

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Система прогнозування попиту на основі штучного
інтелекту для оптимізації управління запасами в малому
бізнесі»

на здобуття освітнього ступеня бакалавра
зі спеціальності 122 Комп'ютерні науки
освітньо-професійної програми Штучний інтелект

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

_____ Олександр ЮТУШУЙ

Виконав: здобувач вищої освіти гр.ШІД-41
Олександр ЮТУШУЙ

Керівник: Євген ЧИЧКАРЬОВ
д.т.н, професор

Рецензент:

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут інформаційних технологій

Кафедра Штучного інтелекту

Ступінь вищої освіти Бакалавр

Спеціальність 122 Комп'ютерні науки

Освітньо-професійна програма Штучний інтелект

ЗАТВЕРДЖУЮ

Завідувач кафедру Штучного інтелекту

_____ Ольга ЗІНЧЕНКО

«_____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Ютушу Олександр Олександровичу

1. Тема кваліфікаційної роботи: Система прогнозування попиту на основі штучного інтелекту для оптимізації управління запасами в малому бізнесі

керівник кваліфікаційної роботи д.т.н., професор, Євген ЧИЧКАРЬОВ
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «24» 02.2025р. № 56

2. Строк подання кваліфікаційної роботи «23» травня 2025р.

3. Вихідні дані до кваліфікаційної роботи: Історичні дані продажів у форматі CSV, фактори попиту (ціна, день тижня, сезонність), приклади моделей нейронних мереж (MLP, LSTM), технічна документація Flask, Keras, SQLite, вимоги до візуалізації та генерації PDF-звітів, шаблони управління запасами (EOQ, ROP), результати тестування та графічні звіти.

4. Зміст розрахунково-пояснювальної записки

1. Дослідження сучасних підходів до прогнозування попиту в малому бізнесі

2. Аналіз методів управління запасами та їх обмежень
 3. Вибір архітектури та інструментів для реалізації системи (Flask, Keras, SQLite)
 4. Розробка моделі прогнозування з використанням нейронних мереж
 5. Побудова веб-інтерфейсу для керування товарами і введення даних
 6. Інтеграція прогнозу попиту в систему управління запасами
 7. Генерація графіків та PDF-звітів для аналізу прогнозу
 8. Тестування точності моделі, аналіз результатів та оцінка ефективності системи
5. Перелік графічного матеріалу: *презентація*
6. Дата видачі завдання «25» лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	27.02-05.11.25	Виконано
2	Постановка задачі та вибір методів	05.11-12.11.25	Виконано
3	Реалізація моделі прогнозування попиту	13.11-19.11.25	Виконано
4	Розробка веб-інтерфейсу та бази даних	20.11-25.11.25	Виконано
5	Інтеграція прогнозу з інтерфейсом, тестування системи	27.11-03.12.25	Виконано
6	Генерація звітів та підготовка до демонстрації	04.12-10.12.25	Виконано
7	Оформлення роботи: вступ, висновки, реферат	11.12-20.12.25	Виконано
8	Розробка демонстраційних матеріалів	21.12-31.05.25	Виконано

Здобувач вищої освіти _____

Олександр ЮТУШУЙ

Керівник

кваліфікаційної роботи _____
(підпис)

Євген ЧИЧКАРЬОВ

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: стор. 70, табл. 6, рис. 12, джерел 14.

Мета роботи: автоматизація процесу прогнозування попиту на товари у малому бізнесі шляхом розробки веб-додатку з використанням методів машинного навчання.

Об'єкт дослідження: процес управління товарними запасами в умовах малих підприємств.

Предмет дослідження: веб-додаток, який реалізує прогнозування попиту та формування рішень щодо поповнення запасів із використанням штучного інтелекту.

