

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ**

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

Кваліфікаційна робота

на тему: **«ІНТЕРНЕТ МАГАЗИН ОДЯГУ НА ОСНОВІ ВЕБ-ФРЕЙМВОРКУ
DJANGO (PYTHON)»**

на здобуття освітнього ступеня магістра

зі спеціальності 122 Комп'ютерні науки

(код, найменування спеціальності)

освітньо-професійної програми Комп'ютерні науки

(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело*

Владислав ТАРАН

(підпис)

Ім'я, ПРІЗВИЩЕ здобувача

Виконав: здобувач вищої освіти гр. КНДМ - 63

Владислав ТАРАН

(Ім'я, ПРІЗВИЩЕ)

Керівник: к. т. н., Віталій КОТЕЛЯНЕЦЬ

Науковий ступінь,
вчене звання

(Ім'я, ПРІЗВИЩЕ)

Рецензент: _____

Науковий ступінь,
вчене звання

(Ім'я, ПРІЗВИЩЕ)

Київ – 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально–науковий інститут інформаційних технологій

Кафедра Комп'ютерних наук

Ступінь вищої освіти – «Магістр»

Спеціальність підготовки – «122 Комп'ютерні науки»

Освітньо-професійна програма – «Комп'ютерні науки»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Комп'ютерних наук

“ ” _____ Віктор Вишнівський
_____ 2024 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Таран Владислав Петрович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: «Інтернет магазин одягу на основі веб-фреймворку Django (Python)»

Керівник кваліфікаційної роботи Віталій Котолянець, к.т.н., доцент.
(ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 320 від 15.10.2024р.

2. Строк подання студентом роботи 15.12.2024 р.

3. Вихідні дані до роботи: Параметри веб-сервера та бази даних, Архітектура веб-додатка, Кешування та оптимізація продуктивності, Мережеві протоколи та безпека, Науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Сучасні рішення для створення інтернет-магазинів на основі веб-фреймворку Django.

2. Архітектура веб-додатків та переваги використання Django у веб-розробленні.

3. Аналіз ринку електронної комерції та основних технологічних трендів.

4. Процес проєктування інтернет-магазину: функціональні можливості та структура.

5. Створення бази даних інтернет-магазину з використанням ORM Django.

6. Впровадження математичних моделей для аналізу клієнтської поведінки.

-
7. Забезпечення безпеки даних та захисту від несанкціонованого доступу.
-
8. Рекомендації щодо подальшої оптимізації та розвитку інтернет-магазину.
-
9. Висновки за результатами розробки системи.
-
5. Перелік ілюстративного матеріалу: *презентація*
1. Мета роботи, об'єкт та предмет дослідження
 2. Архітектурна модель веб-додатка на основі Django
 3. Структура бази даних інтернет-магазину
 4. Процес інтеграції платіжних систем у систему
 5. Оптимізація продуктивності веб-додатка
 6. Впровадження математичних моделей для аналізу продажів
 7. Алгоритми прогнозування продажів на основі статистичних даних
 8. Дослідження та рекомендації щодо варіантів логічного розгортання безпроводових мереж - гостьові VLAN
 9. Системи захисту даних та механізми безпеки
 10. Висновки та подальші перспективи розвитку проєкту
6. Дата видачі завдання 05.09.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	09.10.2024	вик.
2	Сучасні рішення для веб-розроблення інтернет-магазинів на основі фреймворку Django	23.10.2024	вик.
3	Проектування та моделювання структури бази даних для інтернет-магазину	04.11.2024	вик.
4	Реалізація інтеграції платіжних систем і забезпечення безпеки даних	22.11.2024	вик.
5	Оптимізація продуктивності системи та масштабування веб-додатка	02.12.2024	вик.
6	Вступ, висновки, реферат	10.12.2024	вик.
7	Розробка демонстраційних креслень	14.12.2024	вик.

Здобувач вищої освіти _____
(підпис)

Владислав ТАРАН _____
(Ім'я, ПРІЗВИЩЕ)

Керівник
кваліфікаційної роботи _____
(підпис)

Віталій КОТЕЛЯНЕЦЬ _____
(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 70 стор., 8 рис., 20 джерел.

Мета роботи – створення теоретичної моделі інтернет-магазину.

Об'єкт дослідження – процес створення інтернет-магазину з використанням веб-фреймворку Django.

Предмет дослідження – функціональні можливості інтернет-магазину з використанням веб-фреймворку Django.

Короткий зміст роботи:

Магістерська робота присвячена розробці інтернет-магазину одягу на основі веб-фреймворку Django (Python). У роботі розглянуто актуальність створення інтернет-магазинів в умовах розвитку електронної комерції та сучасні технологічні підходи до їх реалізації. Розроблена система включає функціонал для управління товарами, інтеграцію з платіжними сервісами, зручний інтерфейс для користувачів та високий рівень захисту даних. Особливу увагу приділено оптимізації продуктивності та масштабованості системи з використанням можливостей Django. У роботі також впроваджено математичні моделі для аналізу клієнтської поведінки та прогнозування продажів, що підвищує ефективність бізнес-процесів. Результатом є функціональний інтернет-магазин, що відповідає сучасним вимогам ринку та готовий до подальшого розвитку.

КЛЮЧОВІ СЛОВА: ІНТЕРНЕТ-МАГАЗИН, DJANGO, ВЕБ-РОЗРОБЛЕННЯ, ЕЛЕКТРОННА КОМЕРЦІЯ, ПРОГНОЗУВАННЯ ПРОДАЖІВ, ЗАХИСТ ДАНИХ.

ABSTRACT

Text part of the master's qualification work: 70 pages, 8 pictures, 20 sources.

The purpose of the work – to create a theoretical model of an online store.

Object of research – the process of creating an online store using the Django web framework.

Subject of research – the functional capabilities of an online store using the Django web framework.

Summary of the work:

The master's thesis is dedicated to the development of an online clothing store based on the Django (Python) web framework. The work examines the relevance of creating online stores in the context of the growth of e-commerce and modern technological approaches to their implementation. The developed system includes functionality for product management, integration with payment services, a user-friendly interface, and a high level of data protection. Particular attention is paid to optimizing system performance and scalability using Django's capabilities. The work also implements mathematical models for analyzing customer behavior and predicting sales, which enhance business process efficiency. The result is a functional online store that meets current market requirements and is ready for further development.

KEYWORDS: ONLINE STORE, DJANGO, WEB DEVELOPMENT, E-COMMERCE, SALES FORECASTING, DATA PROTECTION.

ЗМІСТ

ВСТУП.....	9
1 АНАЛІЗ АКТУАЛЬНОСТІ ТА ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
1.1 Актуальність створення інтернет-магазинів	11
1.2 Аналіз ринку електронної комерції.....	15
1.3 Технологічні тренди у веб-розробленні.....	19
1.4 Висновки по розділу	25
2 ТЕОРЕТИЧНА МОДЕЛЬ СТВОРЕННЯ ІНТЕРНЕТ-МАГАЗИНУ.....	26
2.1 Архітектура веб-додатків	26
2.2 Особливості використання Django	30
2.3 Моделювання бази даних у Django	34
2.4 Висновки по розділу	40
3 ОПИС ПОКРАЩЕНЬ ДЛЯ ІСНУЮЧИХ ІНТЕРНЕТ-МАГАЗИНІВ.....	42
3.1 Оптимізація користувацького досвіду	42
3.2 Забезпечення масштабованості з використанням Django	45
3.3 Використання інноваційних технологій у Django-проектах	49
3.4 Висновки по розділу	54
4 МАТЕМАТИЧНА МОДЕЛЬ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ.....	56
4.1 Алгоритми аналізу даних у комерційних системах.....	56
4.2 Прогнозування продажів на основі статистичних даних.....	59
4.3 Інтеграція математичних моделей у веб-додаток на Django	64
4.4 Висновки по розділу	67
ВИСНОВКИ.....	69
ПЕРЕЛІК ПОСИЛАНЬ.....	71
ДОДАТОК А. ПРЕЗЕНТАЦІЯ.....	73

ВСТУП

Електронна комерція сьогодні є одним із найбільш перспективних напрямків розвитку цифрових технологій. Її вплив на сучасну економіку не можна переоцінити, адже саме через онлайн-торгівлю щодня здійснюються мільйони операцій, що забезпечують зручність для споживачів та ефективність для бізнесу. Особливо активно розвивається ринок одягу, де інтернет-магазини стають головним каналом продажів, пропонуючи споживачам широкий асортимент товарів, доступність цін та швидкість обслуговування. Сучасний ринок електронної комерції характеризується високою конкуренцією, де основними викликами стають не лише залучення клієнтів, а й забезпечення високої якості обслуговування, безпеки даних та гнучкості платформ. У таких умовах використання передових технологій розробки, як-от Django, стає необхідністю. Django, створений як фреймворк для швидкої та якісної розробки веб-додатків, забезпечує всі необхідні інструменти для розробки інтернет-магазинів, які відповідають сучасним вимогам. Використання Django дає змогу забезпечити інтегровані рішення, що включають ефективну роботу з базами даних, гнучке налаштування інтерфейсів, реалізацію безпечної авторизації та масштабування систем. Його переваги особливо очевидні у сфері електронної комерції, де швидкість розробки, стабільність роботи та можливість впровадження інновацій мають вирішальне значення. Крім того, Django надає можливість створення масштабованих рішень, які відповідають потребам великих компаній та стартапів.

Однією з найважливіших переваг Django є його багатий набір вбудованих функцій, які дозволяють вирішувати широкий спектр завдань у межах єдиної екосистеми. Наприклад, використання вбудованого адміністративного інтерфейсу значно полегшує управління даними та конфігурацію системи. Крім того, інструменти для роботи з аутентифікацією користувачів забезпечують високий рівень захисту персональних даних, що є критично важливим для успішної роботи інтернет-магазинів. Це дозволяє зосередитися на бізнес-логіці та користувацькому досвіді без зайвих витрат часу на вирішення базових технічних завдань. Розробка інтернет-магазину на основі Django передбачає вирішення ключових завдань, таких як забезпечення швидкого доступу до товарів, зручної системи пошуку та фільтрації, надійного механізму оплати та обробки замовлень. Особливу роль у цьому відіграє моделювання бази даних, яка є основою для забезпечення інтеграції всіх компонентів системи та забезпечення її безперебійної роботи. Django пропонує вбудовані інструменти для роботи з базами даних, що дозволяють оптимізувати процес розробки та зменшити витрати часу.

Об'єктом дослідження є процес створення інтернет-магазину з використанням веб-фреймворку Django. Він охоплює архітектуру системи, функціональні можливості платформи та методи її адаптації до сучасних викликів електронної комерції. У фокусі дослідження перебувають питання інтеграції системи оплати, побудови інтуїтивно зрозумілого інтерфейсу для користувачів та забезпечення високого рівня захисту даних.

Метою цієї роботи є створення теоретичної моделі інтернет-магазину, що відповідає сучасним стандартам продуктивності, зручності та безпеки. У межах дослідження передбачено детальний аналіз можливостей Django, проектування структури системи та розробка рекомендацій щодо її вдосконалення.

Робота базується на комплексному підході, що включає аналітичні дослідження, проектування моделей та застосування статистичного аналізу. Зокрема, буде розроблено математичну модель прогнозування продажів, яка дозволить підприємствам ефективніше планувати свою діяльність. Крім того, у роботі буде розглянуто можливості інтеграції машинного навчання для підвищення точності рекомендацій та оптимізації користувацького досвіду. Іншим аспектом є впровадження технологій масштабованості, які дозволяють адаптувати платформу до зростаючого навантаження в умовах розширення бізнесу. Наприклад, використання систем кешування та оптимізації запитів до бази даних дозволяє значно підвищити продуктивність інтернет-магазину. У свою чергу, інтеграція зовнішніх API для автоматизації процесів, таких як відстеження доставок чи управління акціями, розширює функціональність системи та підвищує її конкурентоспроможність.

Очікується, що результати дослідження сприятимуть підвищенню ефективності інтернет-магазинів, забезпечуючи відповідність сучасним потребам споживачів і бізнесу, а також відкриватимуть нові горизонти для використання Django у інших сферах цифрової економіки.

1 АНАЛІЗ АКТУАЛЬНОСТІ ТА ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Актуальність створення інтернет-магазинів

Створення інтернет-магазинів є актуальним напрямком розвитку бізнесу в умовах цифрової трансформації, яка активно змінює традиційні економічні моделі. Широке поширення інтернету та зростання доступності мобільних пристроїв суттєво вплинуло на зміну поведінки споживачів, зробивши онлайн-шопінг більш зручним та доступним для широких верств населення. Онлайн-торгівля створює нові можливості для бізнесу завдяки мінімізації витрат на фізичну інфраструктуру, забезпеченню більшої гнучкості та оперативності в обслуговуванні клієнтів. Інтернет-магазини допомагають подолати географічні обмеження, дозволяючи компаніям виходити на нові регіональні та міжнародні ринки. Це особливо важливо для малого та середнього бізнесу, який завдяки цифровим платформам отримує інструменти для ефективного ведення конкурентної боротьби навіть на глобальному рівні. Підприємства можуть реалізовувати унікальні бізнес-моделі, такі як пряма співпраця з виробниками або персоналізовані пропозиції для клієнтів на основі аналізу їхніх уподобань і поведінки. Суттєвим стимулом для розвитку онлайн-торгівлі є також зростаюча довіра споживачів до електронних платежів і розширення асортименту товарів, доступних в інтернет-магазинах. Багато споживачів вважають за краще здійснювати покупки в інтернеті через зручність, можливість легко порівнювати ціни та отримувати рекомендації, засновані на індивідуальних уподобаннях. Такі технології, як доповнена реальність (AR), чат-боти з елементами штучного інтелекту та інтеграція з соціальними мережами, сприяють створенню унікального користувацького досвіду та збільшують рівень задоволеності клієнтів.

Інтернет-магазини стали важливою частиною сучасної економіки, пропонуючи компаніям значні можливості для оптимізації операційних витрат. Зменшення витрат на фізичну інфраструктуру, включаючи оренду приміщень, утримання складів і персоналу, дозволяє бізнесам інвестувати в розвиток онлайн-

каналів збуту та впровадження інновацій. Це особливо актуально для малого та середнього бізнесу, який може розширити свою клієнтську базу без значних капіталовкладень, використовуючи доступні цифрові платформи. Більше того, інтернет-магазини відкривають доступ до глобальних ринків, що раніше було прерогативою лише великих корпорацій. Завдяки цьому бізнеси можуть продавати свої товари і послуги за межами локального ринку, пристосовуючись до специфічних потреб різних регіонів. Використання електронної комерції дозволяє не тільки залучати нових клієнтів, але й зменшувати залежність від традиційних каналів збуту, що забезпечує більшу стійкість до економічних змін.

Попит на інтернет-магазини постійно зростає завдяки їх зручності та доступності, що стало важливим фактором у формуванні сучасних споживчих звичок. Споживачі цінують можливість переглядати товари, читати відгуки, порівнювати ціни та здійснювати покупки без необхідності виходити з дому, що особливо важливо в умовах обмежень або браку часу. Зручність інтернет-магазинів також посилюється можливістю швидкого доступу до широкого асортименту товарів, який часто перевищує вибір у звичайних фізичних магазинах.

Цифрові платформи також відкривають нові можливості для компаній у впровадженні персоналізованих пропозицій, що базуються на детальному аналізі даних про поведінку користувачів. Використання алгоритмів штучного інтелекту дозволяє формувати індивідуальні рекомендації, які враховують попередні покупки, уподобання та інтереси клієнтів. Це значно підвищує рівень задоволеності клієнтів і сприяє збільшенню їхньої лояльності. Більше того, персоналізація стимулює повторні покупки, що забезпечує стабільний дохід для бізнесу.

Окрім цього, інтернет-магазини активно використовують інтеграцію з іншими цифровими платформами, такими як соціальні мережі, для залучення нових клієнтів та підвищення обізнаності про бренд. Це створює додаткові канали комунікації зі споживачами, сприяє побудові довготривалих відносин та підвищує конкурентоспроможність на ринку.

Ще одним фактором актуальності створення інтернет-магазинів є зміна глобальних економічних умов. Пандемія COVID-19 змусила багато бізнесів перейти в онлайн, що суттєво прискорило розвиток електронної комерції. Це створило сприятливі умови для інноваційних рішень, таких як інтеграція з мобільними додатками, використання штучного інтелекту для прогнозування продажів і автоматизація логістичних процесів.

Значний внесок у розвиток інтернет-магазинів зробила також поява спеціалізованих платформ і фреймворків для розробки. Такі інструменти, як Django, надають розробникам потужні засоби для швидкого створення та масштабування веб-додатків. Django дозволяє інтегрувати бази даних, автоматизувати обробку запитів, забезпечувати високий рівень безпеки та створювати адміністративні панелі без необхідності витратити ресурси на розробку з нуля. Це дає змогу компаніям скоротити час на запуск продукту, одночасно підвищуючи його якість і надійність. Крім того, такі інструменти пропонують інтеграцію з сучасними технологіями, як-от API, системи управління контентом та інструменти аналітики, що сприяє створенню комплексних і гнучких рішень. Це дозволяє бізнесам легко адаптувати свої інтернет-магазини до нових викликів ринку, впроваджувати нові функції та підтримувати високу продуктивність систем навіть при зростаючому навантаженні. У результаті, використання таких платформ сприяє зосередженню компаній на стратегічному розвитку та маркетингових ініціативах, залишаючи технічну реалізацію на надійних технологічних основах. Окрім технічних переваг, інтернет-магазини відкривають нові горизонти для маркетингових кампаній, дозволяючи компаніям більш точно орієнтуватися на своїх клієнтів. Завдяки аналітичним інструментам, компанії можуть ефективніше планувати рекламні стратегії, використовувати таргетинг, сегментувати аудиторію за демографічними та поведінковими характеристиками, а також оцінювати ефективність кожної кампанії в режимі реального часу. Це забезпечує можливість швидко вносити корективи у стратегії для максимального результату.

Інноваційні технології, такі як доповнена реальність (AR), відіграють ключову роль у створенні інтерактивного користувацького досвіду. Наприклад, AR дозволяє клієнтам віртуально приміряти одяг, що знижує кількість повернень і підвищує довіру до бренду. Віртуальні шоуруми стають важливим інструментом для залучення клієнтів, особливо в умовах відсутності фізичних магазинів.

Інтеграція з чат-ботами на основі штучного інтелекту дозволяє автоматизувати взаємодію зі споживачами, оперативно відповідати на запити та пропонувати персоналізовані рекомендації. Це сприяє підвищенню задоволеності клієнтів і створює конкурентну перевагу для компаній, які використовують такі технології.

