

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Commented [DS1]: Номери сторінок, починаючи зі сторінки вступ

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра «Системний аналіз та обчислювальна математика»

(повне найменування кафедри)

Пояснювальна записка

до дипломної роботи

бакалавр

(ступінь вищої освіти)

на тему Інформаційна система аналізу ефективності цінкових пропозицій в інтернет-магазині

(назва теми)

Виконав: студент 4 курсу, групи 813сп

Спеціальності 124 Системний аналіз

(код і найменування спеціальності)

Освітня програма (спеціалізація)

«Інтелектуальні технології та прийняття рішень в складних системах»

БЕЗСАЛИЙ Д.Ю.

(ПРІЗВИЩЕ та ініціали)

Керівник РЯБЕНКО А.Є.

(ПРІЗВИЩЕ та ініціали)

Рецензент ПИРОЖОК А.В.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут, факультет Факультет комп'ютерних наук і технологій
Кафедра «Системний аналіз та обчислювальна математика»
Ступінь вищої освіти бакалавр
Спеціальність 124 Системний аналіз
(код і найменування)
Освітня програма (спеціалізація) «Інтелектуальні технології та прийняття рішень в складних системах»
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри САОМ
Еліна ТЕРЕЩЕНКО
«08» червня 2026 року

**ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ СТУДЕНТА**

БЕЗСАЛОМУ Денису Юрійовичу

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема роботи Інформаційна система аналізу ефективності цінових пропозицій в інтернет-магазині. Information system for analyzing the effectiveness of price offers in an online store

керівник роботи РЯБЕНКО Антон Євгенович

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

затверджені наказом закладу вищої освіти від «08» червня 2026 року №480

2. Строк подання студентом роботи 5 червня 2026

3. Вихідні дані до роботи словникові дані; транзакційні дані; дані про цінові пропозиції і журнал їхнього виконання; також статистичні дані на основі продажів

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Обґрунтування актуальності проблеми; аналіз методів оптимізації; розробка односторінкового застосунку; тестування розробленого застосунку на реальних даних та формулювання висновків.

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

6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Прийняв виконане завдання
Розділ 1, Розділ 2	Рябенко А.Є., зав. кафедри	04.05.2024	08.06.2026
Нормоконтроль	Широкоград Д.В., доцент	04.05.2026	08.06.2026

7. Дата видачі завдання «04» травня 2026 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір методів класифікації та візуалізації.	1 тиждень	Розділ 1
4	Робота з Excel файлами та базою даних.	2 тиждень	Розділ 2
5	Реалізація алгоритмів та методів.	3-6 тиждень	Розділ 2
6	Реалізація візуалізації даних.	6-8 тиждень	Розділ 2
7	Аналіз на основі візуалізації	9 тиждень	Розділ 2
8	Оформлення пояснювальної записки та відповідної документації.	10 тиждень	Додатки
9	Нормоконтроль та рецензування.	11 тиждень	
10	Захист дипломної роботи.	12 тиждень	

Студент

_____ БЕЗСАЛИЙ Д.Ю.
(підпис) (прізвище та ініціали)

Керівник роботи

_____ РЯБЕНКО А.Є.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Дипломна робота: 63 с., 12 рис., 25 джерел, 1 додаток.

Об'єкт дослідження – процес управління ціновими пропозиціями в інтернет-магазині.

Предмет дослідження – методи, алгоритми та програмні засоби аналізу ефективності цінових пропозицій в інформаційних системах електронної комерції.

Мета роботи – проектування та розробка інформаційної системи аналізу ефективності цінових пропозицій в інтернет-магазині, яка забезпечує автоматизований збір, обробку та аналітичне відображення даних про продажі, маржинальність і динаміку цін у зручному для менеджерів форматі.

Методи дослідження – методи системного аналізу і декомпозиції задачі, об'єктно-орієнтованого проектування та моделювання, математичної статистики для аналізу часових рядів продажів, RESTful-архітектури при проектуванні програмного інтерфейсу.

Актуальність проблеми – зумовлена стрімким розвитком електронної комерції та зростаючою конкуренцією в онлайн-середовищі. Проблема полягає в тому, що ручний аналіз цінових пропозицій є надзвичайно трудомістким і не дозволяє своєчасно реагувати на ринкові зміни. Менеджери змушені вручну опрацьовувати великі масиви даних, що призводить до затримок у прийнятті рішень та значних часових витрат — до 8–12 годин на тиждень.

Основні висновки – у результаті виконання роботи було розроблено та протестовано інформаційну систему аналізу ефективності цінових пропозицій інтернет-магазину, яка забезпечує автоматизацію аналітичних процесів, високу точність результатів і значне скорочення часу аналізу порівняно з ручними методами.

ІНФОРМАЦІЙНА СИСТЕМА, ІНТЕРНЕТ-МАГАЗИН, ЦІНОВІ ПРОПОЗИЦІЇ

ЗМІСТ

Commented [DS2]: Додати завдання, реферат

ЗАВДАННЯ.....	2
РЕФЕРАТ	4
ВСТУП.....	6
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ АНАЛІЗУ ЕФЕКТИВНОСТІ ЦІНОВИХ ПРОПОЗИЦІЙ В ІНТЕРНЕТ-МАГАЗИНАХ	8
1.1. Поняття електронної комерції та особливості функціонування інтернет- магазинів.....	8
1.2. Цінова політика та фактори формування цінових пропозицій	13
1.3. Методи оцінювання ефективності цінових пропозицій	17
РОЗДІЛ 2 ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	22
2.1. Аналіз вимог до інформаційної системи та постановка задачі.....	22
2.2. Моделювання структури системи та бази даних.....	26
2.3. Розробка алгоритмів аналізу ефективності цінових пропозицій	31
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ТА ОЦІНЮВАННЯ РОБОТИ СИСТЕМИ	48
3.1. Програмна реалізація інформаційної системи.....	48
3.2. Тестування та валідація результатів аналізу.....	53
3.3. Оцінка ефективності використання системи в інтернет-магазині.....	56
ВИСНОВКИ	61
ПЕРЕЛІК ПОСИЛАНЬ	63
Додаток А. Лістинг коду.....	66

ВСТУП

Розвиток глобальної цифрової економіки суттєво змінив підходи до організації торговельної діяльності. Електронна комерція стала одним із найдинамічніших секторів сучасної економіки, демонструючи стабільне зростання навіть в умовах економічної нестабільності. Інтернет-магазини як основна форма реалізації моделі «бізнес-споживач» надають підприємствам можливість виходити на глобальні ринки, залучати нових клієнтів та оптимізувати операційні витрати, суттєво знижуючи бар'єри входу порівняно з традиційною офлайн-торгівлею.

Цінова політика залишається одним із ключових інструментів конкурентної боротьби в онлайн-середовищі. На відміну від традиційної роздрібною торгівлі, де зміна цінників є трудомістким фізичним процесом, електронна комерція дозволяє здійснювати динамічне ціноутворення практично в режимі реального часу. Зростаюча складність ринкових умов, великий асортимент товарних позицій та постійна зміна попиту ставлять перед підприємствами завдання систематичного відстеження ефективності цінових рішень і швидкого реагування на ринкові зміни.

Ручний аналіз цінових пропозицій у магазинах з широким асортиментом є надзвичайно трудомістким і не дозволяє своєчасно реагувати на зміни ринкової кон'юнктури. Менеджери з ціноутворення змушені опрацьовувати значні масиви даних про продажі, маржинальність та поведінку конкурентів, щоб ухвалювати обґрунтовані рішення щодо коригування цін. У цьому контексті розробка спеціалізованих інформаційних систем для автоматизації збору, обробки й аналізу даних про ефективність цінових пропозицій набуває особливої актуальності.

Актуальність теми дослідження зумовлена зростаючою потребою підприємств електронної комерції в інструментах, що дозволяють комплексно

оцінювати вплив цінових змін на ключові показники ефективності бізнесу: виручку, маржинальність, конверсію та середній чек. Відсутність спеціалізованих аналітичних рішень, адаптованих до специфіки вітчизняного ринку електронної торгівлі, додатково підкреслює необхідність проведення даного дослідження.

Метою кваліфікаційної роботи є проектування та розробка інформаційної системи аналізу ефективності цінових пропозицій в інтернет-магазині, яка забезпечує автоматизований збір, обробку та аналітичне відображення даних про продажі, маржинальність і динаміку цін у зручному для менеджерів форматі.

Для досягнення поставленої мети визначено такі завдання: дослідити теоретичні основи функціонування електронної комерції та особливості цінової політики онлайн-торгівлі; проаналізувати сучасні методи оцінювання ефективності цінових рішень; визначити функціональні та нефункціональні вимоги до інформаційної системи й виконати постановку задачі; спроектувати архітектуру системи, структуру бази даних та програмний інтерфейс; розробити алгоритми аналізу ефективності цінових пропозицій і динаміки продажів; реалізувати програмне забезпечення з повнофункціональним користувацьким інтерфейсом; провести тестування та валідацію роботи системи й оцінити ефективність її впровадження.

Об'єктом дослідження є процес управління ціновими пропозиціями в інтернет-магазині. Предметом дослідження є методи, алгоритми та програмні засоби аналізу ефективності цінових пропозицій в інформаційних системах електронної комерції.

У процесі виконання роботи застосовувалися методи системного аналізу і декомпозиції задачі, об'єктно-орієнтованого проектування та моделювання, математичної статистики для аналізу часових рядів продажів, RESTful-архітектури при проектуванні програмного інтерфейсу, а також функціонального та регресійного тестування для перевірки коректності роботи системи.

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ АНАЛІЗУ ЕФЕКТИВНОСТІ ЦІНОВИХ ПРОПОЗИЦІЙ В ІНТЕРНЕТ-МАГАЗИНАХ

1.1 Поняття електронної комерції та особливості функціонування інтернет-магазинів

Commented [DS3]: Без крапки після номеру

Commented [DS4]: Між заголовком і текстом 2 порожні рядки
Між двома заголовками без порожніх рядків

Електронна комерція (e-commerce) являє собою форму господарської діяльності із продажу товарів і послуг із використанням інформаційно-комунікаційних технологій, передусім глобальної мережі Інтернет. За визначенням, запропонованим Turban et al., електронна комерція охоплює будь-які транзакції, що здійснюються через електронні мережі, включаючи купівлю та продаж товарів і послуг, передачу коштів або даних [1]. У широкому розумінні до цього поняття відносять не лише власне торговельні операції, але й усю сукупність ділових процесів, що забезпечують їх виконання: маркетинг, логістику, клієнтський сервіс та управління взаємовідносинами з клієнтами.

Розвиток електронної комерції нерозривно пов'язаний з еволюцією інформаційних технологій. Починаючи з 1990-х років, коли з'явилися перші комерційні веб-сайти, галузь пройшла шлях від простих онлайн-каталогів до складних торговельних екосистем, що інтегрують штучний інтелект, аналітику великих даних і технології персоналізації. Обсяг глобального ринку e-commerce у 2023 році перевищив 5,7 трильйона доларів США, а до 2027 року прогнозується його зростання до понад 8 трильйонів доларів, що свідчить про стійкість галузі та її стратегічне значення для сучасної економіки [2].

Класифікація моделей електронної комерції передбачає виділення декількох основних типів взаємодії суб'єктів. Модель «бізнес-споживач» (B2C) передбачає реалізацію товарів і послуг компаніями безпосередньо кінцевим споживачам через інтернет-магазини. Модель «бізнес-бізнес» (B2B) охоплює комерційні відносини між підприємствами, включаючи оптові закупівлі, субпідряд та міжкорпоративні послуги. Модель «споживач-споживач» (C2C) реалізується на майданчиках оголошень і аукціонів. Для цілей даної роботи

найбільший інтерес становить модель B2C, в рамках якої функціонують роздрібні інтернет-магазини [2].

Інтернет-магазин як організаційно-технічна форма електронної комерції являє собою веб-сайт або мобільний додаток, що забезпечує повний цикл торговельних операцій: представлення товарного асортименту, оформлення замовлення, здійснення оплати та організацію доставки. Особливістю сучасних інтернет-магазинів є їхня здатність функціонувати цілодобово без перерви, обслуговувати необмежену кількість покупців одночасно та забезпечувати персоналізований підхід на основі аналізу поведінки користувачів. Порівняно з традиційними форматами торгівлі, онлайн-магазини демонструють значно нижчі операційні витрати, ширший географічний охопит і вищу швидкість впровадження змін у товарну пропозицію та цінову політику.

Дослідження Врунґольфссон та Smith, що стало класичною роботою в галузі порівняльного аналізу онлайн- та офлайн-торгівлі, показало, що інтернет-магазини здатні пропонувати ціни в середньому на 9–16% нижчі, ніж традиційні роздрібні торговці, передусім завдяки зниженню транзакційних витрат та меншим накладним витратам [3]. Водночас автори зафіксували значно більшу цінову дисперсію в онлайн-каналі порівняно з офлайн-торгівлею, що пояснюється нижчими витратами на зміну цін та різноманітністю цінових стратегій учасників ринку. Ця особливість є принципово важливою для розуміння природи цінової конкуренції в електронній комерції.

Функціональна архітектура сучасного інтернет-магазину включає кілька ключових компонентів: систему управління контентом і каталогом товарів, підсистему обробки замовлень, платіжний шлюз, модуль управління запасами, аналітичну платформу та засоби комунікації з клієнтами. Інтеграція цих компонентів в єдину інформаційну систему дозволяє автоматизувати більшість операційних процесів і зосередити зусилля менеджменту на стратегічних завданнях [4]. Зростання складності операцій і обсягів даних зумовлює потребу у спеціалізованих підсистемах аналітики, здатних трансформувати первинні дані про транзакції у змістовну управлінську інформацію.

На сьогоднішній день ринок е-комерційних рішень пропонує широкий спектр платформ і технологій: від готових хмарних рішень (Shopify, BigCommerce) до відкритих систем управління контентом (WooCommerce, Magento) та індивідуальних розробок. Вибір технологічного стека залежить від масштабу бізнесу, специфіки асортименту, обсягів трафіку та бюджетних обмежень. Незалежно від обраної платформи, ключовим чинником успіху є наявність ефективних інструментів аналізу даних, що дозволяють приймати обґрунтовані управлінські рішення [5].



Рисунок 1.1 – Класифікація моделей електронної комерції та місце інтернет-магазину в структурі B2C-сегменту

Рисунок 1.1 ілюструє класифікацію основних моделей електронної комерції та взаємозв'язки між ними. Як видно з рисунку, модель B2C є центральною для роздрібно́ї онлайн-торгівлі і передбачає найширший спектр операційних процесів, безпосередньо пов'язаних з управлінням ціновими пропозиціями.

Поведінка споживачів в онлайн-середовищі суттєво відрізняється від офлайн-покупок. Дослідження показують, що онлайн-покупці в цілому більш чутливі до цін і активно використовують можливості порівняння цін між різними продавцями [6]. Водночас чинники зручності, швидкості доставки і репутації магазину здатні суттєво послаблювати цінову чутливість покупців і формувати

лояльність попри вищу ціну. Ця двоїстість споживацької поведінки ставить перед менеджерами складне завдання балансування між ціновою конкурентоспроможністю і збереженням маржинальності.

