

Київський національний торговельно-економічний університет

Кафедра комп'ютерних наук та інформаційних систем

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Розробка web-сайту торговельного підприємства з
інтерактивним модулем логістики»**

Студента 4 курсу, 13 групи,
спеціальності
122 «Комп'ютерні науки»

Тарнавського
Ростислава
Віталійовича

підпис студента

Науковий керівник
доктор технічних наук, професор

Краскевич Валерій
Євгенович

підпис керівника

Гарант освітньої програми
кандидат технічних наук, доцент

Демідов Павло
Георгійович

підпис керівника

Київ 2021

Київський національний торговельно-економічний університет

Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем
Спеціальність 122 «Комп'ютерні науки»

Зав. кафедри _____ **Затверджую**
Пурський О.І.
«20» грудня 2020р.

Завдання на випускню кваліфікаційну роботу (проект) студента

Тарнавського Ростислава Віталійовича

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи (проекту)
«Розробка web-сайту торговельного підприємства з інтерактивним модулем логістики»
Затверджена наказом ректора від «04» грудня 2020 р. № 4111
2. Строк здачі студентом закінченої роботи 29 травня 2021 року
3. Цільова установка та вихідні дані до роботи
Мета роботи: метою є обґрунтування та розробка web-сайту торговельного підприємства, з урахуванням сучасних тенденцій побудови організаційних та функціональних інформаційних структур підприємства торгівлі, а також .
Об'єкт дослідження: процес розробки інформаційної системи управління торговельним підприємством.
Предмет дослідження: засоби створення інформаційної системи управління торговельним підприємством
4. Перелік графічного матеріалу _____

5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Пурський О.І.	15.12.2020 р.	15.12.2020 р.
2	Пурський О.І..	15.12.2020 р.	15.12.2020 р.
3	Пурський О.І.	15.12.2020 р.	15.12.2020 р.

6. Зміст випускної кваліфікаційної роботи (проекту) (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА

1.1. Аналіз предметної області.

1.2. Класифікація веб-сайтів.

1.3 Види Інтернет-сайтів

РОЗДІЛ 2. РОЗРОБКА

2.1. Системний аналіз предметної області.

2.2. Проектування інфологічної моделі предметної області.

2.3. Розробка веб-сторінок.

РОЗДІЛ 3. ТЕСТУВАННЯ

3.1. Поняття тестування.

3.2. Тестування проекту.

3.3. Порівняння з аналогами.

3.4. Модуль логістики

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

7. Календарний план виконання роботи

№ пор.	Назва етапів випускного кваліфікаційного проекту	Строк виконання етапів роботи	
		За планом	фактично
1	2	3	4
1	<i>Вибір теми випускного кваліфікаційного проекту</i>	<i>01.10.2020</i>	<i>01.10.2020</i>
2	<i>Розробка та затвердження завдання на випускний кваліфікаційний проект</i>	<i>15.12.2020</i>	<i>15.12.2019</i>
3	<i>Вступ</i>		
4	<i>Розділ 1.</i>		

5	Розділ 2.		
6	Розділ 3.		
7	Висновки		
8	Здача випускного кваліфікаційного проекту на кафедрі науковому керівнику		
9	Попередній захист випускного кваліфікаційного проекту		
10	Виправлення зауважень, зовнішнє рецензування випускного кваліфікаційного проекту		
12	Представлення готового зшитого випускного кваліфікаційного проекту на кафедрі		
13	Публічний захист випускного кваліфікаційного проекту	За розкладом роботи ЕК	

Зав. кафедри
комп'ютерних наук та інформаційних систем
д.ф.-м.н., проф.

О.І.Пурський

8. Дата видачі завдання «15» грудня 2020 р.

9. Керівник випускної кваліфікаційної роботи (проекту)

Краскевич В.Є.

(прізвище, ініціали, підпис)

10. Гарант освітньої програми

Демідов П.Г.

(прізвище, ініціали, підпис)

11. Завдання прийняв до виконання студент-дипломник

Тарнавський Р.В.

(прізвище, ініціали, підпис)

12. Відгук керівника випускної кваліфікаційної роботи (проекту)

Керівник випускної кваліфікаційної роботи (проекту)

30.05.2021 р.
(підпис, дата)

13. Висновок про випускню кваліфікаційну роботу (проект)

Випускна кваліфікаційна робота (проект) студента Тарнавського Р.В.
(прізвище, ініціали)

може бути допущена до захисту в екзаменаційній комісії.

Гарант освітньої програми _____ Демідов П.Г.
(підпис, прізвище, ініціали)

Завідувач кафедри _____ Пурський О.І.
(підпис, прізвище, ініціали)

« _____ » _____ 2021 р.

ЗМІСТ

ВСТУП	10
РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА	12
1.1 Аналіз предметної області	12
1.2 Класифікація веб-сайтів.	16
1.3 Види Інтернет-сайтів	17
РОЗДІЛ 2. РОЗРОБКА ВЕБ-САЙТУ	23
2.1 Системний аналіз предметної області	23
2.2 Проектування інфологічної моделі предметної області	25
2.3 Розробка веб-сторінок.	28
РОЗДІЛ 3. ТЕСТУВАННЯ	38
3.1. Поняття тестування	38
3.2. Тестування сайту	39
3.3. Порівняння з аналогами.	42
3.4. Модуль логістики	54
Висновок	60
Список використаної літератури	62

АНОТАЦІЯ

До дипломної роботи Тарнавського Ростислава Віталійовича на тему: «Розробка web-сайту торговельного підприємства з інтерактивним модулем логістики»

Призначенням цієї дипломної роботи являється процес створення Інтернет-магазину на тематику продуктового маркету з інтерактивним модулем логістики.

Було досліджено аналогічні проекти та їх особливі відмінності між собою та цільовим проектом. На основі даного аналізу була розроблена концепція, за якою було створено проект.

SUMMARY

The purpose of this thesis is the process of creating an online store on the subject of the grocery market.

Similar projects and their special differences between them and the target project were investigated. Based on this analysis, the concept behind the project was developed.

СКОРОЧЕНІ УМОВНІ ПОЗНАЧЕННЯ

БД – бази даних;

СКБД – системи управління базами даних;

PM – реляційна модель;

ER-моделі – моделі типу «сутність – зв'язок»;

HTML – Hypertext Markup Language;

CSS – Cascading Style Sheets;

JS – JavaScript;

PHP – Personal Home Page Tools.

ВСТУП

Сьогодні, якою б діяльністю ми не займались, з кожним днем переконуємось все більше, що нам необхідний комп'ютер, адже коли вся інформація записана на паперах це так незручно. Але тут з'явився комп'ютер і бази даних, які значно зменшили труднощі, пов'язані з цим.

Досить часто в останні роки ми чуємо слово "веб-сайт". Сучасний світ зробив крок у бік цифрових технологій і крок цей подібний до стрибка в безодню. Ми дуже звикли до користування мережею Інтернет — зручного і що головне, миттєвого доступу до необхідної нам інформації. Наше життя буквально просочене інформацією і ми звикли одержувати її й ділитися нею у найкоротші строки.

Сам по собі Інтернет не є чимось новим та незвичним. Навіть в Україні, де ця мережа розвивалась із помітним запізненням, сьогодні ледь не у кожній домівці люди користуються мережею, черпаючи інформацію з різноманітних сайтів. Основною перевагою та перспективою цієї технології є те, що онлайн технологія може бути поєднана з оффлайн справами (покупками, роботою, відпочинком, тощо), а отже нашим життям, роблячи його ще комфортнішим та залишаючи більше часу для себе. Адже, для прикладу, пошук необхідного товару, консультація та перегляд відгуків значно економить наш час на подорожі по багатьом магазинам міста. Також мережа Інтернет пропонує значні переваги і для самих магазинів та підприємців, які створюють персональні вебсайти, збільшуючи тим самим кількість потенційних клієнтів та партнерів, а також економлячи кошти на оренду торгових приміщень та рекламі.

Бази даних нині широко застосовуються для збереження інформації на веб-ресурсах. В них можуть міститися різноманітні типи даних. MySQL [1], яка використовується в даному проекті, так як підтримує наступні типи: числові, строчкові, календарні та NULL. Така різноманітність типів даних в одній БД дає можливість зручного використання та підключення її до веб-ресурсу.

Метою кваліфікаційної роботи є створення бази даних «Магазин з продажу продуктів», використовуючи СКБД MySQL [2] і створення сайту, використовуючи мову програмування PHP [4] та CSS.

На сьогоднішній час, тема кваліфікаційної роботи є досить актуальною в Україні. Стрімкий ріст користувачів Інтернету та перехід комерції в онлайн торгівлю є головною причиною популярності та доцільності створення вебсайтів по наданню послуг та продажу товарів в мережі. Сфера застосування проектів цієї теми є досить широкою. Досить швидко та не виходячи із своєї домівки можна знайти необхідний товар і придбати його дешевше, аніж в простому магазині, адже в цьому випадку в ціну товару не закладена оренда торговельного приміщення. Наразі, ця ніша в сфері послуг ще не є достатньо заповненою та розвинутою з точки зору якості, безпеки та зручності користування, а тому потребує вдосконалення, розвитку та іноваційних рішень.

Для досягнення поставленої мети необхідно було вирішити наступні завдання:

- створення Інтернет-магазину;
- розробка інтуїтивно зрозумілого інтерфейсу;
- інтеграція бази даних MySQL [6];
- тестування сайту.

Проект створювався за допомогою таких технологій як: HTML, CSS, PHP [5], SQL, СКБД MySQL [3] та середовища розробки Sublime Text 3.

Об'єктом дослідження є веб-сайт, націлений та створений за концепцією Інтернет-магазину з продажу продуктів харчування. Даний проект має базу даних товарів [9], контакти для зворотнього зв'язку та інші типічні для сайтів такої концепції розділи.

РОЗДІЛ 1. ТЕОРЕТИЧНА ЧАСТИНА

1.1. Аналіз предметної області

В теоретичній частині описані програми та мови програмування, які я використовував під час розробки проекту.

Нині стало актуально користуватися різноманітними Інтернет-послугами та сервісами, тому що це зручно як для користувача (потенційного клієнта), так і власника ресурсу.

Чим це зручно для клієнта – є можливість не виходячи з житла, не витрачаючи час на подорож до відповідного закладу замовити ту чи іншу послугу через мережу [13]. Для власника – автоматизація процесу обробки, оформлення та інформування про замовлення.

Тому необхідно під час розробки встановити конкретні завдання:

- визначити інтереси цільового користувача;
- створити модель, за якою буде розроблений сайт;
- розробити адаптований для користувача інтерфейс;
- підтримка та інноваційні оновлення після релізу.

Даний проект націлений на продаж харчової продукції, тому необхідно врахувати дану інформацію та використати її для створення більш зручного інтерфейсу та функціоналу [10]. Враховуючи цей аспект, як варіант – створити в карточці продукту розділ з вартістю продукту, фото товару та його назву. Веб-сайт розбитий на 3 частини [1]:

- header_block;
- merge_block;
- footer_block.

Header_block – шапка сайту, розбита на 3 функціональні посилання («Продукты», «Корзина», «Кабинет») та пошукову строку.

Merge_block – область простору, виділений для розміщення в собі двох інших блоків – left_block та right_block. Left_block – має в собі асортимент товару [8]. Створена сітка за допомогою grid, а тому для зручності та універсальності розбитий на блоки, а вони в свою чергу

застосовують flex. Right_block – відображає детальну інформацію після натискання на блок одного з товарів.

Footer_block – область, створена для логічного завершення сайту. Виконує роль «заглушки». При бажанні можна розмістити посилання, інформацію, або ж додати функцію розширеної навігації для сайту.

JavaScript фреймворки:

JavaScript фреймворки – це інструменти для побудови динамічних веб/мобільних/настільних додатків на Javascript. Розробники використовують js-фреймворки там, де неможливо(складно або довго) виконувати завдання звичайними засобами. У переважній більшості випадків, фреймворки використовуються для написання, так званих, Single Page Applications.

Односторінковий веб-застосунок – це веб-застосунок чи веб-сайт, який вміщується на одній сторінці. В односторінковому веб-застосунку весь необхідний код - HTML, JavaScript, та CSS - завантажується разом із сторінкою або динамічно довантажується за потребою, зазвичай, у відповідь на дії користувача. Сторінка не оновлюється і не переправляє користувача на іншу сторінку. Взаємодія з застосунком часто здійснюється через динамічний зв'язок з веб-сервером.

За допомогою JavaScript можуть бути розроблені як повні веб-сайти, так і функціональні модулі (онлайн-інструменти). Зазвичай, повні кадри більше підходять для першого завдання, а для інших рекомендується використовувати більш світлі кадри або бібліотеки.

Рамки забезпечують прозору структуру програми та реалізуються за допомогою програми Найчастіше використовуються MVC (Model-View-Controller), Model-View-Presenter (MVP) і Model-View-ViewModel (MVVM).

