

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 вебсайту інтернет-магазину фруктів «FreshFruit»
з використанням HTML, CSS, PHP та JavaScript

Виконав: студент IV курсу, групи СН-42
спеціальності 122 Комп'ютерні науки
(шифр і назва спеціальності)

Олійник І.І.
(підпис) (прізвище та ініціали)

Керівник Дмитроца Л.П.
(підпис) (прізвище та ініціали)

Нормоконтроль Шимчук Г.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри Боднарчук І.О.
(підпис) (прізвище та ініціали)

Рецензент Голотенко О.С.
(підпис) (прізвище та ініціали)

Тернопіль
2025

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)

« 27 » січня 2025 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

Студенту Олійник Ірині Іванівні
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка вебсайту інтернет-магазину фруктів «FreshFruit» з використанням HTML, CSS, PHP та JavaScript

Керівник роботи Дмитроца Леся Павлівна, к.т.н, доцент кафедри КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 07 » травня 2025 року № 4/7-444

2. Термін подання студентом завершеної роботи 26 червня 2025 р.

3. Вихідні дані до роботи Літературні та інтернет джерела щодо розробки інтернет-магазину включаючи матеріали з веб-програмування, сучасних рішень у сфері електронної комерції.

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

Вступ. 1. Аналіз цільової аудиторії та визначення вимог до розробки інтернет-магазину «freshfruit». 2. Проектування та технічне планування інтернет-магазину «freshfruit».

3. Створення та функціональна реалізація інтернет-магазину «FreshFruit».

4. Безпека життєдіяльності, основи охорони праці. Висновки. Перелік джерел. Додатки.

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

1.Титульний слайд. 2. Мета кваліфікаційної роботи, об'єкт та предмет дослідження.

3.Переваги та недоліки інтернет-магазинів. 4. Найпопулярніші рішення конкурентів в сфері онлайн-продажу фруктів. 5. Аналіз потенційних користувачів. 6.Визначення вимог

7. Архітектура сайту та розподіл ролей. 8. Вибір технологій для реалізації. 9.Врахування

UI/UX при розробці сайту «FreshFruit». 10.Проектування бази даних. 11. Реалізація основних блоків інтрнет-магазину. 12. Тестування функціональності сайту. 13. Висновки .

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Сенчишин В.С., кандидат технічних наук, доцент кафедри МТ		

7. Дата видачі завдання 27 січня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	27.01.2025	
2.	Підбір та опрацювання літературних джерел по темі кваліфікаційної роботи	08.05.2025-14.05.2025	
3.	Виконання дослідження щодо розробки інтернет-магазинів	15.05.2025-26.05.2025	
	Розроблення вебсайту інтернет-магазину фруктів «FreshFruit»		
4.	Оформлення розділу «Аналіз цільової аудиторії та визначення вимог до розробки інтернет-магазину»	27.05.2025-28.06.2025	
5.	Оформлення розділу «Проектування та технічне планування інтернет-магазину FreshFruit»	29.05.2025-31.05.2025	
6.	Оформлення розділу «Створення та функціональна реалізація інтернет-магазину FreshFruit»	01.06.2025-04.06.2025	
7.	Виконання завдання до підрозділу «Безпека життєдіяльності»	05.06.2025-06.06.2025	
8.	Виконання завдання до підрозділу «Основи охорони праці»	05.06.2025-06.06.2025	
9.	Оформлення кваліфікаційної роботи	07.06.2025-10.06.2025	
10.	Нормоконтроль	11.06.2025-13.06.2025	
11.	Перевірка на плагіат	16.06.2025	
12.	Попередній захист кваліфікаційної роботи	18.06.2025	
13.	Захист кваліфікаційної роботи	27.06.2025	

Студент

(підпис)

Олійник І.І.

(прізвище та ініціали)

Керівник роботи

(підпис)

Дмитроца Л.П.

(прізвище та ініціали)

АНОТАЦІЯ

Розробка вебсайту інтернет-магазину фруктів «FreshFruit» з використанням HTML, CSS, PHP та JavaScript // Кваліфікаційна робота освітнього рівня «Бакалавр» // Олійник Ірина Іванівна // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СН-42 // Тернопіль, 2025 // С. – 85, рис. – 52, табл. – 9, кресл. – 0, додат. – 2, бібліогр. – 38.

Ключові слова: інтернет-магазин, фрукти, веброзробка, сервер, автоматизація, дизайн, тестування, клієнт.

Кваліфікаційна робота присвячена, розробці функціонального вебсайту інтернет-магазину фруктів «FreshFruit» з використанням HTML, CSS, PHP та JavaScript. Метою дослідження є створення зручного, привабливого та технологічно ефективного ресурсу для автоматизації продажів та врахування потреб як покупців, так і власників бізнесу.

У першому розділі кваліфікаційної роботи проаналізовано актуальність створення вебресурсу для продажу фруктами онлайн. Досліджено стан ринку інтернет-магазинів в Україні та світі, їх переваги і недоліки. Визначено особливості цільової аудиторії та конкурентного середовища. Сформульовано технічні вимоги до сайту і перелік повноцінного функціоналу для ефективної роботи інтернет-магазину «FreshFruit».

У другому розділі викладено технічне планування вебресурсу. Обґрунтовано вибір технологій, описано вимоги до хостингу та обрано найкращий варіант, визначено структуру клієнтської та серверної частини, продумано аспекти UI/UX-дизайну. Особливу увагу приділено проектуванню бази даних для збереження всієї інформації сайту.

У третьому розділі описано реалізацію всіх модулів сайту та проведено тестування ключових аспектів функціональності інтернет-магазину.

ANNOTATION

Development of the «FreshFruit» Online Fruit Store Website Using HTML, CSS, PHP, and JavaScript // Qualification work of the educational level «Bachelor» // Oliinyk Iryna Ivanivna // Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SN-42 // Ternopil, 2025 // P. – 85, fig. –52, tabl. – 9, chair. – 0, annexes. – 2, references – 38.

Keywords: online store, fruits, web development, server, automation, design, testing, client.

This qualification work is dedicated to the development of a functional website for the FreshFruit online fruit store using HTML, CSS, PHP, and JavaScript. The aim of the research is to create a convenient, attractive, and technologically advanced resource to automate sales and address the needs of both customers and business owners.

The first chapter analyzes the relevance of creating a web resource for selling fruits online. It investigates the state of the online store market in Ukraine and worldwide, highlighting their advantages and disadvantages. The specifics of the target audience and competitive environment are defined. Technical requirements for the site and a list of full-fledged functionalities necessary for the effective operation of the FreshFruit online store are formulated.

The second chapter presents the technical planning of the web resource. The choice of technologies is justified; hosting requirements and the selected database management system are described. The structure of the client and server sides is determined, and UI/UX design aspects are considered. Special attention is given to the design of the database for storing all site information.

The third chapter describes the implementation of all site modules and the testing of key aspects of the online store's functionality.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

CSS – Cascading Style Sheets – Каскадні таблиці стилів.

ER-діаграма – Entity-Relationship diagram – Діаграма сутність-зв'язок.

HTML – Hypertext Markup Language – Мова розмітки гіпертексту.

JS – JavaScript.

PHP – Hypertext Preprocessor – Препроцесор гіпертексту.

POST – Hypertext Transfer Protocol POST – Метод надсилання даних у протоколі HTTP.

SQL – Structured Query Language – Структурована мова запитів.

UI – User Interface – Інтерфейс користувача.

UML – Unified Modeling Language – Уніфікована мова моделювання.

UX – User Experience – Досвід користувача.

БД – База даних.

ПК – Персональний комп'ютер.

ЗМІСТ

РОЗДІЛ 1. АНАЛІЗ ЦІЛЬОВОЇ АУДИТОРІЇ ТА ВИЗНАЧЕННЯ ВИМОГ ДО РОЗРОБКИ ІНТЕРНЕТ-МАГАЗИНУ «FRESHFRUIT»	10
1.1 Дослідження сучасних тенденцій розвитку інтернет-магазинів.....	10
1.2 Переваги та недоліки інтернет-магазинів з позиції покупця і продавця ...	11
1.3 Особливості реалізації онлайн-продажу свіжих фруктів	13
1.3.1 Аналіз потенційних користувачів онлайн-сервісу «FreshFruit»	14
1.3.2 Вивчення основних конкурентів та учасників ринку	15
1.4 Постановка задачі, та визначення функціональних вимог до інтернет-магазину	19
1.4.1 Основна мета розробки вебсайту «FreshFruit» та її відповідність потребам ринку	19
1.4.2 Визначення ключових функцій інтернет-магазину «FreshFruit».....	20
РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА ТЕХНІЧНЕ ПЛАНУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ «FRESHFRUIT»	22
2.1 Загальний підхід до побудови вебресурсу	22
2.1.1 Вибір ефективної архітектури для інтернет-рішення	22
2.1.2 Розподілення ролей між клієнтською та серверною частинами.....	24
2.2 Добір інструментів реалізації інтернет-магазину FreshFruit	27
2.2.1 Вибір мов розмітки, стилів (HTML, CSS) і фреймворку (Bootstrap) для побудови інтерфейсу	28
2.2.2 Використання JavaScript для побудови динамічних елементів	29
2.2.3 Застосування PHP для серверної логіки	30
2.2.4 Обґрунтування вибору бази даних для збереження інформації	31
2.2.5 Технічні критерії вибору хостингової платформи	32
2.3 Використання UI/UX для сайту FreshFruit	33
2.4. Проектування бази даних.....	36
РОЗДІЛ 3. СТВОРЕННЯ ТА ФУНКЦІОНАЛЬНА РЕАЛІЗАЦІЯ ІНТЕРНЕТ-МАГАЗИНУ «FRESHFRUIT»	43

3.1 Реалізація модуля реєстрації користувачів і входу до системи	43
3.2 Формування структури стартової (головної) сторінки	46
3.3 Реалізація розділу з акційними пропозиціями, каталогом товарів та фільтрацією.....	50
3.4 Впровадження блоків інформації про переваги сервісу та контактної інформації сайту.....	56
3.5 Додавання функціоналу зібрання відгуків і оформлення підписки	58
3.6 Програмування логіки кошика та оформлення замовлення.....	60
3.7 Проведення тестових перевірок працездатності та зручності ресурсу	63
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	68
4.1 Вимоги ергономіки до організації робочого місця оператора ПК.....	68
4.2 Психологічні чинники небезпеки при розробці сайту	70
4.3 Загальні вимоги безпеки з охорони праці для веб-розробника.....	72
ВИСНОВКИ.....	74
ПЕРЕЛІК ДЖЕРЕЛ	76
ДОДАТКИ	

ВСТУП

Актуальність теми. Інтернет став невіддільною частиною нашого життя, і все більше людей обирають онлайн-покупки замість традиційних магазинів. Це особливо зручно, коли йдеться про щоденні продукти, зокрема свіжі фрукти. Проте попит на якісний сервіс постійно зростає – користувачі очікують простий і зрозумілий інтерфейс, швидке оформлення замовлень і надійну доставку. Саме тому створення інтернет-магазину, який відповідає сучасним вимогам та орієнтований на зручність користувача, є важливим напрямком розробки у сфері вебтехнологій та електронної комерції.

Мета і задачі дослідження. Метою даної кваліфікаційної роботи освітнього рівня «Бакалавр» є створення функціонального та зручного інтернет-магазину «FreshFruit», який сприятиме підвищенню якості онлайн-послуг із продажу фруктів, відповідатиме потребам цільової аудиторії та забезпечуватиме конкурентоспроможність ресурсу.

Для досягнення поставленої мети необхідно виконати такі завдання:

- проаналізувати стан сучасного ринку інтернет-магазинів;
- дослідити переваги й недоліки онлайн-торгівлі з боку продавця і покупця;
- визначити особливості електронної комерції в сфері продажу фруктів;
- сформулювати цілі та технічні вимоги до функціонування сайту;
- обґрунтувати вибір інструментів і технологій для реалізації вебресурсу;
- спроектувати архітектуру сайту, його базу даних та інтерфейс;
- реалізувати модулі інтернет-магазину та провести тестування його функціональності.

Практичне значення одержаних результатів. У результаті реалізації виконання кваліфікаційної роботи працюючий вебсайт інтернет-магазину «FreshFruit» із можливістю реєстрації користувачів, оформлення замовлень, перегляду акцій та підписки на оновлення.

РОЗДІЛ 1. АНАЛІЗ ЦІЛЬОВОЇ АУДИТОРІЇ ТА ВИЗНАЧЕННЯ ВИМОГ ДО РОЗРОБКИ ІНТЕРНЕТ-МАГАЗИНУ «FRESHFRUIT»

1.1 Дослідження сучасних тенденцій розвитку інтернет-магазинів

Впродовж останнього десятиліття ринок онлайн-торгівлі стрімко зростає. Ще 7 – 10 років тому онлайн-покупки для багатьох людей здавалися чимось незвичним або ж навіть і небезпечним. Люди остерігалися вводити дані своєї картки, боялися, що товар може бути не якісним або гроші за товар можуть бути зняті, а він взагалі не прийде. Але з часом ситуація кардинально змінилася. Сьогодні купити щось через інтернет – це така ж звичайна справа як сходити до магазину. Люди звикли до зручності: не треба стояти в чергах, витратити час на поїздки чи шукати потрібний товар по всьому місту. Все є в одному місці – інтернеті, і головне – можна одразу порівняти ціни, прочитати відгуки покупців, які вже замовляли певний товар, а також порівнюючи це на різних вебсайтах вибрати найкращий варіант для себе. За даними аналітичних дослідження продажі через інтернет-магазини кожного року збільшується на 15-20%. Онлайн магазини стали частиною повсякденного життя для мільйонів людей: більше ніж 33% населення зі всього світу роблять покупки в інтернеті [1].

Важливо зазначити, що інтернет-магазини можуть бути представниками малого або середнього бізнесу, який є ключовою рушійною силою економіки. Однак більшість вітчизняних МСП відстають у цифровій трансформації, що призводить до зниження продуктивності, конкурентоспроможності та посилення «цифрового розриву» між регіонами і підприємствами [2].

Цифрова трансформація малого та середнього бізнесу в Україні, зокрема в контексті євроінтеграції, значною мірою впливає на розвиток електронної комерції, зокрема інтернет-магазинів. Завдяки цьому все більше підприємств впроваджують сучасні цифрові інструменти для автоматизації процесів продажу та покращення взаємодії з клієнтами, що підвищує конкурентоспроможність українських онлайн-платформ на міжнародному рівні [3].

Зростає не лише кількість покупців, а й самих інтернет-магазинів. Хтось продає одяг, хтось техніку, а хтось – продукти. Раніше здавалося, що саме продукти зручно купувати тільки на ринку або в супермаркеті, але все змінилось. Особливо це стало помітно після 2020 року, коли багато хто вперше замовили продукти додому через те що почався карантин та боязнь заразитися вірусом, який може призвести до смерті сильніша ніж боязнь купити продукти харчування через інтернет. Проте згодом виявилось, що це зручно, швидко і зовсім не страшно [4].

З іншого боку, через таку популярність зросла і конкуренція. Інтернет-магазинів стало дуже багато, і тепер просто створити сайт – недостатньо. Важливо, щоб він був зручним, працював швидко і добре виглядав як на комп'ютері, так і на телефоні. Бо якщо людині щось незрозуміло або складно – вона просто закриває вкладку й піде до іншого продавця.

Ще один момент – це вузька спеціалізація. Якщо раніше магазини намагались продавати все підряд, то тепер усе частіше з'являються невеликі онлайн-платформи, які зосереджуються на чомусь конкретному: тільки каві, тільки екотоварах, або як у моєму випадку – тільки фруктах.. Такий підхід дозволяє краще підійти до потреб клієнтів, зробити сервіс більш персоналізованим і зручним.

1.2 Переваги та недоліки інтернет-магазинів з позиції покупця і продавця

Онлайн-магазини, з початком стрімкого розвитку цифрових технологій, стали невід'ємною частиною світу. Проте, як і будь-яка річ, інтернет-магазини мають свої переваги та недоліки як з погляду користувача, так і з погляду технічної та організаційної точки зору, тобто продавця.

Переваги використання інтернет магазинів:

– Цілодобовий доступ. Онлайн-магазини доступні 24/7, без вихідних та перерв. Достатньо мати підключення до Інтернету і весь асортимент буде у кількох кліках від користувача [5].

– Гнучкість у розробці функціоналу. Розробники можуть створювати адаптивні інтерфейси, які автоматично підлаштовуються під різні пристрої, наприклад ПК, планшети та смартфони. Це дозволяє охопити ширшу аудиторію та зробити користувацький досвід комфортнішим.

– Розширені можливості до вподобань користувача. Системи аналітики дозволяють вивчати поведінку відвідувачів сайту. На основі цих даних сайт може показувати релевантні продукти.

– Автоматичне управління товарами та замовленнями. За допомогою інтегрованих модулів адміністратори можуть зручно керувати товаром зі складу, обробляти замовлення, оновлювати ціни та акції. Це полегшує управління великим обсягом продукції.

– Зменшення витрат на оренду та персонал. Інтернет-магазин не потребує фізичного приміщення та відповідно не потрібна велика кількість персоналу щоб цим керувати. Це дає змогу зекономити кошти, а частину з них можна інвестувати в покращення платформи чи сервісу доставки.

– Можливість інтеграції з зовнішніми сервісами. Онлайн-магазин можна легко підключити до платіжних систем, служб доставки та CRM (Customer Relationship Management).

– Просте масштабування. У випадку зростання бізнесу, його можна легко масштабувати: додати нові категорії, локалізації, функції без потреби в додаткових площадок чи персоналі.

Щодо недоліків, то їх теж є чимало, ось приклади:

– Відсутність живого огляду товару. Для багатьох людей важливо купувати речі, на які вони можуть подивитися наживо. Особливо важливо це у випадку з продуктами харчування. Клієнт не може оцінити запах, м'якість чи зовнішній вигляд товару перед покупкою, а може лише довіритися написаній інформації на сайті.

- Потреба в грамотному UI/UX-дизайні. Недостатньо просто розмістити каталог товарів, сайт має бути інтуїтивно зрозумілим та логічно структурованим. Якщо користувач не зможе знайти потрібне, то він буде мати можливість перейти до конкурентів, тобто на інший більш зрозумілий сайт.
- Залежність технічної стабільності сайту. Якщо сайт «впаде» або буде повільно завантажуватися – це може прямо вплинути на рівень продажів і репутацію.
- Ризики шахрайства та безпеки даних. Питання безпеки є ключовим аспектом в інтернет-магазинах. Потрібно забезпечити надійний захист персональних та платіжних даних клієнтів, використовувати відповідні SSL (Secure Sockets Layer) сертифікати, двоетапну авторизацію та регулярно проводити перевірку безпеки [6].
- Ускладнене повернення товарів. Якщо з продуктом щось не так, то процес повернення може займати деякий період часу.
- Витрати на просування. Інтернет-магазин без SEO (Search Engine Optimization), реклами, присутності в соціальних мережах є практично невидимий. Щоб прикликати нових клієнтів, потрібно вкладати ресурси в просування маркетингу [7].

1.3 Особливості реалізації онлайн-продажу свіжих фруктів

У сучасному світі продукти харчування, зокрема фрукти, займають досить важливе місце в щоденному житті кожного. Разом із зростанням темпу життя багато хто з людей вже не має часу ходити по супермаркетах щоб отримати стиглі яблука чи соковиті апельсини. Саме тому ринок онлайн-продажу фруктів поступово все більше зростає.

Фрукти – це така категорія товарів, яка потребує особливої уваги. Вони швидко псуються, дуже чутливі до транспортування, мають обмежений термін зберігання та повинні підтримуватися певними вимогами зберігання і саме тому замовлення через інтернет викликає у покупців чимало сумнівів. Але попит на

подібні сервіси все одно зростає. Чимало людей голові платити за зручність і час, який вони економлять, обираючи онлайн замовлення замість походу в магазин.

Предметна область даного дослідження охоплює створення вебсайту, який дозволить користувачу зручно та без зайвих клопотів купувати свіжі фрукти онлайн. У фокусі саме ті завдання, які мають вирішуватись через інтерфейс: зрозумілий каталог, можливість швидкого вибору товару, фільтрація за категоріями, інтеграція із системами оплати та зворотний зв'язок.

1.3.1 Аналіз потенційних користувачів онлайн-сервісу «FreshFruit»

Під час створення інтернет-магазину «FreshFruit» важливо не просто зробити зручний сайт, а й чітко розуміти для кого він буде призначений. Знання аудиторії допомагає краще налаштувати візуальний стиль, зручність користування та спосіб спілкування з клієнтами [8].

Основними покупцями є жителі міста, віком приблизно від 25 до 45 років. Це активні люди, які цінують свій час і звикли купувати онлайн. Серед них є молоді батьки, що прагнуть забезпечити дітей свіжими фруктами, звичайні молоді люди, які прагнуть відчувати нові смаки та спробувати екзотичні фрукти, яких немає в звичайних супермаркетах, також офісні працівники, які шукають щось корисне для перекусу, і ті, хто свідомо слідкують за своїм здоров'ям. Не варто забувати і за старше покоління, частина з них тільки починає робити перші кроки в онлайн-покупках, тому вони можуть стати постійними клієнтами, якщо сервіс буде простим, інтуїтивним і без зайвих складнощів.

