

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка веб-магазину «SproutGarden» з використанням HTML5,
CSS3, JavaScript, PHP та MySQL

Виконала: студентка IV курсу, групи СН-42

спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

(підпис)

Самойленко Ю.С.

(прізвище та ініціали)

Керівник

(підпис)

Млинко Б.Б

(прізвище та ініціали)

Нормоконтроль

(підпис)

Шимчук Г.В

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Боднарчук І.О.

(прізвище та ініціали)

Рецензент

(підпис)

Стадник Н.Б

(прізвище та ініціали)

Тернопіль
2026

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Боднарчук І.О.
(підпис) (прізвище та ініціали)
« 8 » червня 2026 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)
за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)
Студентці Самойленко Юлії Сергіївни
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка веб-магазину «SproutGarden» з використанням HTML5, CSS3, JavaScript, PHP та MySQL

Керівник роботи Млинко Богдана Богданівна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «14» травня 2026 року № 4/9-239

2. Термін подання студентом завершеної роботи 24 червня 2026 р.

3. Вихідні дані до роботи Джерела з літератури та інтернету про створення клієнтської та серверної частин веб-магазину.

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз предметної області та постановка завдання на розробку. Проєктування веб-магазину «SproutGarden». 2.1. Огляд веб-технологій для розробки веб-магазину.

2.2. Архітектура веб-магазину. 2.3. Проєктування бази даних. 2.4. Проєктування фірмового стилю. 3. Розробка веб-магазину «SproutGarden». 3.1. Реалізація клієнтської частини.

3.2. Реалізація серверної частини. 3.3. Реалізація адміністративної панелі. 3.4. Тестування та перевірка працездатності веб-магазину. 4. Безпека життєдіяльності, основи охорони праці.

4.1. Ергономічні проблеми безпеки життєдіяльності. 4.2. Заходи, що покращують умови праці оператора. Висновки. Перелік джерел. Додатки.

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

1. Вступ. 2. Актуальність теми та мета роботи. 3. Аналіз предметної області. 4. Функціональні та нефункціональні вимоги. 5. Вибір веб-технологій для розробки веб-магазину.

6. Архітектура веб-магазину. 7. Проєктування бази даних. 8. Дизайн веб-магазину. 9. Макети. 10. Реалізація клієнтської частини. 11. Реалізація серверної частини. 12. Адміністративна панель. 13. Висновки.

АНОТАЦІЯ

Розробка веб-магазину «SproutGarden» з використанням HTML5, CSS3, JavaScript, PHP та MySQL // Кваліфікаційна робота освітнього рівня «Бакалавр» // Самойленко Юлія Сергіївна // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СН-42 // Тернопіль, 2026 // С. 65, рис. – 21, слайди – 13, додат. – 17, бібліогр. – 35.

Ключові слова: веб-магазин, електронна комерція, HTML, CSS, JavaScript, PHP, MySQL, адміністративна панель.

Кваліфікаційна робота присвячена практичній реалізації веб-магазину саджанців та рослин «SproutGarden». В першому розділі кваліфікаційної роботи описано актуальність теми розробки сучасних веб-сайтів електронної комерції та особливості функціонування веб-магазинів рослинної продукції. Висвітлено основні характеристики предметної області та проведено аналіз існуючих веб-магазинів. Розглянуто функціональні й нефункціональні вимоги до системи, а також перспективні напрями розвитку інформаційних систем електронної комерції.

В другому розділі кваліфікаційної роботи розглянуто процес проєктування веб-магазину «SproutGarden». Обґрунтовано вибір технологічного стеку для реалізації веб-магазину. Подано архітектуру системи, структуру бази даних, фірмовий стиль веб-магазину, а також розроблені макети сторінок у середовищі Figma.

В третьому розділі кваліфікаційної роботи описано практичну реалізацію клієнтської та серверної частин веб-магазину. Проаналізовано процес створення HTML-структур, CSS-оформлення, клієнтської логіки засобами JavaScript, реалізацію бізнес-логіки засобами PHP та взаємодію з базою даних MySQL. Проведено розробку адміністративної панелі для керування товарами, замовленнями, користувачами та повідомленнями клієнтів, а також тестування і перевірку працездатності створеного веб-магазину.

ANNOTATION

Development of the "SproutGarden" Web Store Using HTML5, CSS3, JavaScript, PHP, and MySQL // Qualification work of the educational level «Bachelor» // Samoilenko Yuliia // Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SN-42 // Ternopil, 2026 // P. 65, fig. – 21, slides – 13, annexes. – 17, references – 35.

Keywords: online store, e-commerce, HTML, CSS, JavaScript, PHP, MySQL, administrative panel.

The qualification thesis is devoted to the practical implementation of the “SproutGarden” online store for seedlings and plants.

The first chapter describes the relevance of developing modern e-commerce web systems and the specific features of online stores for plant products. The main characteristics of the subject area are presented, an analysis of existing online stores is conducted, and the functional and non-functional requirements for the system are defined.

The second chapter examines the design process of the “SproutGarden” online store. The choice of the technology stack is justified, and the system architecture, database structure, corporate identity, and page layouts developed in Figma are presented.

The third chapter describes the practical implementation of the client-side and server-side components of the online store. The development of HTML structures, CSS styling, JavaScript functionality, PHP business logic, and MySQL database interaction is discussed. An administrative panel was developed, and the functionality of the created online store was tested and verified.

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

HTML – HyperText Markup Language.

CSS – Cascading Style Sheets.

SQL – Structured Query Language.

PHP – Personal Home Page Tools.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ НА РОЗРОБКУ	9
1.1 Актуальність теми та опис предметної області.....	9
1.2 Аналіз існуючих веб-магазинів та їх функціональних можливостей	10
1.3 Перспективні напрями розвитку інформаційних систем електронної комерції.....	14
1.4 Постановка задачі розробки веб-магазину «SproutGarden»	16
1.4.1 Функціональні вимоги.....	17
1.4.2 Нефункціональні вимоги.....	18
1.5 Висновок до першого розділу	19
РОЗДІЛ 2. ПРОЄКТУВАННЯ ВЕБ-МАГАЗИНУ «SPROUTGARDEN».....	20
2.1 Огляд веб-технологій для розробки веб-магазину «SproutGarden» ...	20
2.1.1 Технології HTML і CSS	21
2.1.2 Серверні мови програмування: JavaScript і PHP	21
2.1.3 База даних MySQL.....	23
2.1.4 Visual Studio Code як середовище розробки.....	24
2.1.5 XAMPP як локальне серверне середовище	24
2.2 Архітектура веб-магазину	25
2.3 Проєктування бази даних	27
2.4 Проєктування фірмового стилю.....	29
2.4.1 Розробка логотипу	29
2.4.2 Вибір кольорової палітри та шрифтів.....	31
2.4.3 Розробка макетів сторінок у Figma	32
2.5 Висновок до другого розділу	37
РОЗДІЛ 3. РОЗРОБКА ТА ТЕСТУВАННЯ ВЕБ-МАГАЗИНУ «SPROUTGARDEN»	39
3.1 Реалізація клієнтської частини.....	39

3.1.1 HTML-структури сторінок	39
3.1.2 Створення CSS-стилів	43
3.1.3 Реалізація клієнтської логіки засобами JavaScript	46
3.2 Реалізація серверної частини веб-магазину	47
3.2.1 Реалізація бізнес-логіки засобами PHP	47
3.2.2 Взаємодія з базою даних MySQL.....	49
3.2.3 Реалізація адміністративної панелі	51
3.3 Тестування та перевірка працездатності веб-магазину	55
3.4 Висновок до третього розділу	56
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	57
4.1 Ергономічні проблеми безпеки життєдіяльності	57
4.2 Заходи, що покращують умови праці оператора	58
4.3 Висновок до четвертого розділу	59
ВИСНОВКИ	61
ПЕРЕЛІК ДЖЕРЕЛ	62
ДОДАТКИ	

ВСТУП

Актуальність теми. Внаслідок стрімкого розвитку інформаційних технологій та активного переходу бізнесу в онлайн-середовище суттєво зростає потреба у створенні зручних та функціональних веб-магазинів. Сучасні користувачі очікують швидкого доступу до товарів, простого процесу оформлення замовлення та інтуїтивно зрозумілого інтерфейсу. Тому розробка веб-магазину «SproutGarden» є актуальним напрямком сучасних досліджень у сфері веб-розробки та електронної комерції, оскільки поєднує сучасні технології з потребою користувачів у зручному онлайн-обслуговуванні.

Мета і задачі дослідження. Метою даної кваліфікаційної роботи на здобуття ступеня бакалавра є розробка веб-магазину «SproutGarden».

Для досягнення поставленої мети потрібно виконати такі завдання:

- проаналізувати існуючі веб-магазини та їх функціональні можливості;
- дослідити сучасні веб-технології, що використовуються для розробки веб-магазинів;
- спроектувати структуру бази даних для зберігання інформації про користувачів, товари та замовлення;
- розробити інтерфейс користувача веб-магазину з урахуванням принципів зручності та ергономіки;
- реалізувати серверну частину системи;
- виконати тестування працездатності розробленого веб-магазину.

Практичне значення одержаних результатів.

Практичне значення роботи полягає у створенні повнофункціонального веб-магазину «SproutGarden», який може бути використаний для продажу товарів для саду, будинку, дачі та городу в онлайн-форматі. Розроблена система забезпечує зручну взаємодію користувачів із каталогом товарів, автоматизує процеси формування замовлень та може бути адаптована для використання в реальних умовах малого та середнього бізнесу.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ НА РОЗРОБКУ

1.1 Актуальність теми та опис предметної області

Глобальна цифрова трансформація докорінно змінила архітектуру торговельних відносин, перетворивши електронну комерцію на фундамент сучасної економіки. Сучасний етап розвитку світової економіки характеризується активною цифровізацією всіх сфер господарської діяльності. Одним із ключових елементів цього процесу є електронна комерція, яка трансформувалася з альтернативного каналу збуту в базовий інструмент функціонування бізнесу. Електронна комерція стала важливим інструментом адаптації бізнесу та забезпечення населення необхідними товарами навіть за умов порушення роботи традиційної торговельної мережі. Водночас стрімкий розвиток онлайн-торгівлі супроводжується низкою викликів, серед яких удосконалення правового регулювання, забезпечення безпеки електронних платежів та підвищення стійкості логістичної інфраструктури [1].

За результатами сучасних досліджень ринок електронної комерції в Україні продовжує активно розвиватися навіть в умовах економічної нестабільності та воєнного стану. Онлайн-торгівля дозволяє підприємствам розширювати ринки збуту, зменшувати витрати на утримання фізичних торговельних точок та забезпечувати безперервність бізнес-процесів. Для споживачів електронна комерція створює можливість швидкого пошуку товарів, порівняння цін, отримання детальної інформації про продукцію та здійснення покупок у будь-який час доби [2].

Предметною областю даної роботи є інтернет-торгівля товарами для рослинництва та садівництва. Сучасний ринок товарів для догляду за рослинами охоплює широкий спектр продукції: кімнатні та садові рослини, насіння, ґрунти, добрива, садовий інвентар, засоби догляду та декоративні елементи. Популярність домашнього озеленення, ландшафтного дизайну та

садівництва сприяє зростанню попиту на подібну продукцію, що створює передумови для розвитку спеціалізованих веб-магазинів.

Веб-магазин є програмною системою, яка забезпечує електронний продаж товарів через веб-інтерфейс. До основних функцій такого ресурсу належать перегляд каталогу товарів, пошук продукції, отримання інформації про характеристики товарів, формування кошика покупок, оформлення замовлення та взаємодія з користувачами. Ефективність роботи веб-магазину значною мірою залежить від зручності інтерфейсу, швидкодії системи, безпеки обробки даних та надійності функціонування серверної частини.

Актуальність теми кваліфікаційної роботи полягає у необхідності створення сучасного веб-магазину «SproutGarden», який забезпечить зручний доступ користувачів до товарів для рослинництва та садівництва, надасть можливість швидкого пошуку та замовлення продукції, а також реалізує ефективні механізми взаємодії між покупцями та адміністрацією магазину.

1.2 Аналіз існуючих веб-магазинів та їх функціональних можливостей

Ринок інтернет-магазинів рослин та товарів для садівництва характеризується значною кількістю спеціалізованих платформ, які пропонують широкий асортимент кімнатних і садових рослин, насіння, добрив, ґрунтів та супутніх товарів для догляду за рослинами. Серед популярних веб-ресурсів на українському ринку можна виділити Florium.ua [3], Agro-Market.ua [4] та Floriya.market [5]. Кожен із зазначених веб-магазинів має власні особливості реалізації функціоналу, дизайну та взаємодії з користувачами, що безпосередньо впливає на зручність використання та ефективність здійснення покупок.

Веб-магазин Florium.ua спеціалізується на продажі саджанців, насіння, декоративних рослин та супутніх товарів для садівництва. Платформа пропонує широкий асортимент продукції, систему пошуку, каталог товарів із розподілом

за категоріями, особистий кабінет користувача, кошик покупок та можливість авторизації за допомогою електронної пошти або номера телефону. Перевагою ресурсу є великий вибір товарів та наявність детальних описів продукції. Разом із тим головна сторінка містить значну кількість банерів, акційних пропозицій, кнопок та рекламних елементів, що створює надмірне візуальне навантаження на користувача. Через велику кількість інформаційних блоків пошук необхідного товару може ускладнюватися, особливо для нових відвідувачів сайту. На рисунку 1.1 зображено головну сторінку інтернет-магазину Florium.ua.

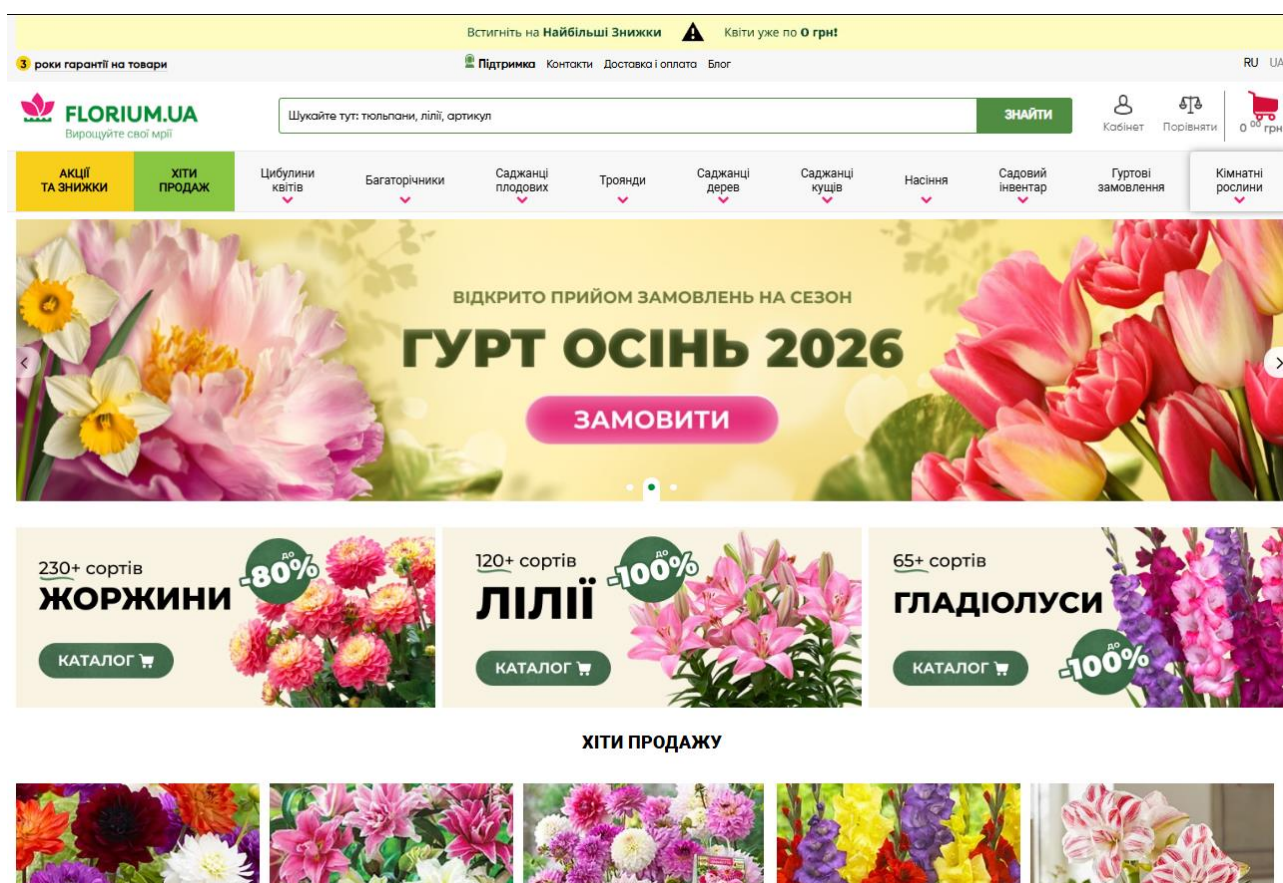


Рисунок 1.1 – Вигляд головної сторінки Florium.ua

Agro-Market.ua є одним із найбільших українських веб-магазинів товарів для садівництва та рослинництва. Платформа забезпечує користувачам доступ до широкого каталогу продукції, системи пошуку, особистого кабінету, кошика покупок та функціоналу авторизації за номером телефону або електронною

поштою. Перевагою ресурсу є значний асортимент товарів та наявність різноманітних акційних пропозицій. Водночас дизайн сайту є перевантаженим великою кількістю рекламних блоків, банерів та графічних елементів. Частина рекламних матеріалів містить зображення, створені за допомогою технологій штучного інтелекту, які не зовсім гармонійно поєднуються із загальним стилем ресурсу. Надмірна кількість візуальних елементів негативно впливає на сприйняття інтерфейсу та ускладнює навігацію сайтом. Головну сторінку Agro-Market.ua наведено на рисунку 1.2.

