

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук та технологій
(повне найменування факультету)

Кафедра комп'ютерних систем та мереж
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавра

(ступінь вищої освіти (освітній ступінь))

на тему РОЗРОБКА ВЕБСЕРВІСУ «FASTPURCHASE»

Виконав: студент 4 курсу, групи КНТз-510ін
спеціальності _____

123 «Комп'ютерна інженерія»

(код і найменування спеціальності)

Освітня програма (спеціалізація) _____

«Комп'ютерна інженерія»

ЗУЄВА К. О.

(ПРИЗВИЩЕ та ініціали)

Керівник КУДЕРМЕТОВ Р.К.

(ПРИЗВИЩЕ та ініціали)

Рецензент СТЕПАНЕНКО М.В.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування вищого навчального закладу)

Факультет Комп'ютерних наук і технологій
Кафедра «Комп'ютерні системи та мережі»
Ступінь вищої освіти бакалаврський
Спеціальність 123 Комп'ютерна інженерія
(код і найменування)
Освітня програма (спеціалізація) Комп'ютерна інженерія
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮЗавідувач кафедри КУДЕРМЕТОВ Р.К.

“ ” 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

ЗУСВОЇ Карини Олексіївни

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка вебсервісу «FastPurchase»керівник проєкту (роботи) к. т. н., доцент, КУДЕРМЕТОВ Равіль Камілович

(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

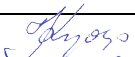
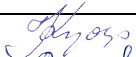
затверджені наказом вищого навчального закладу від “01” квітня 2024 року № 105

2. Строк подання студентом проєкту (роботи) 31 травня 2024 року3. Вихідні дані до проєкту (роботи) опис предметної області, наявні засоби та технології розробки версервісів4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1) Аналіз технічного завдання;2) Моделювання програмного забезпечення;3) Проєктування вебсервісу;4) Розробка вебсервісу.

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

Креслення 1 – Алгоритм оформлення замовлення;Плакат 1 – Схема бази даних;Плакат 2 – Прототипи сторінок вебсервісу;Плакат 3 – Сторінки вебсервісу.

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4	КУДЕРМЕТОВ Р. К., к.т.н., доцент		
нормоконтроль	ПОЛЬСЬКА О.В., ст. викл.		

7. Дата видачі завдання 01.03.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз предметної області та існуючих аналогів, виділення основних користувачів, їх можливості та обов'язки	01.03.2024 р.	
2	Визначення вимоги до вебсервісу	10.03.2024 р.	
3	Моделювання прецедентів вебсервісу, створення діаграм послідовності прецедентів	20.03.2024 р.	
4	Проектування бази даних, створення схеми бази даних, вибір СКБД	01.04.2024 р.	
5	Вибір архітектури вебсервісу	10.04.2024 р.	
6	Проектування інтерфейсу користувача та вибір програмного забезпечення	15.04.2024 р.	
7	Розробка бази даних	20.04.2024 р.	
8	Розробка інтерфейса користувача	28.04.2024 р.	
9	Тестування системи	10.05.2024 р.	
10	Оформлення отриманих результатів у ПЗ	15.05.2024 р.	
11	Оформлення графічного матеріалу	25.05.2024 р.	
12	Проходження нормоконтролю, перевірки на плагіат	31.05.2024 р.	
	рецензування		

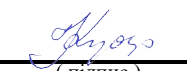
Студент (ка)


 (підпис)

Карина ЗУСВА

(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)


 (підпис)

Равіль КУДЕРМЕТОВ

(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

ПЗ: 66 сторінок, 34 рисунків, 2 таблиці, 18 джерел.

АРХІТЕКТУРА КЛІЄНТ-СЕРВЕР, ВЕБСЕРВІС, ІНТЕРНЕТ-МАГАЗИН, СИСТЕМА КЕРУВАННЯ БАЗАМИ ДАНИХ, СИСТЕМА КЕРУВАННЯ ВМІСТОМ

Об'єкт розробки – вебсервіс «FastPurchase».

Метою дипломного проєкту є розробка вебсервісу «FastPurchase» для продажу продуктів в мережі інтернет, який надасть користувачам можливість зручно і швидко замовляти продукти харчування не виходячи з дому.

Перший розділ присвячено аналізу технічного завдання. Проводиться детальний огляд предметно області, виділяються основні користувачі, їх можливості та обов'язки. Проводиться аналіз вже існуючих аналогів, створюються вимоги до вебсервісу, що буде розроблятися.

У другому розділі проводиться моделювання прецедентів інтернет-магазину на основі аналізу технічного завдання. Створюється діаграма прецедентів вебсервісу, описуються діаграми послідовності прецедентів.

У третьому розділі описано проєктування системи, що розробляється. Обирається архітектури вебсервісу, проєктується база даних та майбутній інтерфейс користувача. Проводиться аналіз та вибір програмного забезпечення.

Четвертий розділ присвячено розробці системи. Описано етапи розробки бази даних та вебінтерфейсу користувача.

ЗМІСТ

Вступ	6
1 Аналіз технічного завдання	8
1.1 Опис предметної області	8
1.2 Аналіз вимог до вебсервісу	11
2 Моделювання програмного забезпечення	24
2.1 Прецедент «Реєстрація».....	24
2.2 Прецедент «Придбання товарів».....	26
2.3 Прецедент «Повторне замовлення товару» ».....	30
3 Проєктування вебсервісу.....	32
3.1 Архітектура вебсервісу.....	32
3.2 Проєктування бази даних.....	33
3.3 Аналіз та вибір СКБД.....	38
3.4 Проєктування інтерфейсу користувача.....	41
3.5 Аналіз та вибір засобів розробки.....	45
4 Розробка вебсервісу.....	48
4.1 Розробка бази даних	48
4.2 Розробка інтерфейсу користувача.....	53
Висновки	63
Перелік джерел посилення	65

Графічний матеріал:

Креслення 1 - Алгоритм оформлення замовлення;

Плакат 1 – Схема бази даних;

Плакат 2 – Прототипи сторінок вебсервісу;

Плакат 3 – Сторінки вебсервісу.

ВСТУП

Послуги, які надаються за допомогою всесвітньої мережі Інтернет сьогодні змінюють все. Так само, як залізничні перевезення впливають на економіку країни, так і вебсервіси докорінно змінюють правила електронної комерції. Передаючи великий обсяг даних більш ефективно і дешевше, ніж раніше, вони забезпечують зв'язок між додатками, виконуваними в будь-яких віддалених точках земної кулі. В результаті між підприємцями і споживачами встановлюється швидкий, зручний і продуктивний зв'язок.

Сьогодні люди щодня використовують мережу Інтернет для перегляду біржових котирувань, придбання споживчих товарів та читання свіжих новин. І найчастіше цього рівня взаємодії цілком достатньо. Але заснована на передачі тільки текстової інформації мережа не забезпечує належної взаємодії програм, особливо при пересиланні великого обсягу інформації. Потрібно більш дієвий спосіб кооперації додатків, що використовує автоматичне виконання команд, які раніше необхідно було вводити в браузері вручну.

Мережа в її сучасному стані погано підтримує програмно-орієнтовану взаємодію. Приватним особам та компаніям, які займаються комерцією з використанням мережі Інтернет, необхідний спосіб публікації посилань на їх застосування і дані, практично такий же, як при публікації посилань на їх вебсторінки. Інтернет-додатки повинні мати можливість знаходити інші інтернет-додатки, звертатися до них і автоматично взаємодіяти з ними. Забезпечуючи "спілкування" між програмами, вебсервіси розширюють можливості Інтернету. За рахунок підтримки вебсервісів розміщені в різних вузлах Інтернету додатки можуть взаємодіяти безпосередньо, як ніби вони є частиною однієї великої інформаційної системи. Вебсервіси в основному створюються для реалізації послуг в мережі Інтернет, в тому числі і в інтернет-магазинах [1].

За декілька років популярність інтернет-магазинів виросла у десятки разів. Покупки в інтернеті привабливі тим, що відповідає необхідність бігати по місту.

Порівнювати ціни і характеристики товарів тепер стало набагато легше, адже все зібрано в одному місці. Можна легко вибрати саме той товар, який за всіма параметрами відповідає інтересам покупця.

Після замовлення товару залишиться тільки трохи почекати. Незабаром кур'єр сам доставить його додому або ж на робоче місце. Саме це багато і цінують і особливо актуально в розмірах великого міста мільйонника, адже особистого часу у людей все менше і менше. Також деякі майданчики використовують послуги пошти для пересилання товарів. Це теж зручно, якщо таких товарів немає в регіоні то після замовлення можна отримати необхідну річ і за помірну ціну.

Ще однією перевагою по праву вважається низька ціна. У більшості випадків при замовленні онлайн можна пристойно заощадити. Секрет полягає в тому, що для зберігання товару інтернет-магазину необхідно оплачувати лише склад, який може знаходитися за межею великого міста або ж взагалі в провінційних областях, в той час як традиційні магазини змушені розщедрюватися і на оренду торгового приміщення. І це відбивається на ціні не в кращу сторону. До того ж для обслуговування онлайн магазину буде потрібно мінімум персоналу. А це означає, що менше коштів йде на зарплату, страхові виплати та інші поточні витрати [2].

Тому розробка вебсервісу, який надаватиме можливість користувачу придбання продуктів харчування онлайн є досить актуальним завданням.

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Опис предметної області

Даний вебсервіс розроблюється для інтернет-магазину продуктів, для можливості реалізації продажу продуктів.

Інтернет-магазин - місце в інтернеті, де відбувається надання послуги продажу товарів споживачеві (юридичній або фізичній особі), враховуючи доставку. При цьому розміщення споживацької інформації, замовлення товару і угода відбуваються там само, всередині мережі (на сайті інтернет-магазину) [3].

Інтернет-магазин продуктів є частковою автоматизацією роботи супермаркету, або навіть розширенням його функціоналу в мережі Інтернет.

Супермаркет - великий універсам з продажу повного асортименту продуктів харчування і напоїв, а також предметів домашнього господарства (товари для дітей, мило, порошки для прання, засоби для миття посуду, предмети санітарії та гігієни, паперові вироби, книги в паперових обкладинках, кімнатні рослини), товари для домашніх тварин (собачий і котячий корм, предмети гігієни, іграшки і т. п.), автомобільні товари, вітальні листівки, косметика, посуд, ліки (продаються без рецепта), побутова техніка та т. п. [4].

В супермаркеті можна виділити наступні ключові ролі: клієнт (відвідувач), касир, консультант. Для визначення обов'язків та можливостей кожної з ключових ролей приводиться опис їх дій у супермаркеті.

Клієнт (відвідувач) супермаркету: заходячи до магазину, може взяти корзину, для того щоб було зручно набирати товари. Він підходить до прилавків, вибирає товар та може його візуально оцінити, подивитися на ціну, виробника, дату виготовлення, термін придатності. Можлива ситуація, коли клієнт передумав брати товар. Для цього він витягує його з корзини та повертає назад на полицю товарів. Після того як клієнт обрав усі необхідні для нього товари і поклав їх до корзини він йде на касу. Касир сканує усі товари за допомогою штрих-коду і на екрані касового апарату відображається інформація про назву товару, його

кількість, одиниці виміру та ціну. Після того як усі товари було проскановано, касир називає загальну суму покупки і пропонує клієнту обрати спосіб оплати: готівкою або карткою за допомогою терміналу. Якщо за якоюсь причиною клієнту не вистачило грошей, або відмовив платіжний термінал, касир має можливість скасувати даний чек. У випадку успішного проведення грошової операції клієнт отримує необхідні чеки, у тому рахунку й фіскальний чек, який надається спеціальним апаратом.

Консультант, коли до нього підходить відвідувач з питанням, повинен дати на нього відповідь або допомогти у вирішенні проблем вибору, або інших проблем, які є у відвідувача.

Отже для кожної з цих ролей ми можемо визначити можливості або обов'язки. Для клієнта супермаркету це:

- переглянути необхідну інформацію про товар (ціну, дату виготовлення, термін придатності, виробника);
- отримати консультацію про товар від персоналу супермаркету;
- повернути товар до його придбання (мається на увазі на полицю);
- придбати товар;
- можливість різних способів оплати (платіжна картка, готівка);
- можливість додаткових послуг (придбати пакет, поповнити мобільний рахунок);
- отримати необхідні чеки;
- повернути товар, якщо він не належної якості (при наявності фіскального чеку) в рамках діючого законодавства.

Обов'язки касира:

- обслуговування клієнтів;
- надання додаткових послуг (пакет, поповнення мобільного рахунку);
- скасування чеку;
- видача чеків.

Обов'язки консультанта:

- надання консультацій у режимі реального часу;

- допомога у вирішенні проблем, що склалися перед відвідувачем.

На основі опису предметної області були виділені такі задачі для інтернет-магазину:

- зручний інтерфейс користувача;
- наявність зворотнього зв'язку;
- можливість завести користувачеві особистий кабінет, для подальшого введення звітності, або зручного користування сайтом;
- реалізація акцій, знижок, тощо.

Користувачі інтернет-магазину та їх характеристика наведені у таблиці 1.1.

Таблиця 1.1 - Користувачі інтернет-магазину та їх характеристика

Класи користувачів	Характеристика користувачів
Відвідувач інтернет-магазину	Відвідувач інтернет-магазину – це людина, яка заходячи до сайту має можливість переглядати асортимент товарів, акції або робити замовлення товару, але не має можливості створювати список бажаних товарів та отримувати бонуси. Для того, щоб подолати дане обмеження, відвідувач повинен зареєструватися на сайті. Зареєстрований відвідувач переходить до класу користувачів «Клієнт інтернет-магазину».
Клієнт інтернет-магазину	Клієнту інтернет-магазину, у порівнянні з відвідувачем, з кожною покупкою надаються бонуси, які він може використовувати при наступних покупках.
Консультант інтернет-магазину	Консультант інтернет-магазину - людина, яка вільно володіє комп'ютером та надає консультацію відвідувачам сайту, коли вони цього потребують, за допомогою спеціальної форми, що розміщується на сайті. Також може оброблювати замовлення клієнтів.
Адміністратор сайту	Адміністратор інтернет-магазину наповнює сайт текстовою, графічною та іншими видами інформації, тобто робить сайт інтуїтивно зрозумілим для користувача, і підтримує його працездатним. За необхідності адміністратор може редагувати будь-яку інформацію на сайті (товари, категорії товарів).

1.2 Аналіз вимог до вебсервісу

Для аналізу функціональності та інтерфейсу було обрано наступні інтернет-магазини продуктів:

- «Ub.ua - marketplace» [5];
- «Інтернет магазин продуктів з доставкою додому – Produktoff» [6];
- «NOVUS Інтернет магазин» [7].

««Ub.ua - marketplace»» - інтернет-магазин, що дозволяє користувачам робити замовлення продуктів. За допомогою даного інтернет-магазину користувач може:

- переглянути асортимент товару, який можна придбати, його опис та характеристики, відгуки про товар;
- зареєструватися для замовлення товару;
- створити список бажаних товарів у особистому кабінеті;
- переглянути попередні замовлення;
- відправити повідомлення до адміністратору сайту з побажаннями або наріканнями;
- залишити коментар до пакування або доставки товару;
- залишити відгук про товар;
- поставити рейтинг товару.

На головній сторінці інтернет-магазину «Ub.ua - marketplace» розміщуються такі елементи:

- блок навігації по сайту, та кнопки пошуку, реєстрації, корзини (зверху);
- каталог товарів, у вигляді меню;
- спеціальні пропозиції товарів для поточного тижня (по центру);
- інформація про інтернет-магазин (знизу).

Вигляд головної сторінки інтернет-магазину «Ub.ua - marketplace» приведений на рисунку 1.1.

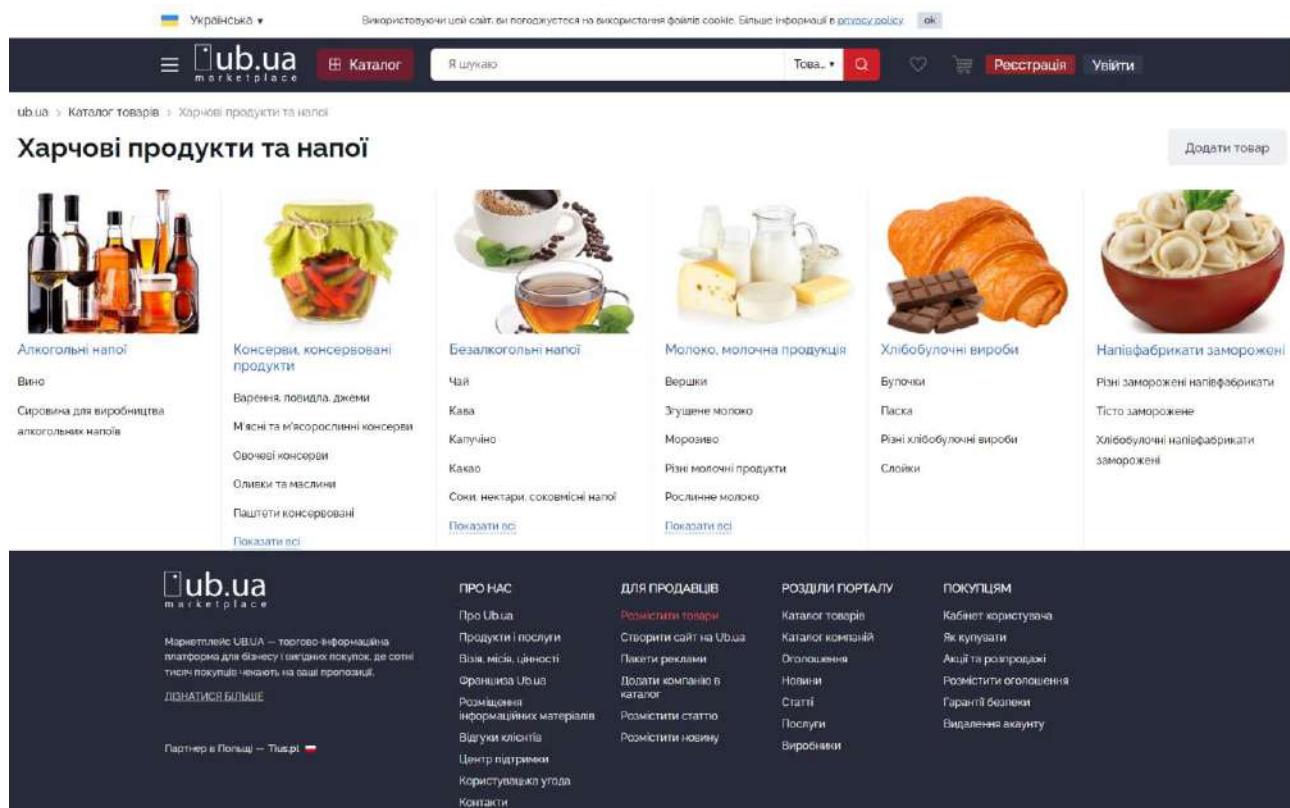


Рисунок 1.1 – Головна сторінка інтернет-магазину «Ub.ua - marketplace»

Для замовлення товарів необхідно спочатку їх вибрати та додати до кошику. Для цього за допомогою пошуку товарів або каталогу товарів потрібно знайти необхідний товар. Вигляд каталогу товарів за категорією «морозиво» приведений на рисунку 1.2.

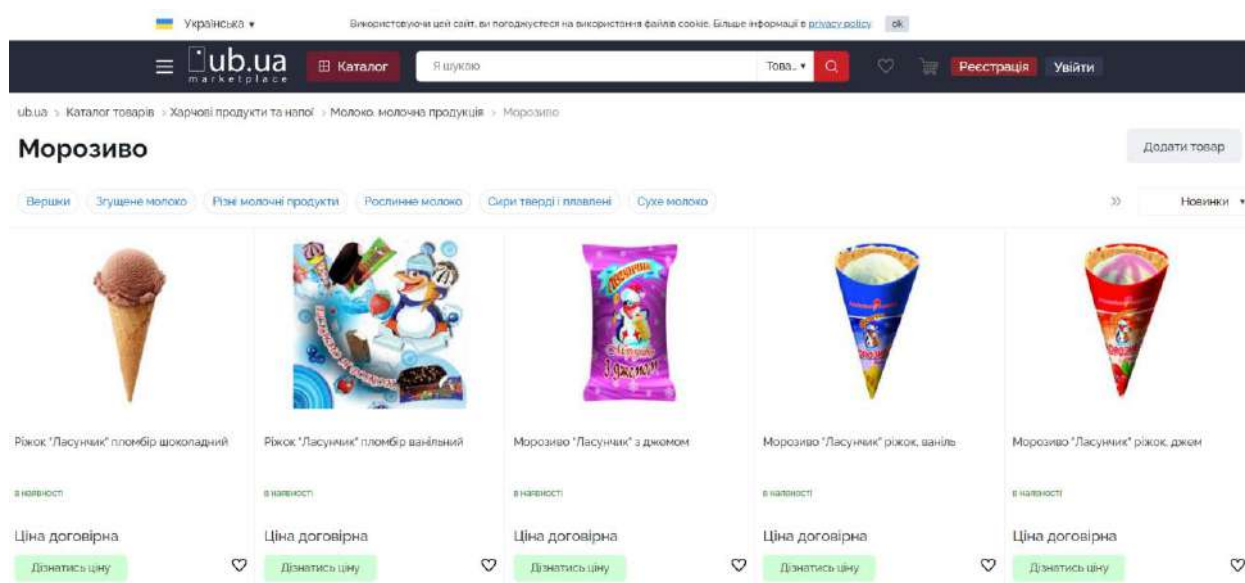


Рисунок 1.2 – Каталог товарів за категорією «морозиво» «Ub.ua - marketplace»

Характеристики товару та відгуки про нього можна натиснувши на картку товару. Розширена картка товару приведена на рисунку 1.3.