Переваги побудови програми в рамках JS:

- Ефективність - Проекти, розроблені протягом багатьох місяців з сотнями рядків коду, тепер можуть бути реалізовані набагато швидше з добре підготовленими готовими шаблонами та функціями. Код стає значно меншим

і чистішим, що позитивно впливає на швидкість розвитку, а також на підтримку і усунення помилок програмного коду. 13

- Безпека - найкращі структури JavaScript мають фірмову систему безпеки і підтримуються великою спільнотою, члени та користувачі якої виступають лише тестерами.

- Витрати - більшість кадрів відкриті і безкоштовні. Оскільки вони допомагають програмістам швидше розвивати власні рішення, остаточна ціна веб-програми буде нижчою. Ви можете легко реалізувати SPA (Single Page Application). Наявність структури означає модульність програми, що полегшує роботу багатьох програмістів одночасно.

- Можливість швидко створити мобільний або комп'ютерний мультиплатформовий додаток з веб-версії за допомогою PhoneGap або Apache Cordova.

Існує багато програм на js-фреймворках, і цей сегмент швидко зростає.

Angular

Angular.js часто називають MVW (Model-View-Whatever) фреймворком, а основні переваги для нових і середніх компаній включають: швидке кодування, швидке тестування всіх частин програми та двосторонні канали передачі даних 14 (зміни, які миттєво відображаються у користувачеві). В даний час найпопулярнішою односторонньою прикладною рамкою для односторінкових додатків (додатків з одним SPA-сайтом) і є найбільша спільнота програмістів.

Angular2 має великий список функцій, які дозволяють розробляти все, починаючи від Інтернету, до настільних комп'ютерів і мобільних додатків. Рамка побудована на Microsoft TypeScript. Angular2 включає в себе архітектуру на основі компонентів, вдосконалену DI (ін'єкцію залежностей), ефективні служби журналів, взаємодії компонентів і багато іншого.

Обидві версії Angular є найкращим вибором для високоякісних корпоративних додатків або середовищ програмування, перш ніж читати код.

ReactJS Це бібліотека, а не фреймворк і доводить свою ефективність в динамічних додатках з високим трафіком (пропускнуою здатністю) як приклад Facebook і Instagram. Це найшвидше зростаюча платформа JS, сьогодні існує близько 1000 авторів Github.

У MVC (Model-View-Controller) React.js працює як "V" і може бути легко інтегрована в будь-яку архітектуру. Використання віртуального дерева DOM забезпечує більш високу продуктивність у порівнянні з кутом 1.x. Респонденти можуть бути створені та повторно використані в інших програмах або навіть відправлені для загального користування.

Відповідь важко вивчити, але це полегшує та спрощує створення додатків.

Ідеально підходить для складних, високопродуктивних програмних рішень.

Vue.js

Vue 2.0 був введений в 2016 році, ввів найкраще в Ember, React і Angular. Vue.js пропонує двонаправлене підключення до даних (як у AngularJS), візуалізацію на стороні сервера (як у Angular2 та ReactJS), Vue-cli (інструмент лісів) та додаткову підтримку JSX. Автори стверджують, що Vue2 є одним з найшвидших вільних мислителів. 15 Vue.js буде найкращим вибором для швидкої розробки багатоплатформних додатків. Це може бути міцною основою для ефективних односторінкових додатків (SPA) та реальних рішень, де продуктивність важливіша, ніж хороша організація коду або структура програми.

Ember.js

У 2015 році Ember була визнана найкращою структурою JS, залишивши React та AngularJS. В даний час у нього є велика спільнота розробників, регулярні оновлення та широко використовувані кращі практики в Java Script. Ember має двостороннє підключення до даних, як у AngularJS, все ще синхронізуючи вигляд і модель. Використання Fastboot.js забезпечує швидку

візуалізацію (перерахунок) дерева DOM на стороні сервера, покращуючи представлення складних інтерфейсів користувача.

Ember.js широко використовується в складних багатофункціональних вебдодатках і веб-сайтах. Провідні користувачі включають Chipotle, Blue Apron, Nordstrom, Kickstarter, LinkedIn, Netflix та багато іншого. Більш того, найпростіше навчитися і має багато навчальних посібників та онлайн-каталогів.

Meteor.js

Meteor є одним з найпопулярніших JavaScript фреймів, який має багато функцій для створення бекенда і переднього коду відтворення, управління базами даних і бізнес-логіки. З моменту його випуску в 2012 році його екосистема різко зростає. Ця повнофункціональна платформа дозволяє швидко створювати веб- та мобільні додатки. З точки зору продуктивності, всі зміни в базі даних негайно передаються користувальницькому інтерфейсу і назад без значних втрат часу через відмінності в мові або у відповідь на серверний час.

Meteor.js охоплює всі етапи циклу розробки програмного забезпечення і використовується для створення веб-додатків реального часу, таких як Mazda, ІКЕА, Honeywell і багато інших.

1.2. Класифікація веб-сайтів

Веб-сайт може бути реалізованим одним із типів – блогом, веб-порталом, сайтом-візиткою, корпоративним сайтом, промо-сайтом, Інтернет-магазин, віртуальною енциклопедією, та великою кількістю різних інших типів.

При цьому елементи які складають любий із названих сайтів, можуть бути дуже різноманітні, це текстова інформація, аудіо і відео-кліпи, чати, об'яви, форуми, статистичні зображення і анімація, а також багато іншого.

Веб-сайти бувають двох типів статистичні і динамічні:

Статистичний веб-сайт складається із статистичних сторінок, які створюються вручну, а лише потім зберігаються і завантажуються на сайт. Їх вміст постійний і змінюється дуже рідко. Коли необхідно змінити вміст сторінки, веб-майстер модифікує сторінку на робочому столі, зберігає а потім завантажує на сайт. Інформація на сайті оновлюється рідко. Сайт створений за допомогою HTML (Hyper Text Markup Language) мови розмітки гіпертексту, являється веб-сайтом статистичного типу і складається з декілька статистичних сторінок.

Динамічний веб-сайт це система яка готова до інформаційного оновлення. Він заснований на шаблонних сторінках, в які вставляється інформація, яка потім поновлюється, або змінюється. Вміст зберігається в БД.

1.3. Види Інтернет-сайтів

Блоги – це спеціальний тип веб-сайту, який дозволяє регулярно додавати персональні записи в online. В такому веб-журналі можна розміщувати статті коментарі та і по суті любую інформацію

Веб-портал – це веб-сайт який представляє користувачу мережі різні інтерактивні сервіси, (новини, пошук, пошту, інформацію про погоду), які працюють в рамках веб-сайту. Структура і набір функціональних модулів сайту-портал влаштовані таким чином, щоб відвідувач сайту міг у повному обсязі отримувати потрібну інформацію. Як правило, на сайті-порталі завжди є стрічка новин і подій, форуми, де можна обговорити цікаву тему, висловити свою думку. Також портали працюють як точки доступу до інформації у Інтернеті або сайти, що допомагають користувачам у пошуку потрібної інформації через Інтернет. Такі портали представляють інформацію з різних джерел або тем об'єднаним способом і також називають навігаційними сайтами. Класифікувати портали можна за різними ознаками, але найчастіше вдаються до класифікації за призначенням. В даний час за цією ознакою розрізняють три основних типи порталів:

Загальнодоступні, або горизонтальні, портали (звані іноді мега-порталами), такі як Yahoo!, Lycos, Excite, Rambler. Ці портали призначені для найширшої аудиторії, що відбивається на утриманні наданої ними інформації та послуг – зазвичай вони носять загальний характер (наприклад, новини про політичні події і культурного життя, електронна пошта, новинні розсилки і т.д.). Сфера діяльності таких порталів перетинається зі сферою діяльності засобів масової інформації, тому останнім часом спостерігаються процеси злиття загальнодоступних порталів і засобів масової інформації в рамках однієї компанії.

Вертикальні портали призначені для специфічних видів ринку і обслуговує аудиторію, що користується послугами цього ринку або працює на ньому. Прикладами таких порталів можуть служити програми B2C (Business-to-consumer), наприклад туристичні агентства, що надають послуги з бронювання місць в готелях, замовлення і доставки квитків, доступу до карт і відомостей про автомобільні маршрути і т.д., або портали типу B2B (business-to-business), що дозволяють своїм клієнтам реалізовувати спільні бізнес-операції (наприклад, вибирати постачальників і здійснювати закупівлю товарів, проводити аукціони і т.д.). Число подібних порталів останнім часом швидко зростає.

Корпоративні портали призначені для співробітників, клієнтів і партнерів одного підприємства (іноді вони називаються B2E-портали (Business-to-employees)). Користувачі такого порталу дістають доступ до призначених їм сервісів і прикладних програм в залежності від їх ролі і персонального профілю, і це найбільш цікава категорія порталів в плані реалізації корпоративної інфраструктури і інтеграції додатків. Призначенням корпоративного порталу є надання зовнішнім і внутрішнім користувачам можливості персоніфікованого доступу до всіх корпоративних даних і додатків (включаючи неструктуровані і різнорідні дані), об'єднання ізольованих моделей бізнесу, інтеграція різних корпоративних додатків (у тому числі додатків бізнес-партнерів), забезпечення повноцінного

цілодобового доступу всіх користувачів до ресурсів компанії 24 години на добу незалежно від їх місця перебування [2].

Сайт-візитка – це міні-сайт, який складається не більше ніж з 10 сторінок і надає найбільш важливу інформацію про компанію, товари і послуги, а також контактні дані. Переваги: економія коштів на розповсюдження інформації про фірму, а також проведення маркетингових досліджень, клієнти легко можуть знайти потрібну інформацію про продукти, а також про компанії, співробітники менше відволікаються на телефонні дзвінки надаючи інформацію, яку можна розмістити на сайті. Потенційні зацікавлені партнери можуть легше знайти інформацію про фірму в мережі. Мета створення сайту візитки – заявити про себе і про діяльність компанії мільйонам потенційних клієнтів, які є користувачами Інтернету і щоденного пошуку товарів і послуг в мережі. Сайт візитка ідеально підходить для малого бізнесу або підприємців.

Корпоративний сайт – це сайт, який виконує функції як сайту-візитки, так і електронного магазину або каталогу. Зазвичай це дизайн, який зроблений на основі логотипу і корпоративного стилю, інформації про компанію. Такий сайт як правило, містить новини, інформаційні сторінки, каталог товарів або послуг та форму заявки або форму для зв'язку. Корпоративні сайти можуть містити системи, які дозволяють взаємодіяти з партнерами (обмін документами, швидке формування замовлення з урахуванням історії переговорів, онлайн переговори і так далі). Корпоративний сайт містить повну інформацію про організацію: цілі, завдання, політика, керівництво, принципи роботи, плани розвитку [3].

Промо-сайт – це Інтернет-ресурс, основна мета якого полягає в тому, щоб піднести певний товар, послугу, подію або бренд і схилити покупця саме до його вибору. У цьому й полягає головна відмінність від сайту-візитки чи порталу, в яких подається загальна інформація про всі продукти без акценту на якому-небудь одному. Можна провести певну аналогію між промо-сайтом і буклетом або афішою. Як правило, промо-сайти використовуються для залучення уваги під час невеликих по тривалості рекламних кампаній або ж

формування іміджевої складової діяльності фірми. Зазвичай один промо-сайт присвячується одному продукту і максимально фокусується на взаємодії з користувачами. Зазвичай такі створюють напередодні події або виходу нового товару. Ще однією відмінністю є велика кількість графіки і мультимедіа-презентацій нового товару. Найчастіше, промо-сайти є інтерактивними, тобто відвідувач може впливати, наприклад, на колір товару або його комплектацію, щоб краще зрозуміти як потенційна покупка буде виглядати в іншому розфарбуванні. Результатом створення і просування промо-сайту стає збільшення продажів за рахунок збільшення пізнаваності продукту і отримання про нього більшої кількості інформації [4].

Інтернет-магазин – це веб-сайт який представляє товар або послугу і який дозволяє зробити замовлення на покупку в режимі online. Користувач обирає варіант для розрахунку який його влаштовує, спосіб доставки товару, і отримує рахунок на оплату. Загальнодоступність мережі Інтернет дозволяє багатьом організаціям використовувати електронні канали зв'язку, щоб здійснювати продаж своїх товарів і послуг online. Там де з'являються платоспроможні клієнти для яких існують відповідні потреби дуже швидко знаходяться рішення для задоволення цих потреб. Одним з таких рішень є використання електронних засобів що дозволяє розширити коло потенційних клієнтів. В результаті навіть маленька фірма в відносно короткий період часу може трансформуватись в дійсного комерційного гіганта. Електронна комерція (e-commerce) термін який використовують для визначення комерційної активності в Інтернет мережі. Визначає можливість реалізації покупок, продаж, сервісного обслуговування, проведення маркетингових заходів з допомогою мережі [2]. Основні переваги Інтернет магазинів, які роблять їх вигідними і все більш популярними:

- Економія часу. Основною перевагою Інтернет магазинів є економія часу. Швидше за все кожен при відвідуванні звичайного магазину стикався з ситуацією, коли необхідно знайти консультанта, потім дочекатися коли принесуть потрібний товар і,

врешті-решт, вистояти в черзі. Крім цього необхідно дістатися до магазину, а це час, якого при нинішньому ритмі життя і так не вистачає. З Інтернет магазином таких проблем не буде, покупцю варто лише підібрати на сайті необхідний товар, після чого просто оформити замовлення. Максимум це займе кілька хвилин часу.