Короткий зміст роботи: У кваліфікаційній роботі проаналізовано сучасні методи прогнозування попиту та управління товарними запасами в умовах малого бізнесу. Розглянуто підходи до моделювання залежностей між ціною, сезоном, днем тижня та впливом маркетингових кампаній на обсяги продажів. Обґрунтовано доцільність використання нейронних мереж як інструменту прогнозування на основі історичних даних продажів. Розроблено веб-застосунок, що реалізує інтерфейс для керування товарами, введення даних, імпорту CSV-файлів, автоматичного прогнозування попиту та сповіщення про критичний рівень запасів. Візуалізація результатів реалізована за допомогою інтерактивних графіків. Інструментарій системи включає мову програмування Python (Flask), бібліотеки машинного навчання (Keras, scikit-learn), базу даних SQLite, а також модуль генерації PDF-звітів (ReportLab). Було проведено тестування моделі прогнозування, оцінено точність результатів та ефективність прийнятих рішень. Отримані результати підтвердили практичну цінність застосованого підходу в оптимізації запасів та підвищенні ефективності управління в умовах малого бізнесу.

КЛЮЧОВІ СЛОВА: ПОПИТ, ПРОГНОЗУВАННЯ, ШТУЧНИЙ ІНТЕЛЕКТ, ЗАПАСИ, НЕЙРОННА МЕРЕЖА, FLASK, PYTHON, CSV, PDF, KERAS, ВІЗУАЛІЗАЦІЯ, ОПТИМІЗАЦІЯ.

ЗМІСТ

ВСТУП.....	8
1 ТЕОРЕТИЧНІ ОСНОВИ ПРОГНОЗУВАННЯ ПОПИТУ І УПРАВЛІННЯ ЗАПАСАМИ.....	10
1.1 Сучасні підходи до прогнозування попиту в малому бізнесі	10
1.2 Управління товарними запасами: проблеми, методи, моделі.....	177
1.3 Використання методів машинного навчання та нейромереж у логістичних завданнях	233
Висновки до розділу 1	Error! Bookmark not defined. 8
2 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ.....	30
2.1 Характеристика процесів управління запасами в малому бізнесі	30
2.2 Постановка задачі прогнозування попиту та критерії ефективності	34
2.3 Огляд існуючих рішень та обґрунтування вибору інструментів реалізації	399
Висновок до розділу 2.....	499
3 РЕАЛІЗАЦІЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ	51
3.1 Архітектура та компоненти веб-застосунку	51
3.2 Реалізація прогнозування з використанням нейронної мережі	55
3.3 Інтерфейс керування запасами, сповіщення та генерація звітів	599
Висновок до розділу 3.....	64
ВИСНОВКИ	66
ПЕРЕЛІК ПОСИЛАНЬ	69
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (ПРЕЗЕНТАЦІЯ)	70

ВСТУП

У сучасних умовах розвитку малого бізнесу, коли конкуренція, мінливість попиту та логістичні виклики є критичними факторами, здатність точно прогнозувати споживчий попит та ефективно управляти запасами набуває особливої важливості. Традиційні методи ручного обліку або прогнозування на основі інтуїції поступово втрачають актуальність через свою обмежену точність, низьку масштабованість та залежність від людського фактору. Водночас, впровадження інструментів штучного інтелекту (ШІ), зокрема методів машинного навчання, відкриває нові перспективи для аналізу історичних даних і виявлення прихованих закономірностей у поведінці споживачів. Особливо актуальним є створення таких інтелектуальних систем на основі вебтехнологій, що забезпечують простоту доступу, масштабованість і зручну інтеграцію в бізнес-процеси підприємств. Мова програмування Python, зокрема з використанням фреймворку Flask, у поєднанні з бібліотеками машинного навчання (Keras, scikit-learn) дозволяє створити гнучку систему прогнозування, яка буде ефективною навіть у випадку обмеженого набору даних. Така система має потенціал не лише автоматизувати процес оцінки майбутнього попиту, а й покращити управління залишками, оптимізувати витрати на зберігання товарів та мінімізувати втрати від дефіциту або надлишкових запасів.