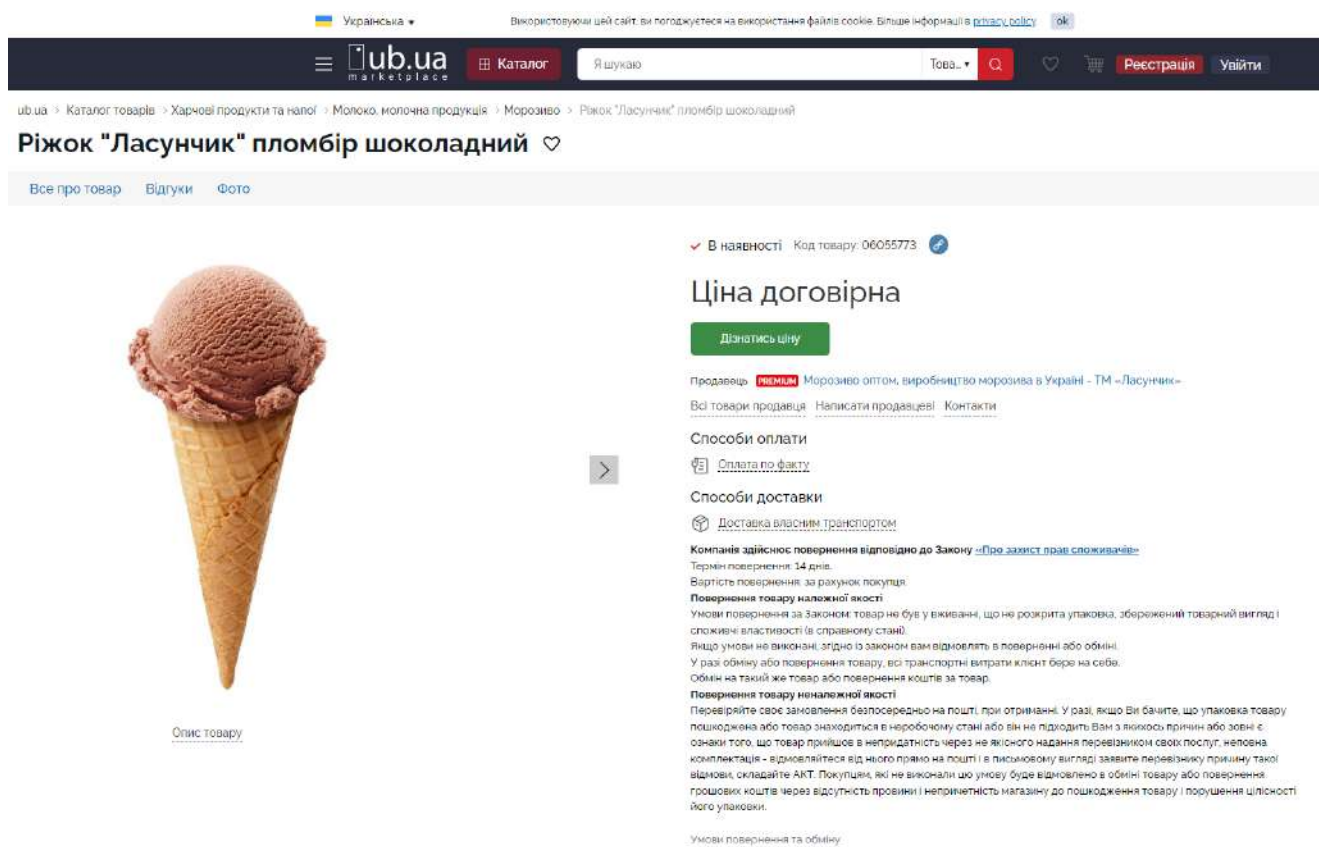


Рисунок 1.3 – Розширена картка товару

Після того як, усі необхідні товари були додані до корзини, потрібно оформити замовлення. Оформлення замовлення відбувається безпосередньо у корзині інтернет-магазину.

Інтернет-магазин продуктів «Produktoff» має майже таку саму структуру головної сторінки, як і у інтернет-магазину «Ub.ua - marketplace», але є деякі доповнення:

- блок з відгуками користувачів, в який будь-який користувач може додати свій відгук у режимі реального часу (ліворуч);
- графік планової доставки (праворуч);
- блок консультації для користувачів (праворуч);

Даний інтернет-магазин має дуже схожу структуру сайту з інтернет-магазином «Ub.ua - marketplace», а значить користувач виконує ті самі дії для формування свого замовлення.

Головна сторінка інтернет-магазину продуктів «Produktoff» наведена на рисунку 1.4.

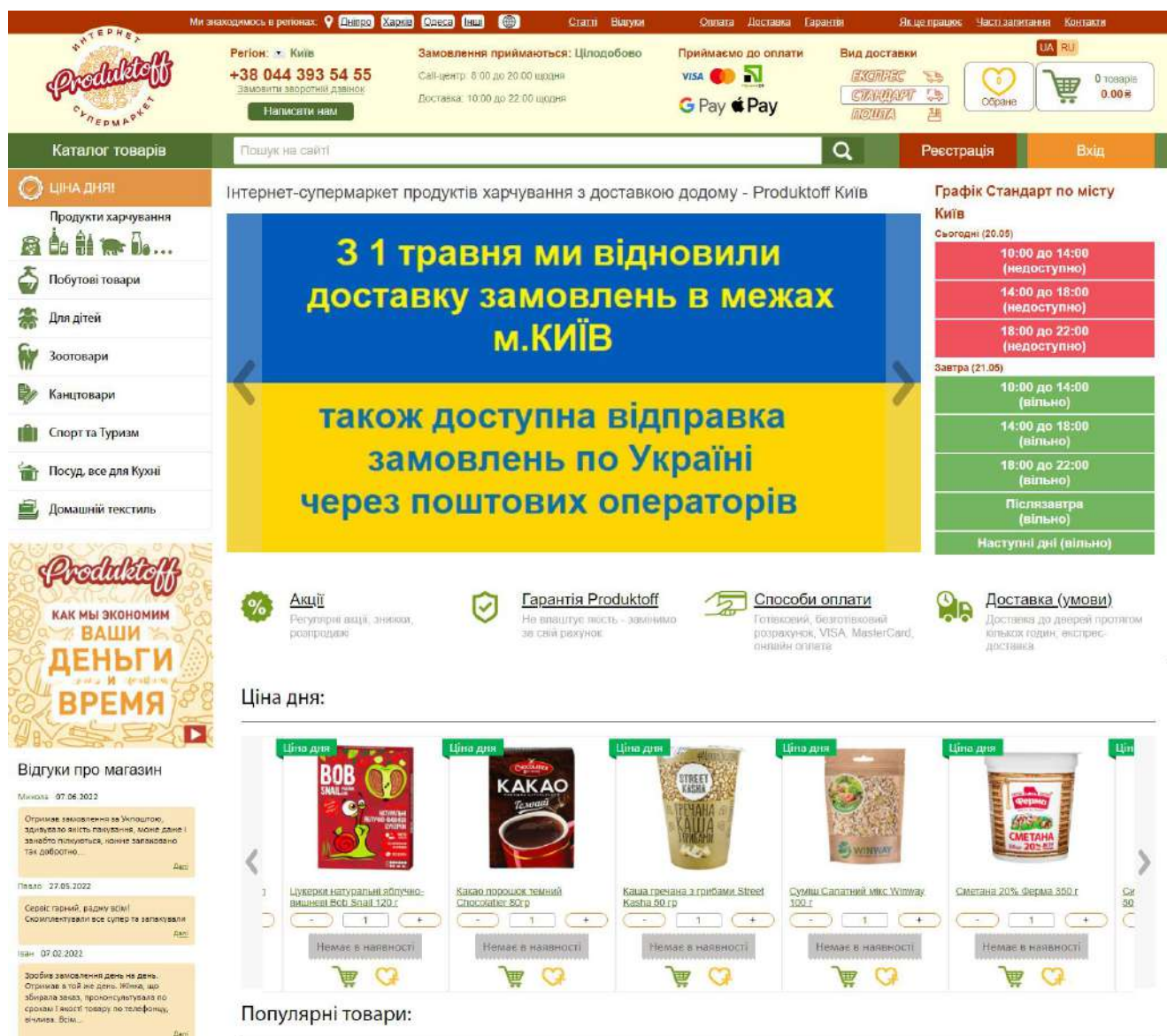


Рисунок 1.4 – Головна сторінка сайту «Produktoff»

Процес формування замовлення аналогічний до алгоритму формування замовлення в інтернет-магазині «Ub.ua - marketplace». Каталог товарів інтернет магазину «Produktoff» наведений на рисунку 1.5.

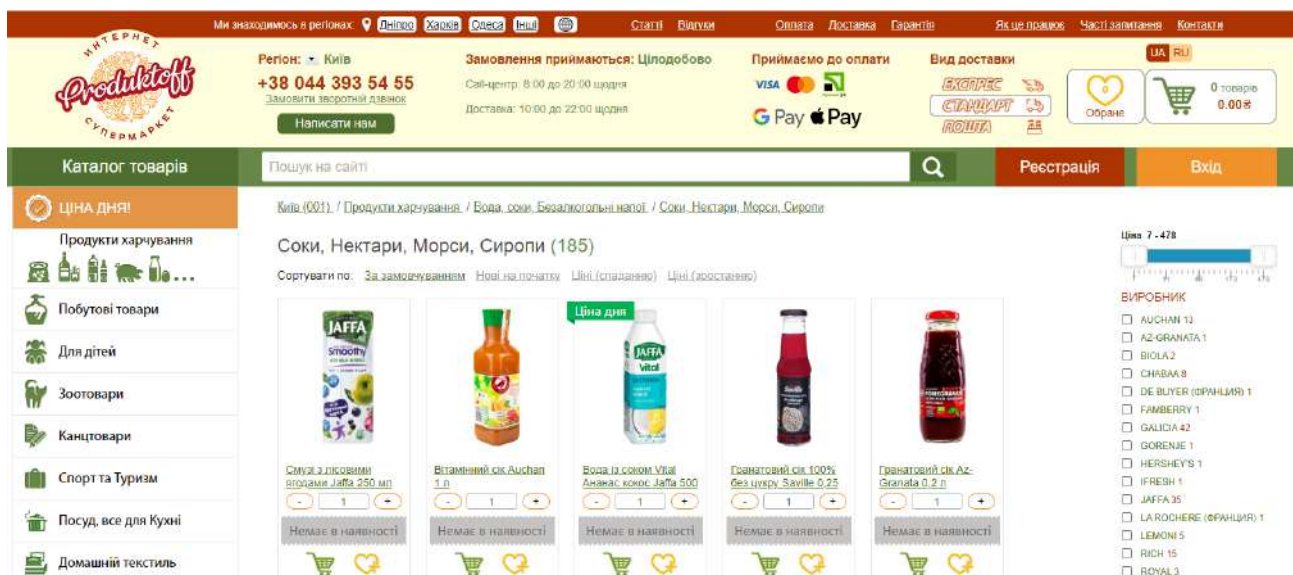


Рисунок 1.5 – Каталог товарів за категорією «Соки, нектари, морси, сиропи» інтернет-магазину «Produktoff»

В розширеній картці товару є деякі зміни у порівнянні з розширеною карткою товарів інтернет-магазину «Ub.ua - marketplace» - немає можливості перегляду та залишення відгуку про товар. Також у порівнянні з інтернет-магазином «Ub.ua - marketplace» немає рейтингу товару. Розширена картка товару інтернет-магазину «Produktoff» наведена на рисунку 1.6.

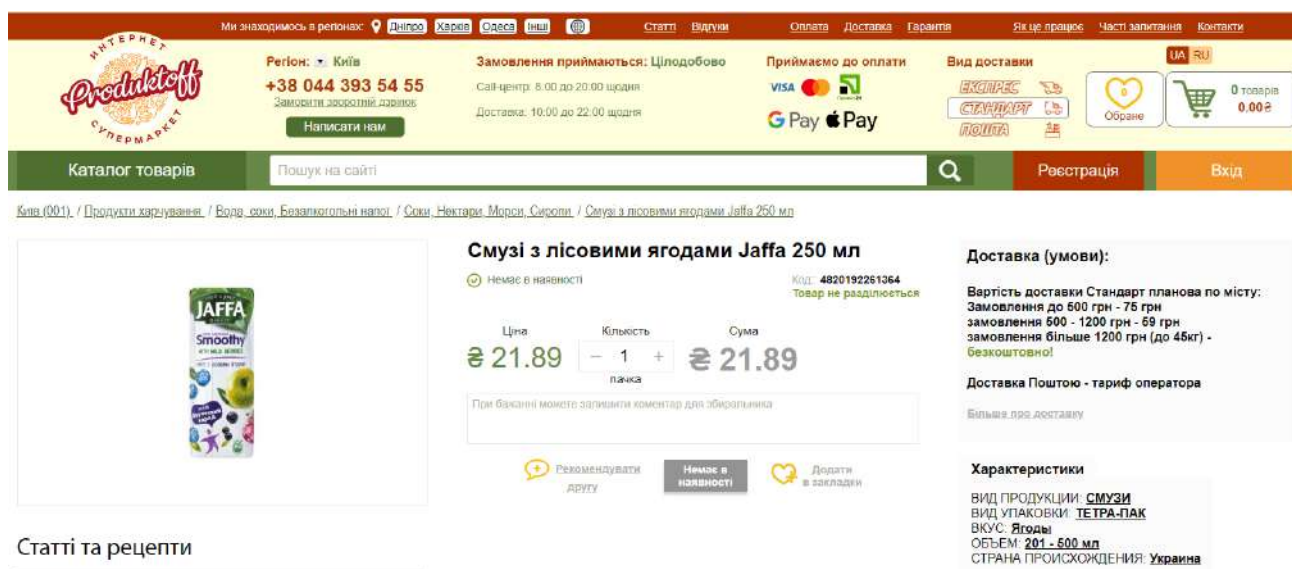


Рисунок 1.6 – Розширена картка товару інтернет-магазину «Produktoff»

Інтернет-магазин «Produktoff» має більший вибір часу доставки, у порівнянні з інтернет-магазином «Ub.ua - marketplace», що дозволяє користувачам розпланувувати час більш ефективніше. Сторінка оформлення замовлення інтернет-магазину «Produktoff» наведена на рисунку 1.7.

Produktoff
Интернет-супермаркет

Регион: Киев
+38 044 331-54-54
Заказы принимаются: Круглосуточно
Сайт: 9:00 до 20:00 ежедневно
Доставка: 1-й до 2-й этаж

Принимаем к оплате: VISA, MasterCard, Apple Pay, Google Pay, PayPal

Варианты доставки: Доставка, Новая точка Delivery, Выбор

2 товара 353,75 ₴

Каталог товаров Поиск по сайту

Оформление заказа

Выберите тип доставки

☒ Плановая ☒ Экспресс доставка ☐ Самовывоз ☐ Доставка по Украине

Доступно Ограничено Недоступно

Выберите время доставки

	Апр. 09 Вторник	Апр. 10 Среда	Апр. 11 Четверг	Апр. 12 Пятница	Апр. 13 Суббота	Апр. 14 Воскресенье	Апр. 15 Понедельник
с 12:00 до 16:00	Доступно	Доступно	Доступно	Доступно	Доступно	Доступно	Доступно
с 16:00 до 20:00	Доступно	Доступно	Доступно	Доступно	Доступно	Доступно	Доступно
с 20:00 до 23:00	Доступно	Доступно	Доступно	Доступно	Доступно	Доступно	Доступно

№ скидочного сертификата

№ скидочного сертификата: 0 Ок

Ваша контактная информация

* - поля, обязательные для заполнения

Имя: * Бородин

Телефон: * +38 (068)580-51-22

Email: b.uzlov76@gmail.com

Оплата: Наличный расчет

Адрес доставки:

Номер подписки: Код подписки: Отслеивать

Доп. информация:

РАСПЕЧАТАТЬ КОРЗИНУ или ОТПРАВИТЬ НА E-MAIL

Согласовывать замену товара со мной, я всегда доступен по телефону

Оформляя заказ Вы соглашаетесь на условия Публичного договора на заказ, приобретение, продажу и доставку товаров с которым можно ознакомиться [здесь](#)

Оформить заказ

Корзина

К оплате без доставки: 353,75 ₴

Объединить с аналогичным 0 Ок

Товар	Единица измерения	Количество	Цена, ₴	Сумма	Удалить
Шедевр Тирация Ароматизированная вода 100 мл Парфюмная композиция Код: 998823	мл	1	216,05	216,05	Удалить
Салат томатный с огурцом Sandora 0.95 л Десертная композиция Код: 719493	л	5	27,54	137,70	Удалить

Вес Вашего заказа не менее 0,225 кг

К оплате без доставки: 353,75 ₴

Рисунок 1.7 – Сторінка оформлення замовлення інтернет-магазину «Produktoff»

Інтернет-магазин продуктів «NOVUS» - має дуже мінімалістичну структуру, у порівнянні з попередніми інтернет-магазинами, за якими проводиться аналіз, що дозволяє користувачеві отримати усю необхідну інформацію, як тільки він потрапить до сайту. Структура головної сторінки інтернет-магазину «NOVUS» наведена на рисунку 1.8.

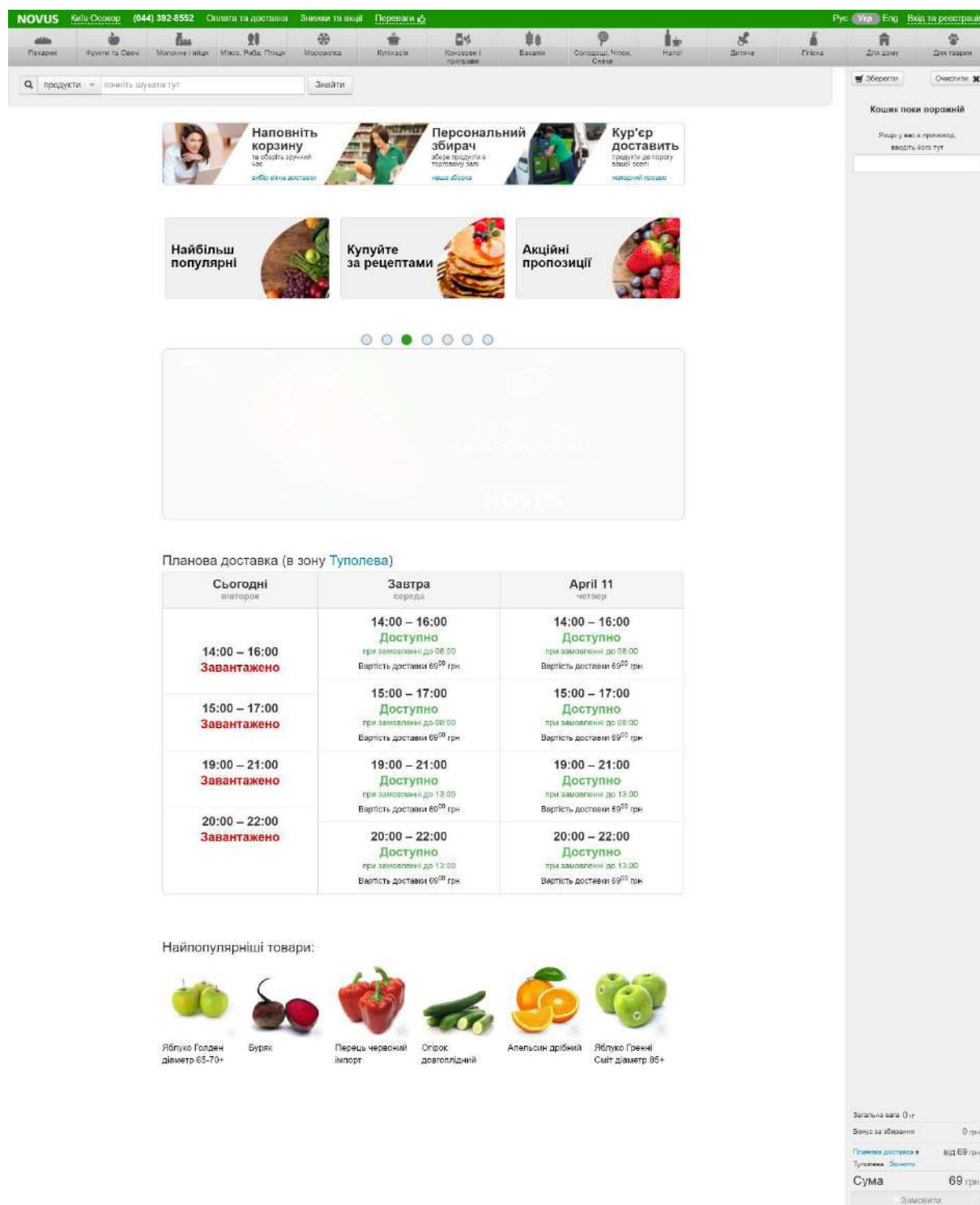


Рисунок 1.8 – Головна сторінка інтернет-магазину «NOVUS»

Так як, структура сайту мінімалістична, то це допомагає користувачеві дуже швидко замовити товар. Структура сторінки каталогу товарів інтернет-магазину «NOVUS» наведена на рисунку 1.9.