- Низькі ціни. Ще одною важливою перевагою Інтернет магазинів є порівняно низькі ціни на товар. У зв'язку з тим, що Інтернет магазин не витрачає кошти на оплату праці консультантів, касирів і всіляких службовців, немає потреби в оренді торгового залу, а це чималі гроші, – ціна товару в Інтернет магазині суттєво відрізняється від магазинної, і є майже оптової. Також в Інтернеті покупці не обмежені одним Інтернет магазином, є спеціальні сервіси, які можуть відразу надати ціни по декількох фірмах. Клієнтові залишається лише уточнити всі тонкощі на сайті і зробити замовлення.
- Великий вибір асортименту. Великий вибір послуг, товарів і послуг. Звичайні магазини обмежені кількістю товару, що викликано прив'язкою до конкретного приміщення, будь то вітрина або торговий зал. Крім того клієнт має доступ до Інтернет магазинах розташованих в інших регіонах і навіть країнах, що дає можливість замовити товар з будь-якої точки світу. Що дає в свою чергу необмежені можливості по вибору асортименту.
- Доставка. До переваг можна віднести і доставку товарів. Кожен Інтернет магазин має службу доставки, що дає можливість клієнту отримати товар по вказаній адресі. Не треба тягнути важкі ящики або пакети, витрачати додаткові гроші на транспорт. Крім цього в багатьох випадках, вартість доставки може входити у вартість товару, що дуже вигідно.

- Відгуки клієнтів. Якісний опис товару і огляд всіляких характеристик. У звичайному магазині перед консультантами в першу чергу стоїть завдання продати товар, а потім вже турбота про клієнта. Тому є великий шанс придбати не якісний товар опираючись на рекомендації консультантів. На відміну від звичайних, в Інтернет магазинах, для кожного товару або послуги, як правило, в описі знаходяться відгуки клієнтів, які дають чітке уявлення про якість і надійність товару.
- Доступність. Маючи під рукою комп'ютер або телефон з доступом до Інтернету, людина легко може здійснити покупку. Не треба кудись їхати чи бігти, і навіть сидячи на робочому місці можна за декілька хвилин зробити замовлення [5].
- Онлайн енциклопедія. В Інтернеті розміщено багато сайтів, що містять освітню інформацію. Наприклад, такими є Інтернет- або веб-енциклопедії. Інтернет-енциклопедія – це веб-сайт, сторінки якого містять енциклопедичні знання про систему пошуку потрібних відомостей. Існують універсальні та спеціалізовані Інтернет-енциклопедії [6].

РОЗДІЛ 2. РОЗРОБКА

2.1. Системний аналіз предметної області

Процес проектування БД являє собою послідовність переходів від неформального словесного опису інформаційної структури предметної області до формалізованого опису об'єктів предметної області в термінах деякої моделі. У загальному випадку можна виділити наступні етапи проектування:

Системний аналіз і словесний опис інформаційних об'єктів предметної області;

– Проектування інфологічної моделі предметної області - частково формалізований опис об'єктів предметної області в термінах деякої семантичної моделі, наприклад, в термінах E-моделі;

– Даталогічне або логічне проектування БД, тобто опис БД в термінах прийнятих діалогічною моделлю даних;

– Фізичне проектування БД, тобто вибір ефективного розміщення БД на зовнішніх носіях для забезпечення найбільш ефективної роботи програми.

На цьому етапі проектування бази даних необхідно провести докладний словесний опис об'єктів предметної області і реальних зв'язків, які присутні між описаними об'єктами.

Існують два підходи до вибору складу і структури предметної області:

1) функціональний підхід – він реалізує принцип руху «від завдань» і застосовується тоді, коли заздалегідь відомі функції певної групи осіб і комплексів задач, для обслуговування інформаційних потреб яких створюється розглянута БД. У цьому випадку ми можемо чітко виділити мінімальний необхідний набір об'єктів предметної області, які повинні бути описані;

2) предметний підхід – коли інформаційні потреби майбутніх користувачів БД жорстко не фіксуються. Вони можуть бути багатоаспектний і вельми динамічними. Ми не можемо точно виділити мінімальний набір об'єктів предметної області, які необхідно описувати. В опис предметної області в

цьому випадку включаються такі об'єкти і взаємозв'язки, які найбільш характерні і найбільш істотні для неї. БД, конструюється при цьому, називається предметної, тобто вона може бути використана при вирішенні безлічі різноманітних, заздалегідь не визначених завдань. Конструювання предметної БД в деякому сенсі здається набагато більш привабливим, однак труднощі загального охоплення предметної області з неможливістю конкретизації потреб користувачів може привести до надмірно складною схемою БД, яка для конкретних завдань буде неефективною.

Був використаний компромісний варіант, який, з одного боку, орієнтований на конкретні завдання або функціональні потреби користувачів, а з іншого боку, враховує можливість нарощування нових додатків.

Дана БД орієнтована на використання в магазині продуктів. БД була створена, базуючись на основних потребах магазину, таких як:

- № відділу та назва;
- № виробника та назва;
- персональні дані;
- список продуктів;
- термін придатності;
- ціна;
- імена працівників;
- стаж;
- оклад;
- інформація про постійних покупців;
- знижки.

Для того, щоб зберігати вище перераховані дані, необхідно створити БД, в якій обов'язковими таблицями мають бути:

- «Otdel» – для зберігання номера відділу і його назви;
- «Proizvoditel» – для зберігання даних про виробника товарів;
- «Product» – дані про продукти (назва, термін зберігання, ціна);

- "Rabotniki" – інформація про працівників магазину;
- "Postoyannie_rokupateli" – інформація про покупців і знижки.

2.2. Проектування інфологічної моделі предметної області

Інфологічна модель застосовується на другому етапі проектування БД, тобто після словесного опису предметної області.

Інфологічна модель – сукупність опису сутностей, атрибутів та ключів, які вміщують в собі дані та пов'язані між собою.

Реляційна модель – це модель даних, де текстова чи числова інформація, що зображається за допомогою таблиць. Кожна таблиця, яка називається відношенням, складається з рядків, які називаються кортежами, та стовпчиків, які називаються атрибутами.

Інфологічне проектування перш за все пов'язано зі спробою уявлення семантики предметної області в моделі БД. РМ даних в силу своєї простоти і лаконічності не дозволяє відобразити семантику, тобто зміст предметної області. Ранні теоретико-графові моделі більшою мірою відображали семантику предметної області. Вони в явному вигляді визначали ієрархічні зв'язки між об'єктами предметної області.

На даний момент не існує єдиної загальноприйнятої системи позначень для ER-моделі і різні CASE-системи використовують різні графічні нотації.

У розробленій БД є 5 таблиць, які з'єднані між собою зв'язком «один до багатьох» (рис. 2.1).

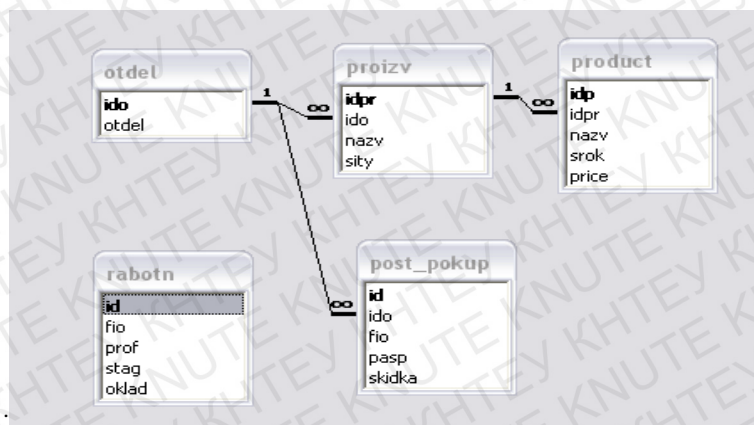


Рис. 2.1. Зв'язок таблиць.

Для створення бази даних необхідно на головній сторінці програми phpMySQL_Admin в поле «створити нову БД» ввести ім'я бази даних і натиснути кнопку «Створити» (рис 2.2).

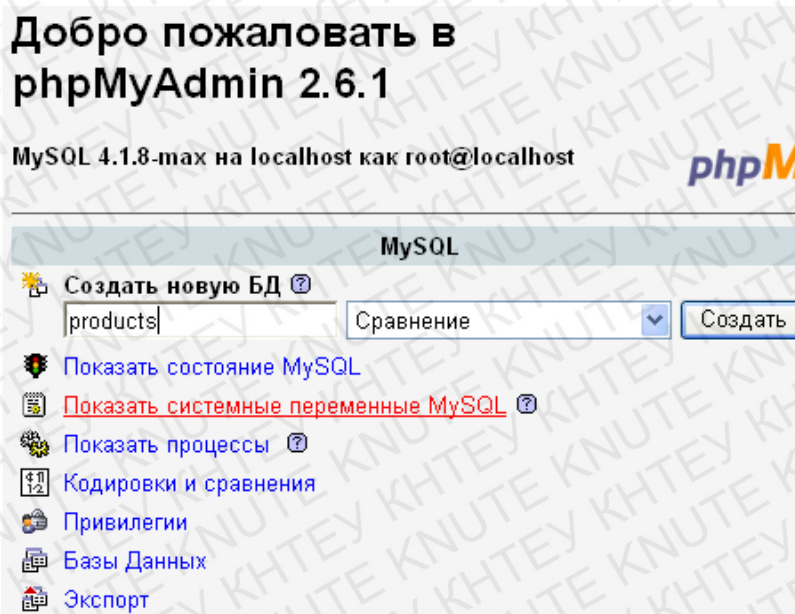


Рис. 2.2. Створення нової бази даних

Далі проілюстрована процедура створення таблиць в новій БД (рис 2.3). Кожна з таблиць була створена задля збереження в собі відповідної, тому кількість полів була обрана за необхідними критеріями.

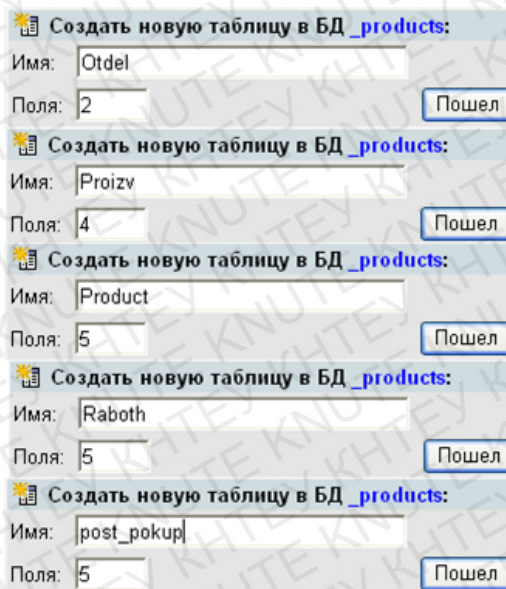


Рис. 2.3. Створення таблиць БД.

Для створення полів у вище вказаних розділах було обрано програму phpMySQL_Admin. У кожний з розділів додано відповідні дані. На

рисунку (рис 2.4) зображено фрагменти полів «Otdel», «Proizv», «Product», «Raboth», «post_pokup».

Поле	Тип	Сравнение	Атрибуты	Ноль	По умолчанию	Дополнительно
<input type="checkbox"/> ido	smallint(6)			Нет		auto_increment
<input type="checkbox"/> otdel	varchar(255)	cp1251_general_ci		Нет		
Поле	Тип	Сравнение	Атрибуты	Ноль	По умолчанию	Дополнительно
<input type="checkbox"/> idpr	smallint(6)			Нет		auto_increment
<input type="checkbox"/> ido	smallint(6)			Нет	0	
<input type="checkbox"/> nazv	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> city	varchar(255)	cp1251_general_ci		Нет		
Поле	Тип	Сравнение	Атрибуты	Ноль	По умолчанию	Дополнительно
<input type="checkbox"/> idp	smallint(6)			Нет		auto_increment
<input type="checkbox"/> idpr	smallint(6)			Нет	0	
<input type="checkbox"/> nazv	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> srok	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> price	varchar(255)	cp1251_general_ci		Нет		
Поле	Тип	Сравнение	Атрибуты	Ноль	По умолчанию	Дополнительно
<input type="checkbox"/> id	smallint(6)			Нет		auto_increment
<input type="checkbox"/> fio	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> prof	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> stag	varchar(25)	cp1251_general_ci		Нет		
<input type="checkbox"/> oklad	mediumint(9)			Нет	0	
Поле	Тип	Сравнение	Атрибуты	Ноль	По умолчанию	Дополнительно
<input type="checkbox"/> id	smallint(6)			Нет		auto_increment
<input type="checkbox"/> ido	smallint(6)			Нет	0	
<input type="checkbox"/> fio	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> rasp	varchar(255)	cp1251_general_ci		Нет		
<input type="checkbox"/> skidka	varchar(10)	cp1251_general_ci		Нет	0	

Рис. 2.4. Створення полів БД.