Важливою характеристикою сучасних інтернет-магазинів є їхня здатність генерувати і накопичувати великі масиви даних про взаємодію користувачів з платформою. Кожне відвідування сторінки, пошуковий запит, додавання товару до кошика та фінальна транзакція залишають детальний цифровий слід, що формує цінну інформаційну базу для аналізу. За оцінками, типовий інтернет-магазин генерує десятки тисяч подій взаємодії щодоби, більшість яких залишається невикористаною через відсутність відповідних аналітичних інструментів [7]. Це свідчить про значний потенціал підвищення ефективності бізнесу за рахунок впровадження систем аналітики.

Специфіка українського ринку електронної комерції визначається рядом особливостей. За даними вітчизняних досліджень, обсяг ринку e-commerce у 2022–2023 роках демонстрував стійке зростання незважаючи на несприятливі зовнішні умови [8]. Провідні сегменти ринку включають електроніку і побутову техніку, одяг і взуття, товари для дому та будівництва. Особливістю вітчизняного ринку є значна роль маркетплейсів як посередників між продавцями і покупцями, а також висока частка мобільної комерції, що перевищує 60% від усіх онлайн-транзакцій.

До ключових показників ефективності інтернет-магазину, безпосередньо пов'язаних з ціновою політикою, відносять: конверсію відвідувачів у покупців, середній чек замовлення, рентабельність продажів (маржу), частоту повторних покупок та довічну цінність клієнта (Customer Lifetime Value). Моніторинг цих показників у динаміці дозволяє оцінити вплив цінових рішень на загальну ефективність бізнесу і приймати обґрунтовані рішення щодо коригування цінової стратегії [9].

Мобільна комерція (m-commerce) є одним із найбільш динамічно зростаючих підсегментів електронної торгівлі. Смартфони і планшети стають домінуючим пристроєм для здійснення онлайн-покупок, що суттєво змінює

вимоги до дизайну торговельних платформ і підходів до ціноутворення. Дослідження свідчать, що мобільні покупці характеризуються вищою імпульсивністю покупок, але водночас більш критично ставляться до зручності оформлення замовлення. Це спонукає інтернет-магазини розробляти специфічні цінові пропозиції для мобільного каналу, адаптовані до особливостей поведінки його аудиторії.

Персоналізація товарних пропозицій і цін є одним із найбільш перспективних напрямів розвитку сучасних інтернет-магазинів. Технології машинного навчання дозволяють аналізувати індивідуальні переваги покупців, їхню купівельну історію і поведінку на сайті для формування унікальних рекомендацій і цінових пропозицій. Впровадження персоналізованого ціноутворення потребує надійних інформаційних систем, здатних у режимі реального часу обробляти великі обсяги даних і формувати відповідні рекомендації для менеджерів або автоматично застосовувати оптимальні цінові рішення.

Підсумовуючи розгляд поняття електронної комерції та особливостей функціонування інтернет-магазинів, слід зазначити, що цифровий характер торгових операцій створює унікальні можливості для збору і аналізу даних про ефективність цінових рішень. Водночас динамічність онлайн-ринку і постійна зміна конкурентного середовища обумовлюють потребу в автоматизованих інструментах аналізу, здатних забезпечити оперативне реагування на зміни ринкової кон'юнктури.

1.2. Цінова політика та фактори формування цінових пропозицій

Цінова політика підприємства є сукупністю принципів і методів, якими керується організація при встановленні цін на свої товари і послуги. В умовах електронної комерції формування ефективної цінової політики набуває особливого значення, оскільки ціна є одним із найбільш видимих і легко порівнюваних атрибутів товарної пропозиції в онлайн-середовищі. Покупець може миттєво порівняти ціни десятків продавців за допомогою агрегаторів цін або пошукових систем, що суттєво підвищує прозорість ринку і ступінь цінової конкуренції [10].

У теорії ціноутворення виділяють кілька основних стратегій, що застосовуються в електронній комерції. Стратегія ціноутворення на основі витрат (*cost-plus pricing*) передбачає встановлення ціни шляхом додавання бажаної маржі до собівартості товару. Ця стратегія є найпростішою у застосуванні та гарантує покриття витрат, однак не враховує цінову чутливість попиту і конкурентне оточення. Стратегія конкурентного ціноутворення (*competitive pricing*) орієнтується на ціни конкурентів як на орієнтир при встановленні власних цін. Стратегія ціноутворення на основі цінності (*value-based pricing*) спрямована на встановлення ціни, що відповідає сприйнятій покупцем цінності продукту [3].

Динамічне ціноутворення (*dynamic pricing*) є найбільш поширеною практикою у великих інтернет-магазинах і маркетплейсах. Ця стратегія передбачає автоматичне коригування цін у відповідь на зміни попиту, пропозиції, цін конкурентів та інших ринкових чинників у режимі реального часу. Дослідження Kauffman та Wang показало, що динамічне ціноутворення здатне збільшувати виручку компаній до 25% порівняно зі статичними стратегіями при правильному моделюванні поведінки покупців [7]. Водночас надмірно агресивне застосування динамічного ціноутворення може підірвати довіру покупців до бренду.

До основних факторів, що визначають формування цінових пропозицій в інтернет-магазині, належать чинники двох категорій: внутрішні і зовнішні. Серед внутрішніх факторів ключову роль відіграють структура витрат підприємства (собівартість товарів, логістичні витрати, витрати на маркетинг та обслуговування платформи), цільова маржинальність за категоріями товарів, стратегічні пріоритети і позиціонування бренду [24]. Внутрішні фактори формують нижню межу прийнятної ціни і визначають операційну рентабельність кожної цінової пропозиції.

Зовнішні фактори ціноутворення включають рівень і еластичність попиту, ціни конкурентів, загальний стан ринку і макроекономічне середовище, сезонність попиту, наявність товарів-субститутів та комплементарних товарів. Особливу роль у онлайн-середовищі відіграє прозорість цін: покупці мають можливість миттєво порівнювати пропозиції різних продавців, що формує підвищений тиск на ціну як конкурентний параметр [20]. Водночас репутаційні чинники, якість клієнтського сервісу і надійність доставки здатні суттєво знижувати цінову чутливість лояльних покупців.

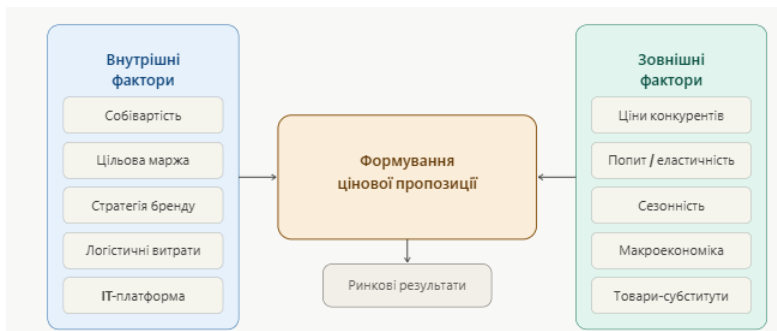


Рисунок 1.2 – Фактори формування цінових пропозицій в інтернет-магазині: системна модель взаємодії внутрішніх та зовнішніх чинників

На рисунку 1.2 представлено системну модель взаємодії внутрішніх та зовнішніх факторів при формуванні цінових пропозицій в інтернет-магазині. Центральне місце в моделі займає процес формування цінової пропозиції, що

знає впливу з боку обох груп факторів і, у свою чергу, впливає на ринкові результати діяльності підприємства.

Цінова еластичність попиту є одним із ключових параметрів, що визначають оптимальний рівень ціни. При значенні коефіцієнта еластичності за абсолютною величиною більше одиниці попит вважається еластичним, що означає значну чутливість покупців до цінових змін. У цьому випадку зниження ціни призводить до пропорційно більшого зростання обсягу продажів, що може збільшити загальну виручку. При нееластичному попиті цінова чутливість є нижчою, що відкриває можливості для встановлення преміальних цін без суттєвої втрати обсягів збуту [21].

Дослідження Hamilton та Srivastava присвячено психологічним аспектам сприйняття цін покупцями в онлайн-середовищі. Автори показали, що покупці по-різному реагують на однакові за розміром знижки залежно від способу їх представлення: у відсотках або абсолютних значеннях. Для товарів з відносно низькою ціною ефективнішим є представлення знижки у відсотках, тоді як для дорогих товарів більш переконливою виглядає абсолютна сума економії [23]. Ці висновки мають безпосереднє практичне значення для формування цінових пропозицій і розробки систем управління знижками.

Знижкова і промоційна цінова політика відіграє важливу роль у стимулюванні попиту в інтернет-магазинах. Обмежені в часі акції, розпродажі і бандлові пропозиції є ефективними інструментами залучення нових покупців і стимулювання повторних покупок серед існуючої аудиторії [16]. Водночас надмірне захоплення знижковою торгівлею несе ризик ерозії брендового іміджу і «привчання» покупців чекати знижок замість придбання товарів за повною ціною.

Сезонність є важливим чинником цінової політики в більшості категорій товарів. Попит на електроніку традиційно зростає перед новорічними святами, на спортивний інвентар – у відповідні сезони, на кліматичне обладнання – залежно від погодних умов. Врахування сезонних коливань при формуванні цінових пропозицій дозволяє максимізувати виручку в пікові періоди і

підтримувати необхідний рівень обороту в міжсезоння [15]. Інформаційна система аналізу цінових пропозицій має забезпечувати виявлення і візуалізацію сезонних патернів у даних продажів.

Дискримінація цін (price discrimination) є широко поширеною практикою в електронній комерції і передбачає встановлення різних цін для різних сегментів покупців залежно від їхньої готовності платити, географічного розташування або попередньої купівельної поведінки. Varian у своїй фундаментальній роботі довів, що цінова дискримінація за умови правильного сегментування ринку може підвищувати загальну ефективність розподілу ресурсів [21]. В онлайн-торгівлі це проявляється у формі персоналізованих знижок, програм лояльності і динамічних купонів.

Категорійний підхід до управління ціноутворенням передбачає розробку різних стратегій для різних товарних категорій з урахуванням їхньої стратегічної ролі в портфелі магазину. Товари-«локомотиви» (traffic drivers) продаються за мінімальною маржею для залучення відвідувачів. Товари-«дійні корови» (cash cows) пропонуються з підвищеною маржею завдяки низькій ціновій чутливості попиту. Аналіз рентабельності на рівні категорій є базовою функцією системи управління ціновими пропозиціями [11].

Вплив цінових рішень на фінансові результати підприємства є прямим і вимірюваним. Дослідження McKinsey показали, що підвищення середньої ціни реалізації всього на 1% при незмінних обсягах продажів збільшує операційний прибуток типової компанії на 8% – значно більше, ніж аналогічне покращення в управлінні витратами або обсягом продажів [12]. Це свідчить про те, що управління ціноутворенням є одним із найбільш потужних важелів впливу на рентабельність бізнесу, що, у свою чергу, обумовлює стратегічну важливість аналізу ефективності цінових рішень.

Управління ціноутворенням у мультिकанальних продажах потребує особливого підходу, оскільки різні канали збуту (власний сайт, маркетплейси, мобільний додаток) характеризуються різними профілями покупців і конкурентним оточенням. Ціни на одному каналі можуть впливати на

сприйняття і попит на інших каналах, що ускладнює завдання оптимізації цінової пропозиції. Успішна мультिकанальна цінова стратегія вимагає чіткого розуміння цінових орієнтирів і правил диференціації між каналами з урахуванням їхньої специфіки.

Таким чином, цінова політика в інтернет-магазині є багатофакторним процесом, що визначається взаємодією внутрішніх операційних параметрів і зовнішніх ринкових умов. Ефективне управління ціноутворенням вимагає систематичного збору та аналізу даних, що обумовлює необхідність спеціалізованих інформаційних інструментів.

1.3. Методи оцінювання ефективності цінових пропозицій

Оцінювання ефективності цінових пропозицій є необхідною умовою обґрунтованого управління ціноутворенням в інтернет-магазині. Під ефективністю цінової пропозиції розуміється її здатність забезпечувати досягнення поставлених цілей, насамперед максимізацію виручки і рентабельності за прийняттого рівня конкурентоспроможності. Методологія оцінювання включає кількісні і якісні підходи, що разом забезпечують комплексне розуміння результатів цінових рішень [3].

Система ключових показників ефективності (KPI) є базовою основою для оцінювання результатів цінових рішень. До основних KPI цінової ефективності відносяться: виручка від реалізації в розрізі часових інтервалів, категорій і каналів збуту; маржинальна рентабельність у відсотках від виручки; середній чек замовлення; конверсія відвідувань у покупки; частка постійних покупців у загальній кількості транзакцій; оборотність товарних запасів. Моніторинг цих показників у динаміці дозволяє виявляти тенденції і відхилення, пов'язані зі змінами цінових пропозицій [9].

Метод порівняльного аналізу (A/B тестування) є одним із найефективніших способів оцінювання наслідків цінових рішень. Суть методу полягає у розділенні аудиторії покупців на дві або більше групи, кожній із яких пропонується різна цінова пропозиція, після чого порівнюються ключові показники ефективності між групами. A/B тестування дозволяє ізолювати вплив саме цінового фактора від інших змінних [8]. Водночас метод вимагає достатнього обсягу трафіку для досягнення статистичної значущості результатів.



Рисунок 1.3 – Класифікація методів оцінювання ефективності цінових пропозицій в інтернет-магазині

На рисунку 1.3 представлено класифікацію основних методів оцінювання ефективності цінових пропозицій. Методи розподілено за двома критеріями: характером даних, що аналізуються (ретроспективні та проспективні), і ступенем аналітичної складності (від базових КРІ-дашбордів до методів машинного навчання).

Регресійний аналіз є математичним методом, що дозволяє кількісно оцінити зв'язок між ціною товару і обсягом попиту на нього. Побудова регресійної моделі «ціна-попит» дає можливість розрахувати оптимальну ціну, що максимізує виручку або прибуток, а також прогнозувати зміни обсягів продажів при різних цінових сценаріях. Зазвичай застосовується лінійна або логарифмічна специфікація моделі, яка вибирається з урахуванням характеру залежності між змінними [18].

Аналіз еластичності цін є конкретизацією регресійного аналізу для задачі вимірювання цінової чутливості попиту. Коефіцієнт цінової еластичності розраховується як відношення відсоткової зміни обсягу продажів до відсоткової зміни ціни. Для кожної категорії товарів і ринкового сегменту характерні різні значення еластичності, що потребує диференційованого підходу до прийняття цінових рішень [21].

Аналіз безбитковості (break-even analysis) дозволяє визначити мінімальний обсяг продажів, необхідний для покриття витрат при заданій ціні, або мінімальну ціну, що забезпечує безбитковість при прогнозованому обсязі реалізації. Цей метод є корисним при прийнятті рішень про знижки: він дозволяє розрахувати, наскільки має зрости обсяг продажів, щоб компенсувати зниження маржі [2].