Економічна ефективність інтернет-магазинів також сприяє розвитку різних моделей співпраці між виробниками та споживачами, забезпечуючи прозорість і зручність у взаємодії. Наприклад, використання моделі прямого продажу (direct-to-consumer) дозволяє уникнути посередників, що зменшує витрати на логістику, зберігання та реалізацію товарів. Для клієнтів це означає доступ до більш вигідних цін, а для виробників — підвищення контролю над процесом продажу та можливість створення персоналізованих пропозицій. Розвиток цифрових каналів дозволяє впроваджувати інноваційні підходи, такі як краудсорсинг і прямі замовлення. Наприклад, платформи з моделлю передзамовлень (pre-order) дозволяють виробникам краще прогнозувати попит і уникати надлишкового виробництва, що знижує витрати та сприяє сталому розвитку. З іншого боку, клієнти отримують можливість брати участь у формуванні асортименту, що підвищує їхню залученість і лояльність до бренду.

Цифрові інструменти, такі як платіжні системи, CRM і аналітичні платформи, дозволяють оптимізувати управління замовленнями, підвищувати швидкість обслуговування та забезпечувати якісний зворотний зв'язок. Завдяки цьому бізнес може гнучко реагувати на зміни попиту, впроваджувати нові рішення та ефективно масштабувати свою діяльність.

У підсумку, розвиток інтернет-магазинів є невід’ємною складовою сучасного бізнесу, що дозволяє компаніям адаптуватися до нових викликів, підвищувати ефективність і задовольняти потреби клієнтів на найвищому рівні. Це підтверджує актуальність даного дослідження та необхідність пошуку інноваційних підходів до створення таких платформ.

1.2 Аналіз ринку електронної комерції

Ринок електронної комерції сьогодні є одним із ключових драйверів глобальної економіки, демонструючи стійке зростання завдяки швидкому поширенню цифрових технологій. Цей сектор характеризується високим рівнем інноваційності, що змінює традиційні моделі бізнесу, оптимізує взаємодію зі споживачами та створює нові можливості для розвитку підприємств різних масштабів. Зростання глобального доступу до високошвидкісного інтернету, масове впровадження мобільних пристроїв та інтеграція багатофункціональних цифрових платформ стали основою для масштабного розвитку онлайн-торгівлі. Такі зміни сприяли активному залученню нових учасників ринку, збільшенню обсягів продажів та формуванню нових стандартів споживання.

Згідно зі статистичними дослідженнями, обсяги світового ринку електронної комерції щороку зростають на 8–12%, що свідчить про стійкий розвиток галузі навіть в умовах економічної нестабільності. Основні сегменти включають продаж товарів повсякденного вжитку, одягу, електроніки, послуг і цифрових продуктів. Зокрема, сектор моди та одягу є одним із найприбутковіших, забезпечуючи понад 20% загального обсягу ринку B2C (business-to-consumer). Цей сегмент вирізняється не лише високим рівнем попиту, але й постійним оновленням асортименту, що стимулює залучення нових клієнтів. Електронна комерція активно адаптується до нових викликів, таких як екологічність, прозорість ланцюгів постачання та підвищення рівня персоналізації послуг. Наприклад, бізнеси все частіше впроваджують екологічно чисті рішення у свої логістичні процеси, що відповідає зростаючим очікуванням клієнтів щодо сталого розвитку. Персоналізований

підхід, який включає рекомендаційні системи та алгоритми штучного інтелекту, дозволяє значно покращити користувацький досвід і збільшити середній чек покупок, тим самим сприяючи зростанню прибутковості підприємств.

Серед провідних регіонів у сфері електронної комерції виокремлюються Азійсько-Тихоокеанський регіон, Північна Америка та Західна Європа, які займають лідируючі позиції за обсягами продажів та кількістю користувачів. Китай залишається найбільшим ринком електронної комерції, забезпечуючи понад 50% глобального обсягу транзакцій завдяки масштабному використанню цифрових платформ, таких як Alibaba та JD.com. Ключовими факторами успіху китайського ринку є широка доступність мобільних платежів, інтеграція з соціальними мережами та активна участь держави у підтримці цифрової економіки.

У той же час, Європа демонструє стійке зростання, орієнтуючись на інноваційні рішення, зокрема створення єдиного цифрового ринку, що сприяє зменшенню бар'єрів для електронної торгівлі між країнами ЄС. Інтеграція електронної комерції з екосистемами соціальних мереж, таких як Instagram і Facebook, дозволяє компаніям ефективніше взаємодіяти з клієнтами, впроваджуючи прямі продажі через платформи соціального комерсу.

Північна Америка, зокрема США, також зберігає провідні позиції завдяки розвинутій логістичній інфраструктурі, високій довірі до електронних платежів і домінуванню таких гігантів, як Amazon. Розвиток підписних сервісів, персоналізованих рекомендацій та швидкої доставки стає визначальним фактором для подальшого зростання цього регіону. Таким чином, регіональні особливості створюють унікальні можливості для розширення електронної комерції, визначаючи її стратегічний розвиток у глобальному контексті.

Важливим чинником успіху електронної комерції є постійне вдосконалення її інфраструктури, яка включає інтеграцію сучасних платіжних систем, розвиток логістичних мереж та впровадження передових технологій безпеки. Платіжні рішення, зокрема підтримка мобільних гаманців і криптовалют, значно спрощують процеси здійснення транзакцій, забезпечуючи їх зручність і швидкість.

Удосконалення логістики, включаючи автоматизацію складів, використання дронів для доставки та розумних систем управління ланцюгами постачання, сприяє підвищенню ефективності роботи компаній і задоволеності споживачів. Сучасні споживачі все більше цінують можливість отримувати свої замовлення у найкоротші терміни, користуватися різними способами оплати, включаючи "купуй зараз – плати пізніше", та бути впевненими у збереженні своїх персональних даних. Для бізнесу це означає необхідність інвестування в інноваційні технології, які не лише забезпечують безпеку транзакцій, але й підвищують гнучкість платформ, дозволяючи швидко адаптуватися до змін ринку.

Електронна комерція відіграє вагомий роль у формуванні сучасних соціально-економічних процесів, створюючи значні можливості для розвитку різних галузей економіки. Вона сприяє розширенню зайнятості в таких сферах, як інформаційні технології, логістика, обслуговування клієнтів та електронна торгівля. За рахунок розвитку електронної комерції малий і середній бізнес (МСП) отримують нові канали для доступу до глобальних ринків, що стимулює їх зростання і конкуренцію. Зокрема, створення онлайн-платформ дозволяє МСП значно скоротити операційні витрати та залучати ширшу аудиторію. Поряд із позитивними змінами електронна комерція створює виклики, які вимагають ефективного вирішення. Одним із ключових питань є регуляція діяльності на цифрових платформах, що включає розробку нормативно-правових актів для забезпечення прозорості транзакцій, захисту персональних даних споживачів та дотримання правил оподаткування. Крім того, у зв'язку з високою швидкістю розвитку цифрових технологій необхідно створювати адаптивні механізми регуляції, які дозволять уникати зловживань та підтримувати рівні умови для всіх учасників ринку.

Особливого значення набуває питання захисту даних у контексті зростаючого обсягу електронних транзакцій. Впровадження стандартів безпеки, таких як шифрування даних і багатофакторна автентифікація, є пріоритетом для компаній, які прагнуть зберегти довіру клієнтів. Податкові аспекти також

відіграють важливу роль у забезпеченні сталого розвитку електронної комерції, адже формування справедливих і прозорих механізмів оподаткування дозволяє уникати нелегальної діяльності та забезпечувати додаткові надходження до бюджету. Електронна комерція виступає потужним двигуном соціально-економічного прогресу, сприяючи створенню робочих місць, стимулюючи розвиток підприємництва та забезпечуючи нові можливості для глобальної інтеграції економік. Водночас, подальший розвиток цього ринку залежатиме від здатності суспільства ефективно реагувати на виклики регуляції та безпеки в цифровому середовищі.

Аналіз ринку електронної комерції свідчить про його високий потенціал для подальшого розвитку, враховуючи його здатність адаптуватися до змінних умов глобального ринку. Інтеграція сучасних технологій, таких як штучний інтелект, блокчейн та інтернет речей (IoT), надає бізнесам можливість підвищувати ефективність процесів, покращувати аналітику та створювати персоналізовані рішення для споживачів. Впровадження інновацій, зокрема автоматизації логістики, технологій доповненої реальності (AR) та віртуальних шоурумів, розширює можливості залучення клієнтів і підвищує їх задоволеність. Вдосконалення користувацького досвіду через інтеграцію багатоканальних платформ і цифрових сервісів сприяє створенню довготривалих відносин із клієнтами та формуванню лояльності до брендів. Ринок електронної комерції демонструє значний вплив на економіку, сприяючи розвитку нових бізнес-моделей, підвищенню рівня зайнятості та створенню доданої вартості у глобальному масштабі. Таким чином, подальший успіх електронної комерції значною мірою залежатиме від здатності учасників ринку ефективно інтегрувати нові технології, забезпечувати високі стандарти безпеки та адаптуватися до змінних потреб споживачів.

1.3 Технологічні тренди у веб-розробленні

Веб-розроблення є однією з найбільш динамічних сфер сучасних технологій, яка постійно трансформується під впливом нових вимог бізнесу, споживачів та технологічних інновацій (Рис. 1.1). Сучасні технологічні тренди визначають не лише напрямки розвитку галузі, але й безпосередньо впливають на ефективність, гнучкість і продуктивність реалізації проєктів, зокрема інтернет-магазинів. Ці тренди спрямовані на підвищення якості обслуговування, розширення функціональних можливостей платформ та створення безпечних, зручних і масштабованих рішень. Нижче наведено ключові технологічні напрямки, що формують майбутнє веб-розроблення.

Одним із основних трендів у сучасному веб-розробленні є широке впровадження мікросервісної архітектури, яка відкриває нові горизонти у проєктуванні та реалізації складних систем. Мікросервіси дозволяють розділяти функціональність веб-додатків на невеликі незалежні сервіси, кожен із яких відповідає за виконання конкретних завдань. Це забезпечує значну гнучкість у процесі розробки, дозволяє окремим командам працювати автономно та зменшує ризики, пов'язані з виникненням помилок. Наприклад, у разі збоїв одного мікросервісу, решта системи залишається працездатною, що забезпечує високу стійкість до збоїв. Мікросервісна архітектура дозволяє легше інтегрувати нові функції, масштабувати систему в залежності від потреб бізнесу та використовувати різні технології для кожного сервісу, оптимізуючи їх продуктивність. Такий підхід широко використовується в масштабних платформах, зокрема в електронній комерції, де необхідно швидко обробляти великі обсяги даних, інтегрувати платіжні системи, забезпечувати роботу складних аналітичних інструментів та адаптуватися до змін ринку. У поєднанні з автоматизацією процесів, мікросервіси сприяють значному скороченню часу розробки та підвищенню якості продукту.

Ще одним важливим напрямком є активне використання хмарних обчислень, які забезпечують революційний підхід до управління ресурсами та масштабування додатків. Хмарні технології дозволяють розробникам швидко розгортати та

модифікувати інфраструктуру, автоматизувати ключові процеси й забезпечувати високу масштабованість веб-додатків. Завдяки використанню платформ, таких як AWS, Google Cloud та Microsoft Azure (Рис.1.1), бізнес отримує доступ до гнучких, надійних та безпечних інструментів для обробки великих обсягів даних, зберігання інформації та швидкої інтеграції нових функцій. Це особливо важливо для електронної комерції, де важливою є можливість оперативного реагування на зміни попиту та сезонні коливання.

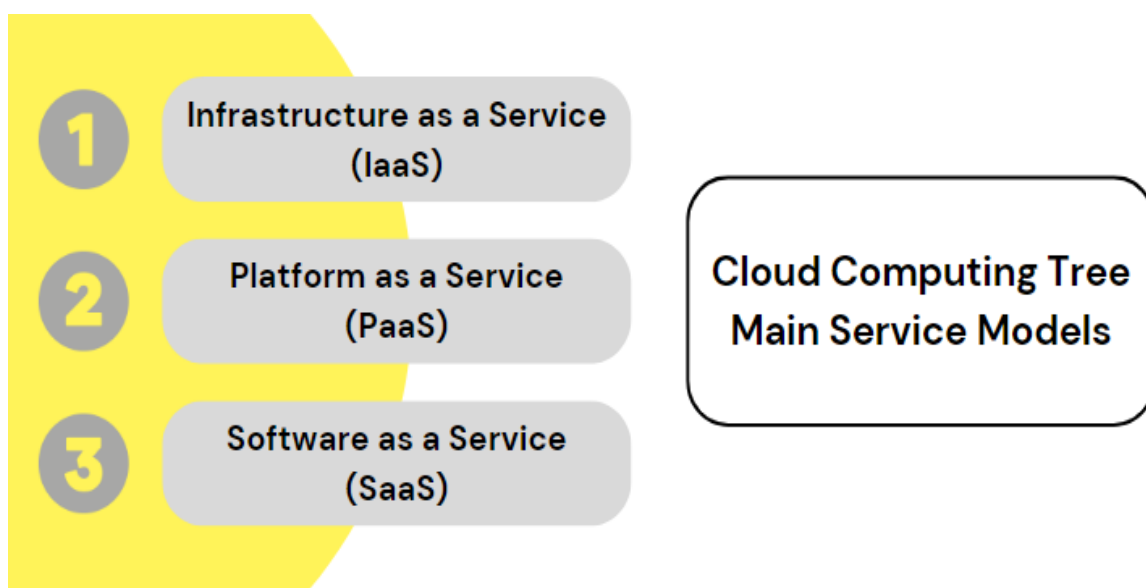


Рисунок 1.1 – Технологічні тренди у веб-розробленні

Використання хмарних обчислень також сприяє зниженню витрат на утримання фізичних серверів та ІТ-інфраструктури. Завдяки моделям оплати "pay-as-you-go" компанії можуть оптимізувати витрати, оплачуючи лише використані ресурси. Крім того, хмарні технології дозволяють легко масштабувати потужності, що є критично важливим під час пікових навантажень, наприклад, під час розпродажів або святкових періодів. Інтеграція інструментів аналізу даних у хмарному середовищі відкриває додаткові можливості для бізнесів, зокрема прогнозування продажів, аналіз поведінки клієнтів і впровадження персоналізованих маркетингових кампаній. Сучасні хмарні платформи також пропонують розширені можливості безпеки, включаючи захист від DDoS-атак,

шифрування даних і контроль доступу на основі ролей. Це забезпечує високий рівень захисту даних клієнтів і сприяє підвищенню довіри до компанії. Таким чином, хмарні обчислення стають важливим інструментом для створення ефективних, надійних і масштабованих веб-додатків, що відповідають потребам сучасного бізнесу.

Тренди у веб-розробленні також включають активне впровадження штучного інтелекту (ШІ) та машинного навчання, які суттєво змінюють способи взаємодії з користувачами та підвищують ефективність бізнес-процесів. Ці технології використовуються для створення рекомендаційних систем, які аналізують попередні покупки, уподобання та поведінку користувачів, забезпечуючи індивідуальний підхід до кожного клієнта. У сфері електронної комерції це дозволяє компаніям значно підвищити конверсію, оскільки клієнти отримують пропозиції, які максимально відповідають їхнім потребам. ШІ також активно застосовується для створення чат-ботів, які автоматизують обробку запитів і забезпечують безперервну підтримку клієнтів. Ці боти можуть відповідати на поширені запитання, допомагати у виборі товарів або послуг, а також проводити анкетування для збору зворотного зв'язку. Завдяки використанню технологій природної мовної обробки (NLP), сучасні чат-боти здатні розуміти контекст розмови та надавати більш точні й релевантні відповіді. Машинне навчання сприяє вдосконаленню аналізу великих обсягів даних, що дозволяє передбачати поведінку користувачів, виявляти тренди та оптимізувати маркетингові кампанії. Наприклад, алгоритми кластеризації допомагають сегментувати клієнтську базу, а моделі прогнозування – передбачати попит на товари чи послуги. Також ШІ використовується для автоматизації рутинних процесів, таких як управління запасами, обробка транзакцій і виявлення шахрайства.

Інтеграція штучного інтелекту та машинного навчання у веб-додатки відкриває нові можливості для бізнесу, забезпечуючи персоналізацію, підвищення задоволеності клієнтів і оптимізацію бізнес-процесів.

Не менш важливим аспектом розвитку веб-розроблення є вдосконалення технологій фронтенду, які сьогодні визначають користувацький досвід та продуктивність веб-додатків. Використання сучасних JavaScript-фреймворків, таких як React, Angular та Vue.js, дозволяє створювати динамічні інтерфейси, які адаптуються до різноманітних пристроїв та забезпечують швидкий відгук на дії користувачів. Ці фреймворки забезпечують ефективне управління компонентами, знижуючи витрати на розробку та підтримку складних інтерфейсів. Зростає популярність підходу серверного рендерингу (SSR), який дозволяє згенерувати HTML-сторінки на стороні сервера перед їх відправкою на клієнт. Це значно покращує швидкість завантаження сторінок, особливо для користувачів із повільними мережевими з'єднаннями, і сприяє підвищенню пошукової оптимізації (SEO). Статичні сайти також набувають популярності завдяки простоті їх розгортання, високій швидкості роботи та низьким витратам на хостинг, що робить їх оптимальним рішенням для багатьох бізнесів.

Ще одним значущим напрямком є впровадження бібліотек і технологій для оптимізації графіки та анімацій, які сприяють створенню більш привабливих і інтерактивних інтерфейсів. Наприклад, бібліотеки на кшталт Three.js або D3.js активно використовуються для побудови 3D-візуалізацій та інтерактивних графіків, що дозволяє покращити взаємодію користувачів із платформою. Такі рішення особливо корисні для електронної комерції, де важливо створити яскраве та запам'ятовуване представлення товарів.

Значний вплив на веб-розроблення має також поширення прогресивних веб-додатків (PWA). Ці додатки поєднують переваги традиційних веб-сайтів і мобільних додатків, пропонуючи швидкий доступ до функціоналу, можливість працювати в офлайн-режимі та високу зручність використання на різних пристроях. PWA дозволяють зберігати дані локально, що забезпечує доступність функцій навіть за відсутності стабільного підключення до інтернету. Також PWA можуть оновлюватися у фоновому режимі, що гарантує актуальність інформації для користувачів.

Для бізнесу впровадження PWA означає значне зниження витрат на розробку та підтримку окремих мобільних додатків для різних платформ. Крім того, PWA забезпечують швидке завантаження сторінок, що є важливим фактором для утримання користувачів і зниження показників відмов. Наприклад, дослідження показують, що затримка в завантаженні сторінки навіть на кілька секунд може значно зменшити рівень конверсії. Впровадження PWA особливо корисне для електронної комерції, оскільки вони сприяють підвищенню залученості клієнтів і спрощують процеси покупки. Наприклад, клієнти можуть додавати товари до кошика чи переглядати каталоги навіть у разі переривання підключення до мережі. Крім того, можливість відправки push-сповіщень через PWA допомагає підтримувати контакт із клієнтами, інформуючи їх про акції, знижки та новинки. Оскільки PWA працюють через веб-браузер, вони не потребують завантаження з магазинів додатків, що спрощує доступ для користувачів. Цей аспект робить їх ідеальним рішенням для бізнесів, які прагнуть швидко розширити свою аудиторію, забезпечуючи високий рівень обслуговування без додаткових складнощів.