2.3. Розробка веб-сторінок

Створено базу даних, яка дала можливість користувачеві виконувати певні дії за допомогою додатків PHP. Спочатку створено HTML сторінки, з яких будуть викликатися додатки PHP.

На головній сторінці розміщено кілька розділів меню: «Продукти», «Корзина», «Кабінет». В кожному з розділів меню міститься відповідна інформація, контент та функціонал. Наприклад, у пункті «Продукти» ... Окрім цього міститься поле для пошуку. Це реалізується за допомогою блоку HTML header та каскадної таблиці стилів CSS (рис 2.5):



Рис. 2.5. Головна сторінка.

Основа веб-сайту- 3 блоки:

- header_block;
- merge_block;
- footer_block.

Header_block – шапка сайту, розбита на 3 функціональні посилання («Продукты», «Корзина», «Кабинет») та пошукову строку (рис 2.6).

```

<header class="header_block" style="display: flex;" >
  <div class="productions"><a class="prod" href="#">Продукты</a>
</div>
  <div class="shoping"><a class="shop" href="#">Корзина</a></div>
  >
  <div class="account"><a class="acc" href="#">Кабинет</a></div>
  <div class="search_panel_container">
    <span class="icon"><i class="fa fa-search"></i></span>
    <div class="search_panel"><input type="search" name="finder" placeholder="Поиск..."></div>
  </div>

```

Рис. 2.6. Header_block. Шапка.

Merge_block – область простору, виділеного для розміщення в собі двох інших блоків – left_block та right_block (рис 2.7).

```

<div class="merge_block">
  <div class="left_block">
  </div>
  <div class="right_block">
  </div>
</div>

```

Рис 2.7. Merge_block. Контент.

Left_block – має в собі асортимент товару. Створена сітка за допомогою grid, а тому для зручності та універсальності розбитий на блоки, а вони в свою чергу застосовують flex (рис 2.8).

```

<div class="left_block">
  <div class="one_block">
    <div id="gridOne">Яблука </div>
    <div id="gridTwo">Груші </div>
  </div>
  <div class="two_block">
    <div id="gridThree">Апельсини </div>
    <div id="gridFour">Мандарини </div>
  </div>
  <div class="three_block">
    <div id="gridFive">Ананаси </div>
    <div id="gridSix">Манго </div>
  </div>
  <div class="four_block">
    <div id="gridSeven">Помело </div>
    <div id="gridEight">Банани </div>
  </div>
  <div class="five_block">
    <div id="gridNine">Папайя </div>
    <div id="gridTen">Кокоси </div>
  </div>
  <div class="six_block">
    <div id="gridEleven">Ківи </div>
    <div id="gridTwent">Малина </div>
  </div>
</div>

```

Рис. 2.8. Left_Block. Таблиця товарів.

Right_block – відображає детальну інформацію після натискання на блок одного з товарів. В цій області можна додати товар до кошика (рис 2.9). На ранньому етапі створення веб-сайту ціна не завантажувалась із бази даних, а була записана як блок.

```

<div class="right_block">
  <div class="right_block_text" style="text-align: center; font-size:
    xx-large;"></div>
  <div class="right_block_image"></div>
  <div style="text-align: center; font-size: large;">0 товар</div>
  <div class="right_block_info">
    <div style="padding-left: 15px;"></div>
    <div style="padding-right: 15px; text-align: center;"></div>
    <a href="#"><div style="text-align: center; margin-top: 80px;
      text-decoration-line: underline; color: red">Додати товар в
      корзину</div></a>
  </div>

```

Рис.2.9. Right_Block. Опис продукту

Footer_block – область, створена для логічного завершення сайту. Виконує роль «заглушки» (рис 2.10).

```
<div class="footer_block"></div>
```

Рис. 2.10. Footer_Block

Для оформлення сайту була застосована каскадна таблиця стилів CSS.

Шапка сайту header_block була оформлена з ціллю не занадто контрастувати із фоновим кольором сайту. Це рішення було обране задля того, щоб не перенавантажувати очі користувача та інтерфейс мав естетичний та гармонічний вигляд.

На рисунку 2.11 показаний код, який відповідає за оформлення шапки сайту. На зображеннях видно такі атрибути, як:

- background-color – він описує колір фоновий колір блоку. В ньому вказаний параметр rgba, який розуміється як три відтінки кольорів – червоний, зелений та синій, а також альфа канал, який дає змогу встановити прозорість;
- box-shadow – це атрибут, який додає тінь до об'єкту. Для більш зручного калібрування позиції тіней використовувався ресурс www.cssmatic.com/box-shadow;
- transition-duration – задає час в мілісекундах, на протязі якого триває анімація. У даному випадку час виконання анімації 0.3 секунди;

```
.header_block{
  background-color: rgba(72, 77, 77, 0.3);
  box-shadow: 0 5px 15px rgba(0,0,0,0.5);
  transition-duration: 0.3s;
  transition-timing-function:ease-in;
}
.header_block:hover{
  background-color: rgba(72, 77, 77, 0.2);
  box-shadow: 0 0 10px 1px rgba(0,0,0,0.4);
}
```

Рис. 2.11. Оформлення шапки сайту.

– `transition-timing-function` – атрибут, який визначає час виконання та прискорення анімації через математичну функцію. У даному рішенні застосовано параметр `ease-in`, яка починається повільно і уповільнюється під кінець;

Далі на рисунку 2.11 застосовується псевдоклас `:hover`. Він визначає стиль елемента після наведення на нього курсору. В даному випадку при наведенні курсору на шапку сайту фоновий колір стає світлішим та зменшується затінена область.

На рисунку 2.12 описано позиціонування блоків. Були використані атрибути:

– `padding` – значення полів навколо вмісту елемента. Властивість `padding` дозволяє задати величину поля одразу для всіх сторін елемента або визначити величину поля одразу для всіх сторін елемента або визначити її для вказаних сторін. Значення в даному випадку вказане у відсотках;

– `margin-right` – встановлює величину відступу від правого краю елемента. Відступом являється відстань від зовнішнього краю правої межі поточного елемента до внутрішньої межі батьківського елемента. Встановлено значення `auto`, яке автоматично визначається браузером;

– `padding-left` – відступ від лівого краю встановлений у відсотках;

– `font-size` – розмір шрифту. Встановлене значення `x-large`;

– min-width та max-width – ширина, займає значення 10%. Це було застосовано для фіксованого позиціонування.

```
.productions, .shoping, .account, .search_annel_container{  
padding: 1.7%;  
margin-right: auto;  
padding-left: 5%;  
font-size: x-large;  
min-width: 10%;  
max-width: 10%;  
}
```

Рис. 2.12. Оформлення блоків в шапці.

Оформлення блоку, в якому розміщена строка пошуку (рис 2.13):

– width – ширина блоку, встановлена в 300 пікселів;
– vertical-align: middle – вирівнювання контенту по середині;
– white-space – атрибут, який встановлює вигляд відображення пробілів між словами. Значення nowrap – пробіли не враховуються, переноси рядків в коді HTML ігноруються, весь текст відображається однією строкою.

– position встановлює спосіб позиціонування елемента відносно вікна браузера чи інших об'єктів на веб-сторінці. Relative – положення елемента встановлюється щодо його вихідного місця. Додавання властивостей left, top, right і bottom змінює позицію елемента і зрушує його в ту чи іншу сторону від первісного розташування.

```
.search_panel_container{  
width: 300px;  
vertical-align: middle;  
white-space: nowrap;  
position: relative;
```

Рис. 2.13. Оформлення блоку пошукової строки.

Merge-block – блок призначений для розміщення контенту в вигляді товару (рис 2.14). Атрибути:

- `display` – встановлюється для адаптивності елементів та їх вирівнювання;
- `min-height` та `max-height` – висота займає значення 82%. Це було застосовано для фіксованого позиціонування;
- `margin-left` та `-right` встановлює величину відступу від лівого та правого країв елемента зі значенням 10 пікселів;
- `padding` – відступи по 10 пікселів по вертикалі, горизонталі, зправа та зліва.

```
.merge_block{
  display: flex;
  min-height: 82%;
  max-height: 82%;
  margin-left: 10px;
  margin-right: 10px;
  padding-top: 10px;
  padding-bottom: 10px;
  padding-left: 10px;
  padding-right: 10px;
}
```

Рис. 2.14. Оформлення merge-block.

`Left_block` має наступні атрибути (рис 2.15):

- `max-width` та `min-width` – максимальна та мінімальна ширина елемента. Фіксована.
- `display` – визначає положення вигляду в документі. `Grid` – є перехресним набором горизонтальних і вертикальних ліній, що перетинаються - один набір визначає стовпці, інший рядки. Елементи можуть бути розміщені на сітці, дотримуючись цих стовпців та рядків.
- `text-align` – визначає вирівнювання тексту. `Center` – вирівнює текст по центру елемента.
- `grid-template-columns` – кількість та позиціонування колонок відносно таблиці `grid`.
- `grid-template-rows` – кількість рядків та їх позиціонування відносно таблиці `grid`.
- `justify-content` – вирівнює елементи гнучкого контейнера, коли елементи не використовують весь доступний простір на головній осі контейнера (по

горизонталі). Це робиться після того, як застосовуються розміри і автоматичні відступи, за винятком ситуації, коли принаймні у одного елемента flex-grow більше нуля. При цьому не залишається ніякого вільного простору для маніпулювання. Space-content – флекси рівномірно розподіляються по всьому рядку. Порожній простір перед першим і після останнього елементів дорівнює половині простору між двома сусідніми елементами.

```
.left_block{
  /*background-color: red;*/
  max-width: 64%;
  min-width: 64%;
  display: grid;
  text-align: center;
  grid-template-columns: 2fr 2fr;
  grid-template-rows: 1fr 1fr 1fr;
  justify-content: space-around;
}
```

Рис 2.15. Оформлення left_block.

Right_block має атрибути (рис 2.16):

- min-width та max-width – мінімальна та максимальна ширина, встановлена зі значенням 64%;
- border-left-width – ширина лівої рамки, 10 пікселів.

```
.right_block{
  /*background-color: blue;*/

  min-width: 34%;
  max-width: 34%;
  border-left-width: 10px;
}
```

Рис. 2.16. Оформлення Right_block.

Footer_block (рис 2.17):

- width: all – вказує розмір елемента по всій доступній ширині відносно інших елементів.

- background-color: rgba – фоновий колір елемента з альфа-каналом;
- height – висота, 8%;

В даного елемента присутня анімація при наведенні на неї курсору. Вона реалізована за допомогою псевдокласу :hover, як вказано на рисунку 2.17:

- background-color – в даному випадку значення кольору було встановлене таким чином, щоб відтінок став більш світлим;
- position: relative – позиціонування встановлюється відносно вікна та розміру браузера;
- min-width: all – ширина займає всю доступну область.

```
.footer_block{
  width: all;
  background-color: rgba(72, 77, 77, 0.3);
  height: 8%;
}

.footer_block:hover{
  background-color: rgba(72, 77, 77, 0.2);
  position: relative;
  min-width: all;
}
```

Рис. 2.17. Оформлення Footer_block.

- display: flex – адаптивне розміщення на екрані відносно розміру вікна;
- flex-direction: row – візуальне вирівнювання та масштабування рядків із рівномірними відступами між сусідніми елементами;
- min-width – мінімальна ширина 50 відсотків;
- padding-top – відступ від верхнього елемента 10 пікселів;
- padding-bottom – відступ від нижнього елемента 10 пікселів;
- justify-content: space-around – вирівнює елементи гнучкого контейнера, коли елементи не використовують весь доступний простір на головній осі контейнера. Space-content – флекси рівномірно розподіляються по всьому рядку. Порожній простір перед першим і після останнього елементів дорівнює половині простору між двома сусідніми елементами.

```

.sub_block{
  display: flex;
  flex-direction: row;
  min-width: 50%;

  padding-top: 10px;
  padding-bottom: 10px;

  justify-content: space-around;
}

```

Рис. 2.18. Оформлення таблиці товарів.

Ефект збільшення зображення товару при наведенні на нього курсору миші застосований за допомогою коду, вказаного на рисунку 2.19.

```

.img_zoom:hover{
  transform: rotate(0deg) scale(1.5) skew(0deg) translate(0px);
  -webkit-transform: rotate(0deg) scale(1.5) skew(0deg) translate(0px);
  -moz-transform: rotate(0deg) scale(1.5) skew(0deg) translate(0px);
  -o-transform: rotate(0deg) scale(1.5) skew(0deg) translate(0px);
  -ms-transform: rotate(0deg) scale(1.5) skew(0deg) translate(0px);
}

```

Рис. 2.19. Анімація збільшення зображення товару.