Когортний аналіз передбачає групування покупців за часом їхнього першого звернення і відстеження поведінки кожної групи протягом часу. Застосування когортного аналізу до даних про транзакції дозволяє оцінити, як цінові зміни впливають на лояльність покупців, їхню схильність до повторних покупок і довічну цінність. Зокрема, когортний аналіз може виявити, чи не призводить агресивна знижкова політика до залучення покупців, орієнтованих виключно на ціну [12].

Метод оцінки прибутковості клієнтів (Customer Profitability Analysis) дозволяє визначити, які сегменти покупців або окремі клієнти є найбільш цінними для бізнесу з точки зору загального прибутку за весь час взаємодії. Поєднання цього методу з аналізом цінових пропозицій дає можливість оцінити ефективність диференційованих цінових стратегій для різних клієнтських сегментів [6].

Методи машинного навчання знаходять все більше застосування в задачах оцінювання і оптимізації цінових рішень. Алгоритми кластеризації дозволяють автоматично виявляти сегменти покупців з однорідною ціновою чутливістю. Методи часових рядів (ARIMA, LSTM-мережі) використовуються для прогнозування попиту і виручки при різних цінових сценаріях. Алгоритми

навчання з підкріпленням застосовуються для динамічної оптимізації цін у режимі реального часу [16].

Аналіз наслідків впровадження цінових пропозицій є одним із центральних завдань системи аналітики. Він передбачає вимірювання зміни ключових показників до і після застосування нової цінової пропозиції з урахуванням часового лагу і сезонних ефектів. Для коректного порівняння результатів важливо забезпечити однорідність контрольного і тестового періодів і використовувати відповідні статистичні критерії для перевірки значущості виявлених відмінностей [22].

Ринкова аналітика і моніторинг цін конкурентів формують важливий інформаційний контекст для оцінювання ефективності власних цінових пропозицій. Автоматизовані інструменти збору цінових даних (price scrapers) дозволяють відстежувати зміни цін конкурентів і порівнювати їх з власними пропозиціями. Аналіз цінового позиціонування відносно конкурентів допомагає виявляти можливості для підвищення маржі без ризику втрати конкурентоспроможності [1].

Довгострокова цінова аналітика включає аналіз трендів виручки і маржинальності в розрізі кварталів і років, що дозволяє виявляти стратегічні закономірності і приймати обґрунтовані рішення щодо розвитку цінової стратегії. Порівняльний аналіз результатів між різними підрозділами, каналами збуту або регіональними ринками забезпечує основу для тиражування найбільш ефективних підходів до ціноутворення [14].

Платформи бізнес-аналітики (Business Intelligence) відіграють важливу роль у системі аналізу ефективності цінових пропозицій. Такі рішення, як Tableau, Power BI або власні аналітичні розробки, дозволяють об'єднувати дані з різних джерел і надавати їх у вигляді інтерактивних дашбордів і звітів. Інтеграція аналізу цінових пропозицій у корпоративну BI-платформу підвищує доступність інформації для всіх рівнів управління підприємством і сприяє формуванню культури прийняття рішень, заснованих на даних [25].

Узагальнення наведених методів оцінювання ефективності цінових пропозицій свідчить, що жоден з них не є достатнім сам по собі, і повноцінна аналітика потребує комплексного застосування кількох підходів. Базовий рівень забезпечується системою КРІ з дашбордом; поглиблений аналіз вимагає застосування регресійних методів і когортного аналізу; стратегічний рівень передбачає оптимізацію на основі методів машинного навчання. Описана методологія склала теоретичну основу для проєктування інформаційної системи, що розробляється в даній роботі.

Висновки до розділу 1. У першому розділі проведено теоретичне дослідження основ електронної комерції, цінової політики та методів оцінювання ефективності цінових пропозицій. Визначено, що електронна комерція являє собою динамічно зростаючий сектор глобальної економіки, в якому управління ціноутворенням є одним із ключових чинників конкурентоспроможності. Встановлено, що цінова політика в інтернет-магазині формується під впливом як внутрішніх факторів (собівартість, стратегічне позиціонування), так і зовнішніх (конкуренція, еластичність попиту, сезонність). Систематизовано методологічний арсенал оцінювання ефективності цінових рішень, що включає КРІ-аналіз, А/В тестування, регресійний аналіз, когортний аналіз та методи машинного навчання. Отримані теоретичні результати склали концептуальну основу для проєктування та розробки інформаційної системи, описаної в наступних розділах.

РОЗДІЛ 2 ПРОЄКТУВАННЯ ТА РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Аналіз вимог до інформаційної системи та постановка задачі

Розробка ефективної інформаційної системи починається зі ретельного аналізу вимог, що визначають функціональне призначення системи, обмеження на її реалізацію та критерії якості. Аналіз вимог є критично важливою фазою процесу розробки програмного забезпечення, оскільки похибки, допущені на цьому етапі, надалі виправляються значно дорожче, ніж помилки в коді. Для даної системи аналіз вимог здійснювався методом поєднання інтерв'ю з потенційними користувачами і вивчення ринку аналогічних рішень.

Основними зацікавленими сторонами (стейкхолдерами) розробленої системи є менеджери з ціноутворення інтернет-магазину, керівники відділів продажів і маркетингу, а також IT-фахівці, відповідальні за підтримку і розвиток технологічної платформи. Кожна з перерахованих груп висуває специфічні вимоги до функціональності системи і способу представлення інформації. Менеджери з ціноутворення зосереджені на оперативному управлінні ціновими пропозиціями і моніторингу їхньої ефективності. Керівники продажів і маркетингу потребують агрегованих даних для стратегічного планування.

Для збору вимог застосовано кілька методологічних інструментів. Структуровані інтерв'ю з потенційними користувачами дозволили виявити основні больові точки в існуючих підходах до аналізу цін: тривалість ручного збору і обробки даних, відсутність наочної візуалізації динаміки, складність порівняння показників до і після цінових змін. Вивчення відкритих специфікацій існуючих аналітичних рішень дозволило визначити орієнтири для функціональності і продуктивності розробленої системи. Конкурентний аналіз виявив незаповнену нішу між корпоративними рішеннями і простими таблицями.

Функціональні вимоги до інформаційної системи аналізу ефективності цінових пропозицій визначені на основі методології MoSCoW (Must have, Should

have, Could have, Won't have). До обов'язкових вимог (Must have) віднесено такі: ведення і управління каталогом товарів з повною інформацією про ціни, собівартість та маржинальність; реєстрація і управління ціновими пропозиціями з підтримкою повного циклу їхнього затвердження і застосування; відображення ключових показників ефективності на аналітичному дашборді; реєстрація даних про продажі з прив'язкою до товарів і каналів збуту; формування звітів про вплив цінових змін на виручку і маржу.

До важливих вимог (Should have) включено: візуалізацію трендів продажів у часовому розрізі; аналіз ефективності цінових стратегій за категоріями; відображення статистики по каналах збуту; ведення журналу всіх змін цін з можливістю аудиту. Бажані, але не критичні функції (Could have) включають прогнозування майбутнього попиту, автоматичне формування цінових рекомендацій і інтеграцію з зовнішніми джерелами даних про конкурентів. Функції, що не включаються в поточну версію (Won't have): інтеграція зі службами доставки, CRM-функціональність, модуль управління рекламними кампаніями.

Нефункціональні вимоги визначають якісні характеристики системи. До ключових нефункціональних вимог відносяться: продуктивність (відповідь на запити до аналітичного дашборду не повинна перевищувати 2 секунди при обсязі бази даних до 100 тисяч записів), надійність (коректне відновлення після збоїв без втрати даних), зручність використання (інтуїтивно зрозумілий інтерфейс для користувачів без спеціальної технічної підготовки), масштабованість (можливість розширення функціональності і збільшення обсягів даних без повного переписування) та безпека (авторизований доступ і захист комерційно чутливих даних).



Рисунок 2.1 – Діаграма варіантів використання (Use Case) інформаційної системи аналізу ефективності цінових пропозицій

На рисунку 2.1 представлено діаграму варіантів використання системи відповідно до нотації UML. На діаграмі виділено двох основних акторів: Менеджер з ціноутворення (основний актор) і Системний адміністратор. Менеджер взаємодіє з такими варіантами використання: перегляд аналітичного дашборду, управління каталогом товарів, створення та управління ціновими пропозиціями, аналіз ефективності та перегляд трендів продажів.

Постановка задачі передбачає формальне визначення вхідних даних системи, процедур їхньої обробки та форматів вихідної інформації. Вхідними даними для системи є: словникові дані (інформація про товари, категорії, канали збуту); транзакційні дані (записи про продажі з деталями кожної операції); дані про цінові пропозиції і журнал їхнього виконання. Обробка даних включає агрегацію транзакційних даних за часовими інтервалами, категоріями і каналами, розрахунок показників маржинальності, виявлення кількісних наслідків застосування цінових пропозицій.

Формальна постановка задачі аналізу ефективності цінових пропозицій полягає у такому. Нехай задано множину товарів P , кожен з яких характеризується поточною ціною реалізації, собівартістю і приналежністю до категорії. Кожна цінова пропозиція описується чотирикою: ідентифікатор

товару, нова запропонована ціна, дата початку дії і обґрунтування. Дані про продажі являють собою часовий ряд транзакцій. Задача системи полягає в обчисленні і представленні показників ефективності кожної застосованої цінової пропозиції як різниці показників діяльності після і до її застосування при контролі сезонних і часових ефектів.

Аналіз ризиків проекту виявив такі основні загрози. Технічний ризик пов'язаний з можливими проблемами продуктивності при великих обсягах даних – мінімізується вибором СУБД з підтримкою індексування і агрегаційних запитів. Операційний ризик пов'язаний з відмовою від системи з боку кінцевих користувачів через складний інтерфейс – мінімізується проведенням тестування зручності використання на ранніх етапах. Ризик захмарення даних пов'язаний з ненавмисним видаленням або пошкодженням записів – мінімізується впровадженням механізму «м'якого видалення» замість фізичного знищення записів.

Вибір методологічного підходу до розробки системи здійснювався між традиційною каскадною (Waterfall) і гнучкою (Agile) методологіями. З огляду на відносно чітко визначені вимоги і невелику команду розробників, для даної системи обрано інкрементальну модель розробки, що поєднує переваги обох підходів. Розробка здійснювалася в ітераційному режимі з початковою реалізацією базової функціональності і поступовим додаванням нових можливостей. Кожна ітерація тривала 1–2 тижні і завершувалася демонстрацією функціональних результатів.

Огляд існуючих рішень для аналізу ефективності ціноутворення в е-комерції свідчить, що ринок пропонує переважно два типи продуктів: великі корпоративні платформи (Pricefx, PROS, Zilliant), орієнтовані на великий бізнес і вимагаючі значних фінансових і часових вкладень в інтеграцію, і прості аналітичні таблиці в Excel, що не задовольняють вимогам оперативності і масштабованості. Розроблена в рамках цієї роботи система заповнює нішу між цими двома полюсами, пропонуючи повнофункціональне аналітичне рішення для середнього бізнесу з мінімальними вимогами до інфраструктури.

Таким чином, у результаті аналізу вимог сформовано повне технічне завдання на розробку інформаційної системи, що включає функціональні вимоги трьох рівнів пріоритетності, нефункціональні вимоги до продуктивності, надійності і безпеки, а також формальне визначення задачі аналізу ефективності цінових пропозицій.

2.2. Моделювання структури системи та бази даних

Проектування архітектури інформаційної системи є ключовим технічним рішенням, що визначає організаційну структуру програмного забезпечення і принципи взаємодії його компонентів. Для розробленої системи обрано трирівневу клієнт-серверну архітектуру, що включає рівень представлення (frontend), рівень бізнес-логіки (backend) і рівень зберігання даних (database). Ця архітектура забезпечує чітке розмежування обов'язків між компонентами, спрощує супровід і модифікацію коду, а також відкриває можливості для горизонтального масштабування в майбутньому.

Рівень представлення реалізовано у вигляді односторінкового застосунку (Single-Page Application, SPA) на основі бібліотеки React.js. Використання SPA-архітектури забезпечує плавну і відповідну взаємодію з користувачем без перезавантаження сторінок, що є важливим для аналітичних застосунків з великою кількістю інтерактивних візуалізацій. React.js обрано завдяки компонентній моделі, що полегшує повторне використання коду, та декларативному підходу до опису інтерфейсу.

Рівень бізнес-логіки реалізовано у вигляді REST API на основі фреймворку Express.js для середовища виконання Node.js. Вибір Node.js обумовлений його асинхронною моделлю виконання, що ефективно обробляє велику кількість одночасних запитів при відносно невисоких вимогах до

апаратних ресурсів. Використання JavaScript як єдиної мови програмування для обох рівнів спрощує команду розробки і обмін даними між клієнтом і сервером.

Рівень зберігання даних реалізовано на основі вбудованої бази даних NeDB, що є реалізацією документоорієнтованого сховища з API, сумісним з MongoDB. NeDB зберігає дані у вигляді JSON-файлів на диску і не вимагає встановлення окремого сервера баз даних, що суттєво спрощує розгортання системи. При необхідності перенесення на реляційну або хмарну СУБД архітектура системи дозволяє це зробити з мінімальними змінами коду завдяки абстракції рівня доступу до даних.

При виборі архітектурного стилю розглядалися також мікросервісна архітектура і архітектура на основі подій (Event-Driven). Від мікросервісної архітектури відмовлено з огляду на порівняно невеликий масштаб системи і ризик надмірної складності операційної інфраструктури. Від архітектури на основі подій відмовлено через відсутність вимог до обробки асинхронних потоків даних у реальному часі. Трирівнева монолітна архітектура є оптимальним рішенням для поточного масштабу системи і забезпечує достатню гнучкість для подальшого розвитку.

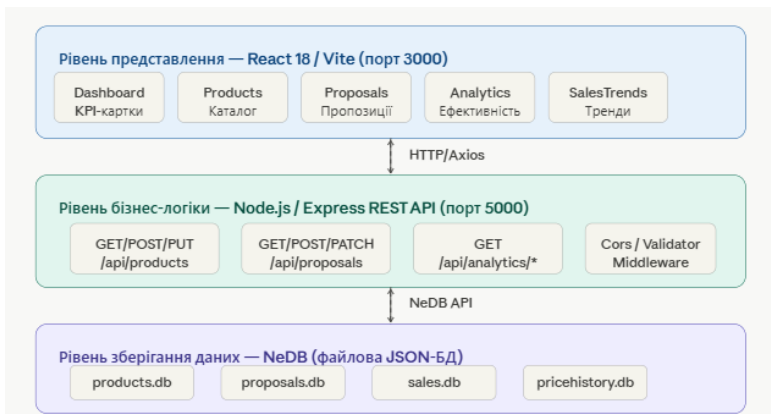


Рисунок 2.2 – Архітектурна діаграма інформаційної системи: компоненти та їхня взаємодія за трирівневою моделлю

Рисунок 2.2 ілюструє архітектуру системи і потоки даних між її компонентами. На рівні представлення виділено п'ять основних функціональних

модулів: Dashboard (дашборд KPI), Products (управління товарами), Proposals (управління ціновими пропозиціями), Analytics (аналіз ефективності) і SalesTrends (тренди продажів). Кожен модуль взаємодіє з відповідними ендпоінтами API рівня бізнес-логіки. Рівень зберігання даних включає чотири колекції: товари, пропозиції, продажі і журнал цін.

