, швидкість завантаження та доступність усіх функцій. Це дозволяє охопити широку аудиторію та мінімізувати ризики втрати клієнтів, які здійснюють покупки з мобільних пристроїв.

Центральним елементом системи є інтерактивний каталог товарів, у якому вся продукція класифікована за категоріями та підкатегоріями. Кожна товарна позиція містить зображення, назву, опис, ціну, рейтинг, доступні варіації (наприклад, розмір або колір) та кнопку «Додати в кошик». Для зручності користувачів реалізовано систему фільтрації за параметрами (ціна, наявність, популярність, рейтинг), а також пошуковий функціонал, що дозволяє оперативно знаходити потрібні товари.

Інтерактивний кошик дозволяє переглядати обрані позиції, змінювати їх кількість, видаляти або оновлювати замовлення. Процес оформлення замовлення реалізовано через зручну багатоступеневу форму, де користувач послідовно заповнює контактні дані, обирає спосіб доставки та оплати. Оплата здійснюється через інтеграцію з українською платіжною системою LiqPay, що дозволяє приймати онлайн-платежі через банківські картки. Всі транзакції здійснюються через захищене HTTPS-з'єднання, що відповідає вимогам безпеки для електронної комерції.

Для зареєстрованих користувачів доступний особистий кабінет, який виконує роль централізованого сховища інформації про всі взаємодії з магазином. У кабінеті можна переглядати історію замовлень, дублювати попередні покупки, змінювати пароль, редагувати контактну інформацію, зберігати кілька адрес доставки. Також передбачена можливість залишати відгуки до придбаних товарів, що сприяє формуванню системи оцінювання та довіри серед покупців.

З боку адміністратора реалізовано потужну панель керування, яка дозволяє створювати, редагувати й видаляти товари, керувати категоріями, змінними атрибутами, цінами та описами. Доступна статистика продажів, стан виконання замовлень, історія активності клієнтів. WooCommerce також надає

можливість створювати купони та знижки, що дає змогу проводити маркетингові кампанії без використання сторонніх сервісів. Адміністративна панель дозволяє ефективно контролювати всі процеси без залучення технічних спеціалістів. Додатково реалізовано контактну сторінку з вбудованою картою Google Maps.

Функціональність сайту розширено за допомогою мультимовності, яка реалізована через плагіни типу TranslatePress. Це дозволяє надати доступ до вмісту сайту для клієнтів із різних регіонів, з можливістю перемикання мови інтерфейсу.

У сфері безпеки передбачено встановлення відповідних плагінів, які обмежують кількість спроб входу, захищають від спаму, інформують про підозрілу активність. Крім того, реалізовано базову SEO-оптимізацію: автоматичну генерацію sitemap, встановлення мета-тегів, Open Graph-розмітку для соціальних мереж, зручну URL-структуру та можливість редагування метаданих для кожної сторінки.

У структурі сайту реалізовано ієрархію товарних категорій. Наприклад, головними категоріями виступають «Одяг», «Аксесуари», «Подарункові набори», а підкатегоріями – «Футболки», «Куртки», «Сумки» тощо. Кожна категорія має своє зображення, опис та пріоритет у меню.

Додавання товарів здійснюється через адміністративну панель, де вказується назва, артикул, ціна, знижка, опис, зображення, категорія, ключові атрибути (наприклад, колір і розмір). Для складних товарів, які мають варіації, створено варіативні позиції з динамічним вибором характеристик на фронтенді.