Актуальність теми зумовлена зростаючою потребою малого бізнесу в доступних, інтелектуальних інструментах, які дозволяють приймати обґрунтовані рішення щодо постачання, закупівель та планування. У добу цифрової трансформації компанії прагнуть автоматизувати свої аналітичні процеси, зменшити залежність від суб'єктивних оцінок і покладатися на дані. Розробка вебсистеми з функціоналом прогнозування попиту, управління товарами, візуалізації результатів та генерації звітів у PDF дозволяє забезпечити комплексну підтримку прийняття рішень.

Мета дослідження: Розробити вебсистему для прогнозування попиту на товари із використанням історичних даних продажів, методів машинного навчання та візуалізації, що дозволяє оптимізувати управління запасами у малому бізнесі.

Завдання дослідження:

- Провести аналіз сучасних підходів до прогнозування попиту та управління запасами;
- Обґрунтувати вибір архітектури та технологій (Flask, Keras, SQLite);
- Реалізувати функціонал додавання товарів, оновлення запасів, імпорту CSV;
- Побудувати модель прогнозування попиту з урахуванням чинників (ціна, день тижня, сезонність);
- Реалізувати графіки прогнозу та генерацію PDF-звітів;
- Провести тестування системи та оцінити її точність і зручність використання.

Об'єкт дослідження: Процес аналізу історичних даних продажів для прогнозування майбутнього попиту на товари.

Предмет дослідження:

Інтелектуальна система прогнозування попиту, що реалізована у вигляді вебзастосунку та використовує нейронні мережі для автоматичного формування попиту і прийняття рішень щодо управління запасами.

Методи дослідження: Аналіз літературних джерел, статистичні методи прогнозування, машинне навчання (нейронні мережі), візуалізація даних (matplotlib), програмування на Python, розробка вебінтерфейсу (HTML/CSS/JS/PHP), генерація PDF (ReportLab). Розроблена система може бути впроваджена у малих торгових підприємствах, магазинах та сервісах для покращення планування постачання, скорочення надлишкових запасів і зменшення втрат від дефіциту. Система також має потенціал застосування в освітньому середовищі для демонстрації застосування AI у бізнес-аналітиці, а також для подальшої інтеграції у CRM-системи, платформи електронної комерції та внутрішні управлінські панелі малого бізнесу.

1 ТЕОРЕТИЧНІ ОСНОВИ ПРОГНОЗУВАННЯ ПОПИТУ І УПРАВЛІННЯ ЗАПАСАМИ

1.1 Сучасні підходи до прогнозування попиту в малому бізнесі

Прогнозування попиту є фундаментальним елементом в управлінні ланцюгами постачання, оскільки воно визначає, скільки товару потрібно виробити, закупити або доставити, щоб задовольнити майбутні потреби споживачів. У контексті малого бізнесу цей процес набуває критичної ролі через обмеженість ресурсів, фінансову вразливість і високу залежність від точності управлінських рішень. На відміну від великих корпорацій, малі підприємства не мають змоги утримувати великі запаси або формувати значні резерви. Будь-яка похибка в прогнозі може призвести до втрати клієнтів (у випадку дефіциту) або до заморожування оборотних коштів (у випадку надлишку). Крім того, малі підприємства часто мають вузький асортимент, обмежену географію постачання і чутливу ціну закупки, що підвищує вагу кожної управлінської помилки. Сучасні методи прогнозування дають змогу малому бізнесу вийти за межі інтуїтивних оцінок і перейти до науково обґрунтованого аналізу. Вони дозволяють враховувати сезонність, маркетингові впливи, поведінкові особливості споживачів, а також тренди, що змінюються у часі. Застосування аналітики на основі історичних даних дає змогу адаптуватися до мінливого ринку, мінімізувати витрати і підвищити рентабельність операційної діяльності.

Існує декілька основних підходів до прогнозування попиту, які використовуються у практиці малого бізнесу:

1. Статистичні методи прогнозування

Статистичні методи є одними з найдавніших і найширше використовуваних у сфері прогнозування попиту. Вони базуються на обробці й аналізі наявних історичних даних про продажі з метою виявлення регулярностей, трендів, сезонних та циклічних коливань.