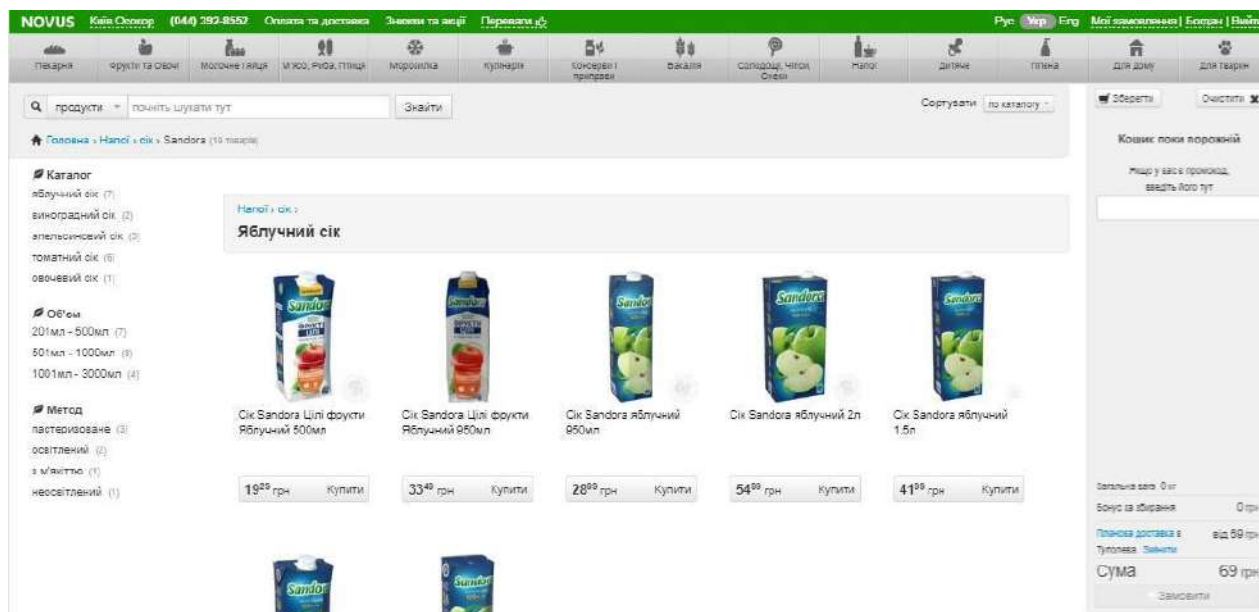


Рисунок 1.9 – Структура сторінки каталогу товарів інтернет-магазину «NOVUS»

Структура сторінки розширеної картки товару та сторінки оформлення замовлення інтернет-магазину «NOVUS» наведені на рисунку 1.10 та рисунку 1.11.

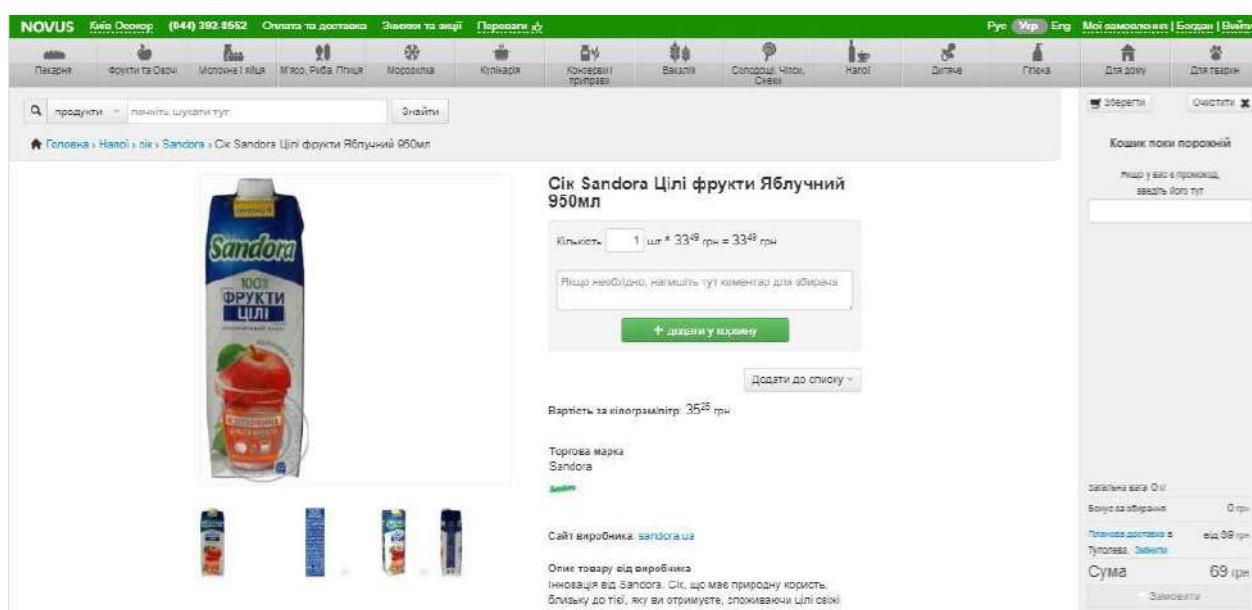


Рисунок 1.10 – Сторінка розширеної картки товару інтернет-магазину «NOVUS»

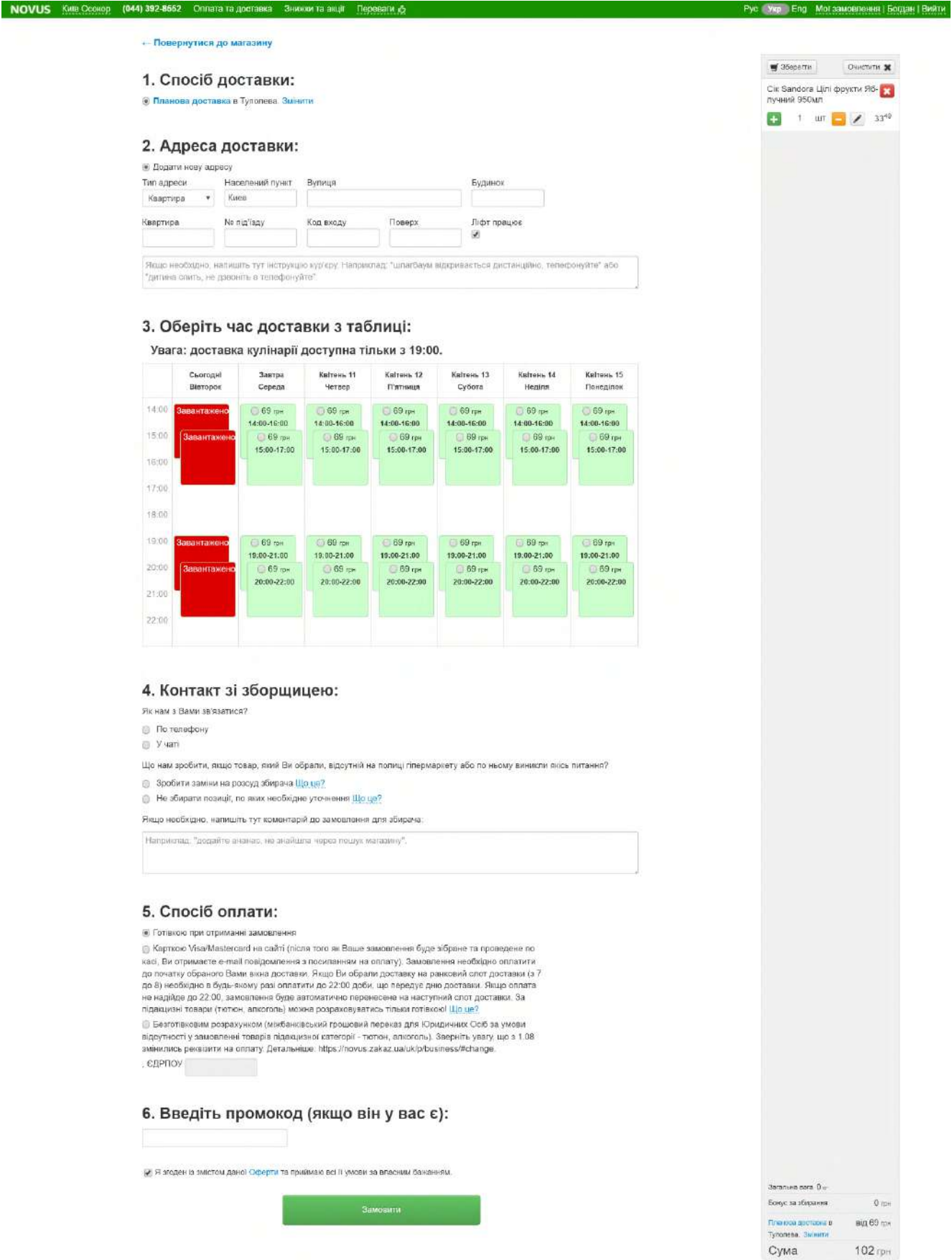


Рисунок 1.11 – Сторінка оформлення замовлення інтернет-магазину «NOVUS»

На основі аналізу цих трьох інтернет-магазинів сформована таблиця порівняння, яка наведена у таблиці 1.2.

Таблиця 1.2 – Порівняльна характеристика аналогів інтернет-магазинів продуктів

Назва інтернет-магазину	Інтернет-магазин «Ub.ua - marketplace»	Інтернет- магазин «Produktoff»	Інтернет- магазин «NOVUS»
Реєстрація на сайті	+	+	+
Форма зворотного зв'язку	-	+	-
Онлайн консультація користувачів	-	+	-
Зручний дизайн	-	+	+
Відгуки про товари	+	-	-
Можливість замовлення товару без мінімальної суми замовлення	-	-	+
Вибір зручного часу доставлення	-	+	+
Бонуси та знижки	+	+	+
Можливість замовлення товару по всій Україні	+	+	-
Пошук товару	+	+	+
Рейтинг товару	+	-	-
Вибір способу оплати	+	+	+
Вибір способу доставки	+	+	+

На основі цієї порівняльної характеристики інтернет-магазин повинен виконувати такі функції:

- реєстрація користувачів;
- забезпечення конфіденційності особистих даних користувачів сайту;
- надання інтерфейсу до бази даних товарів, що продаються (у вигляді каталогу, прайс-листу);
- надання консультацій користувачу он-лайн;
- робота з корзиною інтернет-магазину;
- оформлення замовлень з вибором способів оплати та доставки;
- надання необхідної та потрібної інформації для користувача;

- можливість залишити відгук, для поліпшення роботи вебсервісу;
- пошук товарів за назвою.

1.2.1 Опис функціональних вимог

Реєстрація. Для більш ефективного використання сайту, та зменшення часу замовлення відвідувач сайту повинен бути зареєстрований. Для цього йому повинна надаватися можливість зареєструватися на сайті, шляхом заповнення форми реєстрації (обов'язково вказавши своє ім'я, адресу електронної пошти та бажаний пароль). Після цього в особистому кабінеті він може вказати список бажаних товарів та адресу доставки, і при кожному замовленні витратити набагато менше часу.

Вхід до особистого кабінету. Відвідувач сайту, при наявності у нього особистого кабінету, може до нього зайти, вказавши у відповідній формі свій логін або адресу електронної пошти та пароль. Система має забезпечувати конфіденційність усіх даних користувача, і перевіряти унікальність полів. Також система повинна надавати можливість альтернативного входу за допомогою різних соціальних мереж.

Редагування корзини інтернет-магазину. Редагування корзини інтернет-магазину передбачає що клієнт може додавати товари до корзини з представленого асортименту товарів або видаляти їх з корзини. Також він може змінювати кількість товарів безпосередньо у корзині.

Оформлення замовлення. Оформлення замовлення клієнтом відбувається в корзині покупок, при наявності хоча б одної одиниці товару. Далі клієнт заповнює необхідну інформацію для доставки товару, та має можливість вибрати зручний для нього спосіб доставки та оплати. Після вибору усіх пунктів та заповнення необхідної інформації він очікує дзвінок оператора для підтвердження замовлення, та може переглядати статус свого замовлення (очікує підтвердження, в обробці, відправлено). Клієнт також може залишити додаткову інформацію в спеціальному полі про замовлення, яка стосується пакування або доставки.

Оплата товару. Оплата товарів відбувається при оформленні замовлення або безпосередньо при отриманні. Для оплати товарів, клієнту надаються різні

способи оплати: як за допомогою банківської картки, так і готівкою при отриманні товару на пошті або від кур'єра. При оплаті банківською картою безпека надається зі сторони банку, так як саме банк надає необхідну інформацію для оплати. Клієнт сам вирішує яким саме способом йому зручніше розрахуватися. Інформація про банківські операції користувача в інтернет-магазині повинна зберігатися лише на стороні банку.

Вихід із системи. Після завершення усіх необхідних дій, користувачі мають можливість вийти з системи.

1.2.2 Технічні питання

Система інтернет-магазину буде працювати на клієнт-серверній архітектурі. Для цього потрібен сервер, який буде приймати запити від клієнтів на перегляд сайту та отримання необхідної інформації з бази даних. Система має підтримувати усі види браузерів на різних платформах, наприклад Mozilla Firefox, Opera, Google Chrome.

1.2.3 Вимоги до зовнішніх інтерфейсів

Апаратний інтерфейс. Так як розробкою є вебсервіс, то вимог до апаратного інтерфейсу не виявлено.

Інтерфейс програмного забезпечення. Для того щоб система інтернет-магазину могла звертатися до бази даних, та отримувати від неї потрібну інформацію вебсервіс повинен бути написаний на мові скриптів, що можуть створювати доступ до бази даних. Система вимагає бази даних для зберігання інформації про товари, користувачів, транзакцій, наприклад MySQL або PostgreSQL. Також потрібен DNS (простір доменних імен) для іменування на інтернет. Для останнього користувачу потрібен WEB-браузер для взаємодії з системою.

1.2.4 Опис нефункціональних вимог

Продуктивність:

- система повинна обслужити 100 000 користувачів за день з середньою тривалістю сеансу 5 хвилин;

- всі Інтернет-сторінки, які генеруються системою, повинні повністю завантажувати щонайбільше за 10 секунд;
- завантаження відповідей на запити на екран повинна займати не більше 7 секунд після того, як користувач відіслав запит.

Безпека:

- дані про пароль та ідентифікатор користувача мають зберігатися у зашифрованому вигляді в базі даних;
- система повинна автоматично вийти з особистих кабінетів всіх клієнтів після бездіяльності;
- система не повинна залишати файли cookie на комп'ютері клієнта, що містять пароль користувача;
- редагувати інформацію про товари має можливість тільки адміністратор сайту;
- система повинна дозволяти клієнтам переглядати тільки замовлення, які зроблені ним особисто, але не іншими клієнтами.

Надійність. Система забезпечує зберігання всіх баз даних на резервних комп'ютерах з автоматичним перемиканням.

Надійність загальної програми залежить від надійності окремих компонентів. Основним елементом надійності системи є резервне копіювання бази даних, яка постійно підтримується та оновлюється, щоб відобразити останні зміни.

Таким чином, загальна стійкість системи залежить від стабільності сервера та його базової операційної системи.

Доступність. Система повинна бути доступна в будь-який час, тобто користувач може отримати до неї доступ за допомогою WEB-браузера, обмеженого лише часом простою сервера, на якому працює система. У разі збою обладнання або пошкодження бази даних буде показана сторінка непрацездатності. Також в разі збою обладнання або пошкодження бази даних резервні копії бази даних повинні вилучатись з сервера і зберігатися адміністратором. Потім служба має перезапуститися.

Портативність. Додаток заснований на HTML і мові сценаріїв. Таким чином, частина кінцевого користувача є повністю портативною, і будь-яка система, яка використовує будь-який WEB-браузер, повинна мати можливість використовувати функції системи, включаючи будь-яку апаратну платформу, яка доступна або буде доступна в майбутньому.

Кінцевий користувач може використовувати цю систему на будь-якій ОС: Windows, Linux і т.д, та на будь-якому пристрої: на ПК, ноутбуках і КПК і т.д.

2 МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Моделювання програмного забезпечення починається з діаграми прецедентів інтернет-магазину. В діаграмі прецедентів вказуються актори, тобто ті, хто безпосередньо користується інтернет-магазином та їх прецеденти, тобто дії, які вони можуть зробити в інтернет-магазині. З аналізу предметної області було прийнято виділити таких акторів: відвідувач сайту, клієнт, адміністратор сайту та консультант. Загальна діаграма прецедентів представлена на рисунку 2.1.

2.1 Прецедент «Реєстрація»

Основний виконавець: відвідувач сайту. Зацікавлені сторони та їх потреби:

- відвідувач сайту (хоче зареєструватися в системі, для подальшого отримання бонусів та зменшення витрат часу для замовлення товару);
- компанія (хоче мати більший прибуток від проданого товару).

Результати (постумови): користувача зареєстровано в системі.

Основний потік (основний успішний сценарій):

- відвідувач на сайті натискає кнопку «Зареєструватися»;

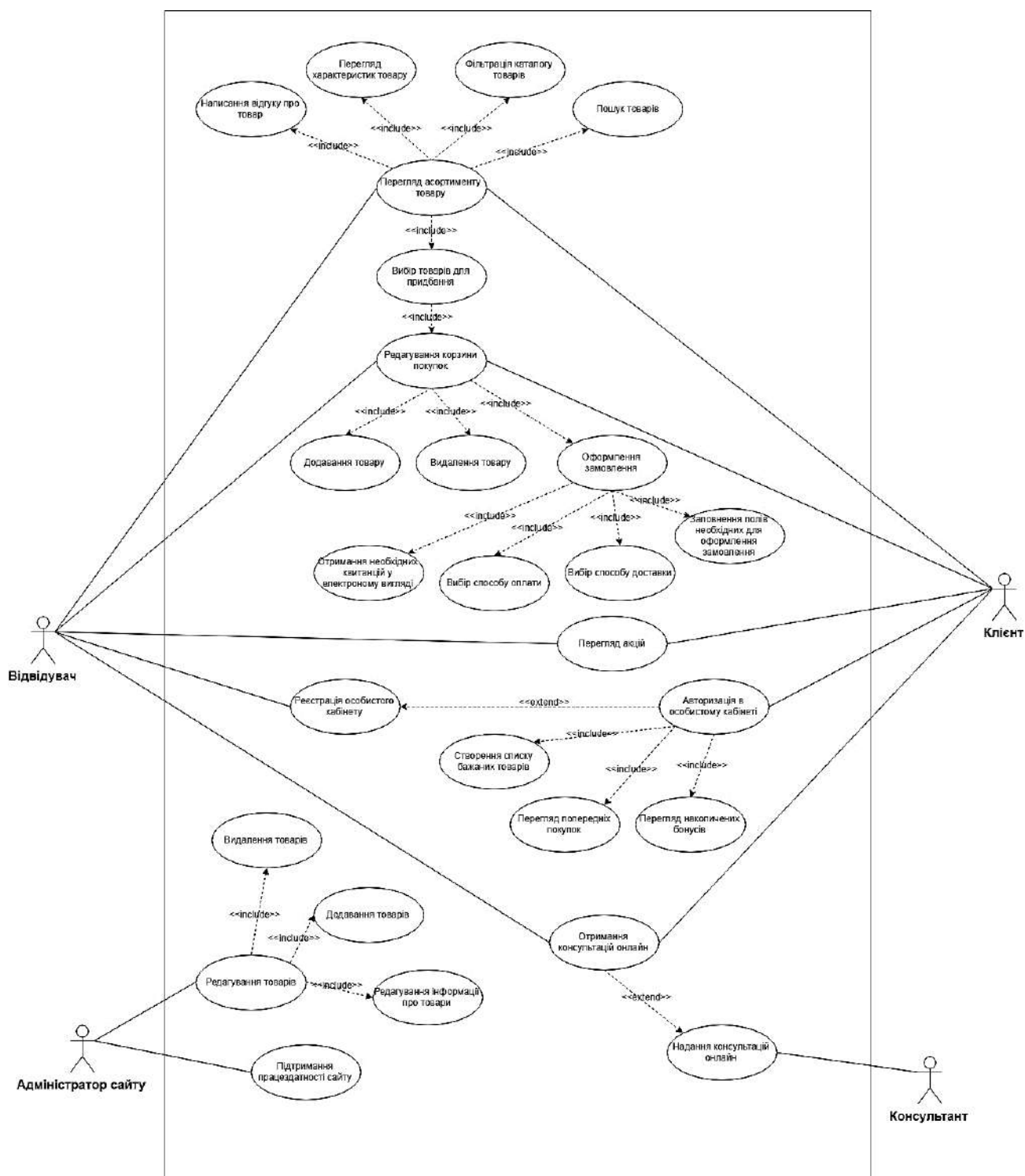


Рисунок 2.1 – Діаграма прецедентів інтернет-магазину

- система відкриває форму для реєстрації;
- відвідувач заповнює необхідну інформацію для реєстрації;
- система проводить валідацію даних в заповнених полях;
- відвідувач натискає кнопку «Зареєструватися»;

- система надсилає на вказану в формі реєстрації електронну пошту або телефон код підтвердження реєстрації;
- система відкриває форму для перевірки коду підтвердження;
- користувач вводить в поле код підтвердження;
- система реєструє відвідувача.