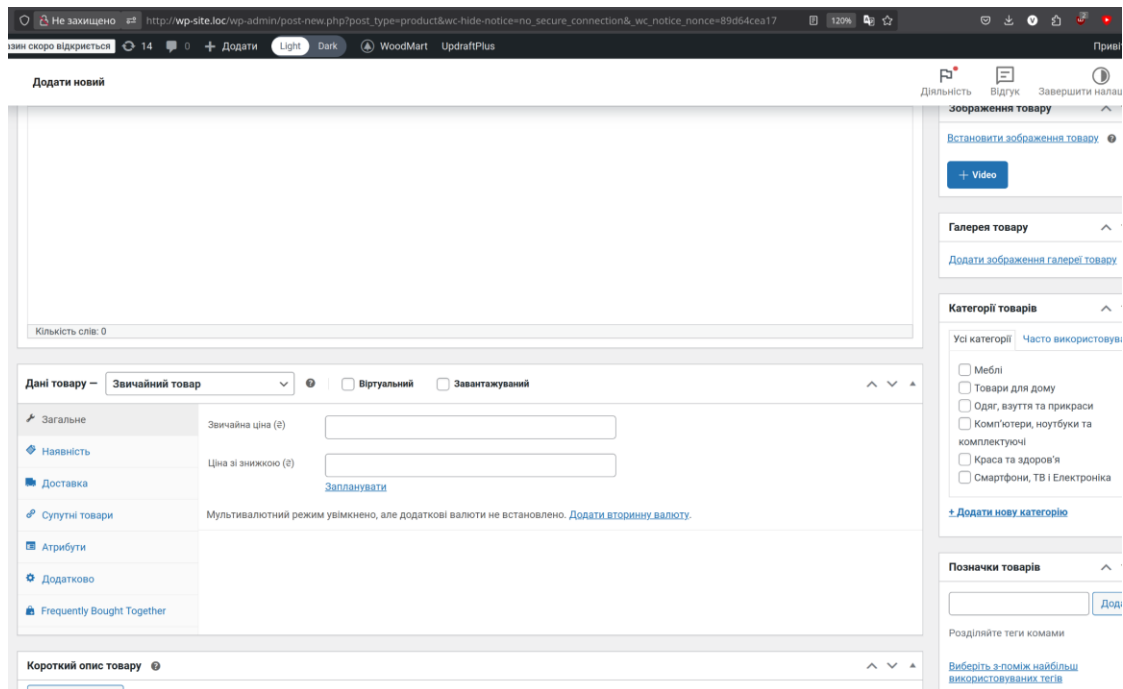


Рис. 2.8 - Інтерфейс створення товару

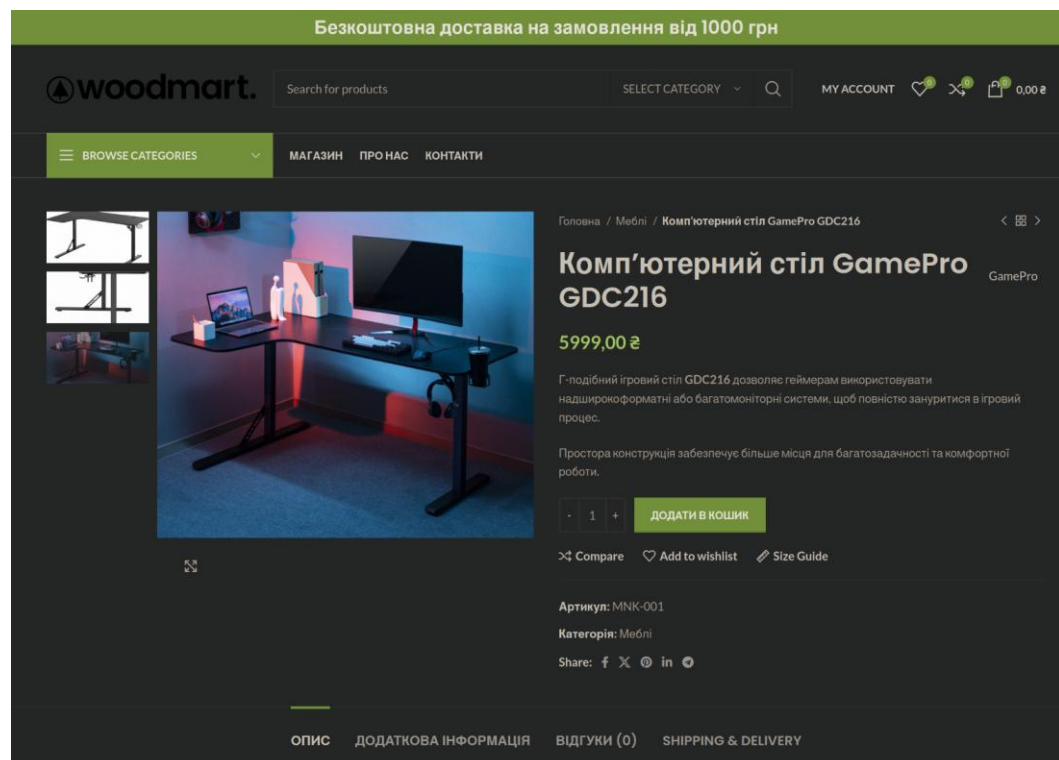


Рис. 2.9 - Перегляд товару на сайті користувачем

Усі товари відображаються в загальному каталозі, де реалізовано фільтрацію за ціною, категорією, наявністю, рейтингом. Для зручності користувача реалізовано пошук за назвою товару, сортування та функцію

швидкого перегляду. Оформлення замовлення відбувається у кілька етапів: додавання товару до кошика, введення контактної інформації, вибір доставки та оплати.

ТОВАР	ЦІНА	КІЛЬКІСТЬ	ПРОМІЖНИЙ ПІДСУМОК
Комп'ютерний стіл GamePro GDC216	5999,00 ₴	1	5999,00 ₴

Код купону

ПІДСУМКИ КОШИКА

Проміжний підсумок: 5999,00 ₴

Доставка: ☒ Нова Пошта, на відділення
☐ БЕЗКОШТОВНО на відділення Укрпошти
☐ Самовивіз

Доставка до Відділення №1: вул. Князя Володимира 109-К, Рівне, Рівненська, 1.

Загалом: 5999,00 ₴

Рис. 2.10 - Кошик з обраними товарами

Оплата реалізована через плагін інтеграції з платіжною системою LiqPay. Всі платежі проходять через захищене з'єднання HTTPS. Користувач отримує лист підтвердження після завершення оплати, а адміністратор бачить статус кожного замовлення в панелі керування.

ПЛАТІЖНІ ДАНІ

Ім'я * Прізвище *

Назва компанії (необов'язково)

Країна / Регіон *

Телефон *

E-mail адреса *

☐ I would like to receive exclusive emails with discounts and product information

☐ Доставити на іншу адресу?