Окрему увагу варто приділити підліткам і молоді 14-20 років. Ця аудиторія активно користується смартфонами, звикла робити покупки в кілька кліків і чутлива до візуальної привабливості, тобто щоб було усе стильно та швидко.

Ще одна цікава група цільової аудиторії – це майстри-рукодільники. Які займаються створенням фруктових букетів і боксів на замовлення. Такий напрям зараз у тренді, особливо серед тих, хто просуває свої вироби в соціальних мережах. Ці люди зацікавлені в широкому виборі якісних фруктів та в купівлі

великих обсягах. Якщо надати їм гнучкі умови та стабільний сервіс, то можна налагодити співпрацю на довгий період.

Таке різноманіття цільових груп відкриває нові можливості. Якщо врахувати потреби кожного з них – від досвідченого онлайн-шопера до новачків, то можна створити справді зручну, привабливу й функціональну платформу, яка стане помічником у щоденних та особливих покупках для будь-кого.

1.3.2 Вивчення основних конкурентів та учасників ринку

На сьогоднішній день ринок онлайн-торгівлі продуктами, зокрема свіжими фруктами, активно зростає. У цьому сегменті вже є кілька помітних гравців, які задають напрямок для всіх інших. Тож основними гравцями є такі платформи як:

– Rozetka. Хоча цей сайт більше відомий технікою та товарами для дому, з часом вони додали і продукти, включаючи фрукти. На рисунку 1.1 зображено вікно з каталогом фруктів, на ньому одразу помітно безліч інформації, кнопок та фільтрів, які важко сприймаються людиною. Складність магазину полягає в тому, що інтерфейс орієнтований на широкий асортимент, а не на окрему нішу – фрукти. Щоб знайти потрібний продукт, іноді доводиться пройти через безліч фільтрів, підкатегорій, рекламних банерів та пропозицій від різних продавців. Це може збити з пантелику навіть досвідченого онлайн-покупця.

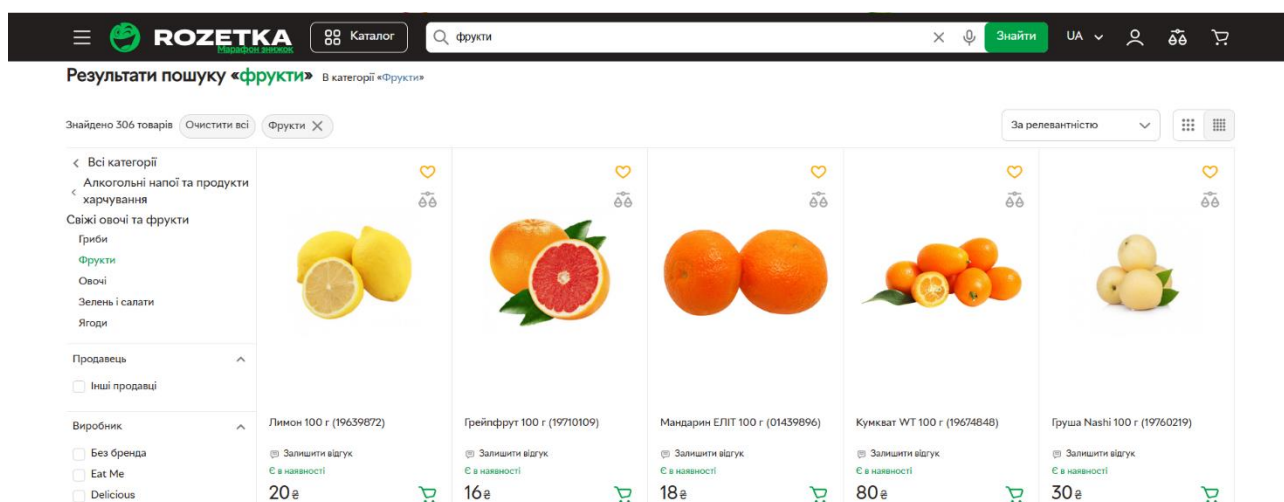


Рисунок 1.1 – Інтерфейс інтернет-магазину Rozetka

Ще один нюанс – модель маркетплейсу, яку використовує Rozetka. Дуже часто товари пропонуються не самою Rozetka, а сторонніми продавцями. Це ускладнює гарантію якості, а в категорії «Фрукти» це особливо критично. Один і той самий товар, одне і те саме зображення фрукту може мати кілька варіантів із різними цінами, строками доставки, умовами зберігання, і користувачеві важко швидко прийняти рішення.

– Fruit Shop. Це типове рішення для малого чи середнього бізнесу, яке не прагне охопити усі можливі категорії товарів, а зосереджується на вузькому напрямку – продажу свіжих фруктів.

На рисунку 1.2 зображено головну товарну сторінку сайту. Візуально вона виконана досить просто: кожен товар має чітке фото, вказану назву, вагу, ціну та окрему кнопку «Додати у корзину». Вся сторінка побудована у форматі плиток, які логічно впорядковані в сітку. Це створює відчуття структури й спрощує перегляд, особливо для користувача, який шукає щось конкретне. Інтерфейс лаконічний, без надмірної кількості вкладок. Присутність кнопки «Корзина» та номера телефону вгорі забезпечує зручний доступ до головних функцій.

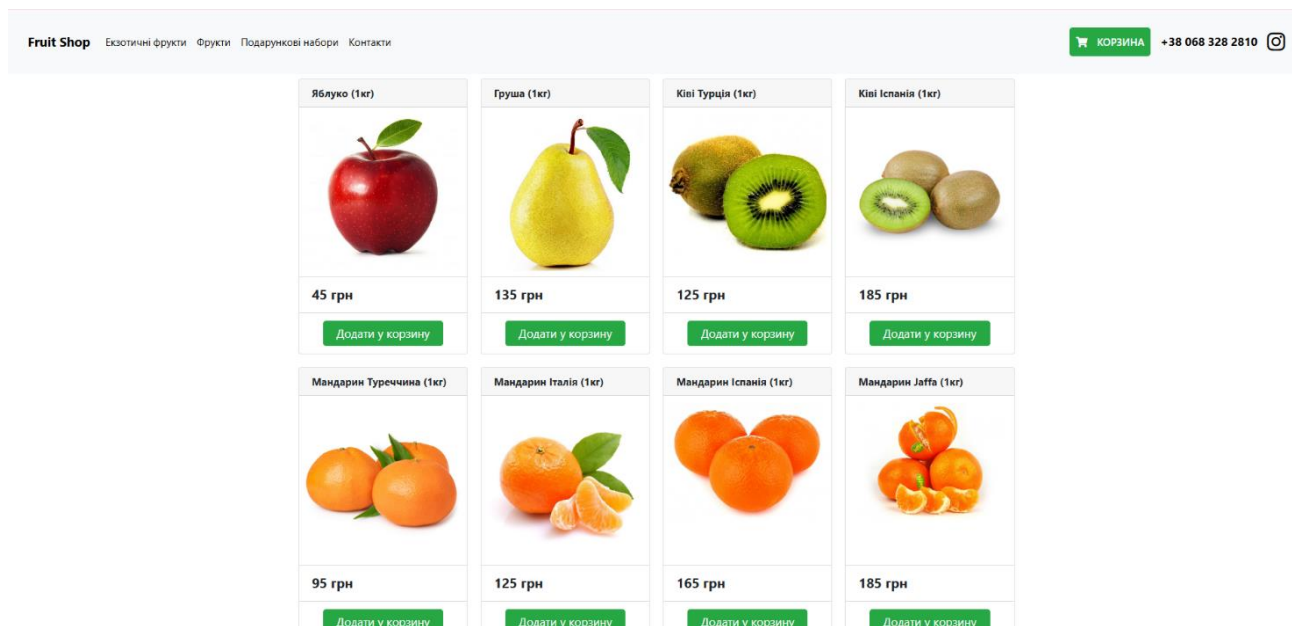


Рисунок 1.2 – Інтерфейс інтернет-магазину Fruit Shop

Проте, попри свою простоту, інтерфейс має низку обмежень. Наприклад, на сторінці відсутні фільтри. Це ускладнює пошук, якщо товарів буде більше, ніж кілька десятків. Також не вистачає коротких описів – наприклад, про смакові властивості, сорт або умови зберігання.

Ще одним недоліком є відсутність явних інструментів для персоналізації. Немає кабінету користувача, історії замовлень.

Перевагою сайту є його швидкість, чистий дизайн і мінімалістичний підхід, який дозволяє без проблем оформити замовлення навіть тим користувачам, які не надто впевнено користуються інтернетом.

– Glovo. Основна ідея платформи – доставляти будь-які товари максимально швидко, до 1 год. Glovo не є класичним інтернет-магазином, а посередником між покупцем та місцевими торговими точками.

На рисунку 1.3 зображено вікно, де зображено каталог з фруктами, можна побачити, що функціонал трішки складний, проте його не складно зрозуміти.

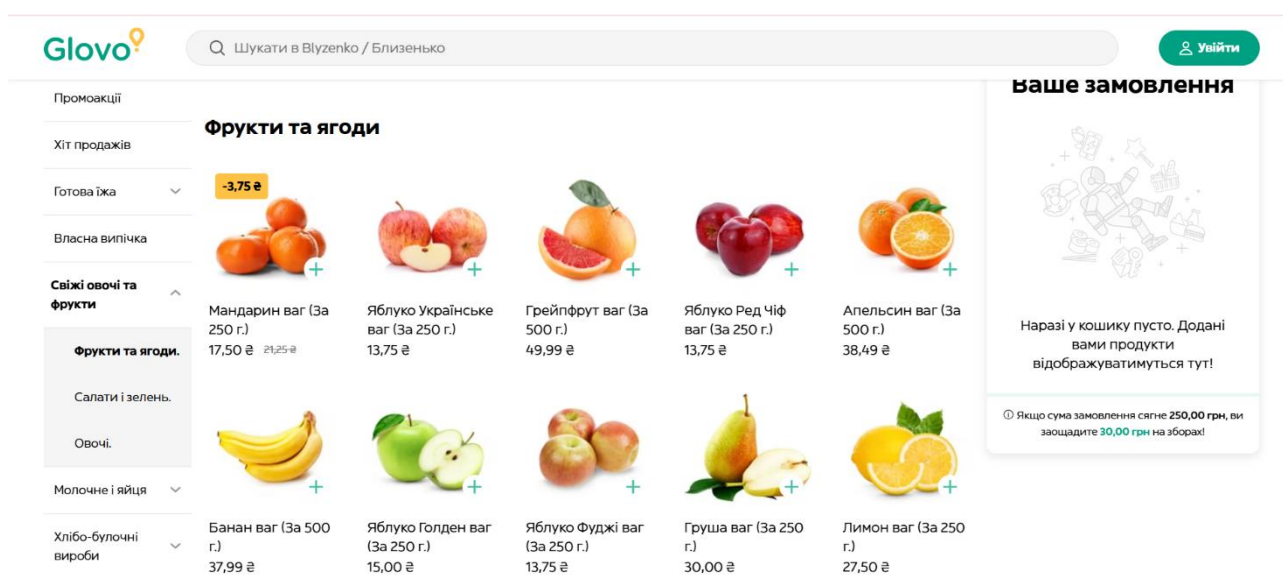


Рисунок 1.3 – Інтерфейс доставки Glovo

Через Glovo можна замовити продукти, у тому числі й фрукти. Проте вибір у цій категорії, як правило, обмежений найпопулярнішими позиціями – банани, яблука, апельсини. Про спеціалізацію на фруктовій продукції не йдеться: сервіс

не гарантує стабільного асортименту або якості, оскільки залежить від наявності в конкретному магазині.

Таким чином, Glovo більше підходить для швидких, спонтанних покупок, а не для планового придбання свіжих фруктів із широким вибором. Його функціонал не розрахований на детальне представлення продуктів, опис, або формування корзини з конкретних категорій, що є важливим у спеціалізованих онлайн-магазинах.

– FruitBoutique. Цей інтернет-магазин позиціонує себе як онлайн-майданчик з продажу екзотичних фруктів в Україні. Особливу увагу зроблено на незвичайні продукти, які складно знайти в звичайних супермаркетах. На рисунку 1.4 показано фрагмент каталогу сайту, де показані найпопулярніші товари.

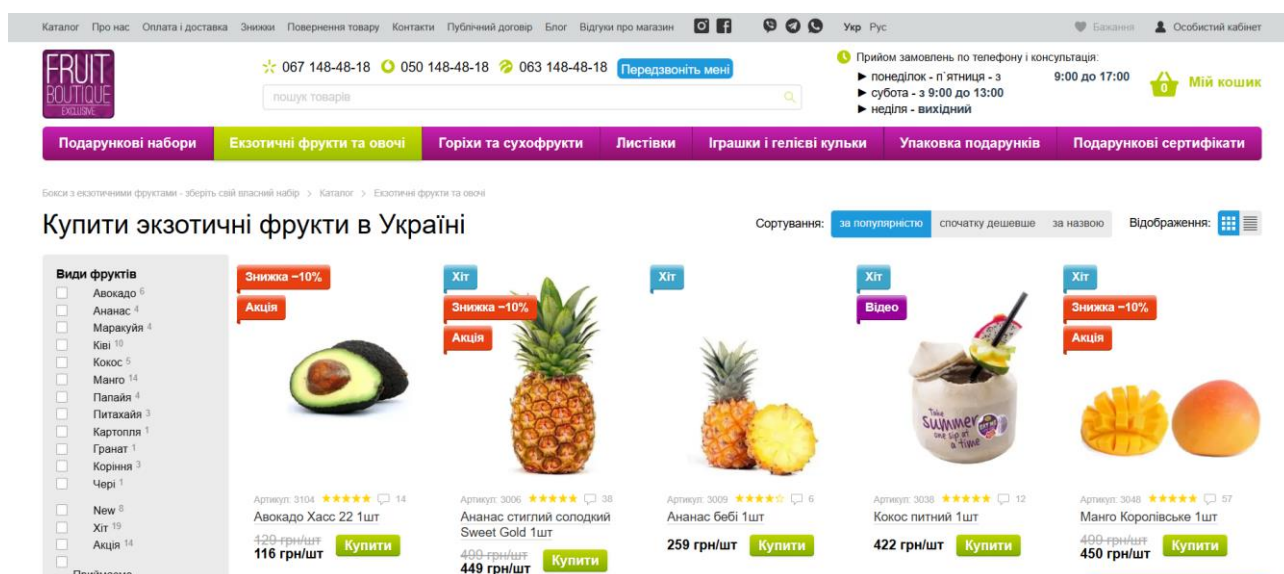


Рисунок 1.4 – Інтерфейс доставки FruitBoutique

Щодо вигляду, візуально сторінка оформлена яскраво, а структура побудована з урахуванням зручності – наявність фільтрів за видами фруктів, акційні пропозиції та позначки «хіт», що спрощує вибір. Цей сайт відрізняється сторінкою «подарункові набори», де кожен покупець може обрати вже готовий та сформований подарунок.

Функціонально сайт орієнтований на швидке замовлення: у кожного товару є фото, рейтинг, актуальна ціна, а також кнопка «Купити», що дозволяє здійснити покупку в кілька кліків.

Проте даний сайт має незначний недолік – перевантажене меню, яке складно сприймається людиною на перший погляд.

1.4 Постановка задачі, та визначення функціональних вимог до інтернет-магазину

Перед розробкою будь-якого вебресурсу важливо чітко сформулювати, яку саме проблему він має вирішувати, які завдання виконувати та які очікування має задовольняти. У випадку з «FreshFruit» йдеться не просто про онлайн-продаж фруктів, а мета значно глибша: забезпечити клієнтам зручний доступ до свіжої продукції, а бізнесу – сучасний інструмент для керування торговими процесами.

Враховуючи зростаючий попит на здорове харчування та тенденцію до онлайн-купівель, створення ефективного, привабливого та надійного вебсайту є логічним кроком у розвитку компанії. На цьому етапі важливо визначити стратегічні цілі, конкретні задачі, які ставляться перед розробкою, та сформулювати перелік вимог, що забезпечать якісну реалізацію задуманого проекту.

1.4.1 Основна мета розробки вебсайту «FreshFruit» та її відповідність потребам ринку

Основною метою створення вебсайту «FreshFruit» є формування ефективного онлайн-майданчик для продажу свіжих фруктів, який дозволить зручно замовляти продукти, отримувати актуальну інформацію про наявність товарів та взаємодіяти з компанією в онлайн середовищі.

Серед ключових цілей можна виокремити:

- Розширення ринку збуту шляхом охоплення нової онлайн-аудиторії [9].

- Створення простого та зручного сайту замовлення товарів для споживачів.
- Підвищення рівня магазину серед конкурентів бізнесу через впровадження сучасних цифрових рішень.
- Покращення обслуговування клієнтів завдяки автоматизованим процесам.
- Формування репутації надійної та стильної компанії.

1.4.2 Визначення ключових функцій інтрнет-магазину «FreshFruit»

Для того щоб інтернет-магазин «FreshFruit» повноцінно виконував покладені на нього функції, необхідно передбачити низку обов'язкових функціональних вимог [10], які забезпечать зручність, стабільність і ефективність роботи як користувачів, так і для продавців фруктів, її наведено в таблиці 1.1.

Таблиця 1.1 – Функціональні вимоги для «FreshFruit»

Функціональна вимога	Опис
1	2
Реєстрація та авторизація користувача	Можливість створити особистого кабінету з функцією входу/виходу в систему
Каталог товарів	Відображає асортимент фруктів із фотом, цінами
Фільтрація товарів	Можливість фільтрувати товари за алфавітом та за ціною
Кошик замовлення	Додавання товарів до кошика, редагування кількості та розрахунок вартості
Оформлення замовлення	Заповнення контактної особистої інформації, вибір оплати
Пошук по сайту	Швидкий пошук за назвою товару

Продовження таблиці 1.1

1	2
Зворотній зв'язок	Форма для звернення користувачів, тобто залишення відгуку
Адаптивний дизайн	Коректне відображення сайту на різних пристроях
Підписка на розсилку	Можливість отримувати e-mail повідомлення з акціями або персональними пропозиціями
Контактна інформація	Можливість робити замовлення або консультації за телефонним номером або електронною поштою

Запропоновані функціональні можливості є базовими та необхідними для ефективної роботи сучасного інтернет-магазину. Враховуючи це все при розробці вебсайту «FreshFruit» сприятиме підвищенню лояльності клієнтів, зростанню кількості замовлень і як наслідок – успішному розвитку бізнесу.

РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА ТЕХНІЧНЕ ПЛАНУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ «FRESHFRUIT»

2.1 Загальний підхід до побудови вебресурсу

Розробка вебсайту – це не лише процес написання коду, а передусім системний підхід до вирішення поставленої задачі. В основі створення будь-якої вебсторінки лежить чітке планування, вибір відповідної архітектури, а також грамотний розділ функціональності між окремими компонентами. У випадку інтернет-магазину фруктів «FreshFruit» основною метою є забезпечення швидкого, зручного та безпечного доступу користувача до замовлення продукції. Для цього необхідно створити ефективну систему взаємодії між інтерфейсом користувача та серверною логікою, яка зможе забезпечити обробку дій користувача в режимі реального часу, перевірку даних, збереження інформації та її безпечну передачу.

2.1.1 Вибір ефективної архітектури для інтернет-рішення

Для створення вебсайту обрано клієнт-серверну архітектурну модель. Це одна з найбільш перевірених і надійних структур у сфері розробки вебресурсів. Суть цієї моделі полягає у чіткому розумінні ролей між учасниками взаємодії: з одного боку знаходиться користувач, який ініціює запити, а з іншого – система, яка ці запити обробляє.

У межах клієнт-серверної архітектури користувач виконує певні дії, які система сприймає як запити. Ці запити передаються на сервер, де обробляються згідно з закладеною логікою. Далі система формує відповіді і повертає її користувачеві. Такий підхід дозволяє централізовано зберігати дані, забезпечити їх захист і цілісність, контролювати доступ а також гарантувати послідовність обробки запитів. [11]

На рисунку 2.1 зображено структурну схему взаємодії клієнтів із сервером та базою даних через інтернет.