Найбільш поширені статистичні методи включають:

Методи ковзного середнього — дозволяють згладити випадкові коливання в даних, визначаючи середнє значення за певний період. Вони широко використовуються у роздрібній торгівлі, де сезонність або циклічність мають значний вплив на обсяги продажів.

Експоненційне згладжування — розширює метод ковзного середнього за рахунок надання більшої ваги останнім значенням ряду. Розрізняють просте, подвійне та трійне експоненційне згладжування, кожне з яких застосовується для даних з відповідними характеристиками: стабільністю, трендом або сезонністю.

Регресійні моделі — передбачають побудову математичної функції, яка описує залежність між попитом (залежною змінною) і факторами, що його визначають (незалежними змінними). Прикладами є лінійна та поліноміальна регресія. Їх використовують для оцінки впливу таких факторів, як ціна, маркетингова активність, конкуренція тощо.

Авторегресійні моделі (AR, ARMA, ARIMA, SARIMA) — використовуються для моделювання часових рядів і дозволяють робити прогнози на основі минулих значень ряду та похибок моделі. ARIMA є однією з найефективніших моделей для даних без чітко вираженої сезонності, тоді як SARIMA розширює її можливості для врахування сезонних компонентів.

Перевагами статистичних методів є:

- відносна простота у реалізації та розумінні;
- наявність великої кількості перевірених моделей та інструментів для їх реалізації (наприклад, у Python, R, Excel);
- можливість побудови базових моделей прогнозування без потреби у великих обсягах даних.

Однак ці методи також мають ряд обмежень:

- вони часто не враховують зовнішні фактори, що впливають на попит (наприклад, зміни споживчих вподобань, конкуренція);
- не завжди ефективно працюють із нелінійними або високовимірними даними;

- менш стійкі до аномалій, шумів або раптових змін тренду.

Попри ці недоліки, статистичні методи залишаються важливим інструментом у практиці малого бізнесу, особливо як стартовий етап до впровадження більш складних моделей, таких як методи машинного навчання.

2. Експертні методи

Експертні методи прогнозування попиту базуються на інтуїції, знаннях та досвіді осіб, які глибоко обізнані з ринковим середовищем, товаром, клієнтською базою або специфікою галузі. На відміну від статистичних та математичних моделей, ці методи не потребують великої кількості кількісних даних, а натомість покладаються на якісні оцінки, припущення і суб'єктивну думку фахівців.

До найпоширеніших експертних методів належать:

Метод Делфі — ітеративний процес збору експертних оцінок шляхом кількох раундів анкетування, в ході яких експерти анонімно висловлюють думки, після чого отримують узагальнені результати попереднього раунду й можуть переглянути свої оцінки. Це сприяє досягненню більш узгодженого прогнозу без впливу авторитету чи домінування окремих осіб.

Метод сценаріїв — передбачає розробку кількох можливих сценаріїв розвитку ситуації (наприклад, оптимістичний, песимістичний, базовий) з подальшою оцінкою попиту в кожному з них. Цей підхід дозволяє враховувати невизначеність майбутнього та можливі ризики.

Анкетування та фокус-групи — використовується для збору думок клієнтів, партнерів, працівників або інших зацікавлених осіб щодо очікувань, уподобань або поведінки у майбутньому. Такий метод часто застосовується при виведенні нового продукту на ринок або під час швидкої зміни споживчих трендів.

Оцінка продажів торговим персоналом — базується на досвіді менеджерів із продажу, які найкраще знають своїх клієнтів і можуть інтуїтивно передбачити попит на найближчий період.

Переваги експертних методів:

- швидка реалізація в умовах обмежених або відсутніх історичних даних;
- можливість врахування унікальних контекстних факторів (наприклад, політична ситуація, поведінкові моделі споживачів);
- гнучкість і доступність у використанні для малого бізнесу.