Місто *

Відділення/поштамат *

Нотатки до замовлення (необов'язково)

ВАШЕ ЗАМОВЛЕННЯ

ТОВАР	ПРОМІЖНИЙ ПІДСУМОК
Комп'ютерний стіл GamePro GDC216 × 1	5999,00 ₴
Проміжний підсумок	5999,00 ₴
Купон: nuwm2025	-200,00 ₴ ×
Загалом	5799,00 ₴

Доставка: ☒ Нова Пошта, на відділення
☐ БЕЗКОШТОВНО на відділення Укрпошти
☐ Самовивіз

☐ Банківський переказ
☐ Готівка при отриманні
☐ Оплата картою, Apple Pay, Google Pay ☒ **plata by mono**
☒ **LiqPay LIQPAY** >>

Оплата за допомогою платіжної системи LiqPay

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

Рис. 2.11 - Сторінка оформлення замовлення

Тестовий режим

LIQPAY

Дані про оплату

Оплата замовлення №27977

До сплати: 5799.00 UAH

Pay

Оплатити через Google Pay

або

Карта Privat24 Раiffeisen

Номер картки

0000 0000 0000 0000

Термін дії CV2

MM/YY

Відправити квитанцію на e-mail

Натискаючи на кнопку «Оплатити», ви підтверджуєте що ознайомлені з переліком інформації про послугу та приймаєте умови публічного договору

Оплатити 5799.00 UAH

Скасувати оплату

3700 безкоштовно з мобільних +38-073-716-11-31 для дзвінків з-за кордону

Допомога онлайн

24 Privat24 Raiffeisen ID Check LIQPAY

Рис. 2.12 - Сторінка оплати

Крім базових сторінок, створено окремі секції «Про нас», «Контакти», «Доставка та оплата», «Відгуки». У розділі контактів розміщено інтерактивну карту, форми зворотного зв'язку та необхідну контактну інформацію підприємства.