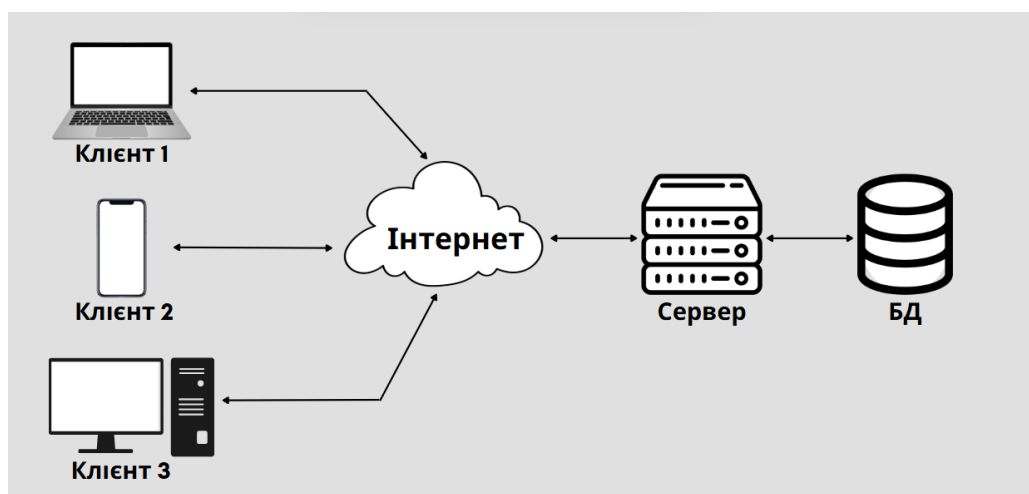


Рисунок 2.1 – Схема клієнт-серверної архітектури

Цю модель доцільно використовувати саме для інтернет-магазину з кількох причин:

- Вона забезпечує контроль над усіма діями, що відбувається в системі
- Дає можливість централізовано оновлювати інформацію про товари, ціни, наявність та замовлення.
- Один працівник має можливість сам обслуговувати декілька процесів одночасно і без особливих зусиль [12].
- Підвищувати рівень безпеки при роботі з конфіденційними даними, такими як контактна інформація, паролі в систему, адреси чи способи оплати.
- Дає змогу масштабувати систему у разі зростання кількості користувачів або розширення функціоналу.

Клієнт-серверна архітектура дозволяє уникнути дублювання логіки і непотрібного перенесення великих обсягів даних. Це робить систему швидшою більш структурованою і гнучкою до змін, що особливо важливо для проектів, орієнтованих на реальний ринок, де може змінюватися асортимент.

2.1.2 Розподілення ролей між клієнтською та серверною частинами

У сучасному світі веброзробки чітко розмежування обов'язків між клієнтською та серверною частинами є основою. Інтернет-магазин «FreshFruit», як типовий приклад системи електронної комерції, потребує раціонального підходу до організації цього поділу для забезпечення швидкої, коректної обробки запитів користувачів, стабільності та зручності користування.

Клієнтська частина відповідає за взаємодію з кінцевим користувачем – покупцем, який заходить на сайт через браузер або мобільний пристрій. У веб-магазині вона надає інтерфейс, через який користувач може робити будь-які дії на сайт, наприклад: реєстрація перегляд каталогу товарів і так далі. Клієнтська частина як правило реалізується за допомогою мов HTML, CSS, JavaScript та сучасних фреймворків.

Щоб добре зрозуміти і наочно показати взаємодію частини складної системи без зайвого коду та до його написання доцільно використати UML-діаграми [13].

На рисунку 2.2 зображено UML-діаграму варіантів використання, яка показує сценарій взаємодії клієнта з онлайн-магазином. Діаграма показує основні функції клієнта, а також відображає зв'язки між актором (клієнтом) і прецедентами (випадками використання системи).

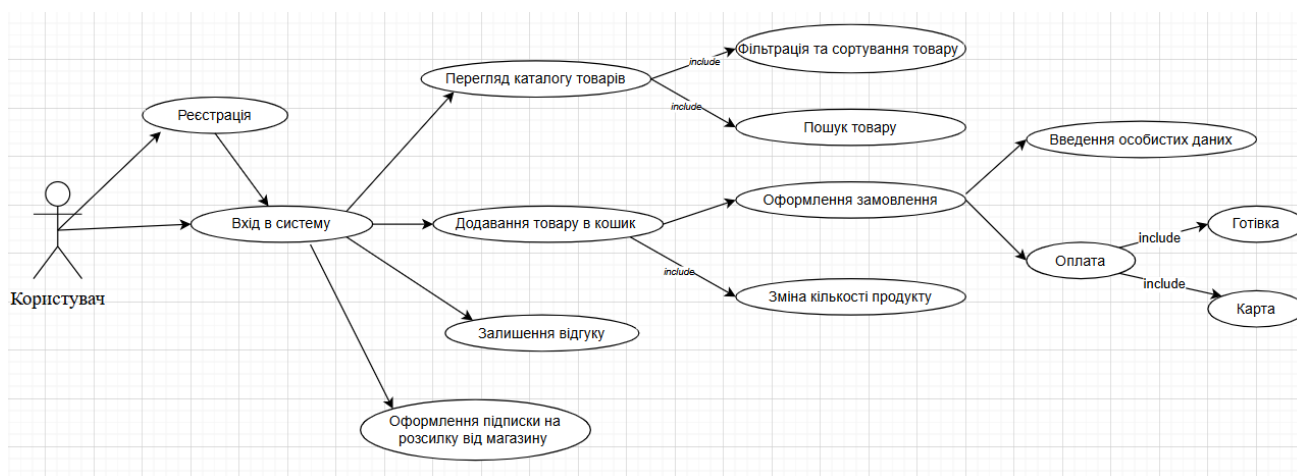


Рисунок 2.2 – Діаграма прецедентів покупця в онлайн-магазині

На діаграмі ми бачимо, що одразу при вході на сторінку відкривається вікно реєстрації, де можна або зареєструватися або якщо вже є наявний акаунт то просто увійти в свій профіль.

Після входу в систему користувач може:

- Переглядати каталог товарів, який включає в себе можливість фільтрувати, сортувати та робити пошук продукту, в моєму випадку фрукту.
- Керування кошиком – додавання, видалення та зміна кількості товарів .
- Якщо в кошику є хоча б один продукт, то покупець має можливість оформити замовлення, де потрібно вводити свої особисті дані (адреса для доставки, прізвище) та поля з вибором оплати (карта чи готівка)
- Після проведення замовлення є можливість залишити відгук на сайті
- Оформлення підписки на розсилку від магазину – поля з вводом електронної пошти, на яку будуть надсилатися акції та персональні пропозиції.

Всі ці дії відбуваються через інтерфейс клієнтської частини, але потребують обробки на стороні сервера. Відповідно кожна дія супроводжується HTTP (Hypertext Transfer Protocol)-запитом, що передається на сервер, а той – повертає результат у вигляді відповіді.

Серверна частина – це логічне ядро системи, що виконує всю «невидиму» роботу: обробку замовлень, збереження даних, перевірка прав доступу, комунікацію з платіжними сервісами, логіку керування товарами, обробку відгуками та інше.

Основні компоненти серверної частини:

- База даних: зберігає інформацію про товари, користувачів, замовлення, відгуки.
- API (Application Programming Interface) – сервер: обробляє запити клієнтів, взаємодіє з базою даних та іншими серверами.
- Модуль автентифікації: перевіряють права доступу та сесії користувачів.
- Система логів: фіксує критичні події та помилки.

Одну із ключових ролей у серверній частині відіграє адміністратор, який має доступ до широкого спектра дій, необхідних для підтримки ефективної роботи онлайн-магазину. Його функції зображено на UML-діаграмі з рисунку 2.3 нижче.

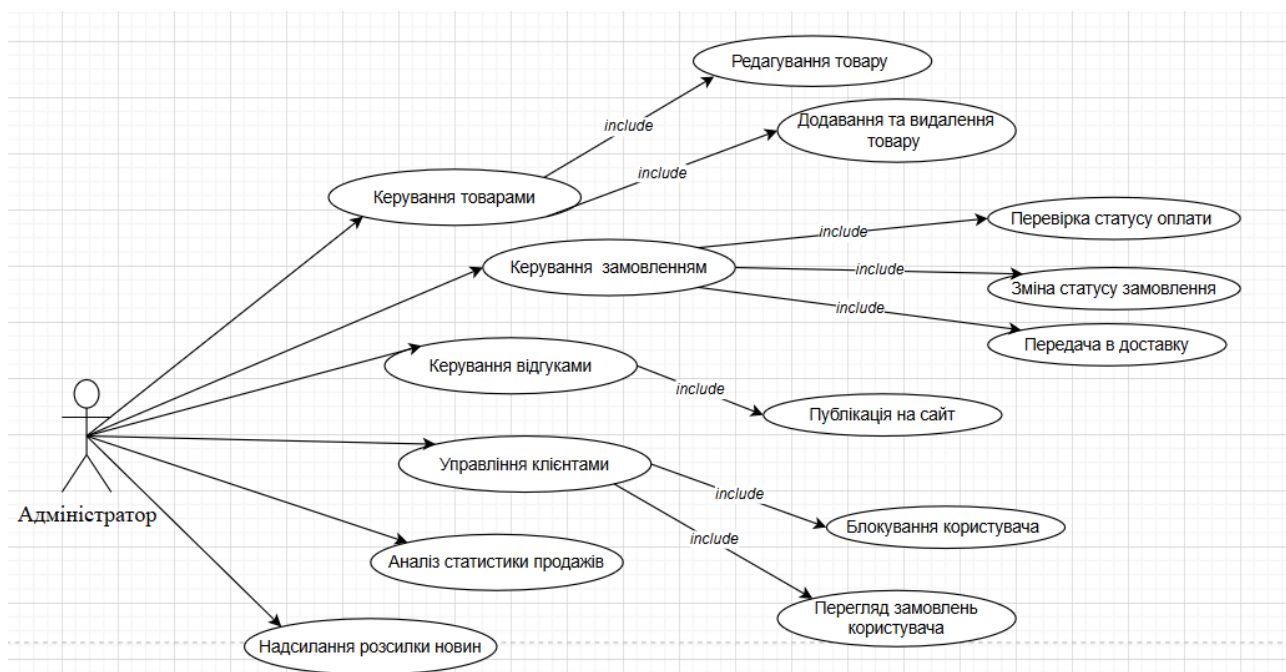


Рисунок 2.3 – Діаграма прецедентів адміністратора в онлайн-магазині

На діаграмі бачимо актора-адміністратора, який виконує такі дії:

- Керування товарами. Адміністратор має змогу додавати нові товари з усіма необхідними атрибутами: назвою, ціною, фото, категорією. За потреби також можна швидко змінити наявні дані, наприклад оновити ціну, змінити статус, прибрати його з продажу.
- Керування замовленнями. Можливість перевірити оплату, змінити статус: «в обробці», «відправлено», «отримано» та «скасовано» і вже після цього адміністратор передати замовлення до служби доставки.
- Управління клієнтами. Система дозволяє переглядати список клієнтів, їхню активність та історію покупок. При виявленні підозрілої поведінки(наприклад, не припустимих відгуків, багато негативних відгуків без оформлення замовлення) його можна заблокувати.

- Аналіз статистики продажів. Можливість адміністратора проводити щомісячну статистику на різні попити продуктів та будування для візуалізації динаміки продажів графіки, діаграми. Це може краще вплинути на маркетингову діяльність.

- Керування відгуками. Адміністратор має змогу перевірити нові відгуки перед публікацією, видаляти недоречні або образливі, відповідати клієнтам від імені компанії.

2.2 Добір інструментів реалізації інтернет-магазину FreshFruit

Під час розробки вебсайту дуже важливо обрати не найновіші або найпоширеніші інструменти, а потрібно підібрати ті, які зможуть забезпечити технічну реалізацію функціоналу. У межах даного проекту було прийнято рішення використовувати кластерний стек веб-технологій, який поєднує перевірені часом мови та інструменти, адаптовані під потреби малого та середнього бізнесу.

Реалізацію клієнтської частини здійснено за допомогою мов HTML та CSS, які забезпечують структуру та візуальне оформлення сторінок. Для створення інтерактивності і реагування інтерфейсу на дії користувача використовується JavaScript. Серверна логіка розроблена на мові PHP, що є ефективною у зв'язці з обраною системою управління базами даних MySQL. Це дозволяє надійно зберігати та обробляти інформацію користувачів, результати запитів і цифрових оцінок.

У якості фреймворку для прискорення верстки та зберігання адаптивності застосовано Bootstrap. Хостинг-платформою було обрано The Host – сервіс з підтримкою PHP. Такий набір засобів дозволяє створити повноцінний веб-продукт із мінімальними витратами на підтримку, забезпечуючи при цьому якісний користувацький досвід.

Такий підхід досліджено при аналізі технологій веб-розробки клієнт-серверного рішення для інтернет-комерції, які представлені у тезах про

дослідження, які були опубліковані у матеріалах VIII Міжнародної студентської науково-технічної конференції «Природничі та гуманітарні науки. Актуальні питання та подані в додатку А кваліфікаційної роботи [14].

Наступні підрозділи детально обґрунтовують вибір кожного інструменту окремо.

2.2.1 Вибір мов розмітки, стилів (HTML, CSS) і фреймворку (Bootstrap) для побудови інтерфейсу

Жоден веб-застосунок не може існувати без основ: структури та оформлення. У цьому проекті основою візуального представлення стала пара: HTML та CSS. Таке поєднання давно зарекомендувало себе як універсальне рішення, які підтримується всіма браузерами, легко адаптується та має низький поріг для розробників початківців [15].

HTML5 використовується як головний інструмент для побудови логічної структури веб-сторінки. Саме ця версія HTML дозволяє використовувати семантичні теги (наприклад, <section>, <article>, <nav>, <header>, <footer>), що не лише покращує читабельність коду, але і спрощує SEO-оптимізацію та доступність для користувачів із особливими потребами [16].

З іншого боку, CSS відповідає за візуальне оформлення інтерфейсу. Його використання дозволяє створити адаптивну верстку, застосувати анімації, трансформації, тіні, кольорові переходи, макети з Flexbox та Grid Layout – тобто ті інструменти, що забезпечують сучасний вигляд ресурсу на будь-якому пристрої [17].

У таблиці 2.1 зібрано переваги та недоліки мов програмування стилів. Після цього можна буде краще зрозуміти інші технології та переконатися що саме для інтернет-магазину «FreshFruit» підходять саме HTML та CSS.

Таблиця 2.1 – Порівняльна таблиця мов розмітки та стилів

Інструмент	Переваги	Недоліки
HTML + CSS	Простота, підтримка браузером, адаптивність	Потребує доповнення JS для логіки
XML	Складна структура, не адаптована під UI	Висока складність
Markdown	Зручно для статей, документації	Не підходить для повноцінних сайтів

На додачу до CSS, у проєкті використовується фреймворк Bootstrap. Він дозволяє максимально пришвидшити процес верстки завдяки готовим компонентам: кнопкам, формам, модальним вікнам, сіткам, навігації та інше. Це суттєво знижує кількість коду, підвищує стабільність візуальних елементів та забезпечує автоматичну адаптацію інтерфейсу під різні роздільні здатності екранів. Bootstrap також спрощує реалізацію темної або світлої тем, сіткових макетів та забезпечує однорідний стиль сторінок.

2.2.2 Використання JavaScript для побудови динамічних елементів

Коли йдеться про інтерактивність – JavaScript залишається безальтернативним інструментом. Він не просто оживляє сторінку, а дозволяє реалізовувати логіку на боці клієнта: перевірка форм, динамічні підказки, обробку подій (натискання, наведення, прокрутка), завантаження даних без перезавантаження сторінки (через Fetch/AJAX) та багато іншого [18].

У інтернет-магазині «FreshFruit» JavaScript відповідає за:

- Перевірку правильності введення користувацьких даних ще до їх надсилання на сервер. Наприклад одразу перевіряє чи введено email правильно чи не пропущено обов'язкових полів. Це економить час користувача і знижує навантаження на сервер, бо некоректні запити просто не відправляються
- Відображення повідомлень про помилки чи успішну дію.

- Реалізацію інтерактивних елементів такі як випадаючі списки, каруселі або таби
- Анімацію завантаження або переходу між елементами сторінки.

Ще одним плюсом JS є можливість працювати з DOM (Document Object Model) – структурованим представленням HTML-документа. Це дозволяє швидко змінювати вміст сторінки без повного оновлення.

На рисунку 2.4 зображено структура об'єктної моделі документа

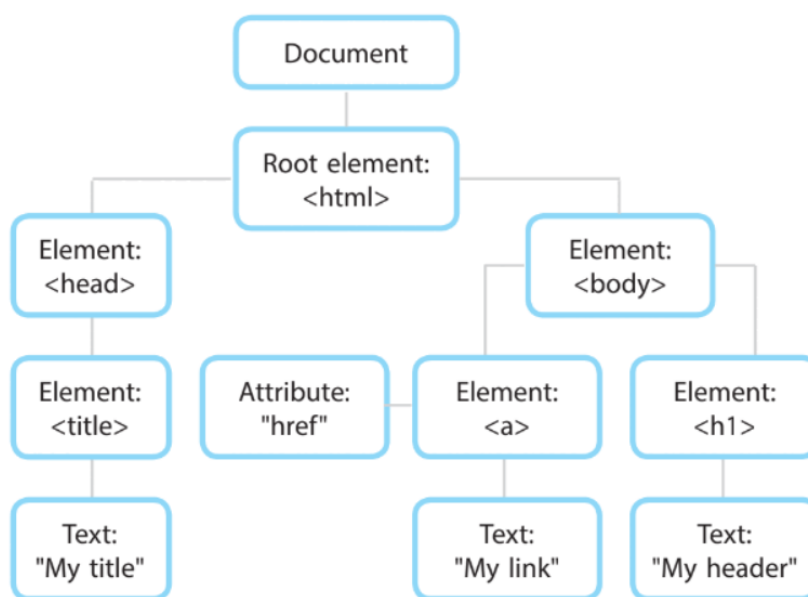


Рисунок 2.4 – Дерево вузлів DOM

DOM не є мовою програмування, але без нього JavaScript не мав би уявлення про вебсторінку, HTML-документ [19].

2.2.3 Застосування PHP для серверної логіки

Для обробки запитів, роботи з базою даних і реалізації всієї бізнес-аналітики застосовується PHP – один з найпоширеніших мов програмування для серверної частини веб-застосунків. Його популярність пояснюється простотою, широкою спільнотою підтримки, великої кількості готових рішень та ідеальної сумісності з MySQL [20].

У інтернет-магазині «FreshFruit» PHP виконує низку ключових функцій:

- Приймає дані з форм через \$_POST, \$_GET.
- Проводить серверну перевірку введеної інформації.
- Взаємодіє з базою даних – зчитує, змінює, додає або видаляє записи.
- Обробляє сесії користувачів, авторизацію, реєстрацію та інше.

PHP легко розгортається на більшості хостингів, у моєму випадку «The Host», не потребує складних налаштувань і дозволяє створювати динамічні сторінки.

2.2.4 Обґрунтування вибору бази даних для збереження інформації

Для зберігання, обробки та аналіз даних у веб-застосунку обрана реляційна система керування базами даних MySQL. Вона забезпечує стабільну роботу з великим обсягом структурованої інформації, має високу продуктивність та зручний інструментарій для запитів [21].

База даних містить ключові сутності, які пов'язані між собою: наприклад, таблиці користувачів, анкети, відповіді, результати цифрового аналізу. Для доступу до цих даних використовується SQL-запити, які PHP генерує динамічно [22].

Переваги використання MySQL:

- Оптимізована для роботи з веб-програмами.
- Підтримує транзакції, індексацію, зовнішні ключі.
- Підтримка більшості платформ [23].
- Має візуальні інструменти для адміністратора (phpMyAdmin)/
- Безкоштовна та з відкритим вихідним кодом
- Дані зберігаються у форматі, що дозволяє легко аналізувати їх, будувати звіти або експортувати у зручні формати для зовнішньої обробки (наприклад JSON).

2.2.5 Технічні критерії вибору хостингової платформи

Хостинг- це послуга з надання місця на сервері, що має постійний доступ до мережі Інтернет, для зберігання вебсайту та забезпечення його доступності для користувачів у будь-який час. Простими словами, хостинг – це «дім» для вебсайту. Саме там зберігаються всі файли, бази даних, скрипти та інші ресурси, які забезпечують роботу вебсайту [24].

Після аналізу ринку хостингових послуг в Україні було відібрано кілька варіантів, які відповідали базовим вимогам.

У таблиці 2.2 подано порівняння з основними параметрами декількох хостингів.

Таблиця 2.2 – Порівняльна таблиця хостинг-провайдерів

Критерії	The Host	Tuthost	HostPro
Підтримка PHP	До PHP 8.2	До PHP 8.1	До PHP 7.4
База даних MySQL	Необмежено	3	1
Безкоштовний SSL	Так	Так	Ні
Резервне копіювання	Автоматичне	Вручну	Частково
Панель керування	cPanel	ISPmanager	Своя панель
Час доступності підтримки	24/7	Робочі години	До 20:00
Ціна на рік	650 грн.	720 грн.	850 грн.

Хостинг «The Host» був обраний завдяки оптимальному поєднанню технічних характеристик, безпеки, надійності та вартості.

2.3 Використання UI/UX для сайту FreshFruit

Інтерфейс користувача – це перше, що бачить відвідувач, і те що формує враження ще до того як він зробить перший клік. У випадку «FreshFruit» було застосовано сучасні принципи UX/UI-дизайну, орієнтовані на простоту сприйняття, логіку навігації та візуальну естетику, що викликає довіру й бажання залишитися на сторінці довше.