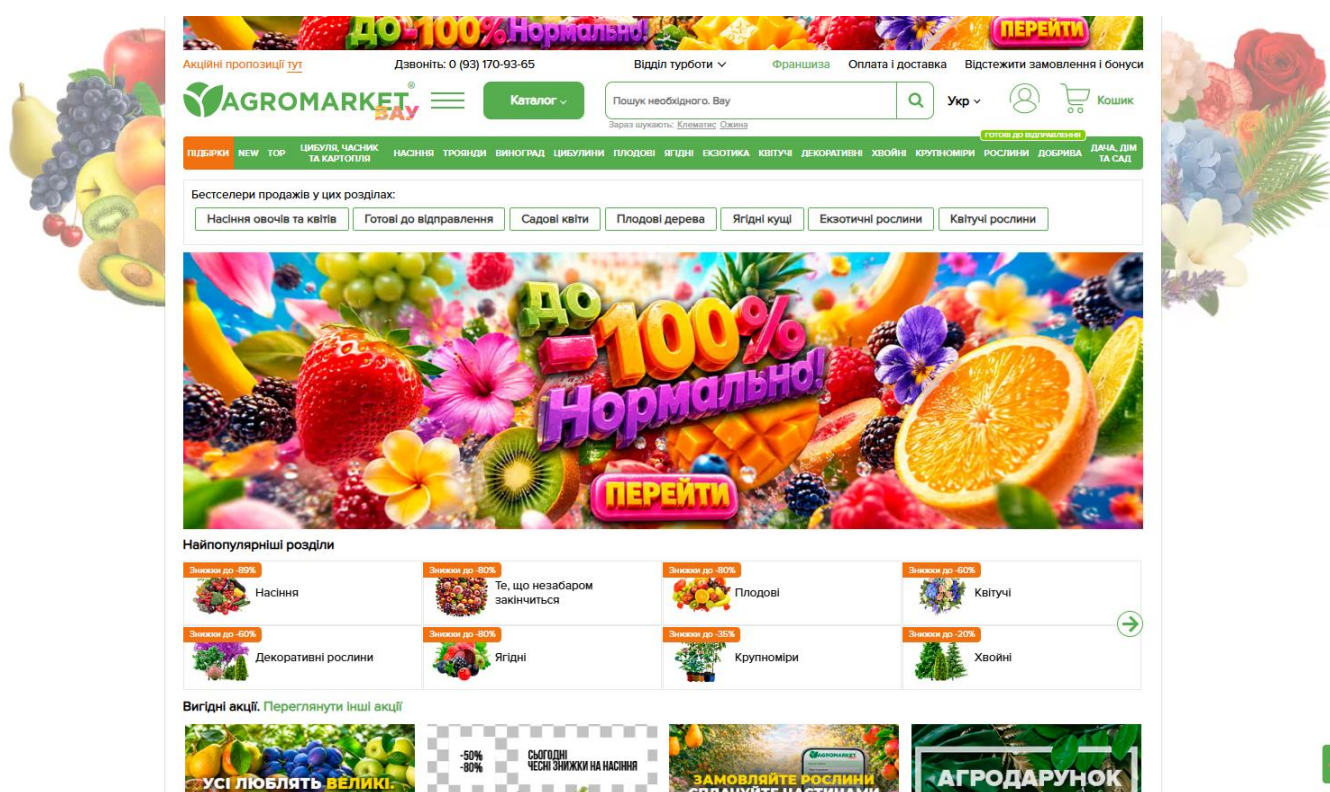


Рисунок 1.2 – Вигляд головної сторінки Agro-Market.ua

Інтернет-магазин Floriya.market орієнтований на продаж кімнатних рослин, декоративних квітів та супутніх товарів для догляду за ними. На відміну від попередніх платформ, сайт вирізняється сучасним мінімалістичним дизайном, гармонійною кольоровою палітрою та продуманою структурою сторінок. Інтерфейс не перевантажений рекламними матеріалами, що забезпечує комфортну взаємодію користувача із сайтом та дозволяє

зосередитися на товарах. Водночас недоліком ресурсу є відсутність функціоналу реєстрації та авторизації користувачів, що обмежує можливості персоналізації сервісу та ведення історії замовлень. Зовнішній вигляд головної сторінки Floriya.market представлено на рисунку 1.3.

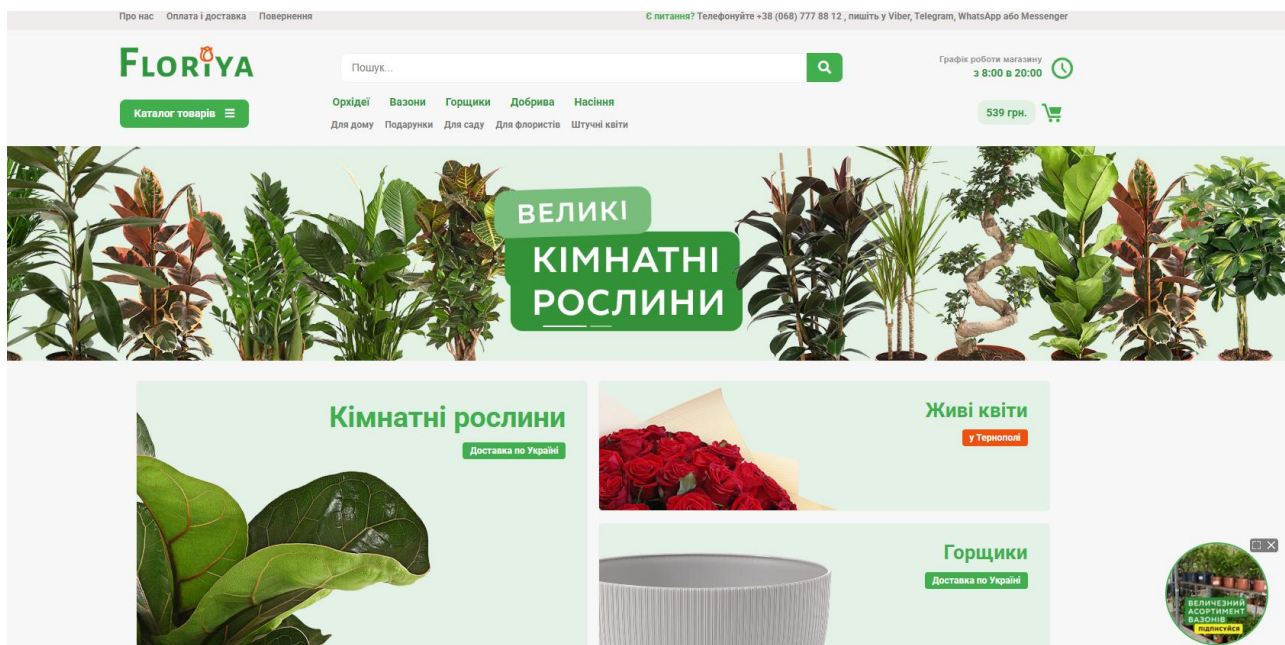


Рисунок 1.3 – Вигляд головної сторінки Floriya.market

Проведений аналіз існуючих веб-магазинів дозволив виявити низку недоліків, які негативно впливають на користувацький досвід. Основною проблемою більшості платформ є перевантаження інтерфейсу рекламними блоками, великою кількістю банерів та надмірною кількістю елементів навігації. Крім того, окремі ресурси мають недостатньо продуману структуру сторінок або обмежений функціонал взаємодії з користувачем.

Отримані результати свідчать про доцільність створення веб-магазину «SproutGarden», який поєднуватиме сучасний мінімалістичний дизайн, зручну навігацію та необхідний функціонал для роботи користувачів. При розробці системи планується використати переваги існуючих рішень. Це дозволить створити зручний та сучасний веб-магазин для продажу рослин і товарів для садівництва.

1.3 Перспективні напрями розвитку інформаційних систем електронної комерції

Сучасні інформаційні системи електронної комерції активно розвиваються в напрямі використання інтелектуальних методів аналізу даних, математичного моделювання та прогнозування поведінки користувачів. Одним із перспективних підходів є застосування стохастичних моделей для опису та аналізу інформаційних потоків, що виникають під час взаємодії користувачів із веб-ресурсом. У дослідженні [6] розглянуто математичне моделювання інформаційних сигналів за допомогою дискретних умовних лінійних випадкових процесів, що дозволяє досліджувати складні процеси, які мають випадковий характер. Подібні підходи можуть використовуватися для аналізу динаміки відвідуваності веб-магазину, змін попиту на товари та прогнозування активності користувачів.

Важливим напрямом розвитку інформаційних систем є впровадження технологій машинного навчання. Як показано у роботі [7], поєднання математичних моделей та алгоритмів керованого навчання дозволяє підвищити точність класифікації даних і виявлення прихованих закономірностей. Для веб-магазинів такі технології можуть застосовуватися під час формування персоналізованих рекомендацій, сегментації клієнтів, прогнозування попиту та автоматизованого аналізу поведінки користувачів.

Подальший розвиток цього підходу представлено в дослідженні [8], де розглянуто використання математичних моделей і методів машинного навчання для виділення інформативних ознак та підвищення ефективності класифікації даних. У сфері електронної комерції подібні методи можуть бути використані для аналізу історії покупок, визначення вподобань клієнтів та побудови систем підтримки прийняття рішень, що сприятиме підвищенню ефективності роботи веб-магазину.

Значний потенціал для розвитку інформаційних систем мають також методи математичного моделювання випадкових і шумових процесів. У

монографії [9] розглядаються підходи до аналізу стохастичних сигналів та їх статистичних характеристик. Використання таких методів дозволяє здійснювати обробку великих масивів даних, виявляти приховані закономірності та підвищувати достовірність прогнозів. Для електронної комерції це може бути корисним під час аналізу користувацької активності, оцінювання ефективності рекламних кампаній та прогнозування навантаження на систему.

Теоретичною основою багатьох сучасних інформаційних систем залишаються методи теорії ймовірностей, математичної статистики та випадкових процесів [10]. Саме вони забезпечують можливість аналізу великих обсягів даних, оцінювання ризиків, прогнозування подій та прийняття обґрунтованих управлінських рішень. У веб-магазинах статистичні методи можуть використовуватися для дослідження поведінки користувачів, оцінювання популярності товарів і підвищення ефективності маркетингових стратегій.

Окрему увагу приділяють розвитку моделей умовних лінійних випадкових процесів з дискретним часом [11]. Такі моделі дозволяють досліджувати статистичні характеристики складних процесів та визначати їх закономірності. У контексті електронної комерції їх можна використовувати для прогнозування кількості замовлень, аналізу сезонних коливань попиту та оцінювання ефективності функціонування веб-магазину на основі накопичених даних.

Перспективним напрямом є також використання багатовимірних математичних моделей для аналізу взаємопов'язаних даних. У роботі [12] розглянуто властивості багатовимірних умовних лінійних випадкових процесів, які можуть бути застосовані для дослідження складних багатоканальних інформаційних потоків. Для веб-магазину такі методи відкривають можливість одночасного аналізу великої кількості показників, зокрема відвідуваності сайту, кількості замовлень, популярності товарів та результативності маркетингових

заходів, що сприятиме підвищенню якості управлінських рішень та конкурентоспроможності електронного бізнесу.

Таким чином, перспективи розвитку інформаційних систем електронної комерції пов'язані з упровадженням математичних методів аналізу даних, технологій машинного навчання та інтелектуальної обробки інформації. Використання таких підходів дозволить створювати більш адаптивні, ефективні та орієнтовані на потреби користувачів веб-магазини, здатні швидко реагувати на зміни ринку та забезпечувати високий рівень обслуговування клієнтів.

1.4 Постановка задачі розробки веб-магазину «SproutGarden»

На основі проведеного аналізу існуючих веб-магазинів товарів для вирощування рослин було визначено основні вимоги до майбутньої системи. Дослідження показало, що більшість наявних платформ забезпечують широкий асортимент товарів, проте часто мають перевантажений інтерфейс, надмірну кількість рекламних елементів, складну навігацію та недостатню увагу приділяють зручності користувача. Крім того, окремі платформи не підтримують реєстрацію користувачів або мають обмежені можливості персоналізації та керування товарами.

З огляду на це, проблемою, яку необхідно вирішити в межах даної роботи, є відсутність зручного, функціонального та візуально збалансованого веб-магазину товарів для садівництва, який би поєднував простоту використання, сучасний дизайн та необхідний набір інструментів для здійснення покупок і керування каталогом товарів.

Метою розробки є створення веб-магазину «SproutGarden», який забезпечуватиме зручний доступ до каталогу товарів, швидкий пошук необхідної продукції, комфортний процес оформлення замовлень та ефективно адміністрування системи.

1.4.1 Функціональні вимоги

Вимоги до програмного забезпечення є формалізованим описом функціональних можливостей та характеристик системи, які визначають її поведінку під час роботи. Вони відображають сукупність умов, правил та обмежень, відповідно до яких має функціонувати веб-сайт. У контексті розробки веб-магазину «SproutGarden» вимоги визначають, які саме функції повинні бути реалізовані на сайті та яким чином користувач взаємодіє з системою. Вимоги можуть бути функціональними (описують можливості ПЗ) та нефункціональними (визначають кількісні та якісні характеристики реалізації функціоналу) [13].

На основі проведеного аналізу предметної області та існуючих аналогів було визначено функціональні вимоги до веб-магазину «SproutGarden».

Основними функціональними вимогами до системи є:

- реєстрація та авторизація користувачів із можливістю створення особистого облікового запису;
- перегляд каталогу товарів із поділом на категорії та підкатегорії;
- пошук товарів за назвою та перегляд продукції за категоріями;
- перегляд детальної інформації про товар, включаючи назву, опис, ціну та фотографії;
- додавання товарів до кошика покупок та керування його вмістом;
- додавання товарів до списку бажань для подальшого перегляду або придбання;
- оформлення замовлень та передача інформації про них до системи;
- форма зворотного зв'язку для клієнтів;
- зберігання інформації про користувачів, товари та замовлення у базі даних;
- надання адміністратору можливості додавання, редагування та видалення товарів;

- керування користувачами та перегляд інформації про оформлені замовлення через адміністративну панель;

- перегляд статистичної інформації щодо роботи веб-магазину адміністратором.

Реалізація зазначених функцій дозволить забезпечити повноцінне функціонування веб-магазину та створити зручні умови для здійснення онлайн-покупок товарів.

1.4.2 Нефункціональні вимоги

Нефункціональні вимоги визначають якісні характеристики програмної системи та встановлюють вимоги до її продуктивності, зручності використання, безпеки та надійності.

Для веб-магазину «SproutGarden» визначено такі нефункціональні вимоги:

- інтерфейс системи повинен бути інтуїтивно зрозумілим та забезпечувати зручну навігацію між сторінками і розділами каталогу;

- веб-магазин повинен мати сучасний мінімалістичний дизайн із використанням кольорової палітри, що відповідає тематиці веб-магазину;

- система повинна коректно працювати на персональних комп'ютерах, планшетах і смартфонах завдяки використанню адаптивного дизайну;

- повинен забезпечуватися захист персональних даних користувачів та розмежування прав доступу між звичайними користувачами й адміністраторами;

- система повинна забезпечувати стабільну роботу та збереження даних у базі даних;

- архітектура програмної системи повинна передбачати можливість подальшого розширення функціоналу без суттєвих змін основних компонентів;

- програмний код повинен бути структурованим і придатним для супроводу та модернізації.

Дотримання наведених нефункціональних вимог дозволить створити надійний, безпечний та зручний веб-магазин, який відповідатиме сучасним вимогам електронної комерції та очікуванням користувачів.

1.5 Висновок до першого розділу

У першому розділі дипломної роботи було досліджено предметну область електронної комерції та обґрунтовано актуальність розробки веб-магазину товарів для садівництва та рослинництва «SproutGarden». Проведений аналіз сучасного розвитку електронної торгівлі показав, що онлайн-продажі рослин, саджанців, насіння та супутніх товарів набувають все більшої популярності завдяки зручності здійснення покупок, широкому асортименту продукції та доступності послуг незалежно від місця перебування користувача.

У ході дослідження було проаналізовано існуючі веб-магазини, зокрема Florium.ua, Agro-Market.ua та Floriya.market. Аналіз дозволив визначити їх сильні сторони та виявити низку недоліків, серед яких перевантаженість інтерфейсу, надмірна кількість рекламних елементів, складна навігація, відсутність окремих механізмів персоналізації та недостатня увага до зручності користувача. Отримані результати підтвердили доцільність створення нового веб-рішення, орієнтованого на забезпечення комфортного користувацького досвіду та сучасного дизайну.

На основі проведеного аналізу було сформульовано постановку завдання розробки веб-магазину «SproutGarden». Визначено мету проєкту, яка полягає у створенні зручного веб-магазину для продажу рослин і товарів для садівництва. Окрему увагу було приділено формуванню функціональних і нефункціональних вимог до майбутнього веб-магазину.

Результати першого розділу стали основою для подальшого проєктування веб-магазину «SproutGarden», вибору технологій реалізації та розробки архітектури програмної системи, що будуть розглянуті у наступних розділах роботи.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ВЕБ-МАГАЗИНУ «SPROUTGARDEN»

2.1 Огляд веб-технологій для розробки веб-магазину «SproutGarden»

Існує безліч інструментів і платформ для створення веб-сайтів, тому кожен етап, від вибору мови програмування до використання спеціалізованих фреймворків і CMS, вимагає уважного та поінформованого підходу [14].

Технології веб-розробки – сукупність інструментів, які використовуються для створення сайтів. Вони охоплюють дві основні галузі: frontend і backend. Кожна з них відіграє важливу роль у створенні повноцінного веб-додатку або сайту:

– Frontend відповідає за те, який вигляд має сторінка і як з нею взаємодіє користувач у браузері. Це містить структуру, дизайн, анімації та інтерфейси. Основні технології, які використовуються в цьому виді розробки – HTML, CSS, JavaScript. Для створення динамічних та інтерактивних веб-застосунків застосовуються фреймворки та бібліотеки, наприклад, React.js, Angular або Vue.js.

– Backend-розробка фокусується на серверній стороні сайту, обробці запитів від клієнтів, управлінні даними та бізнес-логікою – тобто на тому, що відбувається «за лаштунками». Для створення цієї частини веб-додатків використовуються різні мови програмування: Python, PHP, Java, Ruby [14].

Frontend з backend взаємопов’язані та взаємозалежні. У такий спосіб вони забезпечують повноцінне функціонування і дарують користувачам позитивний досвід від використання.

Технології та програми для веб-розробки діляться на кілька категорій залежно від сфери застосування та призначення. Існують інструменти для створення призначеного для користувача інтерфейсу, мови програмування і фреймворки для реалізації функціонала веб-додатків, а також різні механізми для тестування готового софту [14].

2.1.1 Технології HTML і CSS

Для розробки користувацького інтерфейсу веб-магазину «SproutGarden» було обрано технології HTML5 та CSS3.

HTML – це стандартизована мова розмітки гіпертексту, яка служить скелетом кожної веб-сторінки в інтернеті. Вона визначає структуру документа: де розмістити заголовок, текст, зображення, посилання чи форму, а браузер перетворює ці інструкції на візуальний контент, доступний мільярдам користувачів [15].

CSS – стилі, які використовують для визначення зовнішнього вигляду та форматування веб-сторінок. Вона визначає різні аспекти дизайну: кольори, шрифти, розміри, розташування елементів, наявність анімацій. Основна роль CSS при створенні веб-сайтів і додатків полягає в тому, щоб забезпечити привабливий зовнішній вигляд, а також спростити процес оновлення та зміни сторінок. Використання цієї технології покращує масштабованість проєктів і полегшує їхню підтримку [16].

Поєднання HTML5 та CSS3 дозволяє створювати сучасні веб-інтерфейси, що відповідають актуальним вимогам веб-розробки. Завдяки використанню семантичної розмітки HTML5 забезпечується логічна структура сторінок і покращується їх доступність для користувачів та пошукових систем. CSS3, своєю чергою, надає широкі можливості для реалізації адаптивного дизайну, що дозволяє коректно відображати веб-ресурс на персональних комп'ютерах, планшетах і смартфонах. Саме тому зазначені технології були обрані як основа для розробки користувацької частини веб-магазину «SproutGarden».