Відображення товару в right_block було реалізовано за допомогою JavaScript. Код написаний на JS показаний на наступних рисунках, а саме:

– створення змінних (рис. 2.20);

```

<script>
let itemsBlock = document.querySelector('.left_block');
let previewBlockTitle = document.querySelector('.right_block .right_block_text');
let previewBlockPhoto = document.querySelector('.right_block .right_block_image img');
let previewBlockDescription = document.querySelector('.right_block .right_block_info .right_block_description');
let previewBlockPrice = document.querySelector('.right_block .right_block_info .right_block_price');

```

Рис. 2.20. Створення локальних змінних.

– Для рендерингу зображення в правому блоці написаний код, показаний на рисунку 2.21. З даного опису можна побачити створення зарезервованих для контенту комірок, які після натискання на блок товару в left_block, повертають значення в зарезервовані комірки та відображають їх у right_block.

```

function renderPreviewBlock(item) {
  // Передається товар і відображається зправа
  previewBlockTitle.textContent = item.title;
  previewBlockPhoto.setAttribute('src', item.photo);
  previewBlockDescription.textContent = item.description;
  previewBlockPrice.textContent = item.price;
}

function selectItem(item) {
  // Відображається вибраний товар
  renderPreviewBlock(item);
}

function renderSmallItem(item) {
  // Генерується HTML із інформації про товар
  let itemBlock = document.createElement('div');
  itemBlock.innerHTML = `<div class="small_block">
    <div class="itemTitle">${item.title}</div> </div>Ціна - ${item.price}</div>
  </div>`;
  itemBlock.onclick = () => selectItem(item);
  return itemBlock;
}

function renderSmallItems(items) {
  // Очищення товарів
  itemsBlock.innerHTML = '';

  // Кожен товар відображається зліва
  items.forEach(item => {
    let renderedItem = renderSmallItem(item);
    itemsBlock.appendChild(renderedItem);
  });
}

```

Рис.2.21. Функція відображення зображення в right_block.

РОЗДІЛ 3. ТЕСТУВАННЯ

3.1. Поняття тестування

Перш за все – сучасний Інтернет-ресурс має стабільно працювати, тому необхідно своєчасно знаходити баги та виправляти їх, для цього потрібно проводити різні тести проекту. Тестування сайту – це необхідний крок, в процесі виконання якого можливо знайти та виправити помилки [16].

Для якісного тестування необхідно вносити та описувати всі помилки, або ж відхилення від запланованого вигляду чи виконання завдання [20].

Залежно від складності сайту визначається час необхідний для його перевірки і налагодження. Тестування проекту може займати до 50% відсотків від часу розробки [17]. Окрім цього – вартість даного процесу також може сягати значення до половини виділеного бюджету на розробку проекту, тому під час процесу розробки потрібно намагатись робити якомога менше помилок в коді.

Процес тестування можна виконати не лише власними силами, а й за допомогою спеціалізованих ресурсів, які не лише вказують на помилки в коді, а й надають рекомендації по оптимізації коду. Також можливо в нашій час звернутися до команди тестувальників, але дане рішення не являється актуальним для бюджетного проекту.

Інтернет ресурси мають також певні стандарти, сформовані в період розвитку, аналізу та вимог пошукових систем [18]. Виходячи з основних стандартів розробки, складається тест план, за яким виконується перевірка. Однозначно неправильно проводити тестування всіх сайтів по одному списку, тому перед початком робіт тест план опрацьовується безпосередньо під проект.

В процесі розробки зустрічалась велика кількість багів у вигляді не написаних знаків синтаксису в використовуваних мовах програмування. Дані

типи помилок не важко виправити, тому вони не спричиняли великого дискомфорту під час розробки, та зазвичай швидко вирішувалися.

3.2. Тестування проекту

Як приклад (рис 3.1) – при наявності графічних артефактів при відображенні зображень через невірне позиціонування, тим самим вони можуть накладатися одне на одне, що не є запланованим рішенням і це можна вирішити лише знайшовши хибний фрагмент коду та виправивши його [19].



Рис. 3.1. Невідповідність позиціонування на ранньому етапі розробки.

Перший тип тестування – тест графічного інтерфейсу в ручному режимі. Він розуміється як різностороння взаємодія з елементами інтерфейсу. Як приклад – при натисканні на кнопку з посиланням «Продукти» ми бачимо, що дана кнопка відправляє нас за адресою: D:/Project/FruitMarket/Productions.html, вказану в коді (рис. 3.2).



Рис.3.2. Тест кнопок header_block'у. Продукти.

При наведенні курсору на зображення в таблиці товарів – спрацьовує псевдоклас, який виконує функцію масштабування, збільшуючи картинку. На початку розробки даного функціоналу було сплановано додати атрибут `transition-duration:0.5s` для плавності збільшення, але згодом дана ідея була відхилена. В фінальній версії зображення одразу ж збільшується при наведенні курсору на товар «Апельсини» (рис 3.3).

Продукти	Корзина	Кабінет
<p>Апельсини</p>  <p>Ціна - 15 грн/кг</p>	<p>Мандарини</p>  <p>Ціна - 17.55 грн/кг</p>	<p>Ананаси</p>  <p>Ціна - 63.20 грн/кг</p>
<p>Манго</p>  <p>Ціна - 72 грн/кг</p>	<p>Помело</p>  <p>Ціна - 45 грн/кг</p>	<p>Банани</p>  <p>Ціна - 34.80 грн/кг</p>
<p>Папайя</p>  <p>Ціна - 93 грн/кг</p>	<p>Кокоси</p>  <p>Ціна - 70 грн/кг</p>	<p>Ківі</p>  <p>Ціна - 40 грн/кг</p>
<p>Малина</p>  <p>Ціна - 18 грн/кг</p>	<p>Груші</p>  <p>Ціна - 10 грн/кг</p>	<p>Яблуко</p>  <p>Ціна - 8 грн/кг</p>

Рис. 3.3. Масштабування зображень товарів.

По натисканню на кнопку «Додати товар в корзину» – у шапці сайту під кнопкою корзини з'являється символ зірки для привернення уваги до оновлення даних в корзині (рис 3.4).

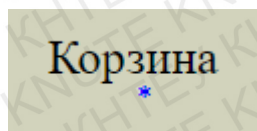


Рис. 3.4. Оновлення товару в корзині.

Перевірка на справність роботи пошукової строки. Для тестування було введено одне зі слів-хештегів внесених до бази даних, яке викликає вертикальний список товарів, підлягаючих під дану категорію.

Можна назвати проблемною розробку пошукової строки, яка знаходить через БД інформацію та відображає її. Даний баг був спричинений через невірний підхід до даного рішення і в процесі був замінений іншою функцією та алгоритмом пошуку, після чого проблема була вирішена.

Довгий час було неможливо створити функцію, яка б додавала товари до корзини та вірно відображала кількість обраною екземплярів, але з часом після пошуків інформації по вирішенню даної проблеми в мережі було знайдений варіант, який частково виправляв дану помилку. При цьому далі зустрічались проблеми із нестабільністю роботи даної функції, після чого було прийняте рішення повністю замінити її на більш просту, але стабільно виконуюючи свою функцію обчислення кількості товарів.

3.3. Порівняння з аналогами

Наразі в мережі існує велика кількість Інтернет ресурсів, та майже не існує таких, які б були повністю оригінальними, та не мали аналогів [11]. Кожен проект націлений зацікавити клієнта, тому має мати особливості, які б дали перевагу перед конкурентами. Для цього проведено порівняння та аналіз проекту який розробляється, та сторонніх Інтернет магазинів [12]. Для порівняння обрані 4 веб-сайти:

- Napoli;
- Novus;
- Gastronom;
- Produktoff.

Як зазначено на сайті (рис.1.1), магазин «Napoli» – це:

- продукти та побутова хімія із Європи;
- алкоголь (лише офіційна продукція, наявність ліцензії);
- оригінальна парфюмерія і косметика по низьких цінах;
- товари для будинку та багато іншого в наявності магазину;
- а також більше 10000 товарів в асортименті;
- мінімальні строки доставки по всій Україні;
- бонуси від 2% до 6% постійним клієнтам;
- європейська якість.

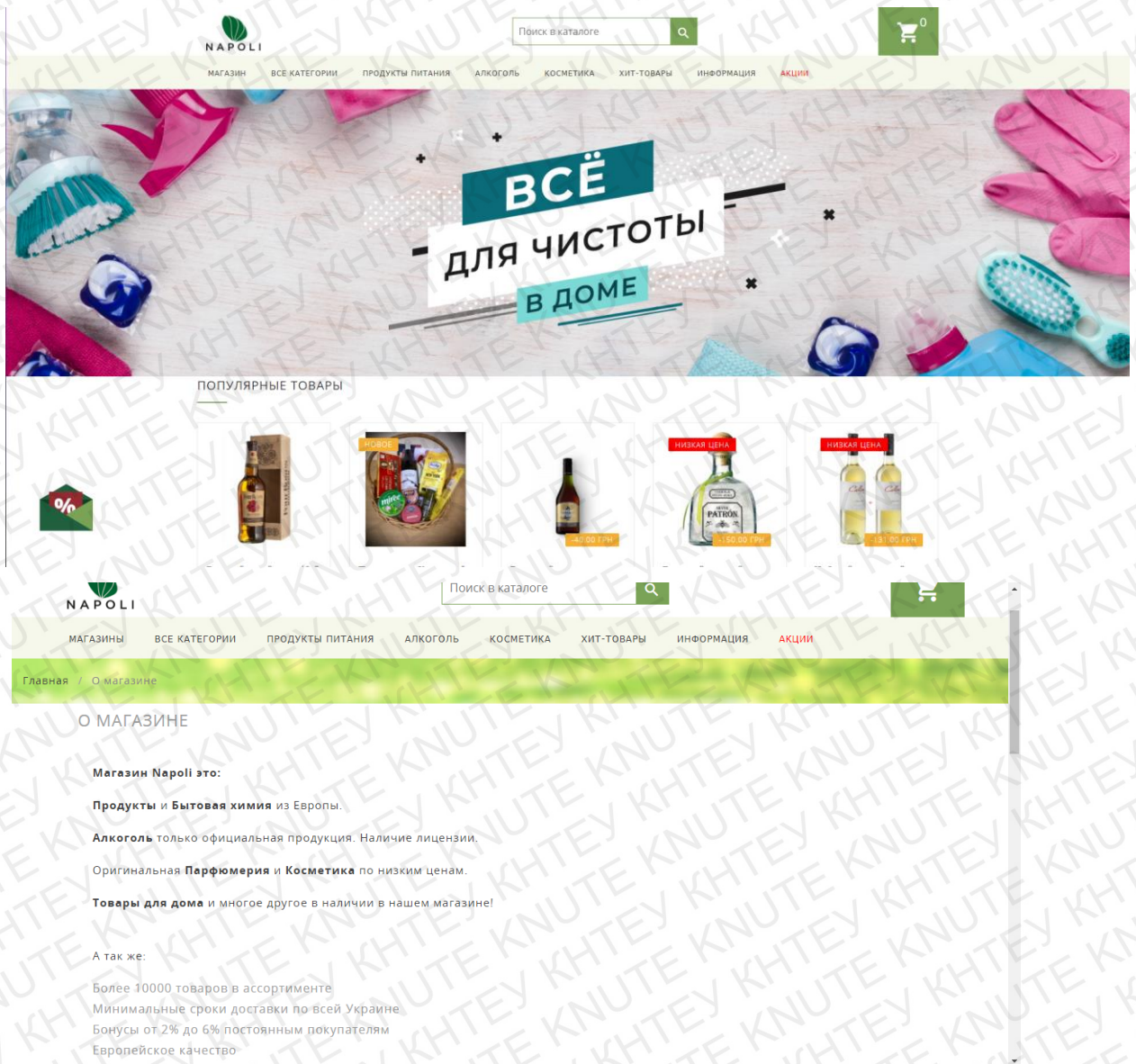


Рис.1.1. Описание магазину «Napoli».

Магазин «Novus» пропонує наступні особливості (рис.1.2):

- персональний збірщик;
- свіжі продукти;
- відсутність націнки;
- «холодний» процес;
- вчасна доставка;
- легке повернення.