Під час розробки макетів ключовим орієнтиром став рівень інтуїтивності – кожен елемент інтерфейсу повинен бути розміщений з урахуванням логіки користувацької поведінки. Також було вибрано розробити односторінковий сайт інколи зі спливаючими вікнами. Така структура забезпечує користувача усією необхідною інформацією без потреби додаткових переходів, що позитивно впливає на показники утримання конверсії [25].

Вибір кольорової палітри та типографіки є одним з найважливіших аспектів у створенні вебсайту. Правильно підібрані кольори не лише роблять сайт привабливим зовні, але й формують емоційне сприйняття користувача, впливають на його настрій та поведінку. Кольори можуть викликати різні асоціації – від відчуття довіри і спокою до енергійності та натхнення – тому їх вибір повинен відповідати загальній концепції бренду і цільової аудиторії, у випадку «FreshFruit» - було обрано саме спокій та легкість [26]. Крім того, контрастність кольорів має велике значення для зручності сприйняття інформації, адже вона впливає на читабельність тексту та видимість важливих елементів інтерфейсу [27].

Основним кольором, який задається всьому візуальному сприйняттю інтернет-магазину фруктів, є відтінок #fdfaf (рис. 2.6) – ніжний, світлий, майже кремовий. Цей колір виконує роль фону, створюючи відкритості і легкості. Світлий фон є ідеальним полотном для демонстрації продуктів, особливо яскравих фруктів, оскільки він не відволікає увагу, а навпаки підкреслює природні кольори товарів, роблячи їх більш привабливими [28]. Ніжність цього відтінку викликає теплі, заспокійливі емоції, що сприяє тривалому перебуванні

на сайті. Відсутність надмірної насиченості фону також робить навігацію більш комфортною для очей. Також цей колір символізує натуральність і екологічність, що є важливою складовою іміджу свіжих фруктів.

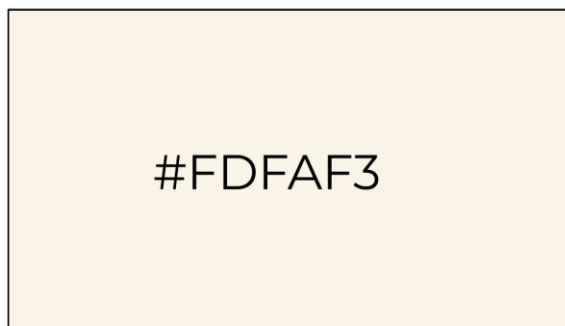


Рисунок 2.6 – Вигляд бежевого кольору

Контрасним елементом у дизайні виступає темно-зелений колір #2E6D4D (рис. 2.7), який використовується для заголовків, кнопок, акцентних деталей і меню. Цей відтінок символізує природність, здоров'я, екологічність і свіжість, що повністю відповідає тематиці продукту [29]. Завдяки такому поєднанні кольорів – світлому теплому фону і глибокому насиченому зеленому – інтерфейс виглядає збалансовано і гармонійно, викликає позитивні емоції та створює атмосферу, яка спонукає до покупки натуральних продуктів.

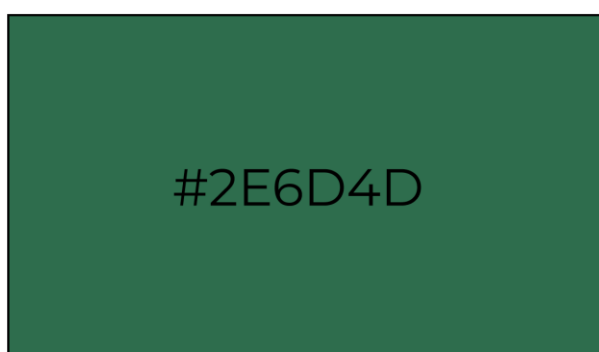


Рисунок 2.7 – Вигляд темно-зеленого кольору

Типографіка, у свою чергу, є не менш значущою складовою дизайну. Шрифти передають стиль, характер і настрій сайту, допомагають структуру інформацію, роблять її легшою для сприйняття і забезпечують комфортне

читання. Вдалий вибір шрифтів дозволяє ефективно розподілити акценти між заголовками, підзаголовками і основним текстом, підвищуючи загальну «юзабіліті» сайту. Крім того, типографіка впливає на впізнаваність бренду і може стати його важливою візуальною складовою [30].

Основним шрифтом є Montserrat (рис. 2.8) – сучасний геометричний шрифт із відкритими, чіткими формами, який відомий своєю гарною читабельністю навіть при невеликих розмірах тексту. Montserrat ідеально підходить для заголовків і підзаголовків, додаючи дизайну строгість, легкість сприйняття і водночас деяку елегантність.

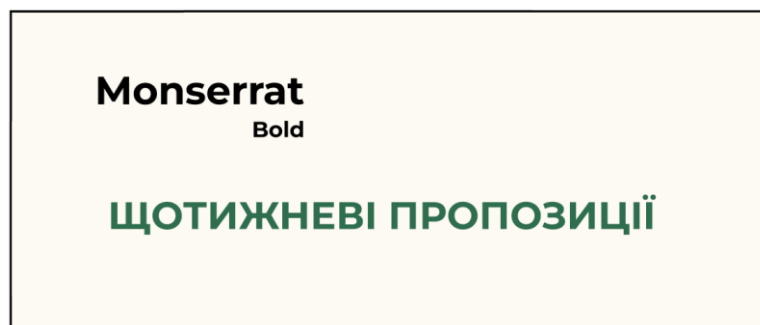


Рисунок 2.8 – Зображення шрифту Montserrat

Для основного тексту застосовується шрифт sans-serif, який має простий, чистий вигляд і забезпечує комфортне читання великих обсягів інформації (рис. 2.9). Sans-serif шрифти візуально легкі, не перевантажують сприйняття і добре поєднуються з Montserrat, створюючи баланс між акцентованим заголовком і деталізованим текстовим наповненням. Це сприяє легкому і природному сприйняттю інформації, що особливо важливо для інтернет-магазину, де користувачі прагнуть швидко знаходити потрібні товари і читати опис без зайвого напруження очей.



Рисунок 2.9 – Зображення шрифту Sans Serif

Таким чином, дизайн сайту є прикладом впровадження сучасних трендів веб-дизайні, поєднуючи в собі естетику і комфорт користувача. Ретельний вибір кольорової палітри та шрифтів створює сприятливі умови для довготривалої взаємодії клієнтів із сайтом, що в кінцевому результаті позитивно впливає на продажі та імідж компанії.

2.4. Проектування бази даних

У процесі розробки інтернет-магазину з продажами фруктів одним із ключових етапів стало проектування, створення та реалізація бази даних. Сама база даних є основою всього функціонування сайту, оскільки в ній зберігається критично важлива інформація – від даних користувачів і замовлень до описів товарів, відгуків і підписок на розсилку. Ретельне продумування структури, взаємозв'язків та логіки взаємодії між таблицями дозволяє забезпечити безпечну роботу ресурсів.

В основі будь-якої реляційної бази даних лежать таблиці, які зберігають інформацію у вигляді рядків і стовпців. Кожна таблиця відображає окремий тип об'єктів предметної області (наприклад, користувачі, товари, замовлення).

Ключовим моментом проектування є визначення типів зв'язків між таблицями:

– Зв’язок «один до одного» – кожен запис в одній таблиці відповідає одному записі в іншій. Приклад з FreshFruit: кожен користувач має один унікальний профіль.

– Зв’язок «один до багатьох» – один запис у таблиці А може відповідати багатьом записам у таблиці Б, але кожен запис у таблиці Б відповідає лише одному запису у таблиці А. Приклад з FreshFruit: один користувач може зробити багато замовлень.

– Зв’язок «багато до багатьох» – записи в одній таблиці можуть відповідати багатьом записам в іншій і навпаки. Для цього реалізація такого зв’язку створюється проміжна таблиця. Приклад з FreshFruit: товар може входити у багато замовлень, а одне замовлення містить багато товарів [31].

Детальніше про зв’язки, де і який використовується в інтернет-магазині фруктів буде розглянуто під час побудови ER-діаграми.

Усього було створено шість функціональних таблиць: *users*, *products*, *orders*, *orders_products*, *reviews* і *subscribers*. Кожна з них виконує конкретну роль, логічно ізольована за своїм призначенням, але при цьому тісно пов’язана з іншими через ключі. Завдяки цьому забезпечується цілісність і зв’язність усієї інформаційної структури магазину. Нижче у таблицях 2.3 – 2.7 наведено опис кожної таблиці БД з поясненням полів та їх типів.

Таблиця 2.3 – Таблиця БД для користувачів, *users*.

Поле	Тип даних	Опис
id	int(11)	Унікальний ідентифікатор
name	varchar(30)	Ім’я користувача
email	varchar(200)	Електронна пошта
password	varchar(200)	Проль

Таблиця користувачів зберігає основну інформацію про зареєстрованих користувачів, забезпечуючи авторизацію та ідентифікацію в системі.

Таблиця 2.4 – Таблиця БД для підписки, *subscribers*.

Поле	Тип даних	Опис
id	int(11)	Ідентифікатор
email	varchar(200)	Електронна пошта

Таблиця *subscribers* використовується для зберігання адрес електронної пошти користувачів, які оформляли підписку на новини та розсилки.

Таблиця 2.5 – Таблиця БД для замовлення, *orders*.

Поле	Тип даних	Опис
id	int(11)	Ідентифікатор
details	text	Список товарів
summary	decimal(6,2)	Підсумкова вартість
city	varchar(200)	Місто доставки
street	varchar(200)	Вулиця доставки
house	int(11)	Номер будинку
id_user	int(11)	Ідентифікатор користувача

Таблиця фіксує всі замовлення, включаючи дані доставки та вартість. Це дозволяє системі обробляти покупки та логістику.

Таблиця 2.6 – Таблиця БД для відгуків, *reviews*.

Поле	Тип даних	Опис
id	int(11)	Ідентифікатор
name	varchar(200)	Ім'я автора
text	text	Текст відгуку

Зберігає користувацькі відгуки про товар або сервіс. Ця таблиця дозволяє формувати довіру до продукту шляхом відкритих оцінок.

Таблиця 2.7 – Таблиця БД для товарів, *products*.

Поле	Тип даних	Опис
id	int(11)	Ідентифікатор
name	varchar(200)	Назва товару
price	Decimal(6,2)	Ціни без знижки
Sale_price	Decimal(6,2)	Ціни зі знижкою
photo	text	Посилання на зображення

Таблиця містить дані про всі товари, доступні для покупки. Вона забезпечує виведення продукції на сайті з урахуванням ціни, акції та фотографії.

Для того щоб створити дані таблиці в БД використовуються за допомогою SQL-запити. У лістингу 2.1 наведено приклад цього ж запиту для створення таблиці користувачів.

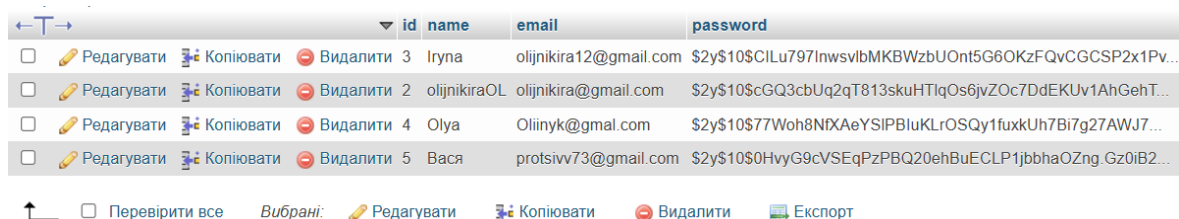
Лістинг 2.1 – SQL-запит для таблиці користувачів

```
CREATE TABLE users (
  id INT(11) NOT NULL AUTO_INCREMENT,
  name VARCHAR(30) NOT NULL,
  email VARCHAR(200) NOT NULL UNIQUE,
  password VARCHAR(200) NOT NULL
  PRIMARY KEY (id)
);
```

У цій таблиці визначено чотири поля. Поле *id* є цілим числом довжиною до 11 знаків, але не може містити проміжних значень і автоматично збільшується з кожним новим записом – користувачем (AUTO_INCREMENT). Це поле також є первинним ключем (PRIMARY KEY), тобто забезпечує унікальність коного користувача. Поле *name* призначене для зберігання імені користувача у форматі текстового рядка довжиною 30 символів, тут є присутній атрибут, який означає що це обов'язкове поле для заповнення (NOT NULL). Аналогічно, поле *email* також є текстовим і обов'язковим, але додатково має обмеження унікальності (UNIQUE), що унеможлиблює реєстрацію кількох користувачів з однаковою електронною адресою. Пароль користувача зберігається у полі *password*, яке

також має формат текстового рядка довжиною до 200 символів і не допускає порожніх значень.

Приклад вже створеної та заповненої таблиці в БД зображено на рисунку 2.10, де зображено всі поля, які описувалися вище та додатково вигляд хешованого паролю, який забезпечує захист особистих даних від проникнення шахраїв [32].



	id	name	email	password
☐ Редагувати Копіювати Видалити	3	Iryna	olijnikira12@gmail.com	\$2y\$10\$CILu797InwsvlbMKBWzbUOnt5G6OKzFQvCGCSP2x1Pv...
☐ Редагувати Копіювати Видалити	2	olijnikiraOL	olijnikira@gmail.com	\$2y\$10\$cGQ3cbUq2qT813skuHTlqOs6jvZ0c7DdEKUv1AhGehT...
☐ Редагувати Копіювати Видалити	4	Olya	Oliinyk@gmail.com	\$2y\$10\$77Woh8NfXAeYSIPBluKlROSQy1fuxkUh7Bi7g27AWJ7...
☐ Редагувати Копіювати Видалити	5	Вася	protsiiv73@gmail.com	\$2y\$10\$0HvyG9cVSEqPzPBQ20ehBuECLP1jbbhaOZng.Gz0iB2...

↑ ☐ Перевірити все Вибрані: ☐ Редагувати ☐ Копіювати ☐ Видалити ☐ Експорт

Рисунок 2.10 – Приклад таблиці users

Варто розглянути ще хоча б один SQL-запит для кращого розуміння створення таблиці в БД. Для реалізації функціональності інтернет-магазину особливо важливо забезпечити належну організацію даних, що стосується товарів. Тому у лістингу 2.2 показано запит саме для таблиці products.

Лістинг 2.2 – SQL запит для таблиці products

```
CREATE TABLE products (
  id INT(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(200) NOT NULL,
  price DECIMAL(6,2) NOT NULL,
  sale_price DECIMAL(6,2) NOT NULL,
  photo TEXT NOT NULL
);
```

В цій таблиці все аналогічно як в минулій, яка розглядалася. Поле *id* забезпечує унікальність кожного запису й автоматично збільшується. Назва товару зберігається в полі *name*. Особливу увагу варто звернути на поля *price* та *sale_price*, які мають тип DECIMAL (6,2). Це дозволяє зберігати ціни з точністю до сотих часток гривні (наприклад, 99.99), що важливо для коректного обліку

вартості та знижок. Поле *photo* містить посилання на зображення товару, яке забезпечує візуальне представлення продукції на сайті.

Приклад вже створеної та заповненої таблиці продуктів в БД можна побачити на рисунку 2.11.

	id	name	price	sale_price	photo
☐ Редагувати ☐ Копіювати ☐ Видалити	1	Абрикос, 1кг	89.99	89.99	персик.jpg
☐ Редагувати ☐ Копіювати ☐ Видалити	2	Чорниця, 1кг	250.99	250.99	чорниця.png
☐ Редагувати ☐ Копіювати ☐ Видалити	3	Грейпфрут, 1кг	69.99	69.99	грейпфрут.jpeg
☐ Редагувати ☐ Копіювати ☐ Видалити	4	Пітахая, 1шт	39.99	39.99	пітахая.webp
☐ Редагувати ☐ Копіювати ☐ Видалити	5	Виноград, 1кг	110.95	110.95	виноград.jpg
☐ Редагувати ☐ Копіювати ☐ Видалити	6	Лайм, 1кг	209.99	209.99	лайм.png
☐ Редагувати ☐ Копіювати ☐ Видалити	7	Ківі Червоне, 1шт	69.99	69.99	Ківі червоне.png
☐ Редагувати ☐ Копіювати ☐ Видалити	8	Лимон, 1кг	75.99	75.99	лимон.jpg
☐ Редагувати ☐ Копіювати ☐ Видалити	9	Банани, 1кг	39.99	39.99	banana.webp
☐ Редагувати ☐ Копіювати ☐ Видалити	10	Черешня, 1кг	122.99	122.99	черешня.webp
☐ Редагувати ☐ Копіювати ☐ Видалити	11	Ківі, 1кг	110.99	110.99	ківі.jpg
☐ Редагувати ☐ Копіювати ☐ Видалити	12	Гранат, 1кг	191.99	191.99	гранат.jpg
☐ Редагувати ☐ Копіювати ☐ Видалити	13	Манго, 1кг	223.99	223.99	манго.webp
☐ Редагувати ☐ Копіювати ☐ Видалити	14	Кавун, 1кг	108.99	108.99	кавун.webp
☐ Редагувати ☐ Копіювати ☐ Видалити	15	Слива, 1кг	215.99	215.99	слива.jpg
☐ Редагувати ☐ Копіювати ☐ Видалити	16	Полуниця Ананасова, 1кг	325.99	325.99	Strawberry pineapple.png

Рисунок 2.11 – Приклад таблиці products

Після створення цих таблиць інші частини бази даних створюються за таким самим принципом. Єдина логіка побудови структури допомагає підтримувати порядок у даних, полегшує роботу з ними та дозволяє легко змінювати або розширювати систему в майбутньому.

З метою побудови надійної і логічно цілісної бази даних для інтернет магазину, на етапі проектування було створено ER-діаграму (Entity-Relationship), яка відображає основні сутності, їх атрибути та вза'ємозв'язки. Побудова такої діаграми – це одна з найважливіших етапів під час роботи з базою даних, адже вона дозволяє візуально побачити структуру системи, оцінити логіку взаємодії між об'єктами й уникнути типових помилок, пов'язаних з надмірним дублюванням або суперечливими зв'язками [33].

На зображенні 2.12 представлено ER-діаграму, яка була побудована у процесі реалізації FreshFruit.

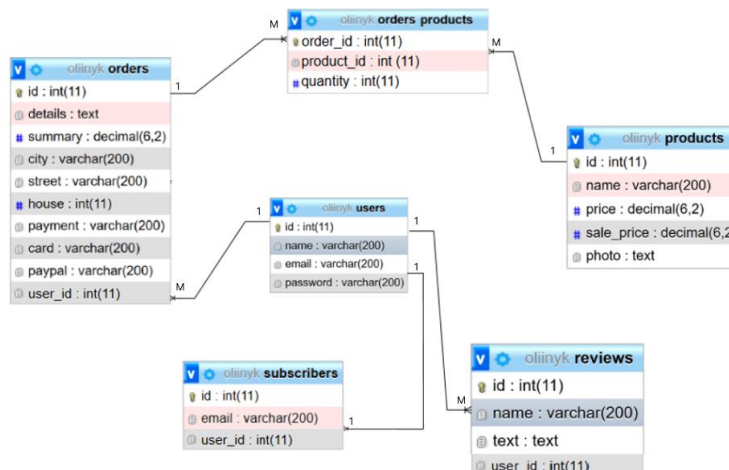


Рисунок 2.12 – ER діаграма структури бази даних для інтернет-магазину FreshFruit

З даної діаграми можна побачити всю структуру взаємозв'язків між таблицями. Для кращого розуміння огляд було занесено у таблицю 2.8.

Таблиця 2.8 – Зв'язки між сутностями бази даних

Сутність 1	Сутність 2	Тип зв'язку	Опис
users	orders	Один до багатьох	Один користувач може мати кілька замовлень
users	reviews	Один до багатьох	Один користувач може писати кілька відгуків
orders	products	Багато до багатьох (через order_products)	Одне замовлення може містити кілька товарів і навпаки
orders	order_products	Один до багатьох	Одне замовлення включає кілька позицій
products	order_products	Один до багатьох	Один товар може входити до багатьох замовлень
subscribers	users	Один до одного	Один користувач може лише раз зробити підписку

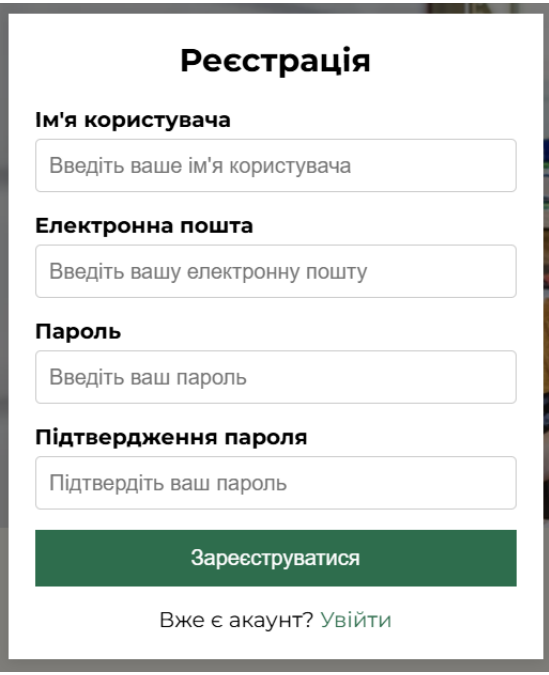
Всі описані таблиці, їх зв'язки та ER-діаграма сформовані до потреб інтернет-магазину, що забезпечує коректне збереження і обробку даних у системі.