2.1.2 Серверні мови програмування: JavaScript і PHP

Серверні мови програмування відіграють важливу роль у створенні сучасних веб-додатків, оскільки забезпечують обробку даних, взаємодію з

базами даних та реалізацію бізнес-логіки системи. Для розробки веб-магазину «SproutGarden» було обрано мови програмування JavaScript та PHP.

JavaScript – мова програмування, яка найчастіше використовується для створення інтерактивних веб-сторінок. Вона дозволяє реалізувати ряд складних рішень для веб-сайтів і надає можливість на боці клієнта взаємодіяти з користувачем [15].

JavaScript дозволяє:

- створювати сценарії веб-сторінок;
- динамічно змінювати зовнішній вигляд веб-сторінок;
- асинхронно обмінюватися даними з сервером;
- створювати односторінкові та прогресивні веб-застосунки;
- програмувати на боці сервера [15].

PHP – одна з найпопулярніших мов для веб-програмування. Її відомість пов’язана з простотою вивчення та гнучкістю. Одна з важливих особливостей PHP – можливість створювати сторінки з динамічним контентом. Також використовується для генерації HTML, виведення даних із баз, роботи з формами та обробки користувацького введення. Мова легко інтегрується з іншими технологіями, що дає змогу створювати великі проекти з її допомогою [16].

Для розробки веб-магазину «SproutGarden» було обрано мови програмування JavaScript та PHP, оскільки вони широко використовуються під час створення сучасних веб-застосунків і забезпечують реалізацію як клієнтської, так і серверної частин системи.

JavaScript доцільно використовувати для створення інтерактивних елементів інтерфейсу користувача та забезпечення динамічної взаємодії зі сторінками веб-сайту без їх повторного завантаження. Завдяки цій мові можна реалізувати навігаційні меню, випадаючі списки, форми введення даних, анімаційні ефекти та інші елементи, що покращують зручність користування веб-ресурсом.

PHP обрано як основну серверну мову програмування завдяки її простоті, широкій підтримці веб-серверів і зручній інтеграції з системами керування базами даних. Використання PHP дозволяє організувати обробку запитів користувачів, виконувати взаємодію з базою даних, забезпечувати реєстрацію та авторизацію користувачів, керування каталогом товарів, оформлення замовлень та інші функції, необхідні для роботи веб-магазину.

2.1.3 База даних MySQL

Для зберігання та обробки даних веб-магазину «SproutGarden» було обрано систему керування базами даних MySQL. Вона є однією з найпоширеніших реляційних систем керування базами даних, яка використовується для створення веб-застосунків різної складності. MySQL забезпечує надійне зберігання інформації, високу швидкість обробки запитів та підтримку великої кількості одночасних підключень [17].

MySQL використовує реляційну модель даних, що дозволяє організовувати інформацію у вигляді взаємопов'язаних таблиць. Такий підхід забезпечує цілісність даних, зменшує дублювання інформації та спрощує її обробку. Система підтримує мову структурованих запитів SQL, за допомогою якої виконуються операції створення, пошуку, оновлення та видалення даних [17].

Використання MySQL є доцільним для веб-магазину «SproutGarden», оскільки дозволяє ефективно зберігати відомості про користувачів, товари, замовлення, кошик покупок та список бажань. Крім того, система легко інтегрується з мовою програмування PHP, що забезпечує зручну реалізацію серверної частини веб-сайту.

Таким чином, MySQL є надійним та продуктивним засобом для організації бази даних веб-магазину «SproutGarden».

2.1.4 Visual Studio Code як середовище розробки

Для розробки веб-магазину «SproutGarden» було обрано інтегроване середовище розробки Visual Studio Code. Це безкоштовний редактор вихідного коду, розроблений компанією Microsoft, який підтримує велику кількість мов програмування та технологій веб-розробки [18].

Visual Studio Code вирізняється зручним інтерфейсом, високою швидкодією та широкими можливостями налаштування. Середовище підтримує підсвічування синтаксису, автоматичне завершення коду, вбудовані засоби налагодження та інтеграцію із системами контролю версій. Крім того, функціональність редактора може бути значно розширена за допомогою спеціальних розширень, що спрощують роботу з HTML, CSS, JavaScript, PHP та базами даних.

Використання Visual Studio Code є доцільним під час розробки веб-магазину «SproutGarden», оскільки це середовище забезпечує зручну роботу з усіма обраними технологіями проєкту. Зокрема, воно дозволяє ефективно створювати та редагувати веб-сторінки, серверні сценарії PHP, таблиці стилів CSS і програмний код JavaScript в межах єдиного робочого простору.

Таким чином, Visual Studio Code є сучасним та функціональним середовищем розробки, яке забезпечує комфортну організацію процесу створення веб-магазину та підвищує продуктивність роботи розробника.

2.1.5 XAMPP як локальне серверне середовище

Для локального розгортання та тестування веб-магазину «SproutGarden» було обрано програмний пакет XAMPP. Це безкоштовне кросплатформне серверне середовище, яке містить необхідні компоненти для розробки та запуску веб-застосунків, зокрема веб-сервер Apache, систему керування базами даних MySQL (MariaDB), а також інтерпретатори мов PHP і Perl [19].

Використання ХАМРР дозволяє створити локальний веб-сервер на персональному комп'ютері без необхідності оренди хостингу або розміщення проєкту в мережі Інтернет. Це значно спрощує процес розробки, тестування та налагодження веб-сайтів на початкових етапах їх створення.

Однією з переваг ХАМРР є простота встановлення та налаштування. Усі необхідні компоненти інтегровані в єдиний програмний пакет, що дає змогу швидко розгорнути середовище для роботи з РНР-скриптами та базами даних MySQL. Крім того, ХАМРР надає веб-інтерфейс керування серверними службами через панель адміністрування, що спрощує запуск і зупинку необхідних сервісів.

Вибір ХАМРР для розробки веб-магазину «SproutGarden» обумовлений його сумісністю з технологіями РНР та MySQL, які використовуються в проєкті. Завдяки цьому забезпечується можливість повноцінного тестування функціональності веб-магазину, перевірки роботи бази даних і налагодження серверної логіки до розміщення системи на реальному веб-сервері.

2.2 Архітектура веб-магазину

Архітектура програмної системи визначає принципи організації її компонентів, способи їх взаємодії та обміну даними. Від правильного вибору архітектури залежить продуктивність, масштабованість, безпека та зручність подальшого супроводу програмного продукту.

Для розробки веб-магазину «SproutGarden» було обрано клієнт-серверну архітектуру, яка є однією з найпоширеніших моделей побудови сучасних веб-застосунків. У межах цієї архітектури взаємодія між користувачем і системою здійснюється за принципом «запит – відповідь». Клієнт надсилає запит до сервера, після чого сервер обробляє отримані дані та повертає відповідь користувачу.

Клієнтом у системі виступає веб-браузер користувача, який забезпечує відображення сторінок веб-магазину та взаємодію з інтерфейсом. Серверна

частина відповідає за обробку запитів, реалізацію бізнес-логіки, взаємодію з базою даних та формування відповідей для клієнта. Такий підхід дозволяє централізовано керувати даними та забезпечувати їх узгодженість для всіх користувачів системи.

Для веб-магазину «SproutGarden» доцільним є використання трирівневої клієнт-серверної архітектури, схема якої наведена на рисунку 2.1.

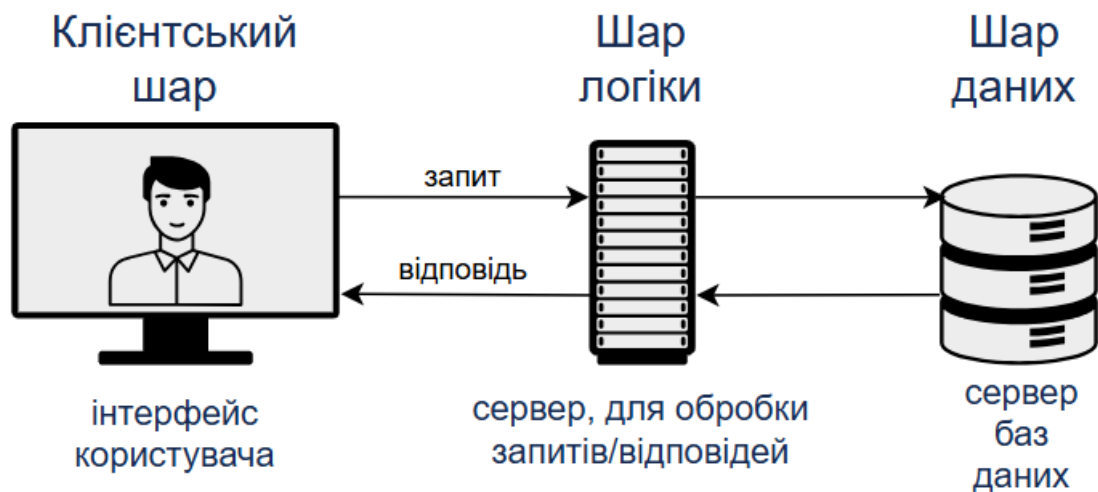


Рисунок 2.1 – Трирівнева клієнт-серверна архітектура

У трирівневій архітектурі система поділяється на три окремі рівні:

- шар представлення (або клієнтський шар), який забезпечує взаємодію користувача із системою через веб-інтерфейс;
- шар логіки, що виконує обробку запитів користувачів та реалізує бізнес-логіку веб-магазину;
- шар даних, який відповідає за зберігання та обробку інформації в базі даних.

Клієнт взаємодіє із сервером застосунку, не маючи прямого доступу до бази даних. Сервер застосунку обробляє запити, виконує необхідні операції та отримує або зберігає дані у базі даних. Така структура підвищує рівень безпеки системи та спрощує її супровід і модернізацію [20].

Використання трирівневої архітектури забезпечує можливість незалежного розвитку окремих компонентів системи без суттєвого впливу на

інші її частини. Крім того, вона дозволяє підвищити продуктивність, масштабованість та надійність веб-застосунку.

Серед основних переваг клієнт-серверної архітектури можна виділити централізоване управління даними, підтримку одночасної роботи великої кількості користувачів, зручність адміністрування та можливість подальшого розширення функціональних можливостей системи. Завдяки цим перевагам клієнт-серверна модель є оптимальним рішенням для реалізації веб-магазину «SproutGarden».

2.3 Проєктування бази даних

База даних є одним із ключових компонентів веб-магазину «SproutGarden», оскільки забезпечує зберігання та обробку інформації про користувачів, товари, замовлення та інші дані, необхідні для функціонування системи. Правильно спроектована база даних дозволяє забезпечити цілісність даних, швидкий доступ до інформації та надійність роботи веб-застосунку.

Для реалізації веб-магазину було обрано реляційну систему керування базами даних MySQL. Реляційна модель даних дозволяє організувати інформацію у вигляді взаємопов'язаних таблиць, що забезпечує зручність зберігання та обробки даних.

У процесі проєктування бази даних було визначено шість основних сутностей:

- users – зберігає інформацію про користувачів та адміністраторів системи;
- products – містить дані про товари, доступні для продажу;
- orders – зберігає інформацію про оформлені замовлення;
- cart – містить товари, додані користувачами до кошика;
- wishlist – зберігає список бажаних товарів користувача;
- message – містить повідомлення, надіслані через форму зворотного зв'язку.

Для візуалізації структури бази даних було розроблено ER-діаграму, яка відображає основні сутності та зв'язки між ними. Спроектовану ER-діаграму наведено на рисунку 2.2.

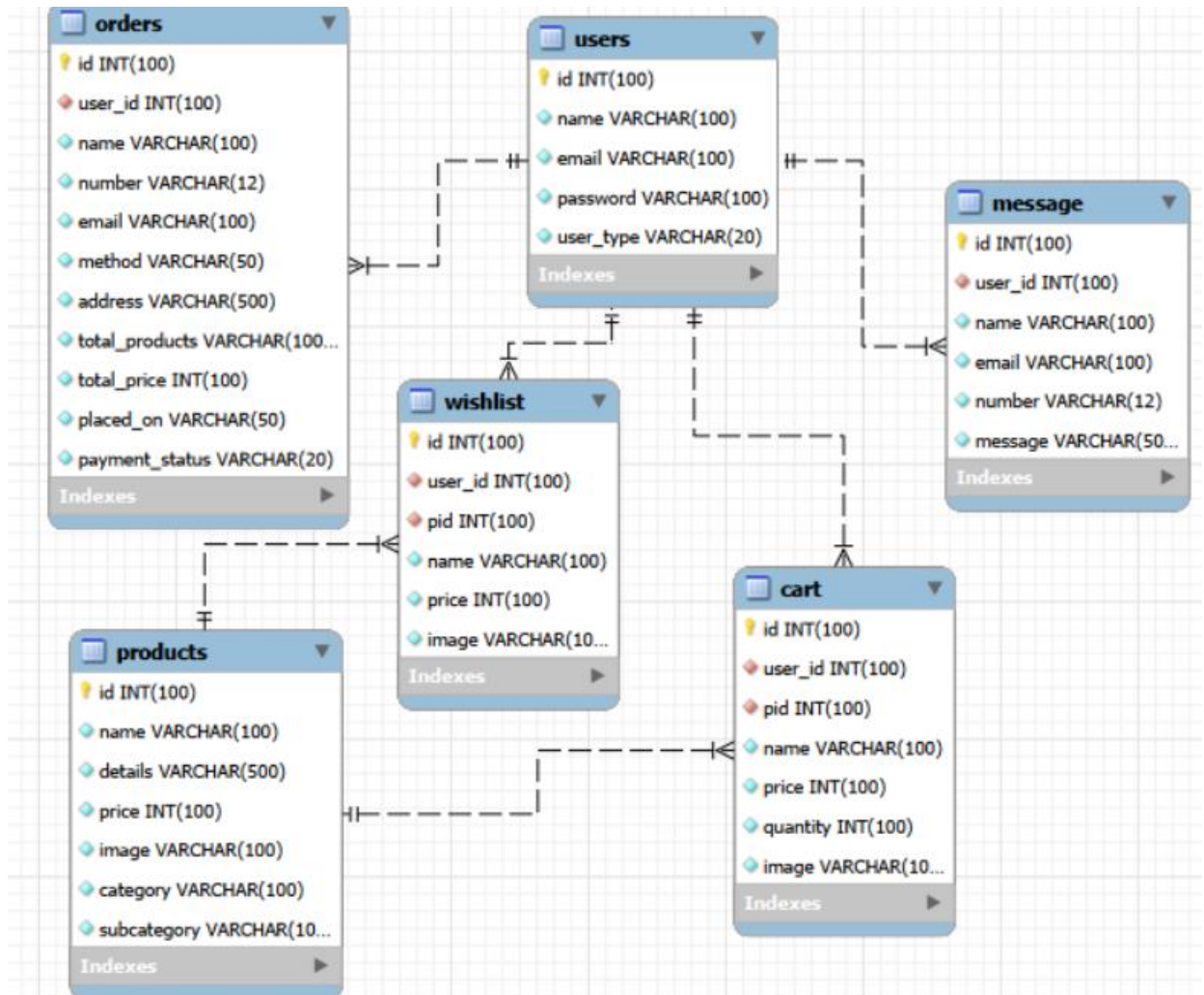


Рисунок 2.2 – ER-діаграма бази даних веб-магазину «SproutGarden»

Між таблицями існують логічні зв'язки. Кожен користувач може мати декілька замовлень, записів у кошику та списку бажань. Таблиці *cart*, *wishlist* та *orders* містять поле *user_id*, яке пов'язує відповідні записи з конкретним користувачем. Таблиці *cart* та *wishlist* також містять поле *pid*, що використовується для зв'язку з таблицею *products* і дозволяє визначити товар, який було додано до кошика або списку бажань.

Завдяки такій структурі забезпечується централізоване зберігання інформації та можливість швидкого отримання даних про користувачів, товари та замовлення.

2.4 Проектування фірмового стилю

Фірмовий стиль є важливим елементом будь-якого веб-магазину, оскільки забезпечує впізнаваність бренду та формує позитивне враження у користувачів. До основних складових фірмового стилю належать логотип, кольорова палітра, шрифтове оформлення та графічні елементи інтерфейсу. Під час розробки фірмового стилю веб-магазину «SproutGarden» було враховано основні принципи створення логотипів: простота, унікальність, асоціативність, читабельність та відповідність тематиці бренду [21].

Фірмовий стиль забезпечує єдність і консистентність у всіх аспектах бренду. Він визначає правила використання логотипу, кольорів, шрифтів та інших елементів дизайну, що дозволяє створити цілісний візуальний образ компанії на всіх платформах – від веб-сайту до графічних матеріалів. Консистентність сприяє підвищенню довіри користувачів та формує впізнаваність бренду [21].

Також важливим є те, що правильно підібране поєднання елементів фірмового стилю допомагає підкреслити унікальність та індивідуальність бренду, роблячи його більш помітним і запам'ятовуваним для споживачів [21].

2.4.1 Розробка логотипу

Першим етапом створення фірмового стилю стала розробка логотипу веб-магазину «SproutGarden». Назва магазину поєднує слова Sprout (паросток) та Garden (сад), тому основною ідеєю логотипу стало поєднання елементів садівництва та росту рослин.

У процесі розробки фірмового стилю було створено кілька варіантів логотипу, що відрізнялися за ступенем складності та композиційним рішенням. Основною метою даного етапу було знайти найбільш вдалий візуальний варіант, який би поєднував простоту, впізнаваність та універсальність використання [21].

На початковому етапі було розроблено декілька концепцій логотипу, які зображено на рисунку 2.3. Серед них були як більш деталізовані варіанти, так і мінімалістичні. Проведений аналіз показав, що складні варіанти виглядають перевантаженими та гірше масштабуються при використанні в різних середовищах.



Рисунок 2.3 – Варіанти розробки логотипу

У результаті було обрано найбільш простий та лаконічний варіант логотипу (рисунок 2.4), який не містить зайвих декоративних елементів. Такий

підхід забезпечує кращу читабельність, універсальність та адаптивність логотипу для використання як у цифрових інтерфейсах, так і в друкованих матеріалах.



Рисунок 2.4 – Обраний варіант логотипу

Розробка логотипу здійснювалася у векторному графічному редакторі CorelDRAW. Використання векторної графіки дозволяє масштабувати логотип без втрати якості, що є важливою перевагою при застосуванні в різних форматах. Для текстової частини логотипу було обрано шрифт DM Sans, який характеризується сучасним мінімалістичним дизайном та високою читабельністю.

2.4.2 Вибір кольорової палітри та шрифтів

Одним із важливих етапів проєктування інтерфейсу веб-магазину «SproutGarden» є вибір кольорової палітри та шрифтів. Від правильного поєднання кольорів залежить зручність сприйняття інформації користувачами, загальна естетична привабливість веб-магазину та формування його фірмового стилю [22].

Під час розробки дизайну було обрано природну кольорову гаму, яка відповідає тематиці магазину рослин та садівництва. Основними кольорами стали різні відтінки зеленого, що асоціюються з природою, свіжістю та екологічністю. Для фону використано світло-зелений колір, що створює відчуття легкості та не перевантажує користувача великою кількістю яскравих

елементів. Темно-зелений колір застосовується для кнопок, активних елементів інтерфейсу та навігаційного меню, що дозволяє акцентувати увагу користувача на важливих функціональних компонентах системи.