Альтернативні потоки (розширення).

Вказані некоректні дані в основному потоку «система проводить валідацію даних в заповнених полях»:

- система інформує відвідувача сайту, в яких полях вказані некоректні дані та просить його виправити їх;
- користувач виправляє дані у вказаних полях;
- система проводить валідацію даних у виправлених полях.

Користувач не натиснув кнопку «Зареєструватися» та перейшов до іншої сторінки сайту: система видаляє усі заповнені дані з форми.

Система відкрила форму для перевірки коду підтвердження, але код підтвердження не дійшов до відвідувача впродовж однієї хвилини:

- на формі з'являється кнопка «Відправити код ще раз»;
- користувач натискає кнопку;
- система відправляє код ще раз.

Діаграма послідовності для прецеденту «Реєстрація» наведена на рисунку 2.2.

2.2 Прецедент «Придбання товарів»

Основний актор: клієнт інтернет-магазину.

Зацікавлені сторони та їх потреби:

- клієнт (хоче придбати необхідні йому товари);
- компанія (хоче мати більший прибуток від проданого товару).

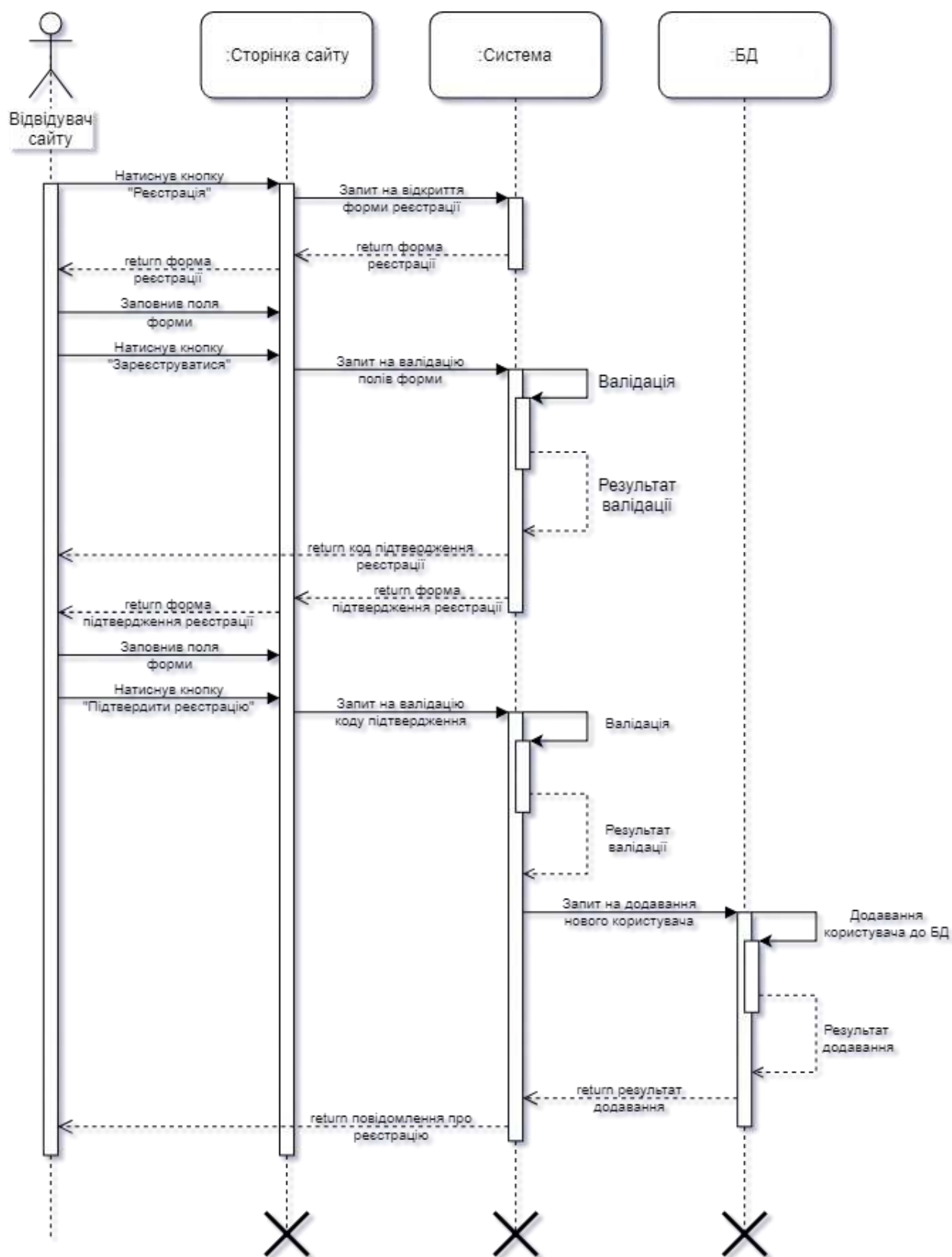


Рисунок 2.2 – Діаграма послідовності «Реєстрація»

Результати (постумови): клієнт придбав товар. Компанія отримала прибуток від продажу цього товару.

Основний потік (основний успішний сценарій):

- клієнт вибирає товарну категорію, що представлена у панелі навігації;
- система відкриває сторінку з асортиментом товару вибраної категорії;
- клієнт переглядає асортимент товару;
- клієнт вибирає у наданому асортименту товар та натискає кнопку

«Додати до коззини»;

- система ідентифікує товар з бази даних;
- система додає товар до коззини покупок клієнта;
- клієнт натискає кнопку «Перейти до коззини»;
- система відкриває вікно коззини покупок користувача;
- клієнт натискає кнопку «Оформити замовлення»;
- система відкриває форму з полями для оформлення замовлення;
- клієнт заповнює адресу замовлення;
- клієнт вибирає спосіб доставки;
- клієнт вибирає спосіб оплати;
- клієнт натискає «Придбати товар»;
- система проводить валідацію вибраних та заповнених полів;
- система реєструє заявку у системі.

Альтернативні потоки (розширення).

Клієнт вийшов з системи в основному потоку «клієнт натискає «Придбати товар»:

- система видаляє усі заповнені та вибрані поля;
- система відкриває головну сторінку сайту.

Система знайшла не вибране або незаповнене поле в основному потоку «система проводить валідацію вибраних та заповнених полів»:

- система інформує клієнта про незаповнене поле;
- клієнт заповнює незаповнене поле.

Діаграма послідовності для прецеденту «Придбання товарів» наведена на рисунку 2.3.

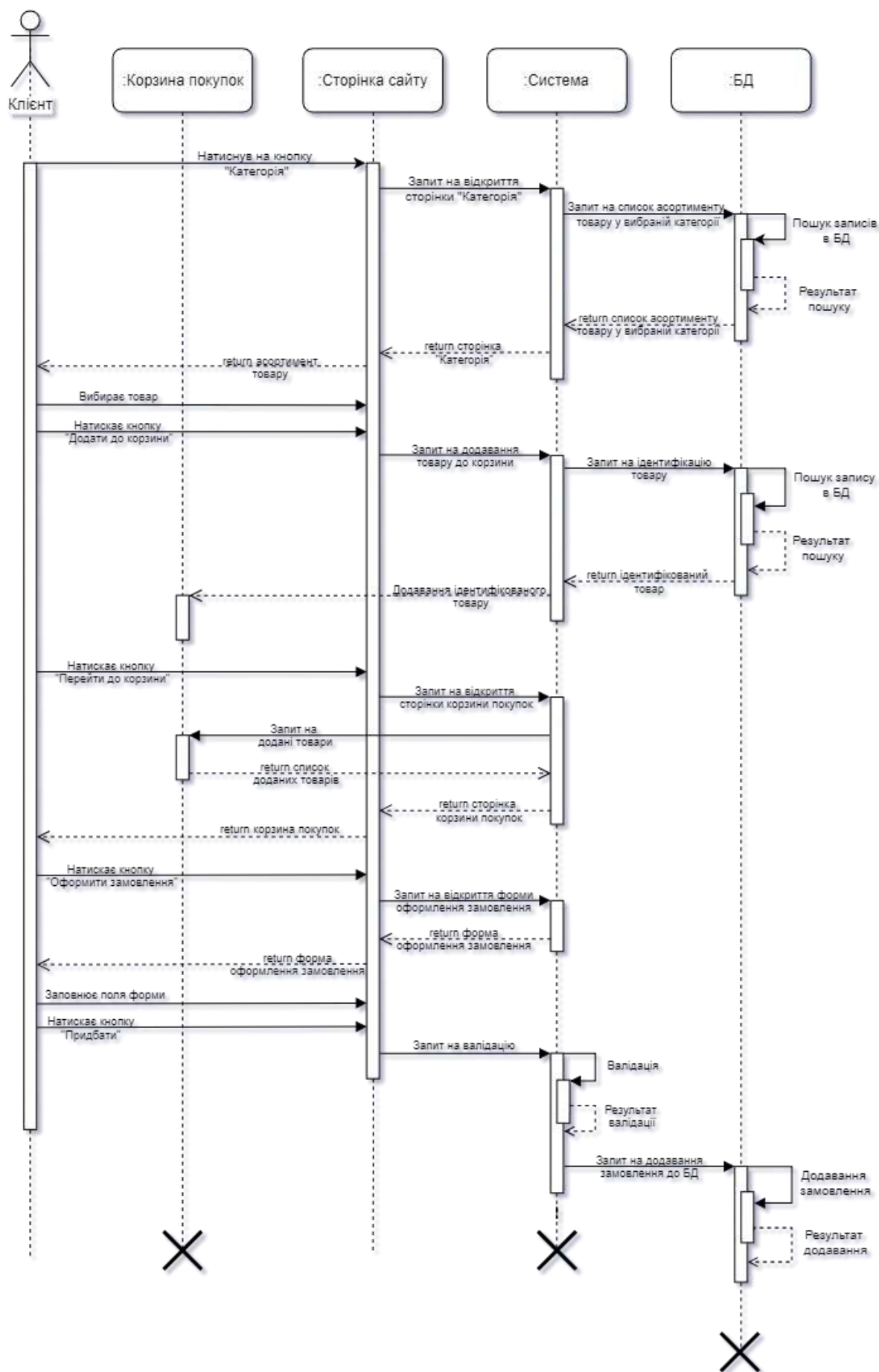


Рисунок 2.3 – Діаграма послідовності «Придбання товарів»

2.3 Прецедент «Повторне замовлення товару»

Основний актор: клієнт інтернет-магазину.

Зацікавлені сторони та їх потреби:

- клієнт (хоче придбати той самий список товарів, які вже покупав до цього);
- компанія (хоче мати більший прибуток від проданого товару).

Результати (постумови): клієнт придбав товар. Компанія отримала прибуток від продажу цього товару.

Основний потік (основний успішний сценарій):

- клієнт натискає кнопку «Вхід до особистого кабінету»;
- система відкриває форму з полями автентифікації клієнта;
- клієнт заповнює поля даними, які використовувалися при реєстрації;
- система проводить валідацію заповнених полів, та автентифікацію клієнта;
- система відкриває сторінку особистого кабінету клієнта;
- клієнт натискає кнопку «Історія покупок»;
- система шукає історію покупок даного клієнта;
- система відкриває сторінку з історією покупок даного клієнта;
- клієнт натискає кнопку «Придбати знову»;
- система додає товар з історії покупок до кошика;
- система відкриває сторінку оформлення замовлення;
- клієнт вибирає спосіб доставки;
- клієнт вибирає спосіб оплати;
- клієнт натискає «Замовити»;
- система проводить валідацію вибраних та заповнених полів;
- система реєструє замовлення у системі.

Альтернативні потоки (розширення):

Система не автентифікувала користувача в основному потоку «система проводить валідацію заповнених полів, та автентифікацію клієнта»: система інформує користувача про невірні дані.

Система знайшла не вибране або незаповнене поле в основному потоку «система проводить валідацію вибрані та заповнені поля»:

- система інформує клієнта про незаповнене поле;
- клієнт заповнює незаповнене поле.

Діаграма послідовності для прецеденту «Повторне замовлення товару» наведена на рисунку 2.4.

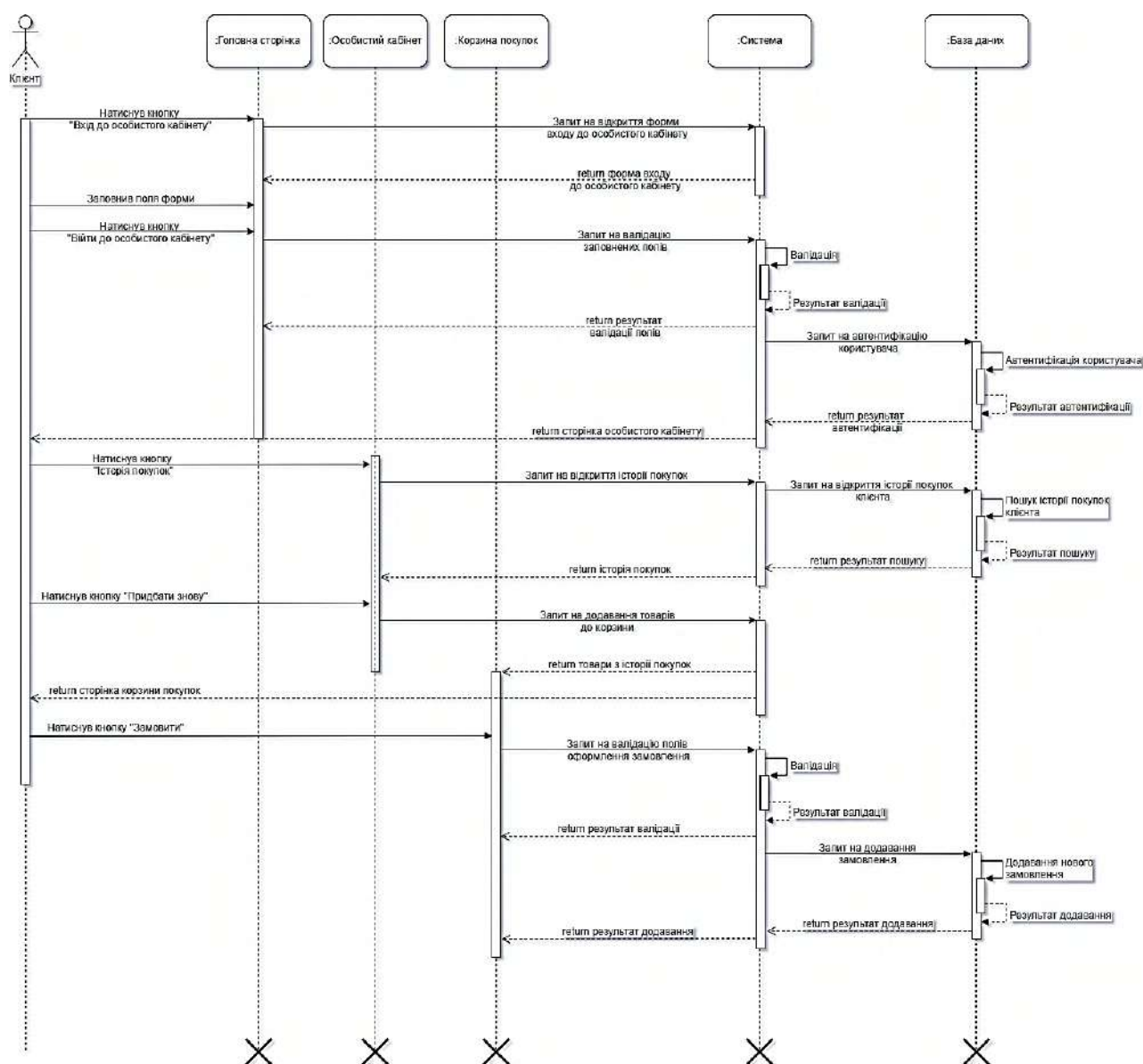


Рисунок 2.4 – Діаграма послідовності «Повторне замовлення товару»

3 ПРОЄКТУВАННЯ ВЕБСЕРВІСУ

3.1 Архітектура вебсервісу

Для вебсервісу, що розробляється, була обрана архітектура «клієнт-сервер».

«Клієнт - сервер» (англ. Client-server) - обчислювальна або мережева архітектура, в якій завдання або мережеве навантаження розподілені між постачальниками послуг, званими серверами, і замовниками послуг, званими клієнтами. Фактично клієнт і сервер - це програмне забезпечення. Зазвичай ці програми розташовані на різних обчислювальних машинах і взаємодіють між собою через обчислювальну мережу за допомогою мережевих протоколів, але вони можуть бути розташовані також і на одній машині [8].

Основний принцип технології «клієнт-сервер» полягає в поділі функцій додатка на три групи:

- введення і відображення даних (взаємодія з користувачем);
- прикладні функції, характерні для даної предметної області;
- функції управління ресурсами (файловою системою, базою даних і т.д.).

Тому, в будь-якому додатку виділяються наступні компоненти:

- компонент представлення даних;
- прикладний компонент;
- компонент управління ресурсами [8].

Клієнт і сервер взаємодію один з одним в мережі Інтернет або в будь-який інший комп'ютерної мережі за допомогою різних мережевих протоколів, наприклад, IP протокол, HTTP протокол, FTP та інші. Протоколів насправді дуже багато і кожен протокол дозволяє надавати ту чи іншу послугу. Наприклад, за допомогою HTTP протоколу браузер відправляє спеціальне HTTP повідомлення, в якому зазначено яку інформацію і в якому вигляді він хоче отримати від сервера, сервер, отримавши таке повідомлення, відсилає браузеру у відповідь схоже по структурі повідомлення (або кілька повідомлень), в якому міститься потрібна інформація, як правило це HTML документ.

Повідомлення, які посилають клієнти отримали назви HTTP запити. Запити мають спеціальні методи, які говорять сервера про те, як обробляти повідомлення. А повідомлення, які посилає сервер отримали назву HTTP відповіді, вони містять крім корисної інформації ще й спеціальні коди стану, які дозволяють браузеру дізнатися те, як сервер зрозумів його запит.

Архітектура вебсервісу наведена на рисунку 3.1.

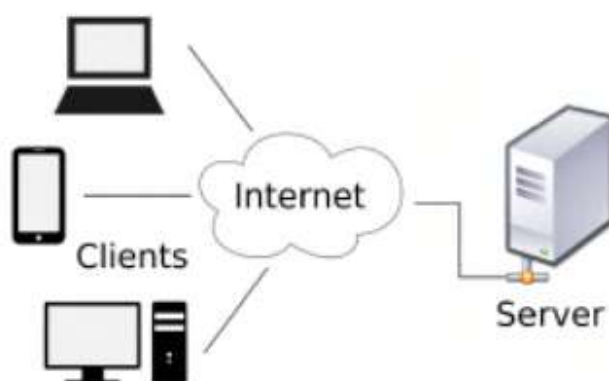


Рисунок 3.1 – Архітектура вебсервісу

3.2 Проєктування бази даних

Проєктування бази даних починається з побудови семантичної моделі предметної області. Цю стадію прийнято називати концептуальним проєктуванням бази даних.

Концептуальне проєктування бази даних - процес створення моделі використовуваної на підприємстві інформації, що не залежить від будь-яких фізичних аспектів її представлення. Перша фаза процесу проєктування бази даних називається концептуальним проєктуванням бази даних. Вона полягає в створенні концептуальної моделі даних для аналізованої частини підприємства. Ця модель даних створюється на основі інформації, записаної в специфікаціях вимог користувачів. Концептуальне проєктування бази даних абсолютно не залежить від таких подробиць її реалізації, як тип обраної цільовий СКБД, набір створюваних

прикладних програм, використовувані мови програмування, тип обраної обчислювальної платформи, а також від будь-яких інших особливостей фізичної реалізації [9].

Виходячи з описаної предметної області та вимог до продукту можна виділити такі сутності:

- продукти (Products);
- користувачі (Customers);
- статуси аккаунту (Account_Statutes);
- категорії (Categories);
- коментарі до продуктів (Products_Comments);
- рейтинг продуктів (Product_Rating);
- замовлення (Orders);
- способи оплати (Payments);
- способи доставки (Deliveries);
- статус замовлення (Order_Statutes);
- деталі замовлення (Order_Details);
- атрибути (Attributes);
- значення атрибутів (Attributes_Values);
- список бажань (Wish List);
- елементи списку бажань (Wish List Entries).

Стрижневими сутностями для даного вебсервісу є «Продукти» та «Користувачі».

Для прикладу будуть наведені властивості атрибутів сутностей «Продукти», «Категорії» та «Користувачі».

Сутність «Продукти» складається з таких атрибутів:

- Product_ID – унікальний ідентифікатор продукту;
- Category_ID – унікальний ідентифікатор категорії продукту;
- Product_Name – поле з назвою продукту;
- Product_Manufacture – поле з виробником продукту;

- Product_Description – поле з описом продукту;
- Product_Size – поле зі значенням одиниць виміру продукту;
- Product_Measure – поле з одиницею виміру продукту;
- Product_Price – поле з ціною продукту;
- Product_Stock – поле з кількістю продукту, що залишилося;
- Product_Image – поле зі шляхом до фотографії продукту.

Сутність «Категорії» складається з таких полів:

- Category_ID – унікальний ідентифікатор категорії;
- Category_Name – поле з назвою категорії;
- Category_Parent_ID – поле з ідентифікатором батьківської категорії по відношенню до даної категорії;
- Category_Description – поле з описом категорії;
- Category_Image – поле зі шляхом до фотографії категорії.