РОЗДІЛ 3. СТВОРЕННЯ ТА ФУНКЦІОНАЛЬНА РЕАЛІЗАЦІЯ ІНТЕРНЕТ-МАГАЗИНУ «FRESHFRUIT»

3.1 Реалізація модуля реєстрації користувачів і входу до системи

Однією з найважливіших функцій будь-якого інтернет-магазину є система реєстрації. У випадку FreshFruit, ця функція реалізована максимально просто, але з дотриманням усіх базових вимог до безпеки та зручності користувача, навіть для тих, хто рідко користується інтернет-магазинами. Реєстрація дозволяє користувачеві створити особистий кабінет, через який він зможе оформити замовлення, зайти на сайт, переглядати історію покупок, отримувати персоналізовані знижки та пропозиції.

Форма реєстрації містить звичайні для користувача поля: ім'я, електронну пошту, пароль і повторення пароля для підтвердження. Вона реалізована візуально дуже просто та красиво, відкривається як окремим вікном, щоб людина одразу зорієнтувалася, що потрібно робити. Вікно реєстрації зображено на рисунку 3.1.



The image shows a registration form titled "Реєстрація" (Registration). It contains four input fields: "Ім'я користувача" (Username) with placeholder "Введіть ваше ім'я користувача", "Електронна пошта" (Email) with placeholder "Введіть вашу електронну пошту", "Пароль" (Password) with placeholder "Введіть ваш пароль", and "Підтвердження пароля" (Confirm password) with placeholder "Підтвердіть ваш пароль". Below these fields is a green button labeled "Зареєструватися" (Register). At the bottom, there is a link "Вже є акаунт? Увійти" (Already have an account? Log in).

Рисунок 3.1 – Вікно реєстрації магазину FreshFruit

Усі поля реєстрації є обов'язковими для заповнення. На стороні клієнта реалізовано базову перевірку введених даних – наприклад якщо поле залишили проміжним або вказати електронну адресу в неправильному форматі, то система не дозволить перейти далі, поки користувач не виправить помилку. Ця перевірка виконується за допомогою JavaScript, який відповідає за валідацію без потреби перезавантаження сторінки (рис. 3.2).

```
// Get the value of the entered email
const emailInput = document.getElementById('email-input');
const emailValue = emailInput.value.trim();
const errorMessage = document.getElementById('error-message');
const thankYouMessage = document.getElementById('thank-you-message');

// Regular expression to validate email format
const emailRegex = /^[a-zA-Z0-9._-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;

// Check if the email is valid
if (!emailRegex.test(emailValue)) {
    // If the email is invalid, show the error message
    errorMessage.style.display = 'block';
    thankYouMessage.style.display = 'none';
    return; // Stop further execution
}

// If the email is valid, hide the error message
errorMessage.style.display = 'none';

// Create form data to send
const formData = new FormData();
formData.append('email', emailValue);

// Send the data to the PHP script
fetch('sys/subscribe.php', {
    method: 'POST',
    body: formData
})
```

Рисунок 3.2 – JavaScript-код для перевірки email і відправки даних

Після того як усі поля заповнено коректно, форма передає дані на сервер, де вони обробляються за допомогою PHP-скриптів (email передається у вигляді FormData на сервер через fetch POST-запитом). Тут перевіряється чи не існує вже користувач з такою самою електронною адресою в базі даних. Якщо такий акаунт існує, то виводиться «Помилка: ця електронна пошта вже використовується». Якщо ні – дані зберігаються, але перед цим пароль шифрується з використанням сучасної функції password_hash(), що забезпечує

безпеку даних користувача. Таким чином, навіть якщо база даних буде зламана, зломисник не зможе дізнатися справжні паролі.

Аналогічно реалізована і форма входу (рис. 3.3). Вона має два поля: електронна адреса та пароль. Коли користувач вводить ці дані та натискає кнопку «Увійти», виконується перевірка. Спочатку система шукає в базі користувача з такою електронною поштою. Якщо його не знайдено, то виводиться повідомлення про помилку. Якщо такий користувач існує, перевіряється пароль з тим, що в БД. Для цього використовується функція `password_verify()`, яка дозволяє безпечно перевірити відповідність хешового пароля. Якщо все правильно – користувач авторизується і отримує доступ до свого кабінету. Приклад HTML – коду, де показано поля, та що є в вікні авторизації зображено на рисунку 3.4.

Рисунок 3.3 – Вікно авторизації на сайт FreshFruit

```
<!-- Вікно входу -->
<div id="login-popup" class="popup-overlay" style="display: none;">
  <div class="popup-window">
    <div class="form login-form">
      <h2>Вхід в акаунт</h2>
      <form action="/login" method="POST" id="login-form">
        <label for="login-email">Електронна пошта</label>
        <input type="email" id="login-email" name="email" required placeholder="Введіть вашу електронну пошту">
        <label for="login-password">Пароль</label>
        <input type="password" id="login-password" name="password" required placeholder="Введіть ваш пароль">
        <button type="submit" id="login-button">Увійти</button>
      </form>
      <p>Ще немає акаунта? <a href="#" id="show-register-form">Зареєструватись</a></p>
    </div>
  </div>
</div>
```

Рисунок 3.4 – HTML-код вікна реєстрації

Обидві форми витримані у фірмовому стилі FreshFruit з уніфікованим дизайном на основі HTML і CSS. Кольори та шрифти підібрані для зручності і читабельності. Світлі тони, округлі поля вводу та плавна анімація кнопок при наведенні курсора. Форми адаптовані під мобільні пристрої, тобто елементи автоматично підлаштовуються під екран і мають оптимальний розмір для введення з телефону.

3.2 Формування структури стартової (головної) сторінки

Коли мова йде про головну сторінку, то варто розпочати її опис з хедеру. Хедер – це верхня частина сайту, яка супроводжує користувача на кожній частині сайту. Саме тут зосереджені найважливіші навігаційні елементи. Усе продумано так щоб навіть новий відвідувач інтуїтивно розумів, куди натискати і як швидко знайти потрібне

Починається хедер з логотипа. Логотип асоціюється зі свіжістю, фруктами та натуральністю – саме тим, що закладено в основу концепції проекту. Технічно логотип реалізовано як HTML-елемент `<img>`, у якого вказано шлях до зображення PNG. Назва інтернет-магазину FreshFruit утворена зі слів «fresh» і «fruit», тобто «свіжі фрукти» [34]. Це підкреслює ідею на рівні тексту. При натисканні на логотип або назву користувач повертається на головну сторінку сайту – така поведінка вже звична для більшості інтернет-магазинів, тому її реалізація є обов’язкова.

Поряд із логотипом розташовані основні пункти меню. Першим із них є посилання на головну сторінку. Воно служить своєрідною точкою відліку, звідки користувач може почати знайомство сайтом або якщо потрібно почати все спочатку. Далі йде розділ «Акції», який веде на сторінку зі знижками та спеціальними пропозиціями. Наступним розміщено пункт «Каталог» - один з ключових розділів магазину, адже саме там зібрані всі товари які пропонує FreshFruit.

Окремим пунктом навігаційного меню є «Переваги». Це посилання веде користувача до секції, де перераховано ключові переваги покупки саме у FreshFruit – швидка доставка, висока якість продуктів, зручний сервіс, екологічна упаковка. Це свого роду коротка аргументація, яка формує довіру. Далі в меню йде пункт «Відгуки» [35]. Тут потенційний покупець може ознайомитися з думками інших клієнтів, які вже робили замовлення. Наявність цього розділу важлива для іміджу магазину – реальні історії та позитивні коментарі підвищують довіру та стимулюють до покупки.

З технічної боку для реалізації логотипу, навігаційного меню та кошика використовуються HTML-структури, що зображено на рисунку 3.5. Щоб організувати клікабельність навігаційного меню потрібно в коді додати посилання на певний блок сторінки. Окремо для стилізації застосовуються CSS-правила – вони відповідають за колір – темно-зелений, розмір – 16, поведінку елементів при наведенні – в нашому випадку при наведенні текст стає жирним шрифтом і темно-зеленим кольором (рисунок 3.6).

```

<!-- Хедер -->
<!-- Контейнер для фіксованого хедера -->
<div class="header-container">
  <header class="header">
    <div class="logo">
      <a href="index.html">
        
      </a>
    </div>

    <!-- Горизонтальне меню для ПК -->
    <nav class="nav">
      <a href="#">Головна</a>
      <a href="#weeklyDeals">Акції</a>
      <a href="#catalog">Каталог</a>
      <a href="#advantages">Переваги</a>
      <a href="#reviews">Відгуки</a>
      <a href="#contacts">Контакти</a>
    </nav>
  </div>

```

Рисунок 3.5 – HTML-код реалізації хедеру

```

.nav {
  display: flex;
  gap: 20px;
}
.nav a {
  text-decoration: none;
  color: #333;
  font-size: 16px;
  font-weight: 400;
  margin: 5px;
}
.nav a:hover {
  color: #2e6d4d;
  font-weight: bold;
}

```

Рисунок 3.6 – Фрагмент CSS коду поведінки елементів навігаційного меню при наведенні

Ще однією частиною хедера є блок «Кошик». Він представлений іконкою, яка реалізована зображенням SVG. Біля кошика розміщується лічильник товарів – динамічний елемент, який показує кількість доданих товарів. З технічного боку цей лічильник реалізований за допомогою PHP, який витягує дані з сесії користувача. При кожному додаванні товару у файл PHP, який відповідає за додавання *add_to_cart.php*, зберігається масив товарів у сесії (`$_SESSION['cart']`), а потім кожній сторінці за допомогою `count($_SESSION['cart'])` підраховується кількість елементів кошику. Але щоб лічильник оновлювався без перезавантаження сторінки було використано JavaScript. Після натискання кнопки «Додати в кошик» на стороні JavaScript відправляється запит до PHP-файлу, який обробляє додавання, а у відповіді повертається оновлена кількість товарів. Ця цифра одразу вставляється в DOM-елемент із лічильником через `document.querySelector('.cart-count').textContent = updatedCount`.

Щодо адаптивності – хедер також розмічений із урахуванням різних розмірів екранів. За допомогою CSS медіа-запитів він перебудовується в мобільну версію: меню ховається за кнопкою-бургером (рис.3.7), логотип змінює розмір, а кошик стискається до компактного вигляду. У мобільній версії меню

відкривається JavaScript, який додає або прибирає клас `open` до меню-контейнера, що дозволяє показати або приховати список пунктів за потреби.

```
<!-- Кнопка «гамбургер» для мобільних -->
<button class="menu-toggle" aria-label="Відкрити меню">
  &#9776;
</button>

<div class="cart-icon" onclick="toggleCart()">
  
  <span id="cart-count">0</span>
```

Рисунок 3.7 – HTML-код створення кнопки «гамбургер» для мобільних версій

Вигляд хедеру зі всіма елементами зображено на рисунку 3.8.



Рисунок 3.8 – Зображення хедеру інтернет-магазину

Одразу під хедером сайту FreshFruit розташований банер – яскравий, привабливий візуальний блок, який одразу захоплює увагу відвідувача. Він виконує не лише естетичну, а й інформаційну функцію: знайоми з головною ідеєю магазину – свіжих фруктів, акцентує на актуальних пропозиціях та формує емоційне враження.

Банер створено за допомогою HTML-блоку `<section>`, який займає всю ширину вікна браузера (`width: 100%`) і має фіксовану адаптивну висоту. Для кращої адаптивності використовуються властивості `background-size: cover` і `background-position: center`, щоб картинка залишалася пропорційною і красиво обрізаною на різних пристроях.

Вигляд банеру можна побачити на рисунку 3.9.



Рисунок 3.9 – Банер для інтернет-магазину FreshFruit

Усередині банера розміщений текстовий контент, зокрема назва інтернет-магазину «FreshFruit» та слоган «Соковитий вибір для вашого здоров'я». Ці елементи оформлені за допомогою тегів та з використанням шрифтів Monserrat та Sans Serif.

3.3 Реалізація розділу з акційними пропозиціями, каталогом товарів та фільтрацією

Після вітального банеру на головній сторінці інтернет-магазину FreshFruit розміщується один із ключових і найпривабливіших блоків користувача – розділ «Щотижневі пропозиції». Це не просто декорована частина сайту, а добре продуманий маркетинговий інструмент, завдяки якому клієнт бачить, які товари зараз у тренді, мають дешевшу ціну або персональну акцію [36].

Уся логіка блоку побудована таким чином, щоб привернути увагу відвідувача буквально з перших секунд. Спочатку розміщується короткий заголовок і невеликий опис, який пояснює, що саме тут зібрані актуальні пропозиції тижня – товари, які часто купують або які мають спеціальні знижки.

Це пояснення в користувача викликає бажання переглянути картки товарів нижче.

Основна частина цього блоку – це динамічна вітрина з чотирма товарами, які змінюються щотижня. Кожен товар відображається у вигляді окремої картинки з фотографією, назвою фрукту, одиницею виміру – спочатку закреслено, яка ціна без акції, а потім ціна з акцією, що дає ще більше бажання придбати товар та кнопка «Купити», яка при наведенні стає не бежева, а чорна. Кольорова гама реалізовується за допомогою CSS, який показаний на рисунку 3.10.

```
.sale-item img {
  width: 200px;
  height: 190px;
  margin-bottom: 10px;
  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
}

.sale-item p {
  font-size: 18px;
  margin-bottom: 10px;
  color: #333;
  font-weight: 200;
}

.sale-price {
  color: #6d6d6d;
  font-weight: 300;
  margin-bottom: 10px;
  font-size: 16px;
}

.original-price {
  text-decoration: line-through;
  color: #999;
  margin-right: 10px;
  font-size: 14px;
}
```

Рисунок 3.10 – Фрагмент CSS-коду для щотижневих пропозицій

У коді вище можна побачити:

- .sale-item img – зображення товару: розміром 200 на 190px та легка тінь (візуально приваблює картинку)
- .sale-item p – опис товару: шрифт 18px, сірий колір
- .sale-price – акційна ціна: темно-сірий колір розміром 16px та легке виділення
- .original-price – стара ціна: перекреслена, світло сіра, менша за розміром

При натисканні кнопки щоб купити товар він попадає в кошик. Функція яка це виконує показано на рисунку 3.11.

```
document.querySelectorAll('.add-to-cart').forEach(button => {
  button.addEventListener('click', function () {
    addToCart(this);
  });
});
```

Рисунок 3.11 – JavaScript-код додавання товару в кошик для кнопки «Купити»

А от щодо адаптивності було використано фреймворк Bootstrap щоб це реалізовувалося простіше. На екранах десктопів (від 1024px і більше) відображається чотири товари і це виходить лише один ряд (оскільки щотижневі акції можуть бути лише 4 продуктів) – саме такий варіант вважається оптимальним для широких екранів. Якщо це екран планшета або мобільного телефона (менше 1023) то фрукти відображаються по 2 в ряд.

На рисунку 3.12 показано щотижневі пропозиції та акції на екранах ноутбука.

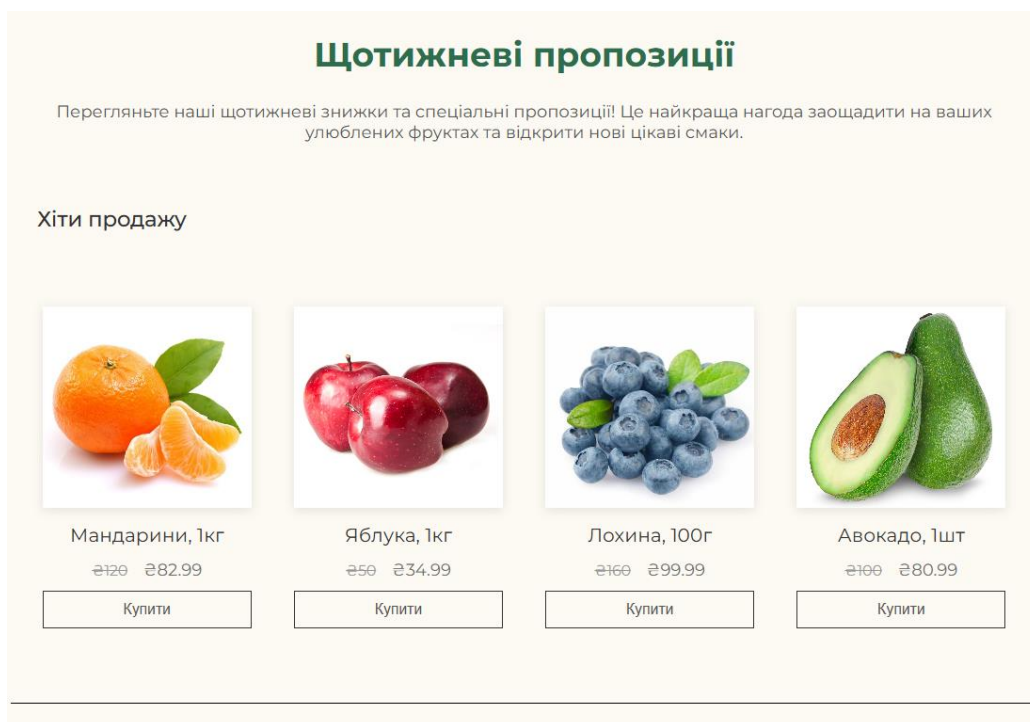


Рисунок 3.12 – Вигляд блоку «Щотижневі пропозиції»

Після секції «Щотижневі пропозиції» одразу йде великий блок з повноцінним каталогом усіх товарів, доступних для замовлення. Основна мета – надати клієнтові можливість швидко знайти потрібні продукти, ознайомитися з їхнім виглядом, ціною та додати в кошик у кілька кліків. Візуально блок ділиться на дві основні частини: панель фільтрації та карти товарів.

Зліва розміщено компактну панель фільтрів, яка дозволяє впорядкувати товари за певними критеріями, а саме: товари за ціною, за алфавітом. А також є можливість скинути всі фільтри натиснувши на кнопку хрестика. Панель виконано у мінімалістичному стилі, щоб не навантажувати інтерфейс і залишати максимум простору для самих товарів. Вигляд фільтрації зображено на рисунку 3.13.

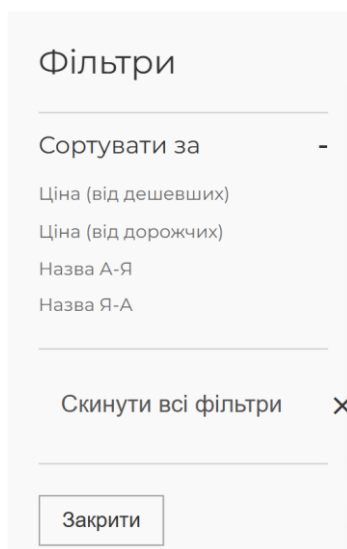


Рисунок 3.13 – Фільтр для каталогу товару

У HTML була створена секція `filters`, яка містить кнопки для сортування товарів. Для кожної дії передбачається відповідний клас, який використовується як селектор у JavaScript-кодi.

У JS-кодi за допомогою `querySelector` зчитує кнопки, список товарів (`.product-list`), та створюється копія масиву товарів для можливості скидання сортування (`originalProductItems`).

Реалізовано функції `sortProducts()` та `sortByName()`, які сортують DOM-елементи за ціною або назвою відповідно. Сортування виконується прямо у браузері без звернення до сервера. Оновлення списку відбувається через `updateProductList()`, яка перезаписує вміст HTML.

Також реалізована функція `resetFilters()`, яка скидає всі фільтри й повертає вихідний порядок товарів.

На рисунку 3.14 показано код з фрагментом фільтрації товару.

```
// Кнопки сортування
const sortPriceAscButton = document.querySelector('.sort-price-asc');
const sortPriceDescButton = document.querySelector('.sort-price-desc');
const sortNameAscButton = document.querySelector('.sort-name-asc');
const sortNameDescButton = document.querySelector('.sort-name-desc');
const resetFiltersButton = document.querySelector('.reset-filters-button');

// Список товарів
const productList = document.querySelector('.product-list');
let productItems = Array.from(productList.querySelectorAll('.product-item'));
// Масив для збереження початкового порядку товарів
const originalProductItems = [...productItems];
// Функція для сортування товарів по ціні
function sortProducts(byPriceDesc = false) {
    const sortedItems = productItems.sort((a, b) => {
        const priceA = parseFloat(a.querySelector('.price').textContent.replace('€', '').trim());
        const priceB = parseFloat(b.querySelector('.price').textContent.replace('€', '').trim());
        return byPriceDesc ? priceB - priceA : priceA - priceB;
    });
    updateProductList(sortedItems);
}
// Функція для сортування товарів по алфавіту
function sortByName(isDescending = false) {
    const sortedItems = productItems.sort((a, b) => {
        const nameA = a.querySelector('h3').textContent.trim().toLowerCase();
        const nameB = b.querySelector('h3').textContent.trim().toLowerCase();
        return isDescending ? nameB.localeCompare(nameA) : nameA.localeCompare(nameB);
    });
    updateProductList(sortedItems);
}
```

Рисунок 3.14 – JavaScript код для фрагменту фільтрації каталогу товарів

У каталозі інтернет-магазину FreshFruit продукти розміщено у спеціальних контейнерах, які візуально виглядають акуратно, впорядковано та приємно для ока, що створює відчуття простору та легкості при перегляді.