Для позначення повідомлень про помилки, видалення даних або попереджувальних дій використовується червоний колір. Такий підхід відповідає загальноприйнятим принципам веб-дизайну та дозволяє швидко привернути увагу користувача до важливих повідомлень.

Для текстового оформлення було обрано шрифт DM Sans. Даний шрифт характеризується високою читабельністю як на великих екранах, так і на мобільних пристроях.

Таким чином, обрана кольорова палітра та типографіка повністю відповідають тематиці веб-магазину «SproutGarden», забезпечують гармонійний зовнішній вигляд та покращують сприйняття інформації.

2.4.3 Розробка макетів сторінок у Figma

Проектування інтерфейсу веб-магазину «SproutGarden» здійснювалося у графічному редакторі Figma, який дозволяє створювати інтерактивні макети користувацького інтерфейсу та забезпечує зручну візуалізацію майбутнього веб-застосунку. Використання Figma дало змогу попередньо спроектувати структуру сторінок, розташування елементів та загальний стиль інтерфейсу до початку програмної реалізації.

На рисунку 2.5 наведено макет головної сторінки веб-магазину «SproutGarden», де відображено хедер із логотипом та кнопками навігації, головний банер, та блоки з товарами.

Для створення візуального наповнення веб-магазину «SproutGarden» були використані фотографії з відкритої фотоплатформи Pexels [23]. Даний ресурс надає якісні зображення, які можуть безкоштовно використовуватися в комерційних і некомерційних проєктах.

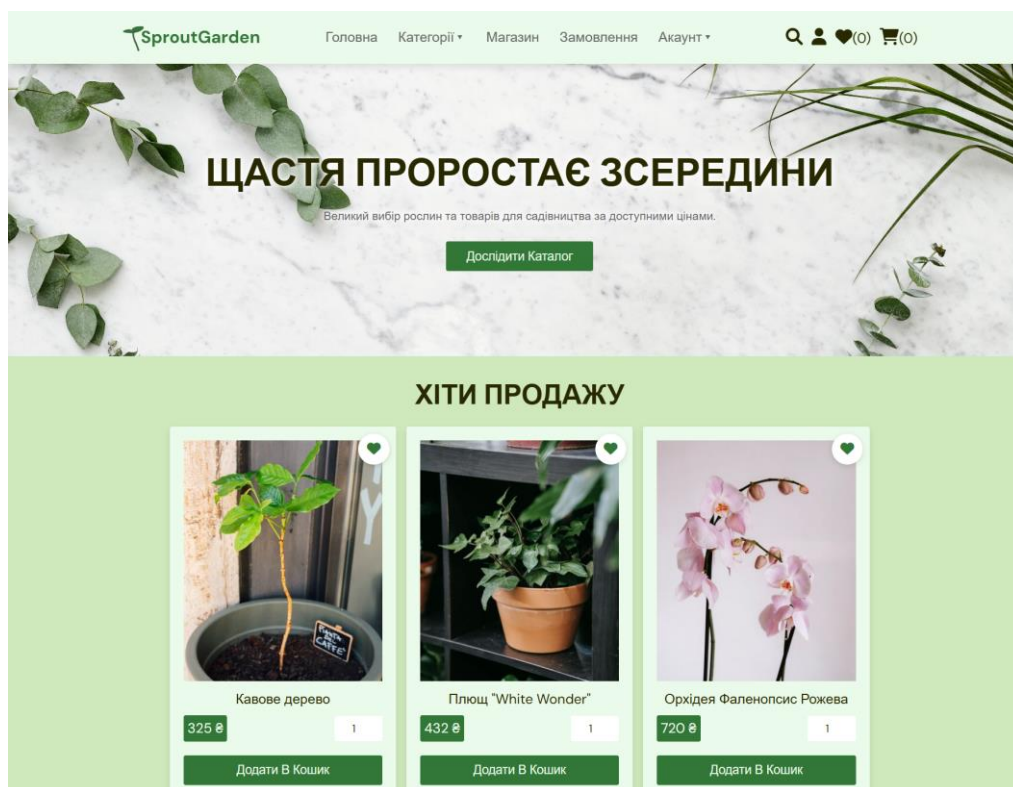


Рисунок 2.5 – Макет головної сторінки веб-магазину «SproutGarden»

На рисунку 2.6 представлено макет сторінки каталогу товарів, на якій реалізовано сітку товарних карток та фільтрація за категоріями та підкатегоріями.

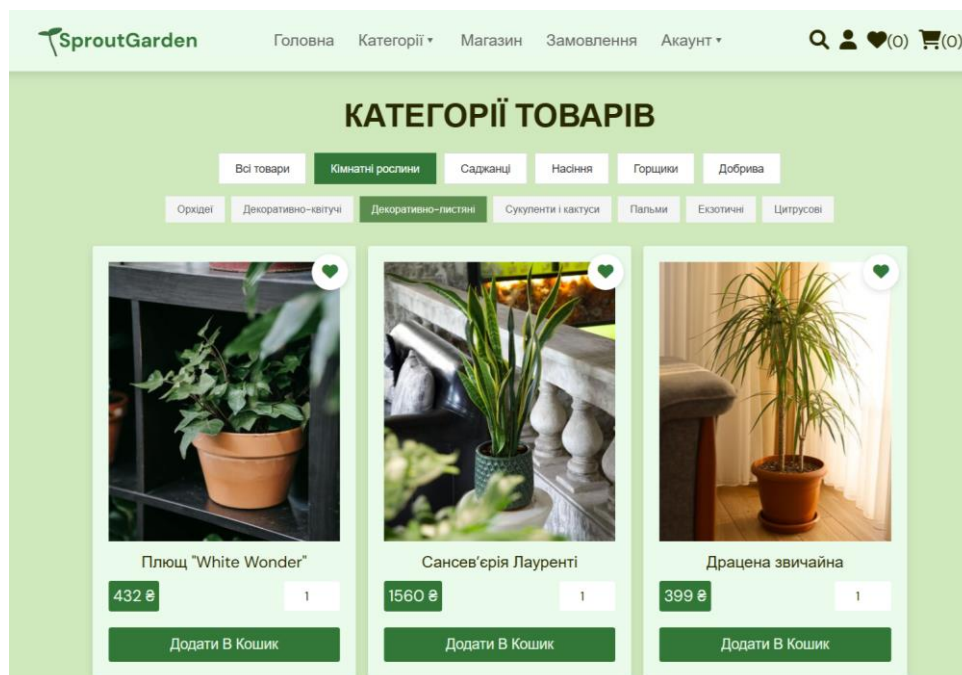


Рисунок 2.6 – Макет сторінки каталогу товарів

На рисунку 2.7 показано фрагмент макету сторінки перегляду товару, де користувач може ознайомитися з детальною інформацією про продукт.

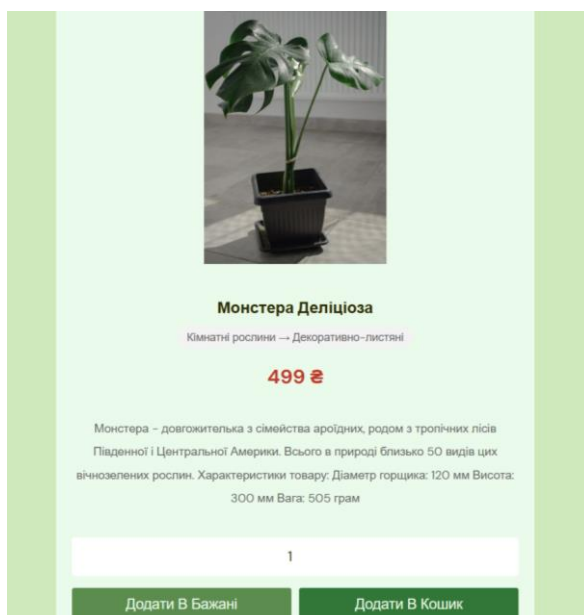


Рисунок 2.7 – Макет сторінки перегляду товару

На рисунку 2.8 зображено макет сторінки кошика, яка містить список обраних товарів, можливість зміни кількості та підсумкову вартість замовлення.

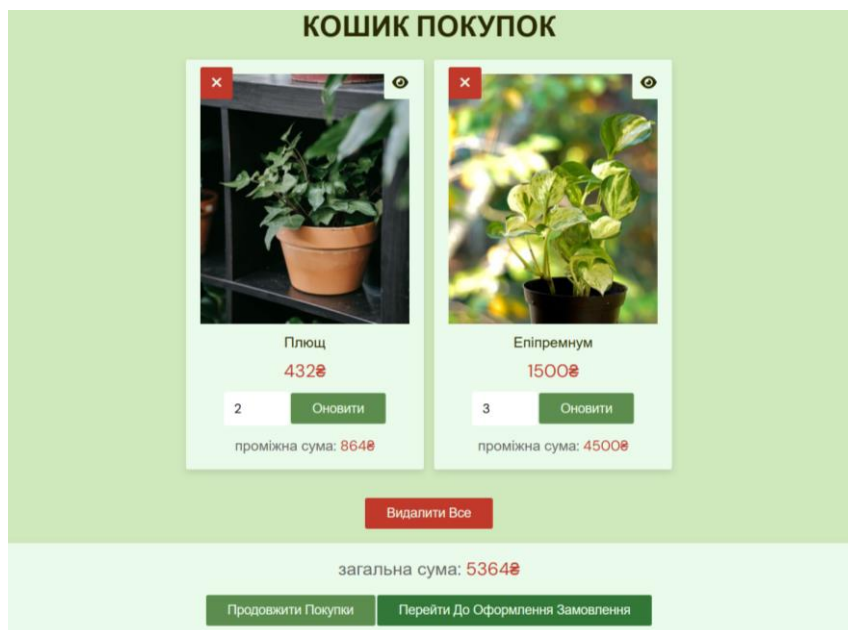


Рисунок 2.8 – Макет сторінки кошика

На рисунку 2.9 представлено макет сторінки оформлення замовлення, який включає форму для введення персональних даних, адреси доставки та вибору способу оплати.

Плющ (432€ x 2) Епіпремнум (1500€ x 3)

загальна сума: 5364€

Оформити Замовлення

ваше ім'я та прізвище:

ваш номер телефону:

електронна пошта:

метод оплати:

1-й рядок адреси:

2-й рядок адреси:

місто:

область:

країна:

поштовий індекс:

Рисунок 2.9 – Макет сторінки оформлення замовлення

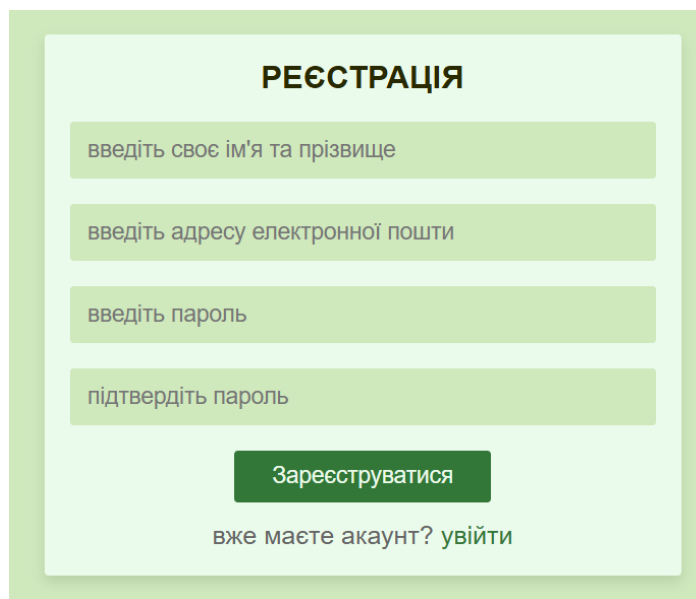
На рисунку 2.10 показано макет сторінки авторизації користувача.

ВХІД

[не маєте акаунту? зареєструватися](#)

Рисунок 2.10 – Макет сторінки входу

На рисунку 2.11 показано макет сторінки реєстрації користувача.



The registration form is titled "РЕЄСТРАЦІЯ" and is set against a light green background. It contains four input fields: "введіть своє ім'я та прізвище", "введіть адресу електронної пошти", "введіть пароль", and "підтвердіть пароль". Below the fields is a dark green "Зареєструватися" button and a link "вже маєте акаунт? увійти".

Рисунок 2.11 – Макет сторінки реєстрації

На рисунку 2.12 показано фрагмент макету сторінки «Про нас».

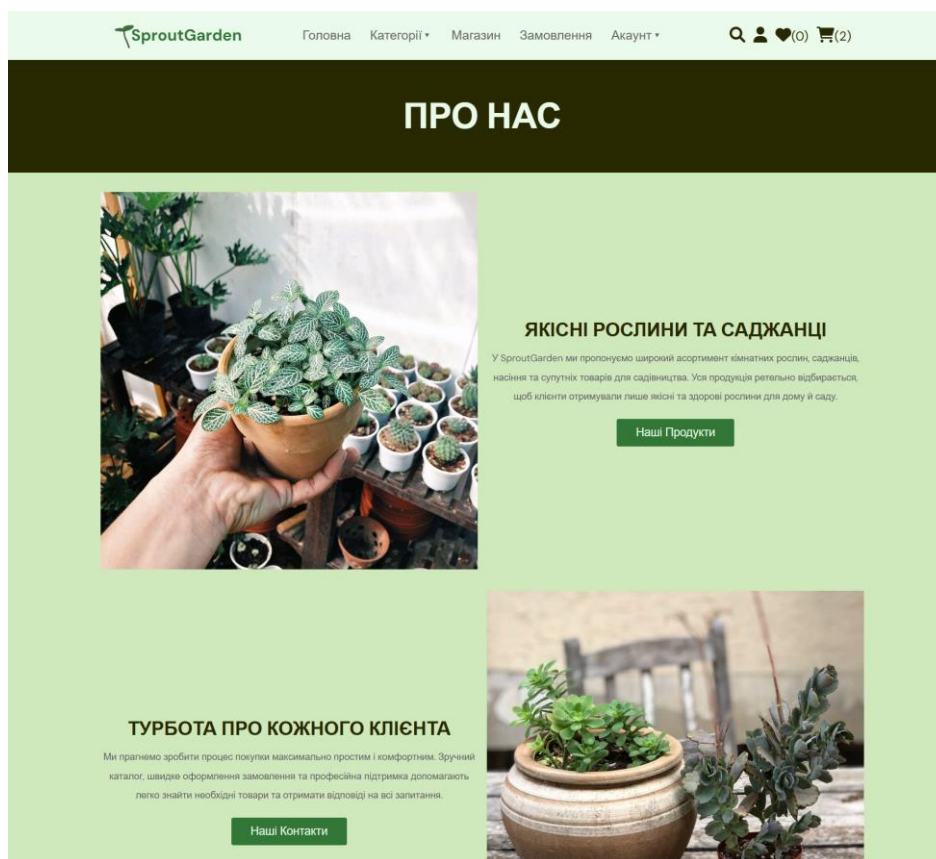


Рисунок 2.12 – Макет сторінки «Про нас»

На рисунку 2.13 показано фрагмент макету сторінки «Про нас» із відгуками користувачів, а також футером сайту.

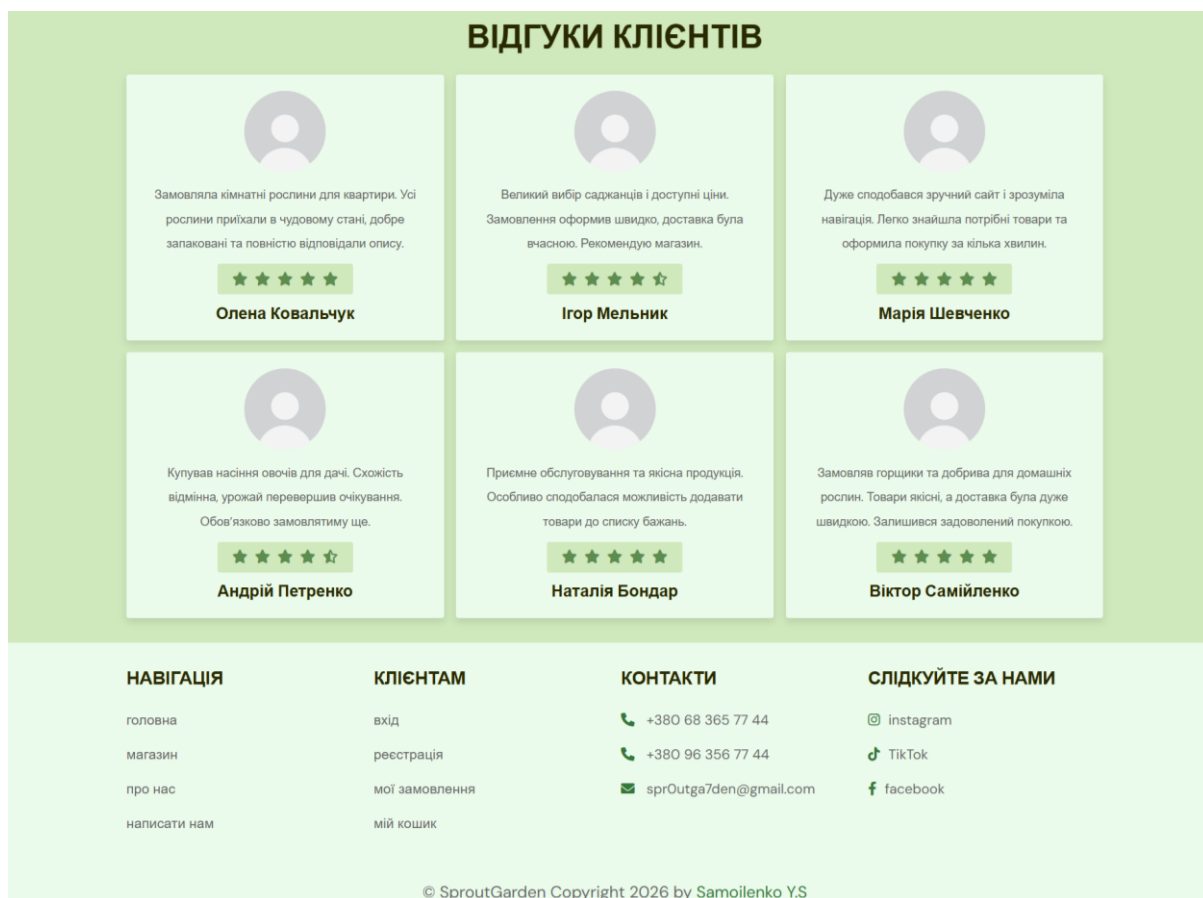


Рисунок 2.13 – Макет сторінки «Про нас» із відгуками і футером

Крім основних сторінок, у Figma за тим самим принципом і стилем було розроблено макети додаткових сторінок інтернет-магазину, зокрема сторінки бажаних товарів, історії замовлень, форму зворотного зв'язку та пошуку товарів.