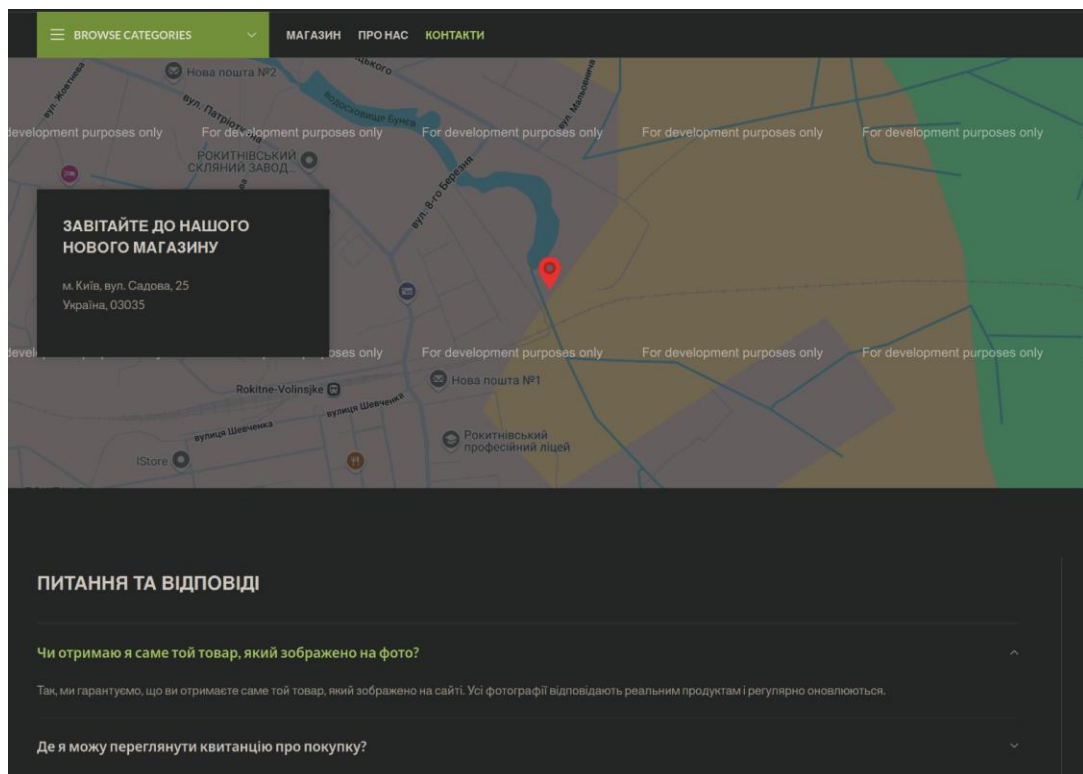


Рис. 2.13 - Контактна сторінка з картою та зворотним зв'язком

З боку адміністратора реалізовано зручну панель управління: доступ до замовлень, перегляд історії покупок, статистика, купони на знижки, керування асортиментом. Це дозволяє ефективно підтримувати роботу магазину без залучення технічних фахівців.

Після завершення розробки сайт було протестовано у різних браузерях та на різних пристроях. Завдяки використанню адаптивної теми сайт коректно відображається на мобільних телефонах, планшетах та десктопах. Також протестовано сценарії замовлення, обробки платежів, надсилання листів та функціонування особистого кабінету.