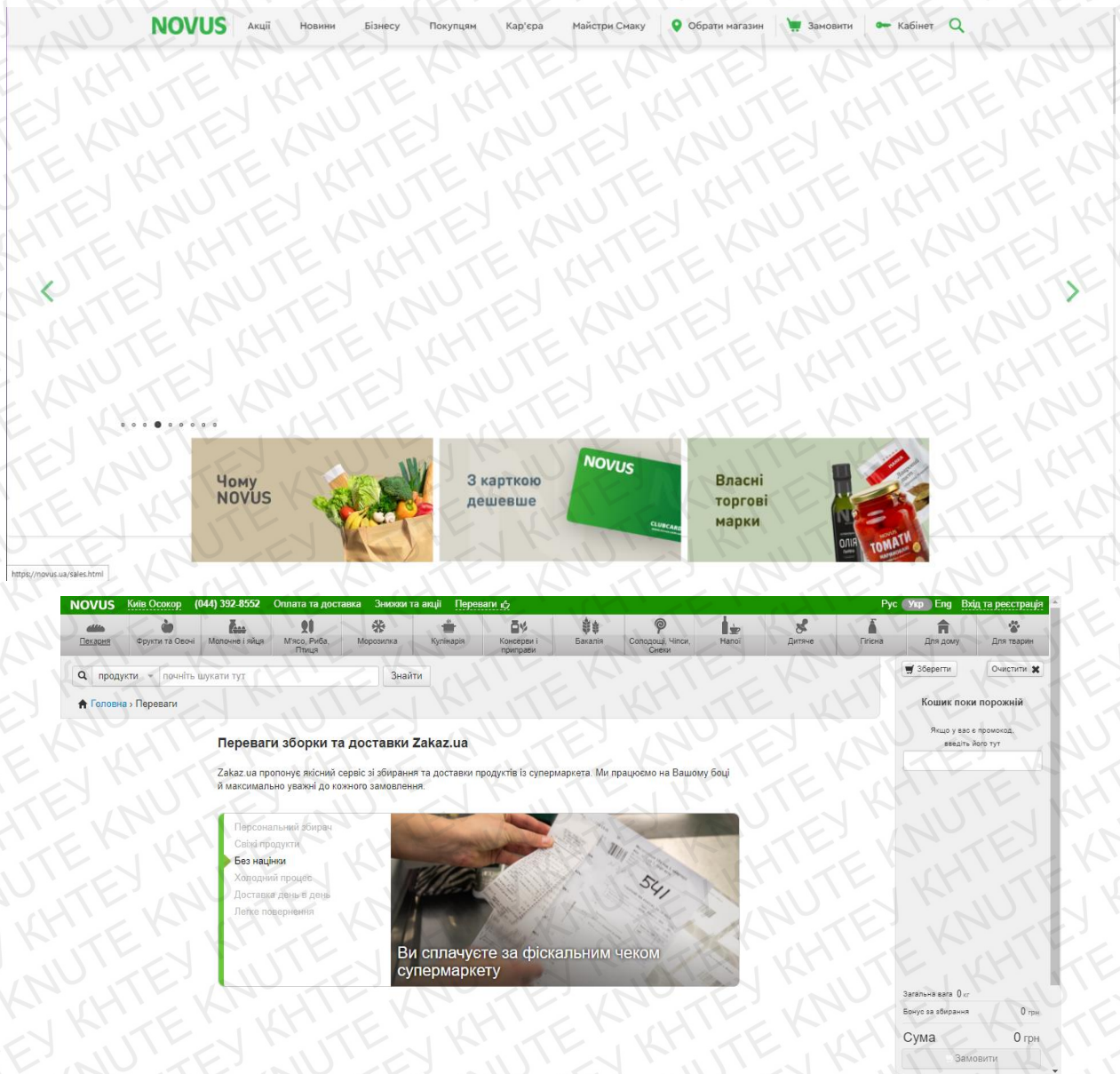


Рис.1.2. Опис магазину «Novus».

Відмінними рисами магазину «Gastronom» (рис.1.3) є:

- надійність;
- економія часу;
- зручність.

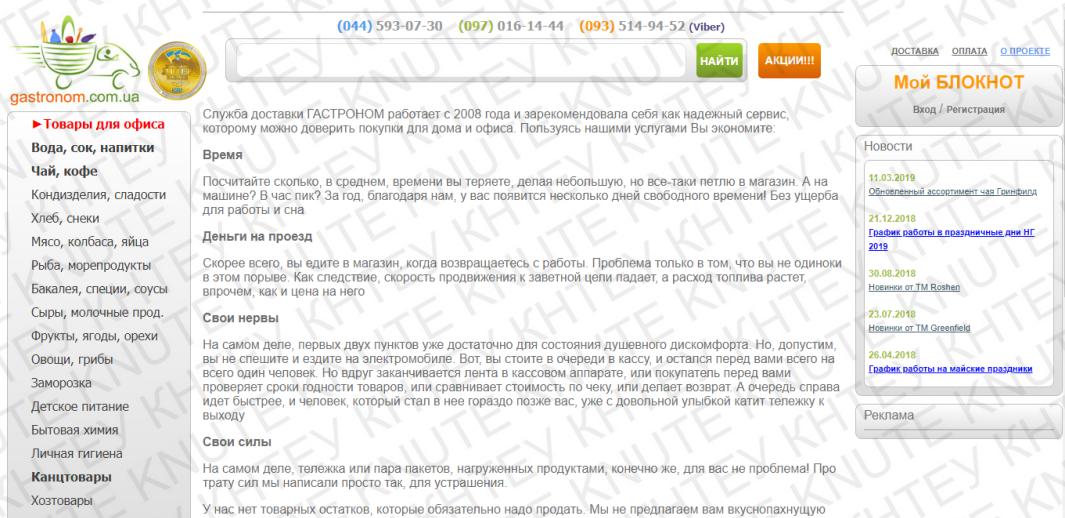
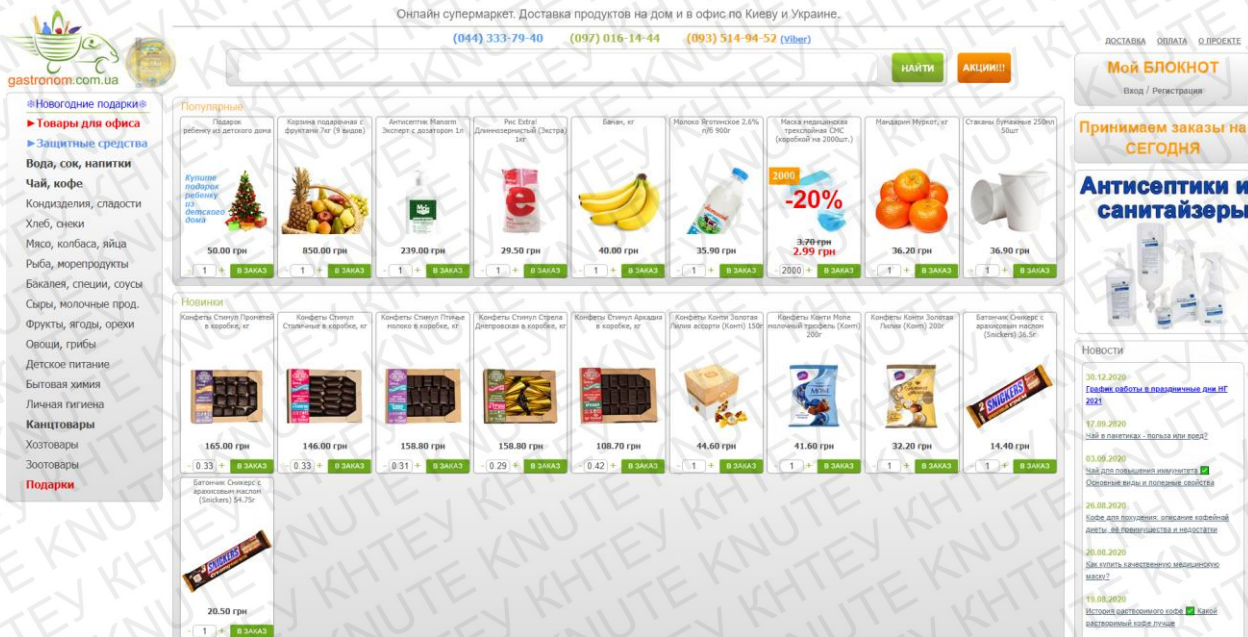


Рис.1.3. Опис магазину «Gastronom».

Порівняння сайтів у технічному плані:

– «Naroli» пропонує користувачеві меню на головній сторінці, що складається з розділів. Дане меню виконане в вигляді горизонтального списку (рис. 1.4).



Рис.1.4. Головна сторінка «Napoli».

При наведенні курсору на пункти меню – розкривається більш детальний перелік класифікацій товарів (рис.1.6).

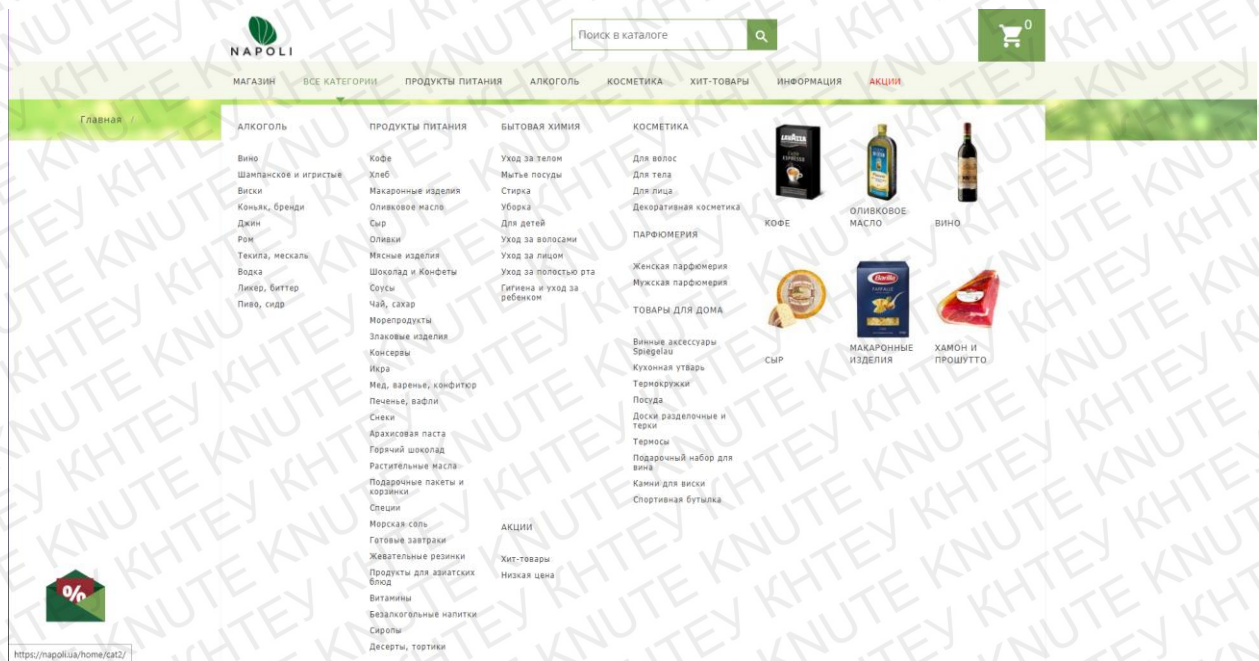


Рис.1.5. Селектори розділів товарів «Napoli».

Також присутній логотип, пошукова строка, кнопка входу до персонального кабінету та корзина. Окрім цього в кінці сайту знаходиться його опис, кнопка повернення до початку сторінки, контактна інформація та посилання (рис.1.6).



Рис.1.6. Кінець головної сторінки «Naroli».

– «Novus» складається з:

- шапки сайту (кнопка з посиланням на головну сторінку, селектор вибору регіону, контактний номер, та кнопки з іншими посиланнями на розділи магазину) (рис.1.7);



Рис.1.7. Шапка сайту «Novus».

- меню з нерозгортуючими розділами продукції («Пекарня», «Фрукти і овочі», «Молочні та яйця» і т.п.) (рис.1.8);

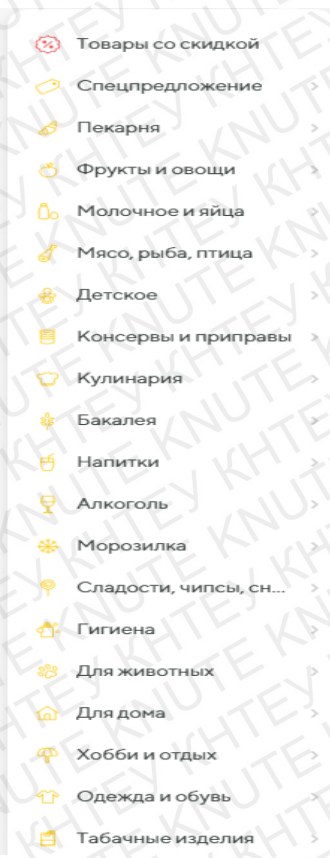


Рис.1.8. Меню вибору розділів товарів «Novus».

- поля пошуку з фільтром (рис.1.9.);



Рис.1.9. Пошукова строка «Novus».