2.5 Висновок до другого розділу

У другому розділі було виконано проєктування веб-магазину «SproutGarden» та визначено основні технологічні рішення для його реалізації. Проведено огляд веб-технологій, що використовуються при створенні веб-магазинів, зокрема HTML, CSS, JavaScript, PHP та системи керування базами

даних MySQL. Обґрунтовано вибір інструментів розробки, таких як Visual Studio Code та локальне серверне середовище XAMPP.

Також у межах розділу було визначено архітектуру веб-магазину. Виконано проектування бази даних, у якій містяться основні таблиці та зв'язки між ними для забезпечення коректного збереження та обробки інформації про користувачів, товари, кошик, замовлення та інші сутності.

Окрему увагу приділено формуванню фірмового стилю веб-магазину, розробці логотипу, вибору кольорової палітри та шрифтів, що забезпечують цілісне візуальне сприйняття системи. Крім того, створено макети сторінок у середовищі Figma, які попередньо візуалізують вигляд майбутнього веб-магазину.

Отже, результати другого розділу створюють повну основу для подальшої реалізації веб-магазину «SproutGarden», забезпечуючи узгодженість між технічними рішеннями, структурою системи та її користувацьким інтерфейсом.

РОЗДІЛ 3. РОЗРОБКА ТА ТЕСТУВАННЯ ВЕБ-МАГАЗИНУ «SPROUTGARDEN»

3.1 Реалізація клієнтської частини

Клієнтська частина веб-магазину відповідає за взаємодію користувача з системою та забезпечує відображення всіх елементів інтерфейсу у браузері. Саме вона формує зовнішній вигляд веб-магазину та визначає зручність його використання. Реалізація клієнтської частини включає створення структури сторінок, їх стилізацію та забезпечення базової інтерактивності.

У межах даного підрозділу розглядаються основні технології та підходи, що були використані для побудови користувацького інтерфейсу веб-магазину «SproutGarden».

3.1.1 HTML-структури сторінок

Розробка користувацької частини веб-магазину «SproutGarden» виконувалася за допомогою мови розмітки HTML5. Використання HTML дозволило сформувати структуру веб-сторінок, організувати навігацію між розділами системи та забезпечити відображення елементів користувацького інтерфейсу.

Веб-магазин складається з набору взаємопов'язаних сторінок, кожна з яких реалізує окремий функціональний модуль. Основними сторінками є:

- home.php – головна сторінка веб-магазину;
- shop.php – каталог товарів;
- view_page.php – детальна інформація про товар;
- cart.php – кошик покупок;
- wishlist.php – список бажань;
- orders.php – перегляд замовлень користувача;
- contact.php – форма зворотного зв'язку;

– login.php та register.php – авторизація і реєстрація користувачів.

Для підвищення зручності підтримки програмного коду застосовано модульний підхід до побудови інтерфейсу. Повторювані елементи оформлення винесені в окремі файли header.php та footer.php, які підключаються до сторінок за допомогою PHP-інструкції include.

Шапка сайту містить логотип веб-магазину, головне меню навігації, перелік категорій товарів, елементи пошуку, кнопки доступу до кошика та списку бажань, а також інформацію про обліковий запис користувача.

Реалізовано багаторівневе меню категорій товарів, завдяки якому користувач може перейти до певної категорії або підкатегорії продукції безпосередньо з навігаційного меню. У лістингу 3.1 наведено фрагмент коду меню категорій товарів.

Лістинг 3.1 – Фрагмент меню категорій товарів

```
<a href="#">Категорії▼</a><ul><li>
  <a href="shop.php?category=Кімнатні рослини">
    Кімнатні рослини▶</a><ul>
<li><a href="shop.php?subcategory=Орхідеї">Орхідеї</a></li>
  <li><a href="shop.php?subcategory=Декоративно-
квітучі">Декоративно-квітучі</a></li>
  <li><a href="shop.php?subcategory=Декоративно-
листяні">Декоративно-листяні</a></li>
  <li><a href="shop.php?subcategory=Сукуленти та
кактуси">Сукуленти та кактуси</a></li>
  <li><a
href="shop.php?subcategory=Пальми">Пальми</a></li>
  <li><a
href="shop.php?subcategory=Екзотичні">Екзотичні</a></li>
  <li><a
href="shop.php?subcategory=Цитрусові">Цитрусові</a></li>
</ul></li>
```

Головна сторінка веб-магазину складається з декількох блоків. Перший блок містить рекламний банер із коротким описом магазину та кнопкою переходу до каталогу товарів. У лістингу 3.2 наведено фрагмент HTML-коду реалізації головного банера.

Лістинг 3.2 – Фрагмент HTML-коду головного банера

```
<section class="home">
  <div class="content">
    <h3>Щастя проростає зсередини</h3>
    <p>
      Великий вибір рослин та товарів для
      садівництва за доступними цінами.
    </p>
    <a href="about.php" class="btn">
      дослідити каталог
    </a>
  </div>
</section>
```

Для представлення товарів використовується блочна структура у вигляді карток. Кожна картка містить фотографію товару, його назву, ціну, поле вибору кількості та кнопки додавання до кошика і списку бажань. У лістингу 3.3 наведено фрагмент HTML-коду реалізації картки товару.

Лістинг 3.3 – Фрагмент HTML-коду картки товару

```
<form action="" method="POST" class="box">
  
  <div class="name">
    Назва товару
  </div>
  <div class="price">
    450 ₪
  </div>
  <input type="number"
    name="product_quantity"
    value="1"
    min="1"
    class="qty">
  <input type="submit"
    value="Додати в кошик"
    class="btn">
</form>
```

На сторінці каталогу реалізовано механізм фільтрації товарів за категоріями та підкатегоріями. Для цього використовується набір посилань, що

передають параметри через URL-запит. У лістингу 3.4 наведено фрагмент HTML-коду фільтрації товарів.

Лістинг 3.4 – Фрагмент HTML-коду фільтрації товарів

```
<div class="catalog-top">
  <a href="shop.php">
    Всі товари
  </a>
  <a href="shop.php?category=Кімнатні рослини">
    Кімнатні рослини
  </a>
  <a href="shop.php?category=Саджанці">
    Саджанці
  </a>
</div>
```

Нижня частина веб-магазину реалізована у файлі footer.php та містить додаткові навігаційні посилання, контактні дані, інформацію для клієнтів та посилання на сторінки соціальних мереж. У лістингу 3.5 наведено фрагмент HTML-коду для реалізації футеру веб-магазину «SproutGarden».

Лістинг 3.5 – Фрагмент HTML-коду футеру

```
<section class="footer">
  <div class="box-container">
    <div class="box">
      <h3>навігація</h3>
      <a href="home.php">головна</a>
      <a href="shop.php">магазин</a>
      <a href="about.php">про нас</a>
      <a href="contact.php">написати нам</a>
    </div>
  </section>
```

Таким чином, використання HTML5 дозволило створити структурований та зручний користувацький інтерфейс веб-магазину «SproutGarden». Реалізовано структуру сторінок, яка забезпечує швидкий доступ до функціональних можливостей системи.

3.1.2 Створення CSS-стилів

Для забезпечення сучасного та зручного інтерфейсу веб-магазину «SproutGarden» було використано каскадні таблиці стилів CSS3. За допомогою CSS реалізовано оформлення сторінок, налаштування кольорової гами, розміщення елементів та адаптивність інтерфейсу.

Під час розробки було використано світло-зелено кольорову гаму. Для спрощення підтримки стилів використовувалися CSS-змінні, які містять основні кольори інтерфейсу, тіні та службові параметри. Лістинг 3.6 демонструє оголошення глобальних змінних стилів.

Лістинг 3.6 – Визначення кольорової палітри веб-магазину

```
:root{
  --green1:#337738;
  --red:#c0392b;
  --green2:#5c8d4f;
  --black:#282900;
  --white:#ebfbeb;
  --light-color:#666;
  --light-white:#ccc;
  --light-bg:#cfe8bc;
  --box-shadow:0 .5rem 1rem rgba(0,0,0,.1);
}
```

Для забезпечення єдиного стилю оформлення всіх сторінок було використано глобальне скидання відступів та налаштування шрифту DM Sans, який підключається через сервіс Google Fonts. У лістингу 3.7 наведено глобальні налаштування сторінок.

Лістинг 3.7 – Глобальні налаштування сторінок

```
*{
  font-family: 'DM Sans', sans-serif;
  margin:0; padding:0;
  box-sizing: border-box;
  outline: none; border:none;
  text-decoration: none;
  transition:all .2s linear;
}
```

```

html{
  font-size: 62.5%;
  overflow-x: hidden;
  scroll-behavior: smooth;
  scroll-padding-top: 6.5rem;
}

body{
  background: var(--light-bg);
}

```

Для побудови структури сторінок використовувався модуль Flexbox. Зокрема, він застосовується для вирівнювання елементів шапки сайту. У лістингу 3.8 наведено фрагмент стилізації шапки веб-магазину.

Лістинг 3.8 – Стилізація шапки веб-магазину

```

.header .flex{
  display: flex;
  align-items: center;
  justify-content: space-between;
  padding: 2rem;
  max-width: 1200px;
  margin: 0 auto;
  position: relative;
}

```

Для реалізації багаторівневого випадаючого меню категорій товарів було використано вкладені списки та псевдокласи наведення курсора. У лістингу 3.9 наведено фрагмент стилів випадаючого меню.

Лістинг 3.9 – Стилів випадаючого меню

```

.header .flex .navbar ul li ul{
  position: absolute;
  left: 0;
  width: 15rem;
  background-color: var(--white);
  animation: fadeIn .2s linear;
  display: none;
}

.header .flex .navbar ul li a:hover{
  background-color: var(--green1);
  color: var(--white);
}

```

Каталог товарів реалізовано у вигляді адаптивної сітки. Для цього застосовано модуль CSS Grid Layout, який автоматично розміщує картки товарів залежно від ширини екрана користувача. Лістинг 3.10 демонструє стилі для сітки каталогу товарів.

Лістинг 3.10 – Сітка каталогу товарів

```
.products .box-container{
  max-width: 1200px;
  margin:0 auto;
  display: grid;
  grid-template-columns: repeat(auto-fit, 33rem);
  gap:1.5rem;
  align-items: flex-start;
  justify-content: center;
}
```

Кожен товар відображається у вигляді окремої картки з фотографією, назвою, ціною та елементами керування. Для створення візуального ефекту використовуються тіні та заокруглення кутів, приклад яких подано у лістингу 3.11.

Лістинг 3.11 – Оформлення картки товару

```
.products .box-container .box{
  background-color: var(--white);
  border:var(--border);
  box-shadow: var(--box-shadow);
  border-radius: .5rem;
  padding:2rem;
  text-align: center;
  position: relative;
}
```

Для кнопок було створено окремі стилі, що забезпечують однаковий зовнішній вигляд на всіх сторінках системи. При наведенні курсора колір кнопок змінюється.

Особливу увагу приділено оформленню головної сторінки веб-магазину. Для першого екрана використано фонове зображення та ефект світіння тексту.

Реалізовано плавні переходи між станами елементів, стилізацію смуги прокручування та анімацію появи окремих блоків інтерфейсу.

Стилізація була реалізована для всіх сторінок веб-магазину, включаючи сторінки каталогу товарів, товарних позицій, кошика, оформлення замовлення, авторизації та реєстрації. Було застосовано єдині принципи побудови інтерфейсу, кольорову палітру, шрифтове оформлення та компоненти.

Таким чином, використання CSS3 дозволило створити єдиний візуальний стиль веб-магазину «SproutGarden», забезпечивши комфортну взаємодію користувачів із системою.

3.1.3 Реалізація клієнтської логіки засобами JavaScript

Для реалізації інтерактивної поведінки веб-магазину «SproutGarden» було використано мову програмування JavaScript. Основним завданням клієнтської логіки є забезпечення динамічної взаємодії користувача з інтерфейсом без перезавантаження сторінки.

Зокрема, за допомогою JavaScript реалізовано керування відображенням елементів навігації та користувацького меню. У лістингу 3.12 наведено фрагмент коду JavaScript, що відповідає за перемикання видимості меню користувача та навігаційної панелі.

Лістинг 3.12 – Скрипт керування меню та панеллю користувача

```
let userBox = document.querySelector('.header .flex .account-box');
document.querySelector('#user-btn').onclick = () =>{
    userBox.classList.toggle('active');
    navbar.classList.remove('active');
}
let navbar = document.querySelector('.header .flex .navbar');
document.querySelector('#menu-btn').onclick = () =>{
    navbar.classList.toggle('active');
    userBox.classList.remove('active');
}
window.onscroll = () =>{
    userBox.classList.remove('active');
    navbar.classList.remove('active');}
```

У наведеному кодї реалізовано перемикання класу `active` для елементів меню користувача та навігаційної панелі. При натисканні на кнопку користувача відкривається відповідний блок інформації, при цьому навігаційне меню автоматично приховується, і навпаки. Це дозволяє уникнути накладання елементів інтерфейсу.

Додатково, при прокручуванні сторінки (`window.onscroll`) всі випадаючі меню автоматично закриваються, що забезпечує більш зручну та передбачувану поведінку інтерфейсу для користувача.

Таким чином, використання JavaScript дозволило реалізувати базову інтерактивність веб-магазину та покращити загальний користувацький досвід.

3.2 Реалізація серверної частини веб-магазину

Серверна частина веб-магазину «SproutGarden» реалізована із використанням мови програмування PHP та системи керування базами даних MySQL. Основним завданням серверної логіки є обробка запитів користувача, взаємодія з базою даних, управління товарами, а також забезпечення функціонування механізмів реєстрації, авторизації та роботи з кошиком.

Архітектура серверної частини побудована за модульним принципом, що дозволяє розділити функціональні блоки системи на окремі компоненти.

Основна логіка взаємодії користувача із системою реалізується через обробку HTTP-запитів, які надходять від клієнтської частини. PHP-скрипти виконують перевірку отриманих даних, формують SQL-запити до бази даних та повертають відповідні результати для відображення на сторінці.

3.2.1 Реалізація бізнес-логіки засобами PHP

Бізнес-логіка веб-магазину «SproutGarden» реалізована за допомогою мови програмування PHP та побудована на обробці HTTP-запитів користувача, взаємодії з базою даних MySQL та формуванні динамічного контенту сторінок.

Основними функціональними модулями є авторизація користувачів, робота з товарами, управління списком бажань (wishlist), кошиком (cart) та оформлення замовлення.

Одним із ключових елементів системи є механізм автентифікації користувачів. Реєстрація та вхід у систему реалізовані через сторінку login.php, яка здійснює перевірку введених даних користувача та визначає його роль у системі. Після надсилання форми вхідні дані проходять фільтрацію та шифрування пароля за допомогою функції md5. Далі виконується запит до бази даних для перевірки відповідності користувача. У разі успішної авторизації користувачу створюється сесія та здійснюється перенаправлення залежно від його ролі (адміністратор або звичайний користувач).

Окремим елементом бізнес-логіки є механізм виходу з системи, який реалізовано у файлі logout.php. Його основна задача полягає у завершенні сесії користувача та очищенні всіх сесійних даних.

Крім механізмів авторизації, важливу роль відіграє сторінка shop.php, яка відповідає за відображення каталогу товарів та взаємодію користувача з ними. На цій сторінці реалізовано додавання товарів до списку бажань і кошика, а також перевірку на дублювання записів у базі даних.

Логіка додавання товарів до кошика аналогічно передбачає перевірку на дублювання та можливість перенесення товарів зі списку бажань у кошик.

Окремим функціональним модулем є сторінка оформлення замовлення (checkout.php). Вона відповідає за формування замовлення на основі товарів, що знаходяться у кошику користувача. У лістингу 3.13 наведено фрагмент коду, що реалізує підрахунок вартості товарів у кошику та формування списку замовлення.

Лістинг 3.13 – Формування та збереження замовлення (checkout.php)

```
$cart_total = 0;
    $cart_products[] = '';
    $cart_query = mysqli_query($conn, "SELECT * FROM `cart` WHERE
user_id = '$user_id'") or die('query failed');
    if(mysqli_num_rows($cart_query) > 0){
```

```

        while($cart_item = mysqli_fetch_assoc($cart_query)){
            $cart_products[] = $cart_item['name'].'
('.$cart_item['quantity'].') ';
            $sub_total = ($cart_item['price'] *
$cart_item['quantity']);
            $cart_total += $sub_total;
        }
    }
    $total_products = implode(', ', $cart_products);

```

Після формування даних виконується перевірка на порожній кошик та дублювання замовлення. У разі успішної перевірки інформація записується до таблиці orders, а вміст кошика очищується.

Таким чином, було реалізовано бізнес-логіку всього веб-магазину, яка охоплює повний цикл взаємодії користувача із системою – від автентифікації та перегляду товарів до оформлення та завершення замовлення. Фрагменти програмного коду клієнтської частини веб-магазину «SproutGarden» наведені у додатку А.

3.2.2 Взаємодія з базою даних MySQL

Веб-магазин «SproutGarden» використовує реляційну базу даних MySQL для зберігання та обробки всієї інформації системи. Взаємодія з базою даних здійснюється за допомогою PHP та розширення MySQLi, що забезпечує виконання SQL-запитів і отримання результатів у динамічному режимі.

Структура бази даних спроектована таким чином, щоб забезпечити цілісність даних та підтримку основних функціональних модулів веб-додатку. Зв'язки між таблицями реалізуються через поле user_id, яке забезпечує зв'язок між користувачем та його діями в системі. Аналогічно поле rid використовується для зв'язку товарів із кошиком та списком бажань.

Взаємодія серверної частини веб-магазину «SproutGarden» з базою даних MySQL реалізована за допомогою розширення MySQLi у процедурному стилі. Підключення до бази даних здійснюється через окремий конфігураційний файл config.php, який підключається у всіх основних модулях системи за допомогою

директиви `@include`. Це забезпечує централізоване керування параметрами підключення до бази даних та дозволяє уникнути дублювання коду. Усі SQL-запити виконуються через глобальний об'єкт підключення `$conn`, який ініціалізується один раз і використовується повторно протягом усього життєвого циклу роботи веб-магазину.

У всіх ключових модулях системи – каталозі товарів (`shop.php`), кошику (`cart`), списку бажань (`wishlist`), оформленні замовлення (`checkout.php`) та авторизації користувачів (`login.php`) – реалізовано взаємодію з базою даних через SQL-запити типу `SELECT`, `INSERT`, `DELETE`.

Для підвищення безпеки та коректності даних використовується функція `mysqli_real_escape_string()`, яка захищає від SQL-ін'єкцій шляхом екранування вхідних даних користувача. Додатково застосовується перевірка існування записів перед вставкою нових даних, що дозволяє уникнути дублювання інформації у таблицях. У лістингу 3.14 наведено фрагмент коду, що реалізує підключення до бази даних MySQL та встановлення з'єднання між веб-застосунком і сервером бази даних.