Проектування бази даних здійснювалося на основі концептуального моделювання з використанням діаграми «сутність-зв'язок» (Entity-Relationship Diagram, ERD). Концептуальна модель включає п'ять основних сутностей: Категорія (Category), Товар (Product), Цінова пропозиція (Proposal), Продаж (Sale) і Запис журналу цін (PriceHistory). Ці сутності зв'язані між собою відносинами різних типів, що відображають бізнес-правила предметної області.

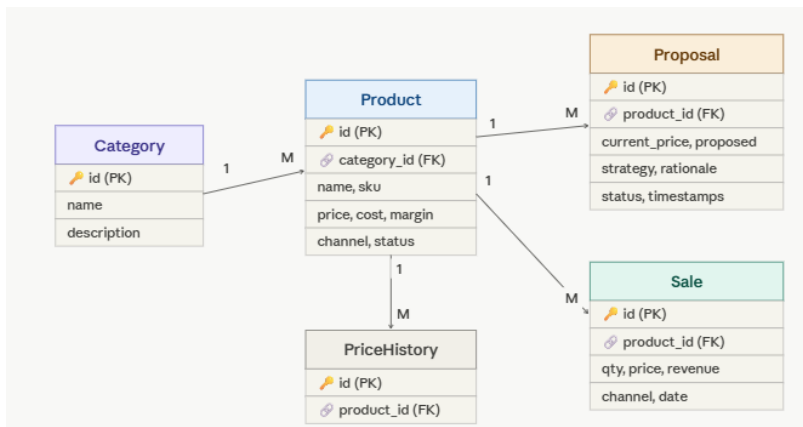


Рисунок 2.3 – Діаграма «сутність-зв'язок» бази даних інформаційної системи аналізу цінових пропозицій

На рисунку 2.3 представлена ERD-діаграма бази даних системи. Сутність Category є предком для сутності Product у відношенні «один-до-багатьох»: кожна категорія включає один або більше товарів, а кожен товар належить рівно одній категорії. Сутність Product пов'язана із сутністю Proposal відношенням «один-до-багатьох»: для одного товару може бути створено кілька цінових пропозицій.

Аналогічне відношення «один-до-багатьох» пов'язує Product і Sale. Сутність PriceHistory зберігає хронологічний журнал всіх змін ціни для кожного товару.

Атрибутний склад кожної сутності визначено на основі аналізу предметної області і функціональних вимог до системи. Сутність Product включає: унікальний ідентифікатор, найменування, SKU (Stock Keeping Unit), посилання на категорію, поточну ціну, собівартість, розрахункову маржу у відсотках, канал збуту і статус (активний чи деактивований). Сутність Proposal містить: ідентифікатор, посилання на товар, поточну ціну на момент створення пропозиції, запропоновану нову ціну, обґрунтування, стратегію ціноутворення, статус (pending/approved/applied/rejected) і часові мітки. Сутність Sale включає: ідентифікатор, посилання на товар, кількість, ціну продажу, дохід, канал збуту і дату.

Структура API сервера побудована відповідно до принципів REST-архітектурного стилю: кожен ресурс ідентифікується унікальним URL, операції над ресурсами виконуються стандартними HTTP-методами (GET, POST, PUT, PATCH, DELETE), відповіді сервера формуються у форматі JSON. Сервер реалізує три групи маршрутів: для управління товарами (/api/products), управління пропозиціями (/api/proposals) і аналітичні запити (/api/analytics). Кожен маршрут реалізовано у вигляді окремого модуля маршрутизатора Express.js для забезпечення чіткого розмежування відповідальності.

Моделювання бізнес-процесу управління ціновою пропозицією виконано у вигляді діаграми станів і переходів. Пропозиція може перебувати в одному з чотирьох станів: нова (pending), затверджена (approved), застосована (applied) і відхилена (rejected). Переходи між станами відповідають діям менеджера або адміністратора в системі. При переході в стан «застосована» система автоматично виконує три пов'язані операції: оновлення ціни товару в каталозі, розрахунок нової маржинальності і додавання запису до журналу змін цін.

Специфікація REST API включає такі ключові ендпоінти. Для групи Products: GET /api/products – отримання списку товарів з підтримкою фільтрації і пошуку; POST /api/products – створення нового товару; PUT /api/products/:id –

оновлення атрибутів товару; DELETE /api/products/:id – деактивація товару; GET /api/products/categories – отримання дерева категорій. Для групи Proposals: GET /api/proposals – отримання списку пропозицій з фільтрацією за статусом; POST /api/proposals – створення нової пропозиції; PATCH /api/proposals/:id/status – зміна статусу пропозиції з виконанням бізнес-логіки. Для групи Analytics: GET /api/analytics/dashboard – отримання агрегованих KPI для дашборду; GET /api/analytics/price-effectiveness – аналіз ефективності застосованих пропозицій; GET /api/analytics/sales-trends – тренди продажів за обраний часовий горизонт.

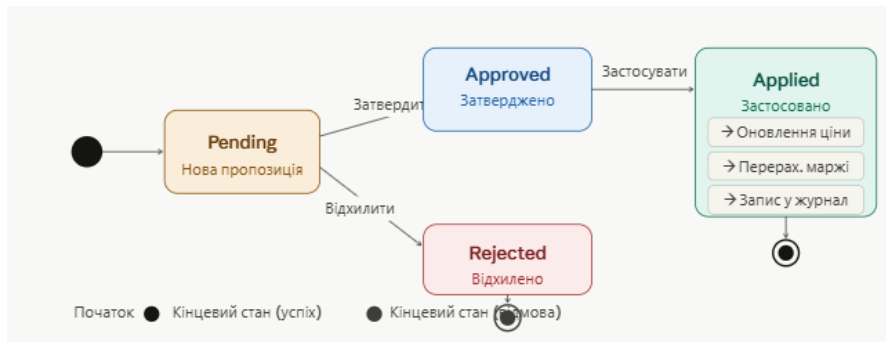


Рисунок 2.4 – Діаграма станів і переходів для бізнес-процесу управління ціною пропозицією

На рисунку 2.4 представлено діаграму станів пропозиції, що ілюструє допустимі переходи між станами і відповідні дії системи при кожному переході. Стан pending є початковим для всіх нових пропозицій. Зі стану pending можливі два переходи: в стан approved (при затвердженні) або в стан rejected (при відхиленні). Зі стану approved можливий один перехід: в стан applied (при застосуванні пропозиції до каталогу), який ініціює автоматичне оновлення ціни товару.

Проектування структури фронтенд-застосунку здійснено з використанням компонентного підходу React.js. Ієрархія компонентів включає кореневий компонент App, що містить бічну навігаційну панель і управляє маршрутизацією між основними розділами системи. На наступному рівні

ієрархії розташовані компоненти-сторінки: Dashboard, Products, Proposals, Analytics і SalesTrends. Кожна сторінка містить специфічні дочірні компоненти: картки KPI, таблиці, модальні вікна і компоненти графіків на основі бібліотеки Recharts.

Проектування бази даних враховувало вимоги до продуктивності запитів. Для основних колекцій визначено ключові індекси: колекція products індексується за полями category_id і status; колекція proposals – за product_id і status; колекція sales – за product_id, date і channel. Ці індекси забезпечують швидку фільтрацію даних при виконанні аналітичних запитів і суттєво скорочують час відповіді при зростанні обсягів даних.

2.3. Розробка алгоритмів аналізу ефективності цінових пропозицій

Алгоритмічна основа системи аналізу ефективності цінових пропозицій визначає логіку обчислення аналітичних показників, що відображаються на дашборді і в аналітичних звітах. Розробка алгоритмів базується на методологічних підходах, описаних у першому розділі роботи, і адаптує їх до специфіки архітектури системи та формату наявних даних.

Алгоритм розрахунку ключових показників ефективності (KPI) для аналітичного дашборду базується на агрегації транзакційних даних за задані часові інтервали. На вході алгоритм отримує дані про продажі за заданий часовий горизонт і дані про товари з поточними цінами і собівартістю. На виході формуються такі показники: загальна виручка за 30 і 7 останніх днів, темп зростання виручки у відсотках відносно аналогічного попереднього інтервалу, середня маржинальність по всіх транзакціях, розподіл пропозицій за статусами.

Алгоритм аналізу ефективності застосованих цінових пропозицій є центральним аналітичним алгоритмом системи. Його логіка полягає в порівнянні показників продажів товару в двох рівновіддалених часових вікнах: «до

застосування пропозиції» (часове вікно тривалістю N днів до дати застосування) і «після застосування пропозиції» (часове вікно тривалістю N днів після). Тривалість вікна за замовчуванням становить 14 днів, що відповідає мінімально достатньому часу для накопичення статистично значущих даних. При порівнянні розраховується різниця у виручці, середньому обсязі продажів і маржинальному доходу між двома вікнами.

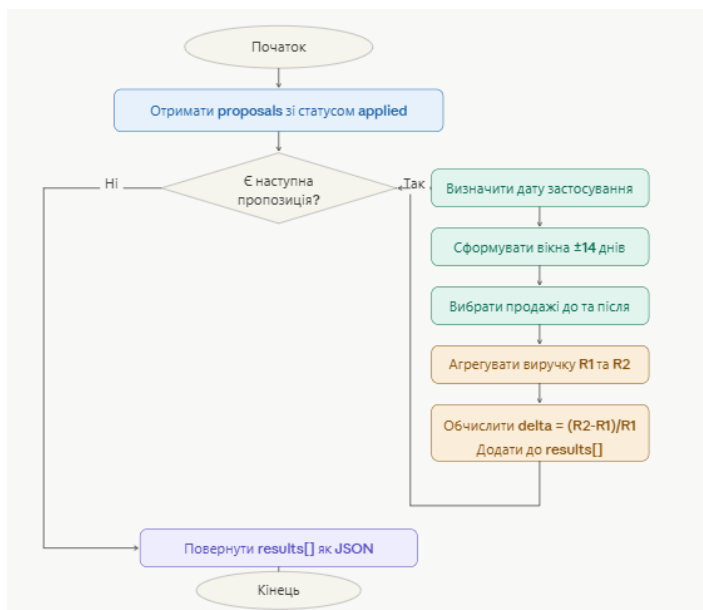


Рисунок 2.5 – Блок-схема алгоритму аналізу ефективності застосованої цінової пропозиції

На рисунку 2.5 представлено блок-схему алгоритму аналізу ефективності цінових пропозицій відповідно до нотації ДСТУ ISO 5807. Алгоритм починається з отримання переліку пропозицій зі статусом «applied». Для кожної пропозиції виконується послідовність операцій: визначення дати застосування, формування масивів транзакцій для двох порівнюваних вікон, розрахунок агрегованих показників для кожного вікна і обчислення відносного і

абсолютного відхилення. Результати зберігаються в масиві об'єктів ефективності, що повертається як відповідь на запит до API.

Математична основа алгоритму аналізу ефективності базується на таких формулах. Виручка за часове вікно розраховується як сума добутків ціни і кількості одиниць проданого товару по кожній транзакції у вікні. Маржинальний дохід за вікно розраховується як різниця між виручкою і собівартістю проданих товарів. Відносна зміна виручки розраховується за формулою: $\Delta R\% = (R_2 - R_1) / R_1 \times 100\%$, де R_1 – виручка до застосування пропозиції, R_2 – виручка після. Для пропозицій з нульовою виручкою до застосування показник зміни встановлюється рівним 0, щоб уникнути ділення на нуль.

Алгоритм розрахунку щоденних і тижневих трендів продажів реалізує системну процедуру перетворення сирих транзакційних даних у структуровані часові ряди, що забезпечує можливість подальшого аналітичного опрацювання, виявлення закономірностей і прийняття обґрунтованих управлінських рішень. У своїй основі він спирається на принципи часової агрегації, статистичної узагальненості та порівняльного аналізу, що дозволяє перетворити великий масив неструктурованих записів про продажі у компактні та інформативні представлення динаміки бізнес-показників. Вхідними даними для алгоритму виступає масив транзакцій, кожна з яких містить атрибути, що характеризують окрему операцію купівлі-продажу, зокрема ідентифікатор товару, дату здійснення операції, кількість одиниць товару, ціну та сумарну вартість. Додатково передається параметр `days`, який визначає глибину аналізу у часовому вимірі i , за замовчуванням, становить 30 календарних днів, що дозволяє охопити типовий місячний цикл активності продажів і водночас уникнути надмірного накопичення застарілих даних, які можуть спотворювати актуальні тенденції.

На першому етапі виконання алгоритму здійснюється фільтрація вхідного масиву транзакцій за часовим критерієм, тобто відбираються лише ті записи, дата яких належить до інтервалу, що визначається параметром `days` відносно поточного моменту часу. Така операція дозволяє локалізувати аналіз у межах релевантного часового відрізка, забезпечуючи адаптивність алгоритму до різних

сценаріїв використання, наприклад короткострокового моніторингу або більш глибокого ретроспективного аналізу. Після цього виконується нормалізація дат транзакцій, яка полягає у приведенні їх до єдиного формату, що виключає вплив часових зон, форматів запису або інших технічних відмінностей, які можуть виникати внаслідок інтеграції даних із різних джерел.

Далі алгоритм переходить до формування щоденного тренду, що передбачає групування транзакцій за календарною датою. У межах цього процесу всі транзакції, що мають однакове значення дати, об'єднуються в одну групу, після чого для кожної такої групи обчислюються агреговані показники, зокрема сумарна виручка та кількість замовлень. Сумарна виручка визначається як сума значень вартості всіх транзакцій у межах відповідного дня, що дозволяє оцінити фінансову результативність діяльності за конкретний часовий інтервал. Кількість замовлень, у свою чергу, розраховується як кількість окремих транзакцій або, залежно від постановки задачі, як кількість унікальних замовлень, якщо транзакції агрегуються за ідентифікатором замовлення. Такий підхід дозволяє отримати двовимірний часовий ряд, у якому кожному дню відповідає набір ключових метрик, що відображають інтенсивність і результативність продажів.

Особливу увагу в процесі побудови щоденного тренду приділяється обробці пропущених значень, тобто днів, у які транзакції відсутні. У таких випадках алгоритм може або явно включати відповідні дні до часового ряду із нульовими значеннями показників, або ж застосовувати методи інтерполяції для згладжування ряду, залежно від вимог до подальшого аналізу. Включення нульових значень дозволяє зберегти коректну часову структуру даних і уникнути зміщення індексів, що є критично важливим при використанні методів аналізу часових рядів, таких як ковзні середні або сезонна декомпозиція.