Інтеграція веб-додатків із системами інтернету речей (IoT) є важливим трендом, який активно трансформує сучасні бізнес-моделі. IoT відкриває нові можливості для автоматизації бізнес-процесів, забезпечуючи реальний моніторинг товарів у реальному часі, оптимізацію складських операцій і покращення взаємодії з клієнтами за допомогою інтелектуальних пристроїв. Наприклад, компанії можуть використовувати сенсори для відстеження стану та переміщення товарів на всіх етапах постачання, що знижує витрати та ризики, пов'язані з втратами чи пошкодженнями. Автоматизація управління складськими запасами, здійснювана за допомогою IoT, дозволяє значно підвищити ефективність управління ресурсами. Розумні пристрої, інтегровані з веб-додатками, забезпечують контроль залишків, автоматичне замовлення продукції та оптимізацію використання складських площ. Це створює значні переваги для бізнесу, особливо у сфері електронної комерції, де швидкість і точність управління ланцюгами постачання мають вирішальне значення. Інтерактивні пристрої IoT також сприяють покращенню користувацького

досвіду. Наприклад, розумні дзеркала у магазинах можуть синхронізуватися з веб-додатками, дозволяючи клієнтам переглядати додаткову інформацію про товари чи створювати віртуальні примірки. Крім того, IoT сприяє впровадженню систем віддаленого обслуговування, які забезпечують більш зручний та індивідуалізований підхід до клієнтів.

Однак інтеграція IoT із веб-додатками вимагає адаптації до обробки великих обсягів даних і впровадження надійних систем безпеки. Забезпечення захисту даних, що передаються між пристроями та додатками, є ключовим викликом для розробників. Це включає використання шифрування, багаторівневого контролю доступу та регулярного оновлення програмного забезпечення для захисту від кібератак.

IoT у поєднанні з веб-додатками створює потужний інструмент для оптимізації бізнесу, покращення обслуговування клієнтів і формування нових стандартів продуктивності в різних галузях.

Таким чином, технологічні тренди у веб-розробленні створюють основу для розвитку інноваційних, ефективних і адаптивних веб-рішень. Їх впровадження дозволяє бізнесу здобувати стратегічні переваги на ринку, підвищувати продуктивність процесів та оптимізувати взаємодію зі споживачами. Використання сучасних технологій, таких як мікросервісна архітектура, хмарні обчислення, штучний інтелект, прогресивні веб-додатки (PWA) та інтеграція з IoT, забезпечує компаніям не лише гнучкість і надійність, але й можливість швидко реагувати на зміни ринку. Водночас ці технології сприяють створенню нових стандартів у користувацькому досвіді, роблячи веб-додатки більш зручними, функціональними та безпечними. Такий підхід дозволяє бізнесу адаптуватися до сучасних викликів, розширювати свою аудиторію та забезпечувати стабільний розвиток у довгостроковій перспективі..

1.4 Висновки по розділу

Розділ "Аналіз актуальності та огляд предметної області" висвітлив ключові аспекти розвитку електронної комерції та веб-розроблення, підкресливши їх значення для сучасного бізнесу. Було розглянуто актуальність створення інтернет-магазинів, що зумовлена глобальною цифровою трансформацією, зростанням попиту на онлайн-торгівлю та перевагами для бізнесу, такими як зниження витрат і розширення географії продажів.

Аналіз ринку електронної комерції підтвердив його динамічне зростання, спричинене інтеграцією сучасних технологій, зміною поведінки споживачів і впровадженням нових бізнес-моделей. Сегментація ринку показала значний потенціал таких галузей, як продаж товарів повсякденного вжитку, моди, електроніки та послуг. Особливо важливим є розвиток регіональних ринків, таких як Азійсько-Тихоокеанський регіон, Західна Європа та Північна Америка, де електронна комерція є основним драйвером економіки.

Окрему увагу було приділено технологічним трендам, які сьогодні формують майбутнє веб-розроблення. Було виявлено, що мікросервісна архітектура, хмарні обчислення, штучний інтелект, прогресивні веб-додатки (PWA) та інтеграція з інтернетом речей (IoT) є основними напрямками, що забезпечують адаптивність, безпеку та інноваційність сучасних веб-рішень. Їх впровадження дозволяє бізнесу підвищувати продуктивність, знижувати витрати та покращувати користувацький досвід, що є критично важливим у конкурентному середовищі.

Проведений аналіз засвідчив необхідність інтеграції сучасних технологій у процес розроблення інтернет-магазинів, що сприятиме їх відповідності сучасним вимогам ринку та забезпечить сталий розвиток у довгостроковій перспективі..

2 ТЕОРЕТИЧНА МОДЕЛЬ СТВОРЕННЯ ІНТЕРНЕТ-МАГАЗИНУ

2.1 Архітектура веб-додатків

Архітектура WEB-додатків (Рис.2.1) є фундаментом для створення сучасних, адаптивних і продуктивних систем, які відповідають зростаючим потребам бізнесу та споживачів. Особливо у випадку інтернет-магазинів, архітектура повинна гарантувати не лише високу продуктивність, безпеку та гнучкість, але й здатність підтримувати безперервну роботу навіть за умов пікових навантажень.

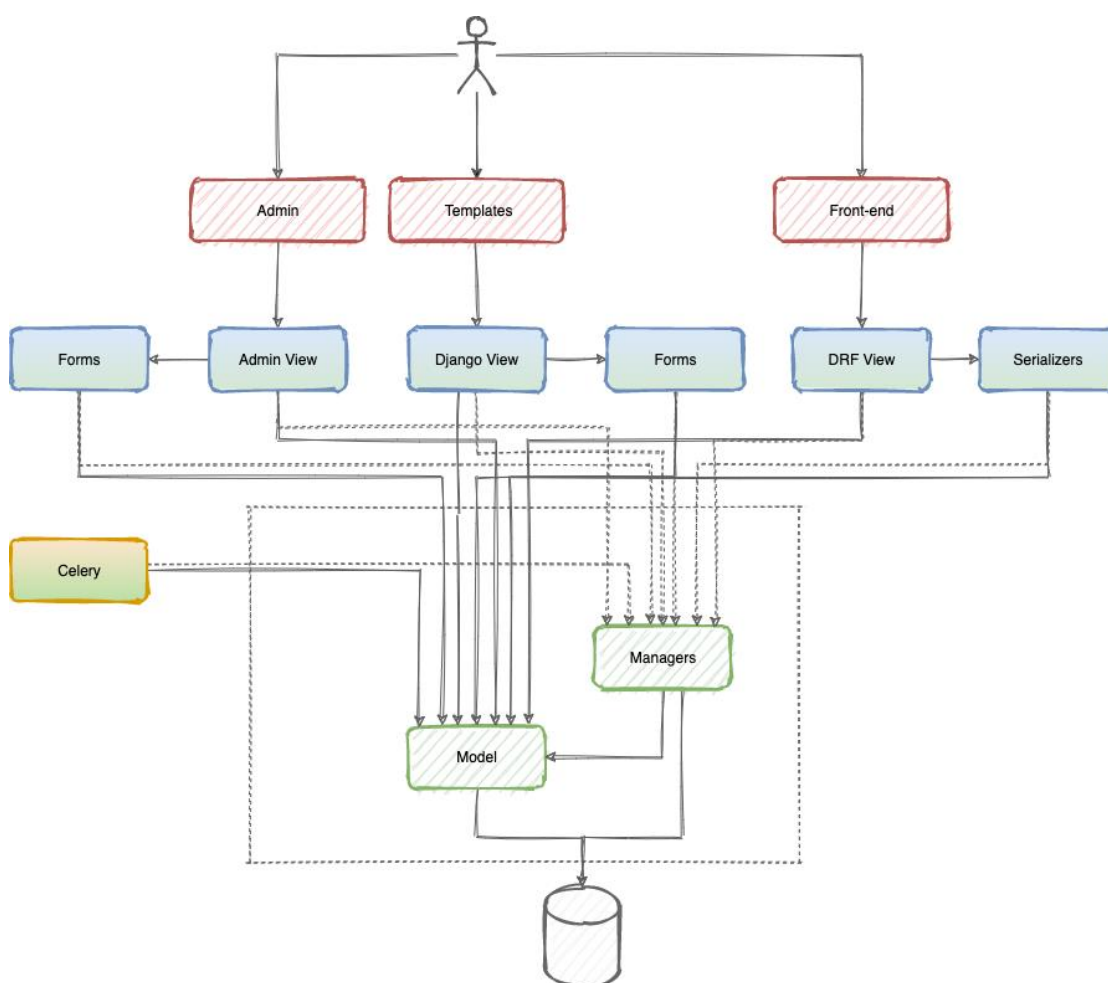


Рисунок 2.1 – Архітектура WEB-додатків

Ефективна архітектура є основою для інтеграції передових технологій, таких як штучний інтелект, автоматизація бізнес-процесів та аналітика даних, що дозволяє створювати конкурентоспроможні рішення. Основні компоненти

архітектури веб-додатків можна розділити на декілька рівнів, які забезпечують комплексну роботу системи, взаємодіючи між собою в реальному часі.

Клієнтський рівень (Frontend):

Клієнтський рівень відповідає за взаємодію користувача з додатком. Він включає інтерфейс користувача, який розробляється за допомогою сучасних технологій, таких як HTML5, CSS3 і JavaScript-фреймворки (React, Angular, Vue.js). Основними завданнями цього рівня є відображення інформації, забезпечення зручної навігації та інтерактивності, а також обробка запитів до сервера.

Серверний рівень (Backend):

Серверний рівень обробляє логіку додатка, виконує запити до бази даних і обробляє інформацію, отриману від клієнта. Для розробки серверної частини інтернет-магазинів часто використовуються фреймворки, такі як Django (Python), які забезпечують високу продуктивність, зручність розробки та вбудовані інструменти для роботи з базами даних, авторизації та безпеки.

База даних:

База даних є ключовим компонентом для зберігання інформації про товари, замовлення, користувачів та інші елементи. Реляційні бази даних, такі як PostgreSQL або MySQL, є популярним вибором завдяки їхній надійності та масштабованості. Крім того, у випадках великих обсягів даних можуть використовуватися NoSQL-рішення, такі як MongoDB, для забезпечення високої продуктивності.

API-інтерфейси:

API забезпечують зв'язок між клієнтським і серверним рівнями, а також інтеграцію з іншими сервісами, такими як платіжні системи, сервіси доставки чи сторонні аналітичні платформи. RESTful API та GraphQL є стандартними підходами для реалізації цієї взаємодії.

Системи кешування:

Для підвищення продуктивності та зменшення навантаження на сервер використовуються системи кешування, такі як Redis або Memcached. Вони

дозволяють зберігати часто запитувані дані в оперативній пам'яті, що значно прискорює їх доступ.

Системи безпеки:

Захист даних користувачів та забезпечення безпеки транзакцій є критично важливими аспектами архітектури. Включення таких механізмів, як шифрування даних, багаторівневий контроль доступу, захист від SQL-ін'єкцій та інших кіберзагроз, є стандартом для сучасних інтернет-магазинів.

Системи масштабування:

Для забезпечення стабільної роботи під час високих навантажень використовуються методи масштабування, зокрема горизонтальне (додавання серверів) та вертикальне (збільшення ресурсів сервера).

Архітектура веб-додатків повинна бути модульною, що дозволяє легко інтегрувати нові функції та забезпечувати швидку адаптацію до змін у бізнес-вимогах і технологічних інноваціях. Модульність сприяє зменшенню витрат на розробку та модернізацію, забезпечуючи при цьому простоту управління системою. Це особливо важливо для інтернет-магазинів, які постійно розширюють функціонал відповідно до потреб клієнтів і ринкових умов.

Гнучкість архітектури дозволяє швидко впроваджувати інновації та ефективно адаптуватися до змінних умов ринку, мінімізуючи ризики збоїв. Наприклад, система може бути легко масштабована за допомогою автоматичного балансування навантаження, що забезпечує стабільну роботу навіть за значного збільшення кількості користувачів, наприклад, під час святкових розпродажів або рекламних акцій. Це включає як горизонтальне масштабування (додавання нових серверів), так і вертикальне (збільшення потужності існуючих серверів). Безпека архітектури є критичним елементом, особливо для систем, які працюють із фінансовими транзакціями та персональними даними користувачів. Вона включає використання сучасних методів шифрування даних, багаторівневу автентифікацію, контроль доступу на основі ролей, а також впровадження механізмів захисту від DDoS-атак і SQL-ін'єкцій. Крім того, регулярне тестування на проникнення

(penetration testing) та моніторинг безпеки допомагають виявляти та усувати потенційні вразливості. Гнучка архітектура також сприяє підвищенню ефективності бізнесу, дозволяючи оперативно інтегрувати нові функціональні можливості, такі як інтерактивні каталоги, персоналізовані рекомендації чи інтеграція з платіжними системами. Завдяки цьому покращується користувацький досвід, знижуються показники відмов і підвищується рівень лояльності клієнтів. Усі ці фактори роблять архітектуру ключовим інструментом забезпечення довготривалого успіху інтернет-магазину на конкурентному ринку.

Додатково, ефективна архітектура веб-додатків значно покращує користувацький досвід, забезпечуючи високу продуктивність, швидке завантаження сторінок, інтерактивність та адаптивність інтерфейсу під різні пристрої. Це особливо важливо для сучасних інтернет-магазинів, де кожна секунда очікування може суттєво вплинути на рівень конверсії та задоволеність клієнтів. Інтеграція інструментів для попереднього завантаження даних і компресії зображень дозволяє мінімізувати обсяги передаваних даних, скорочуючи час завантаження сторінок навіть при низькій швидкості інтернет-з'єднання. Використання технологій, таких як lazy loading, дозволяє завантажувати лише ті елементи, які користувач бачить на екрані, що знижує навантаження на сервер і прискорює рендеринг контенту. Крім того, інтеграція розподілених серверів CDN (Content Delivery Network) забезпечує швидку доставку статичних ресурсів, таких як зображення, стилі та скрипти, незалежно від географічного розташування користувача. Це дозволяє глобальним інтернет-магазинам залишатися конкурентоспроможними, гарантуючи однаковий рівень швидкості для клієнтів із різних регіонів.

Сучасні фреймворки та бібліотеки, наприклад, React або Vue.js, у поєднанні з Django на серверній стороні, додають інтерактивності додаткам, створюючи динамічні користувацькі інтерфейси. Адаптивний дизайн, який автоматично підлаштовується під різні розміри екранів, робить взаємодію з платформою зручною як на смартфонах, так і на десктопах. Такий комплексний підхід до

архітектури веб-додатків дозволяє забезпечити не лише технічну надійність, але й високу якість взаємодії, що є ключовим чинником успіху у сфері електронної комерції. Продуманий архітектурний дизайн сприяє створенню інтуїтивно зрозумілих і легких у використанні інтерфейсів, що знижує рівень відмов і підвищує залученість користувачів. Застосування технологій персоналізації, таких як рекомендаційні системи та динамічний контент, дозволяє адаптувати функціонал платформи до потреб і уподобань кожного клієнта. Правильний вибір архітектурного підходу гарантує стабільність роботи системи, полегшує її масштабування в разі зростання навантаження, сприяє зростанню бізнесу та забезпечує його конкурентоспроможність у динамічному середовищі сучасного ринку.

2.2 Особливості використання Django

Django є одним із найпопулярніших веб-фреймворків для розробки веб-додатків, зокрема інтернет-магазинів. Його популярність пояснюється простотою використання, багатим функціоналом "з коробки", високою продуктивністю та активною спільнотою. Django побудовано на мові програмування Python, яка відома своєю зрозумілістю, що полегшує вивчення та використання. Крім того, Python має потужну екосистему бібліотек, які доповнюють функціонал Django, дозволяючи легко інтегрувати сучасні технології, такі як машинне навчання та аналітика даних. Django також вирізняється підтримкою швидкої розробки завдяки вбудованим інструментам, таким як ORM, системи маршрутизації та автоматизоване створення адміністративних панелей, що робить його ідеальним вибором для широкого спектру завдань, включаючи створення інтернет-магазинів.

Основні особливості Django:

Модульність і структура MVC (Model-View-Controller):

Django реалізує підхід Model-View-Template (MVT) (Рис. 2.2), який є подібним до MVC. Ця структура дозволяє чітко розділяти бізнес-логіку, представлення даних і інтерфейс користувача. Модульність забезпечує простоту

розширення та підтримки проєктів, що особливо важливо для великих інтернет-магазинів із широким функціоналом.

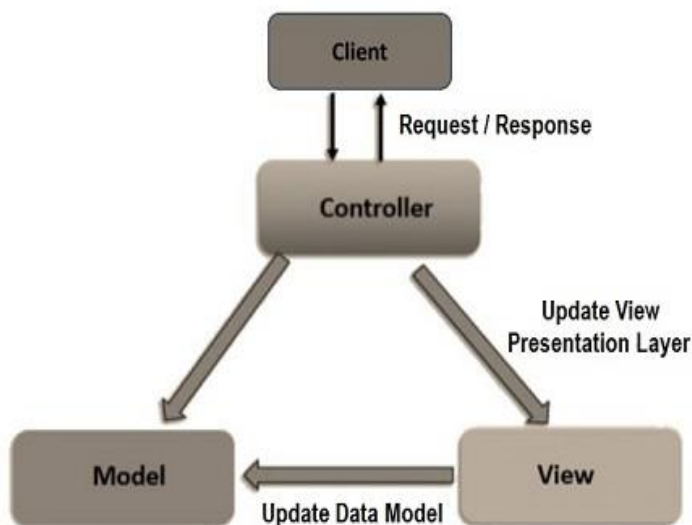


Рисунок 2.2 – Model-View-Template

Адміністративна панель:

Однією з найбільш корисних функцій Django є автоматично створювана адміністративна панель. Вона дозволяє розробникам і адміністраторам керувати базами даних, змінювати інформацію про товари, замовлення та користувачів без необхідності писати додатковий код.

Інтеграція з базами даних:

Django підтримує різноманітні реляційні бази даних, такі як PostgreSQL, MySQL та SQLite. Крім того, ORM (Object-Relational Mapping) дозволяє працювати з базами даних безпосередньо через Python-код, що знижує ризик помилок і спрощує процес розробки.

Безпека:

Django пропонує вбудовані засоби захисту від поширених атак, таких як SQL-ін'єкції, XSS (міжсайтовий скриптинг) та CSRF (міжсайтові запити підробки). Також забезпечується підтримка шифрування даних, управління доступом користувачів та інших важливих аспектів безпеки.