- корзина з детальною інформацією доданого в неї товару (ціна, вага).
Окрім цього є поле для вводу промо-коду на знижки та клавіші очищення і збереження списку товарів (рис.1.10);

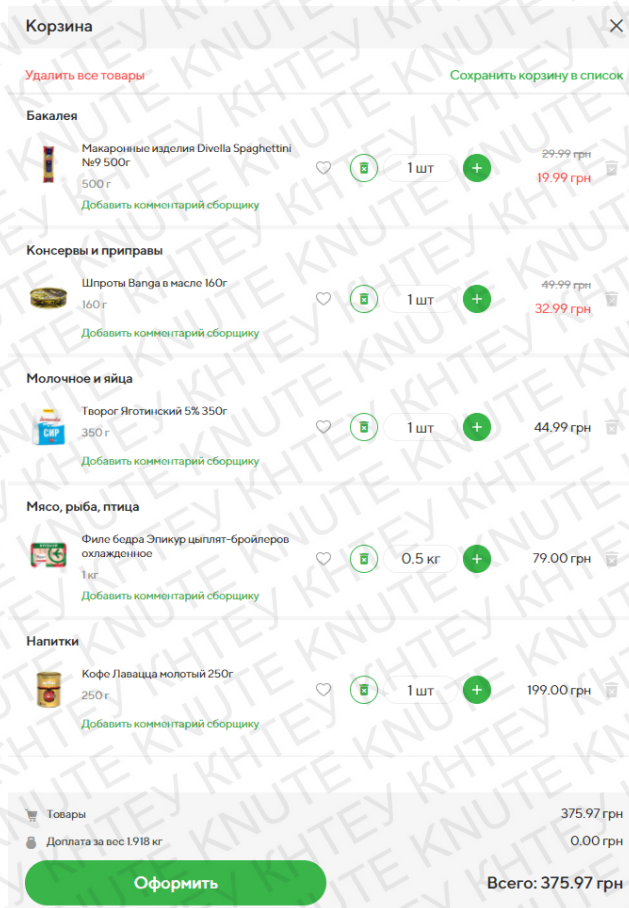


Рис.1.10. Корзина «Novus».

- блок з можливістю вибору послуг (замовлення товару на обраний час, збірка продуктів в торговому залі, доставка заморожених продуктів) (рис.1.11);

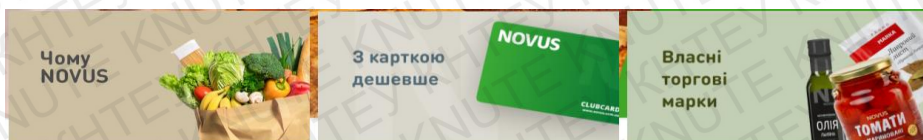


Рис.1.11. Блок зі спец. послугами «Novus».

- блок з кнопками популярних розділів («Самі популярні», «Придбайте по рецептах» та «Акційні пропозиції») (рис.1.12);



Рис.1.12. Блок з популярними розділами «Novus».

- банер-блок, по натисканню на якому можливо перейти на відповідну сторінку (акційна сторінка «До відпочинку готовий», «До відпочинку готовий», «Весь текстиль» та ін.) (рис.1.13);



Рис.1.13. Банер-блок «Novus».

- таблиця з інформацією про графік самовивозу (рис.1.14);

График самовывоза
Ближайшая доставка сегодня с 13:00 до 15:00

Сегодня, 03 Марта	Завтра, 04 Марта	Пятница, 05 Марта
10:00 - 12:00 Недоступно	10:00 - 12:00 24,00 грн	10:00 - 12:00 24,00 грн
11:00 - 13:00 Недоступно	11:00 - 13:00 24,00 грн	11:00 - 13:00 24,00 грн
12:00 - 14:00 Недоступно	12:00 - 14:00 24,00 грн	12:00 - 14:00 24,00 грн
13:00 - 15:00 24,00 грн	13:00 - 15:00 24,00 грн	13:00 - 15:00 24,00 грн
14:00 - 16:00 24,00 грн	14:00 - 16:00 24,00 грн	14:00 - 16:00 24,00 грн
15:00 - 17:00 24,00 грн	15:00 - 17:00 24,00 грн	15:00 - 17:00 24,00 грн
16:00 - 18:00 24,00 грн	16:00 - 18:00 24,00 грн	16:00 - 18:00 24,00 грн
17:00 - 19:00 24,00 грн	17:00 - 19:00 24,00 грн	17:00 - 19:00 24,00 грн
18:00 - 20:00 24,00 грн	18:00 - 20:00 24,00 грн	18:00 - 20:00 24,00 грн

Доступно Закрывается Недоступно

Вы можете выбрать время доставки при оформлении заказа
Если вам не подходят доступные окна доставки, попробуйте выбрать другой способ или адрес доставки

Рис.1.14. Таблица графіку самовивозу «Novus».

- акційні товари (рис.1.15);



Рис.1.15. Блок акційних товарів «Novus».

- футер-блоку з посиланнями на різноманітні розділи сайту («Розрахунок та доставка», «Знижки і акції», група в Facebook) (рис 1.16).



Рис.1.16. Футер-блок «Novus».

- будова сайту «Gastronom»:
- блок з пошуковою строкою (рис.1.17);



Рис.1.17. Пошукова строка «Gastronom»

- зображення з логотипом сайту (рис.1.18);



Рис.1.18. Логотип «Gastronom».

- блок контактних номерів (рис.1.20);

(044) 593-07-30 (097) 016-14-44 (093) 514-94-52 (Viber)

Рис.1.19. Контактні номери «Gastronom».

- меню вибору розділів (рис.1.20);

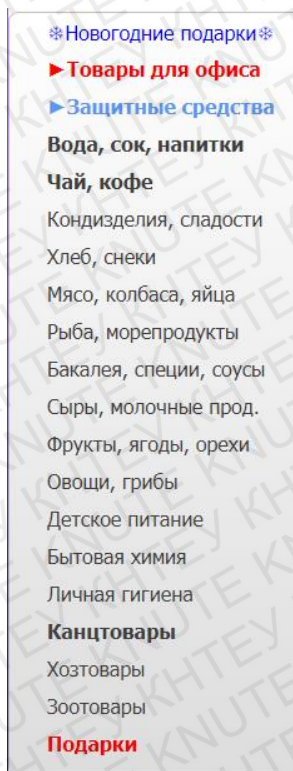


Рис.1.20. Меню выбора продуктов «Gastronom».

- таблица товаров избранного раздела (рис.1.21);

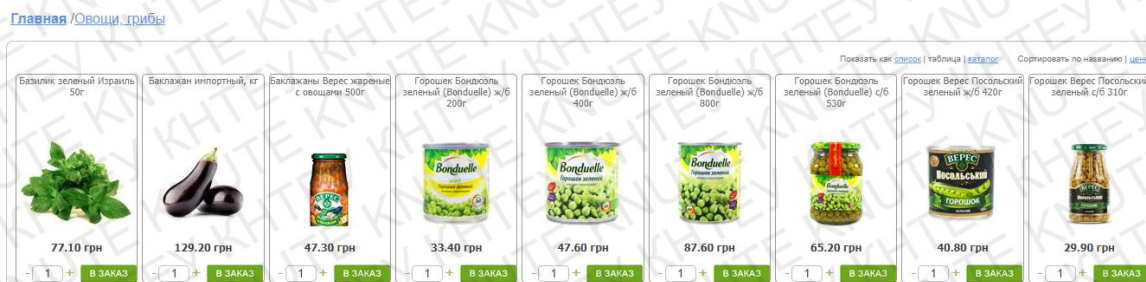


Рис.1.21. Таблица продуктов «Gastronom».

- кнопки входа та реєстрації на сайті (рис.1.22);

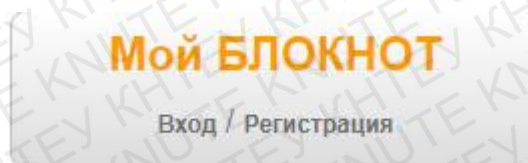


Рис.1.22. Блок входа до особистого кабінету «Gastronom».

- блок новин сайту з посиланнями на сторінки (рис.1.23);

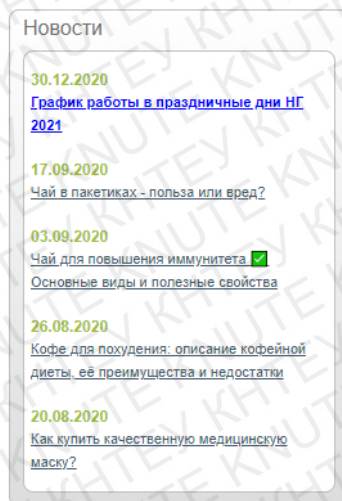


Рис.1.23. Блок новин «Gastronom».

- блок тексту, який описує товар вибраного розділу(рис.1.24);

[Главная](#) / [Условия доставки](#) /

Заказ по телефону можно сделать с понедельника по пятницу с 9:00 до 18:00.

На сайте заказы принимаются круглосуточно.

Условия доставки:

АКЦИЯ!!! До 31.08 снижена стоимость доставки!!!

По Киеву:	По Украине:
Осуществляется собственным транспортом с понедельника по пятницу в удобное для Вас время.	Осуществляется через Новую почту.
Мы доставляем БЕСПЛАТНО заказы на сумму от 1500 грн.	Отправка заказа производится после 100% предоплаты.
При заказе от 1000 до 1500 грн. - стоимость доставки 40 грн. 0 грн.	Услуги Новой почты оплачивает Заказчик или Получатель.
При заказе от 500 до 1000 грн. - стоимость доставки 70 грн. 0 грн.	Если Ваш заказ менее 1500 грн. в него автоматически добавится сумма "Доставки". Эта сумма будет убрана оператором.
При заказе до 500 гривен - дополнительно оплачивается 400 грн. 70 грн.	По Украине отправляем только те виды продуктов питания, которые не требуют особых условий хранения и транспортировки. Например, чай, кофе, соусы и т.д.
Никаких иных доплат: "за пакеты", "за вес", "за сборку", "за подъем" - у нас НЕТ!	
Возможное время выполнения заказа от 3-х часов с момента его подтверждения .	

Рис.1.24. Інформація по обраному розділу «Gastronom».

З мінусів Інтернет-магазину «Gastronom» можна виділити надмірно помітні надписи. Хоча вони привертають увагу до певних елементів тексту чи об'єктів інтерфейсу, вони також виглядають не зовсім естетично та неприємно для людського зору. Окрім цього опис товарів також має візуальні ляпи, такі як недостача контрастності з фоном та перевантаженість інформації на одну сторінку. Загалом вибір візуального оформлення даного сайту являється на теперішній момент морально застарілим.

Даний сайт, як можна помітити – схожий і за своєю будовою на «Gastronom», а також «Napoli»:

Як можна помітити – більшість елементів цих сайтів є в деякій мірі схожими. Тому – для того, щоб зацікавити клієнта перед розробниками ставиться ціль – розробити проект таким чином, щоб він мав якомога більше переваг перед конкурентами. В даному порівнянні позитивно виражені сайти «Naroli» та «Novus» у візуальному та технічному плані.

В порівнянні із проектом, який розроблений – сайти-лідери порівняння в деяких аспектах виглядають більш естетично. Як приклад – таблиця товарів сайту «Фруті маркет» (рис.1.25) виглядає не так привабливо як у «Naroli» (рис.1.26).



Рис.1.25. Таблиця товарів «Фруті маркет».

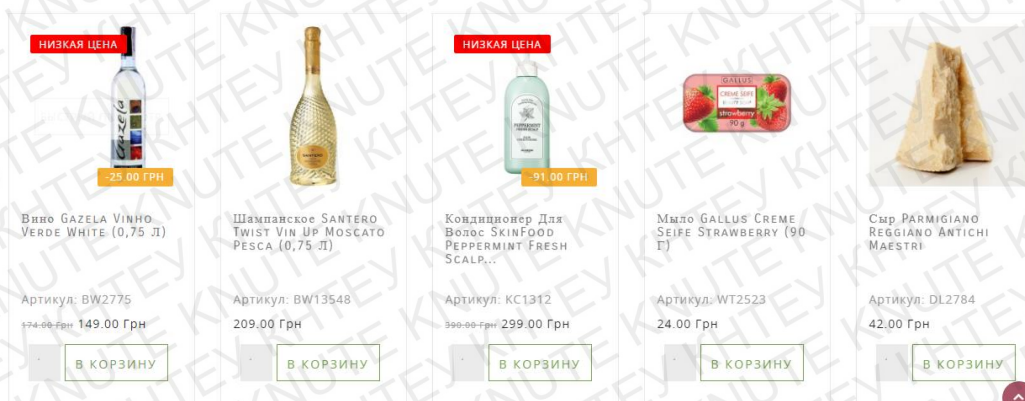


Рис.1.26. Таблиця товарів «Naroli».