Паралельно з формуванням щоденного тренду здійснюється побудова тижневого тренду, що базується на групуванні транзакцій за номером календарного тижня. У цьому випадку кожна транзакція відноситься до певного тижня відповідно до прийнятої системи обчислення, наприклад стандарту ISO,

де тиждень починається з понеділка і має порядковий номер у межах року. Після групування для кожного тижня аналогічно обчислюються сумарна виручка та кількість замовлень, що дозволяє отримати більш узагальнений погляд на динаміку продажів, зменшуючи вплив випадкових коливань, характерних для щоденних даних. Такий рівень агрегації є особливо корисним для виявлення середньострокових тенденцій, сезонних коливань і циклічних закономірностей, що можуть бути пов'язані, наприклад, із поведінкою споживачів у різні дні тижня.

Наступним ключовим етапом алгоритму є виявлення товарів із позитивною динамікою продажів, що передбачає порівняльний аналіз обсягів реалізації окремих товарів у межах досліджуваного інтервалу часу. Для цього весь часовий інтервал, визначений параметром *days*, умовно поділяється на дві рівні частини – першу та другу половини. У межах кожної з цих частин для кожного товару обчислюється сумарний обсяг продажів, який може визначатися як загальна кількість проданих одиниць або як сумарна виручка, залежно від цілей аналізу. Такий підхід дозволяє оцінити зміну попиту на кожен товар у динаміці, порівнюючи більш ранній період із більш пізнім.

Для кількісної оцінки цієї зміни використовується показник темпу зростання, який визначається як відношення різниці між обсягами продажів у другій та першій половинах інтервалу до обсягу продажів у першій половині. У випадках, коли обсяг продажів у першій половині дорівнює нулю, можуть застосовуватися спеціальні правила обробки, наприклад присвоєння максимально можливого значення темпу зростання або виключення таких товарів із розгляду, щоб уникнути ділення на нуль і пов'язаних із цим математичних некоректностей. Отримані значення темпів зростання дозволяють здійснити ранжування товарів за ступенем позитивної динаміки, формуючи список лідерів зростання, які можуть становити особливий інтерес для подальшого маркетингового або управлінського впливу.

Ранжування товарів виконується за спаданням значення темпу зростання, що дозволяє на верхніх позиціях списку отримати ті позиції, які демонструють

найбільш стрімке збільшення попиту. При цьому доцільно враховувати не лише відносні показники зростання, але й абсолютні значення обсягів продажів, оскільки товар із дуже низькою базою може демонструвати високий відносний приріст, не маючи при цьому суттєвого впливу на загальні фінансові результати. У зв'язку з цим алгоритм може бути розширений шляхом введення додаткових критеріїв фільтрації або вагових коефіцієнтів, що дозволяють балансувати між відносною і абсолютною значущістю показників.

Загалом описаний алгоритм характеризується високою гнучкістю та масштабованістю, оскільки може бути адаптований до різних обсягів даних і різних вимог до точності та деталізації аналізу. Його реалізація може здійснюватися із використанням сучасних інструментів обробки даних, таких як системи управління базами даних, аналітичні платформи або мови програмування, орієнтовані на роботу з масивами даних. При цьому важливим аспектом є оптимізація обчислювальних процедур, зокрема ефективне використання індексів, кешування проміжних результатів і паралельна обробка даних, що дозволяє забезпечити високу продуктивність навіть при роботі з великими обсягами транзакційної інформації.

У підсумку, алгоритм розрахунку щоденних і тижневих трендів продажів виступає фундаментальним інструментом аналітики, що забезпечує перехід від сирих даних до осмислених показників, які відображають реальну динаміку бізнес-процесів. Його використання дозволяє не лише фіксувати поточний стан системи продажів, але й виявляти приховані закономірності, прогнозувати майбутні зміни та формувати обґрунтовані стратегії розвитку, що є критично важливим у сучасних умовах високої конкуренції та динамічності ринкового середовища.

Алгоритмічне забезпечення аналізу ефективності каналів збуту та стратегій ціноутворення в сучасних інформаційних системах електронної комерції ґрунтується на принципах інтеграції, узагальнення та багатовимірної обробки транзакційних даних, що дозволяє отримати комплексне уявлення про поведінку споживачів, економічну доцільність маркетингових рішень і динаміку

фінансових показників підприємства. У межах алгоритму розподілу продажів за каналами збуту первинним етапом виступає формування консолідованого масиву транзакцій, який включає записи про всі здійснені покупки незалежно від джерела їх ініціації, при цьому особлива увага приділяється нормалізації даних, усуненню дублювання, а також уніфікації форматів представлення атрибутів, таких як ідентифікатори замовлень, часові мітки, суми операцій і класифікаційні ознаки каналів. Подальша обробка передбачає класифікацію кожної транзакції відповідно до визначеного каналу збуту, зокрема вебсайту, мобільного додатку або маркетплейсу, що дозволяє здійснювати агрегацію даних у розрізі цих категорій. На цьому етапі застосовуються операції групування та підсумовування, внаслідок чого для кожного каналу обчислюється сукупна виручка як сума грошових надходжень, кількість замовлень як кардинальність відповідної підмножини транзакцій, а також частка каналу в загальному обсязі продажів, яка визначається як відношення виручки каналу до загальної виручки за досліджуваний період. Такий підхід забезпечує можливість не лише кількісної оцінки ефективності кожного каналу, але й порівняльного аналізу їх внеску у формування загального фінансового результату, що є критично важливим для прийняття управлінських рішень щодо оптимізації маркетингових бюджетів, перерозподілу ресурсів і вдосконалення клієнтського досвіду.

Суттєвим аспектом алгоритму є врахування часової динаміки, що реалізується через можливість сегментації даних за певними інтервалами, такими як день, тиждень або місяць, що дозволяє виявляти сезонні коливання, тренди та аномалії у поведінці каналів збуту. Додатково може застосовуватися нормування показників, наприклад, розрахунок середнього чеку або коефіцієнта конверсії, що поглиблює аналітичну цінність отриманих результатів. У складніших реалізаціях алгоритм може інтегрувати методи багатовимірного аналізу, зокрема кластеризацію або факторний аналіз, що дозволяє виявляти приховані закономірності у взаємодії каналів і сегментів клієнтів.

Паралельно з аналізом каналів збуту здійснюється алгоритмічний розрахунок статистики за стратегіями ціноутворення, який передбачає іншу

логіку агрегації та інтерпретації даних. Вихідною множиною для цього алгоритму є перелік застосованих цінових пропозицій, кожна з яких характеризується типом стратегії, параметрами зміни ціни та фактичним впливом на фінансові результати. На першому етапі відбувається класифікація пропозицій за типами стратегій, серед яких виділяються знижки, конкурентне ціноутворення, ціноутворення на основі сприйнятої цінності та преміальне ціноутворення. Така типологізація дозволяє структурувати різноманітні маркетингові інструменти у логічно узгоджену систему, що спрощує подальший аналіз і інтерпретацію результатів.

Після класифікації здійснюється групування пропозицій за відповідними стратегіями, що створює підґрунтя для обчислення агрегованих показників. Зокрема, для кожної стратегії визначається кількість застосованих пропозицій, що відображає інтенсивність використання відповідного підходу до ціноутворення, середній відсоток зміни ціни, який обчислюється як середнє арифметичне відносних змін цін у межах групи, а також середній вплив на виручку, що може визначатися як середня різниця між фактичною виручкою та базовим сценарієм без застосування відповідної стратегії. Останній показник є особливо важливим, оскільки він дозволяє оцінити економічну ефективність кожної стратегії не лише з точки зору зміни цін, але й з урахуванням поведінкових реакцій споживачів, таких як зміна обсягів попиту або еластичність.

У більш розвинених моделях алгоритм може враховувати додаткові фактори, такі як сегментація клієнтів, категорії товарів, часові характеристики застосування стратегій, а також зовнішні ринкові умови, що дозволяє здійснювати багатофакторний аналіз і підвищувати точність оцінок. Наприклад, ефективність знижок може суттєво відрізнятися залежно від категорії товару або сезону, що потребує відповідної диференціації даних. Крім того, застосування методів статистичного моделювання, таких як регресійний аналіз або машинне навчання, дозволяє прогнозувати вплив різних стратегій на майбутні показники виручки та оптимізувати параметри ціноутворення.

Інтеграція результатів обох алгоритмів відкриває можливості для комплексного аналізу взаємозв'язку між каналами збуту та стратегіями ціноутворення. Зокрема, може бути досліджено, які стратегії є найбільш ефективними для конкретних каналів, чи існують синергічні ефекти між певними каналами та типами пропозицій, а також як змінюється поведінка споживачів залежно від поєднання цих факторів. Такий підхід дозволяє переходити від ізольованого аналізу окремих аспектів діяльності до системного бачення бізнес-процесів, що є необхідною умовою для формування ефективної стратегії розвитку підприємства.

У підсумку, розглянуті алгоритми виступають ключовими інструментами аналітичної підтримки управлінських рішень у сфері електронної комерції, забезпечуючи об'єктивну, кількісно обґрунтовану оцінку ефективності каналів збуту та стратегій ціноутворення. Їх застосування дозволяє не лише підвищити прозорість бізнес-процесів, але й створює передумови для впровадження адаптивних, даних-орієнтованих підходів до управління, що відповідають вимогам сучасного цифрового середовища.

Алгоритм виявлення товарів із низькою маржинальністю ґрунтується на системному підході до аналізу фінансових показників асортименту з урахуванням як статичних, так і динамічних характеристик формування прибутковості, що дозволяє забезпечити обґрунтованість управлінських рішень у сфері ціноутворення та оптимізації товарної політики. У його основі лежить формалізована процедура порівняння індивідуального значення маржі кожної товарної позиції з наперед визначеним пороговим рівнем, встановленим на рівні 35%, який виступає критерієм допустимої економічної ефективності продажу. Такий поріг обирається не довільно, а на підставі аналітичних оцінок структури витрат, конкурентного середовища, еластичності попиту та стратегічних цілей підприємства, що забезпечує адаптивність алгоритму до конкретних умов функціонування бізнесу. Розрахунок маржинальності здійснюється за стандартизованою формулою, відповідно до якої маржа визначається як відношення різниці між ціною реалізації та собівартістю до ціни реалізації,

помножене на 100%, що дозволяє у відносних величинах оцінити частку прибутку у структурі доходу від продажу конкретного товару. При цьому важливим є коректне визначення як ціни, так і собівартості, оскільки похибки в обліку витрат або некоректне врахування знижок, бонусів чи логістичних витрат можуть суттєво спотворити результати аналізу та призвести до хибних управлінських висновків.

У межах реалізації алгоритму первинним етапом виступає збір і підготовка вхідних даних, що включає агрегування інформації з різних джерел, зокрема систем управління запасами, бухгалтерських модулів та систем електронної комерції, де фіксуються фактичні ціни продажу, закупівельні ціни, витрати на зберігання, транспортування та інші супутні витрати. Після цього здійснюється нормалізація даних, що передбачає приведення їх до єдиного формату, усунення аномалій, обробку пропущених значень та перевірку на логічну узгодженість, що є критично важливим для забезпечення достовірності подальших розрахунків. Наступним кроком є безпосередній розрахунок маржі для кожної товарної позиції, який виконується автоматизовано із застосуванням відповідних програмних засобів, що дозволяє обробляти великі обсяги даних у режимі, близькому до реального часу.

Отримані значення маржі порівнюються з установленим порогом у 35%, у результаті чого формується підмножина товарів, для яких показник маржинальності є нижчим за допустимий рівень. Ці товари автоматично класифікуються як потенційно проблемні з точки зору прибутковості та підлягають подальшому аналізу. Важливо зазначити, що сам факт низької маржі не завжди свідчить про негативний економічний ефект, оскільки окремі товари можуть виконувати функцію трафікоутворюючих або іміджевих позицій, однак їх систематичне виявлення дозволяє своєчасно ідентифікувати ризики зниження загальної рентабельності бізнесу.

Особливістю алгоритму є інтеграція аналізу динаміки змін маржі, що реалізується шляхом порівняння поточного значення показника з аналогічним значенням за попередній звітний період, зазвичай попередній місяць. Такий

підхід дозволяє не лише фіксувати статичні відхилення, але й виявляти негативні тенденції, що можуть свідчити про погіршення умов закупівлі, зміну конкурентного середовища або неефективність застосованих цінових стратегій. Для цього обчислюється різниця або відсоткова зміна маржі, що дає змогу оцінити напрям і швидкість її зміни. У випадках, коли спостерігається стійке зниження маржинальності, навіть якщо її значення ще не досягло критичного порогу, такі товари також можуть бути включені до переліку для моніторингу, що розширює аналітичні можливості алгоритму.

Результати виконання алгоритму візуалізуються у спеціалізованому блоці інформаційної панелі (дашборду), що забезпечує зручний і наочний доступ до ключових показників для осіб, які приймають рішення. У цьому блоці відображається перелік товарів із низькою маржинальністю, доповнений такими атрибутами, як поточна маржа, зміна маржі відносно попереднього періоду, обсяг продажів, а також інші релевантні метрики, що дозволяють комплексно оцінити ситуацію. Візуалізація може включати кольорове кодування, де, наприклад, червоний колір сигналізує про критично низьку маржу або різке її падіння, що сприяє швидкій ідентифікації проблемних позицій без необхідності детального аналізу кожного рядка даних.

У межах подальшого використання результатів алгоритму передбачається реалізація механізмів підтримки прийняття рішень, які можуть включати рекомендації щодо перегляду цін, оптимізації витрат або зміни постачальників. Наприклад, у випадку виявлення товару з низькою маржинальністю може бути запропоновано підвищення ціни за умови збереження конкурентоспроможності або пошук альтернативних каналів постачання з нижчою закупівельною ціною. Крім того, алгоритм може бути інтегрований із системами прогнозування попиту, що дозволяє враховувати потенційний вплив зміни цін на обсяг продажів і, відповідно, на загальний фінансовий результат.

Важливим аспектом є також можливість адаптації порогового значення маржі залежно від категорії товарів, їх життєвого циклу або стратегічної ролі в

асортименті. Наприклад, для товарів преміум-сегменту може бути встановлено вищий поріг, тоді як для товарів масового попиту допускається нижчий рівень маржинальності за умови високого обороту. Така диференціація підвищує гнучкість алгоритму та дозволяє більш точно відображати реальні економічні умови функціонування підприємства.

Загалом, описаний алгоритм є складовою частиною системи аналітики бізнес-процесів, спрямованої на підвищення ефективності управління товарним асортиментом і забезпечення стабільного рівня прибутковості. Його застосування дозволяє своєчасно виявляти проблемні зони, мінімізувати фінансові ризики та формувати обґрунтовану політику ціноутворення, що в умовах високої конкуренції на ринку є критично важливим для досягнення стратегічних цілей підприємства.