Масштабованість:

Django може використовуватися для проєктів будь-якого масштабу. Він підтримує горизонтальне масштабування, що дозволяє обробляти значні навантаження, характерні для великих інтернет-магазинів.

Застосування Django для інтернет-магазинів:

Швидка розробка:

Швидка розробка є однією з ключових переваг Django, оскільки фреймворк забезпечує великий набір вбудованих функцій, які суттєво скорочують час створення проєктів. Інструменти для авторизації, управління сесіями, маршрутизації, інтеграції з базами даних і автоматичного створення адміністративного інтерфейсу дозволяють розробникам зосередитися на бізнес-логіці, а не витратити час на реалізацію базових функцій. Це особливо важливо в умовах конкурентного середовища електронної комерції, де швидкість запуску нових продуктів і функцій є критичною для успіху. Завдяки модульній структурі Django, команди розробників можуть впроваджувати зміни поетапно, мінімізуючи ризики та зберігаючи стабільність системи. Наприклад, додавання нових модулів для оплати, інтеграції з CRM чи впровадження рекомендаційної системи можна виконувати поступово, не порушуючи основну роботу платформи. Також Django підтримує повторне використання коду через додатки, що дозволяє зменшити витрати часу на розробку схожих функцій для різних проєктів.

Додатково, завдяки інтеграції з інструментами розробки, такими як Docker для контейнеризації або CI/CD-системами (наприклад, GitLab CI/CD або Jenkins), Django сприяє автоматизації процесів тестування та розгортання, що ще більше пришвидшує доставку оновлень кінцевим користувачам. Це робить Django ідеальним вибором для швидкої розробки та масштабування проєктів в електронній комерції..

Гнучкість:

Django дозволяє легко інтегрувати зовнішні сервіси, такі як платіжні шлюзи (Stripe, PayPal), аналітичні платформи (Google Analytics) та сервіси доставки. Це

забезпечує розширення функціоналу інтернет-магазинів відповідно до бізнес-потреб.

Підтримка багатокористувацьких систем:

Django забезпечує функціонал для створення складних багатокористувацьких систем, що дозволяє одночасно працювати адміністраторам, менеджерам і клієнтам. Це включає персоналізовані панелі, розмежування прав доступу та управління великою кількістю одночасних сесій.

Масштабованість для високих навантажень:

Завдяки підтримці розподілених систем кешування, використанню асинхронних процесів та інтеграції з хмарними платформами, Django є ефективним вибором для проєктів із високими вимогами до продуктивності.

Спільнота та документація:

Django має активну спільноту розробників, що забезпечує швидке вирішення технічних проблем. Крім того, велика кількість ресурсів, включаючи офіційну документацію, полегшує навчання та використання фреймворку. Django є універсальним і потужним інструментом для створення інтернет-магазинів, що надає розробникам значний спектр можливостей для швидкого та ефективного втілення проєктів будь-якої складності. Завдяки вбудованим механізмам безпеки, масштабованості, багатофункціональності та простоті використання, Django дозволяє розробникам концентруватися на ключових завданнях, скорочуючи час на розробку базових функцій.

Особливістю Django є можливість інтеграції з потужними інструментами аналітики, такими як Tableau, Power BI або внутрішні Python-бібліотеки (наприклад, Pandas), що дозволяє створювати інтерактивні дашборди та здійснювати глибокий аналіз даних. Це особливо важливо для електронної комерції, де такі аналітичні інструменти допомагають відстежувати поведінку клієнтів, прогнозувати продажі, аналізувати ефективність маркетингових кампаній і приймати обґрунтовані бізнес-рішення. Django має вбудовані можливості для створення REST API, що дозволяє розробникам забезпечувати інтеграцію з

зовнішніми системами, такими як CRM (наприклад, Salesforce, Zoho CRM), ERP (SAP, Odoo) або SCM (Oracle SCM Cloud). Це сприяє автоматизації бізнес-процесів, зокрема управління запасами, логістикою, замовленнями та взаємодією з клієнтами. Завдяки такій інтеграції підприємства можуть досягти вищої ефективності та знизити операційні витрати.

Ще однією перевагою Django є підтримка впровадження сучасних технологій персоналізації, заснованих на штучному інтелекті. Використання бібліотек машинного навчання, таких як Scikit-learn, TensorFlow або PyTorch, дозволяє створювати адаптивні рекомендаційні системи, які підлаштовуються під вподобання користувачів, аналізуючи їхню поведінку, попередні покупки та перегляди. Наприклад, система може рекомендувати товари на основі алгоритмів кластеризації або глибокого навчання, що значно підвищує ймовірність здійснення покупки. Такі рішення можуть бути інтегровані безпосередньо у веб-додаток, забезпечуючи миттєвий доступ до рекомендацій.

Додатково, Django підтримує роботу з великими обсягами даних через підключення до баз даних, таких як PostgreSQL, MySQL або NoSQL-рішень (MongoDB). Це дозволяє ефективно обробляти транзакції, управляти користувачами, а також реалізовувати складні аналітичні запити в реальному часі. Завдяки цьому платформа стає чудовим вибором для створення складних, високонавантажених додатків, зокрема в галузі електронної комерції..

Django не лише відповідає потребам сучасного ринку, але й відкриває перспективи для впровадження інноваційних рішень у цифрові продукти, забезпечуючи стабільність, продуктивність та довготривалу конкурентоспроможність бізнесу.

2.3 Моделювання бази даних у Django

Моделювання бази даних у Django є однією з ключових частин розробки веб-додатків, оскільки саме база даних зберігає інформацію, необхідну для роботи системи. Django забезпечує простий і потужний спосіб роботи з базами даних

завдяки своїй системі Object-Relational Mapping (ORM). ORM дозволяє розробникам створювати, змінювати й керувати базами даних за допомогою Python-коду, що значно знижує складність і ризик помилок. Цей підхід сприяє скороченню часу на розробку та забезпечує гнучкість у налаштуванні бази даних відповідно до потреб бізнесу.

Django ORM (Рис.2.3) автоматично генерує SQL-запити, дозволяючи розробникам працювати з даними на високому рівні абстракції.

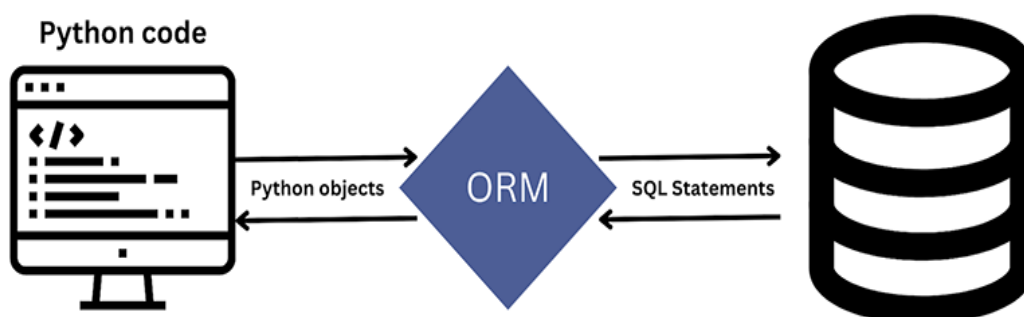


Рисунок 2.3 – Django ORM

Наприклад, замість написання складних SQL-запитів для вибірки даних, ORM надає можливість використовувати інтуїтивно зрозумілий Python-синтаксис, що спрощує процес розробки і підтримки коду. Крім автоматизації рутинних завдань, таких як створення та оновлення таблиць, Django ORM підтримує функціонал для роботи з великими обсягами даних, зокрема шардінг та реплікацію. Ці можливості роблять Django ефективним вибором для масштабованих рішень, таких як інтернет-магазини з багатотисячною базою користувачів і високим навантаженням. Інтеграція з інструментами кешування, такими як Redis, дозволяє значно підвищити швидкість обробки запитів і покращити загальну продуктивність додатків.

Основи моделювання бази даних у Django (Рис.2.4):

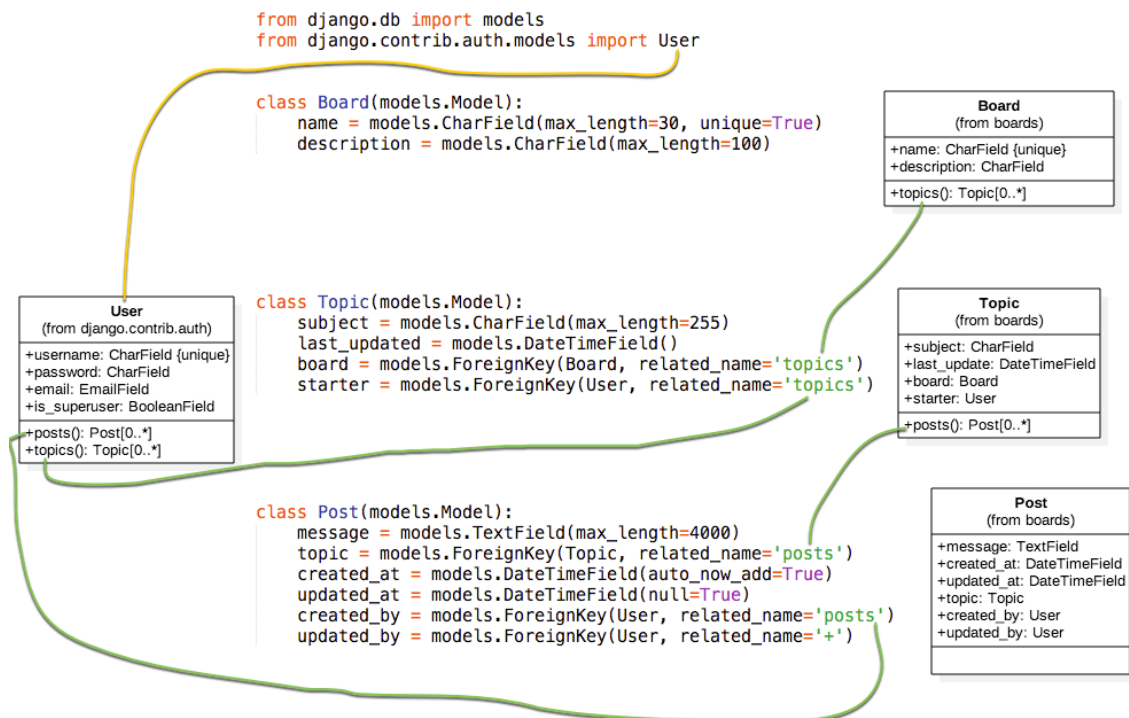


Рисунок 2.4 – Моделювання бази даних

Створення моделей:

Моделі в Django визначають структуру бази даних і є основою для роботи з даними. Кожна модель є класом Python, який відповідає одній таблиці в базі даних. Поля моделі представляють колонки таблиці, а їх типи відповідають типам даних у базі. Наприклад:

```

class Product(models.Model):
    name = models.CharField(max_length=100)
    description = models.TextField()
    price = models.DecimalField(max_digits=10, decimal_places=2)
    stock = models.IntegerField()
    created_at = models.DateTimeField(auto_now_add=True)
  
```

У цьому прикладі модель Product описує таблицю для зберігання інформації про товари.

Міграції:

Django використовує систему міграцій для створення і оновлення бази даних відповідно до змін у моделях. Команди `makemigrations` і `migrate` автоматично генерують SQL-запити, що спрощує управління схемою бази даних.

Зв'язки між моделями:

Django підтримує різні типи зв'язків між таблицями: "один до одного", "один до багатьох" і "багато до багатьох". Наприклад:

```
class Order(models.Model):
    customer = models.ForeignKey('Customer', on_delete=models.CASCADE)
    product = models.ManyToManyField('Product')
    total_price = models.DecimalField(max_digits=10, decimal_places=2)
    created_at = models.DateTimeField(auto_now_add=True)
```

У цьому прикладі таблиця Order пов'язана з таблицями Customer і Product через зовнішні ключі.

Вбудовані функції ORM:

Django ORM дозволяє виконувати складні запити до бази даних без написання SQL-коду. Наприклад, отримання всіх товарів із ціною більше 100:

```
expensive_products = Product.objects.filter(price__gt=100)
```

Валідація даних:

Поля моделей у Django підтримують вбудовану валідацію, яка гарантує коректність даних перед їх збереженням у базі. Наприклад, поле `DecimalField` автоматично перевіряє, чи є введені значення числовими. Для більш складної валідації можна використовувати методи `clean()` у моделях, які дозволяють виконувати перевірку залежностей між полями. Наприклад:

```

class Product(models.Model):
    name = models.CharField(max_length=100)
    price = models.DecimalField(max_digits=10, decimal_places=2)
    stock = models.IntegerField()
    def clean(self):
        if self.price < 0:
            raise ValidationError('Ціна не може бути від'ємною.')
        if self.stock < 0:
            raise ValidationError('Кількість на складі не може бути
від'ємною.')

```

Такий підхід забезпечує точність і узгодженість даних, які зберігаються в базі.

Особливості для інтернет-магазинів:

Моделювання товарів і категорій:

Інтернет-магазини часто вимагають складної структури для зберігання інформації про товари, їх категорії та властивості. Django дозволяє створювати гнучкі моделі, які можна адаптувати до потреб бізнесу. Наприклад:

```

class Category(models.Model):
    name = models.CharField(max_length=100)

class Product(models.Model):
    name = models.CharField(max_length=100)
    category = models.ForeignKey('Category', on_delete=models.CASCADE)

```

Кошик і замовлення:

Моделі для зберігання кошиків і замовлень можуть містити дані про клієнтів, обрані товари, кількість, вартість і статус замовлення. Наприклад, модель для кошика може виглядати так:

```

class Cart(models.Model):
    user = models.ForeignKey('User', on_delete=models.CASCADE)
    product = models.ForeignKey('Product', on_delete=models.CASCADE)
    quantity = models.IntegerField(default=1)

```

```

added_at = models.DateTimeField(auto_now_add=True)

class Order(models.Model):
    user = models.ForeignKey('User', on_delete=models.CASCADE)
    products = models.ManyToManyField('Product', through='OrderItem')
    total_price = models.DecimalField(max_digits=10, decimal_places=2)
    status = models.CharField(max_length=50, choices=[('Pending',
'Pending'), ('Completed', 'Completed'), ('Cancelled', 'Cancelled')])
    created_at = models.DateTimeField(auto_now_add=True)

class OrderItem(models.Model):
    order = models.ForeignKey('Order', on_delete=models.CASCADE)
    product = models.ForeignKey('Product', on_delete=models.CASCADE)
    quantity = models.IntegerField()

```

Це дозволяє автоматизувати процес обробки замовлень, включаючи розрахунок загальної вартості, статусів замовлень та інтеграцію з платіжними системами.

Аудит і логування:

Django підтримує створення моделей для відстеження змін у базі даних, що є корисним для моніторингу діяльності в інтернет-магазині. Наприклад, можна створити модель для запису історії змін в обраних таблицях:

```

class AuditLog(models.Model):
    table_name = models.CharField(max_length=100)
    action = models.CharField(max_length=50, choices=[('INSERT',
'Insert'), ('UPDATE', 'Update'), ('DELETE', 'Delete')])
    timestamp = models.DateTimeField(auto_now_add=True)
    user = models.ForeignKey('User', on_delete=models.SET_NULL,
null=True)
    details = models.JSONField()

```

Ця модель дозволяє зберігати інформацію про те, які зміни було зроблено, ким і коли, що підвищує прозорість та контроль за діяльністю в системі.

Переваги використання Django для моделювання бази даних:

Швидка розробка: Django автоматично створює необхідну структуру бази даних і дозволяє швидко адаптувати її до змін.

Безпека: Django забезпечує захист від SQL-ін'єкцій і автоматичне шифрування конфіденційних даних.

Простота інтеграції: ORM дозволяє легко працювати з різними типами баз даних, не залежачи від їх конкретної реалізації.

Гнучкість: Моделі в Django можуть бути адаптовані для підтримки специфічних бізнес-процесів, що забезпечує їх універсальність.

Django надає ефективні інструменти для створення та управління базами даних, які дозволяють розробникам зосередитися на бізнес-логіці та функціоналі веб-додатків, не витрачаючи часу на ручне написання SQL-запитів..

2.4 Висновки по розділу

Розділ "Теоретична модель створення інтернет-магазину" охоплює основи архітектури веб-додатків, особливості використання фреймворку Django, а також підходи до моделювання бази даних, що є критично важливими для ефективної розробки та підтримки інтернет-магазинів. Аналіз архітектурних рішень показав, що чітка структура компонентів і модульний підхід дозволяють створювати масштабовані, гнучкі та надійні системи. Використання Django забезпечує швидку розробку завдяки вбудованим інструментам, таким як ORM, міграції та адміністративна панель, що спрощує управління базою даних та основними бізнес-процесами.

Моделювання бази даних у Django, базуючись на принципах ORM, дозволяє ефективно працювати з різними типами баз даних, зменшувати кількість помилок завдяки вбудованій валідації даних і автоматизації процесів. ORM надає розробникам можливість використовувати інтуїтивно зрозумілий Python-синтаксис для створення запитів, що спрощує обробку даних навіть у складних структурах. Розглянуті підходи до створення моделей для товарів, кошиків,

замовлень і логування не лише забезпечують підтримку складних бізнес-процесів, таких як управління каталогами, обробка замовлень і моніторинг діяльності системи, але й сприяють масштабованості та адаптації бази даних до змін потреб бізнесу. Інструменти міграцій Django дозволяють швидко вносити зміни до структури бази даних, а інтеграція з інструментами кешування, такими як Redis, допомагає оптимізувати продуктивність системи, забезпечуючи швидкий доступ до даних та зниження навантаження на сервер.

Дослідження підтвердило, що Django є потужним інструментом для розробки інтернет-магазинів, забезпечуючи необхідну функціональність, гнучкість та безпеку. Цей фреймворк дозволяє створювати сучасні рішення, які відповідають вимогам електронної комерції, а його широкі можливості сприяють впровадженню інновацій, оптимізації бізнес-процесів та підвищенню конкурентоспроможності підприємств у цифровій економіці.

3 ОПИС ПОКРАЩЕНЬ ДЛЯ ІСНУЮЧИХ ІНТЕРНЕТ-МАГАЗИНІВ

3.1 Оптимізація користувацького досвіду

Оптимізація користувацького досвіду (UX) (Рис. 3.1) є одним із ключових напрямів покращення існуючих інтернет-магазинів, що безпосередньо впливає на лояльність клієнтів, показники конверсії та загальну ефективність бізнесу. Успішна оптимізація UX повинна базуватися на ретельному аналізі поведінки користувачів, проведенні А/В-тестування для визначення найбільш ефективних рішень, а також впровадженні сучасних технологій, таких як штучний інтелект і машинне навчання.