Атрибути сутності «Користувачі»:

- Customer_ID – унікальний ідентифікатор користувача;
- Customer_FName – поле з ім'ям користувача;
- Customer_LName – поле з прізвищем користувача;
- Customer_Email – поле з емейлом користувача;
- Customer_Phone – поле з мобільним телефоном користувача;
- Customer_Password – поле з паролем користувача у зашифрованому вигляді;
- Customer_Bonus – поле з накопиченими бонусами користувача;
- Account_Status_ID – ідентифікатор статусу аккаунту.

Між сутностями «Категорії» та «Продукти» існує зв'язок 1:М, обов'язковий зі сторони «Продукти», оскільки кожен з продуктів обов'язково повинен відноситися до категорії. Сутності «Категорії» і «Продукти», а також зв'язок між ними наведений на рисунку 3.2.

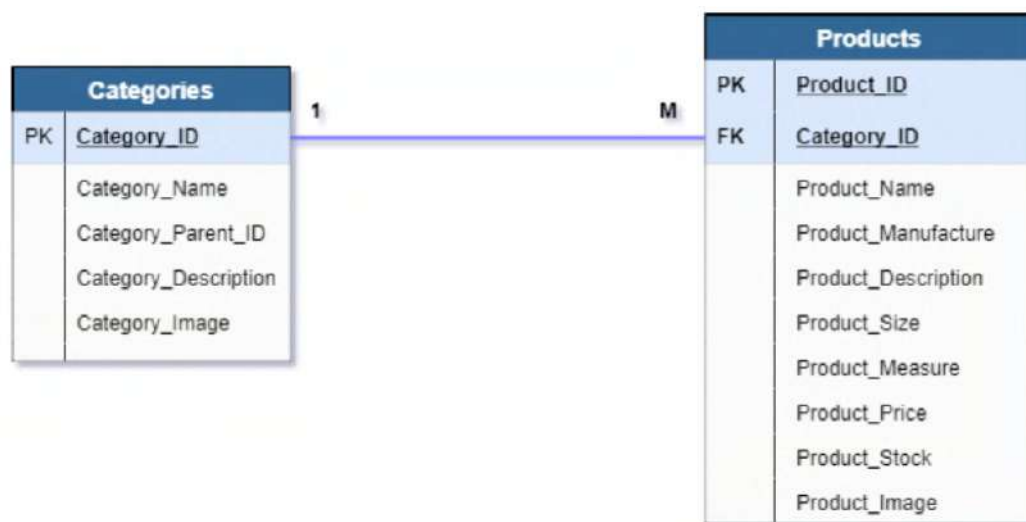


Рисунок 3.2 – Сутності «Категорії», «Продукти» і зв'язок між ними

Між сутностями «Продукти» і «Коментарі до продуктів» існує зв'язок 1:M, обов'язковий зі сторони «Коментарі до продуктів», коментар обов'язково повинен мати ідентифікатор того продукту, до якого користувач вирішив залишити коментар. Сутності «Продукти» і «Коментарі до продуктів», а також зв'язок між ними наведений на рисунку 3.3.

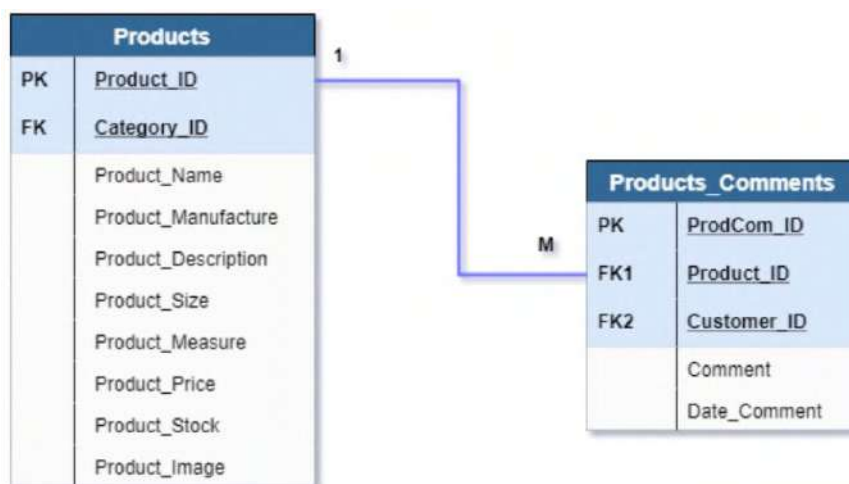


Рисунок 3.3 - Сутності «Продукти», «Коментарі до продуктів» і зв'язок між ними

Сутності «Користувачі» та «Статус аккаунта» об'єднані зв'язком 1:M, обов'язковим зі сторони «Користувачі», оскільки кожен користувач повинен мати статус свого перебування на сайті. Сутності «Користувачі» і «Статуси аккаунту», а також зв'язок між ними наведений на рисунку 3.4.

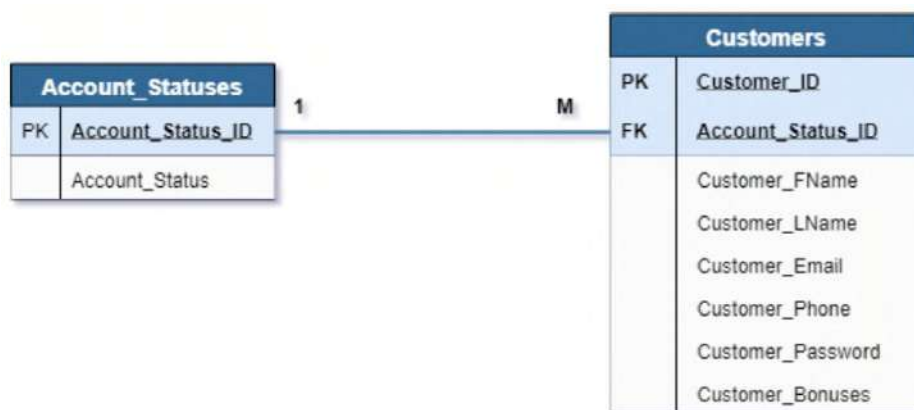


Рисунок 3.4 - Сутності «Користувачі», «Статуси аккаунту» і зв'язок між ними

Сутності «Користувачі» та «Коментарі до продуктів» об'єднані зв'язком 1:M, обов'язковим зі сторони «Коментарі до продуктів», оскільки кожен коментар повинен мати ідентифікатор свого користувача, який написав коментар. Сутності «Користувачі» і «Коментарі до продуктів», а також зв'язок між ними наведений на рисунку 3.5.

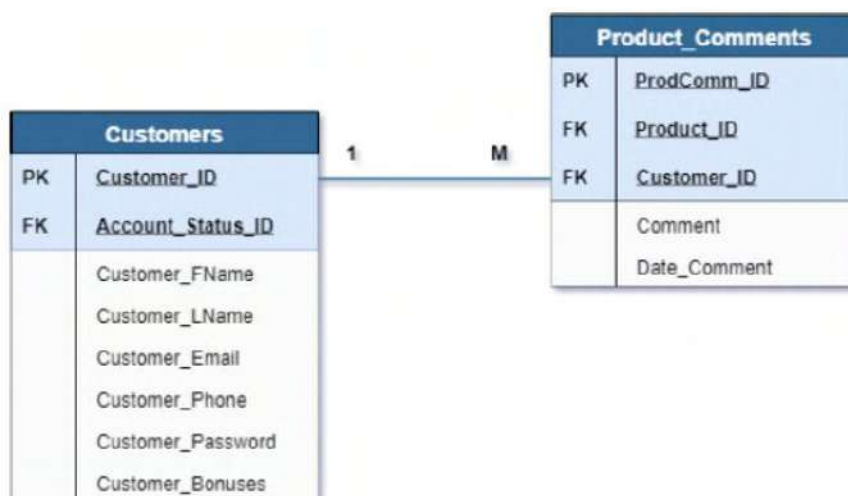


Рисунок 3.5 - Сутності «Користувачі», «Коментарі до продуктів» і зв'язок між ними

Усі інші зв'язки та сутності з їх атрибутами наведені у ER-діаграмі на рисунку 3.6.

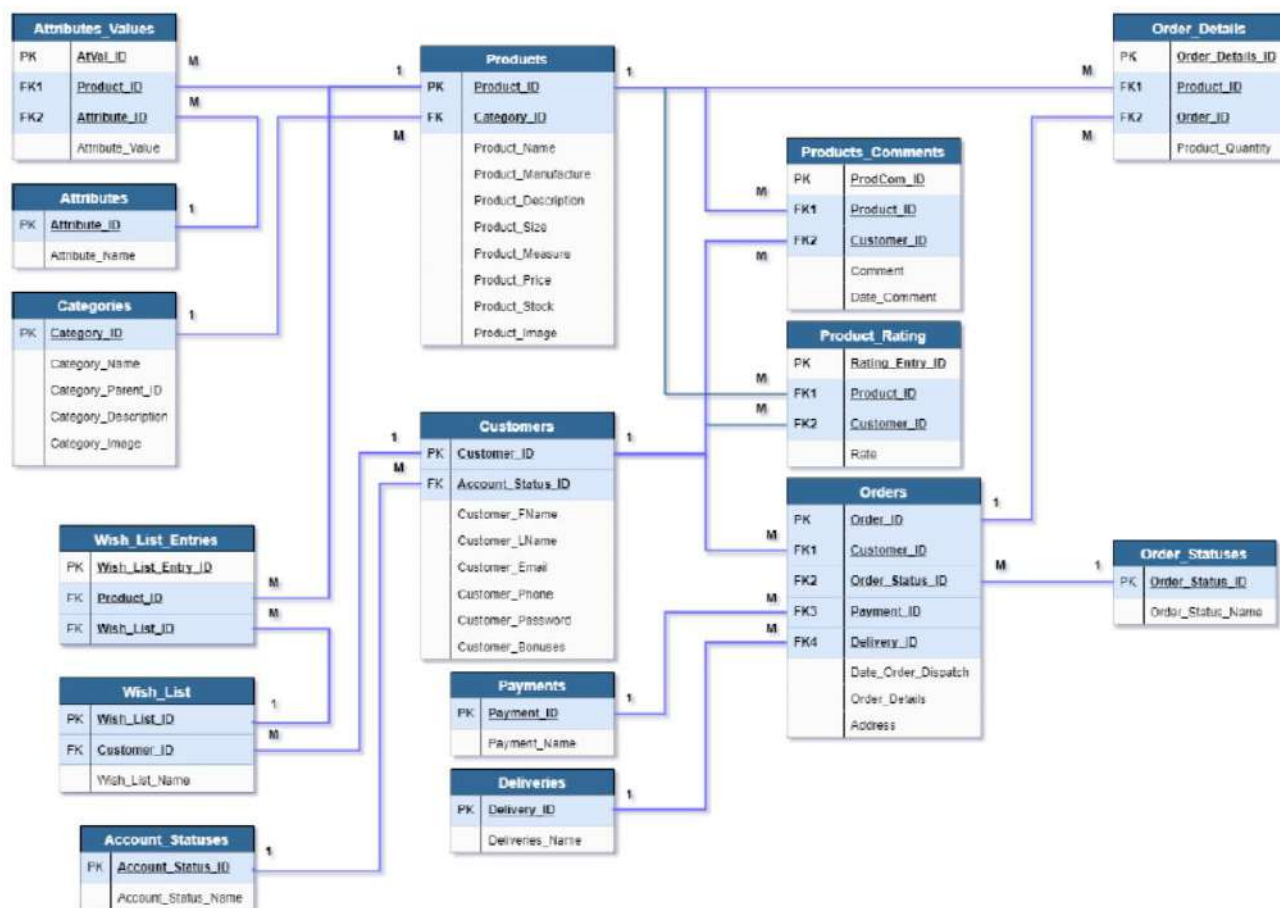


Рисунок 3.6 – ER-діаграма бази даних вебсервісу

3.3 Аналіз та вибір СКБД

Для вирішення задачі дипломного проекту вибір СКБД здійснюється за кількома критеріями:

- вартість продукту СКБД;
- спрощення задачі, за рахунок різних програмних компонентів СКБД;
- швидкодія;
- наявність вкладених запитів;
- кросплатформеність БД;
- максимальний розмір БД, яку СКБД може підтримувати, та кількість файлів для її підтримки.

Виходячи з даних критеріїв, для створення бази даних сайту курсового проєкту було проаналізовано три СКБД, а саме MS SQL SERVER, MySQL та PostgreSQL.

MySQL – кросплатформна СКБД, що має програмні інтерфейси для мов C, Java, PHP, Python. Завдяки цьому широко використовується для створення власних додатків. Обмежень на максимальний розмір таблиці немає. Розмір таблиці обмежений її типом. У загальному випадку тип MyISAM обмежений граничним розміром файлу в файловій системі операційної системи. Наприклад, в NTFS цей розмір теоретично може бути до 32 ексабайт. У разі InnoDB одна таблиця може зберігатися в декількох файлах, які мають єдиний табличний простір. Розмір останнього може досягати 64 терабайт. MySQL підтримує більшість функціоналу SQL. Робота з MySQL відбувається у консольному режимі, але існують графічні оболонки для більш універсального використання (phpMyAdmin, HeidiSQL, Workbench). MySQL дозволяє оптимізувати створення і налаштування БД за допомогою виконання команд для роботи зі скриптами SQL. З недоліків можна виділити те, що MySQL має платну підтримку [10].

PostgreSQL – безкоштовна об'єктно-реляційна СКБД, яка пропонується в різних версіях для широкого числа платформ – від Unix – систем і Linux, до MS Windows і MacOS. В PostgreSQL розмір бази даних може досягати практично необмежених обсягів. Число записів та індексів в таблиці також не має обмежень. Максимальний розмір таблиці може становити 32 гігабайт, а записи – 1,6 терабайт. СКБД має підтримку стандартних, часткових і функціональних індексів, планувальником запитів, систему блокування, контроль буферної і кеш-пам'яті, табличними просторами. Всі ці функції PostgreSQL зумовлюють високу продуктивність за рахунок масштабованості і низької потреби в апаратних ресурсах. PostgreSQL краще стискає і розтискає дані, дозволяючи зберегти більше даних на дисковому просторі. До недоліків можна віднести те, що PostgreSQL не має графічних оболонок для простоти використання [11].

Microsoft SQL SERVER - це серверна база даних від Microsoft, де для написання скриптів використовується Transact-SQL.

Найважливіші характеристики даної СКБД :

- простота адміністрування;
- можливість підключення до WEB;
- швидкодія і функціональні можливості механізму сервера СКБД;
- наявність засобів віддаленого доступу.

У комплект засобів адміністративного управління даної СКБД входить цілий набір спеціальних майстрів і засобів автоматичної настройки параметрів конфігурації. Також дана БД оснащена чудовими засобами тиражування, що дозволяють синхронізувати дані ПК з інформацією БД і навпаки. Вхідний в комплект поставки сервер OLAP дає можливість зберігати і аналізувати всі наявні у користувача дані. В принципі дана СКБД являє собою сучасну повнофункціональну базу даних, яка ідеально підходить для малих і середніх організацій. Необхідно зауважити, що SQL Server поступається іншим розглянутим СКБД по двох важливих показників: програмованність і засоби роботи. При розробці клієнтських БД додатків на основі мов Java, HTML часто виникає проблема недостатності програмних засобів SQL Server і користуватися цією СКБД буде важче, ніж іншими СКБД. Загальносвітовою тенденцією в XXI столітті став практично повсюдний перехід на платформу LINUX, а SQL Server функціонує тільки в середовищі Windows. Тому використання SQL Server доцільно, тільки якщо для доступу до вмісту БД використовується виключно стандарт ODBC, в іншому випадку краще використовувати інші СКБД [12].

Виходячи з переваг і недоліків даних СКБД, для вирішення задачі курсового проєкту вибрана СКБД MySQL, за кількома ключовими факторами. Завдяки кросплатформенності даної СКБД, можна переносити цю базу даних на інші операційні системи, для різних цілей. Вкладені запити надають можливість обчислень полів, для більшої автоматизації роботи з даною СКБД. Завдяки тому що, запити можна зберігати у файлах, можна зберігати «скелет» бази даних, як резервну копію створення бази даних, або для більш швидкої виконання запитів відображення інформації, зокрема для адміністратора бази даних. І останнім

ключовим фактором є те, що для MySQL існує дуже багато графічних оболонок, що дозволяють швидше створювати бази даних.

3.4 Проєктування інтерфейсу користувача

Інтерфейсом користувача називається візуальне оформлення сторінок проєкту. Для того, щоб показати як саме буде оформлюватися сторінка створюються макети або прототипи цих самих сторінок. Для курсового проєкту були створені прототипи декількох сторінок інтернет-магазину: сторінка асортименту товарів за категорією, головна сторінка сайту, сторінка авторизації.

Прототип сайту - схеми його сторінок (wireframes), зібрані в структуру і частково або повністю імітують роботу інтерактивних елементів і серверної частини.

Для даного вебсервісу були спроектовані чотири прототипи сторінок: головна сторінка, сторінка з категоріями товарів, сторінка з товарами, кабінет користувача.

Для кожної з цих сторінок є два блоки, які повторюються – «шапка» сайту (header) та «підвал» сайту (footer).

В «шапці» сайту знаходяться такі елементи:

- логотип;
- блок навігації;
- кнопки входу та реєстрації;
- кнопка пошуку;
- кнопка переходу до корзини покупок.

В «підвалі» сайту може розміщуватися контактна інформація, кнопки соціальних мереж, та будь-яка інша інформація, тому на прототипах сайту «підвал» називається просто «Footer».

Головна сторінка сайту – інформаційна. На ній першою секцією після «шапки» сайту є секція, що дозволяє користувачеві перейти до категорії товарів, якщо він не зацікавлений в інформації споживача. Потім йде слайдер, який дає інформацію користувачеві про акції, які проходять в даний момент в інтернет-магазині. За цим йдуть дві секції, які допомагають користувачеві зробити вибір: робити замовлення товарів чи ні, та як це зробити. І останньою секцією – секція, що дозволяє користувачеві залишити відгук про поліпшення сайту або про деякі зауваження про будь-яку роботу вебсервісу та інтернет-магазину в цілому. Прототип головної сторінки інтернет-магазину представлено на рисунку 3.7.

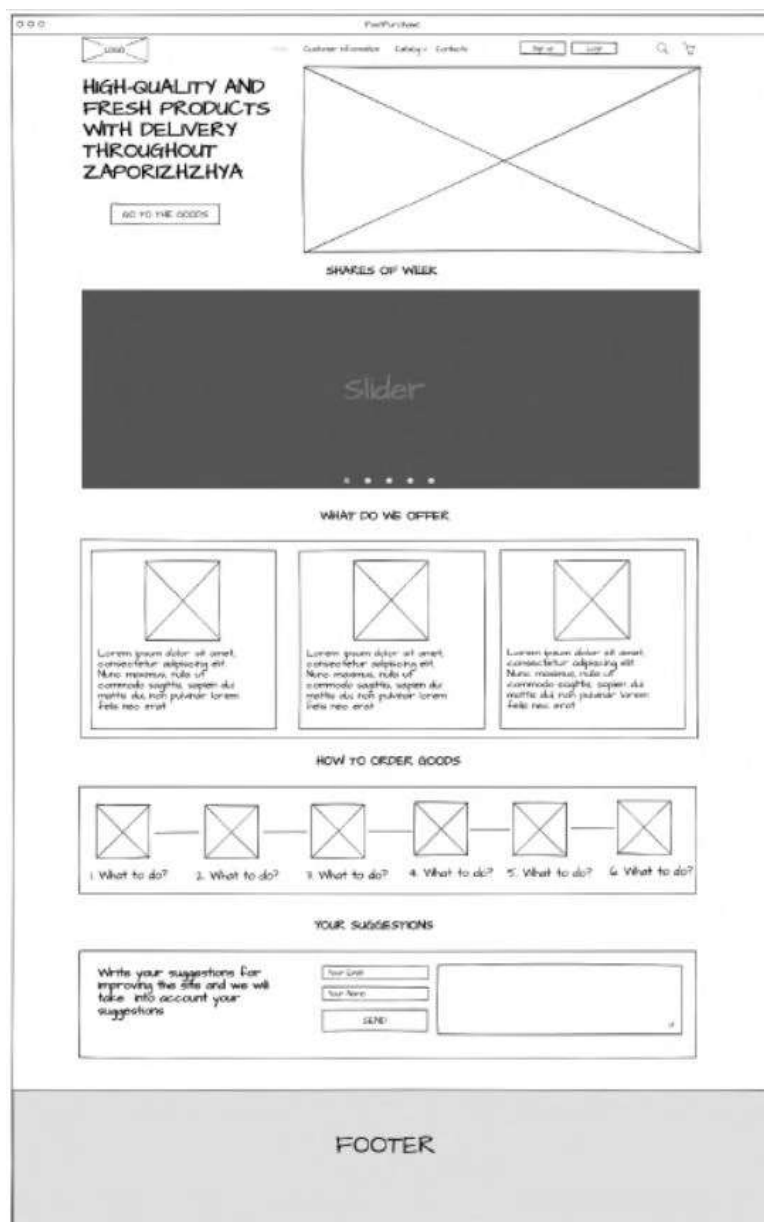


Рисунок 3.7 – Прототип головної сторінки інтернет-магазину

Сторінка категорії товарів складається з блоків, які в свою чергу мають фотографію, що ідентифікує категорію, заголовок з назвою категорії та кнопку переходу до товарів даної категорії. Прототип сторінки категорії товарів наведений на рисунку 3.8.