Лістинг 3.14 – Приклад підключення до бази даних (`config.php`)

```
<?php
$conn = mysqli_connect('localhost','root','','shop_db') or
die('connection failed');
?>
```

Особливу роль відіграє сторінка `checkout.php`, де відбувається підсумовування вартості товарів у кошику та формування запису замовлення у таблиці `orders`. Після успішного оформлення замовлення вміст кошика користувача автоматично видаляється, що забезпечує актуальність даних.

Таким чином, взаємодія з базою даних MySQL у веб-магазині «SproutGarden» була побудована на використанні стандартних SQL-операцій через PHP, що забезпечує стабільну роботу системи та ефективне управління даними користувачів і товарів.

3.2.3 Реалізація адміністративної панелі

Адміністративна панель – це спеціальний інтерфейс, який дозволяє керувати вмістом сайту, мобільного додатку чи будь-якої іншої цифрової платформи без необхідності втручання в програмний код [24]. Для забезпечення ефективного керування веб-магазином «SproutGarden» було реалізовано адміністративну панель, яка надає адміністратору доступ до основних функцій системи.

Після авторизації адміністратор отримує доступ до панелі керування, яка містить статистичну інформацію про роботу веб-магазину та інструменти для управління його вмістом.

На рисунку 3.1 представлено головну сторінку адміністративної панелі.

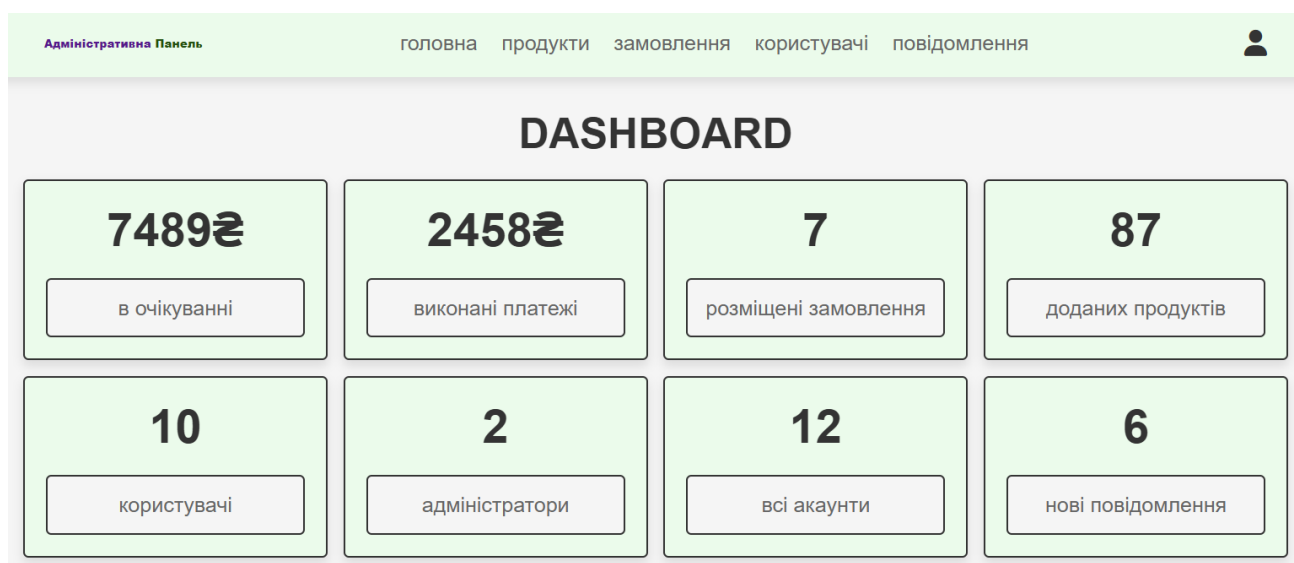


Рисунок 3.1 – Головна сторінка адміністративної панелі

На головній сторінці відображається узагальнена статистика роботи системи. Адміністратор може переглядати загальну суму виконаних та невиконаних замовлень, кількість зареєстрованих користувачів та адміністраторів, кількість товарів у каталозі, кількість замовлень і повідомлень, отриманих від клієнтів. Такий підхід дозволяє швидко оцінити поточний стан роботи веб-магазину та оперативно реагувати на зміни.

Однією з ключових функцій адміністративної панелі є управління товарами. Для цього було реалізовано окремий модуль, що забезпечує виконання основних операцій CRUD (Create, Read, Update, Delete) [25].

На рисунку 3.2 наведено фрагменти сторінки керування товарами.

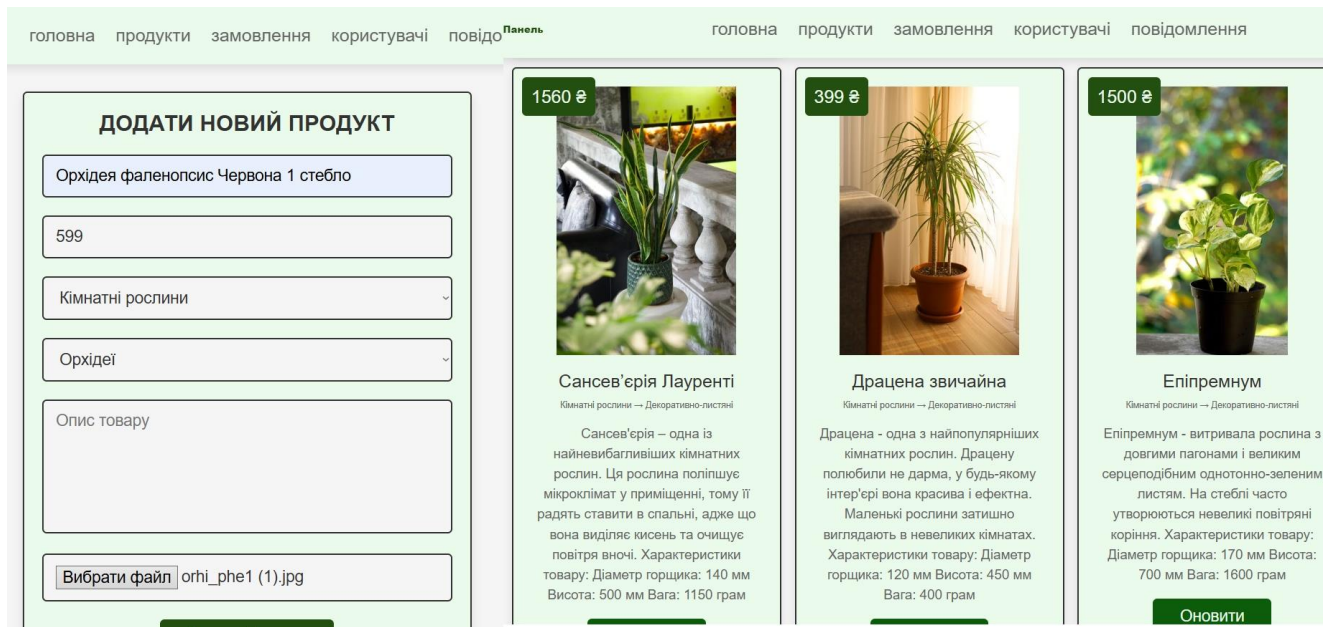


Рисунок 3.2 – Сторінка управління товарами

Даний модуль дозволяє додавати нові товари до каталогу, редагувати інформацію про існуючі товари, оновлювати їх характеристики та видаляти застарілі позиції. Для кожного товару зберігається назва, ціна, опис та фотографія. Усі зміни автоматично записуються до бази даних і відображаються на користувацькій частині сайту.

Для контролю процесу обробки замовлень реалізовано окрему сторінку керування замовленнями, яка зображена на рисунку 3.3.

На цій сторінці адміністратор може переглядати інформацію про всі оформлені замовлення. Крім перегляду інформації передбачено можливість зміни статусу замовлення. Замовлення можуть перебувати у стані «В обробці» або «Виконано», що дозволяє контролювати етапи їх виконання.

Адміністративна Панель головна продукти замовлення користувачі повідомлення

ЗАМОВЛЕННЯ

<p>id користувача: 16 дата замовлення: 17 травня 2026 р. ім'я: Yulia Samoilenko номер телефону: 380688888888 електронна пошта: samojlenkoulia327@gmail.com адреса: вул. Хрещатик, 36, Київ, Україна - 01001 товари: Сансев'єрія Лауренті (1), Каланхое Біле (2), Горщик для орхідей прозорий (1) сума: 2458€ спосіб оплати: Післяплата</p> <p>Виконано</p> <p>Оновити Видалити</p>	<p>id користувача: 17 дата замовлення: 18 травня 2026 р. ім'я: Василь Василенко номер телефону: 3024164849 електронна пошта: user01@gmail.com адреса: вул. Зелена, 37, Чернівці, Україна - 19701 товари: Орхідея Фаленопсис Фіолет (4), Плющ (4) сума: 4608€ спосіб оплати: Післяплата</p> <p>В обробці</p> <p>Оновити Видалити</p>	<p>id користувача: 24 дата замовлення: 18 травня 2026 р. ім'я: Марія Козенко номер телефону: 380987777788 електронна пошта: marinak@gmail.com адреса: Проспект Олександра Поля, 110, Дніпро, Україна - 12345 товари: Кактус Опунція Мікродазис (2) сума: 600€ спосіб оплати: Банківська картка</p> <p>В обробці</p> <p>Оновити Видалити</p>
--	---	---

Рисунок 3.3 – Сторінка керування замовленнями

Для адміністрування облікових записів реалізовано модуль керування користувачами, який представлено на рисунку 3.4.

АКАУНТИ КОРИСТУВАЧІВ

<p>id користувача: 10 ім'я користувача: admin A електронна пошта: admin01@gmail.com Тип акаунту: Адміністратор</p> <p>Видалити</p>	<p>id користувача: 16 ім'я користувача: Самойленко Юлія електронна пошта: samojlenkoulia327@gmail.com Тип акаунту: Клієнт</p> <p>Видалити</p>	<p>id користувача: 17 ім'я користувача: Василь Висиленко електронна пошта: user01@gmail.com Тип акаунту: Клієнт</p> <p>Видалити</p>
<p>id користувача: 18 ім'я користувача: Марія Дубенко електронна пошта: user02@gmail.com Тип акаунту: Клієнт</p> <p>Видалити</p>	<p>id користувача: 19 ім'я користувача: Іванна Іваненко електронна пошта: user03@gmail.com Тип акаунту: Клієнт</p> <p>Видалити</p>	<p>id користувача: 20 ім'я користувача: Сергій Бобер електронна пошта: user04@gmail.com Тип акаунту: Клієнт</p> <p>Видалити</p>

Рисунок 3.4 – Сторінка керування користувачами

Даний розділ дозволяє переглядати список зареєстрованих користувачів та адміністраторів системи. Адміністратор може контролювати наявні облікові записи та за необхідності видаляти користувачів із системи.

На рисунку 3.5 наведено модуль роботи з повідомленнями для організації зворотного зв'язку між клієнтами та адміністрацією веб-магазину.

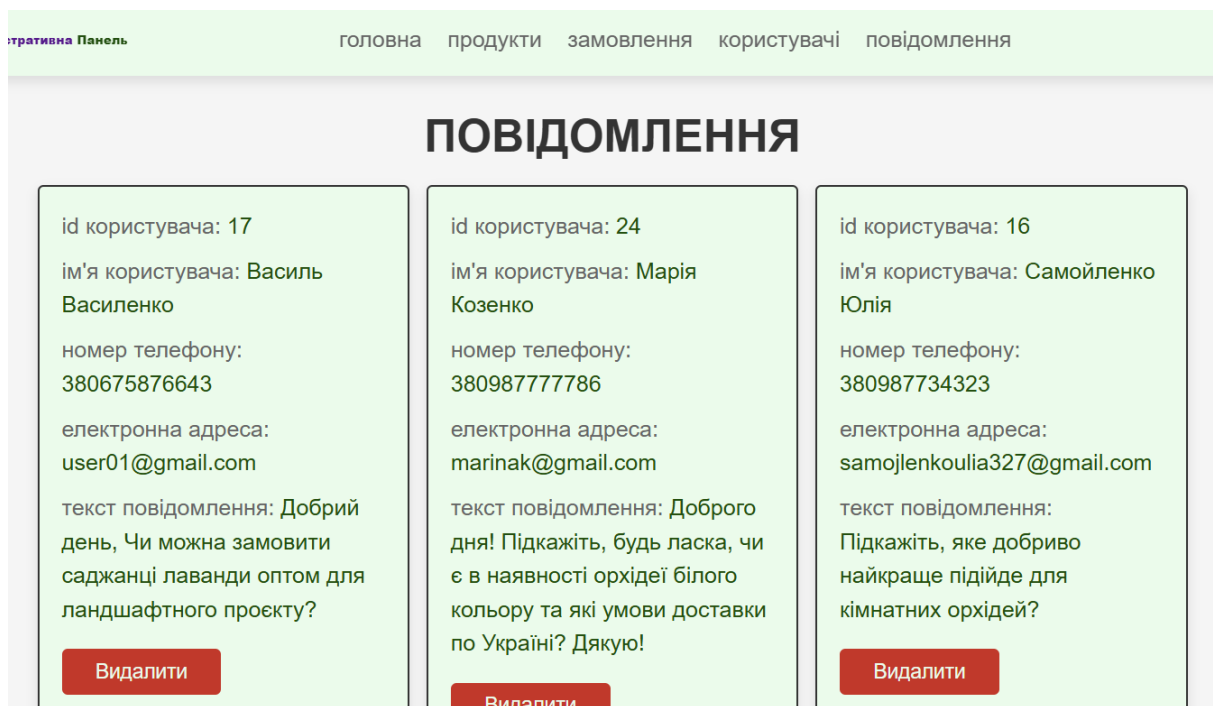


Рисунок 3.5 – Сторінка повідомлень від клієнтів

На даній сторінці відображаються повідомлення, які користувачі надсилають через форму зворотного зв'язку. Після опрацювання повідомлення може бути видалене із системи. Це дозволяє підтримувати комунікацію з клієнтами та своєчасно реагувати на їхні звернення.

Фрагменти програмного коду адміністративної панелі подано у додатку Б. Таким чином, реалізована адміністративна панель забезпечує повний контроль над функціонуванням веб-магазину «SproutGarden». Вона дозволяє керувати товарами, замовленнями, користувачами та повідомленнями клієнтів, а також отримувати актуальну статистичну інформацію про діяльність системи.

3.3 Тестування та перевірка працездатності веб-магазину

Тестування є важливим етапом розробки веб-магазину, який дає можливість перевірити правильність роботи всіх компонентів, виявити можливі помилки та забезпечити стабільне функціонування програмного продукту. Для веб-магазину «SproutGarden» було проведено комплексне тестування клієнтської та серверної частин системи, а також перевірку взаємодії з базою даних.

Під час тестування було перевірено коректність роботи основного функціоналу веб-магазину. Зокрема, виконувалася перевірка процесів реєстрації та авторизації користувачів, перегляду каталогу товарів, роботи меню категорій, додавання товарів до кошика, оформлення замовлення та надсилання повідомлень через форму зворотного зв'язку.

Окрему увагу було приділено тестуванню адміністративної панелі. Перевірялася коректність виконання CRUD-операцій, зміна статусів замовлень, управління обліковими записами користувачів, перегляд повідомлень клієнтів та формування даних на головній сторінці адміністратора.

У процесі розробки було виявлено декілька помилок, які могли впливати на працездатність системи. Однією з них стала проблема некоректного відображення випадаючого меню категорій товарів. Під час наведення курсора на окремі пункти меню вкладені категорії перекривалися іншими елементами сторінки та не відображалися повністю. Для усунення цієї помилки було скориговано значення властивостей у CSS-стилях.

Також було виявлено помилку під час додавання товарів до кошика. Користувач міг додати один і той самий товар кілька разів, що призводило до дублювання записів у таблиці кошика. Для вирішення проблеми було реалізовано перевірку наявності товару перед створенням нового запису та оновлення кількості вже доданого товару.

Для налагодження веб-магазину використовувалися інструменти розробника браузера, журнал помилок PHP та засоби адміністрування бази

даних MySQL. Це дозволило оперативно виявляти причини помилок та перевіряти правильність обміну даними між сервером і клієнтською частиною застосунку.

Також було проведено тестування адаптивності інтерфейсу на різних розмірах екранів. Перевірялася коректність відображення сторінок на персональних комп'ютерах, планшетах та мобільних пристроях. Результати тестування показали, що елементи інтерфейсу автоматично підлаштовуються під розміри екрана та зберігають зручність використання.

За результатами проведеного тестування було підтверджено коректну роботу функціональних можливостей веб-магазину «SproutGarden». Усі виявлені помилки були усунуті, що забезпечило стабільну роботу системи.

3.4 Висновок до третього розділу

У третьому розділі було розглянуто процес програмної реалізації веб-магазину «SproutGarden».

У ході розробки було створено інтерфейс із використанням технологій HTML5, CSS3 та JavaScript. Для зберігання та обробки даних використано систему керування базами даних MySQL, а серверна логіка реалізована засобами PHP.

Особливу увагу приділено створенню адміністративної панелі, яка надає можливість керувати товарами, замовленнями, користувачами та повідомленнями клієнтів.

Також проведено комплексне тестування системи, під час якого перевірено працездатність основних функцій, виявлено та усунуто програмні помилки.

Таким чином, у результаті виконання третього розділу було реалізовано повнофункціональний веб-магазин «SproutGarden», який забезпечує зручний процес перегляду та придбання товарів для користувачів.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Ергономічні проблеми безпеки життєдіяльності

Під час роботи з персональним комп'ютером важливого значення набувають питання безпеки життєдіяльності користувача. Тривала робота за комп'ютером супроводжується значним навантаженням на органи зору, нервову систему та опорно-руховий апарат, що може призводити до зниження працездатності, втоми та погіршення самопочуття [26].

Однією з основних ергономічних проблем є статичне навантаження на організм під час тривалого перебування в сидячому положенні. Неправильна поза під час роботи спричиняє перенапруження м'язів шиї, плечового пояса та спини. Наслідком цього можуть бути порушення постави, болі в поперековому відділі хребта та розвиток захворювань опорно-рухової системи [28].

Важливим фактором безпеки життєдіяльності є дотримання режиму праці та відпочинку. Згідно з трудовим законодавством України нормальна тривалість робочого часу не повинна перевищувати 40 годин на тиждень, а працівникам мають надаватися перерви для відпочинку та харчування [26, 30].

Суттєвий вплив на безпеку діяльності людини мають її соціально-психологічні якості. Причинами багатьох нещасних випадків є недисциплінованість, недостатня увага до вимог безпеки, переоцінка власних можливостей та схильність до невиправданого ризику [27].

Окремою проблемою є психоемоційне навантаження, яке виникає під час виконання складних завдань програмування та проектування інформаційних систем. Зі зростанням емоційного напруження спочатку спостерігається підвищення працездатності людини, однак після досягнення певного рівня напруги продуктивність починає знижуватися, збільшується кількість помилок та погіршується якість прийняття рішень [27]. Тому необхідно уникати перевтоми та надмірного стресу під час роботи.