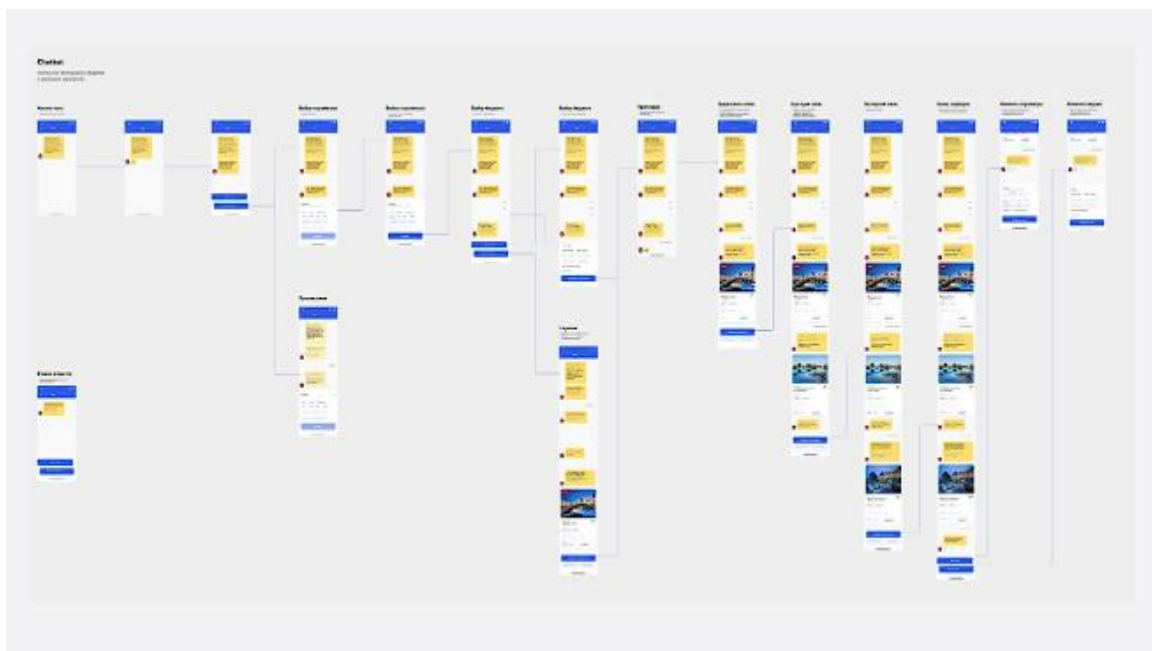


Рисунок 3.1 – Ілюстративна схема оптимізації UX

Одним із важливих елементів є впровадження аналітичних інструментів, таких як Google Analytics) (Рис. 3.2) або спеціалізовані UX-аналітики, для моніторингу дій користувачів на сайті. Це дозволяє ідентифікувати вузькі місця, де користувачі стикаються з труднощами, і усувати їх за допомогою оптимізації дизайну чи функціоналу. Крім цього, сучасні тренди, такі як мінімалізм,

інклюзивність, адаптивний дизайн і інтерактивний контент, сприяють підвищенню залученості користувачів та їхньої загальної задоволеності.

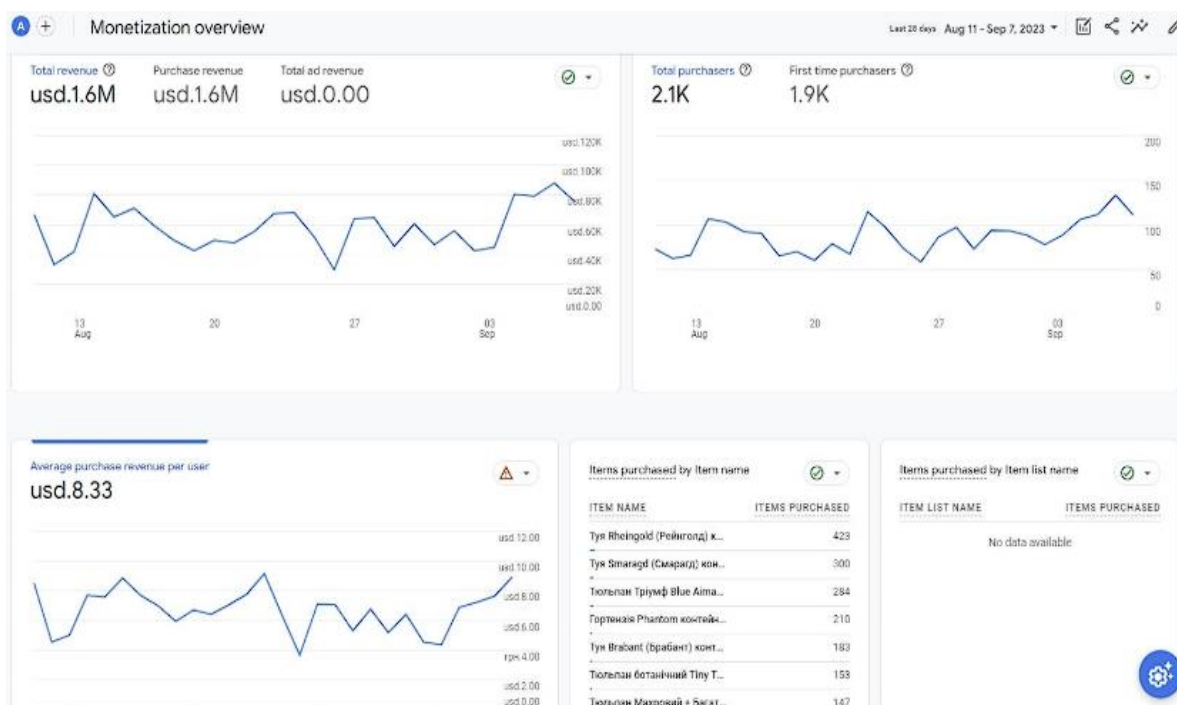


Рисунок 3.2 – Ілюстративна схема оптимізації UX

Крім того, регулярне тестування впроваджених змін із залученням реальних користувачів є важливим кроком для забезпечення високої якості UX. Результати таких досліджень можуть допомогти не лише покращити функціональність сайту, але й створити більш позитивне враження від взаємодії з брендом.

Ключові аспекти оптимізації UX:

Покращення навігації:

Забезпечення інтуїтивно зрозумілої навігації дозволяє користувачам швидко знаходити потрібні товари або послуги. Для цього важливо використовувати чітку категоризацію, функціональні фільтри та зручну пошукову систему. Наприклад, впровадження автозаповнення в рядку пошуку або системи рекомендацій на основі переглядів.

Мобільна адаптація:

З огляду на зростаючу популярність мобільних пристроїв, важливо забезпечити адаптивний дизайн, який гарантує коректне відображення сторінок на різних пристроях. Впровадження AMP (Accelerated Mobile Pages) може значно покращити швидкість завантаження мобільної версії.

Швидкість завантаження сторінок:

Швидкість завантаження є критичним фактором для утримання користувачів. Використання кешування, оптимізація зображень, скорочення кількості HTTP-запитів і впровадження CDN (Content Delivery Network) сприяють покращенню цього показника.

Персоналізація контенту:

Завдяки аналізу даних про поведінку користувачів, можна створювати персоналізовані пропозиції, рекомендації та акції. Наприклад, використання систем машинного навчання дозволяє автоматично формувати списки рекомендованих товарів для кожного клієнта.

Зручність оформлення замовлень:

Процес оформлення замовлення має бути простим і зрозумілим. Важливо зменшити кількість кроків, необхідних для завершення покупки, та забезпечити різноманітність способів оплати і доставки.

Оптимізація UX є невід'ємною складовою успішного розвитку інтернет-магазинів. Реалізація таких заходів, як покращення навігації, адаптивний дизайн, швидкість завантаження сторінок, персоналізація та зручний процес оформлення замовлень, дозволяє значно підвищити рівень задоволеності клієнтів. Крім цього, важливо враховувати такі фактори, як інтерактивність інтерфейсу, інклюзивність дизайну, використання сучасних аналітичних інструментів для моніторингу поведінки користувачів та впровадження зручних інструментів зворотного зв'язку для клієнтів. Додатково, використання технологій доповненої реальності (AR) та віртуальної реальності (VR) може надати користувачам можливість взаємодіяти з товарами у новий спосіб, що покращує залученість і створює унікальний досвід.

Також автоматизація обслуговування клієнтів через чат-боти та інші інструменти штучного інтелекту значно знижує навантаження на службу підтримки, забезпечуючи швидку та якісну допомогу.

У підсумку, ці зміни не лише сприяють зростанню лояльності клієнтів, але й дозволяють формувати сильний імідж бренду, збільшувати конверсії та створювати довготривалі конкурентні переваги у швидко змінному середовищі електронної комерції. Постійний аналіз результатів впроваджених рішень, їх оновлення та адаптація до змін ринку є ключем до підтримки успішного функціонування інтернет-магазину.

3.2 Забезпечення масштабованості з використанням Django

Масштабованість є однією з ключових характеристик успішного інтернет-магазину, особливо в умовах постійного зростання навантаження на систему через збільшення кількості користувачів, обсягів даних і транзакцій. Django забезпечує всі необхідні інструменти для створення масштабованих і стабільних рішень завдяки своїй модульній архітектурі, яка дозволяє розробникам розділяти додаток на автономні компоненти, зручні для розвитку та підтримки.

Одна з головних переваг Django — це вбудовані засоби роботи з базами даних, які підтримують ефективну взаємодію з реляційними базами (PostgreSQL, MySQL) та NoSQL-рішеннями (MongoDB, Redis). Django ORM (Object-Relational Mapping) оптимізує виконання запитів до бази даних, що забезпечує високу продуктивність навіть при значних обсягах даних. Крім того, можливість горизонтального масштабування бази даних, наприклад, через шардінг або реплікацію, допомагає справлятися зі зростаючим навантаженням. Django також підтримує інтеграцію з сучасними технологіями, такими як балансувальники навантаження (наприклад, Nginx чи AWS Elastic Load Balancer), кешування (Memcached, Redis) і обробка великих обсягів запитів у реальному часі через Celery або RabbitMQ. Завдяки цьому можна ефективно розподіляти навантаження між серверами, зменшувати час обробки запитів і підвищувати стійкість системи.

Для глобальних інтернет-магазинів важливим аспектом масштабованості є географічне розподілення ресурсів, що досягається через використання CDN (Content Delivery Network) для доставки статичних файлів, таких як зображення чи JavaScript. Django легко інтегрується з такими сервісами, як Cloudflare чи Amazon CloudFront, забезпечуючи швидке завантаження контенту для користувачів з різних регіонів. Завдяки підтримці контейнеризації за допомогою Docker і оркестрації через Kubernetes, Django-проекти можуть бути швидко масштабовані в хмарному середовищі, що дозволяє оперативно адаптуватися до пікових навантажень, наприклад, під час розпродажів чи акцій. Django надає розробникам усі необхідні засоби для створення масштабованих інтернет-магазинів, які залишаються стабільними й продуктивними навіть за умов інтенсивного зростання бізнесу. Фреймворк дозволяє реалізовувати стратегії горизонтального масштабування за допомогою розподілених систем, таких як Kubernetes чи Docker Swarm, які забезпечують гнучкість і надійність інфраструктури. Використання розподілених черг завдань, таких як Apache Kafka чи RabbitMQ, сприяє швидкій обробці запитів навіть при великому навантаженні. Django підтримує сучасні рішення для роботи з мікросервісами, що дозволяє розподіляти функціональні модулі системи між різними серверами. Це забезпечує стабільну роботу навіть за умови збоїв одного з компонентів системи. Оптимізація продуктивності включає в себе впровадження CDN для доставки статичних файлів, таких як зображення чи скрипти, що знижує навантаження на сервер і покращує швидкість завантаження сторінок.

Django надає широкий спектр можливостей для забезпечення масштабованості, що робить його ідеальним інструментом для розробки сучасних інтернет-магазинів. Завдяки своїй модульній архітектурі, цей фреймворк дозволяє легко розділяти проєкт на незалежні компоненти, які можуть масштабуватися автономно. Це спрощує розвиток системи в умовах зростання навантаження та появи нових функцій.

Одна з основних переваг Django — це ефективний механізм взаємодії з базами даних через вбудований ORM, який оптимізує запити та забезпечує високу продуктивність навіть при роботі з великими обсягами даних. Підтримка різних типів баз даних, таких як PostgreSQL, MySQL або NoSQL (MongoDB, Redis), дозволяє налаштовувати систему відповідно до потреб проєкту. Крім того, Django легко інтегрується з рішеннями для горизонтального масштабування, такими як шардінг чи реплікація баз даних, що забезпечує стабільну роботу під час збільшення обсягів транзакцій. Інтеграція з технологіями балансування навантаження, такими як Nginx або HAProxy, дозволяє розподіляти запити між кількома серверами, знижуючи навантаження на кожен окремий вузол. Використання кешування (наприклад, через Redis чи Memcached) допомагає зменшити кількість звернень до бази даних, прискорюючи відгук системи для користувачів. Для глобальних інтернет-магазинів важливо забезпечити однаково швидкий доступ до контенту з різних регіонів. Django легко інтегрується з CDN (Content Delivery Network) — такими як Cloudflare або Akamai, — для доставки статичних ресурсів, що забезпечує швидке завантаження сторінок незалежно від географічного розташування користувача.

Підтримка контейнеризації через Docker і оркестрації з Kubernetes дозволяє швидко розгортати нові екземпляри додатка в хмарних середовищах. Це особливо корисно в періоди пікових навантажень, наприклад, під час акцій або сезонних розпродажів, коли кількість користувачів може значно зростати.

Django також підтримує масштабовану роботу із завданнями в реальному часі через інтеграцію з Celery або RabbitMQ, що забезпечує асинхронну обробку даних і підвищує ефективність роботи. Усі ці функції дозволяють створювати високопродуктивні, стабільні та гнучкі системи, які легко адаптуються до змін і зростання бізнесу, роблячи Django надійною основою для сучасних інтернет-магазинів.

Основні аспекти забезпечення масштабованості:

Горизонтальне масштабування:

Django підтримує використання розподілених серверів для обробки запитів, що дозволяє збільшувати продуктивність системи в умовах підвищеного навантаження. Для цього застосовуються технології балансування навантаження, наприклад Nginx або AWS Elastic Load Balancer.

Системи кешування:

Інтеграція Django з Redis або Memcached дозволяє значно зменшити час обробки запитів, зберігаючи часто використовувані дані в оперативній пам'яті. Це критично важливо для інтернет-магазинів, де швидкість відповіді впливає на користувацький досвід.

Асинхронна обробка завдань:

Використання Celery у поєднанні з брокером повідомлень, таким як RabbitMQ, забезпечує виконання складних завдань у фоновому режимі, знижуючи навантаження на основний сервер.

Оптимізація бази даних:

Django ORM дозволяє ефективно працювати з великими обсягами даних, але для максимізації продуктивності рекомендується використовувати індексацію, шардінг та реплікацію баз даних.

Моніторинг і аналіз:

Впровадження інструментів моніторингу, таких як Prometheus або New Relic, дозволяє виявляти вузькі місця та оптимізувати продуктивність системи в реальному часі.

Приклад інтеграції кешування з Redis:

```
from django.core.cache import cache
# Збереження даних у кеші
cache.set('key', 'value', timeout=60*15)
# Отримання даних із кешу
value = cache.get('key')
```



```

if value is None:
    value = 'default_value'
    cache.set('key', value)

```

Використання Django для забезпечення масштабованості дозволяє створювати надійні та продуктивні інтернет-магазини, здатні обробляти великі обсяги даних і трафіку. Інтеграція з сучасними технологіями, такими як кешування, асинхронна обробка завдань, розподіл навантаження та мікросервісна архітектура, гарантує стабільну роботу системи навіть за умов пікових навантажень. Використання хмарних платформ, таких як AWS, Google Cloud чи Azure, дозволяє динамічно масштабувати ресурси залежно від поточного навантаження, забезпечуючи ефективність витрат. Додатково, регулярне впровадження інструментів моніторингу та аналізу, таких як Grafana чи Splunk, сприяє своєчасному виявленню потенційних вузьких місць у системі. Впровадження механізмів автоматичного відновлення після збоїв (self-healing) підвищує надійність платформи та мінімізує час простою. Постійний аналіз результатів роботи системи дозволяє ідентифікувати критичні точки для оптимізації, забезпечуючи довготривалий успіх проекту.

3.3 Використання інноваційних технологій у Django-проектах

Інноваційні технології відіграють ключову роль у розвитку сучасних Django-проектів, забезпечуючи їх конкурентоспроможність і відповідність актуальним потребам користувачів. Django є гнучким фреймворком, який дозволяє легко інтегрувати новітні технології, такі як штучний інтелект, машинне навчання, хмарні сервіси, блокчейн, автоматизація бізнес-процесів, обробка великих даних (Big Data), а також сучасні рішення в галузі кібербезпеки та інтеграції з мікросервісною архітектурою. Це фреймворк, що дозволяє адаптуватися до швидко змінюваних тенденцій, забезпечуючи можливість впровадження аналітичних систем для моніторингу, технологій доповненої та віртуальної реальності для створення унікального користувацького досвіду, а також автоматизації процесів за

допомогою штучного інтелекту. Розробники мають змогу інтегрувати складні технологічні рішення, такі як системи рекомендацій, платіжні шлюзи з підтримкою криптовалют, та забезпечувати розширену взаємодію з IoT-пристроями, що робить Django потужним інструментом для створення інноваційних продуктів.

Основні напрями використання інновацій:

Штучний інтелект і машинне навчання:

Django забезпечує інтеграцію з популярними бібліотеками Python, такими як TensorFlow, Scikit-learn та PyTorch, що відкриває широкі можливості для впровадження інтелектуальних функцій у сучасні інтернет-магазини. Це дозволяє використовувати потужність машинного навчання для підвищення ефективності бізнес-процесів та покращення користувацького досвіду. Одним із найпоширеніших застосувань є персоналізація контенту. Моделі машинного навчання, інтегровані через Django, аналізують поведінку користувачів, їхні попередні покупки та перегляди, щоб формувати індивідуальні рекомендації. Це підвищує конверсію, оскільки покупцям пропонуються товари, які відповідають їхнім інтересам і потребам. Додатково, такі моделі використовуються для прогнозування попиту на товари. За допомогою бібліотек TensorFlow чи Scikit-learn можна створювати прогнози на основі історичних даних про продажі, сезонних трендів або зовнішніх факторів, як-от святкові періоди. Це допомагає оптимізувати управління запасами, знижуючи витрати на зберігання та запобігаючи дефіциту товарів.

Ще однією важливою сферою є виявлення шахрайських транзакцій. Використовуючи PyTorch або інші бібліотеки для розробки моделей класифікації, можна аналізувати патерни транзакцій у реальному часі, виявляючи підозрілі операції. Це забезпечує додатковий рівень безпеки для клієнтів та компанії.