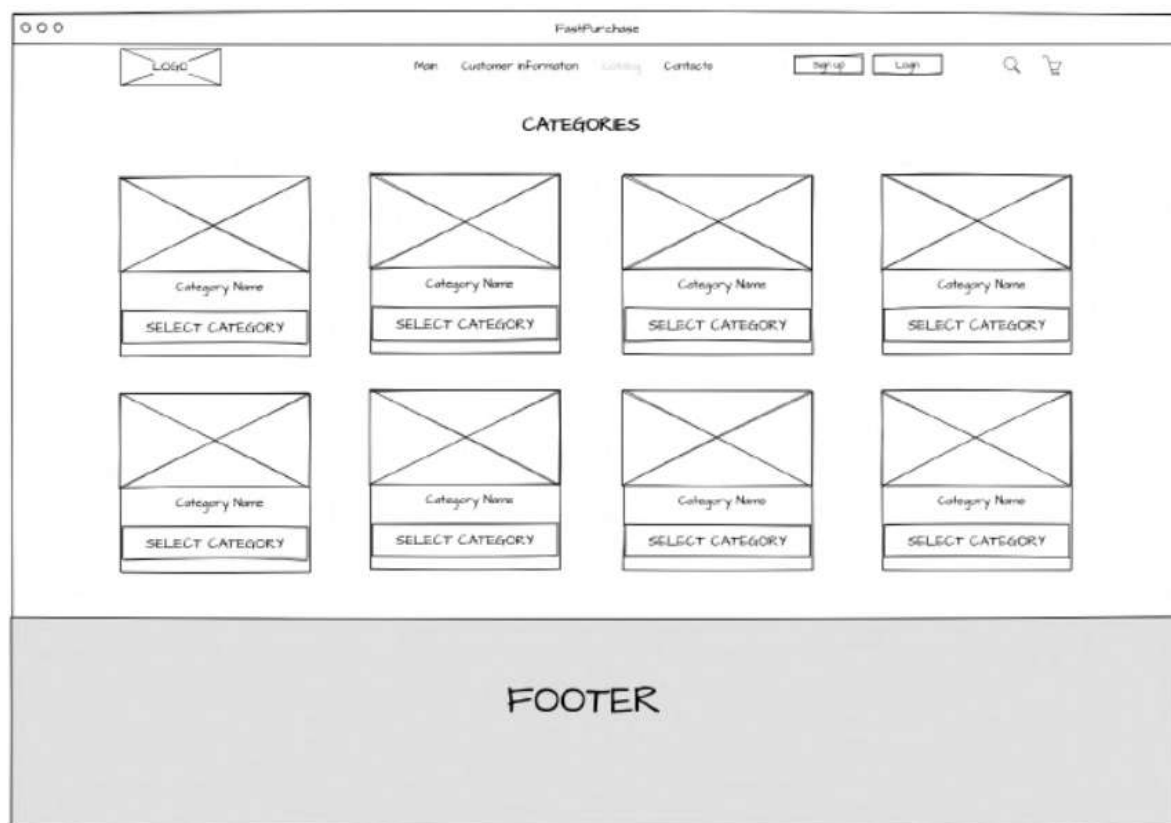


Рисунок 3.8 – Прототип сторінки категорії товарів

Наступним прототипом є сторінка з товарами якоїсь категорії. На прототипі даної сторінки видно блок з фільтрами для більш зручного пошуку товарів, та блок з самими товарами (картки товарів). У свою чергу, картка товару має фотографію, що візуально дає зрозуміти, що це за товар, назва товару, його короткий опис, ціна, та дві кнопки: додавання до кошика товару та додавання до списку бажань. Також до цієї сторінки можна перейти за допомогою панелі навігації в «шапці» сайту. Прототип сторінки з товарами категорії наведений на рисунку 3.9.

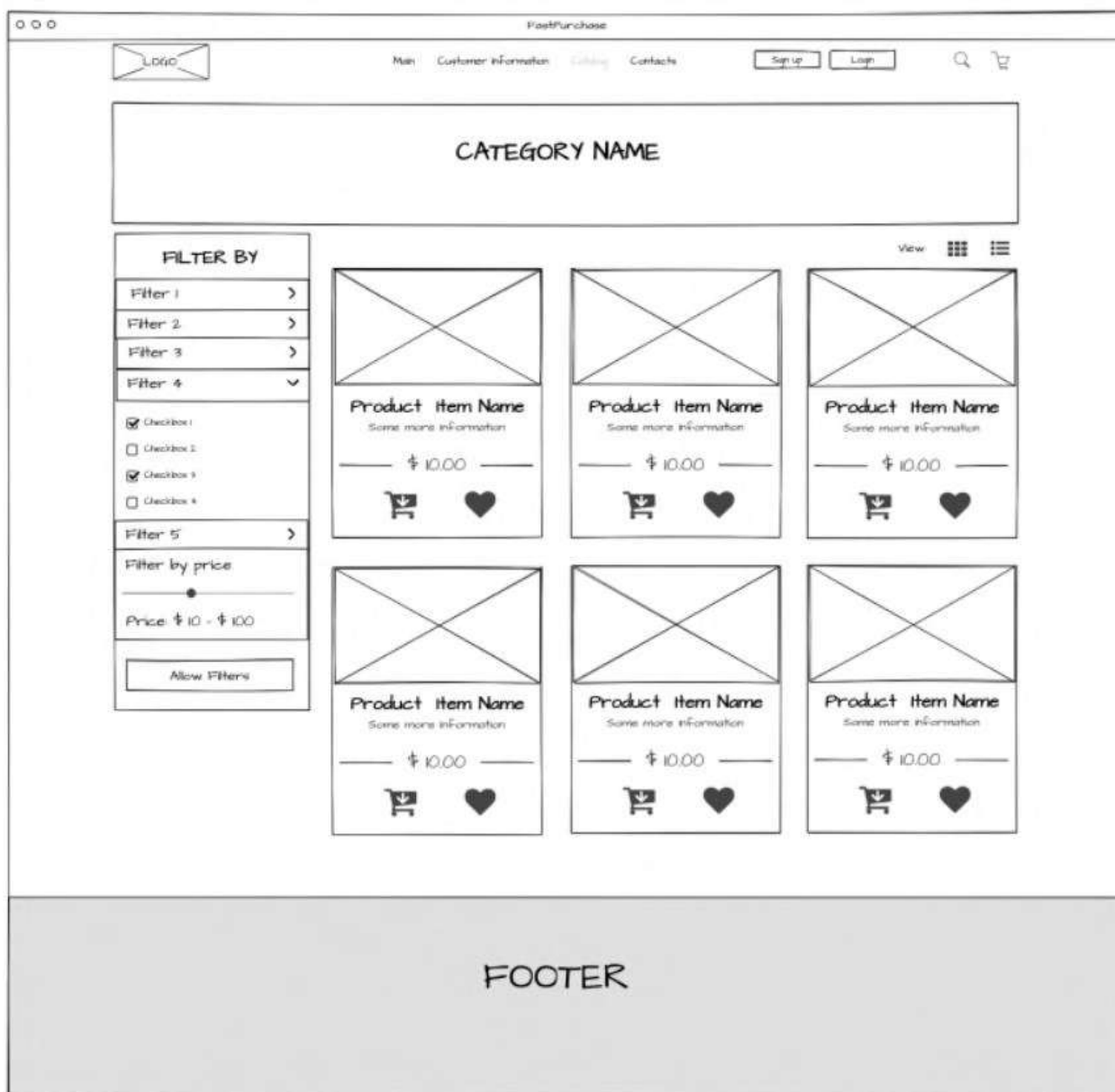


Рисунок 3.9 – Прототип сторінки з товарами категорії

Останнім прототипом сторінки є сторінка особистого кабінету користувача. Коли користувач заходить до особистого кабінету, він бачить зліва своє ім'я та прізвище, і бонуси, які він зміг отримати. Також, нижче знаходяться кнопки переходу для отримання різної інформації користувача: загальна інформація користувача, історія замовлень, список бажань. На прототипі цієї сторінки прикладом є історія замовлень. За допомогою цієї сторінки, користувач може переглядати усі замовлення, які він робив за увесь час користування інтернет-магазином, а також робити повторне замовлення такого самого набору товарів. Прототип сторінки особистого кабінету наведений на рисунку 3.10.



Рисунок 3.10 – Прототип сторінки особистого кабінету користувача

3.5 Аналіз та вибір засобів розробки

Так як даний вебсервіс реалізується для інтернет-магазину продуктів, то як будь-якому інтернет-магазину необхідний графічний інтерфейс. У даному випадку графічний інтерфейс реалізується за допомогою сторінок в інтернеті, тому для графічного інтерфейсу потрібні такі засоби розробки:

- HTML – мова гіпертекстової розмітки;
- CSS – таблиця каскадних стилів;
- JavaScript;

HTML - це мова розмітки гіпертексту. Основним елементом у цій мові розмітки є теги. Теги являють собою парну (як правило) конструкцію, яка

потрібна для того, щоб задати певні властивості тексту або якимось іншим інформаційним даних, що знаходяться всередині цієї конструкції [13].

Каскадні таблиці стилів (Cascading Style Sheets) - це мова, яка відповідає за візуальне представлення документів користувачеві. CSS працює зі шрифтами, з кольорами символів і фону, з полями, з рядками, з висотою і з шириною елементів відображення, з фоновими зображеннями, з позиціонуванням елементів і т.д. [14].

JavaScript - мова програмування, яка дозволяє створити динамічно-оновлюваний контент, управляє мультимедіа, анімує зображення, і т.д. [15].

Сучасний JavaScript - це «безпечна» мова програмування загального призначення. Вона не надає низькорівневих засобів роботи з пам'яттю, процесором, так як спочатку була орієнтована на браузері, в яких це не потрібно.

Що ж стосується інших можливостей - вони залежать від оточення, в якому запущений JavaScript. У браузері JavaScript вміє робити все, що відноситься до маніпуляції зі сторінкою, взаємодії з відвідувачем і, в якійсь мірі, з сервером:

- створювати нові HTML-теги, видаляти існуючі, змінювати стилі елементів, ховати, показувати елементи і т.п.;
- реагувати на дії відвідувача, обробляти кліки миші, переміщення курсора, натискання на клавіатуру і т.п.;
- посилати запити на сервер і завантажувати дані без перезавантаження сторінки (ця технологія називається "AJAX");
- отримувати і встановлювати cookie, запитувати дані, виводити повідомлення.

Щодо серверної мови програмування, яка буде робити запити до бази даних, для отримання необхідних даних, то було обрано PHP.

PHP (рекурсивний акронім словосполучення PHP: Hypertext Preprocessor) - це поширена мова програмування загального призначення з відкритим вихідним кодом. PHP спеціально сконструйований для веброзробок та його код може впроваджуватися безпосередньо в HTML. Дану мову було обрано завдяки великій та зрозумілій документації, великій спільноті та великій кількості рішень для вебсервісу в Інтернеті [16].

Також доцільно використовувати систему керування вмістом, для полегшення розробки вебсервісу.

Система керування вмістом (CMS) - це набір скриптів для створення, редагування і управління контентом сайту.

Суть роботи системи керування вмістом укладена в схемі поділу вмісту сайту і його дизайну. Користувачеві надається можливість вибрати шаблон - заготовку, в якій заздалегідь визначено оформлення сторінки, і залишається тільки заповнити його потрібною інформацією. Більшість систем управління вмістом ґрунтується на використанні візуального редактора (WYSIWYG - від англ. What You See Is What You Get - «що бачиш, то і отримаєш») - програми, що дозволяє за допомогою інтуїтивно зрозумілого інтерфейсу додавати або змінювати інформацію на сайті. Варто відзначити, що сайт не складається з сукупності сторінок як такої, а формується динамічно. Доданий контент зберігається в базі даних, наприклад, MySQL, і використовується при генерації сторінки після отримання відповідного запиту з боку клієнта [17].

Для даного вебсервісу було обрано систему керування вмістом OpenCart.

OpenCart - платформа інтернет-магазину. Може бути встановлений на будь-якому вебсервері який підтримує PHP і MySQL. Не є комерційним і розповсюджується безкоштовно. Побудований OpenCart за принципом MVC і підтримує мульти-шоп. Призначена безпосередньо для створення інтернет-магазину з повноцінним набором функцій і вимагає мінімальних зусиль при установці і настройці, а також дуже простий в адмініструванні, так як не має зайвих компонентів. Код в OpenCart відповідає основним принципам паттерну Model-View-Controller, при цьому структура дуже проста, будучи відмінним каркасом для створення інтернет-магазину будь-якої складності [18].

4 РОЗРОБКА ВЕБСЕРВІСУ

4.1 Розробка бази даних

Як було описано в пункті 3.2, база даних складається з 15 таблиць:

- продукти (Products);
- користувачі (Customers);
- статуси аккаунту (Account_Statutes);
- категорії (Categories);
- коментарі до продуктів (Products Comments);
- рейтинг продуктів (Product_Rating);
- замовлення (Orders);
- способи оплати (Payments);
- способи доставки (Deliveries);
- статус замовлення (Order_Statutes);
- деталі замовлення (Order_Details);
- атрибути (Attributes);
- значення атрибутів (Attributes_Values);
- список бажань (Wish List);
- елементи списку бажань (Wish List Entries).

Для кожної з 15 таблиць були визначені типи та додаткові властивості кожного з її атрибутів. Так як опис атрибутів для сутностей «Продукти», «Категорії», «Користувачі» був проведений в пункті 3.2, то для даних сутностей були приведені лише їх структури, які наведені на рисунках 4.1 – 4.3.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Product_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Category_ID	INT	11	☐	☐	☐	Не типове
3	Product_Name	VARCHAR	255	☐	☐	☐	Не типове
4	Product_Manufacture	VARCHAR	255	☐	☐	☐	Не типове
5	Product_Description	VARCHAR	255	☐	☒	☐	NULL
6	Product_Size	FLOAT		☐	☒	☐	NULL
7	Product_Measure	VARCHAR	255	☐	☒	☐	NULL
8	Product_Price	FLOAT		☐	☐	☐	Не типове
9	Product_Stock	INT	11	☐	☒	☐	NULL
10	Product_Image	VARCHAR	255	☐	☐	☐	Не типове

Рисунок 4.1 – Структура таблиці «Продукти»

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Customer_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Customer_FName	VARCHAR	255	☐	☐	☐	Не типове
3	Customer_LName	VARCHAR	255	☐	☐	☐	Не типове
4	Customer_Email	VARCHAR	255	☐	☐	☐	Не типове
5	Customer_Phone	VARCHAR	255	☐	☐	☐	Не типове
6	Customer_Password	VARCHAR	255	☐	☐	☐	Не типове
7	Account_Status_ID	INT	11	☐	☐	☐	Не типове

Рисунок 4.2 – Структура таблиці «Користувачі»

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Category_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Category_Name	VARCHAR	255	☐	☐	☐	Не типове
3	Category_Parent_ID	INT	11	☐	☒	☐	NULL
4	Category_Description	VARCHAR	255	☐	☒	☐	NULL
5	Category_Image	VARCHAR	255	☐	☐	☐	Не типове

Рисунок 4.3 – Структура таблиці «Категорії»

Таблиця «Статуси аккаунту» має такі атрибути:

- Account_Status_ID – унікальний ідентифікатор статусу аккаунту;
- Account_Status – значення статусу аккаунту.

Структура таблиці «Статуси аккаунту» наведена на рисунку 4.4.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Account_Status_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Account_Status	VARCHAR	255	☐	☐	☐	Не типове

Рисунок 4.4 – Структури таблиці «Статуси аккаунту»

Таблиця «Коментарі до продуктів» складається з таких атрибутів:

- ProdComm_ID – унікальний ідентифікатор коментаря до продукту;
- Product_ID – ідентифікатор продукту, до якого пишеться коментар;
- Customer_ID – ідентифікатор користувача, який пише коментар;
- Comment – коментар у текстовому вигляді до продукту;
- Date_Comment – дата залишеного коментарю.

Структура таблиці «Коментарі до продуктів» наведена на рисунку 4.5.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	ProdComm_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Product_ID	INT	11	☐	☐	☐	Не типове
3	Customer_ID	INT	11	☐	☐	☐	Не типове
4	Comment	VARCHAR	255	☐	☐	☐	Не типове
5	Date_Comment	DATE		☐	☐	☐	Не типове

Рисунок 4.5 – Структура таблиці «Коментарі до продуктів»

Атрибути таблиці «Замовлення»:

- Order_ID – унікальний ідентифікатор замовлення;
- Customer_ID – ідентифікатор користувача, що зробив замовлення;
- Order_Status_ID – ідентифікатор статусу замовлення;
- Date_Order_Dispatch – поле з датою, коли було зроблено замовлення;
- Order_Details – поле з коментарем користувача, для побажань щодо замовлення;
- Payment_ID – ідентифікатор способу оплати;
- Address – адреса доставки;
- Delivery_ID – ідентифікатор способу доставки.

Структура таблиці «Замовлення» наведена на рисунку 4.6.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Order_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Customer_ID	INT	11	☐	☐	☐	Не типове
3	Order_Status_ID	INT	11	☐	☐	☐	Не типове
4	Date_Order_Dispatch	DATE		☐	☐	☐	Не типове
5	Order_Details	VARCHAR	255	☒	☒	☒	NULL
6	Payment_ID	INT	11	☐	☐	☐	Не типове
7	Address	VARCHAR	255	☐	☐	☐	Не типове
8	Delivery_ID	INT	11	☐	☐	☐	Не типове

Рисунок 4.6 – Структура таблиці «Замовлення»

Таблиця «Статуси замовлення» має такі атрибути:

- Order_Status_ID – унікальний ідентифікатор статусу замовлення;
- Order_Status – статус замовлення.

Структура таблиці «Статуси замовлення» наведена на рисунку 4.7.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Order_Status_ID	INT	11	☐	☐	☐	AUTO_INCREMENT
2	Order_Status	VARCHAR	255	☐	☐	☐	Не типове

Рисунок 4.7 – Структура таблиці «Статуси замовлення»

Таблиця «Способи оплати» складається з таких атрибутів:

- Payment_ID – унікальний ідентифікатор способу оплати;
- Payment_Name – спосіб оплати.

Структура таблиці «Способи оплати» наведена на рисунку 4.8.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Payment_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Payment_Name	VARCHAR	255	☒	☐	☒	Не типове

Рисунок 4.8 – Структура таблиці «Способи оплати»

Таблиця «Способи доставки» має аналогічні до таблиці «Способи оплати» атрибути, тільки слово «оплати» замінюється на слово «доставки». Структура таблиці «Способи доставки» наведена на рисунку 4.9.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Delivery_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Delivery_Name	VARCHAR	255	☒	☐	☒	Не типове

Рисунок 4.9 – Структура таблиці «Способи доставки»

Атрибути таблиці «Деталі замовлення»:

- Order_Details_ID – унікальний ідентифікатор деталей замовлення;
- Product_ID – ідентифікатор продукту, що входить до замовлення;
- Order_ID – ідентифікатор замовлення, до якого входить продукт;
- Product_Quantity – поле з кількістю продукту, що входить до замовлення.

Структура таблиці «Деталі замовлення» наведена на рисунку 4.10.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Order_Details_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Product_ID	INT	11	☐	☐	☐	Не типове
3	Order_ID	INT	11	☐	☐	☐	Не типове
4	Product_Quantity	INT	11	☐	☐	☐	Не типове

Рисунок 4.10 – Структура таблиці «Деталі замовлення»

Таблиця «Атрибути» має такі поля:

- Attribute_ID – унікальний ідентифікатор атрибуту;
- Attribute_Name – назва атрибуту;

Структура таблиці «Атрибути» наведена на рисунку 4.11.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Attribute_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Attribute_Name	VARCHAR	255	☒	☐	☒	Не типове

Рисунок 4.11 – Структура таблиці «Атрибути»

Таблиця «Значення атрибутів» складається з таких полів:

- AtVal_ID – унікальний ідентифікатор значення атрибуту;
- Product_ID – ідентифікатор продукту, до якого застосовується атрибут;
- Attribute_ID – ідентифікатор атрибуту, який застосовується до продукту;
- Attribute_Value – значення атрибуту.

Структура таблиці «Значення атрибутів» наведена на рисунку 4.12.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	AtVal_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Product_ID	INT	11	☐	☐	☐	Не типове
3	Attribute_ID	INT	11	☐	☐	☐	Не типове
4	Attribute_Value	VARCHAR	255	☒	☐	☒	Не типове

Рисунок 4.12 – Структура таблиці «Значення атрибутів»

Атрибути таблиці «Списки бажань»:

- Wish_List_ID – унікальний ідентифікатор списку бажань;
- Customer_ID – ідентифікатор користувача, якому належить список бажань;
- Wish_List_Name – назва списку бажань.

Структура таблиці «Списки бажань» наведена на рисунку 4.13.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Wish_List_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Customer_ID	INT	11	☐	☐	☐	Не типове
3	Wish_List_Name	VARCHAR	255	☒	☐	☒	Не типове

Рисунок 4.13 – Структура таблиці «Списки бажань»

Таблиця «Елементи списку бажань» має такі атрибути:

- Wish_List_Entry_ID – унікальний ідентифікатор елементу списку бажань;
- Product_ID – ідентифікатор продукту, що входить до списку бажань;
- Wish_List_ID – ідентифікатор списку бажань, до якого входить продукт зазначений у полі Product_ID.