При цьому – шапка з меню розроблюваного проекту «Фруті маркет» (рис.1.27) виглядає однаково лаконічно в порівнянні з «Naroli» (рис.1.28);

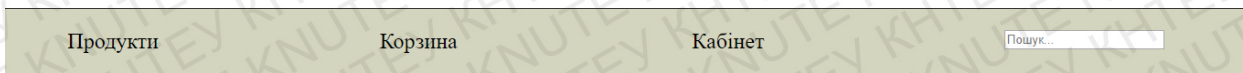


Рис.1.27. Шапка «Фруті маркет».

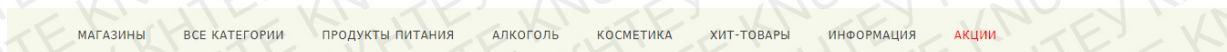


Рис.1.28. Шапка «Naroli».

Як можна побачити з даного дослідження – конструкція сайту «Фруті маркет» має досить сучасні рішення, що являється однією із переваг. Це дає змогу конкурувати з іншими проектами.

3.4 Модуль логістики

Транспортно-логістична діяльність при онлайн торгівлі забезпечує динамізм товаропотоків в процесі діяльності інтернет магазину. Важливість транспортної складової у виробничо-маркетинговій діяльності інтернет магазинів переоцінити неможливо, адже реалізація будь-якої продукції потребує переміщення товарів в просторі. З огляду на вищезазначене, управління транспортною логістикою потребує сучасних підходів при реалізації задля зменшення питомої ваги транспортних витрат в структурі собівартості продукції, забезпечення комфортних умов отримання товару покупцем та розширення географії збуту для продавців.

При торгівлі, що здійснюється за допомогою інтернет сайту необхідно переміщати на різні відстані значну кількість різноманітних товарів, що вимагає високої організації роботи різних каналів та способів реалізації доставки до клієнта з мінімальними зусиллями продавця на організацію та управління цими процесами. Відтак, діяльність інтернет магазину неможливо обмежити якимось одним високоефективним видом доставки товарів. В цьому випадку потрібне поєднання декількох способів доставки товарів до клієнта в залежності від конкретних умов та обставин і саме модуль логістики реалізований з врахуванням вищенаведених задач і та умовностей покликаний вирішити цю проблему.

Для реалізації ефективної діяльності інтернет магазину інтерактивний модуль управління логістикою повинен мати такий функціонал:

- Автоматичний розрахунок вартості доставки кур'єром по Києву та поштовими компаніями по інших містах України.
- Перерахунок вартості доставки в залежності від обраного способу оплати з урахуванням всіх комісій.
- Відображення карти вибору пункту самовивозу та поштових відділень різних компаній.
- Приховування окремих способів доставки або установка для них правила по окремих районах міста Києва.
- Відображення орієнтовного часу доставки.
- Підтримка роботи зі стандартними купонами на знижку.
- Використання ваги з карток товарів для розрахунку вартості доставки.
- Використання габаритів з картки товару для розрахунку вартості доставки
- Демонстрація підказок при введенні назви населеного пункту і автовибір регіону для зручності використання і точності розрахунку.
- Передача замовлень разом з описом номенклатури товарів і каналів доставки в кабінет інтернет магазину.
- Передача замовлень з післяплатою в т.ч. з оплатою по карті або без нього.
- Автоматична і пакетна передача замовлень.
- Відображення трек-номера і внутрішніх статусів замовлення агрегатора інтернет магазину.
- Відображення трек-номера і статусів замовлення транспортної компанії.
- Можливість установки правил по кожному способу доставки:
 - Безкоштовна доставка;
 - Фіксована ціна;

- Фіксована ціна в залежності від ваги;
- Мінімальний і максимальний вагу кошика для доставки;
- Мінімальна і максимальна сума кошика для доставки та інше;

Реалізацію логістичного модуля виконано шляхом створення додаткового інтерфейсу для зручного оформлення доставки замовлених товарів та набору алгоритмів, що забезпечують накопичення, обробку та відображення накопиченої інформації, щодо здійснених замовлень в тому вигляді, який дозволить оператору інтернет магазину з мінімальними зусиллями ефективно організовувати та управляти процесами відвантаження товарів споживачам.

Оформлення замовлення на покупку товару та доставку відбувається в веб інтерфейсі інтернет-магазину(рис.1.29);

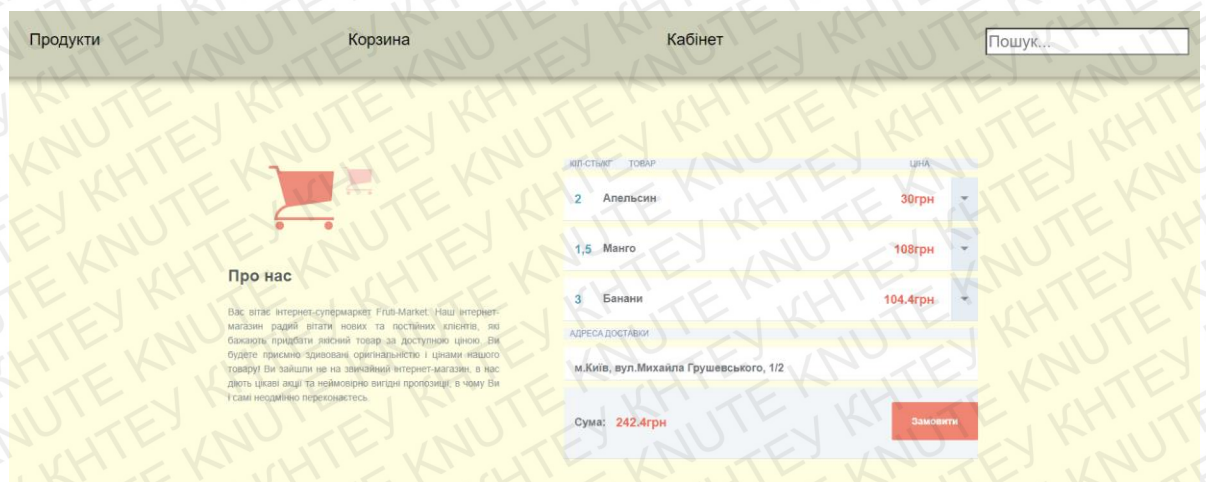


Рис.1.29. Оформлення Замовлення

Оформлення Замовлення відбувається наступним чином

- Покупець обирає спосіб оплати підтверджує замовлення та в разі необхідності перенаправляється на сайт сервісу з онлайн оплати.
- Логістичний модуль зберігає всю інформацію щодо виконаного замовлення в базі даних.
- Логістичний модуль відображає замовлення в Кабінеті Продавця та агрегує однотипні замовлення в розрізі каналів доставки чи логістичних операторів.

- Логістичний модуль відображає виконане замовлення з усіма деталями (кількість, характеристики, деталі способу доставки та відображає орієнтовний час доставки замовлення).

Результатом роботи логістичного модуля при функціонуванні інтернет магазину є скорочення витрат робочого часу працівників через автоматизацію процесів збору та обробки даних, щодо виконаних замовлень та унеможливлення здійснення помилок менеджером інтернет магазину, яких з великою долею ймовірності не вдалося б уникнути при виконанні аналогічних функцій людиною в ручному режимі. Відтак робота логістичного модуля дозволяє уникати додаткових витрат та збоїв в роботі інтернет магазину.

ВИСНОВКИ

В результаті проведеної роботи, було досліджено основи роботи MySQL баз даних і мов програмування, таких як: HTML, CSS, JS – орієнтованих на веб-розробку, а також, як з ними працювати. А також було розроблену базу даних і сайт, за допомогою якого користувач може додавати і переглядати записи в базі даних.

В результаті отриманий досвід створення клієнтоорієнтованого веб-сайту, що теоритично надає перевагу над деякими із альтернативних проєктів.

В ході розробки було поліпшено та розширено знання і вміння по роботі з HTML сторінками. Окрім цього було вивчено альтернативні підходи для вирішення задач за допомогою CSS.

Проаналізував, що PHP і MySQL насправді дуже потужні середовища програмування і зберігання даних і вони дуже швидко працюють та займають дуже мало місця на жорсткому диску [14].

Досліджено, що за допомогою CSS можна отримати продукт задовільної якості на нинішній час, так як це досить потужна та проста мова програмування, яка дає змогу верстати сайт.

Було освоєно середовище розробки Sublime Text 3, яке дозволяє створювати код на: HTML, CSS, JavaScript та PHP. Це середовище виявилось найзручнішим рішенням зі всіх альтернативних, тому що в наявності є весь необхідний функціонал та зручний графічний інтерфейс, що прискорює практичну частину розробки.

Здобуті навички тестування веб-сайтів та аналізу наявних помилок, їх вирішення та обходу в подальшому процесі розробки дали змогу точніше зрозуміти деякі аспекти цієї задачі, хоча це досить кропітливий та відповідальний процес.

Проаналізовано, що у зв'язку з становленням ринкових відносин у нашій країні з'явився і став активно розвиватися новий науково-практичний напрямок - логістика.

Ефективна діяльність логістичної системи покликана задовольнити потреби споживачів у найкращий спосіб, але й водночас мінімізувати витрати будь-якого підприємства.

Досліджено, що звичайний інтернет-магазин, який працює в Україні виключно для українських споживачів, можна розглядати як логістичну систему мікроекономічного рівня, що складається з закупівельної, збутової, маркетингової, складської, інформаційної, фінансової, сервісної, транспортної підсистем та підсистеми управління запасами.

Одним із шляхів оптимізації організації логістичної діяльності Інтернет-магазину є вдосконалення управління асортиментом. Дане завдання можна вирішити за допомогою ABC-аналізу. Крім того для вдосконалення логістичної діяльності пропонується створити нові склади та представництва

В сукупності всі вищевказані пункти дають зрозуміти, що при використанні мов програмування та баз даних у зв'язці – можна отримати якісний кінцевий продукт.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. [Електронний ресурс] – Режим доступу до ресурсу:
<http://www.codenet.ru/db/mysql/mysql.php>
2. [Електронний ресурс] – Режим доступу до ресурсу:
<http://www.webmasterwiki.ru/MySQL>
3. [Електронний ресурс] – Режим доступу до ресурсу:
<http://www.php.su/books/>
4. [Електронний ресурс] – Режим доступу до ресурсу:
<http://www.php.ru>
5. [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.php.net/manual/ru/langref.php>
6. [Електронний ресурс] – Режим доступу до ресурсу:
<http://htmlbook.ru/css>
7. [Електронний ресурс] – Режим доступу до ресурсу:
<http://rbs-webmarket.com/making-site/site-development/concept-of-sites.html>
8. Маклафлин - PHP и MySQL. Исчерпывающее руководство.
9. Руслан Раянов – Как создать продающий сайт с нуля. Краткое руководство для предпринимателя
10. [Електронний ресурс] – Режим доступу до ресурсу:
<https://neoseo.com.ua/sozдание-internet-magazina-s-nulya>
11. [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.openbusiness.ru/html/dop6/intermag.htm>
12. [Електронний ресурс] – Режим доступу до ресурсу:
<https://www.cossa.ru/155/48800/>
13. Ларри Ульман – PHP и MySQL: создание интернет-магазинов
14. Мейерсон М. – Основы интернет-маркетинга. Все, что нужно знать, чтобы открыть свой магазин в интернете.
15. [Електронний ресурс] – Режим доступу до ресурсу:

<https://freshdesign.ua/blog/technology/website-testing>

16. [Электронный ресурс] – Режим доступа до ресурсу:

https://artjoker.ua/ru/uslugi/qa_os_testirovanie/

17. [Электронный ресурс] – Режим доступа до ресурсу:

<https://qalabs.com.ua/testirovanie-saitov-osnovnye-etapy.html>

18. [Электронный ресурс] – Режим доступа до ресурсу:

<https://www.kasper.by/blog/testirovanie-saita/>

19. [Электронный ресурс] – Режим доступа до ресурсу:

<https://www.cssmatic.com/box-shadow>

20. [Электронный ресурс] – Режим доступа до ресурсу:

<https://1ps.ru/blog/dirs/2016/oshibki-validaczii-chto-eto-kak-proverit/>

21. [Роберт Калбертсон, Крис Браун, Гэри Кобб “Быстрое тестирование”](#)

22. [Электронный ресурс] – Режим доступа до ресурсу:

http://www.dut.edu.ua/uploads/1_201_33535647.pdf

23. [Электронный ресурс] – Режим доступа до ресурсу:

<https://dspace.nuph.edu.ua/bitstream/123456789/18518/1/МЕТОД%20ПРАКТ%20РОБОТА%203%20ВИРОБ%20ЛОГ%20новая.pdf>

24. [Электронный ресурс] – Режим доступа до ресурсу:

https://web-creator.ru/articles/about_frameworks

25. [Электронный ресурс] – Режим доступа до ресурсу:

<http://iwsn.ru/blog/show/veb-freymvorki-i-s-chem-ih-edyat.>