Завдяки своїй архітектурі Django дозволяє безшовно інтегрувати ці моделі у бізнес-логіку додатків. Обчислення можуть виконуватися як на сервері, так і у хмарі, залежно від обсягів даних і вимог до продуктивності. Таким чином, Django

створює ідеальне середовище для використання штучного інтелекту в електронній комерції, забезпечуючи сучасний функціонал і високий рівень адаптивності.

Хмарні сервіси:

Інтеграція Django з хмарними платформами, такими як AWS, Google Cloud або Azure, дозволяє ефективно масштабувати проекти, знижувати витрати на інфраструктуру та забезпечувати високу доступність системи. Наприклад, використання хмарних баз даних або обчислювальних ресурсів допомагає обробляти великі обсяги даних без необхідності інвестицій у фізичну інфраструктуру.

Блокчейн:

Django дозволяє створювати додатки з підтримкою технології блокчейн, що відкриває нові можливості для безпечних транзакцій, токенизації товарів та інтеграції криптовалют. Завдяки своїй гнучкості та інтеграції з популярними бібліотеками Python, такими як Web3.py, Django стає потужним інструментом для розробки сучасних рішень на основі блокчейну. Використання блокчейну забезпечує високий рівень безпеки транзакцій завдяки його децентралізованій природі та незмінності даних. У контексті інтернет-магазинів це дозволяє знизити ризики шахрайства, гарантувати прозорість угод і підвищити довіру клієнтів. Наприклад, оплата криптовалютою може бути інтегрована через бібліотеки, які дозволяють працювати з популярними блокчейн-мережами, такими як Ethereum або Bitcoin.

Токенізація товарів є ще однією важливою сферою застосування. За допомогою смарт-контрактів можна створювати унікальні цифрові активи (NFT), які представляють фізичні чи цифрові товари. Це особливо актуально для ринків предметів колекціонування, творів мистецтва або ексклюзивних товарів, де потрібно підтвердження автентичності. Django дозволяє інтегрувати такі рішення, забезпечуючи зручний інтерфейс для управління токенами, відстеження транзакцій і взаємодії зі смарт-контрактами.

Для інтернет-магазинів, які працюють на міжнародному рівні, інтеграція криптовалют відкриває доступ до нових ринків, де традиційні платіжні методи можуть бути недоступними або обмеженими. Django дозволяє інтегрувати платіжні шлюзи для криптовалют, наприклад, Coinbase Commerce або BitPay, забезпечуючи клієнтам зручність і гнучкість у виборі способу оплати.

Завдяки своїй архітектурі Django дозволяє не лише реалізувати базові функції блокчейну, але й забезпечувати масштабованість таких рішень для роботи з великими обсягами даних і транзакцій. Це робить його ідеальним вибором для створення інноваційних додатків в електронній комерції, орієнтованих на використання технологій майбутнього.

Інтернет речей (IoT):

Завдяки використанню Django REST Framework, можна розробляти API для взаємодії з IoT-пристроями, що дозволяє автоматизувати процеси, наприклад, відстеження товарів на складі чи управління логістикою.

Розширена та віртуальна реальність (AR/VR):

Інтеграція Django із зовнішніми бібліотеками для AR/VR відкриває нові можливості для створення інтерактивних каталогів товарів, що дозволяє значно покращити користувацький досвід. Завдяки цьому користувачі можуть оглядати продукцію у віртуальному середовищі, що наближає онлайн-шопінг до реального взаємодії з товарами. Технології доповненої (AR) і віртуальної реальності (VR) дозволяють інтегрувати 3D-моделі товарів у веб-додаток, що працює на основі Django. З використанням бібліотек, таких як Three.js, A-Frame чи AR.js, можна створювати інтерактивні інтерфейси, де клієнти можуть переглядати товар з усіх боків, змінювати його кольори чи розміри, а також перевіряти, як він виглядатиме у реальному середовищі через AR. Django у цьому контексті слугує серверною основою для обробки запитів, зберігання даних про моделі, а також надання API для динамічного завантаження контенту.

Наприклад, у меблевому інтернет-магазині AR дозволяє клієнтам «розміщувати» віртуальні моделі меблів у своїх кімнатах через смартфон чи

планшет, оцінюючи їхній вигляд у реальному просторі. VR, у свою чергу, дає можливість створити віртуальні шоуруми, де користувачі можуть «прогулятися» між товарами у цифровому середовищі, отримуючи детальну інформацію про кожен із них. Для впровадження таких рішень Django легко інтегрується з інструментами 3D-рендерингу та платформами для AR/VR. Використання REST API чи GraphQL дозволяє передавати дані про товари у форматі, що підтримує рендеринг 3D-моделей. Крім того, зберігання 3D-файлів, таких як GLTF чи FBX, може бути організоване через хмарні сервіси або локальні сховища, що інтегруються з Django.

Такі інновації не лише підвищують залученість користувачів, а й допомагають клієнтам приймати більш обґрунтовані рішення про покупку. Це особливо актуально для індустрій, де важливе візуальне сприйняття товару, наприклад, моди, меблів чи електроніки. Django, як універсальний фреймворк, забезпечує стабільність і гнучкість у реалізації таких сучасних рішень.

Приклад інтеграції машинного навчання в Django-проекті:

```
from sklearn.linear_model import LinearRegression
import numpy as np

# Дані для навчання
X = np.array([[1], [2], [3], [4]])
y = np.array([100, 200, 300, 400])

# Створення моделі
model = LinearRegression()
model.fit(X, y)

# Прогнозування
predicted_value = model.predict([[5]])[0]
print(f"Прогнозована ціна: {predicted_value}")
```

Використання інноваційних технологій у Django-проектах дозволяє створювати сучасні, продуктивні та конкурентоспроможні рішення. Інтеграція штучного інтелекту, хмарних сервісів, блокчейну, IoT, технологій великих даних (Big Data) та автоматизації забезпечує широкий спектр можливостей для розвитку бізнесу. Ці технології допомагають вирішувати складні бізнес-задачі, оптимізувати процеси, покращувати взаємодію з клієнтами та розширювати функціонал систем. Наприклад, використання Big Data дозволяє аналізувати великі обсяги даних для створення прогнозів щодо поведінки користувачів, оптимізації запасів на складах і підвищення ефективності рекламних кампаній. Технології штучного інтелекту та машинного навчання сприяють автоматизації рутинних завдань, таких як сортування запитів клієнтів або управління ціноутворенням. Хмарні сервіси забезпечують масштабованість та гнучкість, знижуючи витрати на інфраструктуру.

Інноваційні рішення у Django-проектах створюють умови для швидкої адаптації до нових викликів і потреб ринку, що є важливим чинником для довготривалого успіху в умовах сучасної цифрової економіки.

3.4 Висновки по розділу

Розділ "Опис покращень для існуючих інтернет-магазинів" демонструє, як сучасні технології та підходи можуть трансформувати інтернет-магазини, забезпечуючи їх ефективність, конкурентоспроможність та адаптивність. Інновації, такі як інтеграція штучного інтелекту, хмарних сервісів, блокчейну, IoT і технологій великих даних, є фундаментальними для розвитку електронної комерції. Ці технології дозволяють автоматизувати процеси, покращувати аналітику, персоналізувати взаємодію з користувачами та забезпечувати масштабованість систем.

Наприклад, використання машинного навчання для персоналізації контенту або прогнозування попиту дозволяє підвищити якість обслуговування клієнтів і ефективність бізнес-процесів. Хмарні сервіси допомагають знизити витрати на інфраструктуру та забезпечити стабільність системи навіть за умов пікових

навантажень. Інтеграція блокчейну забезпечує підвищений рівень безпеки та прозорості транзакцій, а використання IoT дозволяє автоматизувати логістичні процеси та покращувати управління складськими запасами.

Впровадження цих технологій сприяє створенню унікального користувацького досвіду, що є важливим фактором для залучення та утримання клієнтів. Завдяки постійному моніторингу інновацій та їх адаптації до змін ринку інтернет-магазини можуть залишатися конкурентоспроможними у динамічному середовищі електронної комерції.

4 МАТЕМАТИЧНА МОДЕЛЬ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ

4.1 Алгоритми аналізу даних у комерційних системах

Аналіз даних є фундаментальним компонентом сучасних комерційних систем, оскільки він дозволяє отримувати цінну інформацію для прийняття рішень, оптимізації процесів та підвищення прибутковості бізнесу. Його значення полягає у здатності розуміти клієнтську поведінку, прогнозувати тренди, оптимізувати ресурси та виявляти аномалії. Наприклад, аналіз покупок і взаємодії користувачів із платформою дозволяє виявляти їхні переваги, формувати персоналізовані пропозиції, а також прогнозувати попит чи планувати запуск нових продуктів. Завдяки цьому компанії можуть підвищувати ефективність роботи, скорочувати витрати та приймати обґрунтовані стратегічні рішення.

У Django-проектах аналіз даних базується на збиранні, очищенні, трансформації та інтерпретації великих обсягів інформації. Важливу роль у цьому відіграють бібліотеки Python, такі як Pandas, NumPy та Scikit-learn. Pandas забезпечує гнучке опрацювання табличних даних, їх фільтрацію та агрегацію, тоді як NumPy використовується для виконання складних математичних операцій із числовими масивами. Scikit-learn, у свою чергу, дозволяє застосовувати алгоритми машинного навчання для розв'язання завдань класифікації, регресії та кластеризації, що є важливим для прогнозування та моделювання. Для кращого розуміння тенденцій у даних активно застосовуються бібліотеки Matplotlib і Seaborn, які дозволяють створювати наочні візуалізації, наприклад графіки, гістограми або теплові карти. Django забезпечує можливість інтеграції з потужними інструментами бізнес-аналітики, такими як Tableau або Power BI, які дають змогу створювати інтерактивні дашборди та виконувати багатовимірний аналіз. У випадках, коли потрібна обробка даних у реальному часі, Django легко інтегрується з потоковими платформами, такими як Apache Kafka або RabbitMQ, що особливо актуально для моніторингу, прогнозування або виявлення шахрайства.

Аналіз даних у рамках Django-проектів охоплює весь цикл роботи з даними — від збору до стратегічного прийняття рішень. Це робить Django ефективним інструментом для реалізації складних і високонавантажених комерційних систем, які базуються на сучасних підходах до роботи з даними.

У комерційних системах дані часто організовані у вигляді багатовимірних масивів або таблиць, де кожен рядок представляє окрему транзакцію, а стовпці містять атрибути, такі як час, вартість, категорія товару тощо. Кожен з цих атрибутів може слугувати основою для аналізу різних аспектів діяльності системи. Наприклад, час транзакції дозволяє визначати пікові періоди продажів, категорія товару — виявляти найбільш популярні продукти, а вартість — аналізувати середній чек клієнта. Організація даних у таких структурах сприяє більш глибокому аналізу за допомогою агрегатних функцій, зокрема обчислення суми продажів за категоріями, середнього значення за період або частки певної категорії у загальному обсязі продажів.

Приклад обробки даних:

```
import pandas as pd
# Завантаження даних
transactions = pd.DataFrame({
    'timestamp': ['2024-12-10 10:00', '2024-12-10 10:05', '2024-12-10 10:10'],
    'amount': [150, 200, 300],
    'category': ['electronics', 'clothing', 'electronics']
})
# Додавання нового стовпця "податок"
transactions['tax'] = transactions['amount'] * 0.2
# Фільтрація за категорією "electronics"
electronics_transactions = transactions[transactions['category']
== 'electronics']
print(electronics_transactions)
```

Для прогнозування комерційних показників, таких як майбутній попит на товари чи обсяги продажів, використовуються моделі регресії.

Математична модель лінійної регресії:

Нехай дано набір спостережень, де x — незалежна змінна (наприклад, час), а y — залежна змінна (наприклад, кількість продажів). Лінійна регресія шукає функцію виду:

де a і b — параметри моделі, які мінімізують функцію втрат:

Реалізація в Python:

```
from sklearn.linear_model import LinearRegression
import numpy as np
# Дані для навчання
X = np.array([[1], [2], [3], [4]]) # Час
y = np.array([100, 200, 300, 400]) # Обсяги продажів
# Навчання моделі
model = LinearRegression()
model.fit(X, y)
# Прогноз для нового значення часу
future_sales = model.predict([[5]])[0]
print(f"Прогнозований обсяг продажів: {future_sales}")
```

Аналіз даних у комерційних системах також включає кластеризацію для сегментації клієнтів, аналіз часових рядів для прогнозування трендів і виявлення аномалій у даних. Кластеризація дозволяє розділяти клієнтів на групи за схожими характеристиками, що полегшує персоналізацію пропозицій та ефективне управління маркетинговими кампаніями. Аналіз часових рядів дає змогу не лише прогнозувати майбутні тренди продажів, але й виявляти сезонні коливання, які можуть впливати на бізнес-процеси.

Додатково, алгоритми виявлення аномалій допомагають ідентифікувати підозрілі транзакції, що є важливим для забезпечення безпеки комерційних систем. Django надає зручні інструменти для інтеграції цих алгоритмів у веб-додатки,

зокрема через використання Django REST Framework для створення API, які забезпечують взаємодію з зовнішніми системами аналізу. Це дозволяє автоматизувати бізнес-процеси, покращувати ефективність роботи системи та забезпечувати високий рівень задоволеності користувачів..

4.2 Прогнозування продажів на основі статистичних даних

Прогнозування продажів є важливим інструментом для підвищення ефективності управління комерційними системами, оскільки воно дозволяє прогнозувати попит, оптимізувати запаси та знижувати ризики надлишкового виробництва або дефіциту товарів. Застосування прогнозування допомагає бізнесу приймати обґрунтовані рішення щодо планування закупівель, управління маркетинговими кампаніями та розробки стратегій цінової політики. У сучасних комерційних системах для прогнозування продажів використовуються статистичні моделі, алгоритми машинного навчання, аналіз часових рядів та гібридні підходи, які поєднують декілька методів для досягнення високої точності прогнозів.

Теоретичне обґрунтування

Основою прогнозування продажів є математичні моделі, які використовують дані про попередні періоди для створення прогнозів. Одним із найпоширеніших підходів є використання лінійної регресії, яка дозволяє виявляти залежності між змінними, такими як час та обсяги продажів. Цей метод забезпечує базову оцінку трендів, але для складніших завдань, таких як моделювання сезонності або врахування впливу зовнішніх факторів, застосовуються більш просунуті методи. Наприклад, автокореляційні функції (ACF) дозволяють аналізувати зв'язки між елементами часових рядів, що є корисним для визначення періодичних патернів.

Більш точні прогнози можуть бути отримані з використанням алгоритмів ARIMA (AutoRegressive Integrated Moving Average), які враховують як тренди, так і сезонні коливання. Для адаптивних прогнозів, де важливо врахувати багатфакторний вплив, застосовуються ансамблеві алгоритми, такі як Random

Forest або XGBoost, що інтегрують багаторівневий аналіз даних та здатні опрацьовувати великі обсяги інформації з високою точністю.

Математична модель ARIMA:

ARIMA є одним із найпотужніших інструментів для аналізу часових рядів. Модель описується рівнянням:

де y_t — значення ряду в момент часу t , ϕ — коефіцієнти авторегресії, θ — коефіцієнти скользячого середнього, ϵ_t — шум.

Реалізація в Python:

```
from statsmodels.tsa.arima.model import ARIMA
import pandas as pd
# Імітація даних продажів
data = pd.Series([100, 120, 130, 125, 150, 170, 165])
# Створення моделі ARIMA
model = ARIMA(data, order=(2, 1, 2))
model_fit = model.fit()
# Прогнозування на 3 періоди вперед
forecast = model_fit.forecast(steps=3)
print("Прогнозовані продажі:", forecast)
# Візуалізація результатів
import matplotlib.pyplot as plt
plt.plot(data, label="Історичні дані")
plt.plot(range(len(data), len(data) + len(forecast)), forecast,
label="Прогноз")
plt.legend()
plt.show()
# Додаткова інформація про модель
print(model_fit.summary())
```

Реалізація з використанням Random Forest:

```
from sklearn.ensemble import RandomForestRegressor
```

```

import numpy as np
# Дані для навчання
X = np.array([[1], [2], [3], [4], [5]]) # Дні
y = np.array([100, 150, 200, 250, 300]) # Продажі
# Навчання моделі
rf_model = RandomForestRegressor(n_estimators=100,
random_state=42)
rf_model.fit(X, y)
# Прогноз на наступні 3 дні
future_days = np.array([[6], [7], [8]])
rf_forecast = rf_model.predict(future_days)
print("Прогнозовані продажі (Random Forest):", rf_forecast)
# Візуалізація
plt.scatter(X, y, color='blue', label='Історичні дані')
plt.plot(future_days, rf_forecast, color='red', label='Прогноз')
plt.legend()
plt.show()

```

Застосування в реальних системах

Прогнозування продажів у реальних системах вимагає врахування багатьох факторів, таких як сезонність, акційні пропозиції, зовнішні економічні умови, рівень конкурентного середовища, поведінка клієнтів та зміни у ринкових тенденціях. Для врахування цих факторів часто використовують ансамблеві методи машинного навчання, такі як Random Forest чи XGBoost, які дозволяють враховувати складну взаємодію між змінними. Наприклад, метод Random Forest дозволяє створювати набір дерев рішень, які оцінюють вплив кожної змінної на результат, а XGBoost надає інструменти для роботи зі складними даними, забезпечуючи високу швидкість навчання та точність прогнозів. Додатково, у таких системах можуть застосовуватися моделі глибокого навчання (Deep Learning), наприклад, LSTM (Long Short-Term Memory) для аналізу часових рядів із довгостроковими залежностями. Це дозволяє враховувати як короткострокові

коливання, так і довготривалі тренди, що забезпечує більш точне прогнозування у динамічних ринкових умовах.

Інтеграція в Django

У контексті Django прогнозування може бути інтегроване через API, які дозволяють автоматично обробляти нові дані, навчати моделі й генерувати прогнози у режимі реального часу. Наприклад, Django REST Framework (Рис.4.1) забезпечує створення ендпоінтів для доступу до прогнозів, що є особливо корисним для великих комерційних платформ.