Структура таблиці «Елементи списку бажань» наведена на рисунку 4.14.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Wish_List_Entry_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Product_ID	INT	11	☐	☐	☐	Не типове
3	Wish_List_ID	INT	11	☐	☐	☐	Не типове

Рисунок 4.14 – Структура таблиці «Елементи списку бажань»

Атрибути таблиці «Рейтинг продуктів»:

- Rating_Entry_ID – унікальний ідентифікатор рейтингу продукту;
- Product_ID – ідентифікатор продукту, якому користувач ставить рейтинг;
- Customer_ID – ідентифікатор користувача, який ставить рейтинг;
- Rate – значення рейтингу, який ставить користувач.

Структура таблиці «Рейтинг продуктів» наведена на рисунку 4.15.

#	Ім'я	Тип даних	Довжина/Набір	Беззнакове	Дозволити NULL	Заповнювати нулями	Типово
1	Rating_Entry_ID	INT	11	☐	☒	☐	AUTO_INCREMENT
2	Product_ID	INT	11	☐	☐	☐	Не типове
3	Customer_ID	INT	11	☐	☐	☐	Не типове
4	Rate	FLOAT		☐	☐	☐	Не типове

Рисунок 4.15 – Структура таблиці «Рейтинг продуктів»

4.2 Розробка інтерфейсу користувача

Згідно прототипів сайту, наведених у пункті 3.4 були створені сторінки інтернет-магазину: головна, сторінка з категоріями, сторінка з товарами категорії та сторінка особистого кабінету. Усі сторінки були виконані в помаранчевих кольорах, що привертає увагу користувача. Головна сторінка поділена на такі секції:

- секція швидкого доступу до вибору категорій;
- секція з акціями, що проходять в інтернет-магазині;
- 2 секції, що містять корисну інформацію користувачеві;
- секція, де користувач може залишити побажання або відгук для поліпшення роботи сайту.

Головна сторінка інтернет-магазину наведена на рисунку 4.16.



Рисунок 4.16 – Головна сторінка інтернет-магазину

Сторінка з категоріями має лише секцію з категоріями продуктів, що продаються. У цій секції кожен блок категорії, містить також посилання для переходу до підкатегорій кожної з категорій для швидкого доступу. Сторінка інтернет-магазину з категоріями наведена на рисунку 4.17.



Рисунок 4.17 – Сторінка інтернет-магазину з категоріями

Сторінка з товарами категорії має такі блоки: блок фільтрів для пошуку (зліва) та блок з самими товарами (справа). Блок фільтрів відображає тільки ті фільтри, якими можна відфільтрувати лише продукти, які безпосередньо знаходяться на сторінці.

Щодо блоку товарів, то на даній сторінці відображається коротка інформація про товари, а саме: назва, одиниці виміру, деякі атрибути та ціна. Також у даному блоці у кожного товару є дві кнопки: додати до корзини, та додати до списку бажань.

Сторінка з товарами категорії наведена на рисунку 4.18.

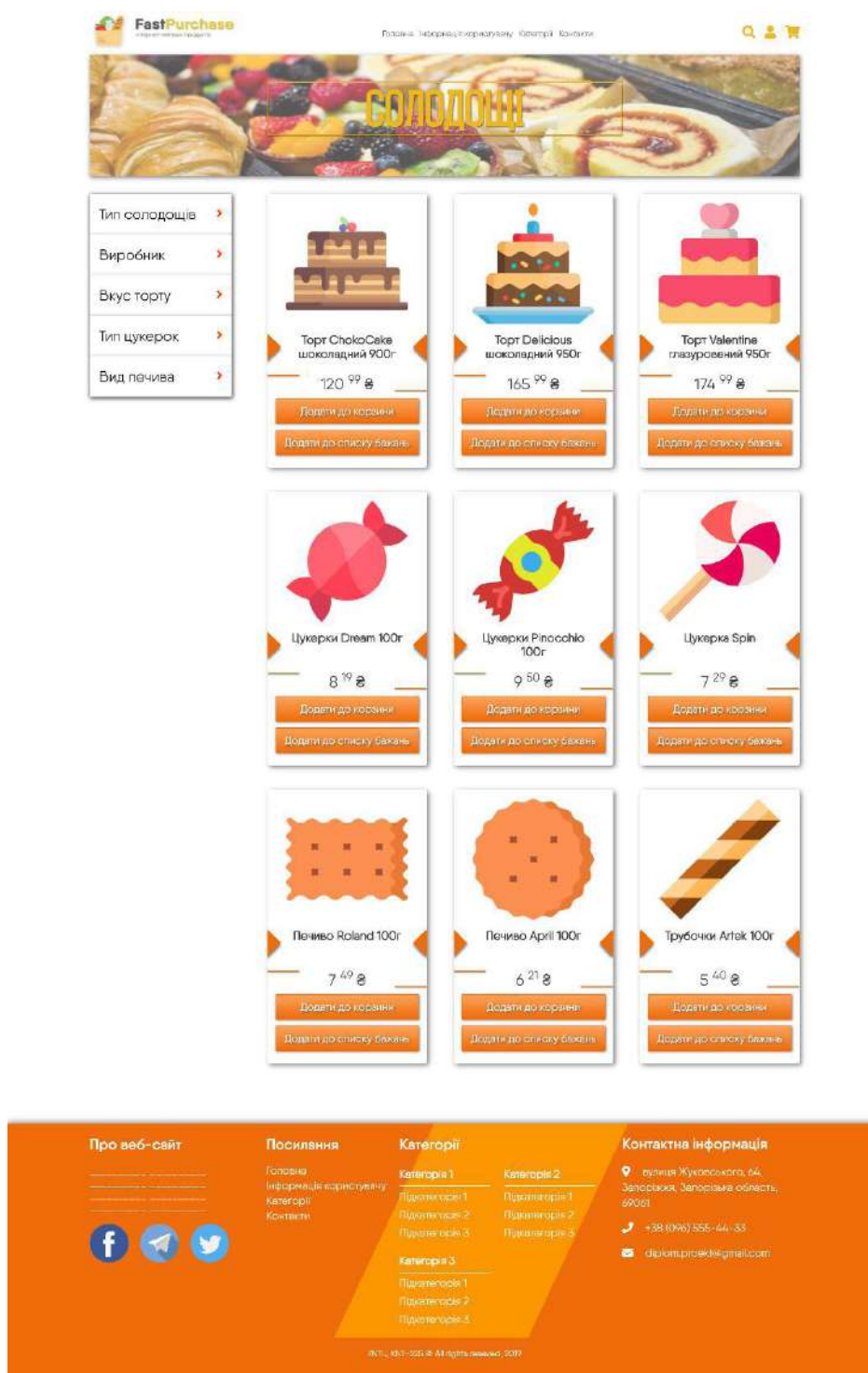


Рисунок 4.18 – Сторінка з товарами категорії

Сторінка особистого кабінету користувача складається з:

- імені та прізвища користувача, який зайшов до особистого кабінету,
- накоплених бонусів;
- кнопок, які відповідають за відображення різних блоків інформації (інформація про мене, історія замовлень, список бажань);
- блок з відображенням інформації користувача (на рисунку 4.19 для прикладу блок з історією замовлення користувача).

Сторінка особистого кабінету користувача наведена на рисунку 4.19.

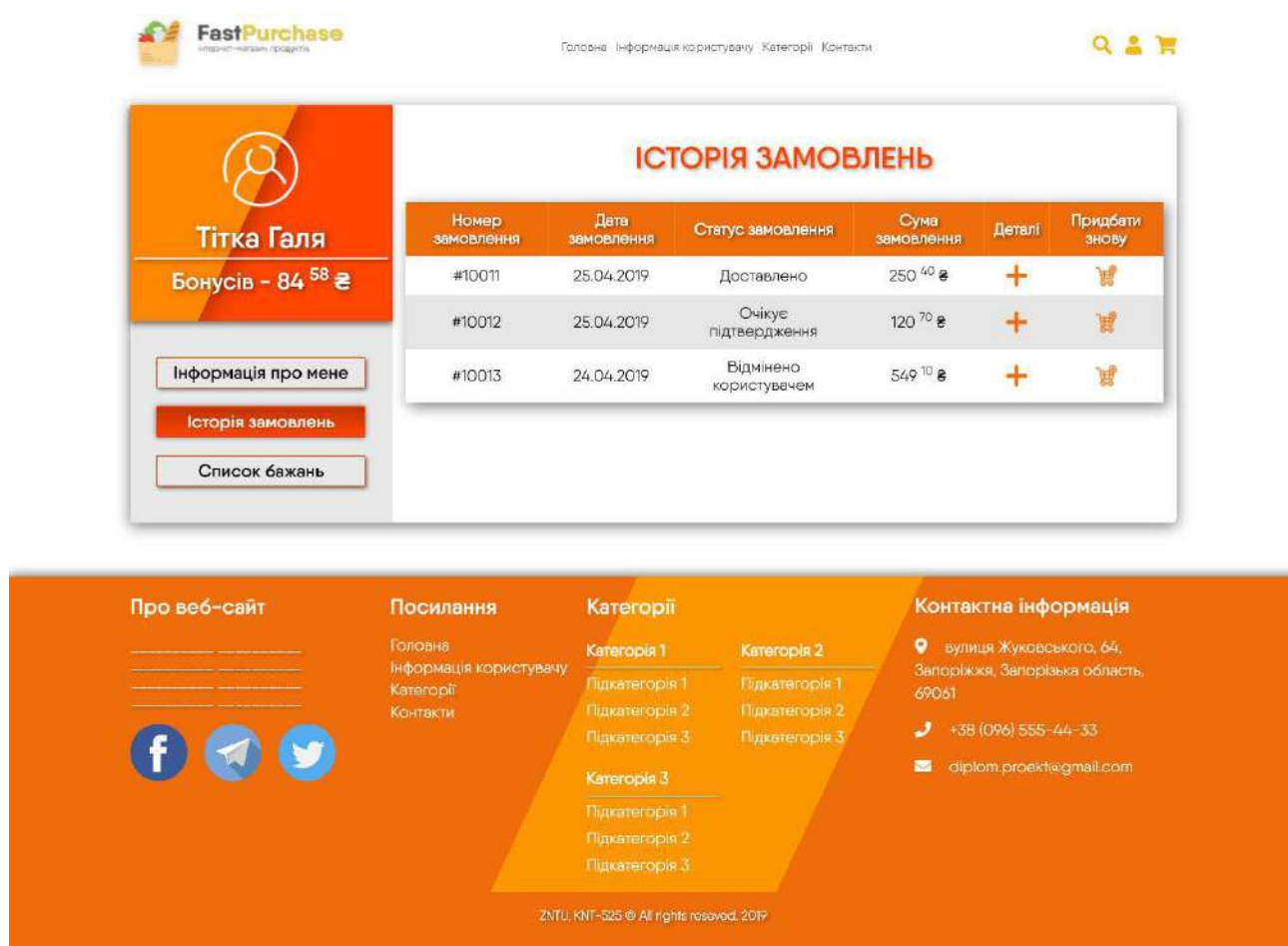


Рисунок 4.19 – Сторінка особистого кабінету користувача

Так як, користувач для того щоб замовити товар, спочатку повинен вибрати категорію, то першою функцією даного вебсервісу буде отримання с бази даних усіх категорій. На лістингу 4.1 наведений код функції, яка отримує с бази даних список категорій, та повертає їх у вигляді масиву.

Лістинг 4.1 – Функція, що повертає список категорій с бази даних у вигляді масиву

```
public static function getCategoriesList(){
    $db = Db::getConnetion();
    $categoryList = array();
    $sql_query = "SELECT Category_ID, Category_Name,
Category_Parent_ID, Category_Image FROM Category ORDER BY
Category_ID";
    $result = mysqli_query($sql_query);
    for($i = 0; $i < mysqli_num_rows($result); $i++){
        $row = mysqli_fetch_array($result, MYSQLI_ASSOC);
        if($row["Category_Parent_ID"] != NULL){
            $categoryList[$row["Category_Parent_ID"]]['subcategories'] +=
            array('id' => $row["Category_ID"], 'name' => $row["Category_Name"]);
        } else {
            $categoryList[$i]['id'] = $row["Category_ID"];
            $categoryList[$i]['name'] = $row["Category_Name"];
            $categoryList[$i]['subcategories'] = array();
            $categoryList[$i]['image_src'] = $row["Category_Image"];
        }
    }
    return $categoryList;
}
```

Тепер потрібно відобразити дані категорії на сторінці. На лістингу 4.2 приведений код, який відображає блоки з категоріями на сторінці. Результат відображення приведений на рисунку 4.20.



Рисунок 4.20 – Результат відображення категорій

Лістинг 4.2 – Відображення блоків з категоріями на сторінці

```

<div class="categories-wrapper">
  <?php foreach($categories as $categoryItem); ?>
  <div class="category__item">
    <?php echo ""; ?>
    <h4 class="category__title">
    <a href="/category/<?php echo $categoryItem['id']; ?>">
    <?php echo $categoryItem['name']; ?>
    </a>
    </h4>
    <div class="category-subcat">
    <ul class="category-subcat__list">
    <?php foreach($categories["subcategories"] as
$subcategoryItem); ?>
    <li class="categories-subcat__item">
    <a href="/category/<?php echo $subcategoryItem['id']; ?>"
class="categories-subcat__link">
    <?php echo $subcategoryItem['name']; ?>
    </a>
    </li>
    <?php endforeach; ?>
    </ul>
    </div>
    <button class="button categories-wrapper__button">
    <a href="/category/all">
    Показати усі товари категорії
    </a>
    </button>
    </div>
    <?php endforeach; ?>
  </div>

```

Після того як користувач обрав категорію, йому треба відобразити усі товари категорії. Назва товару складається з декількох атрибутів товару:

- назви зазначеної у базі даних;
- атрибуту, який відповідає на питання «який це продукт?», тобто смак, колір і т.д.;
- розмір;
- одиниці виміру.

На лістингу 4.3 приведений код, який отримує необхідні для товару з бази даних та повертає їх у вигляді масиву.

Лістинг 4.3 – Отримання необхідної інформації заданої категорії з бази даних та повернення її у вигляді масиву

```

public static function getProductListByCategory($categoryid) {
    $db = Db::getConnection();
    $categoryList = array();
    $sql      =      "SELECT      product.*,      attributes.Attribute_Name,
attributes_values.Attribute_Value
    FROM product, attributes, attributes_values, categories
    WHERE product.Category_ID = categories.Category_ID
    AND attributes_values.Product_ID = product.Product_ID
    AND attributes_values.Attribute_ID = attributes.Attribute_ID
    AND categories.Category_Parent_ID = (
    SELECT categories.Category_ID
    FROM categories
    WHERE Category_Name = ".$categoryid.")
    ORDER BY Product_ID";
    $result = mysqli_query($link, $sql);
    $productsList = array();
    $attributes_counter = 0;
    $items_counter = 0;
    $productItem = array();
    for($i = 0; $i < mysqli_num_rows($result); $i++){
        $row = mysqli_fetch_array($result, MYSQLI_ASSOC);
        if($prev_item == $row["Product_Name"]){
            $productItem["attributes"] += array($row["Attribute_Name"] =>
            $row["Attribute_Value"]);
            if($i == mysqli_num_rows($result) - 1){
                $productsList[$items_counter] = $productItem;
                $items_counter++;
            } else {
                if(($prev_item != $row["Product_Name"]) && ($i != 0)){
                    $productsList[$items_counter] = $productItem;
                    $items_counter++;
                }
                $productItem["name"] = $row['Product_Name'];
                $productItem["price"] = $row['Product_Price'];
                $productItem["size"] = $row['Product_Size'];
                $productItem["measure"] = $row['Product_Measure'];
                $productItem["stock"] = $row['Product_Stock'];
                $productItem["image_src"] = $row['Product_Image'];
                $productItem["attributes"] = array($row["Attribute_Name"] =>
                $row["Attribute_Value"]);
            }
            $prev_item = $row['Product_Name'];
        }
        return $productsList;}

```

Виведення інформації про товари відбувається в такий самий спосіб, як і виведення інформації про категорії.

Наступним кроком користувача є додавання продукту до корзини. На лістингу 4.4 приведений код, що додає товари в корзину, а також перевіряє чи не знаходяться такі самі товари вже в корзині.

Лістинг 4.4 – Функція додавання товару в корзину

```
public static function addProduct($id){
    $db = Db::getConnection();
    $productInCart = array();
    if(isset($_SESSION['products'])){
        $productInCart = $_SESSION['products'];
    }
    if(array_key_exists($id, $productInCart)){
        $productInCart[$id]++;
    } else {
        $productInCart[$id] = 1;
    }
    $_SESSION['products'] = $productInCart;
}
```

Також, для того, щоб користувач міг розуміти скільки товарів він вже придбав, необхідно створити функцію, що підраховує кількість елементів товару в корзині. В лістингу 4.5 приведений код функції, що підраховує кількість елементів в корзині.

Лістинг 4.5 – Функція, що підраховує кількість елементів у корзині

```
public static function countItems(){
    if(isset($_SESSION["products"])){
        $count = 0;
        foreach ($_SESSION["products"] as $id => $quantity) {
            $count = $count + $quantity;
        }
        return $count;
    } else {
        return 0;
    }
}
```

Останнім кроком є оформлення замовлення. Перевірка правильності заповнення форми для замовлення продукту та оформлення замовлення приведені у лістингу 4.6.

Лістинг 4.6 – Функція перевірки оформлення замовлення

```

public function actionCheckout() {
    $categories = array();
    $categories = Category::getCategoriesList();
    $result = false
    if(isset($_POST['submit'])) {
        $userFName = $_POST["userFName"];
        $userLName = $_POST["userLName"];
        $userPhone = $_POST["userPhone"];
        $userComment = $_POST["userComment"];
        $errors = false;
        if(!User::checkName($userFName) ||
!User::checkName($userLName)) $errors[] = "Ви ввели некоректне
ім'я";
        if(!User::checkPhone($userPhone)) $errors[] = "Ви ввели
некоректний телефон";
        if($errors == false){
            $productsInCart = Cart::getProducts();
            if(User::isGuest()){
                $userId = false;
            } else {
                $userId = User::checkLogged();
            }
            $result = Order::save($userFName, $userLName, $userPhone,
$userComment, $userId, $productsInCart);
            if($result) Cart::clear();
        } else {return $errors;}
        } else {
            $productsInCart = Cart::getProducts();
            if($productInCart == false){
                header("Location: /");
            } else {
                $productsIds = array_keys($productsInCart);
                $products = Product::getProductsByIds($productsIds);
                $totalProce = Cart::getTotalPrice($products);
                $totalQuantity = Cart::countItems();
                $userFName = false; $userLName = false; $userPhone = false;
                $userComment = false;
                if(User::isGuest()){
                } else {
                    $userId = User::checkLogged();
                    $user = User::getUserById($userId);
                    $userFName = $user['Customer_FName'];
                    $userLName = $user['Customer_LName'];}}}
        return true;}

```

ВИСНОВКИ

Під час виконання дипломного проєкту був спроектований вебсервіс «FastPurchase» для продажу продуктів в мережі Інтернет. Створений вебсервіс надає користувачу замовити продукти харчування не виходячи з дому.

Розроблений продукт повністю реалізує поставлені задачі, а саме:

- надання користувачу можливість знайти товар за допомогою пошуку;
- надання можливості реєстрації користувачів, для ще більшого скорочення часу на оформлення замовлення;
- забезпечення конфіденційності особистих даних користувачів сайту;
- надання консультації користувачу онлайн;
- дуже швидка робота з корзиною інтернет-магазину;
- надання можливості оформлення замовлень з вибором способів оплати та доставки;
- можливість залишити відгук, для поліпшення роботи вебсервісу.

Основними етапами створення вебсервісу були:

- аналіз технічного завдання;
- моделювання програмного забезпечення;
- проєктування вебсервісу;
- розробка вебсервісу.

В процесі проєктування вебсервісу був проведений аналіз вже існуючих аналогів, на основі яких були виділені задачі вебсервісу, зазначені вище. Також для даного вебсервісу описано три прецеденти та побудовано діаграму послідовності, що характеризують роботу вебсервісу:

- реєстрації;
- оформлення замовлення;
- повторне оформлення замовлення з особистого кабінету користувача.

Була вибрана архітектура вебсервісу – «клієнт – сервер», яка проста в налаштуванні. Також була спроектована база даних, що складається з 15 таблиць,

та задовольняє вимогам даного вебсервісу. Була наведена загальна ER-діаграма, з усіма зв'язками в базі даних. Проведено аналіз сучасних СКБД та обрано MySQL. Також були спроектовані прототипи сторінок вебсервісу, а саме: головна сторінка, сторінка з категоріями, сторінка з товарами категорії та особистий кабінет. Був проведений опис та вибір програмних засобів розробки даного вебсервісу. Наведені структури сутностей бази даних вебсервісу в СКБД MySQL. Також на основі прототипів сторінок, були створені інтерфейси сторінок в інтернеті. Були також наведені основні лістинги програмної реалізації даного вебсервісу.