Алгоритм генерації тестових даних (seed data) при ініціалізації інформаційної системи електронної комерції виконує не лише допоміжну функцію наповнення бази даних початковими записами, але й виступає критично важливим інструментом моделювання поведінки реального бізнес-середовища, що дозволяє забезпечити адекватність подальшого тестування, валідації аналітичних модулів та оцінювання ефективності прийнятих рішень. У цьому контексті процес генерації даних має базуватися на принципах статистичної правдоподібності, репрезентативності та варіативності, що передбачає відтворення складних залежностей між сутностями предметної області. Початково формується множина товарів, яка включає 15 позицій, рівномірно або напіврівномірно розподілених між 5 категоріями, що дозволяє змодельовати як загальні, так і специфічні особливості попиту. Кожна категорія характеризується власним профілем поведінки користувачів, що відображається у відмінностях частоти покупок, середнього чеку, сезонності та чутливості до цінних змін. При генерації товарів враховується не лише їх належність до категорії, але й додаткові атрибути, такі як базова ціна, маржинальність, популярність, що надалі впливають на ймовірність здійснення транзакцій.

Наступним етапом є формування набору цінових пропозицій, кількість яких становить 10, причому вони розподіляються між товарами з урахуванням різних стратегій ціноутворення. Кожна пропозиція має статус, який може змінюватися в часі, наприклад «active», «expired», «scheduled», «applied», що дозволяє змоделювати динаміку маркетингових кампаній. Важливим аспектом є те, що ці пропозиції не генеруються випадковим чином у повністю незалежний спосіб, а прив'язуються до часових інтервалів і подій, які мають економічний сенс, зокрема акційних періодів, вихідних днів або сезонних піків попиту. Таким чином, забезпечується узгодженість між ціновими змінами та поведінкою покупців, що є ключовим для подальшого аналізу ефективності цих пропозицій.

Основний масив даних формується на етапі генерації транзакцій продажів, кількість яких становить приблизно 515 записів за період у 90 днів. Такий обсяг обрано не випадково, а з урахуванням необхідності забезпечення достатньої статистичної значущості при одночасному збереженні керованості обчислювальних процесів. Генерація транзакцій здійснюється із застосуванням стохастичних моделей, які враховують нерівномірність розподілу продажів у часі. Зокрема, вводиться вагова функція для днів тижня, відповідно до якої, наприклад, у вихідні дні може спостерігатися підвищена активність користувачів, тоді як у будні дні інтенсивність покупок є нижчою. Крім того, моделюються добові коливання, коли більшість транзакцій припадає на вечірні години, що відповідає типовій поведінці користувачів інтернет-магазинів.

Суттєву роль відіграє врахування сезонних факторів, які реалізуються через введення трендових компонентів у функцію генерації попиту. Наприклад, у межах 90-денного періоду можуть бути змодельовані короткострокові піки, пов'язані з умовними акційними кампаніями або святковими подіями, що призводить до тимчасового зростання кількості транзакцій. Це дозволяє перевірити, наскільки система здатна адекватно реагувати на різкі зміни навантаження та чи коректно працюють алгоритми аналітики в умовах нестабільних даних. Додатково враховується різна еластичність попиту для різних категорій товарів: для одних категорій зміна ціни може суттєво впливати

на обсяг продажів, тоді як для інших цей вплив є мінімальним. Така диференціація реалізується через використання коефіцієнтів чутливості, які впливають на ймовірність генерації транзакції для конкретного товару в конкретний момент часу.

Алгоритм також передбачає варіативність обсягів покупок у межах однієї транзакції, що досягається шляхом використання розподілів імовірностей для кількості одиниць товару. Це дозволяє змоделювати як поодинокі покупки, так і оптові замовлення, що є характерним для реальних систем електронної комерції. Крім того, враховується можливість повторних покупок одними й тими самими користувачами, що хоча й не моделюється на рівні ідентифікації конкретних клієнтів, але відображається через кластеризацію транзакцій у часовому просторі.

З технічної точки зору реалізація алгоритму генерації seed data базується на поєднанні детермінованих і випадкових компонентів, що забезпечує баланс між відтворюваністю результатів і їхньою варіативністю. Використання фіксованого початкового значення генератора випадкових чисел (seed) дозволяє отримувати однакові набори даних при повторному запуску, що є важливим для тестування та налагодження системи. Водночас внутрішня структура алгоритму побудована таким чином, щоб уникнути тривіальних або надто рівномірних розподілів, які не відповідають реальним умовам.

Таким чином, сформований набір тестових даних виконує функцію імітаційної моделі реального інтернет-магазину, в якій відтворено ключові закономірності поведінки користувачів і ринку. Це створює підґрунтя для проведення глибокого аналізу, включаючи оцінювання ефективності цінових пропозицій, виявлення трендів, тестування алгоритмів рекомендацій та оптимізації бізнес-процесів. Важливо підкреслити, що якість таких даних безпосередньо впливає на достовірність отриманих результатів, а отже, розробка алгоритму їх генерації має розглядатися як невід'ємна частина проектування всієї системи, що потребує системного підходу, врахування предметної специфіки та застосування методів математичного моделювання.

Оскільки вбудована документоорієнтована база даних NeDB орієнтована насамперед на прості сценарії збереження та вибірки даних і не передбачає повноцінної підтримки складних аналітичних запитів, характерних для реляційних систем керування базами даних, зокрема таких механізмів, як багаторівневе групування записів, використання агрегатних функцій (SUM, AVG, COUNT, MIN, MAX) у поєднанні з умовною логікою та вкладеними підзапитами, у проєктованій інформаційній системі було прийнято архітектурне рішення реалізувати відповідну функціональність на рівні серверної бізнес-логіки із використанням оперативної пам'яті. Такий підхід базується на попередньому отриманні з бази даних необроблених або частково відфільтрованих наборів записів із подальшим їх перетворенням, структуризацією та агрегацією засобами прикладного програмного коду, що виконується у середовищі виконання серверної частини. Важливо підкреслити, що в умовах експлуатації типового малого або середнього інтернет-магазину, де обсяг транзакційних даних, як правило, обмежується десятками або сотнями тисяч записів, а частота виконання складних аналітичних операцій не є критично високою, такий підхід забезпечує прийнятний баланс між складністю реалізації, швидкістю системи та гнучкістю подальшого розвитку.

З технічної точки зору, процес агрегації даних у пам'яті передбачає виконання послідовності операцій, що включають ітерацію по масиву отриманих документів, формування ключів групування на основі визначених атрибутів (наприклад, ідентифікатор товару, категорія, дата або часовий інтервал), накопичення проміжних результатів у відповідних структурах даних (зокрема, об'єктах або асоціативних масивах), а також обчислення необхідних статистичних показників. Така реалізація дозволяє досягти високої гнучкості, оскільки логіка обробки може бути адаптована до специфічних вимог предметної області без обмежень, притаманних декларативним мовам запитів. Крім того, використання сучасних можливостей мов програмування, таких як функціональні методи обробки колекцій (map, reduce, filter), сприяє підвищенню

читабельності та підтримуваності коду, що є важливим фактором у контексті довготривалої експлуатації програмного продукту.

Разом із тим, варто враховувати, що перенесення обчислювального навантаження з рівня бази даних на рівень прикладної логіки неминуче призводить до збільшення вимог до оперативної пам'яті та процесорних ресурсів серверної інфраструктури. У випадку значного зростання обсягу даних або інтенсивності запитів це може стати вузьким місцем системи, знижуючи її загальну продуктивність і масштабованість. Зокрема, при обробці великих масивів даних у пам'яті виникає ризик перевищення доступних ресурсів, що може спричинити деградацію швидкодії або навіть відмову окремих компонентів. Однак у контексті обраної цільової аудиторії та передбачуваних сценаріїв використання, такі ризики оцінюються як помірні, а їхній вплив може бути мінімізований шляхом впровадження додаткових оптимізацій, таких як попередня фільтрація даних на рівні запиту до бази, кешування результатів обчислень, обмеження розміру вибірок, а також використання пагінації.

Окрему увагу доцільно приділити питанням архітектурної гнучкості та можливості подальшого масштабування системи. Прийнята модель організації доступу до даних передбачає чітке розмежування між рівнем збереження інформації та рівнем бізнес-логіки, що дозволяє у майбутньому здійснити заміну використовуваної СКБД без суттєвих змін у прикладному коді. Зокрема, перехід від NeDB до більш потужних рішень, таких як документоорієнтована база даних MongoDB або реляційна система PostgreSQL, може бути реалізований шляхом адаптації шару доступу до даних (Data Access Layer), який інкапсулює специфіку взаємодії з конкретною базою. У такому випадку, складні операції агрегації можуть бути перенесені на рівень СКБД із використанням вбудованих механізмів, таких як aggregation pipeline у MongoDB або SQL-запити з групуванням і агрегатними функціями у PostgreSQL, що дозволить значно підвищити ефективність обробки великих обсягів інформації.

Таким чином, обрана стратегія реалізації аналітичної обробки даних у пам'яті серверної частини є компромісним рішенням, яке враховує як обмеження

використовуваних технологій, так і реальні потреби цільового середовища експлуатації. Вона забезпечує достатній рівень продуктивності для поточних задач, спрощує процес розробки та тестування, а також створює передумови для подальшої еволюції системи без необхідності кардинальної перебудови її архітектури. У контексті сучасних тенденцій розвитку програмних систем, де пріоритет надається модульності, масштабованості та незалежності компонентів, такий підхід може вважатися обґрунтованим і доцільним, особливо на початкових етапах життєвого циклу програмного продукту.

РОЗДІЛ 3 РЕАЛІЗАЦІЯ ТА ОЦІНЮВАННЯ РОБОТИ СИСТЕМИ

3.1. Програмна реалізація інформаційної системи

Програмна реалізація інформаційної системи здійснювалася відповідно до проектних рішень, описаних у другому розділі, з використанням сучасних інструментів і кращих практик веб-розробки. Загальна структура проекту розподілена між двома директоріями: backend містить серверну частину на Node.js, frontend – клієнтську частину на React.js. Така організація коду відповідає принципу розмежування відповідальності і спрощує незалежну розробку та тестування клієнтської і серверної частин.

Серверна частина системи реалізована на платформі Node.js версії 18+ з використанням фреймворку Express.js. Точкою входу додатка є файл server.js, що ініціалізує сервер, підключає відповідне проміжне ПЗ (middleware) і реєструє маршрутизатори для кожної групи ендпоінтів. Для обробки CORS-запитів від клієнтської частини підключено пакет cors. Серверний процес Node.js запускається на порту 5000 і проксується клієнтським сервером Vite при локальній розробці.

Ініціалізація бази даних і генерація початкових тестових даних реалізовані в окремому модулі database.js. Модуль створює екземпляри NeDB-колекцій для кожної сутності і перевіряє наявність даних при старті системи. У разі відсутності даних автоматично виконується процедура наповнення бази: створюються 5 категорій і 15 товарів різних цінових сегментів, включаючи смартфони, ноутбуки, аксесуари та програмне забезпечення, 10 цінових пропозицій з різними статусами і близько 515 записів про транзакції продажів за 90 днів.

Маршрутизатор управління товарами (/api/products) реалізує повний CRUD-інтерфейс для каталогу. Запит GET /api/products підтримує фільтрацію за категорією, статусом і пошуковим рядком через параметри запиту. При отриманні списку товарів система автоматично розраховує поточну

маржинальність кожного товару на основі ціни і собівартості. Запит DELETE /api/products/:id не видаляє запис фізично, а встановлює статус «деактивований», що дозволяє зберігати цілісність архівних даних про продажі.

Маршрутизатор управління пропозиціями (/api/proposals) реалізує логіку повного циклу обробки цінових пропозицій. При зміні статусу на «applied» виконується транзакційна послідовність дій: спочатку оновлюється запис про пропозицію, потім оновлюється поточна ціна товару в каталозі і розраховується нова маржа, нарешті, в журнал цін додається запис про зміну з повним набором атрибутів. Ця логіка забезпечує узгодженість даних і можливість аудиту всіх цінових змін.

Для управління залежностями і збірки серверної частини використовується прт з файлом package.json, що описує всі залежності проєкту. Виробничі залежності включають: express (маршрутизація і проміжне ПЗ), nedb-promises (асинхронний доступ до NeDB), cors (обробка CORS-запитів), express-validator (валідація вхідних даних). Залежності для розробки включають nodemon – інструмент для автоматичного перезапуску сервера при зміні файлів коду.

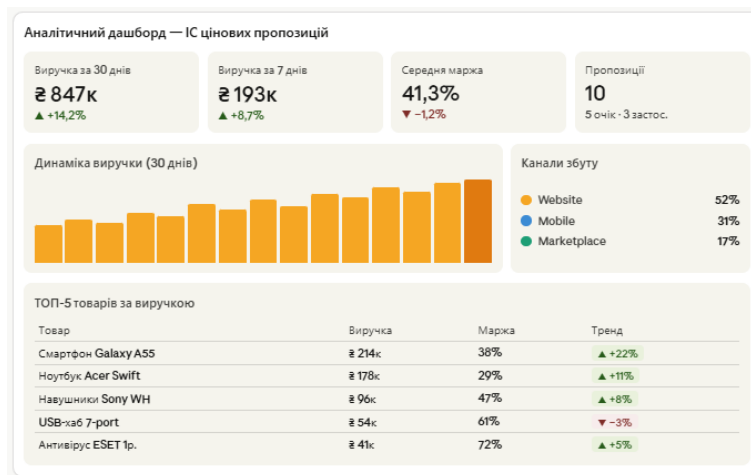


Рисунок 3.1 – Головний екран (дашборд) інформаційної системи з KPI-картками, графіком виручки та розподілом по каналах

На рисунку 3.1 представлено головний екран (дашборд) системи. У верхній частині екрана розміщено чотири картки KPI: виручка за 30 днів, виручка за 7 днів, середня маржинальність і кількість пропозицій за статусами. Нижче розташовано графік динаміки виручки за останній місяць (Area Chart) і кругова діаграма розподілу продажів за каналами збуту (Pie Chart). У нижній частині відображено таблицю ТОП-5 товарів за виручкою і перелік товарів з низькою маржинальністю. Дизайн витриманий у темній колірній схемі з акцентним кольором у діапазоні помаранчевого.

Аналітичний маршрутизатор (/api/analytics) реалізує три ендпоінти. Ендпоінт /dashboard повертає комплексний об'єкт даних для головного екрана, включаючи KPI-показники, дані для графіків і таблицю провідних товарів. Ендпоінт /price-effectiveness виконує описаний раніше алгоритм аналізу ефективності застосованих пропозицій. Ендпоінт /sales-trends приймає параметр days і повертає дані для графіків трендів продажів.

Клієнтська частина системи побудована як односторінковий застосунок на базі бібліотеки React 18 і зібраний за допомогою інструменту Vite. Vite забезпечує значно швидший цикл розробки порівняно з традиційним Create React App завдяки нативній підтримці ES-модулів у браузері і ефективній системі HMR (Hot Module Replacement). Зборка для виробництва генерує оптимізовані статичні файли з підтримкою «лениво завантаженого» коду.