Негативно впливають на безпеку праці також недостатня фізична активність, нерегулярний режим роботи та тривале використання цифрових пристроїв без перерв. Для зниження навантаження на зоровий аналізатор рекомендується періодично змінювати напрямок погляду, виконувати вправи для очей та забезпечувати достатній рівень освітлення робочого місця [28, 29].

Отже, основними ергономічними проблемами безпеки життєдіяльності під час роботи над веб-магазином є статичне навантаження на організм, психоемоційне перенапруження, зорова втома та недотримання режиму праці й відпочинку. Усунення зазначених факторів сприяє підвищенню працездатності та збереженню здоров'я користувачів і розробників програмного забезпечення.

4.2 Заходи, що покращують умови праці оператора

Для забезпечення безпечних і комфортних умов праці оператора персонального комп'ютера необхідно дотримуватися вимог нормативно-правових актів з охорони праці та ергономіки робочого місця.

Одним із найважливіших заходів є правильна організація робочого місця. Робоче місце для виконання робіт у положенні сидячи повинно забезпечувати природне положення тіла працівника та можливість зміни робочої пози протягом робочого дня [31]. Робочий стіл має забезпечувати достатню площу для розміщення монітора, клавіатури та інших засобів праці.

Для підтримання правильної постави необхідно використовувати ергономічне крісло з регулюванням висоти сидіння, нахилу спинки та підлокітників [32]. Використання ергономічних меблів дозволяє знизити навантаження на хребет та запобігти розвитку професійних захворювань.

Особливу увагу слід приділяти розташуванню монітора. Екран повинен знаходитися на відстані приблизно 50–70 см від очей користувача, а верхня межа монітора має розташовуватися на рівні очей або трохи нижче. Таке розташування зменшує навантаження на зір і шийний відділ хребта.

Важливим фактором комфортної роботи є якісне освітлення. Освітлення робочих місць повинно забезпечувати достатню видимість інформації на екрані без виникнення відблисків та різких контрастів [33]. Рекомендується поєднання природного та штучного освітлення.

Для створення сприятливого виробничого середовища необхідно підтримувати нормативні параметри мікроклімату. Комфортний мікроклімат позитивно впливає на самопочуття працівників та їхню працездатність.

Під час роботи з комп'ютерною технікою необхідно також дотримуватися вимог щодо безпеки працівників під час використання екранних пристроїв [34].

Значну роль відіграє забезпечення електробезпеки. Комп'ютерне обладнання повинно бути справним, мати захисне заземлення та підключатися відповідно до вимог Правил улаштування електроустановок. Експлуатація несправної техніки може створювати небезпеку ураження електричним струмом.

Не менш важливим є дотримання правил пожежної безпеки. Робоче приміщення повинно бути забезпечене первинними засобами пожежогасіння, а працівники повинні знати порядок дій у разі виникнення пожежі відповідно до Правил пожежної безпеки в Україні [35].

Таким чином, покращення умов праці оператора досягається шляхом раціональної організації робочого місця, використання ергономічних меблів, забезпечення нормативного освітлення та мікроклімату, дотримання вимог електро- і пожежної безпеки, а також виконання нормативних вимог щодо роботи з екранними пристроями. Це сприяє підвищенню продуктивності праці, зменшенню професійних ризиків та збереженню здоров'я працівників.

4.3 Висновок до четвертого розділу

У четвертому розділі було розглянуто питання безпеки життєдіяльності та охорони праці під час роботи оператора персонального комп'ютера. Проведено аналіз основних ергономічних факторів, що впливають на безпеку та

працездатність користувача, зокрема особливостей організації робочого місця, режиму праці та відпочинку, впливу психоемоційного навантаження, освітлення та мікроклімату приміщення.

Встановлено, що дотримання ергономічних вимог і правил безпечної роботи з персональним комп'ютером сприяє зниженню втоми, зменшенню ризику професійних захворювань, підвищенню продуктивності праці та покращенню загального самопочуття працівника.

Також були розглянуті основні заходи щодо покращення умов праці оператора, серед яких правильна організація робочого місця, забезпечення нормативних параметрів освітлення та мікроклімату, дотримання вимог електро- та пожежної безпеки, а також використання ергономічних меблів і обладнання.

Такий підхід гарантує стабільну роботу над веб-магазином «SproutGarden» і може бути використаний для інших ІТ-проектів із тривалою роботою за комп'ютером.

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи освітнього рівня «Бакалавр» було розроблено веб-магазин «SproutGarden», призначений для продажу товарів для саду, городу, будинку та дачі в онлайн-середовищі.

У першому розділі кваліфікаційної роботи:

- подано аналіз предметної області;
- розглянуто існуючі веб-магазини та їх функціональні можливості;
- висвітлено основні вимоги до майбутнього веб-магазину;
- проаналізовано проблеми, які виникають у подібних веб-магазинах.

У другому розділі кваліфікаційної роботи:

- досліджено сучасні веб-технології для створення веб-магазинів;
- обґрунтовано вибір інструментів розробки;
- сформовано структуру бази даних для зберігання інформації про товари, користувачів і замовлення;

- спроектовано фірмовий стиль та макети інтерфейсу веб-магазину.

У третьому розділі кваліфікаційної роботи:

- розроблено клієнтську та серверну частини веб-магазину;
- реалізовано основну бізнес-логіку засобами PHP;
- запроваджено механізми роботи з базою даних MySQL;
- спроектовано та реалізовано функціонал кошика, списку бажань, оформлення замовлення та інших сторінок веб-магазину;
- протестовано працездатність веб-магазину.

У розділі «Безпека життєдіяльності, основи охорони праці» висвітлено основні ергономічні проблеми до організації робочого місця користувача та запропоновано заходи, що сприяють покращенню умов праці при роботі з комп'ютерною технікою.

Таким чином, розроблений веб-магазин «SproutGarden» повністю відповідає поставленій меті роботи та може бути використаний як готове рішення для онлайн-продажу товарів для саду, городу, будинку та дачі.

ПЕРЕЛІК ДЖЕРЕЛ

1. Білоус-Сергеєва, С., і А. Буцький. «Електронна комерція в Україні: стан та перспективи цифрової трансформації». Вісник Приазовського Державного Технічного Університету. Серія: Економічні науки, вип. 41, 2025, с. 99-106 – URL: <https://doi.org/10.31498/2225-6725.41.2025.348896> (дата звернення: 13.03.2026).
2. Бурка, Богдан О., і Мар'яна С. Шкода. 2025. «Розвиток електронної комерції в Україні в умовах цифрової трансформації, мобільних технологій та економічної нестабільності, спричиненої військовими діями». Журнал стратегічних економічних досліджень, вип. 6 (Червень):8-17. – URL: <https://doi.org/10.30857/2786-5398.2024.6.1>. (дата звернення: 15.03.2026).
3. Florium.ua. – URL: <https://florium.ua/ua/> (дата звернення: 24.03.2026).
4. Agro-Market.ua. – URL: <https://agro-market.net/ua/> (дата звернення: 24.03.2026).
5. Floriya.market. – URL: <https://floriya.market> (дата звернення: 25.03.2026).
6. Fryz M., Mlynko B. Determination of the characteristic function of discrete-time conditional linear random process and its application // Scientific Journal of TNTU. 2023. Vol. 109, № 1. P. 16–23.
7. A. Zaporozhets, Y. Kuts, B. Mlynko, M. Fryz, and L. Scherbak, “EEG Signal Classification Using Linear Process Model-Based Feature Extraction and Supervised Learning,” in *Advanced System Development Technologies II. Studies in Systems, Decision and Control*, M. Bezuglyi, N. Bouraou, V. Mykytenko, G. Tymchyk, and A. Zaporozhets, Eds., Cham: Springer Nature Switzerland, 2025, pp. 235–257. – URL: https://link.springer.com/chapter/10.1007/978-3-031-82035-9_7 (дата звернення: 02.04.2026).
8. M. Fryz, L. Scherbak, B. Mlynko, and T. Mykhailovych, “Linear Random Process Model-Based EEG Classification Using Machine Learning Techniques,” in *Proceedings of the 1st International Workshop on Computer Information*

Technologies in Industry 4.0 (CITI 2023), 2023, vol. 3468, pp. 126–132. – URL: <https://ceur-ws.org/Vol-3468/short5.pdf> (дата звернення: 02.04.2026).

9. V. Babak, A. Zaporozhets, Y. Kuts, M. Fryz, L. Scherbak. Noise signals: Modelling and Analyses. Cham: Springer Nature Switzerland, 2025. 222 p. – URL: <https://doi.org/10.1007/978-3-031-71093-3> (дата звернення: 04.04.2026).

10. Бабак В.П., Марченко Б.Г., Фриз М.Є. Теорія ймовірностей, випадкові процеси та математична статистика. – К.: Техніка, 2004. – 288 с

11. Фриз М.Є., Млинко Б.Б. Умовні лінійні випадкові процеси з дискретним часом та їх властивості // Вісник Хмельницького національного університету. Серія: Технічні науки. 2022 (309), № 3. С. 7–12.

12. Fryz M., Mlynko B. Property analysis of multivariate conditional linear random processes in the problems of mathematical modelling of signals // Technol. Audit Prod. Reserv. 2022. Vol. 3, No 2(65). P. 29–32.

13. WEZOM. Функціональні та нефункціональні вимоги до програмного забезпечення. – URL: <https://wezom.com.ua/ua/blog/funktsionalni-ta-nefunktsionalni-vimogi-do-programnogo-zabezpechennya> (дата звернення: 15.04.2026).

14. IAMPM. Технології для розробки сайту: що це, які найпопулярніші та як вибрати. – URL: <https://iampm.club/ua/blog/tehnologiyi-dlya-rozrobki-sajtu-shho-cze-yaki-najpopulyarnishi-ta-yak-vibrati/> (дата звернення: 16.04.2026).

15. Engage. Що таке HTML: вичерпний посібник для початківців і профі . – URL: <https://engage.org.ua/shho-take-html-vycherpnyj-posibnyk-dlya-rochatkivziv-i-profi/> (дата звернення: 18.04.2026).

16. CSS.in.ua. Довідник по JavaScript. – URL: <https://css.in.ua/js/objects> (дата звернення: 18.04.2026).

17. MySQL Documentation. MySQL 8.0 Reference Manual. – URL: <https://dev.mysql.com/doc/> (дата звернення: 18.04.2026).

18. Microsoft. Visual Studio Code Documentation. – URL: <https://code.visualstudio.com/docs> (дата звернення: 18.04.2026).

19. Apache Friends. XAMPP Documentation. – URL: <https://www.apachefriends.org/> (дата звернення: 18.04.2026).
20. GeeksforGeeks. Client–Server Model. – URL: <https://www.geeksforgeeks.org/client-server-model/> (дата звернення: 18.06.2026).
21. КНУТД. Сучасні пріоритети у проектуванні основних елементів фірмового стилю. – URL: https://er.knutd.edu.ua/bitstream/123456789/28311/1/APSD_2024_V2_P114-117.pdf (дата звернення: 25.04.2026).
22. Lemarbet. Значення кольору в дизайні інтернет-магазину. – URL: <https://lemarbet.com/ua/otkrytie-internet-magazina/znachenie-tsveta-v-dizajne-internet-magazina/> (дата звернення: 26.04.2026).
23. Pexels. Безкоштовний фотосток. – URL: <https://www.pexels.com/uk-ua/> (дата звернення: 01.05.2026).
24. The Complex Agency. Адмінпанелі для бізнесу: від простого до складного. – URL: <https://the-complex.agency/blog/admin-paneli-dlya-biznesu-vid-prostogo-do-skladnogo> (дата звернення: 04.05.2026).
25. Abitap. CRUD-операції з даними. – URL: <https://abitap.com/1-5-operacziyi-z-danymy-crud/> (дата звернення: 06.05.2026).
26. Атаманчук П.С. Безпека життєдіяльності: навч. посіб. Київ : Центр учбової літератури, 2020. 276 с.
27. Безпека життєдіяльності / за ред. А.С. Белікова.
28. Жидецький В.Ц. Охорона праці користувачів комп'ютерів : підручник. Львів : Афіша, 2020. 176 с.
29. ДСанПін 3.3.2.007-98 „Державні санітарні правила і норми роботи з візуальними дисплейними терміналами (ВДП) електронно-обчислювальних машин”.
30. Кодекс законів про працю України (із змінами і доповненнями).
31. ДСТУ 8604:2015 Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги.

32. ДСТУ 7951:2015 Дизайн і ергономіка. Крісло оператора. Загальні ергономічні вимоги.

33. ДБН В.2.5-28 : 2018. „Природне і штучне освітлення” – К.: Мінрегіон України, 2018. 133 с.

34. НПАОП 0,00-7.15-18 Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями, від 14.02.2018 року №207.

35. Правила пожежної безпеки в Україні. (від 30.12.2014 р.. №1417).

ДОДАТКИ

Фрагменти коду клієнтської частини веб-магазину «SproutGarden»

Лістинг А1 – вміст файлу header.php

```

<?php
if(isset($message)) {
    foreach($message as $message) {
        echo '
            <div class="message">
                <span>' . $message . '</span>
                <i class="fas fa-times"
onclick="this.parentElement.remove();"></i>
            </div>
        ';
    }
}
?>
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com"
crossorigin>
<link
href="https://fonts.googleapis.com/css2?family=DM+Sans:wght@400;500;700&display=swap" rel="stylesheet">
<header class="header">
    <div class="flex">
        <a href="home.php" class="logo">
            
            <span>SproutGarden</span>
        </a>
        <nav class="navbar">
            <ul>
                <li><a href="home.php">головна</a></li>
                <li>
                    <a href="#">Категорії▼</a>
                    <ul>
                        <li>
                            <a href="shop.php?category=Кімнатні рослини">
                                Кімнатні рослини▶
                            </a>
                        </li>
                        <li><a
href="shop.php?subcategory=Орхідеї">Орхідеї</a></li>
                            <li><a href="shop.php?subcategory=Декоративно-
квітучі">Декоративно-квітучі</a></li>
                            <li><a href="shop.php?subcategory=Декоративно-
листяні">Декоративно-листяні</a></li>
                            <li><a href="shop.php?subcategory=Сукуленти">Сукуленти
і кактуси</a></li><li><a
href="shop.php?subcategory=Пальми">Пальми</a></li>

```

```
        <li><a
href="shop.php?subcategory=Екзотичні">Екзотичні</a></li>
        <li><a
href="shop.php?subcategory=Цитрусові">Цитрусові</a></li>
    </ul>
</li>
<li>
    <a href="shop.php?category=Саджанці">
        Саджанці▶
    </a>
    <ul>
        <li><a
href="shop.php?subcategory=Дерева">Дерева</a></li>
        <li><a
href="shop.php?subcategory=Троянди">Троянди</a></li>
        <li><a
href="shop.php?subcategory=Яблуні">Яблуні</a></li>
        <li><a
href="shop.php?subcategory=Лаванда">Лаванда</a></li>
        <li><a
href="shop.php?subcategory=Малина">Малина</a></li>
        <li><a
href="shop.php?subcategory=Ожина">Ожина</a></li>
        <li><a
href="shop.php?subcategory=Виноград">Виноград</a></li>
    </ul>
</li>
<li>
    <a href="shop.php?category=Насіння">
        Насіння▶
    </a>
    <ul>
        <li><a
href="shop.php?subcategory=Овочі">Овочі</a></li>
        <li><a
href="shop.php?subcategory=Квіти">Квіти</a></li>
        <li><a
href="shop.php?subcategory=Газон">Газон</a></li>
        <li><a
href="shop.php?subcategory=Трави">Трави</a></li>
    </ul>
</li>
<li>
    <a href="shop.php?category=Горщики">
        Горщики▶
    </a>
    <ul>
        <li><a href="shop.php?subcategory=Для орхідей">Для
орхідей</a></li>
        <li><a href="shop.php?subcategory=Для квітів">Для
квітів</a></li>
        <li><a href="shop.php?subcategory=Для саду">Для
саду</a></li>
    </ul>
</li>
<li>
    <a href="shop.php?category=Добрива">
```

```

        Добрива▶
    </a>
    <ul>
        <li><a
href="shop.php?subcategory=Підкі">Підкі</a></li>
        <li><a
href="shop.php?subcategory=Тверді">Тверді</a></li>
    </ul>
</li>
</ul>
</li>
        <li><a href="shop.php">Магазин</a></li>
        <li><a href="orders.php">Замовлення</a></li>
        <li><a href="#">Акаунт▼</a>
            <ul>
                <li><a href="login.php">Вхід</a></li>
                <li><a
href="register.php">Реєстрація</a></li>
            </ul>
        </li>
    </ul>
</nav>
<div class="icons">
    <div id="menu-btn" class="fas fa-bars"></div>
    <a href="search_page.php" class="fas fa-search"></a>
    <div id="user-btn" class="fas fa-user"></div>
    <?php
        $select_wishlist_count = mysqli_query($conn,
"SELECT * FROM `wishlist` WHERE user_id = '$user_id'") or
die('query failed');
        $wishlist_num_rows =
mysqli_num_rows($select_wishlist_count);
        ?>
        <a href="wishlist.php"><i class="fas fa-
heart"></i><span><?php echo $wishlist_num_rows; ?></span></a>
    <?php
        $select_cart_count = mysqli_query($conn, "SELECT *
FROM `cart` WHERE user_id = '$user_id'") or die('query failed');
        $cart_num_rows =
mysqli_num_rows($select_cart_count);
        ?>
        <a href="cart.php"><i class="fas fa-shopping-
cart"></i><span><?php echo $cart_num_rows; ?></span></a>
    </div>
    <div class="account-box">
        <p>ім'я: <span><?php echo $_SESSION['user_name'];
?></span></p>
        <p>електронна пошта: <span><?php echo
$_SESSION['user_email']; ?></span></p>
        <a href="logout.php" class="delete-btn">logout</a>
    </div>
</div>
</header>

```

Лістинг А2 – вміст файлу footer.php

```
<section class="footer">
  <div class="box-container">
    <div class="box">
      <h3>навігація</h3>
      <a href="home.php">головна</a>
      <a href="shop.php">магазин</a>
      <a href="about.php">про нас</a>
      <a href="contact.php">написати нам</a>
    </div>
    <div class="box">
      <h3>клієнтам</h3>
      <a href="login.php">вхід</a>
      <a href="register.php">реєстрація</a>
      <a href="orders.php">мої замовлення</a>
      <a href="cart.php">мій кошик</a>
    </div>
    <div class="box">
      <h3>контакти</h3>
      <p> <i class="fas fa-phone"></i> +380 68 365 77 44
</p>
      <p> <i class="fas fa-phone"></i> +380 96 356 77 44
</p>
      <p> <i class="fas fa-envelope"></i>
spr0utga7den@gmail.com </p>
    </div>
    <div class="box">
      <h3>слідкуйте за нами</h3>
      <a href="#"><i class="fab fa-
instagram"></i>instagram</a>
      <a href="#"><i class="fab fa-tiktok"></i>TikTok</a>
      <a href="#"><i class="fab fa-facebook-
f"></i>facebook</a>
    </div>
  </div>
  <div class="credit">&copy; SproutGarden Copyright <?php echo
date('Y'); ?> by <span>Samoilenko Y.S</span> </div>
</section>
```