Отже даний веб-сервіс готовий до використання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- 1 Аліса Вокер. Що таке веб-служби? Archітектура, типи, приклад URL: <https://www.guru99.com/uk/web-service-architecture.html> (дата звернення: 02.03.2024).
- 2 Юзабіліті інтернет-магазину: актуальність і вплив на продажі URL: <https://lemarbet.com/ua/otkrytie-internet-magazina/yuzabiliti-internet-magazina-aktualnost-i-vliyanie-na-prodazhi/> (дата звернення: 02.03.2024).
- 3 Інтернет-магазин URL: <https://uk.wikipedia.org/wiki/Інтернет-магазин> (дата звернення: 02.03.2024).
- 4 Супермаркет URL: <https://uk.wikipedia.org/wiki/Супермаркет> (дата звернення: 02.03.2024).
- 5 Ub.ua - marketplace URL: <https://ub.ua/market/> (дата звернення: 02.03.2024).
- 6 Інтернет-супермаркет продуктів харчування з доставкою додому - Produktoff Київ URL: <https://produktoff.ua/> (дата звернення: 02.03.2024).
- 7 NOVUS – інтернет-магазин URL: <https://novus.zakaz.ua/uk/> (дата звернення: 02.03.2024).
- 8 Клієнт-серверна архітектура URL: https://uk.wikipedia.org/wiki/Клієнт-серверна_архітектура (дата звернення: 10.03.2024).
- 9 Етап концептуального проектування URL: <https://studfile.net/preview/7543084/page:3/> (дата звернення: 10.03.2024).
- 10 СКБД MySQL і доступ до БД в PHP URL: <http://www.znannya.org/?view=mysql-intro> (дата звернення: 10.03.2024).
- 11 PostgreSQL URL: <https://uk.wikipedia.org/wiki/PostgreSQL> (дата звернення: 10.03.2024).
- 12 Microsoft_SQL_Server URL: https://uk.wikipedia.org/wiki/Microsoft_SQL_Server (дата звернення: 10.03.2024).

13 HTML URL: <https://uk.wikipedia.org/wiki/HTML> (дата звернення: 10.03.2024).

14 CSS URL: <https://uk.wikipedia.org/wiki/CSS> (дата звернення: 10.03.2024).

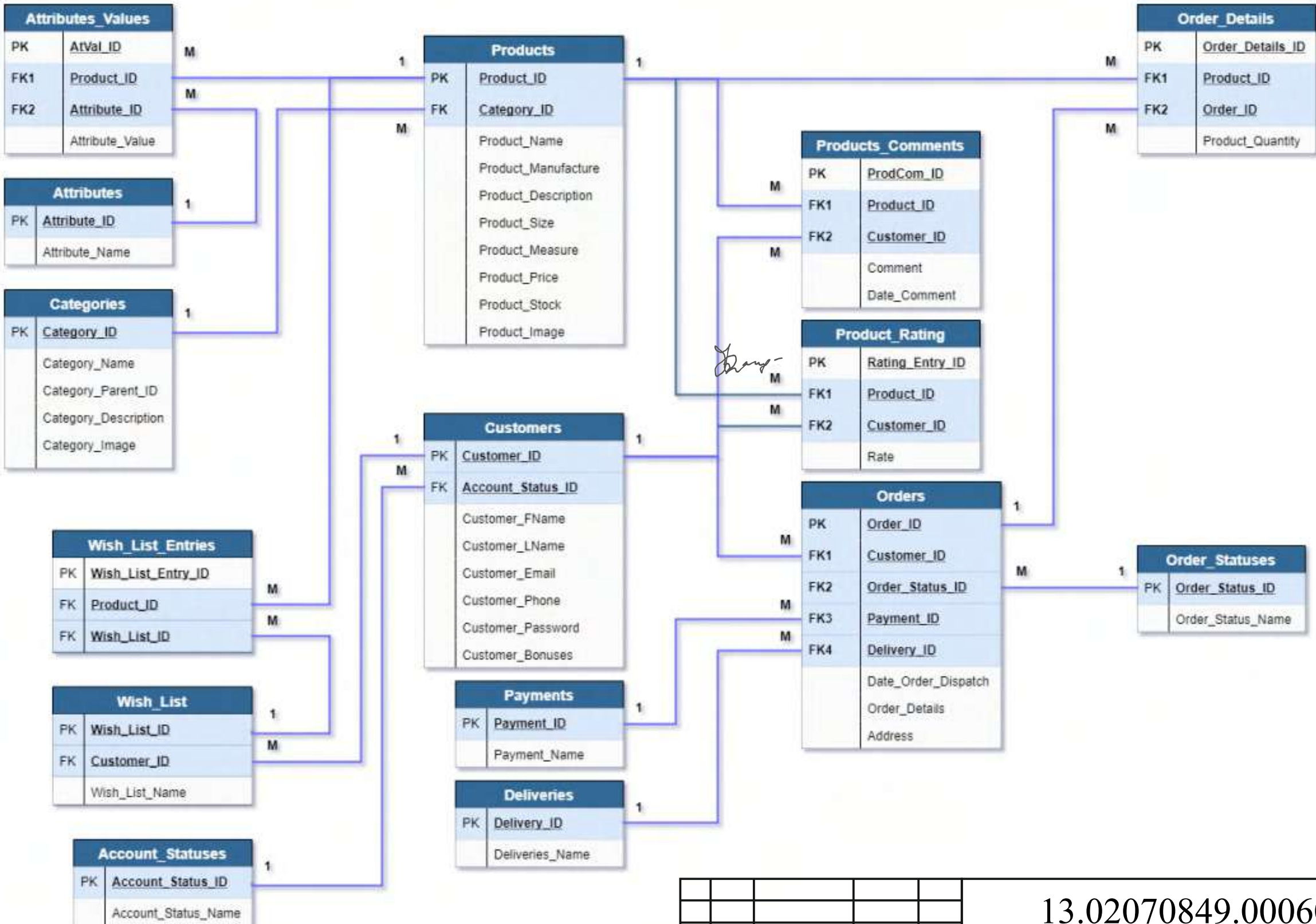
15 JavaScript URL: <https://uk.wikipedia.org/wiki/JavaScript> (дата звернення: 10.03.2024).

16 PHP URL: <https://uk.wikipedia.org/wiki/PHP> (дата звернення: 10.03.2024).

17 Що таке CMS сайту URL: <https://hostiq.ua/wiki/ukr/cms/> (дата звернення: 10.03.2024).

18 OpenCart URL: <https://uk.wikipedia.org/wiki/OpenCart> (дата звернення: 10.03.2024).

СХЕМА БАЗИ ДАНИХ



					13.02070849.00060 Пл1										
					Схема бази даних						Літ.		Маса	Масштаб	
Зм.	Лист	№ документу	Підпис	Дата											
Розробив		Зуєва К.О.													
Перевірив		Кудерметов Р.К.													
											Лист	1	Листів	1	
Н. контр.		Польська О.В.									НУ «Запорізька політехніка», група КНТЗ-510ін				
Затвердив		Кудерметов Р.К.													

Головна сторінка вебсервісу

FastPurchase

Logo

Customer InformationCatalogContactsSign upLogin

High-quality and fresh products with delivery throughout Zaporizhzhya

GO TO THE GOODS

SHARES OF WEEK

Slider

WHAT DO WE OFFER

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc maximus, nulla ut commodo sagittis, sapien dui mattis dui, non pulvinar lorem felis nec erat.

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc maximus, nulla ut commodo sagittis, sapien dui mattis dui, non pulvinar lorem felis nec erat.

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc maximus, nulla ut commodo sagittis, sapien dui mattis dui, non pulvinar lorem felis nec erat.

HOW TO ORDER GOODS

1. What to do?

2. What to do?

3. What to do?

4. What to do?

5. What to do?

6. What to do?

YOUR SUGGESTIONS

Write your suggestions for improving the site and we will take into account your suggestions

Your Email

Your Name

SEND

FOOTER

Сторінка з категоріями вебсервісу

FastPurchase

Logo

Customer InformationCatalogContactsSign upLogin

CATEGORIES

Category Name

SELECT CATEGORY

Category Name

SELECT CATEGORY

Category Name

SELECT CATEGORY

Category Name

SELECT CATEGORY

Category Name

SELECT CATEGORY

Category Name

SELECT CATEGORY

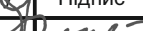
Category Name

SELECT CATEGORY

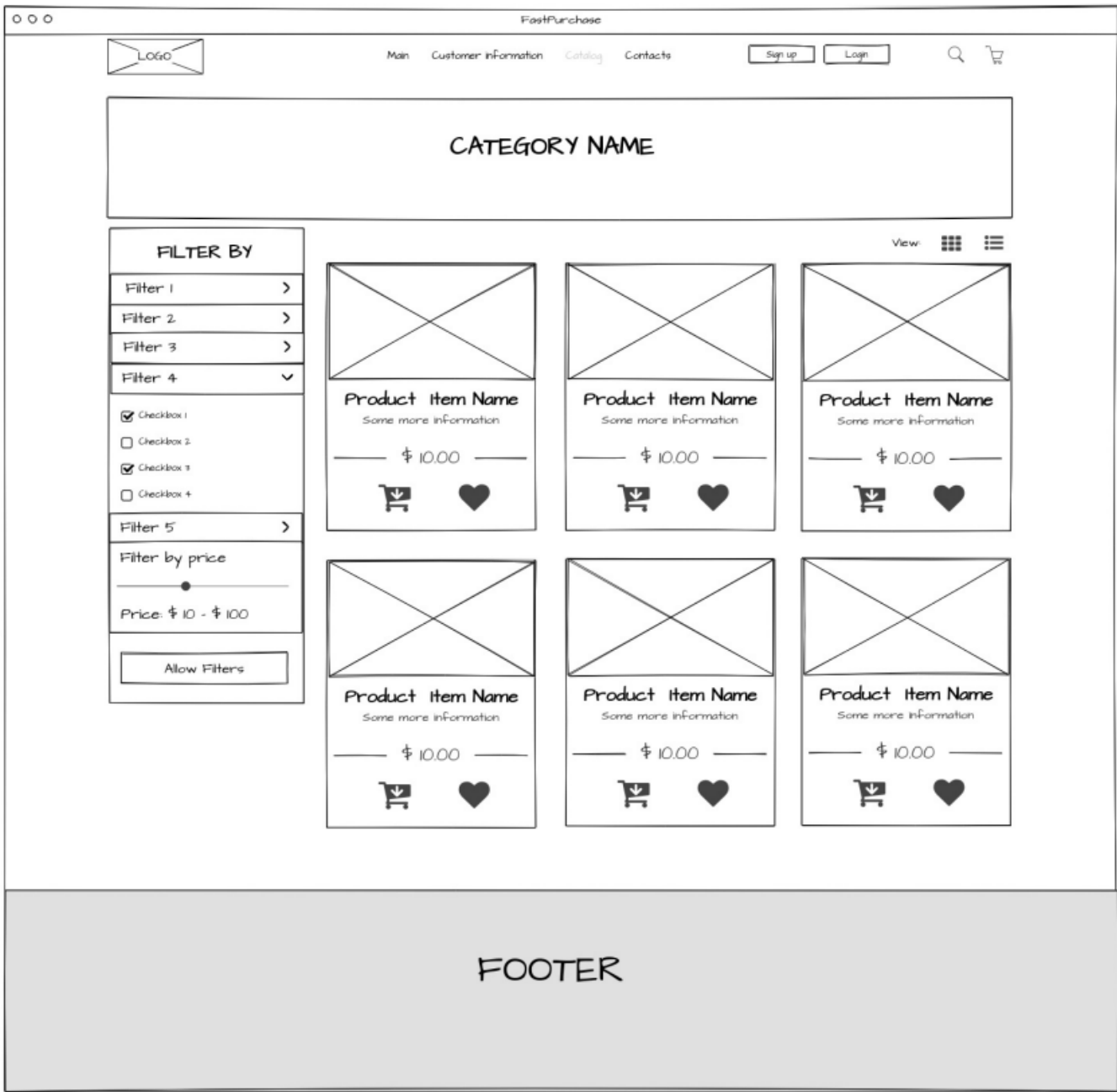
Category Name

SELECT CATEGORY

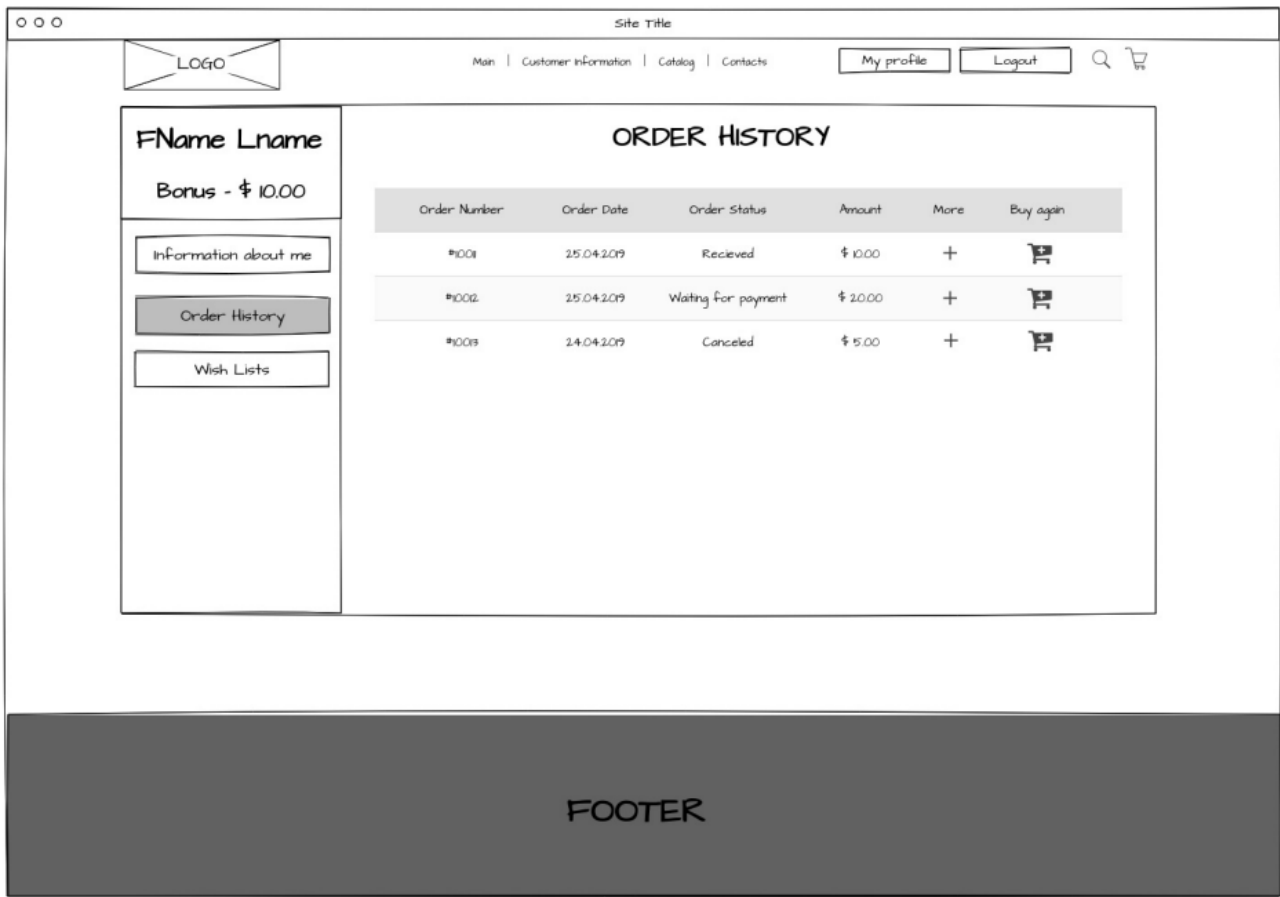
FOOTER

						13.02070849.00060 Пл2				
						Прототипи сторінок вебсервісу	Літ.		Маса	Масштаб
Зм.	Лист	№ документа	Підпис	Дата						
Розробив		Зуєва К.О.								
Перевірив		Кудерметов Р.К.								
Н. контр.		Польська О.В.					Лист 1		Листів 1	
Затвердив		Кудерметов Р.К.					НУ «Запорізька політехніка», група КНТз-510ін			

Сторінка з асортиментом товару категорії



Сторінка особистого кабінету користувача



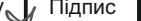
						13.02070849.00060 Пл2			
						Прототипи сторінок вебсервісу	Літ.	Маса	Масштаб
Зм.	Лист	№ документа	Підпис	Дата					
Розробив		Зуєва К.О.							
Перевірів		Кудерметов Р.К.							
							Лист	1	Листів 2
Н. контр.		Польська О.В.							
Затвердив		Кудерметов Р.К.					НУ «Запорізька політехніка», група КНТз-510ін		

Головна сторінка вебсервісу



Сторінка з категоріями вебсервісу

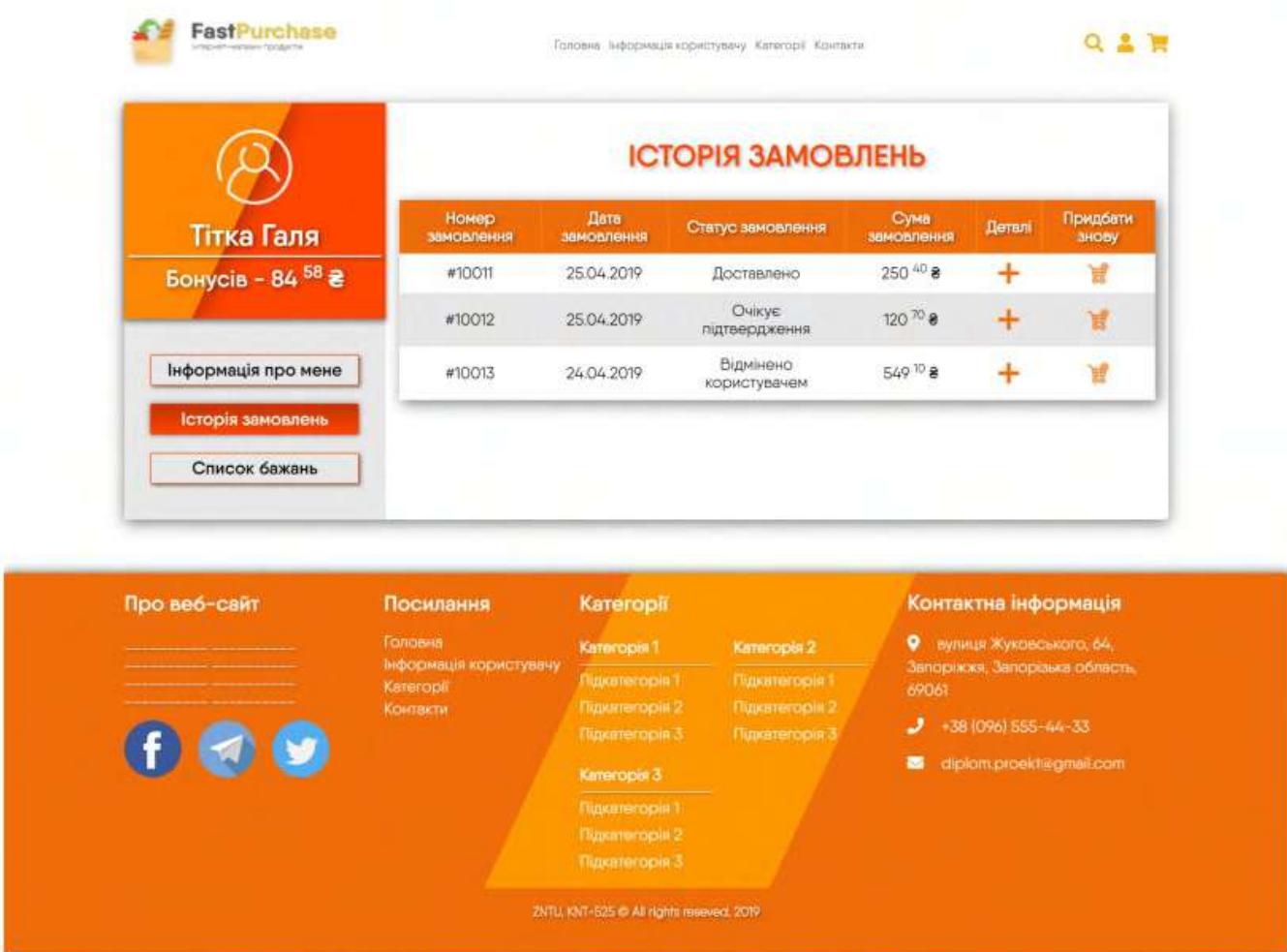


					13.02070849.00060 ПлЗ				
					Сторінки вебсервісу	Літ.		Маса	Масштаб
Зм.	Лист	№ документа	Підпис	Дата					
Розробив		Зуєва К.О.							
Перевірив		Кудерметов Р.К.							
Н. контр.		Польська О.В.				НУ «Запорізька політехніка», група КНТЗ-510ін			
Затвердив		Кудерметов Р.К.							

Сторінка з асортиментом товару категорії



Сторінка особистого кабінету користувача



					13.02070849.00060 ПЛЗ			
					Сторінки вебсервісу	Літ.	Маса	Масштаб
Зм.	Лист	№ документа	Підпис	Дата				
Розробив		Зуєва К.О.						
Перевірів		Кудерметов Р.К.				Лист 1	Листів 2	
Н. контр.		Польська О.В.				НУ «Запорізька політехніка», група КНТЗ-510ін		
Затвердив		Кудерметов Р.К.						