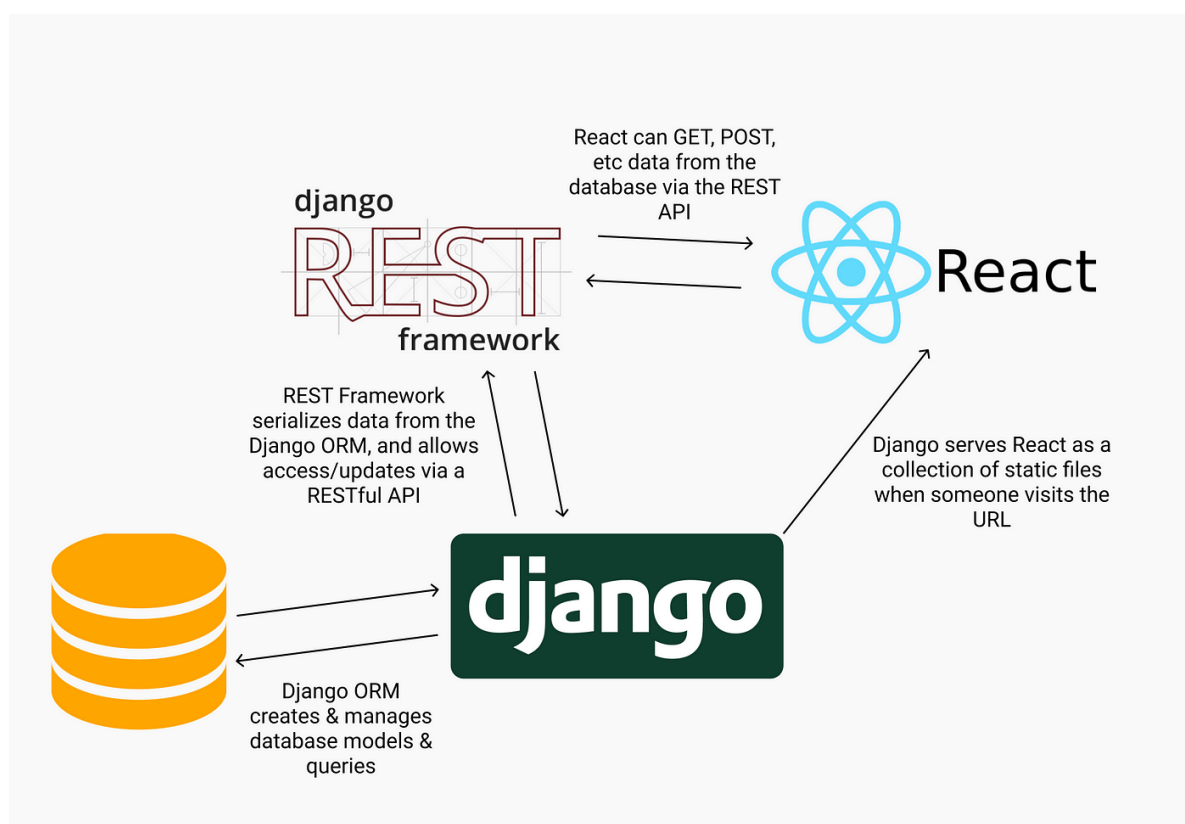


Рисунок 4.1 – Django REST Framework

Цей підхід дозволяє гнучко управляти процесами прогнозування: моделі можуть періодично оновлюватися на основі нових даних, що підвищує їхню точність і релевантність. Використання асинхронних технологій, таких як Celery у поєднанні з Redis, дозволяє обробляти складні запити або виконувати навчання моделі у фоновому режимі. Це допомагає розподілити навантаження на сервер і

зберігати продуктивність системи навіть за умов пікових навантажень. У реальних умовах такі API можуть бути інтегровані з панелями адміністратора для управління моделями, а також з аналітичними інструментами для візуалізації результатів прогнозів і прийняття рішень на їх основі.

Використання статистичних моделей і сучасних алгоритмів машинного навчання відкриває широкі можливості для оптимізації бізнес-процесів, забезпечуючи точні прогнози для управління продажами, мінімізації ризиків втрат і підвищення ефективності використання ресурсів. Такі технології дозволяють аналізувати великі обсяги даних, виявляючи приховані закономірності та тенденції, які можуть бути недоступними при традиційних методах аналізу. Інтеграція статистичних моделей у комерційні системи дає змогу значно вдосконалити процес прийняття рішень. Наприклад, алгоритми прогнозування попиту дозволяють точно оцінити майбутні потреби в продукції, що допомагає уникати дефіциту чи надлишків товарів на складах. Оптимізаційні моделі підтримують ефективне планування логістики та управління ланцюгами постачання, знижуючи операційні витрати та час доставки.

Крім того, алгоритми класифікації й аналізу ризиків використовуються для ідентифікації потенційно шахрайських транзакцій або визначення груп клієнтів із високим ризиком відтоку. Це дозволяє вчасно вживати заходів для їх утримання, пропонуючи персоналізовані акції чи бонуси.

Гнучкість і адаптивність сучасних алгоритмів є особливо важливими в умовах швидких змін у ринкових умовах. Вони дозволяють бізнесу оперативно реагувати на зміни попиту, конкуренції чи зовнішніх факторів, таких як сезонність або економічні кризи. Інтеграція таких рішень у комерційні системи не лише підвищує продуктивність, але й створює конкурентну перевагу, сприяючи побудові більш стійкої та ефективної бізнес-моделі.

Завдяки таким підходам, компанії отримують інструменти для підвищення продуктивності, зниження витрат та збільшення доходів, що сприяє довгостроковій стійкості й успіху на динамічному ринку.

4.3 Інтеграція математичних моделей у веб-додаток на Django

Інтеграція математичних моделей у веб-додаток є важливим етапом створення систем, що базуються на аналізі даних і прогнозуванні. Django, завдяки своїй гнучкості та підтримці бібліотек Python, забезпечує ефективне впровадження складних математичних моделей у комерційні додатки. Це включає автоматизацію розрахунків, створення інтерфейсів для взаємодії з користувачами, організацію обробки великих обсягів даних у реальному часі та адаптацію до змінюваних бізнес-вимог. Завдяки широким можливостям інтеграції, Django дозволяє використовувати алгоритми машинного навчання, моделі прогнозування та методи аналізу даних для створення комплексних рішень. Розробка таких систем не лише підвищує точність і ефективність прийняття рішень, але й сприяє підвищенню конкурентоспроможності бізнесу, забезпечуючи клієнтам інноваційний користувацький досвід.

Основні етапи інтеграції

Першим етапом є розробка математичної моделі, яка відповідає бізнес-завданню. Наприклад, для прогнозування продажів використовується ARIMA, Random Forest або нейронні мережі. Наступним кроком є створення сервісу для обробки даних і генерації прогнозів. Django REST Framework дозволяє створювати API для взаємодії з математичною моделлю.

Математична модель прогнозування продажів

Модель прогнозування базується на часових рядах і визначається рівнянням ARIMA:

де y_t — значення ряду, ϕ — коефіцієнти авторегресії, θ — коефіцієнти скользячого середнього, ϵ_t — шум.

Приклад реалізації сервісу для прогнозування

```
from django.http import JsonResponse
from statsmodels.tsa.arima.model import ARIMA
import pandas as pd

# Функція для прогнозування
def forecast_sales(request):
    # Дані
    data = pd.Series([100, 120, 130, 125, 150, 170, 165])
    # Модель ARIMA
    model = ARIMA(data, order=(2, 1, 2))
    model_fit = model.fit()
    # Прогноз
    forecast = model_fit.forecast(steps=3)
    # Відповідь JSON
    return JsonResponse({"forecast": forecast.tolist()})
```

Інтеграція нейронної мережі для прогнозування

Глибинне навчання часто використовується для складних прогнозів. Модель LSTM (Long Short-Term Memory) дозволяє враховувати довготривалі залежності в даних.

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
import numpy as np

# Дані
X = np.array([[1], [2], [3], [4], [5]]) # Час
Y = np.array([100, 150, 200, 250, 300]) # Продажі
# Форматування для LSTM
X = X.reshape((X.shape[0], 1, X.shape[1]))
# Створення моделі
model = Sequential()
```

```

model.add(LSTM(50, activation='relu', input_shape=(1, 1)))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')
# Навчання
model.fit(X, Y, epochs=200, verbose=0)
# Прогноз
future = np.array([[6]])
future = future.reshape((future.shape[0], 1, future.shape[1]))
forecast = model.predict(future)
print("Прогнозовані продажі:", forecast)

```

Створення API для нейронної мережі

```

from django.views.decorators.csrf import csrf_exempt
from django.http import JsonResponse
import numpy as np
@csrf_exempt
def predict_sales(request):
    # Дані з запиту
    data = request.POST.getlist('sales_data', [])
    data = np.array(data, dtype=float).reshape((-1, 1, 1))
    # Прогноз
    forecast = model.predict(data)
    return JsonResponse({"forecast": forecast.tolist()})

```

Переваги інтеграції

- Автоматизація процесів. Веб-додаток забезпечує автоматичну обробку даних і прогнозування.
- Доступність у реальному часі. API дозволяє клієнтам отримувати прогнози миттєво.
- Масштабованість. Django забезпечує стабільну роботу сервісів навіть при високих навантаженнях.

Інтеграція математичних моделей у веб-додатки на основі Django дозволяє створювати адаптивні, високоефективні та масштабовані рішення для аналізу даних. Це включає автоматизацію прогнозування, інтеграцію моделей машинного навчання, побудову складних аналітичних панелей для моніторингу даних та забезпечення аналітичної підтримки у реальному часі. Django забезпечує зручну інфраструктуру для обробки великих обсягів даних і дозволяє швидко адаптуватися до змін у бізнес-середовищі. Крім прогнозування продажів, такі системи можуть включати модулі для виявлення аномалій, кластеризації клієнтів, оптимізації логістики та управління ресурсами. Наприклад, використання моделей нейронних мереж для аналізу взаємодії з користувачами може сприяти покращенню користувацького досвіду, а інтеграція із зовнішніми сервісами, такими як платіжні шлюзи або CRM, значно розширює функціональність додатка. Завдяки цьому бізнес може не лише швидко реагувати на зміни ринку, але й прогнозувати їх, що створює конкурентну перевагу та сприяє довгостроковій стабільності.

4.4 Висновки по розділу

Розділ "Математична модель аналізу та прогнозування" висвітлив важливість використання сучасних математичних методів і алгоритмів у комерційних системах. Описано різноманітні підходи до аналізу даних, такі як використання часових рядів, статистичних моделей і машинного навчання для створення точних прогнозів та оптимізації бізнес-процесів.

Інтеграція математичних моделей у веб-додатки на основі Django була продемонстрована як гнучкий та ефективний підхід, що дозволяє автоматизувати складні обчислення, створювати адаптивні рішення та забезпечувати підтримку прийняття рішень у реальному часі. Використання бібліотек Python, таких як TensorFlow, Scikit-learn та Statsmodels, полегшує впровадження складних алгоритмів машинного навчання та аналізу часових рядів. Моделі прогнозування, такі як ARIMA, нейронні мережі LSTM і Random Forest, забезпечують точність

прогнозів, враховуючи фактори сезонності, трендів, впливу акційних пропозицій і змін у поведінці клієнтів. Крім того, інтеграція таких моделей через API дозволяє автоматично обробляти нові дані, генерувати прогнози та візуалізувати результати у режимі реального часу. Це сприяє створенню систем, що підвищують гнучкість бізнесу та полегшують адаптацію до змін ринкових умов.

Розглянуто практичну реалізацію цих моделей у контексті веб-додатків, включаючи створення API для доступу до результатів прогнозування. Такий підхід дозволяє створювати масштабовані рішення, що інтегруються з іншими бізнес-системами, забезпечуючи високий рівень продуктивності та точності.

Загалом, математичні моделі аналізу та прогнозування є невід'ємною частиною сучасних комерційних систем, що дозволяє бізнесу ефективно управляти ресурсами, реагувати на зміни ринку та забезпечувати конкурентоспроможність у динамічному середовищі.

ВИСНОВКИ

У даній роботі було розглянуто процес створення інтернет-магазину одягу з використанням веб-фреймворку Django. Об'єктом дослідження став процес створення інтернет-магазину, який охоплює архітектуру системи, функціональні можливості платформи та методи її адаптації до сучасних викликів електронної комерції. У ході роботи було успішно виконано мету дослідження, яка полягала у створенні теоретичної моделі інтернет-магазину, що відповідає сучасним стандартам продуктивності, зручності та безпеки. У межах дослідження проведено аналіз можливостей Django, спроектовано структуру системи та запропоновано рекомендації щодо її вдосконалення.

У процесі роботи було детально досліджено актуальність використання інтернет-магазинів у сучасній економіці, зокрема зростаючу роль електронної комерції у глобальних бізнес-процесах. Проведено глибокий аналіз предметної області, виявлено ключові потреби користувачів і бізнесу, а також визначено основні технологічні тренди у веб-розробленні, такі як адаптивний дизайн, масштабованість і безпека. Описано архітектуру веб-додатків із врахуванням вимог до швидкості обробки запитів, інтеграції з зовнішніми сервісами та забезпечення високого рівня надійності. Особливу увагу приділено використанню Django, яке дозволяє легко впроваджувати сучасні підходи до моделювання баз даних, що сприяє гнучкості, масштабованості та простоті підтримки системи.

Дослідження математичних моделей аналізу та прогнозування продемонструвало, що використання алгоритмів машинного навчання, таких як Random Forest, XGBoost, аналізу часових рядів із застосуванням ARIMA та інтеграції цих моделей у Django дозволяє створювати потужні й ефективні рішення для аналізу та прогнозування бізнес-показників. Ці моделі враховують багатофакторні залежності, сезонність, тренди й випадкові відхилення, що сприяє більш точному прогнозуванню. Впровадження таких моделей забезпечує автоматизацію бізнес-процесів, підвищення точності прогнозів, оперативне

реагування на зміни ринку та адаптацію системи до нових умов. Крім того, моделі можуть інтегруватися з інструментами візуалізації для створення аналітичних панелей, які полегшують прийняття рішень на основі даних.

Інтеграція математичних моделей у веб-додаток з використанням Django REST Framework продемонструвала ефективність створення API для взаємодії з математичними обчисленнями. Цей підхід забезпечує доступність системи у реальному часі, її масштабованість і інтеграцію з іншими бізнес-сервісами, такими як платіжні системи, CRM чи аналітичні панелі. Важливо зазначити, що такий підхід дозволяє не лише автоматизувати складні обчислення, але й адаптувати систему до роботи з великими обсягами даних, забезпечуючи зручність у масштабуванні та розширенні функціоналу. Крім того, API забезпечує простоту у впровадженні нових бізнес-логік, що сприяє оперативному реагуванню на зміну потреб ринку та вимог клієнтів.

Результати дослідження підтверджують ефективність використання Django для створення інтернет-магазинів одягу. Фреймворк забезпечує необхідний рівень безпеки, продуктивності та зручності, відповідаючи вимогам сучасної електронної комерції. Впровадження рекомендованих технологічних рішень дозволяє автоматизувати ключові бізнес-процеси, забезпечувати персоналізацію користувацького досвіду та адаптацію до змін ринкових умов. Запропоновані рішення можуть бути використані не лише для розробки інтернет-магазинів, але й для створення комплексних комерційних платформ, що інтегруються з іншими системами, такими як аналітичні сервіси, платіжні шлюзи та CRM. Це сприяє підвищенню їх конкурентоспроможності, стабільності та гнучкості у відповідь на виклики ринку.

ПЕРЕЛІК ПОСИЛАНЬ

1. Django Software Foundation. Django Documentation Release 4.2 // Online Resource. – Available at: <https://docs.djangoproject.com/>. – 2024.
2. Mitchell T. Machine Learning. – McGraw-Hill Science, 1997. – 414 p.
3. Goodfellow I., Bengio Y., Courville A. Deep Learning. – MIT Press, 2016. – 800 p.
4. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning. – Springer, 2009. – 745 p.
5. Pedregosa F. et al. Scikit-learn: Machine Learning in Python // Journal of Machine Learning Research. – 2011. – Vol. 12. – P. 2825-2830.
6. Russell S., Norvig P. Artificial Intelligence: A Modern Approach. – Pearson, 2020. – 1152 p.
7. McKinney W. Python for Data Analysis. – O'Reilly Media, 2022. – 550 p.
8. Han J., Kamber M., Pei J. Data Mining: Concepts and Techniques. – Morgan Kaufmann, 2011. – 744 p.
9. Zhang C., Ma Y. Ensemble Machine Learning: Methods and Applications. – Springer, 2012. – 352 p.
10. Chollet F. Deep Learning with Python. – Manning Publications, 2021. – 504 p.
11. Hamilton J. Time Series Analysis. – Princeton University Press, 1994. – 800 p.
12. Aggarwal C. Neural Networks and Deep Learning. – Springer, 2018. – 497 p.
13. Bishop C. Pattern Recognition and Machine Learning. – Springer, 2006. – 738 p.
14. Dean J., Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters // Communications of the ACM. – 2008. – Vol. 51, №1. – P. 107-113.
15. Abadi M. et al. TensorFlow: A System for Large-Scale Machine Learning // Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation. – 2016. – P. 265-283.

16. OpenAI. Generative Pre-trained Transformer (GPT): Applications in Natural Language Processing. – 2020.
17. Kalil T. Django 3 By Example: Build Powerful and Reliable Python Web Applications. – Packt Publishing, 2020. – 546 p.
18. Yegulalp S. Advanced Django: Models, Testing, and Deployment. – Independently Published, 2021. – 315 p.
19. Brownlee J. Time Series Forecasting with Python: How to Prepare Data and Develop Models to Predict the Future. – Machine Learning Mastery, 2020. – 556 p.
20. Python Software Foundation. Python Documentation // Online Resource. – Available at: <https://docs.python.org/>.

ДОДАТОК А. ПРЕЗЕНТАЦІЯ



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

1

ДИПЛОМНА РОБОТА
на ступінь вищої освіти магістр
із спеціальності 122 Комп'ютерні науки

ІНТЕРНЕТ МАГАЗИН ОДЯГУ НА ОСНОВІ ВЕБ- ФРЕЙМВОРКУ DJANGO (PYTHON)

Виконав: студент 6 курсу, групи КНДМ-63
Таран Владислав Петрович
Керівник: викладач кафедри Комп'ютерних
наук
д.т.н., професор Котелянець В.В.

Київ - 2025

2

ЗАГАЛЬНА ХАРАКТЕРИСТИКА ДИПЛОМНОЇ РОБОТИ

Тема	<u>Інтернет магазин одягу на основі веб-фреймворку Django (Python)</u>
Мета дослідження	<u>Створення теоретичної моделі інтернет-магазину</u>
Наукове завдання	<u>Розробка теоретичних основ і практичних рекомендацій для створення інтернет-магазину одягу на основі веб-фреймворку Django (Python), який забезпечує високу продуктивність, безпеку та масштабованість.</u>
Об'єкт дослідження	<u>Процес створення інтернет-магазину з використанням веб-фреймворку Django</u>
Предмет дослідження	<u>функціональні можливості інтернет-магазину з використанням веб-фреймворку Django</u>

АНАЛІЗ АКТУАЛЬНОСТІ ТА ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

Створення інтернет-магазинів є актуальним напрямком розвитку бізнесу в умовах цифрової трансформації, яка активно змінює традиційні економічні моделі. Широке поширення інтернету та зростання доступності мобільних пристроїв суттєво вплинуло на зміну поведінки споживачів, зробивши онлайн-шопінг більш зручним та доступним для широких верств населення.

Ринок електронної комерції сьогодні є одним із ключових драйверів глобальної економіки, демонструючи стійке зростання завдяки швидкому поширенню цифрових технологій. Цей сектор характеризується високим рівнем інноваційності, що змінює традиційні моделі бізнесу, оптимізує взаємодію зі споживачами та створює нові можливості для розвитку підприємств різних масштабів. Зростання глобального доступу до високошвидкісного інтернету, масове впровадження мобільних пристроїв та інтеграція багатофункціональних цифрових платформ стали основою для масштабного розвитку онлайн-торгівлі. Такі зміни сприяли активному залученню нових учасників ринку, збільшенню обсягів продажів та формуванню нових стандартів споживання.

Веб-розроблення є однією з найбільш динамічних сфер сучасних технологій, яка постійно трансформується під впливом нових вимог бізнесу, споживачів та технологічних інновацій (Рис. 3.1). Сучасні технологічні тренди визначають не лише напрямки розвитку галузі, але й безпосередньо впливають на ефективність, гнучкість і продуктивність реалізації проектів, зокрема інтернет-магазинів. Ці тренди спрямовані на підвищення якості обслуговування, розширення функціональних можливостей платформ та створення безпечних, зручних і масштабованих рішень. Нижче наведено ключові технологічні напрямки, що формують майбутнє веб-розроблення.

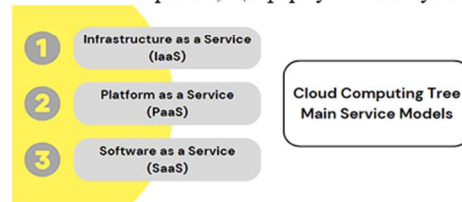


Рисунок 3.1 – Технологічні тренди у веб-розробленні

АРХІТЕКТУРА ВЕБ-ДОДАТКІВ

Архітектура WEB-додатків (Рис. 4.1) є фундаментом для створення сучасних, адаптивних і продуктивних систем, які відповідають зростаючим потребам бізнесу та споживачів. Особливо у випадку інтернет-магазинів, архітектура повинна гарантувати не лише високу продуктивність, безпеку та гнучкість, але й здатність підтримувати безперервну роботу навіть за умов пікових навантажень.

Ефективна архітектура є основою для інтеграції передових технологій, таких як штучний інтелект, автоматизація бізнес-процесів та аналітика даних, що дозволяє створювати конкурентоспроможні рішення. Основні компоненти архітектури веб-додатків можна розділити на декілька рівнів, які забезпечують комплексну роботу системи, взаємодіючи між собою в реальному часі.

- Клієнтський рівень (Frontend)
- Серверний рівень (Backend)
- База даних
- API-інтерфейси
- Системи кешування
- Системи безпеки
- Системи масштабування

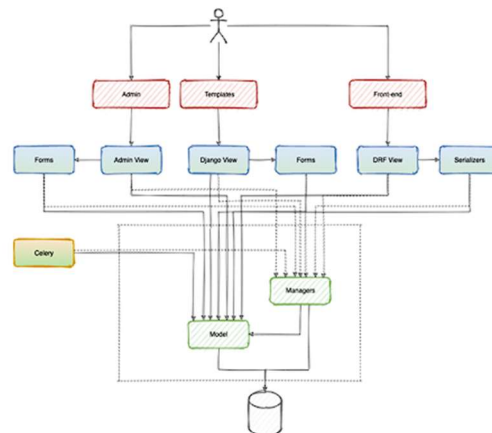


Рисунок 2.1 – Архітектура WEB-додатків

ОСОБЛИВОСТІ ВИКОРИСТАННЯ DJANGO

Основні особливості Django

Модульність і структура MVC (Model-View-Controller)

Django реалізує підхід Model-View-Template (MVT)(Рис. 5.1), який є подібним до MVC. Ця структура дозволяє чітко розділяти бізнес-логіку, представлення даних і інтерфейс користувача. Модульність забезпечує простоту розширення та підтримки проєктів, що особливо важливо для великих інтернет-магазинів із широким функціоналом.

Адміністративна панель

Однією з найбільш корисних функцій Django є автоматично створювана адміністративна панель. Вона дозволяє розробникам і адміністраторам керувати базами даних, змінювати інформацію про товари, замовлення та користувачів без необхідності писати додатковий код.

Інтеграція з базами даних

Django підтримує різноманітні реляційні бази даних, такі як PostgreSQL, MySQL та SQLite. Крім того, ORM (Object-Relational Mapping) дозволяє працювати з базами даних безпосередньо через Python-код, що знижує ризик помилок і спрощує процес розробки

Безпека

Django пропонує вбудовані засоби захисту від поширених атак, таких як SQL-ін'єкції, XSS (міжсайтовий скриптинг) та CSRF (міжсайтові запити підробки). Також забезпечується підтримка шифрування даних, управління доступом користувачів та інших важливих аспектів безпеки

Масштабованість

Django може використовуватися для проєктів будь-якого масштабу. Він підтримує горизонтальне масштабування, що дозволяє обробляти значні навантаження, характерні для великих інтернет-магазинів.

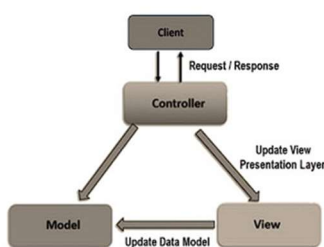


Рисунок 5.1 – Model-View-Template

МОДЕЛЮВАННЯ БАЗИ ДАНИХ У DJANGO

Моделювання бази даних у Django є однією з ключових частин розробки веб-додатків, оскільки саме база даних зберігає інформацію, необхідну для роботи системи. Django забезпечує простий і потужний спосіб роботи з базами даних завдяки своїй системі Object-Relational Mapping (ORM). ORM дозволяє розробникам створювати, змінювати й керувати базами даних за допомогою Python-коду, що значно знижує складність і ризик помилок. Цей підхід сприяє скороченню часу на розробку та забезпечує гнучкість у налаштуванні бази даних відповідно до потреб бізнесу.

Django ORM (Рис.6.1) автоматично генерує SQL-запити, дозволяючи розробникам працювати з даними на високому рівні абстракції.

Наприклад, замість написання складних SQL-запитів для вибірки даних, ORM надає можливість використовувати інтуїтивно зрозумілий Python-синтаксис, що спрощує процес розробки і підтримки коду. Крім автоматизації рутинних завдань, таких як створення та оновлення таблиць, Django ORM підтримує функціонал для роботи з великими обсягами даних, зокрема шардінг та реплікацію. Ці можливості роблять Django ефективним вибором для масштабованих рішень, таких як інтернет-магазини з багатотисячною базою користувачів і високим навантаженням. Інтеграція з інструментами кешування, такими як Redis, дозволяє значно підвищити швидкість обробки запитів і покращити загальну продуктивність додатків.

Основи моделювання бази даних у Django(Рис.6.2)

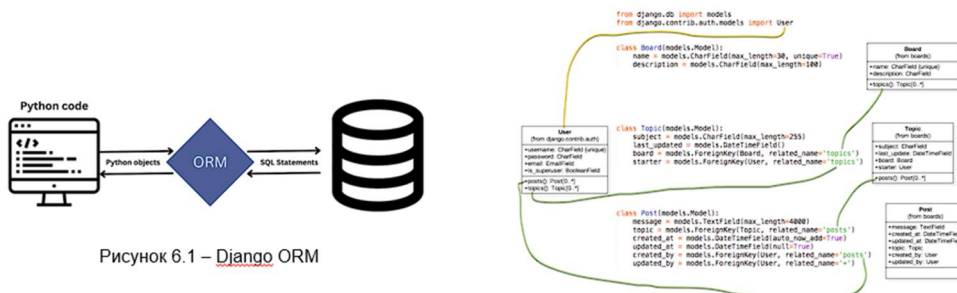


Рисунок 6.1 – Django ORM

Рисунок 6.2 – Моделювання бази даних

СТВОРЕННЯ МОДЕЛЕЙ

Моделі в Django визначають структуру бази даних і є основою для роботи з даними. Кожна модель є класом Python, який відповідає одній таблиці в базі даних. Поля моделі представляють колонки таблиці, а їх типи відповідають типам даних у базі.