Однак, незважаючи на ці переваги, експертні методи мають значні недоліки:

- високий рівень суб'єктивності — прогнози залежать від особистого досвіду, інтуїції та переконань конкретної особи;
- обмежена повторюваність і перевірюваність — ускладнене відтворення результатів у інших умовах або іншим колективом експертів;
- складність масштабування — у разі розширення бізнесу або асортименту стає важко застосовувати метод вручну;
- непридатність до автоматизації — такі прогнози важко інтегрувати у цифрові системи управління запасами або планування.

Зважаючи на це, експертні методи доцільно використовувати як допоміжний інструмент — наприклад, на ранніх етапах, при обмеженості даних, або для перевірки результатів автоматизованих моделей. У поєднанні з кількісними підходами вони можуть посилити надійність системи прогнозування й забезпечити цінну якісну інформацію для прийняття управлінських рішень.

3. Причинно-наслідкові моделі

Причинно-наслідкові моделі прогнозування попиту є важливим інструментом, що дозволяє враховувати взаємозв'язки між попитом і різноманітними зовнішніми факторами, які безпосередньо чи опосередковано впливають на поведінку споживачів. У цих моделях попит розглядається як залежна змінна, яка формується під впливом незалежних змінних, що можуть бути кількісно виміряні та проаналізовані.

До ключових чинників, які можуть враховуватися у таких моделях, належать:

- ціна товару (власна та конкурентів);
- рівень доходів населення (економічна доступність);
- погодні умови (особливо актуальні для сезонних товарів);
- рівень рекламної активності (витрати на маркетинг);
- день тижня або місяця (поведінкові патерни покупців);
- розташування торгової точки або зона доставки;
- попередні дії клієнтів (наприклад, перегляди, замовлення, покинуті кошики).

Однією з найбільш вживаних форм причинно-наслідкового моделювання є множинна лінійна регресія, яка дозволяє вивчити, яким чином декілька незалежних змінних одночасно впливають на попит.

Переваги причинно-наслідкових моделей:

- дозволяють виявити кількісні залежності між факторами та обсягом попиту;
- зручні для інтерпретації, що важливо для малого бізнесу;
- можуть бути використані для сценарного аналізу (як зміниться попит при зміні ціни, реклами тощо);
- застосовні навіть при обмеженому обсязі історичних даних.

Недоліки:

- передбачають лінійність зв'язків, що не завжди відповідає реальності;
- потребують коректної підготовки даних (нормалізація, виключення мультиколінеарності);
- не враховують складних нелінійних взаємодій, які можуть бути краще змодельовані за допомогою машинного навчання;
- чутливі до викидів і пропущених значень у даних.

Для підвищення точності, у сучасних системах такі моделі можуть доповнюватися іншими інструментами — наприклад, регресією з регуляризацією (Ridge, Lasso), логістичною регресією для класифікації поведінки споживача (купить/не купить), а також гібридними моделями, що поєднують регресію з

інтелектуальним аналізом даних. Причинно-наслідкові моделі виступають проміжним рівнем між простими статистичними підходами та складними інтелектуальними системами, дозволяючи бізнесу не лише прогнозувати попит, а й аналізувати причини його змін.

4. Методи машинного навчання

Це сучасний напрям прогнозування, який передбачає використання алгоритмів штучного інтелекту для побудови моделей, що навчаються на історичних даних. Основні переваги:

- здатність до виявлення складних патернів і взаємозв'язків;
- підтримка великої кількості змінних;
- висока точність за наявності достатнього обсягу якісних даних.

До популярних алгоритмів ML для прогнозування попиту належать:

- дерева рішень (Decision Tree, Random Forest);
- метод опорних векторів (SVM);
- градієнтний бустинг (XGBoost, LightGBM);
- нейронні мережі (MLP, LSTM, GRU).

В умовах малого бізнесу особливу цінність мають простота інтеграції, автоматизованість та адаптивність таких моделей до нових даних. Інструменти з відкритим кодом (наприклад, scikit-learn, Keras, TensorFlow) дозволяють реалізовувати повноцінні інтелектуальні рішення без значних фінансових витрат.