Кожен окремий товар виводиться у прямокутному блоці, який має світлий фон, м'які тіні та округлі кути. В середині цього контейнера знаходиться фотографія продукту, його назва, вартість, вага, можливість вибрати кількість та кнопка для додавання товару в кошик.

Для виведення карток товарів із бази даних використовується мова PHP у поєднанні з MySQL-запитами. Вся ця структура зображено на рисунку 3.15.

```
$query = "SELECT id, name, price, sale_price, photo FROM products";
$result = mysqli_query(mysql: $conn, query: $query);

if ($result) {
    $count = 0;

    while ($row = mysqli_fetch_assoc(result: $result)) {
        $count++;

        // Додаємо клас more-items починаючи з 25-го товару
        $classes = 'product-item';
        if ($count > 24) {
            $classes .= ' more-items';
        }
    }
}
```

Рисунок 3.15 – Фрагмент PHP-код для каталогу товарів

При завантаженні сторінки виконується запит до таблиці products в БД, з якої і витягувалися всі дані про кожен товар.

Кожен елемент з отриманого результату виводить окремий HTML-блок класу product-item. Для зручності реалізовано механізм, який приховує частину товарів (починаючи з 25) за допомогою додаткового класу more-item. Це дозволяє організувати функцію без надмірного навантаження інтерфейсу при завантаженні.

Щодо адаптації, то віно реалізована таким самим чином як і блок «Щотижневі пропозиції», тобто на десктопах продукти розташовані 4 в ряд, а на планшетах і телефонах – 2 в ряд.

На рисунку 3.16 зображено каталог товарів інтернет магазину FreshFruit.

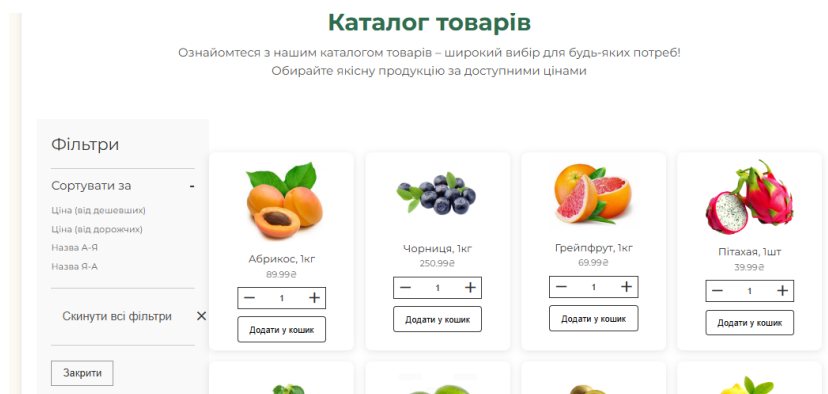


Рисунок 3.16 – Каталог товарів магазину FreshFruit

Зі скріншоту можна побачити, що каталог товарів виглядає яскраво та досить привабливо

3.4 Впровадження блоків інформації про переваги сервісу та контактної інформації сайту

На сайті присутній окремий блок з перевагами, де представлені ключові аспекти сервісу, які підкреслюють турботу про клієнта. Він реалізований за допомогою звичайного HTML-коду без використання складних бібліотек (рис. 3.17). Кожна перевага представлена в окремому контейнері з іконкою, заголовком та коротким описом. У версії для комп'ютерів ці блоки виводяться в один ряд для зручного візуального сприйняття. Під час перегляду сайту на мобільних пристроях структура змінюється завдяки адаптивній верстці – елементи відображаються вертикально один за одним, що забезпечує комфортне ознайомлення з інформацією на невеликих екранах.

```
<section id="advantages" class="advantages">
  <div class="advantage-item">
    
    <h4>Варіанти підбору</h4>
    <p>Обирайте зручний для Вас спосіб отримання товару – самовивіз або доставка. Ми пропонуємо кілька варіантів, щоб Ви могли підібрати найзручніше</p>
  </div>
  <div class="advantage-item">
    
    <h4>Відправка в той же день</h4>
    <p>Замовляйте сьогодні та отримуйте свій товар у найшвидший термін! Швидка доставка – для Вашої зручності та комфорту</p>
  </div>
  <div class="advantage-item">
    
    <h4>Правила охорони праці</h4>
    <p>Ми дбаємо про безпеку наших клієнтів і співробітників. Дотримання правил охорони праці – наш пріоритет для безпечного обслуговування</p>
  </div>
</section>
```

Рисунок 3.17 - HTML-код для блоку з перевагами

На рисунку 3.18 зображено реалізований блок переваг на десктопних версіях.

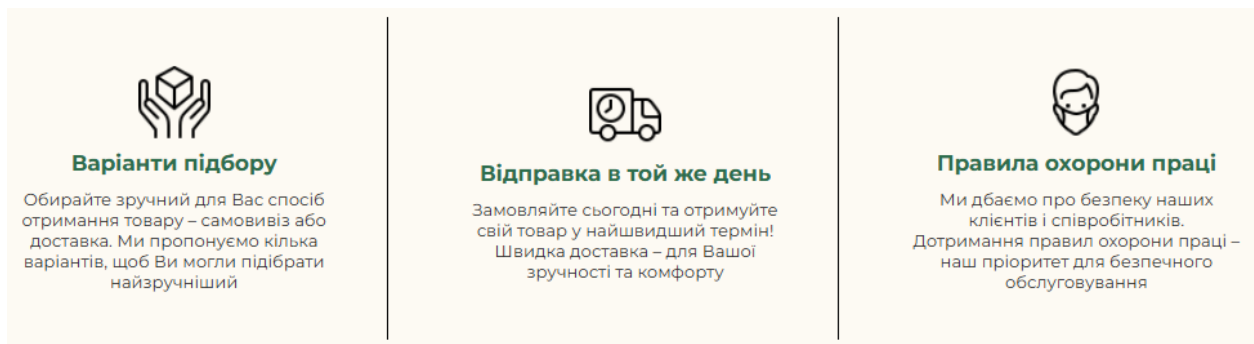


Рисунок 3.18 – Вигляд блоку «Переваги»

Щодо блоку «Контакти», то він оформлений наближено до блоку «Переваги». Ця частинка розроблена у зручному для сприйняття форматі: ліворуч розміщено зображення, а праворуч – самі контакти. Тут користувач може знайти адресу офісу, електронну пошту для зв'язку та номер телефону. Вигляд частини з контактами FreshFruit показано на рисунку 3.19.

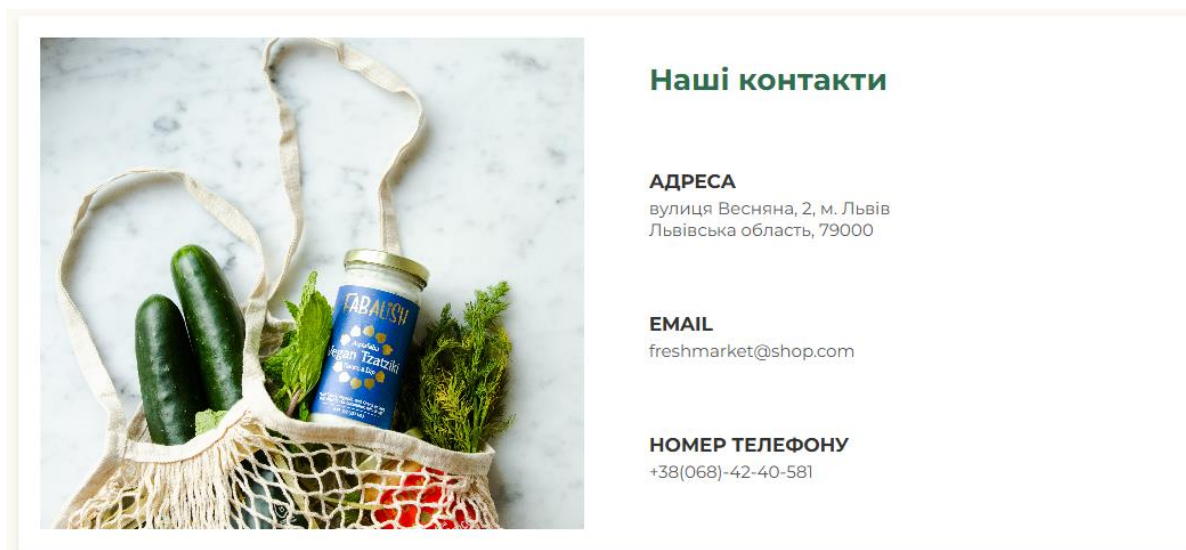


Рисунок 3.19 – Вигляд блоку «Контакти»

Блок створено за допомогою стандартної HTML-структури та стилізовано так, щоб залишалася читабельність на будь-якому пристрої. При адаптації під мобільні екрани елементи розташовуються одним під одним, зберігаючи логіку подачі.

3.5 Додавання функціоналу зібрання відгуків і оформлення підписки

На сторінці FreshFruit реалізовано інтерактивний блок з відгуками клієнтів, який складається з трьох основних частин: заголовка, сітки з відгуками та кнопкою виклику модального вікна для заповнення полів відгуку користувачем.

Відгуки оформлені у вигляді окремих контейнерів, розміщених у горизонтальній сітці (на мобільних версіях у вертикальній сітці). Кожен контейнер містить стилізовану іконку лапок, ім'я користувача (напівжирним шрифтом) та текст відгуку у курсиві. Під цим є кнопка «Залишити відгук». Натиснувши на цю кнопку відкривається вікно з двома обов'язковими полями для вводу: ім'я та прізвище користувача та сам відгук. Важливо зазначити, що відгук може залишити людина, яка не купляла нічого в цьому магазині. Фрагмент коду з реалізації даної функції зображено на рисунку 3.20.

```
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $name = trim(string: $_POST['name'] ?? '');
    $feedback = trim(string: $_POST['feedback'] ?? '');

    if ($name === '' || $feedback === '') {
        echo json_encode(value: [
            'success' => false,
            'message' => 'Всі поля мають бути заповнені.'
        ]);
        exit();
    }

    $nameSafe = mysqli_real_escape_string(mysql: $conn, string: $name);
    $feedbackSafe = mysqli_real_escape_string(mysql: $conn, string: $feedback);

    $query = "INSERT INTO reviews (name, text) VALUES (?, ?)";
    $stmt = mysqli_prepare(mysql: $conn, query: $query);

    if ($stmt) {
        mysqli_stmt_bind_param(statement: $stmt, types: 'ss', var: &$nameSafe, vars: &$feedbackSafe);
        $result = mysqli_stmt_execute(statement: $stmt);
        mysqli_stmt_close(statement: $stmt);

        if ($result) {
            echo json_encode(value: ['success' => true]);
        } else {
            echo json_encode(value: [
                'success' => false,
                'message' => 'Помилка при додаванні відгуку.'
            ]);
        }
    }
}
```

Рисунок 3.20 – Фрагмент PHP-коду реалізації відгуку

Цей код спершу перевіряє чи дані були надіслані методом POST. Потім отримує та обрізає значення імені та тексту відгуку. Якщо хоча б одине поле порожнє – повертає JSON з повідомленням про помилку. Далі виконується

підготовлений SQL-запит для вставки відгуку в базу. У відповідь сервер повертає JSON з результатом – успіх або повідомлення про помилку. На рисунку 3.21 та 3.22 зображено вже реалізований блок, де показано і відгуки інших користувачів, і саме вікно для заповнення.



Рисунок 3.21 – Вигляд частини сайту з відгуками користувачів

Рисунок 3.22 – Поля для введення щоб залишити відгук

Щодо блоку «Підписки» то вона реалізована вікном на фоні фотографії, яка не прокручується та є задньою частиною цілого сайту, що дуже приваблює око. На рисунку 3.23 зображено вікно, де можна підписатися на розсилку повідомлень.

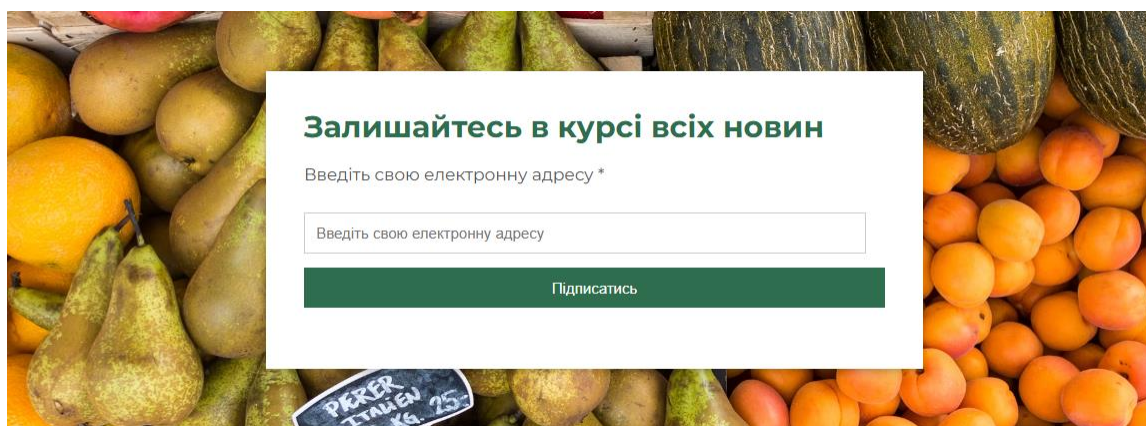


Рисунок 3.23 – Вікно для оформлення підписки

Ця частинка інтернет-магазину реалізована за допомогою PHP, через який зберігається пошта для розсилки, звичайного JavaScript, через який реалізована кнопка «Підписатися» та CSS для реалізації заднього фону.

3.6 Програмування логіки кошика та оформлення замовлення

Блок «Кошик» – це важлива частина інтернет-магазину, яка дозволяє додавати товари для подальшого оформлення замовлення.

Кожен товар на сторінці має кнопку «Додати до кошика». При натисканні на неї спрацьовує JavaScript-функція, яка передає інформацію до сесії PHP (рис. 3.24).

Користувач може додати кілька товарів, а система автоматично оновлює значок кошика у шапці сайту – з відображенням кількості вибраних товарів.

При переході до сторінки кошика(HTML-сторінка) за допомогою PHP та JS витягується перелік усіх обраних товарів. Інтерфейс реалізований зручним і зрозумілим, з можливістю змінити кількість товарів або видалити окрему позицію (рис. 3. 25). Усі зміни в реальному часі оновлюють підсумкову суму.

```
// Масив для зберігання товарів в кошику
let cartItems = [];
let totalAmount = 0;
let cartCount = 0;
// Функція для додавання товару до кошику
function addToCart(button) {
  const item = button.closest('.product-item, .sale-item'); // Підтримуємо обидва класи
  const itemName = item.querySelector('h3, p').innerText.trim(); // Назва товару
  const itemPriceText = item.querySelector('.sale-price, .price').innerText.replace('€', '').trim();
  const itemPrice = parseFloat(itemPriceText);
  const itemQuantity = item.querySelector('input') ? parseInt(item.querySelector('input').value) : 1;
  // Перевірка на валідність кількості
  if (isNaN(itemPrice) || isNaN(itemQuantity) || itemQuantity < 1 || itemQuantity > 10) {
    alert('Кількість товару повинна бути від 1 до 10.');
```

Рисунок 3.24 – Фрагмент JS-коду додавання товару до кошика.

З даного коду можемо побачити, що спочатку створюється масив `cartItems` для зберігання товарів, змінна `totalAmount` для підрахунку суми й `cartCount` – для кількості. Коли користувач натискає кнопку «Додати до кошика», викликається функція `addToCart()`, яка знаходить товарний блок, зчитує назву, ціну та кількість товару. Якщо товар уже є в кошику – оновлюється його кількість і ціна, але не більше 10. Якщо товар новий – додається як новий об'єкт у масив. Вся інформація про кількість і суму відображається прямо на сторінці через DOM-елементи.

Ваша корзина				×
Чорниця, 1кг	€250.99	1		×
Грейпфрут, 1кг	€69.99	1		×
Загальна сума: €320.98				
Оформити замовлення				

Рисунок 3.25 – Вікно Корзини

Після перегляду кошика користувач переходить до сторінки оформлення замовлення, де потрібно ввести свої дані: ім'я, прізвище, адресу доставки, номер телефону та електронну пошту. Усі поля є обов'язковими для заповнення. Додатково використовується валідація через JS (перевірка формату номера телефону та email).

Після натискання кнопки «Оформити замовлення» дані надсилаються на сервер через POST-запит. На сервері перевіряється коректність введення інформації та формуються записи в БД: створюється нове замовлення і до нього додаються всі товари.

На рисунку 3.26 показано PHP-код для оформлення замовлення.

```
$city = trim(string: $_POST['city'] ?? '');
$street = trim(string: $_POST['street'] ?? '');
$house = trim(string: $_POST['house'] ?? '');
$details = trim(string: $_POST['details'] ?? '');
$payment = trim(string: $_POST['payment'] ?? 'отівка');
$summary = floatval(value: $_POST['summary'] ?? 0);
$card = trim(string: $_POST['card'] ?? '');
$paypal = trim(string: $_POST['paypal'] ?? '');
// Перевірка на заповненість полів
if (empty($city) || empty($street) || empty($house) || empty($details) || empty($summary) || empty($payment)) {
    echo json_encode(value: ['success' => false, 'message' => 'Всі поля обов'язкові!']);
    exit;
}
// Вихист від SQL-ін'єкції
$city = mysqli_real_escape_string(mysql: $conn, string: $city);
$street = mysqli_real_escape_string(mysql: $conn, string: $street);
$details = mysqli_real_escape_string(mysql: $conn, string: $details);
$payment = mysqli_real_escape_string(mysql: $conn, string: $payment);

$card = mysqli_real_escape_string(mysql: $conn, string: $card);
$paypal = mysqli_real_escape_string(mysql: $conn, string: $paypal);
// Запит на додавання замовлення в БД
$insert_query = "INSERT INTO orders (details, summary, city, street, house, payment, card, paypal)
VALUES ('$details', '$summary', '$city', '$street', '$house', '$payment', '$card', '$paypal')";

if (mysqli_query(mysql: $conn, query: $insert_query)) {
    echo json_encode(value: ['success' => true, 'message' => 'Замовлення успішно оформлене!']);
} else {
    echo json_encode(value: ['success' => false, 'message' => 'Помилка при оформленні замовлення. Спробуйте ще раз!']);
}
```

Рисунок 3.26 – PHP-код для оформлення замовлення

На рисунку 3.27 вигляд вже реалізованого вікна полів вводу при оформленні замовлення.

Рисунок 3.27 – Вікно оформлення товару

Після успішного замовлення користувачеві виводиться повідомлення «Замовлення успішно оформлено», якщо щось пішло не так, то виводиться повідомлення про помилку.

3.7 Проведення тестових перевірок працездатності та зручності ресурсу

Після завершення етапу розробки інтернет-магазину FreshFruit було проведено повноцінне тестування всіх його функціональних компонентів. Метою тестування є перевірка працездатності сайту, зручності а також привабливості зображення та обробки даних. Це дозволяє виявити можливі помилки ще до запуску сайту для широкого кола користувачів [37].

У процесі тестування була перевірена робота таких функціональних елементів як реєстрація, авторизація, додавання товарів до кошика, перегляд каталогу, роботу випадаючих вікон, фільтрація, оформлення підписки, перевірка правильності переваг та контактів, а також оформлення замовлення.

Першочергово перевірялась коректність відображення всіх сторінок магазину на різних пристроях. Було протестовано роботу сайту на десктопах,

планшетах та мобільних телефонах із різними розширеннями екранів. Усі блоки коректно масштабуються, меню адаптується, а елементи зберігають читабельність та зручність у користуванні. Приклад адаптації під версію телефону (412 пікселів) та планшета (768 пікселів) зображено на рисунку 3.28 та рисунку 3.29.