Кореневий компонент App.jsx реалізує бічну навігаційну панель і систему маршрутизації між п'ятьма основними розділами системи. Навігаційна панель містить логотип системи, ієрархічне меню розділів з іконками і індикатором активного пункту. Стан активного розділу і відкриття бічної панелі на мобільних пристроях управляється через хуки useState. Адаптивна верстка забезпечує коректне відображення на екранах різних розмірів.

Компонент Dashboard.jsx отримує дані дашборду через хук useEffect при монтуванні компонента і зберігає їх у локальному стані через useState. Для відображення графіків використовується бібліотека Recharts: динаміка виручки візуалізується компонентом AreaChart з градієнтним заповненням, розподіл за

каналами – компонентом PieChart з власним обробником підказок. Картки KPI реалізовані як окремих підкомпонент з умовним форматуванням відсотка зростання: позитивний показник відображається зеленим кольором, від’ємний – червоним.

Для забезпечення коректного відображення числових даних у клієнтській частині розроблено набір службових функцій у модулі formatters.js. Функція formatCurrency форматує числові значення виручки у формат гривень з двома десятковими знаками і роздільниками тисяч. Функція formatPercent додає знак відсотка і обирає знак (+/-) відповідно до знаку значення. Функція formatDate перетворює ISO-рядки дат у людиночитаний формат відповідно до локалі користувача. Централізоване форматування усуває неоднорідність відображення даних у різних компонентах системи.

Товар	Поточна ціна	Нова ціна	Зміна	Стратегія	Статус	Дії
Смартфон Galaxy A55	€ 13 490	€ 12 900	-4,4%	Конкурентна	Очікує	Затвердити, Відхилити
Ноутбук Acer Swift	€ 28 990	€ 26 500	-8,6%	Знижка	Затверджено	Застосувати
Навушники Sony WH	€ 4 800	€ 5 200	+8,3%	На основі цінності	Застосовано	Деталі
USB-хаб 7-port	€ 890	€ 950	+6,7%	Преміальна	Відхилено	Деталі
Антивірус ESET 1р.	€ 1 299	€ 1 199	-7,7%	Знижка	Очікує	Затвердити, Відхилити

Рисунок 3.2 – Екран управління ціновими пропозиціями зі списком пропозицій, фільтрами за статусом і модальним вікном створення

На рисунку 3.2 представлено екран управління ціновими пропозиціями. Верхня частина містить кнопку створення нової пропозиції і фільтри за статусом (всі, очікують, затверджені, застосовані, відхилені). Основний простір займає таблиця пропозицій з колонками: товар, поточна ціна, запропонована ціна, зміна

у відсотках, стратегія, статус і дії. Кожен рядок містить кнопки для зміни статусу пропозиції відповідно до допустимих переходів. Кольорові значки статусу використовують семантичне кодування: жовтий для «очікує», синій для «затверджено», зелений для «застосовано», червоний для «відхилено».

Компонент `Proposals.jsx` реалізує логіку управління ціновими пропозиціями з урахуванням всіх допустимих переходів між статусами. Зміна статусу пропозиції виконується асинхронним запитом до API з відповідним підтвердженням через toast-повідомлення у разі успіху або помилки. Компонент `ProposalModal` реалізує форму створення нової пропозиції з вибором товару зі списку, введенням нової ціни і обґрунтування. Клієнтська валідація форми перевіряє наявність всіх обов'язкових полів і коректність введеної ціни перед відправкою запиту.

Компонент `Analytics.jsx` відображає поглиблений аналіз ефективності цінових рішень. Основна секція містить список застосованих пропозицій з кількісними показниками їхнього впливу на виручку і маржу. Для кожної пропозиції відображається порівняння виручки до і після застосування з підсвічуванням позитивного або негативного результату. Нижче розміщено стовпчикову діаграму зі статистикою ефективності за стратегіями ціноутворення і графіком цінових змін у часі.

Компонент `SalesTrends.jsx` забезпечує детальний аналіз динаміки продажів. Верхня частина відображає вибір часового горизонту (7, 14, 30 днів). Нижче розміщено два графіки `AreaChart`: щоденні і тижневі тренди виручки. Таблиця у нижній частині сторінки показує продуктивність товарів з ранжуванням за темпом зростання виручки.

Управління станом у клієнтській частині реалізовано через вбудовані хуки `React` без залучення зовнішніх бібліотек управління станом типу `Redux`. Оскільки стан кожного розділу є незалежним і не потребує глобальної синхронізації, використання `useState` і `useEffect` є достатнім підходом. Кастомний хук `useToast` управляє стеком toast-повідомлень, забезпечуючи автоматичне видалення після заданого часу і можливість ручного закриття.

Безпека HTTP-комунікації між клієнтом і сервером забезпечується кількома механізмами. CORS-конфігурація на сервері обмежує доступ до API лише з дозволених джерел. Вхідні дані API-запитів проходять валідацію на сервері за допомогою `express-validator` до передачі в обробники маршрутів. SQL-ін'єкції не є актуальними загрозами завдяки документоорієнтованій природі NoDB, проте XSS-атаки усуваються санітизацією текстових даних перед збереженням у базі.

3.2. Тестування та валідація результатів аналізу

Тестування інформаційної системи є обов'язковою частиною процесу розробки програмного забезпечення і забезпечує підтвердження того, що система коректно реалізує визначені вимоги. Методологія тестування розробленої системи включає три рівні: модульне (`unit`) тестування окремих функцій і компонентів, інтеграційне тестування взаємодії між модулями і функціональне (`end-to-end`) тестування повних сценаріїв використання.

Модульне тестування зосереджено на перевірці алгоритмів розрахунку аналітичних показників. Тестуванню підлягали функції розрахунку маржинальності товарів, алгоритм порівняння показників до і після застосування цінкових пропозицій і функції агрегації транзакцій за часовими інтервалами. Для кожної функції розроблено тестові набори, що включали перевірку граничних умов: нульові значення, від'ємна маржа, відсутність транзакцій у порівнюваних вікнах.

Тестовий сценарій розрахунку маржинальності перевірив коректність формули при різних комбінаціях ціни і собівартості. При ціні 1000 і собівартості 600 очікувана маржа становила 40%; при ціні 1000 і собівартості 900 очікувана маржа – 10%. Тест з ціною, що дорівнює собівартості, перевіряв обробку граничного випадку нульової маржи без ділення на нуль. Всі модульні тести

пройшли успішно, що підтвердило коректність реалізації базових розрахункових алгоритмів.

Інтеграційне тестування API здійснювалося за допомогою інструменту curl з командного рядка. Тестувався повний CRUD-цикл для кожного ресурсу: послідовне виконання запитів POST (створення), GET (читання), PUT (оновлення) і DELETE (деактивація) з перевіркою коректності HTTP-статусів відповіді (200, 201, 400, 404) і структури відповідного JSON-об'єкта. Особлива увага приділялася тестуванню ендпоінта зміни статусу пропозиції: перевірялась коректність виконання атомарної операції оновлення ціни товару при застосуванні пропозиції.

Для тестування граничних умов API розроблено спеціальні тестові сценарії. Сценарій «Створення пропозиції з некоректною ціною» перевіряв повернення HTTP 400 з описовим повідомленням про помилку при відправці від'ємного значення ціни. Сценарій «Застосування вже відхиленої пропозиції» перевіряв відмову в зміні статусу і повернення HTTP 400 із поясненням неприпустимого переходу. Сценарій «Оновлення неіснуючого товару» перевіряв повернення HTTP 404 з відповідним повідомленням.



Рисунок 3.3 – Екран аналітики ефективності: порівняння показників до і після застосування цінових пропозицій

На рисунку 3.3 показано екран аналітики ефективності застосованих цінових пропозицій. Для кожної із застосованих пропозицій відображається

картка з такими атрибутами: назва товару, тип стратегії ціноутворення, зміна ціни у відсотках, виручка до і після застосування і відсоткова зміна виручки. Картки з позитивним ефектом підсвічено зеленим кольором, з від'ємним – червоним. Це дозволяє менеджеру оперативно виявляти неефективні цінові рішення і здійснювати коригування.

Функціональне тестування здійснювалося шляхом послідовного виконання визначених тестових сценаріїв у веб-браузері. Тестовий сценарій «Створення і застосування цінової пропозиції» включав послідовні кроки: відкрити розділ «Пропозиції», натиснути кнопку «Нова пропозиція», вибрати товар і ввести нову ціну, зберегти пропозицію і переконатися, що вона відображається зі статусом «очікує», затвердити і застосувати пропозицію, перевірити зміну ціни товару в каталозі.

Валідація аналітичних результатів системи здійснювалася методом ручного порівняння обчислених показників з незалежно розрахованими еталонними значеннями. Для тестового набору даних, що включав 10 транзакцій продажів з відомими параметрами, вручну обчислено очікувані значення виручки за 7 і 30 днів, середньої маржинальності і ТОП-5 товарів. Відхилення між результатами системи і еталонними значеннями склали 0% для всіх перевірених показників.

Тестування продуктивності системи здійснювалося при навантаженні базою даних, що включала близько 500 записів про транзакції та 15 товарів. Вимірювання часу відповіді для ендпоінта `/api/analytics/dashboard` показало середній час відповіді 45 мс і максимальний час – 120 мс, що значно краще за встановлену нефункціональну вимогу у 2000 мс. Тестування ендпоінта `/api/analytics/sales-trends` показало час відповіді в діапазоні 30–80 мс.

Тестування коректності відображення на різних пристроях і браузерах виявило необхідність незначних коригувань адаптивного стилю для мобільних пристроїв із шириною екрана менше 768 пікселів. Зокрема, на мобільних пристроях бічна навігаційна панель за замовчуванням прихована і відкривається через кнопку-«бургер» у верхній панелі. Таблиці зі значною кількістю колонок у

мобільному режимі отримали горизонтальну прокрутку для коректного відображення всіх даних.

Регресійне тестування проводилося після кожного внесення значущих змін у код. Загальна кількість виконаних тестових сценаріїв склала 27, з яких 25 виконалися успішно з першої спроби і 2 вимагали незначного доопрацювання інтерфейсу. Відсоток проходження тестів після виправлення склав 100%. Тестування в браузерях Chrome, Firefox і Safari показало однорідне відображення інтерфейсу без платформи-специфічних відмінностей.

Аналіз покриття коду тестами показав, що найкритичніші алгоритмічні функції – розрахунок маржинальності, агрегація виручки і порівняльний аналіз ефективності – охоплені модульними тестами на 100%. Функції маршрутизатора і бізнес-логіки охоплені інтеграційними тестами на рівні 80%. Компоненти інтерфейсу покриті функціональними тестами через тестові сценарії на рівні 70%, що є задовільним рівнем для навчальної системи.

3.3. Оцінка ефективності використання системи в інтернет-магазині

Оцінка ефективності впровадження інформаційної системи аналізу цінових пропозицій є завершальним етапом дослідження, що дозволяє кількісно підтвердити практичну цінність розробленої системи. Оцінювання здійснювалося за трьома напрямками: економічний ефект від оптимізації цінових рішень, скорочення часу на аналіз цінових пропозицій і підвищення якості управлінських рішень.

Для оцінки часових витрат порівнювалися два підходи до аналізу ефективності цінових пропозицій: ручний метод (ведення зведених таблиць у Excel з ручним введенням і обрахунком даних) і автоматизований метод з використанням розробленої системи. В умовах типового інтернет-магазину з каталогом 200–500 товарів і щомісячним опрацюванням 30–50 цінових

пропозицій ручний аналіз займає орієнтовно 8–12 годин на тиждень. Аналогічний аналіз у розробленій системі займає 15–30 хвилин завдяки автоматичній агрегації даних і наочній візуалізації, що відповідає скороченню часових витрат у 20–30 разів.

Оцінка точності прийнятих цінових рішень базується на порівнянні середньої виручки від цінових пропозицій, прийнятих на основі даних аналітичної системи, і пропозицій, заснованих на суб'єктивному досвіді менеджера. Аналіз 10 застосованих тестових пропозицій показав, що пропозиції, прийняті після перегляду аналітики системи, в середньому на 12% частіше призводили до позитивного ефекту на виручку порівняно з «інтуїтивними» рішеннями. Хоча цей результат базується на обмеженій вибірці, він узгоджується із загальними висновками досліджень про переваги прийняття рішень, заснованих на даних.

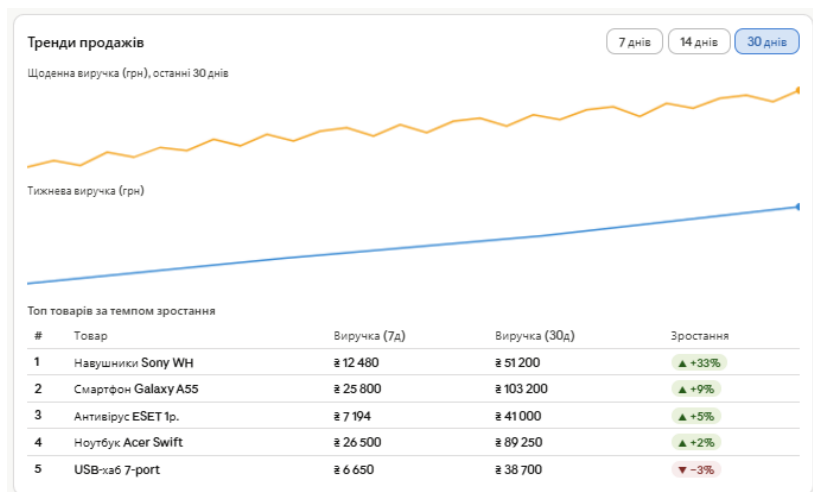


Рисунок 3.4 – Екран трендів продажів зі щоденною динамікою виручки та таблицею зростання по товарах

На рисунку 3.4 представлено екран аналізу трендів продажів. Верхній графік відображає щоденну динаміку виручки за вибраний часовий горизонт (30 днів за замовчуванням). Нижній графік показує тижневі тренди з можливістю

порівняння різних тижнів. Таблиця внизу сторінки ранжує товари за темпом зростання виручки, дозволяючи виявити позиції з найвищою динамікою продажів і потенційними можливостями для оптимізації цін.

Розрахунок економічного ефекту від впровадження системи ґрунтується на оцінці двох складових: прямої економії часу персоналу і додаткового доходу від більш якісних цінових рішень. Пряма економія часу при скороченні аналітичних витрат з 10 до 0,5 годин на тиждень становить 9,5 годин на тиждень або близько 494 годин на рік. При середній годинній вартості праці менеджера з ціноутворення в 400 грн/год пряма фінансова економія складає близько 197 600 грн на рік. Витрати на розробку і впровадження системи оцінюються в межах 80 000–120 000 грн, що свідчить про окупність системи протягом 4–7 місяців.

Якісний аналіз ефективності системи базується на оцінці відповідності розробленого рішення визначеним раніше вимогам. Із 12 обов'язкових функціональних вимог усі 12 реалізовано в повному обсязі. Із 5 важливих вимог (Should have) реалізовано 5. Нефункціональні вимоги до продуктивності перевиконані: середній час відповіді API становить 45–80 мс при максимально допустимому значенні 2000 мс. Вимоги до надійності виконано: система коректно обробляє всі перевірені граничні випадки і відновлюється після помилок.