5. Гібридні моделі

Гібридні моделі прогнозування попиту є інноваційним підходом, який поєднує переваги кількох методів одночасно, дозволяючи підвищити точність, адаптивність і стійкість до змін середовища. Такий підхід базується на комбінуванні різних технік, наприклад: статистичного аналізу з алгоритмами машинного навчання, або ж експертних оцінок у поєднанні з регресійними або класифікаційними моделями.

Мета гібридизації — скористатися сильними сторонами кожного з методів та компенсувати їхні обмеження. Наприклад, статистичні моделі добре описують загальні тренди, але погано справляються з аномаліями, у той час як нейронні мережі здатні вловити складні нелінійні зв'язки, проте потребують великої кількості даних і складного навчання. У гібридному підході ці інструменти можуть працювати разом: статистика — для базового прогнозу, AI — для коригування й адаптації.

Поширеними типами гібридних моделей є:

Stacked Models — коли результати кількох моделей (наприклад, ARIMA, XGBoost, LSTM) подаються як вхідні дані до мета-моделі, яка формує фінальний прогноз.

Rule-based + ML — коли експертна система (набір правил) визначає контекст або граничні значення, а машинне навчання виконує точковий прогноз.

Weighted Hybrid — коли кілька моделей працюють паралельно, а результат — це зважене середнє прогнозів із різними вагами в залежності від ситуації (наприклад, сезон, обсяг продажів, відсутність даних).

Переваги гібридних підходів:

- гнучкість та універсальність застосування;
- висока точність і стійкість до шуму;
- здатність враховувати як кількісні, так і якісні фактори;
- можливість адаптації під різні категорії товарів або регіони.

Складність реалізації гібридних моделей полягає у потребі глибокого розуміння як статистичних, так і інтелектуальних алгоритмів, а також в необхідності обробки великих обсягів даних, калібрування моделей та об'єднання результатів. В умовах малого бізнесу такі моделі можуть бути реалізовані в спрощеному вигляді — наприклад, з використанням базових регресійних моделей у поєднанні з ручними правилами або простими деревами рішень. При цьому, навіть така спрощена гібридна система може забезпечити значно кращу продуктивність у порівнянні з ізольованими підходами.

Сучасні гібридні системи прогнозування попиту є перспективним напрямом для малого бізнесу, адже вони дозволяють поєднати точність і надійність прогнозів з адаптивністю до швидкозмінного середовища. Інтегровані веборієнтовані системи, які комбінують машинне навчання, статистику, візуалізацію та зручний інтерфейс, стають доступним і потужним інструментом для прийняття рішень у сфері управління запасами.

1.2 Управління товарними запасами: проблеми, методи, моделі

Управління запасами є ключовою складовою логістики та операційної діяльності підприємств, оскільки безпосередньо впливає на здатність бізнесу забезпечувати безперебійне обслуговування клієнтів, мінімізувати витрати та уникати втрат від надлишкових або недостатніх залишків продукції. Ефективна система управління запасами дозволяє забезпечити стабільність постачання, уникати ситуацій дефіциту, скоротити витрати на зберігання і транспортування, а також сприяє підвищенню рівня обслуговування клієнтів. Особливої ваги ефективне управління товарними запасами набуває в умовах малого бізнесу, де обмеженість ресурсів (таких як складські площі, обігові кошти, трудові ресурси) посилює наслідки будь-якої помилки у плануванні. Малі підприємства не мають змоги утримувати надмірні товарні залишки або оперативно реагувати на збої у ланцюгу постачання, що робить управління запасами не просто важливою, а критичною функцією. Наприклад, невірно визначений обсяг замовлення або несвоєчасне поповнення запасів можуть спричинити зниження рівня обслуговування, втрату лояльних клієнтів та репутаційні ризики. З іншого боку, надмірне накопичення товару призводить до замороження фінансових ресурсів, збільшення витрат на зберігання, необхідності списання застарілих або зіпсованих товарів, а також знижує обіговість капіталу.