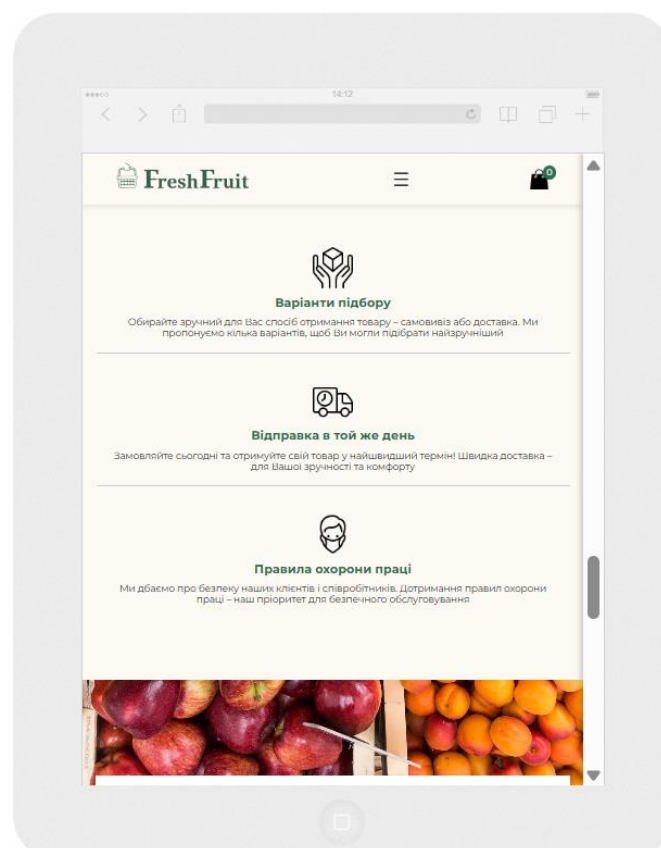
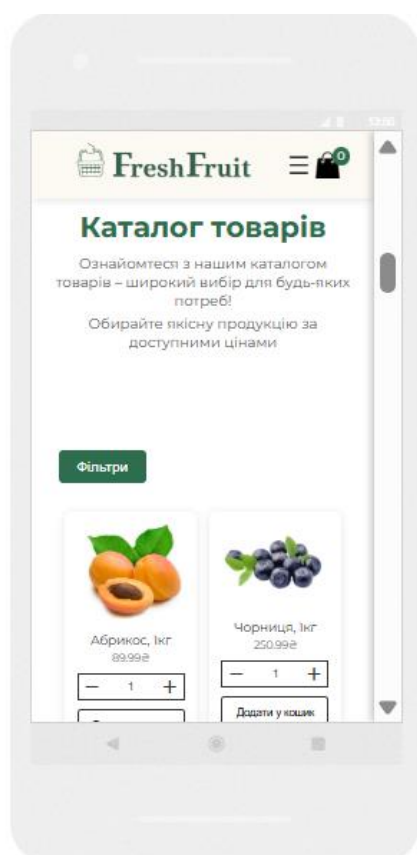


Рисунок 3.28 – Адаптація під телефон Рисунок 3.29 – Адаптація під планшет

З скрінів можна побачити, що товари зображено 2 в ряд на мобільній версії, які і планувалось, якщо екран менший ніж ноутбука то в хедері навігаційне меню під кнопкою «гамбургер», а також переваги сайту розташовані одні під одним.

Окрему увагу приділялося перевірці логіки взаємодії користувача з сайтом. Наприклад, перевірялося чи дійсно працюють поля реєстрації, авторизації та підписки і чи буде повідомлення про успішність чи помилку (рис. 3.30 та 3.31).

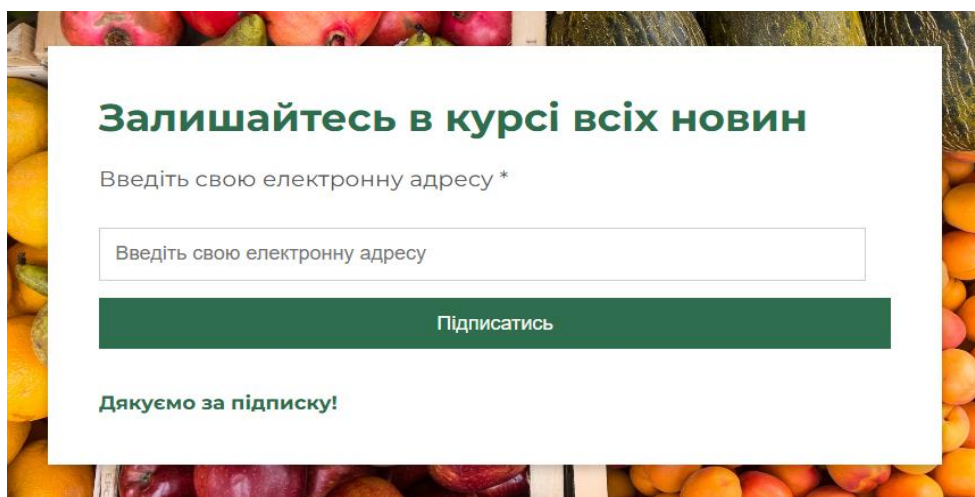


Рисунок 3.30 – Повідомлення про успішність оформлення підписки

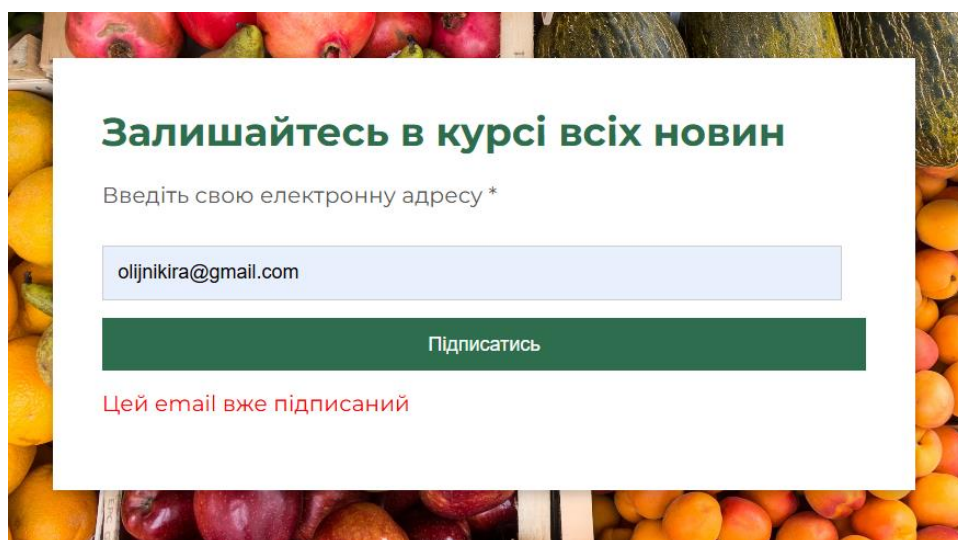


Рисунок 3.31 – Повідомлення про помилку оформлення підписки

Ретельно протестовано і робота кошика. Перевірялося чи оновлюється кількість товарів при додаванні, чи правильно розраховується підсумкова сума, чи можна змінити кількість одиниць товару, а також чи зберігається вміст кошика при переході на інші сторінки.

Ще один важливий аспект – це робота з базою даних. Тестувалося чи створюються нові записи реєстрації користувачів та інші блоки. На рисунку 3.32 та 3.33 наведено приклад.

Реєстрація

Ім'я користувача

Олеся

Електронна пошта

pudlykolesya@gmail.com

Пароль

.....

Підтвердження пароля

.....|

Зареєструватися

Вже є акаунт? [Увійти](#)

Рисунок 3.32 – Реєстрація нового користувача

+ 1 Параметри

← T →				id	name	email	password
☐ Редагувати Копіювати Видалити	3	Iryna	olijnikira12@gmail.com	\$2y\$10\$CILu797I			
☐ Редагувати Копіювати Видалити	5	Вася	protsivv73@gmail.com	\$2y\$10\$0HvyG9c			
☐ Редагувати Копіювати Видалити	7	Олеся	pudlykolesya@gmail.com	\$2y\$10\$VIDNmV			

Рисунок 3.33 - Перевірка чи створився запит в базі даних

Також під час тестування спеціально моделювали ситуації, коли користувач вводить неправильні або неповні дані. Це дозволило перевірити, як сайт реагує на такі дії: чи з'являється повідомлення про помилки, чи залишається він стабільним, без зависань чи раптових перезавантажень. Приклад реалізації даного тестування зображено на рисунку 3.34 та 3.35.

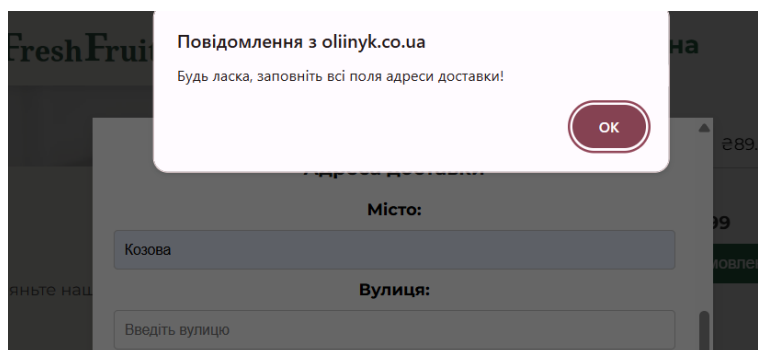


Рисунок 3.34 – Спроба залишити одне порожнє поле при оформленні замовлення

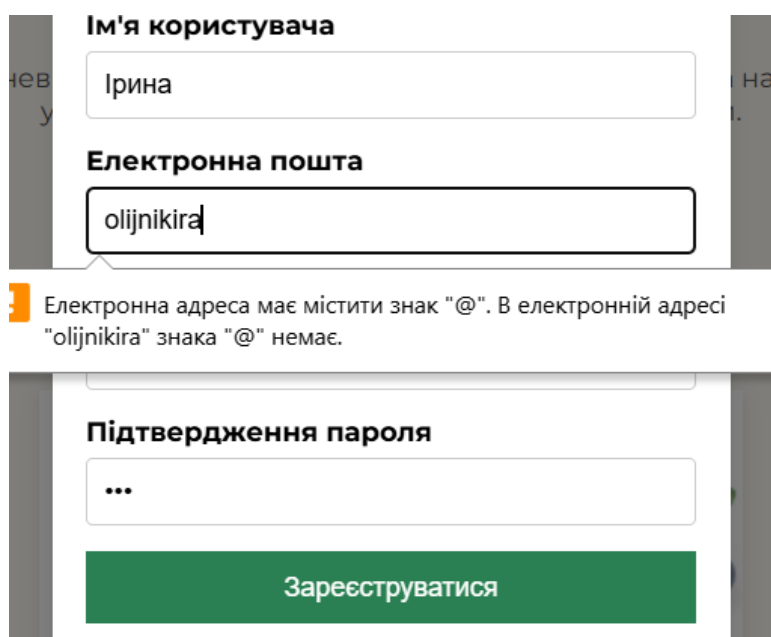


Рисунок 3.35 – Спроба ввести не коректну електронну пошту

Можна сказати, що всі перевірки інтернет-магазину FreshFruit пройшли успішно. Основні опції працюють стабільно, без збоїв, а сам сайт зручний у користуванні. Тепер він повністю готовий до запуску та комфортної роботи з реальними користувачами.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Вимоги ергономіки до організації робочого місця оператора ПК

Організація робочого місця користувача персонального комп'ютера має відповідати сучасним ергономічним нормам, які чітко прописані державними стандартами. Це необхідно не лише для створення комфортного середовища, а й для забезпечення здоров'я та ефективності праці. Основні вимоги закріплені в таких нормативних документах, як ДСТУ 7950:2015, ДСТУ 7951:2015 та ДСТУ 8604:2015, які визначають загальні правила облаштування робочих місць у положенні сидячи, а також параметри крісел операторів.

Робоче місце для сидячої роботи повинно бути зручним, щоб зменшити втому м'язів і не перенавантажувати спину. Стіл має бути на висоті близько 72–75 см, а крісло повинно мати можливість регулювати висоту від 40 до 55 см, щоб кожен міг налаштувати його під свій ріст. Під столом має бути достатньо місця для ніг – не менше 45 см у глибину і 60 см у ширину.

Кімната, де буде стояти комп'ютер, повинна мати площу не менше 6 м², а об'єм — не менше 24 м³ [38].

Крісло користувача комп'ютера має бути стабільним, мобільним і мати можливість налаштування індивідуальних параметрів. Важливо, щоб спинка підтримувала природний вигин хребта, зменшуючи навантаження на спину, вона повинна підтримувати поперек. За стандартом, спинка має бути нахилена назад під кутом приблизно 95–110° відносно сидіння. Також рекомендується наявність підставки для ніг, яка дозволяє зменшити напругу в нижніх кінцівках під час тривалої роботи.

Ще одним важливим аспектом є правильне розміщення монітора. Відстань від очей до екрану має бути не менше 50 і не більше 70 см. Кут зору має складати приблизно 15–20 градусів. Верхній край дисплея не повинен бути вище рівня очей. Оптимально, щоб екран був розташований перпендикулярно до лінії зору

користувача, тоді не буде зайвих відблисків і очі менше втомлюються. На рисунку 4.1 показано як повинен сидіти оператор ПК перед монітором.

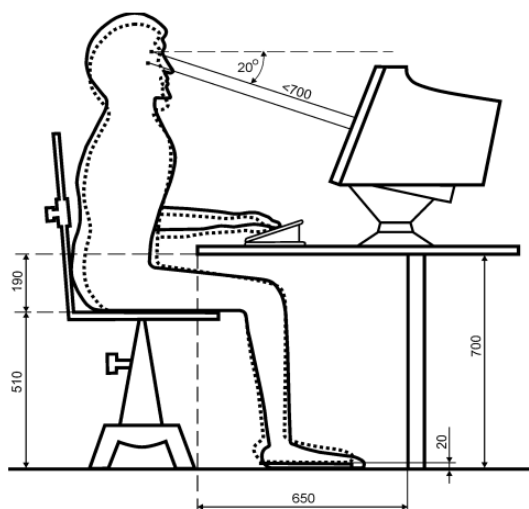


Рисунок 4.1 – Відстані до монітора для офісної роботи

Важливою складовою комфортної та безпечної роботи є меблі, які використовуються на робочому місці. Робочий стіл повинен мати достатню площу для розміщення монітора, клавіатури, миші, документів та інших необхідних матеріалів. Поверхня повинна бути матовою, без блиску, бажано нейтрального кольору, щоб уникнути зайвого навантаження на зір. Усі кути меблів мають бути заокруглені, щоб запобігти травмам. Важливо, щоб у користувача був вільний простір для розміщення ніг та зміни положення тіла, а також можливість тримати руки на столі під прямим кутом – це знижує м'язове напруження під час тривалої роботи.

Не менш важливим фактором є освітлення. Робоче місце має бути добре освітлене, бажано з природним джерелом світла, яке не створює відблисків на екрані монітора. При розміщенні робочого столу необхідно враховувати орієнтацію вікон: світло має падати зліва або спереду, але не ззаду. Якщо використовуються штучні джерела світла, їх яскравість має бути достатньою (не менше 300–500 лк для зорової роботи середньої складності) і рівномірно розподіленою по площі робочого місця. Рекомендується використання

локального освітлення – наприклад, настільної лампи з регулюванням напрямку світла – але без сліпучого ефекту.

Ще одним важливим елементом безпечного робочого середовища є мікроклімат. Відповідно до гігієнічних норм, які прописані в ДСН, оптимальна температура в приміщенні для роботи з комп'ютером має становити 20–24 °С, вологість повітря – на рівні 40–60%, швидкість руху повітря – не більше 0,1–0,2 м/с. Забезпечення цих показників дозволяє уникнути перегрівання або переохолодження організму, зменшує ризик появи втоми, подразнення очей, сухості шкіри та дихальних шляхів. Важливою є також система вентиляції, яка повинна забезпечувати доступ свіжого повітря без створення протягів. В опалювальний сезон слід уникати прямого впливу гарячого повітря з батарей на користувача, а також забезпечити зволоження повітря в разі надмірної сухості.

4.2 Психологічні чинники небезпеки при розробці сайту

Розробка веб-сайтів є складним та інтенсивним видом діяльності, що пов'язаний із високим рівнем психоемоційного навантаження. Веб-розробники працюють у постійному ритмі дедлайнів, виконуючи складні завдання, які вимагають концентрації, уваги до деталей та швидкого реагування на зміни. Ці фактори створюють низку психологічних чинників, що можуть негативно впливати на здоров'я та працездатність працівника.

Одним із ключових факторів є стрес, який виникає через жорсткі строки виконання проектів, необхідність швидко виправляти помилки, а також багатозадачність, коли розробник одночасно виконує кілька завдань. Постійне психоемоційне напруження може призводити до підвищеної тривожності, роздратованості, зниження мотивації та, у перспективі, до професійного вигорання. Професійне вигорання характеризується емоційним виснаженням, цинізмом щодо роботи та зниженням продуктивності, що загрожує як здоров'ю працівника, так і якості виконуваних проектів.

Поведінкові ознаки вигорання проявляються через поступове відчуження від роботи та колективу. Людина починає приходити пізніше, а йти раніше з офісу, неохоче відповідає на робочі повідомлення і часто здригається від дзвінків у Teams чи Zoom, що свідчить про підвищену тривожність і небажання взаємодіяти. Відбувається ізоляція від колег – розробник уникає спільних заходів, тимбілдингів чи неформальних посиденьок, оскільки вони викликають додаткове виснаження замість відпочинку.

Крім того, відзначається підвищене споживання алкоголю та цигарок, як спосіб зняти внутрішнє напруження. На роботі помітні зволікання з виконанням завдань, а також пропуски дедлайнів, що негативно впливає на якість і строки проектів. Вигорання може проявлятися у вигляді підвищеної агресивності або конфліктності, що ускладнює спілкування з колегами і керівництвом. Ці поведінкові зміни є сигналом серйозних психологічних проблем, які потребують уваги, корекції і підтримки як зі сторони керівництва, так і колективу.

Схема поведінкових рис вигорання показано на рисунку 4.2



Рисунок 4.2 – Поведінкові риси вигорання

Важливо також враховувати ефект перевтоми, яка виникає через тривалі робочі зміни без достатніх перерв, роботу у нічний час чи надурочні години. Перевтома знижує увагу та концентрацію, збільшує ризик помилок і травматизму, а також негативно впливає на загальний стан здоров'я – може

викликати порушення сну, зниження імунітету та розвиток хронічних захворювань.

Для зменшення впливу цих психологічних чинників рекомендовано дотримуватися таких правил: організовувати робочий день із врахуванням регулярних перерв (згідно з ДСанПіН 3.3.2.007-98, 10-15 хвилинна перерва після кожної години роботи за ПК), уникати надмірних перевантажень, забезпечувати розумний розподіл завдань між членами команди, а також створювати комфортний психологічний клімат у колективі.

Важливою складовою профілактики стресу є впровадження гнучких графіків роботи, можливість працювати дистанційно, організація навчання з тайм-менеджменту та технік релаксації. Регулярне фізичне навантаження, здоровий спосіб життя і підтримка соціальних контактів також сприяють зниженню негативного впливу стресу.

4.3 Загальні вимоги безпеки з охорони праці для веб-розробника

Робота веб-розробника сайту є типовим прикладом діяльності, що пов'язана з тривалим перебуванням за персональним комп'ютером, високим навантаженням на зорову систему та значним психоемоційним напруженням. У зв'язку з цим особливо важливо дотримуватись вимог охорони праці, що прописані ДСТУ, ДСН та рекомендаціями Міністерства охорони здоров'я України.

Перш за все, розробник повинен пройти вступний та періодичний інструктаж з охорони праці. Це стосується навіть тих, хто працює віддалено – роботодавець все одно несе відповідальність за умови праці. Працівник має знати, як безпечно користуватися технікою, уникати електротравм, перевантажень зору і опорно-рухового апарату [39]

Уся комп'ютерна техніка повинна бути справною. Це стосується як самого комп'ютера, так і моніторів, клавіатур, мишок, блоків живлення, розеток і подовжувачів. З точки зору безпеки, розробник сайту має користуватися

технічно справним обладнанням. Всі електричні пристрої повинні бути заземлені, із захисною ізоляцією. Забороняється використання пошкоджених кабелів і подовжувачів, оскільки це створює небезпеку ураження електричним струмом або виникнення пожежі.

Робоче місце повинно бути чистим, охайним і обладнаним відповідними засобами пожежогасіння. Перед початком роботи веб-розробник має перевірити своє місце відповідно до вимоги ергономіки оператора ПК.

Рекомендується кожні 60 хвилин роботи робити перерву 10–15 хвилин. За цей час краще встати, пройтися, зробити розминку для плечей, шиї, кистей і вправи для очей. Це допомагає уникнути перенапруги, головного болю, болю в спині, а також підвищує продуктивність.

У приміщенні, де працює розробник, має бути свіже повітря, комфортна температура. Слід уникати пересушеного повітря через постійне використання кондиціонерів або обігрівачів. Важливо, щоб у кімнаті не було пилу, вологи або джерел надмірного шуму.

Під час роботи за ноутбуком потрібно використовувати підставку для підняття екрану на правильний рівень і окрему клавіатуру з мишкою. Це дозволяє дотримуватися правильної посадки і зменшує навантаження на руки та шию.

ВИСНОВКИ

У межах виконання кваліфікаційної роботи було здійснено повноцінний цикл створення функціонального інтернет-магазину «FreshFruit», що спеціалізується на реалізації свіжих фруктів. Робота охоплює етапи від попереднього аналізу ринку та постановки задач – до безпосереднього проектування, реалізації та тестування готового вебсайту.