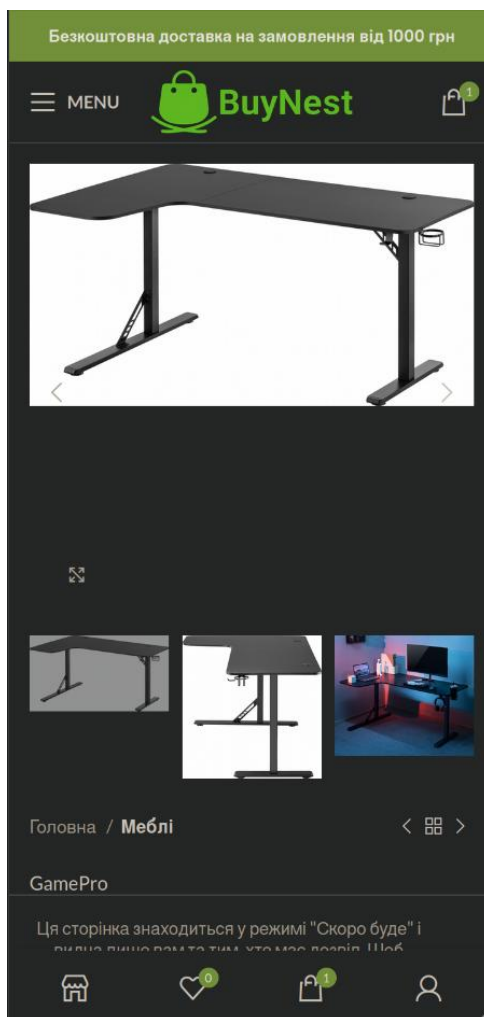


Рис. 2.14 - Мобільна версія сайту

WordPress як CMS забезпечує гнучке керування вмістом сайту. Адміністратор може змінювати структуру меню, створювати нові сторінки, додавати віджети, редагувати шаблони та керувати блоками контенту без програмування. Всі зміни можна вносити у візуальному редакторі, що робить процес обслуговування ресурсу доступним навіть для користувачів без технічної підготовки.

ВИСНОВКИ

У результаті виконання дипломного проєкту було розроблено повноцінне Web-представництво для підприємства, що спеціалізується на реалізації товарів через Інтернет. Проєкт реалізовано на основі CMS WordPress із використанням плагіна WooCommerce, що дозволило створити функціональний, зручний та адаптивний інтернет-магазин із можливістю масштабування в майбутньому.

Розроблений сайт задовольняє поставлені цілі та охоплює повний спектр завдань, необхідних для організації онлайн-продажів. Здійснено інтеграцію функціоналу для представлення товарного каталогу, включаючи опис продукції, сортування за категоріями, прикріплення зображень, варіативність атрибутів та застосування знижок. Реалізовано систему формування та обробки замовлень з можливістю вибору методів доставки та оплати, що значно підвищує зручність для кінцевого користувача.

Сайт виконує роль цілодобового інструменту комунікації з клієнтами та каналу продажів. Завдяки адаптивному дизайну він однаково ефективно функціонує як на стаціонарних пристроях, так і на мобільних, забезпечуючи широке охоплення цільової аудиторії. Упорядкована структура сторінок і логічна навігація сприяють швидкому пошуку необхідної інформації, що позитивно впливає на користувацький досвід.

Перевагою впровадженого рішення є можливість підприємства самостійно змінювати вміст сайту, оновлювати товари, ціни, описи та зображення без необхідності залучення програмістів. Це скорочує витрати на обслуговування проєкту та підвищує його гнучкість у динамічних умовах ринку.

Водночас під час реалізації проєкту були виявлені певні обмеження. Зокрема, наразі не реалізовано функцію автоматичного резервного копіювання бази даних, що є важливим компонентом захисту інформації. Також відсутній механізм автоматичного відображення нових товарів на головній сторінці, що дещо обмежує можливості просування новинок.

Загалом, розроблений інтернет-магазин виконує свої функції відповідно до вимог, поставлених у роботі, та є практичним прикладом використання сучасних веб-технологій у сфері електронної комерції. Сайт готовий до подальшого впровадження в діяльність підприємства та забезпечує перспективу для майбутнього розширення функціоналу відповідно до зростаючих потреб бізнесу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Cross-platform testing: Testing your app on 3 main OS: веб-сайт. URL: <https://training.qatestlab.com/blog/technical-articles/cross-platform-testing-3-main-os/> (дата звернення: 02.04.2025).
2. Using media queries – CSS: Cascading Style Sheets: веб-сайт. URL: https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_media_queries/Using_media_queries (дата звернення: 06.04.2025).
3. Рейтинг CMS 2025 року: веб-сайт. URL: <https://Web24.com.ua/rating-cms-2025/> (дата звернення: 10.04.2025).
4. Make your Website mobile-friendly: веб-сайт. URL: <https://wordpress.com/support/make-your-Website-mobile-friendly/> (дата звернення: 15.04.2025).
5. Responsive Web Design in WordPress: Ensuring Mobile Compatibility: веб-сайт. URL: <https://www.linkedin.com/pulse/responsive-Web-design-wordpress-ensuring-mobile-compatibility> (дата звернення: 21.04.2025).
6. The Importance of Responsive Design in E-commerce Websites: веб-сайт. URL: <https://www.wpfastestcache.com/blog/the-importance-of-responsive-design-in-e-commerce-Websites/> (дата звернення: 28.04.2025).
7. Столяр А.В. Аналіз платформ CMS для створення сайтів: веб-сайт. URL: https://elartu.tntu.edu.ua/bitstream/lib/45903/1/2024_KRB_SN-42_Stolyar.A.V.pdf (дата звернення: 03.05.2025).
8. WooCommerce vs Shopify: Market Share & Statistics: веб-сайт. URL: <https://www.mobiloud.com/blog/woocommerce-vs-shopify-market-share-statistics> (дата звернення: 07.05.2025).
9. Success Equation: Joomla + VirtueMart or WordPress + WooCommerce: веб-сайт. URL: <https://www.shopping-cart-migration.com/carts-reviews/5973-success-equation-joomla-virtuemart-or-wordpress-woocommerce> (дата звернення: 11.05.2025).