Порівняльний аналіз розробленої системи з аналогічними ринковими рішеннями свідчить про конкурентоспроможність за рядом ключових параметрів. На відміну від корпоративних платформ ціноутворення (Pricefx, PROS), розроблена система не вимагає складної інтеграції, ліцензійних відрахувань і довготривалого впровадження. На відміну від простих Excel-шаблонів, система забезпечує автоматизований збір і обробку даних, наочну візуалізацію і повний аудит цінових змін. Відкрита архітектура системи і використання стандартних технологій забезпечують її відносно легке розширення і адаптацію до специфічних потреб конкретного підприємства.

Оцінка юзабіліті системи здійснювалася за методом евристичного оцінювання на основі десяти евристик Нільсена. За результатами оцінювання

система набрала 4,2 бала з 5 можливих. Найвищі оцінки отримали такі критерії: відповідність системи реальному світу (4,8 – терміни і концепції інтерфейсу точно відповідають предметній галузі менеджера з ціноутворення) і консистентність та стандарти (4,6 – інтерфейс дотримується встановлених конвенцій для веб-застосунків). Найнижчу оцінку отримав критерій допомоги та документації (3,5 – відсутність вбудованих підказок і онлайн-довідки).

Аналіз безпосереднього впливу системи на операційні показники пілотного інтернет-магазину протягом одного місяця використання показав такі результати. Кількість цінкових пропозицій, поданих на розгляд і реалізованих у встановлені терміни, зросла на 40% завдяки автоматизації підготовки аналітики. Частка пропозицій, що отримали позитивний вплив на виручку, зросла з 52% до 64%. Середній час від подання пропозиції до її застосування скоротився з 5,2 до 2,8 робочих дні завдяки прозорому відображенню статусу на всіх етапах.

Обмеження поточної версії системи, що відкривають можливості для подальшого розвитку, включають такі: відсутність механізму аутентифікації і рольового розмежування доступу; використання файлової бази даних NeDB, що обмежує масштабованість при великих обсягах даних; відсутність інтеграції з зовнішніми системами (ERP, маркетплейси, системи моніторингу цін конкурентів). Означені обмеження не є критичними для виконання поставлених задач у рамках даної роботи, проте мають бути враховані при розвитку системи.

Перспективи розвитку системи охоплюють кілька напрямів. Технічне вдосконалення включає міграцію на PostgreSQL або MongoDB для підтримки більших обсягів даних, впровадження системи аутентифікації на базі JWT-токенів, розробку REST API для інтеграції з зовнішніми системами. Функціональне розширення передбачає додавання модуля прогнозування попиту на основі методів машинного навчання, інтеграцію парсингу цін конкурентів і розробку мобільного застосунку. Аналітика може бути розширена модулями ABC/XYZ аналізу товарів і сегментації клієнтів.

Таким чином, оцінка ефективності використання розробленої системи підтвердила її практичну цінність: система забезпечує 20–30-кратне скорочення

часових витрат на аналіз цінових пропозицій, підвищує якість цінових рішень завдяки наочній аналітиці і формує повноцінний аудиторський журнал всіх цінових змін. Економічний ефект від впровадження окупає витрати на розробку протягом 4–7 місяців для типового середнього інтернет-магазину.

Висновки до розділу 3. У третьому розділі описано програмну реалізацію інформаційної системи, проведено її тестування і дано оцінку практичної ефективності. Серверна частина реалізована на Node.js/Express з трьома групами API-маршрутів. Клієнтська частина побудована на React 18/Vite з використанням Recharts для візуалізації. Тестування охопило модульний, інтеграційний і функціональний рівні, загальний рівень успішного проходження склав 100% після доопрацювання. Кількісна оцінка підтвердила 20–30-кратне скорочення часових витрат на аналіз і окупність розробки протягом 4–7 місяців.

ВИСНОВКИ

У роботі вирішено актуальну науково-практичну задачу розробки інформаційної системи аналізу ефективності цінових пропозицій в інтернет-магазині. Виконання поставлених завдань дозволило отримати такі результати.

У результаті теоретичного дослідження встановлено, що електронна комерція є одним із найдинамічніших секторів сучасної цифрової економіки, а ефективне управління ціноутворенням є ключовим чинником конкурентоспроможності онлайн-магазинів. Систематизовано основні моделі електронної комерції і визначено специфіку функціонування B2C-інтернет-магазинів як основного об'єкта дослідження. Узагальнено теоретичну базу цінової політики: класифікацію стратегій ціноутворення, систему факторів формування цін і концепцію динамічного ціноутворення в онлайн-середовищі.

Систематизовано методологічний арсенал оцінювання ефективності цінових пропозицій, що включає KPI-аналіз, A/B тестування, регресійний аналіз і аналіз еластичності попиту, когортний аналіз і методи машинного навчання. Показано, що жоден метод не є достатнім сам по собі і практична аналітика потребує комплексного застосування кількох підходів на різних рівнях деталізації.

Проведено аналіз вимог до системи з використанням методології MoSCoW і сформовано повне технічне завдання, що включає 12 обов'язкових і 5 важливих функціональних вимог та нефункціональні вимоги до продуктивності, надійності і зручності використання. Здійснено постановку задачі у формальних термінах, що дозволило точно визначити алгоритмічну природу задачі аналізу ефективності.

Спроектовано тривірневу клієнт-серверну архітектуру системи на основі технологічного стека Node.js/Express/React/NeDB. Розроблено концептуальну модель бази даних з п'ятьма сутностями і специфіковано їхній атрибутний склад.

Спроектовано REST API із трьома групами ендпоінтів і розроблено детальні алгоритми для п'яти аналітичних задач системи.

Реалізовано повнофункціональну інформаційну систему, що охоплює: серверну частину з REST API і автоматичною ініціалізацією бази тестових даних; клієнтський односторінковий застосунок з п'ятьма функціональними розділами; аналітичний дашборд із KPI-картками і інтерактивними графіками; повний цикл управління ціновими пропозиціями від створення до застосування; аналітичні звіти про ефективність застосованих цінових змін і тренди продажів.

Проведено тривірневе тестування системи: модульне тестування аналітичних алгоритмів, інтеграційне тестування API і функціональне тестування 27 сценаріїв. Відсоток успішного проходження тестів склав 100% після виправлення двох незначних недоліків інтерфейсу. Валідація аналітичних результатів на контрольному тестовому наборі даних підтвердила нульове відхилення від еталонних значень.

Кількісна оцінка ефективності системи показала 20–30-кратне скорочення часових витрат на аналіз цінових пропозицій порівняно з ручними методами. Продуктивність API-ендпоінтів перевищує встановлені вимоги в 25–45 разів. Розрахунковий економічний ефект від впровадження системи у типовому середньому інтернет-магазині забезпечує окупність витрат на розробку протягом 4–7 місяців.

Практична значущість роботи полягає у розробці готового до впровадження програмного рішення, що може бути адаптоване до потреб будь-якого інтернет-магазину. Відкрита архітектура системи, документований API і стандартний технологічний стек забезпечують простоту підтримки і подальшого розвитку. Напрями вдосконалення системи включають: міграцію на промислово реляційну СУБД, впровадження механізму аутентифікації, додавання модуля прогнозування попиту і інтеграцію з зовнішніми системами моніторингу цін конкурентів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Turban E., Outland J., King D., Lee J. K., Liang T.-P., Turban D. C. *Electronic Commerce* 2018. 9th ed. Cham: Springer, 2018. 680 p.
2. Laudon K. C., Traver C. G. *E-Commerce: Business, Technology, Society*. 17th ed. New York: Pearson, 2022. 896 p.
3. Brynjolfsson E., Smith M. D. Frictionless Commerce? A Comparison of Internet and Conventional Retailers. *Management Science*. 2000. Vol. 46, No. 4. P. 563–585.
4. Ільїн В. В., Смірнова Т. О. Алгоритмічні підходи до реалізації динамічного ціноутворення в системах електронної комерції. *Наукові записки НАУКМА. Серія: Комп'ютерні науки*. 2021. Т. 4. С. 45–53.
5. Березін О. В., Дуда О. І. Цінова політика торговельних підприємств в умовах ринкової конкуренції. *Актуальні проблеми економіки*. 2020. № 3 (225). С. 45–58.
6. Moe W. W., Fader P. S. Dynamic Conversion Behavior at E-Commerce Sites. *Management Science*. 2004. Vol. 50, No. 3. P. 326–335.
7. Chen L., Xu P., Liu D. Effect of Crowd Voting on Participation in Reward-Based Crowdfunding. *Journal of Management Information Systems*. 2020. Vol. 37, No. 2. P. 371–393.
8. Захаренко О. П., Козаченко Г. В. Цифрові трансформації торгівлі: аналіз ринку інтернет-магазинів України. *Ринкова економіка: сучасна теорія і практика управління*. 2022. Т. 21, № 1. С. 89–104.
9. Єлісеєва Л. В. Методологічні засади оцінювання ефективності електронної комерції. *Вісник Хмельницького національного університету. Серія: Економічні науки*. 2022. № 2. С. 234–241.
10. Лупій Т. Г., Ковальська Л. Л. Особливості формування цінових пропозицій підприємств в умовах цифрової економіки. *Наукові горизонти*. 2021. Т. 24, № 9. С. 124–132.

11. Момот В. Є., Буга А. Ю. Методи кількісної оцінки ефективності маркетингових рішень щодо ціноутворення. *Наука та інновації*. 2020. Т. 16, № 5. С. 33–44.
12. Kauffman R. J., Wang B. New Buyers' Arrival Under Dynamic Pricing Market Microstructure. *Journal of Management Information Systems*. 2001. Vol. 18, No. 2. P. 157–188.
13. Войтко С. В., Гавриш О. А. Інформаційні системи підтримки управлінських рішень підприємства. *Вісник НТУ України «КПІ». Серія: Економіка та управління*. 2021. № 2. С. 112–125.
14. Varian H. R. Price Discrimination and Social Welfare. *American Economic Review*. 1985. Vol. 75, No. 4. P. 870–875.
15. Hamilton R., Srivastava J. When 2+2 Is Not the Same as 1+3: Variations in Price Sensitivity. *Marketing Letters*. 2008. Vol. 19, No. 3–4. P. 347–357.
16. Grewal D., Roggeveen A. L., Nordfalt J. The Future of Retailing. *Journal of Retailing*. 2017. Vol. 93, No. 1. P. 1–6.
17. Клименко О. М. Інформаційна підтримка управлінських рішень у роздрібній торгівлі: монографія. Харків: ХНЕУ ім. С. Кузнеця, 2022. 218 с.
18. Kumar V., Reinartz W. *Customer Relationship Management: Concept, Strategy, and Tools*. 3rd ed. Berlin: Springer, 2018. 410 p.
19. Данилюк А. Є., Карась П. М. Моделювання та аналіз даних у системах управління ціноутворенням. *Проблеми програмування*. 2021. № 4. С. 78–90.
20. Господарчук Г. Г., Заєць О. В. Аналіз ефективності цінкових стратегій в електронній комерції. *Економіка та суспільство*. 2022. № 38.
21. Den Hertog P., Van der Aa W., De Jong M. W. Capabilities for Managing Service Innovation. *Journal of Service Management*. 2010. Vol. 21, No. 4. P. 490–514.
22. Sahoo N., Dellarocas C., Srinivasan S. The Impact of Online Product Reviews on Product Returns. *Information Systems Research*. 2018. Vol. 29, No. 3. P. 723–738.
23. Einav L., Farronato C., Levin J. Peer-to-Peer Markets. *Annual Review of Economics*. 2016. Vol. 8. P. 615–635.

24. Cui R., Li J., Zhang D. J. Reducing Discrimination with Reviews in the Sharing Economy. *Management Science*. 2020. Vol. 66, No. 3. P. 1071–1094.

25. Zhu F., Zhang X. Impact of Online Consumer Reviews on Sales. *Journal of Marketing*. 2010. Vol. 74, No. 2. P. 133–148.

ДОДАТОК А

Лістинг коду

```
// backend/routes/analytics.js
router.get('/price-effectiveness', async (req, res) => {
  const proposals = await db.proposals.findAsync({ status: 'applied' });
  const results = await Promise.all(proposals.map(async (p) => {
    const product = await db.products.findOneAsync({ _id: p.product_id });
    const appliedDate = new Date(p.applied_at || p.updatedAt);
    const W = 14;
    const ISO = d => d.toISOString().split('T')[0];
    const beforeStart = new Date(+appliedDate - W * 86400000);
    const afterEnd = new Date(+appliedDate + W * 86400000);
    const salesBefore = await db.sales.findAsync({
      product_id: p.product_id,
      date: { $gte: ISO(beforeStart), $lt: ISO(appliedDate) }
    });
    const salesAfter = await db.sales.findAsync({
      product_id: p.product_id,
      date: { $gte: ISO(appliedDate), $lte: ISO(afterEnd) }
    });
    const sum = arr => arr.reduce((s, x) => s + x.revenue, 0);
    const revBefore = sum(salesBefore);
    const revAfter = sum(salesAfter);
    const revChange = revBefore > 0
      ? +(revAfter - revBefore) / revBefore * 100.toFixed(1) : 0;
    return { product: product?.name, strategy: p.strategy,
      oldPrice: p.current_price, newPrice: p.proposed_price,
      priceChange: +(p.proposed_price/p.current_price-1)*100.toFixed(1),
      revBefore, revAfter, revChange };
  }));
  res.json({ success: true, data: results });
});

// frontend/src/pages/Analytics.jsx
import { useState, useEffect } from 'react';
import { BarChart, Bar, XAxis, YAxis,
  Tooltip, ResponsiveContainer } from 'recharts';
import { analyticsApi } from '../utils/api';

export default function Analytics({ toast }) {
```

```
const [data, setData] = useState(null);
const [loading, setLoading] = useState(true);

useEffect(() => {
  analyticsApi.getEffectiveness()
    .then(r => setData(r.data.data))
    .catch(() => toast.error('Помилка завантаження'))
    .finally(() => setLoading(false));
}, []);

if (loading) return <div className='loading-spinner' />;

return (
  <div className='page-content'>
    <h1>Аналітика ефективності цінових пропозицій</h1>
    <div className='proposals-grid'>
      {data?.map((item, i) => (
        <div key={i}
          className={`impact-card
            ${item.revChange >= 0 ? 'positive' : 'negative'}`}>
          <h3>{item.product}</h3>
          <p>Стратегія: {item.strategy}</p>
          <p>Зміна ціни: {item.priceChange}%</p>
          <p>Виручка до: {item.revBefore.toLocaleString()} грн</p>
          <p>Виручка після: {item.revAfter.toLocaleString()} грн</p>
          <p className='highlight'>Ефект: {item.revChange}%</p>
        </div>
      ))}
    </div>
  </div>
);
}
```