Лістинг А3 – вміст файлу home.php

```
<?php
@include 'config.php';
session_start();

$user_id = $_SESSION['user_id'];
if(!isset($user_id)){
  header('location:login.php');
}
if(isset($_POST['add_to_wishlist'])){
  $product_id = $_POST['product_id'];
```

```

$product_name = $_POST['product_name'];
$product_price = $_POST['product_price'];
$product_image = $_POST['product_image'];

    $check_wishlist_numbers = mysqli_query($conn, "SELECT * FROM
`wishlist` WHERE name = '$product_name' AND user_id = '$user_id'"
or die('query failed');
    $check_cart_numbers = mysqli_query($conn, "SELECT * FROM `cart`
WHERE name = '$product_name' AND user_id = '$user_id'" or
die('query failed');

    if(mysqli_num_rows($check_wishlist_numbers) > 0){
        $message[] = 'вже додано до списку бажань';
    }elseif(mysqli_num_rows($check_cart_numbers) > 0){
        $message[] = 'вже додано в кошик';
    }else{
        mysqli_query($conn, "INSERT INTO `wishlist`(user_id, pid,
name, price, image) VALUES('$user_id', '$product_id',
'$product_name', '$product_price', '$product_image'") or
die('query failed');
        $message[] = 'продукт додано до списку бажань';
    }
}
if(isset($_POST['add_to_cart'])){

    $product_id = $_POST['product_id'];
    $product_name = $_POST['product_name'];
    $product_price = $_POST['product_price'];
    $product_image = $_POST['product_image'];
    $product_quantity = $_POST['product_quantity'];
    $check_cart_numbers = mysqli_query($conn, "SELECT * FROM `cart`
WHERE name = '$product_name' AND user_id = '$user_id'" or
die('query failed');
    if(mysqli_num_rows($check_cart_numbers) > 0){
        $message[] = 'вже додано в кошик';
    }else{
        $check_wishlist_numbers = mysqli_query($conn, "SELECT *
FROM `wishlist` WHERE name = '$product_name' AND user_id =
'$user_id'" or die('query failed');
        if(mysqli_num_rows($check_wishlist_numbers) > 0){
            mysqli_query($conn, "DELETE FROM `wishlist` WHERE name
= '$product_name' AND user_id = '$user_id'" or die('query
failed');
        }
        mysqli_query($conn, "INSERT INTO `cart`(user_id, pid, name,
price, quantity, image) VALUES('$user_id', '$product_id',
'$product_name', '$product_price', '$product_quantity',
'$product_image'") or die('query failed');
        $message[] = 'продукт додано в кошик';
    }
}
?>
<!DOCTYPE html>

```

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>SproutGarden - Головна</title>
  <link rel="icon" type="image/svg+xml" href="images/logo.svg">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">
  <link rel="stylesheet" href="css/style.css">
</head>
<body>

<?php @include 'header.php'; ?>
<section class="home">
  <div class="content">
    <h3>Щастя проростає зсередини</h3>
    <p>Великий вибір рослин та товарів для садівництва за
доступними цінами.</p>
    <a href="about.php" class="btn">дослідити каталог</a>
  </div>
</section>
<section class="products">
  <h1 class="title">хіти продажу</h1>
  <div class="box-container">
    <?php
      $select_products = mysqli_query($conn, "SELECT * FROM
`products` LIMIT 6") or die('query failed');
      if(mysqli_num_rows($select_products) > 0){
        while($fetch_products =
mysqli_fetch_assoc($select_products)){
          ?>
          <form action="" method="POST" class="box">
            <a href="view_page.php?pid=<?php echo $fetch_products['id'];
?>" class="product-link">
              
              <div class="name">
                <?php echo $fetch_products['name']; ?>
              </div>
            </a>
            <button type="submit" name="add_to_wishlist" class="wishlist-
btn">
              <i class="fas fa-heart"></i>
            </button>
            <div class="product-bottom">
              <div class="price">
                <?php echo $fetch_products['price']; ?> ₾
              </div>
              <input type="number"
                name="product_quantity"

```

```

        value="1"
        min="1"
        class="qty">
    </div>
    <input type="hidden" name="product_id" value="<?php echo
    $fetch_products['id']; ?>">
    <input type="hidden" name="product_name" value="<?php echo
    $fetch_products['name']; ?>">
    <input type="hidden" name="product_price" value="<?php echo
    $fetch_products['price']; ?>">
    <input type="hidden" name="product_image" value="<?php echo
    $fetch_products['image']; ?>">
    <input type="submit"
        value="Додати в кошик"
        name="add_to_cart"
        class="btn">
</form>
    <?php
        }
    }else{
        echo '<p class="empty">no products added yet!</p>';
    }
    ?>
</div>
<div class="more-btn">
    <a href="shop.php" class="option-btn">більше</a>
</div>
</section>
<section class="home-contact">
    <div class="content">
        <h3>Маєте запитання?</h3>
        <p>Ваш комфорт – наш пріоритет. Залишилися запитання? Ми з
        радістю допоможемо!</p>
        <a href="contact.php" class="btn">написати нам</a>
    </div>
</section>
<?php @include 'footer.php'; ?>
<script src="js/script.js"></script>
</body>
</html>

```

Лістинг А4 – вміст файлу login.php

```

<?php
@include 'config.php';
session_start();
if(isset($_POST['submit'])){
    $filter_email = filter_var($_POST['email'],
    FILTER_SANITIZE_STRING);
    $email = mysqli_real_escape_string($conn, $filter_email);
    $filter_pass = filter_var($_POST['pass'],
    FILTER_SANITIZE_STRING);

```

```

    $pass = mysqli_real_escape_string($conn, md5($filter_pass));
    $select_users = mysqli_query($conn, "SELECT * FROM `users`
WHERE email = '$email' AND password = '$pass'") or die('query
failed');
    if(mysqli_num_rows($select_users) > 0){

        $row = mysqli_fetch_assoc($select_users);
        if($row['user_type'] == 'admin'){
            $_SESSION['admin_name'] = $row['name'];
            $_SESSION['admin_email'] = $row['email'];
            $_SESSION['admin_id'] = $row['id'];
            header('location:admin_page.php');
        }elseif($row['user_type'] == 'user'){
            $_SESSION['user_name'] = $row['name'];
            $_SESSION['user_email'] = $row['email'];
            $_SESSION['user_id'] = $row['id'];
            header('location:home.php');
        }else{
            $message[] = 'користувача не знайдено';
        }
    }else{
        $message[] = 'невірний пароль або адреса електронної
пошти!';
    }
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>SproutGarden - Вхід</title>
    <link rel="icon" type="image/svg+xml" href="images/logo.svg">
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">
    <link rel="stylesheet" href="css/style.css">
</head>
<body>
<?php
if(isset($message)){
    foreach($message as $message){
        echo '
        <div class="message">
            <span>'. $message. '</span>
            <i class="fas fa-times"
onclick="this.parentElement.remove();"></i>
        </div>
        ';
    }
}

```

?>

```
<section class="form-container">
  <form action="" method="post">
    <h3>Вхід</h3>
    <input type="email" name="email" class="box"
placeholder="введіть адресу електронної пошти" required>
    <input type="password" name="pass" class="box"
placeholder="введіть пароль" required>
    <input type="submit" class="btn" name="submit"
value="увійти">
    <p>не маєте акаунту? <a
href="register.php">зарєєструватися</a></p>
  </form>
</section>
</body>
</html>
```

Лістинг А5 – вміст файлу script.js

```
let userBox = document.querySelector('.header .flex .account-
box');
document.querySelector('#user-btn').onclick = () =>{
  userBox.classList.toggle('active');
  navbar.classList.remove('active');
}
let navbar = document.querySelector('.header .flex .navbar');

document.querySelector('#menu-btn').onclick = () =>{
  navbar.classList.toggle('active');
  userBox.classList.remove('active');
}
window.onscroll = () =>{
  userBox.classList.remove('active');
  navbar.classList.remove('active');
}
```

Фрагменти програмного коду адміністративної панелі веб-магазину «SproutGarden»

Лістинг В1 – вміст файлу admin_header.php

```

<?php
if(isset($message)){
    foreach($message as $message){
        echo '
        <div class="message">
            <span>' . $message . '</span>
            <i class="fas fa-times"
onclick="this.parentElement.remove();"></i>
        </div>
        ';
    }
}
?>
<header class="header">
    <div class="flex">
        <a href="admin_page.php" class="logo">Адміністративна<span>
Панель</span></a>
        <nav class="navbar">
            <a href="admin_page.php">головна</a>
            <a href="admin_products.php">продукти</a>
            <a href="admin_orders.php">замовлення</a>
            <a href="admin_users.php">користувачі</a>
            <a href="admin_contacts.php">повідомлення</a>
        </nav>
        <div class="icons">
            <div id="menu-btn" class="fas fa-bars"></div>
            <div id="user-btn" class="fas fa-user"></div>
        </div>
        <div class="account-box">
            <p>ім'я: <span><?php echo $_SESSION['admin_name'];
?></span></p>
            <p>електронна пошта: <span><?php echo
$_SESSION['admin_email']; ?></span></p>
            <a href="logout.php" class="delete-btn">Вийти</a>
            <div> <a href="login.php">логіні</a> | <a
href="register.php">реєстрація</a> </div>
        </div>
    </div>
</header>

```

Лістинг В2 – вміст файлу admin_page.php

```
<?php
@include 'config.php';
session_start();
$admin_id = $_SESSION['admin_id'];
if(!isset($admin_id)){
    header('location:login.php');
};
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Адмінпанель - дашборд</title>
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">
    <link rel="stylesheet" href="css/admin_style.css">
</head>
<body>

<?php @include 'admin_header.php'; ?>
<section class="dashboard">
    <h1 class="title">dashboard</h1>
    <div class="box-container">
        <div class="box">
            <?php
                $total_pendings = 0;
                $select_pendings = mysqli_query($conn, "SELECT * FROM
`orders` WHERE payment_status = 'pending'") or die('query
failed');
                while($fetch_pendings =
mysqli_fetch_assoc($select_pendings)){
                    $total_pendings += $fetch_pendings['total_price'];
                };
            ?>
            <h3><?php echo $total_pendings; ?></h3>
            <p>в очікуванні</p>
        </div>
        <div class="box">
            <?php
                $total_completes = 0;
                $select_completes = mysqli_query($conn, "SELECT * FROM
`orders` WHERE payment_status = 'completed'") or die('query
failed');
                while($fetch_completes =
mysqli_fetch_assoc($select_completes)){
                    $total_completes +=
$fetch_completes['total_price'];
```

```

        };
    ?>
    <h3><?php echo $total_completes; ?></h3>
    <p>виконані платежі</p>
</div>
<div class="box">
    <?php
        $select_orders = mysqli_query($conn, "SELECT * FROM
`orders`") or die('query failed');
        $number_of_orders = mysqli_num_rows($select_orders);
    ?>
    <h3><?php echo $number_of_orders; ?></h3>
    <p>розміщені замовлення </p>
</div>
<div class="box">
    <?php
        $select_products = mysqli_query($conn, "SELECT * FROM
`products`") or die('query failed');
        $number_of_products =
mysqli_num_rows($select_products);
    ?>
    <h3><?php echo $number_of_products; ?></h3>
    <p>доданих продуктів</p>
</div>
<div class="box">
    <?php
        $select_users = mysqli_query($conn, "SELECT * FROM
`users` WHERE user_type = 'user'") or die('query failed');
        $number_of_users = mysqli_num_rows($select_users);
    ?>
    <h3><?php echo $number_of_users; ?></h3>
    <p>користувачі</p>
</div>
<div class="box">
    <?php
        $select_admin = mysqli_query($conn, "SELECT * FROM
`users` WHERE user_type = 'admin'") or die('query failed');
        $number_of_admin = mysqli_num_rows($select_admin);
    ?>
    <h3><?php echo $number_of_admin; ?></h3>
    <p>адміністратори</p>
</div>
<div class="box">
    <?php
        $select_account = mysqli_query($conn, "SELECT * FROM
`users`") or die('query failed');
        $number_of_account = mysqli_num_rows($select_account);
    ?>
    <h3><?php echo $number_of_account; ?></h3>
    <p>всі акаунти</p>
</div>
<div class="box">
    <?php

```

```

        $select_messages = mysqli_query($conn, "SELECT * FROM
`message`) or die('query failed');
        $number_of_messages =
mysqli_num_rows($select_messages);
        ?>
        <h3><?php echo $number_of_messages; ?></h3>
        <p>нові повідомлення</p>
    </div>
</div>
</section>
<script src="js/admin_script.js"></script>
</body>
</html>

```

Лістинг В3 – вміст файлу admin_update_product.php

```

<?php
@include 'config.php';
session_start();
$admin_id = $_SESSION['admin_id'];
if(!isset($admin_id)){
    header('location:login.php');
};
if(isset($_POST['update_product'])){
    $update_p_id = $_POST['update_p_id'];
    $name = mysqli_real_escape_string($conn, $_POST['name']);
    $price = mysqli_real_escape_string($conn, $_POST['price']);
    $details = mysqli_real_escape_string($conn, $_POST['details']);
    mysqli_query($conn, "UPDATE `products` SET name = '$name',
details = '$details', price = '$price' WHERE id = '$update_p_id'")
or die('query failed');
    $image = $_FILES['image']['name'];
    $image_size = $_FILES['image']['size'];
    $image_tmp_name = $_FILES['image']['tmp_name'];
    $image_folter = 'uploaded_img/'.$image;
    $old_image = $_POST['update_p_image'];

    if(!empty($image)){
        if($image_size > 2000000){
            $message[] = 'розмір файлу зображення занадто великий!';
        }else{
            mysqli_query($conn, "UPDATE `products` SET image =
'$image' WHERE id = '$update_p_id'") or die('query failed');
            move_uploaded_file($image_tmp_name, $image_folter);
            unlink('uploaded_img/'.$old_image);
            $message[] = 'зображення успішно оновлено!';
        }
    }
    $message[] = 'продукт успішно оновлено!';
}
?>
<!DOCTYPE html>

```

```

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Адмінпанель - оновити товар</title>
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">
  <link rel="stylesheet" href="css/admin_style.css">
</head>
<body>

<?php @include 'admin_header.php'; ?>
<section class="update-product">
<?php
  $update_id = $_GET['update'];
  $select_products = mysqli_query($conn, "SELECT * FROM
`products` WHERE id = '$update_id'") or die('query failed');
  if(mysqli_num_rows($select_products) > 0){
    while($fetch_products =
mysqli_fetch_assoc($select_products)){
    ?>
<form action="" method="post" enctype="multipart/form-data">
  
  <input type="hidden" value="<?php echo $fetch_products['id'];
?>" name="update_p_id">
  <input type="hidden" value="<?php echo
$fetch_products['image']; ?>" name="update_p_image">
  <input type="text" class="box" value="<?php echo
$fetch_products['name']; ?>" required placeholder="оновити назву
продукту" name="name">
  <input type="number" min="0" class="box" value="<?php echo
$fetch_products['price']; ?>" required placeholder="оновити ціну
продукту" name="price">
  <textarea name="details" class="box" required
placeholder="оновити деталі продукту" cols="30" rows="10"><?php
echo $fetch_products['details']; ?></textarea>
  <input type="file" accept="image/jpg, image/jpeg, image/png"
class="box" name="image">
  <input type="submit" value="оновити продукт"
name="update_product" class="btn">
  <a href="admin_products.php" class="option-btn">назад</a>
</form>
<?php
  }
  }else{
    echo '<p class="empty">no update product select</p>';
  }?>
</section>
<script src="js/admin_script.js"></script>

```

```
</body>
</html>
```

Лістинг В4 – вміст файлу admin_script.js

```
let navbar = document.querySelector('.header .flex .navbar');
let userBox = document.querySelector('.header .flex .account-
box');
document.querySelector('#menu-btn').onclick = () =>{
    navbar.classList.toggle('active');
    userBox.classList.remove('active');
}
document.querySelector('#user-btn').onclick = () =>{
    userBox.classList.toggle('active');
    navbar.classList.remove('active');
}
window.onscroll = () =>{
    navbar.classList.remove('active');
    userBox.classList.remove('active');
}
```

Лістинг В5 – SQL-скрипт створення структури бази даних інтернет-магазину

«SproutGarden»

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

CREATE TABLE `cart` (
  `id` int(100) NOT NULL,
  `user_id` int(100) NOT NULL,
  `pid` int(100) NOT NULL,
  `name` varchar(100) NOT NULL,
  `price` int(100) NOT NULL,
  `quantity` int(100) NOT NULL,
  `image` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `message` (
  `id` int(100) NOT NULL,
  `user_id` int(100) NOT NULL,
  `name` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `number` varchar(12) NOT NULL,
  `message` varchar(500) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `orders` (
  `id` int(100) NOT NULL,
  `user_id` int(100) NOT NULL,
  `name` varchar(100) NOT NULL,
```

```

`number` varchar(12) NOT NULL,
`email` varchar(100) NOT NULL,
`method` varchar(50) NOT NULL,
`address` varchar(500) NOT NULL,
`total_products` varchar(1000) NOT NULL,
`total_price` int(100) NOT NULL,
`placed_on` varchar(50) NOT NULL,
`payment_status` varchar(20) NOT NULL DEFAULT 'pending'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `products` (
  `id` int(100) NOT NULL,
  `name` varchar(100) NOT NULL,
  `details` varchar(500) NOT NULL,
  `price` int(100) NOT NULL,
  `category` varchar(100) NOT NULL,
  `subcategory` varchar(100) NOT NULL,
  `image` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `users` (
  `id` int(100) NOT NULL,
  `name` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL,
  `user_type` varchar(20) NOT NULL DEFAULT 'user'
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `wishlist` (
  `id` int(100) NOT NULL,
  `user_id` int(100) NOT NULL,
  `pid` int(100) NOT NULL,
  `name` varchar(100) NOT NULL,
  `price` int(100) NOT NULL,
  `image` varchar(100) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

ALTER TABLE `cart`
  ADD PRIMARY KEY (`id`);

ALTER TABLE `message`
  ADD PRIMARY KEY (`id`);

ALTER TABLE `orders`
  ADD PRIMARY KEY (`id`);

ALTER TABLE `products`
  ADD PRIMARY KEY (`id`);

ALTER TABLE `users`
  ADD PRIMARY KEY (`id`);

ALTER TABLE `wishlist`

```

```
ADD PRIMARY KEY (`id`);

ALTER TABLE `cart`
  MODIFY `id` int(100) NOT NULL AUTO_INCREMENT;
ALTER TABLE `message`
  MODIFY `id` int(100) NOT NULL AUTO_INCREMENT;

ALTER TABLE `orders`
  MODIFY `id` int(100) NOT NULL AUTO_INCREMENT;

ALTER TABLE `products`
  MODIFY `id` int(100) NOT NULL AUTO_INCREMENT;

ALTER TABLE `users`
  MODIFY `id` int(100) NOT NULL AUTO_INCREMENT;

ALTER TABLE `wishlist`
  MODIFY `id` int(100) NOT NULL AUTO_INCREMENT;

COMMIT;
```