В умовах нестабільності ринку та швидких змін попиту, малий бізнес стикається з підвищеною невизначеністю, що ще більше ускладнює процес планування. Часто компанії не мають повноцінних автоматизованих систем управління запасами, що змушує покладатися на ручні розрахунки, які можуть бути

неточними або надто суб'єктивними. У зв'язку з цим, використання сучасних інструментів прогнозування попиту і автоматизованих моделей управління запасами відкриває для малого бізнесу нові можливості для підвищення ефективності, гнучкості та конкурентоспроможності.

Основні проблеми в управлінні запасами:

Непередбачуваність попиту — один із головних викликів у системі управління запасами. Часто споживчий попит є нерегулярним, залежить від зовнішніх факторів (економіка, реклама, конкуренція) та піддається випадковим коливанням. Без точного прогнозу компанії змушені або перестраховуватися, створюючи надмірні запаси, або ризикувати нестачею товару, що негативно позначається на рівні обслуговування клієнтів.

Сезонність та коливання ринку — для багатьох галузей характерна сильна сезонна залежність попиту (наприклад, товари для відпочинку, шкільні приладдя, новорічні декорації). Компанії повинні враховувати ці циклічні зміни в плануванні, інакше ризикують втратити прибуток або стикнутися з надлишками. Також важливо передбачати ефект маркетингових кампаній, розпродажів і свят, що викликають тимчасові піки в попиті.

Обмеженість складських ресурсів — підприємства, особливо малі, мають обмежений фізичний простір для зберігання продукції. Збільшення обсягів запасів автоматично призводить до зростання логістичних витрат, необхідності оренди додаткових площ або закупівлі складського обладнання. Часто виникає потреба у зонуванні складу за температурними або гігієнічними умовами, що ще більше ускладнює управління.

Час виконання замовлення (Lead Time) — інтервал між оформленням замовлення і фактичним отриманням товару. Затримки в постачанні або зміни в графіку логістичних компаній можуть спричинити перебої у формуванні запасів. В умовах глобалізованих поставок (особливо з інших країн) навіть незначне порушення термінів може призвести до втрати ринку або розриву контрактів.

Недостатня автоматизація — у багатьох малих підприємствах відсутні сучасні IT-рішення для моніторингу, обліку та прогнозування запасів. Ручне

введення даних, використання Excel або паперових журналів сприяє зростанню кількості помилок, уповільнює аналітичні процеси, не дозволяє оперативно реагувати на зміни попиту. Відсутність автоматичних нагадувань, звітів і динамічних графіків перешкоджає прийняттю стратегічних управлінських рішень.

Усі ці проблеми в сукупності формують потребу у впровадженні сучасних цифрових інструментів прогнозування, автоматизації обліку та побудови адаптивної системи управління запасами, що здатна швидко реагувати на зміни ринкової кон'юнктури та внутрішні операційні ризики.

Методи управління запасами:

Модель EOQ (Economic Order Quantity) — це одна з найбільш класичних моделей, яка визначає оптимальний розмір замовлення з урахуванням витрат на зберігання та витрат на розміщення замовлень. Основна ідея полягає у тому, щоб знайти такий обсяг постачання, при якому сумарні витрати будуть мінімальними. Модель базується на припущенні сталого попиту, постійного часу поставки та відсутності знижок. Формула EOQ виглядає так: $EOQ = \sqrt{(2DS)/H}$, де D — річний попит, S — вартість одного замовлення, H — вартість зберігання одиниці товару. Для малого бізнесу це дозволяє уникнути як перевантаження складу, так і частих замовлень невеликих партій.

Модель ABC/XYZ — це метод класифікації товарів за важливістю (ABC) та стабільністю споживання (XYZ). У методі ABC товари поділяються на три категорії: A — найбільш вартісні і критичні, B — середньої значущості, C — з низькою вартістю або низьким обігом. У методі XYZ товари оцінюються за ступенем передбачуваності попиту: X — високопрогнозовані, Y — з коливаннями, Z — з нерегулярним або випадковим попитом. Поєднання ABC та XYZ дозволяє сформувати стратегії управління запасами для кожної групи: наприклад, A-X товари потребують точної аналітики та контролю, тоді як C-Z — можуть замовлятися за фактом.