10. CMS для інтернет-магазину: веб-сайт. URL: <https://advermedia.ua/ru/blog/cms-dlya-internet-magazina/> (дата звернення: 15.05.2025).
11. Magento vs WooCommerce vs Shopify vs OpenCart vs PrestaShop: веб-сайт. URL: <http://www.mgt-commerce.com/blog/magento-vs-woocommerce-vs-shopify-vs-opencart-vs-prestashop/> (дата звернення: 19.05.2025).
12. Плюси та мінуси використання WordPress: веб-сайт. URL: <https://thedc.studio/blog/plyusy-ta-minusy-vykorystannya-wordpress> (дата звернення: 24.05.2025).
13. Безпека веб-сайтів: загрози та рішення: веб-сайт. URL: <https://www.onistudio.com.ua/bezpeka-veb-sajtiv/> (дата звернення: 29.05.2025).
14. Кібербезпека для e-commerce та ритейлу: вразливості, загрози та способи захисту: веб-сайт. URL: <https://eska.global/blog/kiberbezpeka-dlya-e-commerce-ta-retail-vrazlivosti-zagrozi-ta-sposobi-zahistu> (дата звернення: 21.05.2025).
15. CMS-плагіни: Переваги та недоліки: веб-сайт. URL: <https://embo-studio.ua/uk/blog/cms-plugins> (дата звернення: 06.06.2025).
16. Joomla: глибокий аналіз переваг і недоліків популярної CMS: веб-сайт. URL: <https://spiderWeb.com.ua/blogu/138-joomla-glibokij-analiz-perevag-i-nedolikiv-populyarnoji-cms> (дата звернення: 21.05.2025).
17. Порівняння CMS Joomla: веб-сайт. URL: <https://mercurio-cms.com/porivnjannja/joomla> (дата звернення: 14.06.2025).
18. Shopify проти WooCommerce: порівняння платформ: веб-сайт. URL: <https://www.okiktenko.com.ua/uk/shopify-proti-woocommerce-porivnyannya-platform> (дата звернення: 22.05.2025).

Додатки

Лістинг програми index.php:

```
<?php
/**
 * The main template file
 */

if( function_exists( 'woodmart_is_woo_ajax' ) && woodmart_is_woo_ajax() ) {
    do_action( 'woodmart_main_loop' );
    die();
}

get_header(); ?>

<?php

// Get content width and sidebar position
$content_class = woodmart_get_content_class();

?>

<div class="site-content <?php echo esc_attr( $content_class ); ?>" role="main">

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

</div><!-- .site-content -->

<?php get_sidebar(); ?>
```

```
<?php get_footer(); ?>
```