У першому розділі роботи виконано теоретичне обґрунтування теми. Досліджено сучасні тенденції розвитку електронної комерції в Україні та світі, виявлено зростаючу роль інтернет-магазинів у структурі споживчого ринку. Розглянуто особливості створення сайтів для торгівлі харчовими товарами, враховуючи специфіку продукції, потреби користувача та вимоги до візуального оформлення і зручності інтерфейсу. Особливу увагу приділено вже існуючим сайтам фруктів та визначенню які в них є плюси та мінуси щоб пізніше мати змогу розробити сайт кращим за аналоги. Також було розгорнуто мету створення сайту – створити сайт привабливим візуально, добре функціональним, адаптивним під різні пристрої та зручним у користуванні.

У другому розділі кваліфікаційної роботи розкрито процес проектування сайту було обрано архітектуру для інтернет-магазину та нею стала клієнт-серверна архітектура. Було розроблено UML діаграму для кращого розуміння ролей клієнта в інтернет-магазині та адміністратора. Обґрунтовано вибір мови програмування PHP для створення серверної логіки, мови розмітки HTML та каскадних таблиць стилів CSS для структурування й оформлення сторінок, а також JavaScript для реалізації інтерактивних елементів (наприклад, підказки, валідація форм та інше).

Також підібрано для усіх елементів аспекти відповідно до принципів UX/UI-дизайну: логічна структура, інтуїтивна навігація, адаптивність до різних екранів, чітка візуалізація інформації. Обрано 2 основні кольори, які би гарно впливали на хороше емоційне ставлення користувача до сайту та впізнаваність бренду. А саме ніжний кремовий який асоціюється з відкритістю та легкістю і

темно-зеленого, більш насиченого для заголовків, який асоціюється з екологічністю та здоров'ям. Щодо шрифтів, то було обрано прості та читабельні.

Особливу увагу було приділено побудові бази даних. Було створено логічну модель даних, в якій передбачено таблиці для зберігання інформації про користувачів, товари, замовлення, категорії продукції, відгуки, підписку на розсилку, а також зворотний зв'язок. Було розроблено ефективну структуру зв'язків між таблицями, яка дозволяє забезпечити швидкий доступ до даних, масштабованість та безпечне збереження інформації. Створено ER діаграму, яка візуально показувала весь функціонал бази даних та на яку опиралося надалі при реалізації самого сайту. Використання MySQL забезпечило надійність у роботі з даними та сумісність з обраною серверною частиною сайту.

Третій розділ присвячено практичній реалізації вебсайту. Послідовно описано створення кожного з модулів: реєстрації користувачів, авторизація, хедеру, головної сторінки з блоком щотижневі пропозиції, каталогу з можливістю фільтрації, підписки на новини, можливість побачити відгуки інших користувачів та написати свій, блок з перевагами даного інтернет-магазину, а також корзини й оформлення замовлення. Усі елементи взаємодії реалізовано з урахуванням сучасних принципів доступності, читабельності та зручності. Було проведено повноцінне тестування працездатності сайту, яке включало перевірку працездатності основних модулів, правильність збереження та виведення інформації з бази даних, валідацію форм, стійкість до помилок користувача, адаптивність на різних пристроях.

У розділі «Безпека життєдіяльності, основи охорони праці» розглянуто основні аспекти створення безпечних умов під час розробки вебсайтів. Висвітлено психологічні чинники небезпеки, зокрема вплив стресу, дедлайнів і перевтоми. Також проаналізовано ергономічні вимоги до організації робочого місця оператора ПК відповідно до державних стандартів. Окрім цього, подано загальні вимоги з охорони праці для користувачів персональних комп'ютерів.

ПЕРЕЛІК ДЖЕРЕЛ

1. Гузенко С., Міщенко С. Тренди українського e-commerce на 2025 рік. Elitweb. [Електронний ресурс]. – 2025. – Режим доступу до ресурсу: <https://elitweb.ua/ua/blog/trendy-ukrainskogo-e-commerce-na-2025-god>. (дата звернення: 08.05.2025).

2. Струтинська, І., Козбур, Г., Дмитроца, Л., Сороківська, О., Мельник, Л., та Грицеляк, Р. (вересень 2021 р.). Щодо концепції цифровізації малих та середніх підприємств в Україні: проблеми та рішення. У 2021 році на 11-й Міжнародній конференції з передових комп'ютерних інформаційних технологій (АСІТ) (с. 276-279). IEEE.

3. Струтинська, І. В., Дмитроца, Л. П., Сороківська, О. А., & Козбур, Г. В. (2024). Особливості цифрового розвитку малого і середнього бізнесу України, країн Європи та G7. In Трансформація бізнесу для сталого майбутнього: дослідження, цифровізація та інновації: монографія (pp. 411-427). : ФОП Паляниця В. А.

4. Створення інтернет-магазину в Україні. Аналізуємо ринок. Lemarbet. [Електронний ресурс] – Режим доступу до ресурсу: <https://lemarbet.com/ua/analitika/sozdanie-internet-magazina-v-ukraine-analiziruemyy-rynok/>. (дата звернення: 10.05.2025).

5. Азізонова М. Інтернет-магазини vs офлайн: де купувати краще. Finance.ua. [Електронний ресурс] – Режим доступу до ресурсу: <https://finance.ua/ua/goodtoknow/internet-mahazyny-vs-oflain-de-kupuvaty-krashche>. (дата звернення: 11.05.2025).

6. Переваги та ризики купівлі в інтернеті. Vest.in. [Електронний ресурс] – Режим доступу до ресурсу: <https://vest.in.ua/ua/interesnaya-informaciya/preimushchestva-i-riski-pokupki-v-internete>. (дата звернення: 10.06.2025).

7. SEO-просування інтернет-магазину. Sendpulse. [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: <https://sendpulse.ua/blog/seo-promotion-of-online-store>. (дата звернення: 11.05.2025).

8. @Альона. Як визначити цільову аудиторію. Idea Bigital. 10.05.2025. [Електронний ресурс] – Режим доступу до ресурсу: <https://ideadigital.agency/blog/yak-viznachiti-tsilovu-auditoriyu/>. (дата звернення: 13.05.2025).

9. Чому створення інтернет-магазину це один з кращих методів розширення бізнесу. LB.ua. [Електронний ресурс]. – 2023. – Режим доступу до ресурсу: https://lb.ua/tech/2023/10/23/579979_chomu_stvorenniya_internetmagazinu_tse.html. (дата звернення: 13.05.2025).

10. Як правильно оформляти замовлення в інтернет-магазині?. Store Torgsoft. [Електронний ресурс] – Режим доступу до ресурсу: https://store.torgsoft.ua/blog/onlajnzamovlennia/?srsltid=AfmBOoohoV1Hlk2ADILfOi2qNcBOoxG3BDxZ_ak0cqy7eDasyhZdJRv1. (дата звернення: 20.05.2025).

11. Клієнт-серверна архітектура. СЕРВЕРНІ WEB-ТЕХНОЛОГІЇ : навчальний посібник / О. С. Бунке. Київ : КПІ ім. Ігоря Сікорського, 2023. С. 1–17.

12. Архітектура клієнт-сервер. Intelligent system design. [Електронний ресурс] – Режим доступу до ресурсу: <https://www.intelsd.com/?tc=175&sc=197&lvl=2>. (дата звернення: 20.05.2025).

13. Марголін О. Що таке UML-діаграми?. Evergreen. [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://evergreens.com.ua/ua/articles/uml-diagrams.html>. (дата звернення: 21.05.2025).

14. Олійник І. Технології Веб-розробки клієнт-серверного рішення для інтернет-комерції. VIII Міжнародна студентська науково - технічна конференція "Природничі та гуманітарні науки. Актуальні питання", Тернопіль, Тернопільський національний технічний університет ім. І.Пулюя (м. Тернопіль, Україна, 24-25 квіт. 2025 р.), 2025. С.184-185.

15. Wolf J. HTML and CSS: The Comprehensive Guide. 2023. 814 с. ISBN 978-1493224227.

16. HTML Довідник тегів. W3schoolsua. [Електронний ресурс] – Режим доступу до ресурсу: <https://w3schoolsua.github.io/tags/index.html#gsc.tab=0>. (дата звернення: 21.05.2025).

17. CSS властивостям. CSS.IN.UA. [Електронний ресурс] – Режим доступу до ресурсу: <https://css.in.ua/css/properties>. (дата звернення: 21.05.2025).

18. Вступ до JavaScript. Javascript.info. [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.javascript.info/intro>. (дата звернення: 01.06.2025).

19. Федорова Н. О. Об'єктна модель документа. Всеосвіта. [Електронний ресурс] – Режим доступу до ресурсу: <https://vseosvita.ua/lesson/obiektna-model-dokumenta-554110.html>. (дата звернення: 01.06.2025).

20. PHP Підключення до MySQL. W3schoolsua. [Електронний ресурс] – Режим доступу до ресурсу: https://w3schoolsua.github.io/php/php_mysql_connect.html. (дата звернення: 02.06.2025).

21. Шпота, Т. О., & Палка, О. В. (2024). Огляд технологій аналізу даних. Збірник тез доповідей XIII Міжнародної науково-практичної конференції молодих учених та студентів „Актуальні задачі сучасних технологій“, 507-507.

22. PHP + MySQL. ZNU. [Електронний ресурс] – Режим доступу до ресурсу: <https://sites.znu.edu.ua/webprog/lect/1222.ukr.html>. (дата звернення: 03.06.2025).

23. Суслов Р. Система управління базами даних MySQL. Lemon School. [Електронний ресурс] – Режим доступу до ресурсу: <https://lemon.school/blog/systema-upravlinnya-bazamy-danyh-mysql>. (дата звернення: 03.06.2025).

24. Що таке хостинг. HOSTIQ. [Електронний ресурс] – Режим доступу до ресурсу: <https://hostiq.ua/ukr/info/what-is-hosting/> (дата звернення: 03.06.2025).

25. У чому особливості створення односторінкових сайтів?. Firtka. [Електронний ресурс]. – 2021. – Режим доступу до ресурсу:

<https://firtka.if.ua/blog/view/u-chomu-osoblivosti-stvorennia-odnostorinkovikh-saitiv>. (дата звернення: 04.06.2025).

26. Як правильний вибір кольорової палітри може підвищити ефективність вашого сайту. CODE4U. [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: <https://code4u.com.ua/news/how-choosing-the-right-color-palette-can-improve-the-effectiveness-of-your-site/>. (дата звернення: 04.06.2025).

27. Скиба, О. П., Назаревич, О. Б., & Шимчук, Г. В. (2019). Комп'ютерна графіка: методичний посібник та завдання до виконання графічних та самостійних робіт для студентів усіх форм навчання спеціальностей 122" Комп'ютерні науки" та 123" Комп'ютерна інженерія" з курсу" Комп'ютерна графіка".

28. Як працює психологія кольору в маркетингу: практичний посібник для бізнесу. SendPulse. [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: <https://sendpulse.ua/blog/color-psychology>. (дата звернення: 04.06.2025).

29. Mykhailov V. Кольори в UX дизайні: підбір і взаємодія. DOU. [Електронний ресурс]. – 2023. – Режим доступу до ресурсу: <https://dou.ua/forums/topic/43046/>. (дата звернення: 04.06.2025).

30. Базовий погляд на типографіку у веб-дизайні. DIZZ. [Електронний ресурс] – Режим доступу до ресурсу: <https://dizz.in.ua/uk/bazovij-poglyad-na-tipografiku-u-veb-dizajni/>. (дата звернення: 06.06.2025).

31. Визначення в базі даних зв'язків між таблицями. StudFile. [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://studfile.net/preview/5720747/page:51/>. (дата звернення: 06.06.2025).

32. Хешування. VPN unlimited. [Електронний ресурс] – Режим доступу до ресурсу: https://www.vpnunlimited.com/ua/help/cybersecurity/hashing?srsId=AfmBOop9OrwhB_xnSMD1p7LdhuvpysT1oOdEsc3u7N0OuWFpFa2Yb0t3. (дата звернення: 06.06.2025).

33. Моралес Д. Діаграма зв'язків сутностей: приклади, символи та вказівки для створення. MindOnMap. [Електронний ресурс]. – 2022. – Режим доступу до

ресурсу: <https://www.mindonmap.com/uk/blog/relationship-diagram/#part1>. (дата звернення: 07.06.2025).

34. Гузенко С., Міщенко С. Як розробити стратегію бренду. elitweb. [Електронний ресурс]. – 2025. – Режим доступу до ресурсу: <https://elitweb.ua/ua/blog/kak-razrabotat-strategiju-brenda> (дата звернення: 07.06.2025).

35. Які сторінки мають бути на сайті інтернет-магазину. FISHDIGITAL. [Електронний ресурс]. – 2023. – Режим доступу до ресурсу: <https://www.fishdigital.agency/blog-yaki-storinki-mayut-buti-na-sayti-internet-magazinu> (дата звернення: 08.06.2025).

36. Акції для збільшення продажів: Які акції можна провести в магазині. HANDMADE HUB. [Електронний ресурс] – Режим доступу до ресурсу: <https://handmade-hub.ua/aktsii-ohliad-osnovnykh-vydiv/> (дата звернення: 08.06.2025).

37. Тестування програмного забезпечення: типи, види та застосування. FoxmindEd. [Електронний ресурс]. – 2023. – Режим доступу до ресурсу: <https://foxminded.ua/testuvannia-prohramnoho-zabezpechennia/>. (дата звернення: 09.06.2025).

38. Організація робочого місця оператора комп'ютерного набору. StudFile. [Електронний ресурс]. – 2025. – Режим доступу до ресурсу: <https://studfile.net/preview/11787126/page:3/> (дата звернення: 10.06.2025).

39. Гандзюк М.П., Желібо Є.П., Халімовський М.О. Основи охорони праці: Підручник. 5-е вид. / За ред. М.П. Гандзюка. – К.: Каравела, 2018. – 384 с.

ДОДАТКИ

Тези VIII Міжнародної студентської науково-технічної конференції
«ПРИРОДНИЧІ ТА ГУМАНІТАРНІ НАУКИ. АКТУАЛЬНІ ПИТАННЯ»

Міністерство освіти і науки України
Тернопільський національний технічний університет
імені Івана Пулюя
Маріборський університет (Словенія)
Технічний університет в Кошице (Словаччина)
Каунаський технологічний університет (Литва)
Львівський національний університет
імені Івана Франка,
Гірничо-металургійна академія ім. Станіслава Сташиця (Польща)
Луцький національний технічний університет,
Чернівецький національний університет
імені Юрія Федьковича,
Вроцлавський економічний університет (Польща)
Університет технологій та економіки
імені Хелени Ходковської (Польща)
Донбаська державна машинобудівна академія



Студентське наукове
товариство



VIII МІЖНАРОДНА
студентська науково - технічна конференція
"ПРИРОДНИЧІ ТА ГУМАНІТАРНІ
НАУКИ.

АКТУАЛЬНІ ПИТАННЯ"

24-25 квітня 2025 р.

(збірник тез конференції)

Тернопіль 2025

УДК 004.8

Олійник І.- ст. гр. СН-42

Тернопільський національний технічний університет імені Івана Пулюя

ТЕХНОЛОГІЇ ВЕБ-РОЗРОБКИ КЛІЄНТ-СЕРВЕРНОГО РІШЕННЯ ДЛЯ ІНТЕРНЕТ-КОМЕРЦІЇ

Науковий керівник: к.т.н, доцент Дмитроца Л.П.

Oliiynyk I.

Ternopil Ivan Puluj National Technical University

THE TECHNOLOGIES OF WEB DEVELOPMENT FOR CLIENT- SERVER SOLUTIONS IN E-COMMERCE

Supervisor: PhD, Associate Professor Dmytrotsa L.P.

Ключові слова: технології, розробка, інтернет-комерція

Keywords: technologies, development, e-commerce

На сучасному етапі розвитку електронної комерції спостерігається стрімке зростання кількості онлайн-покупок. За даними досліджень [1], понад 70% споживачів надають перевагу інтернет-магазинам через зручність, економію часу та розширений асортимент. Відповідно, розробка якісного веб-ресурсу для електронної торгівлі стає критичним фактором конкурентоспроможності будь-якого бізнесу в цифровому середовищі.

Інтернет-комерція — це складна програмна система, яка реалізується із залученням цілої низки технологій. Одним з основних аспектів є використання клієнт-серверної архітектури, яка забезпечує взаємодію між користувачем (клієнтом) та сервером для обробки запитів, обміну даними та виконання операцій. На рівні структури веб-сторінки використовується мова HTML (HyperText Markup Language), що задає логічну розмітку елементів інтерфейсу. Для оформлення зовнішнього вигляду вебсторінки, адаптації під різні пристрої та дотримання принципів UX/UI-дизайну застосовується CSS (Cascading Style Sheets) — мова стилів, що дозволяє задавати кольори, шрифти, відступи, сітку розташування елементів та інші параметри зовнішнього вигляду сайту. Вони дозволяють створювати зручні та привабливі для користувачів інтерфейси, що оптимізуються під різні розміри екрану і мають привабливий вигляд, що важливо для утримання покупців на сайті.

Інтерактивна складова, зокрема динамічна взаємодія з елементами сторінки (перемикання блоків, оновлення вмісту без перезавантаження, робота кошика тощо), реалізується мовою JavaScript. З метою оптимізації користувацького досвіду активно застосовуються JavaScript-бібліотеки та фреймворки, такі як jQuery, React або Vue.js. Ці інструменти допомагають значно прискорити розробку, полегшити обробку даних на стороні клієнта і забезпечити інтерактивність, що безпосередньо покращує досвід користувачів.

На стороні сервера ключову роль відіграє мова PHP, яка відповідає за обробку запитів, реалізацію бізнес-логіки, керування сесій користувачів, авторизацію, обробку форм, генерацію динамічного контенту. Існують готові CMS-платформи, такі як OpenCart, WooCommerce, PrestaShop, які значно спрощують процес розробки, але для створення індивідуальних функціональних модулів часто вдаються до розробки «з

нуля». Вибір PHP обґрунтовано її популярністю та сумісністю з більшістю хостингів, а також наявністю широкої спільноти розробників, що дозволяє легко знаходити рішення для різних задач.

Для збереження структурованих даних про товари, клієнтів, замовлення, доставку та платежі використовується система керування базами даних, найчастіше MySQL або MariaDB. База даних виконує роль центрального сховища, забезпечуючи надійність та цілісність даних завдяки застосуванню транзакцій, зовнішніх ключів та резервного копіювання. Вибір цієї технології є практичним через її високу продуктивність, надійність та підтримку великої кількості користувачів одночасно.

Ще одним невід'ємним компонентом функціонального інтернет-магазину є платіжна система. Для забезпечення електронних розрахунків інтегруються такі сервіси, як PayPal або Stripe. Їх підключення вимагає дотримання протоколів безпеки, зокрема HTTPS, шифрування конфіденційних даних, аутентифікації транзакцій через API. Вибір цих сервісів ґрунтується на їхній надійності, популярності та здатності обробляти різноманітні платіжні методи з мінімальними ризиками для користувачів.

Серед обов'язкових елементів структури інтернет-магазину можна виділити: систему реєстрації та авторизації користувачів, адаптивний кошик покупця, модулі фільтрації та пошуку товарів, форму оформлення замовлення, розділи з відгуками та рейтингами товарів, адмін-панель для керування асортиментом, цінами та залишками на складі. Створення таких функцій потребує ретельного вибору технологій для забезпечення зручності користувачів та ефективного управління бізнесом.

Остаточний вибір технологій залежить від багатьох факторів, таких як бюджет проекту, час на розробку, специфікації та вимоги до продуктивності. Вибір HTML та CSS є абсолютно необхідним для будь-якого веб-ресурсу, оскільки ці мови відповідають за базову структуру і зовнішній вигляд сторінок. Для забезпечення інтерактивності оптимальним вибором є використання JavaScript з бібліотекою React, вона дозволяє швидко створювати складні інтерфейси з високою продуктивністю та зручністю для користувача. PHP є найбільш зручним інструментом для серверної частини, оскільки має велике ком'юніті підтримки, безліч бібліотек і фреймворків, а також можливість інтеграції з популярними CMS, такими як WooCommerce. Для збереження даних і виконання транзакцій вибір MySQL є виправданим завдяки надійності, підтримці транзакцій і масштабованості. А от щодо платіжної системи, то найкращим варіантом було обрано PayPal завдяки його надійності, світовій популярності, зручній інтеграції через API та високому рівню захисту як для покупців, так і для продавців.

Таким чином, розробка інтернет-магазину є мультидисциплінарним процесом, що поєднує знання веб-програмування, дизайну, баз даних, кібербезпеки та електронної комерції. Високий рівень технологій, які були обрані, відповідають найвищим вимогам щодо ефективності, безпеки, зручності та можливості масштабування. Такий підхід дозволяє створити не лише функціональний інтернет-магазин, а й зручний у використанні продукт, що задовольняє потреби як бізнесу, так і його клієнтів.

Література

1. Нові можливості для хостингу e-shop від Tucha. Tucha. URL: <https://tucha.ua/uk/blog/business/novi-mozhливosti-dlya-khostingu-e-shop-vid-tucha> (дата звернення: 14.04.2025).

Візуальна демонстрація інтернет-магазину «FreshFruit»