```
class Product(models.Model):
    name = models.CharField(max_length=100)
    description = models.TextField()
    price = models.DecimalField(max_digits=10, decimal_places=2)
    stock = models.IntegerField()
    created_at = models.DateTimeField(auto_now_add=True)
```

У цьому прикладі модель `Product` описує таблицю для зберігання інформації про товари.

Зв'язки між моделями:

Django підтримує різні типи зв'язків між таблицями: "один до одного", "один до багатьох" і "багато до багатьох".

```
class Order(models.Model):
    customer = models.ForeignKey('Customer', on_delete=models.CASCADE)
    product = models.ManyToManyField('Product')
    total_price = models.DecimalField(max_digits=10, decimal_places=2)
    created_at = models.DateTimeField(auto_now_add=True)
```

У цьому прикладі таблиця `Order` пов'язана з таблицями `Customer` і `Product` через зовнішні ключі.

Django ORM дозволяє виконувати складні запити до бази даних без написання SQL-коду. Наприклад, отримання всіх товарів із ціною більше 100:

```
expensive_products = Product.objects.filter(price__gt=100)
```

СТВОРЕННЯ МОДЕЛЕЙ

Валідація даних:

Поля моделей у Django підтримують вбудовану валідацію, яка гарантує коректність даних перед їх збереженням у базі. Наприклад, поле `DecimalField` автоматично перевіряє, чи є введені значення числовими. Для більш складної валідації можна використовувати методи `clean()` у моделях, які дозволяють виконувати перевірку залежностей між полями. Наприклад:

```
class Product(models.Model):
    name = models.CharField(max_length=100)
    price = models.DecimalField(max_digits=10, decimal_places=2)
    stock = models.IntegerField()
    def clean(self):
        if self.price < 0:
            raise ValidationError('Ціна не може бути від'ємною.')
        if self.stock < 0:
            raise ValidationError('Кількість на складі не може бути від'ємною.')
```

Такий підхід забезпечує точність і узгодженість даних, які зберігаються в базі.

Моделювання товарів і категорій:

Інтернет-магазини часто вимагають складної структури для зберігання інформації про товари, їх категорії та властивості. Django дозволяє створювати гнучкі моделі, які можна адаптувати до потреб бізнесу.

```
class Category(models.Model):
    name = models.CharField(max_length=100)

class Product(models.Model):
    name = models.CharField(max_length=100)
    category = models.ForeignKey('Category', on_delete=models.CASCADE)
```

Кошик і замовлення:

Моделі для зберігання кошиків і замовлень можуть містити дані про клієнтів, обрані товари, кількість, вартість і статус замовлення. Наприклад, модель для кошика може виглядати так:

```
class Cart(models.Model):
    user = models.ForeignKey('User', on_delete=models.CASCADE)
    product = models.ForeignKey('Product', on_delete=models.CASCADE)
    quantity = models.IntegerField(default=1)
    added_at = models.DateTimeField(auto_now_add=True)

class Order(models.Model):
    user = models.ForeignKey('User', on_delete=models.CASCADE)
    products = models.ManyToManyField('Product', through='OrderItem')
    total_price = models.DecimalField(max_digits=10, decimal_places=2)
    status = models.CharField(max_length=50, choices=[('Pending', 'Pending'), ('Completed', 'Completed'), ('Cancelled', 'Cancelled')])
    created_at = models.DateTimeField(auto_now_add=True)

class OrderItem(models.Model):
    order = models.ForeignKey('Order', on_delete=models.CASCADE)
    product = models.ForeignKey('Product', on_delete=models.CASCADE)
    quantity = models.IntegerField()
```

ОПТИМІЗАЦІЯ КОРИСТУВАЦЬКОГО ДОСВІДУ

Оптимізація користувацького досвіду (UX)(Рис. 10.1) є одним із ключових напрямів покращення існуючих інтернет-магазинів, що безпосередньо впливає на лояльність клієнтів, показники конверсії та загальну ефективність бізнесу. Успішна оптимізація UX повинна базуватися на ретельному аналізі поведінки користувачів, проведенні A/B-тестування для визначення найбільш ефективних рішень, а також впровадженні сучасних технологій, таких як штучний інтелект і машинне навчання.

Одним із важливих елементів є впровадження аналітичних інструментів, таких як Google Analytics(Рис. 10.2) або спеціалізовані UX-аналітики, для моніторингу дій користувачів на сайті. Це дозволяє ідентифікувати вузькі місця, де користувачі стикаються з труднощами, і усувати їх за допомогою оптимізації дизайну чи функціоналу. Крім цього, сучасні тренди, такі як мінімалізм, інклюзивність, адаптивний дизайн і інтерактивний контент, сприяють підвищенню залученості користувачів та їхньої загальної задоволеності. Регулярне тестування впроваджених змін із залученням реальних користувачів є важливим кроком для забезпечення високої якості UX. Результати таких досліджень можуть допомогти не лише покращити функціональність сайту, але й створити більш позитивне враження від взаємодії з брендом.

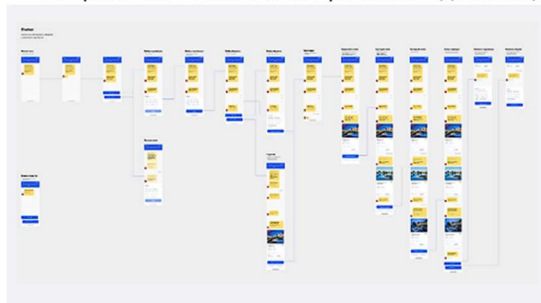


Рисунок 10.1 – Ілюстративна схема оптимізації UX

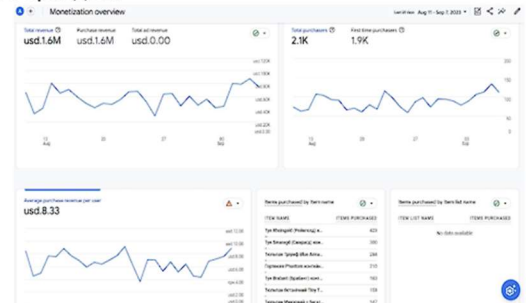


Рисунок 10.2 – Ілюстративна схема оптимізації UX

АЛГОРИТМИ АНАЛІЗУ ДАНИХ У КОМЕРЦІЙНИХ СИСТЕМАХ

11

У комерційних системах дані часто організовані у вигляді багатовимірних масивів або таблиць, де кожен рядок представляє окрему транзакцію, а стовпці містять атрибути, такі як час, вартість, категорія товару тощо. Кожен з цих атрибутів може слугувати основою для аналізу різних аспектів діяльності системи. Наприклад, час транзакції дозволяє визначати пікові періоди продажів, категорія товару — виявляти найбільш популярні продукти, а вартість — аналізувати середній чек клієнта. Організація даних у таких структурах сприяє більш глибокому аналізу за допомогою агрегатних функцій, зокрема обчислення суми продажів за категоріями, середнього значення за період або частки певної категорії у загальному обсязі продажів.

Приклад обробки даних:

```
import pandas as pd

# Завантаження даних

transactions = pd.DataFrame({
    'timestamp': ['2024-12-10 10:00', '2024-12-10 10:05', '2024-12-10 10:10'],
    'amount': [150, 200, 300],
    'category': ['electronics', 'clothing', 'electronics']
})

# Додавання нового стовпця "податок"
transactions['tax'] = transactions['amount'] * 0.2

# Фільтрація за категорією "electronics"
electronics_transactions = transactions[transactions['category'] == 'electronics']

print(electronics_transactions)
```

Для прогнозування комерційних показників, таких як майбутній попит на товари чи обсяги продажів, використовуються моделі регресії.

ПРОГНОЗУВАННЯ ПРОДАЖІВ НА ОСНОВІ СТАТИСТИЧНИХ ДАНИХ

12

Основою прогнозування продажів є математичні моделі, які використовують дані про попередні періоди для створення прогнозів. Одним із найпоширеніших підходів є використання лінійної регресії, яка дозволяє виявляти залежності між змінними, такими як час та обсяги продажів. Цей метод забезпечує базову оцінку трендів, але для складніших завдань, таких як моделювання сезонності або врахування впливу зовнішніх факторів, застосовуються більш просунуті методи. Наприклад, автокореляційні функції (ACF) дозволяють аналізувати зв'язки між елементами часових рядів, що є корисним для визначення періодичних патернів.

Більш точні прогнози можуть бути отримані з використанням алгоритмів ARIMA (AutoRegressive Integrated Moving Average), які враховують як тренди, так і сезонні коливання. Для адаптивних прогнозів, де важливо врахувати багатовимірний вплив, застосовуються ансамблеві алгоритми, такі як Random Forest або XGBoost, що інтегрують багаторівневий аналіз даних та здатні опрацювати великі обсяги інформації з високою точністю.

Математична модель ARIMA:

ARIMA є одним із найпотужніших інструментів для аналізу часових рядів. Модель описується рівнянням:

де — значення ряду в момент часу, — коефіцієнти авторегресії, — коефіцієнти скользячого середнього, — шум.

```
from statsmodels.tsa.arima.model import ARIMA

import pandas as pd

# Імітація даних продажів
data = pd.Series([100, 120, 130, 125, 150, 170, 165])

# Створення моделі ARIMA
model = ARIMA(data, order=(2, 1, 2))

model_fit = model.fit()

# Прогнозування на 3 періоди вперед
forecast = model_fit.forecast(steps=3)

print("Прогнозовані продажі:", forecast)

# Візуалізація результатів
import matplotlib.pyplot as plt

plt.plot(data, label="Історичні дані")

plt.plot(range(len(data), len(data) + len(forecast)), forecast, label="Прогноз")

plt.legend()

plt.show()

# Додаткова інформація про модель
print(model_fit.summary())
```

ВИСНОВКИ

Було розглянуто процес створення інтернет-магазину одягу з використанням веб-фреймворку Django:

1. У межах дослідження проведено аналіз можливостей Django, спроектовано структуру системи та запропоновано рекомендації щодо її вдосконалення.

2. У процесі роботи було детально досліджено актуальність використання інтернет-магазинів у сучасній економіці, зокрема зростаючу роль електронної комерції у глобальних бізнес-процесах. Проведено глибокий аналіз предметної області, виявлено ключові потреби користувачів і бізнесу, а також визначено основні технологічні тренди у веб-розробленні, такі як адаптивний дизайн, масштабованість і безпека.

3. Інтеграція математичних моделей у веб-додаток з використанням Django REST Framework продемонструвала ефективність створення API для взаємодії з математичними обчисленнями. Цей підхід забезпечує доступність системи у реальному часі, її масштабованість і інтеграцію з іншими бізнес-сервісами, такими як платіжні системи, CRM чи аналітичні панелі. .