Лістинг програми class-menu.php:

```
<?php
```

```
namespace XTS\Admin\Modules\Dashboard;
```

```
class Menu {
```

```
    /**
```

```
     * Constructor.
```

```
    */
```

```
    public function __construct( $config ) {
```

```
        $this->get_menu( $config );
```

```
    }
```

```
    protected function get_menu( $config, $child_menu = false ) {
```

```
        if ( ( ! $this->check_show_items( array_column( $config['items'],  
'condition' ) ) ) ) {
```

```
            return;
```

```
        }
```

```
        $classes = ";
```

```
        if ( $child_menu ) {
```

```
            $classes = wd_add_cssclass( 'xts-dropdown xts-sub-menu',  
$classes );
```

```
        } else {
```

```
            $classes = wd_add_cssclass( 'xts-nav', $classes );
```

```
        }
```

```
?>
```

```

<ul class="<?php echo esc_attr( $classes ); ?>">
    <?php foreach ( $config['items'] as $item_data ) : ?>
        <?php $this->get_menu_item( $item_data ); ?>
    <?php endforeach; ?>
</ul>
<?php
}

protected function get_menu_item( $item_data ) {
    $classes    = 'xts-nav-item';
    $link_classes = 'xts-nav-link';

    if ( isset( $item_data['condition'] ) && ! $item_data['condition'] ) {
        return;
    }

    if ( ! empty( $item_data['icon'] ) ) {
        $link_classes = wd_add_cssclass( 'xts-i-' . $item_data['icon'],
$link_classes );
    }

    if ( ! empty( $item_data['class'] ) ) {
        $classes = wd_add_cssclass( $item_data['class'], $classes );
    }

    if ( isset( $item_data['child_menu'] ) && $this->check_show_items(
array_column( $item_data['child_menu']['items'], 'condition' ) ) ) {
        $classes = wd_add_cssclass( 'xts-has-child', $classes );
    }
}

```

```

        foreach ( $item_data['child_menu']['items'] as $child_item_data )
        {
            $classes .= $this->get_active_classes( $child_item_data,
            'xts-active' );
        }
    }

    $classes .= $this->get_active_classes( $item_data, 'xts-active' );

    $link_target = ! empty( $item_data['link']['new_window'] ) ? '_blank' :
    '_self';

    ?>
    <li class="<?php echo esc_attr( $classes ); ?>">
        <a class="<?php echo esc_attr( $link_classes ); ?>" href="<?php
        echo esc_url( $item_data['link']['url'] ); ?>" target="<?php echo esc_attr( $link_target
        ); ?>">

            <span>

                <?php echo esc_html( $item_data['text'] ); ?>

            </span>

        </a>

        <?php if ( isset( $item_data['child_menu'] ) ) : ?>
            <?php $this->get_menu( $item_data['child_menu'], true );
        ?>

        <?php endif; ?>
    </li>
    <?php
}

```

```

protected function get_active_classes( $item_data, $active_class ) {
    $classes = "";

    if ( ! empty( $item_data['type'] ) && 'page' === $item_data['type'] &&
isset( $_GET['page'] ) && $item_data['slug'] === $_GET['page'] ) { // phpcs:ignore
        $classes = '' . add_cssclass( $active_class, $classes );
    } else if ( ! empty( $item_data['type'] ) && 'tab' === $item_data['type']
&& ( isset( $_GET['tab'] ) && $item_data['slug'] === $_GET['tab'] ) || ( ! empty(
$item_data['parentPage'] ) && $item_data['parentPage'] === $_GET['page'] &&
empty( $_GET['tab'] ) ) ) { // phpcs:ignore
        $classes = '' . add_cssclass( $active_class, $classes );
    } elseif ( ! empty( $item_data['type'] ) && 'post_type' ===
$item_data['type'] && isset( $_GET['post_type'] ) && $item_data['slug'] ===
$_GET['post_type'] && ! isset( $_GET['taxonomy'] ) ) { // phpcs:ignore
        $classes = '' . add_cssclass( $active_class, $classes );
    } elseif ( ! empty( $item_data['type'] ) && 'post_type_taxonomy' ===
$item_data['type'] && isset( $_GET['post_type'] ) && $item_data['slug'] ===
$_GET['post_type'] && isset( $_GET['taxonomy'] ) ) { // phpcs:ignore
        $classes = '' . add_cssclass( $active_class, $classes );
    }

    return $classes;
}

/**
 * Check whether there are menu items that meet the conditions.
 *
 * @param $conditions
 *
 * @return bool

```

```
*/  
protected function check_show_items($conditions) {  
    if ( count( $conditions ) > 0 ) {  
        foreach ( $conditions as $condition ) {  
            if ( $condition ) {  
                return true;  
            }  
        }  
  
        return false;  
    }  
  
    return true;  
}  
}
```