Система "точка замовлення" (Reorder Point, ROP) — визначає момент, коли потрібно оформити нове замовлення для поповнення запасів до того, як вони досягнуть нульового рівня. Точка замовлення обчислюється з урахуванням

середнього попиту та часу виконання замовлення: $ROP = d \times L$, де d — середній денний попит, L — час виконання замовлення. У більш розширених варіантах враховується також "страховий запас" (safety stock), який запобігає дефіциту у випадку непередбачуваного зростання попиту чи затримки постачання.

JIT (Just-In-Time) — це метод управління запасами, при якому товари замовляються і поставляються в момент їх фактичної потреби. Такий підхід дозволяє звести до мінімуму складські запаси, зменшити витрати на зберігання та знизити ризики застарівання продукції. Однак JIT вимагає високої точності в прогнозуванні, тісної співпраці з постачальниками та стійких логістичних ланцюгів. Для малого бізнесу ця модель може працювати при стабільному попиті й надійних постачальниках, але є ризик порушення у разі затримок.

Модель безперервного перегляду (Continuous Review) — передбачає постійне відстеження рівня запасів і автоматичне створення замовлення, щойно кількість товару досягає порогового значення. На відміну від періодичного перегляду (наприклад, раз на тиждень), цей підхід дозволяє оперативно реагувати на зміну попиту. В системах з інтеграцією зі складським обліком або POS (point-of-sale) він може забезпечити майже реальний контроль і уникнути дефіциту чи перевантаження.

Ці методи є основою для побудови сучасних систем управління запасами, які можуть бути реалізовані в електронних системах (ERP, WMS) або через кастомні рішення, зокрема у вигляді вебзастосунків з підтримкою машинного прогнозування попиту, що значно підвищує точність і швидкість ухвалення рішень у малому бізнесі.

Моделі прогнозування в управлінні запасами:

Сучасні підходи до управління запасами вимагають інтеграції аналітичних інструментів і алгоритмів прогнозування попиту безпосередньо в системи управління ланцюгами постачання. Завдяки цьому бізнес отримує можливість не лише реагувати на зміни, а й діяти на випередження, адаптуючи свої логістичні та закупівельні стратегії відповідно до очікуваної динаміки попиту.

Одним із прикладів такої інтеграції є:

Поєднання ML-прогнозу з точкою замовлення (ROP) — система аналізує історичні дані продажів, виявляє сезонні патерни, враховує маркетингові кампанії або події, після чого формує прогноз попиту на майбутній період. Згідно з цим прогнозом автоматично визначається нова точка замовлення, що дозволяє уникнути як дефіциту, так і перевищення залишків. У цьому випадку прогноз використовується як динамічна змінна, що коригується щотижня або щодня.

AI-оптимізація запасу під ресурсні обмеження — за допомогою алгоритмів штучного інтелекту відбувається багатофакторна оптимізація: система одночасно враховує бюджет на закупівлю, доступну площу складу, обмеження на термін придатності продукції, сезонні тенденції та наявність товарних замінників. Оптимізація відбувається шляхом побудови моделей, які мінімізують витрати і втрати продажів при обмежених ресурсах.

Інтеграція з CRM та ERP-системами — дозволяє використовувати дані в реальному часі: історія продажів, активність клієнтів, рентабельність товарів, знижки та залишки — усе це автоматично передається в систему прогнозування. На основі цих даних модель формує адаптивний прогноз попиту, який одразу ж впроваджується в модуль управління запасами для розрахунку потреби та формування замовлення.

Такі моделі прогнозування не лише забезпечують вчасне поповнення складу, а й підвищують ефективність планування закупівель, сприяють зменшенню втрат через зниження дефіциту або надлишків, підвищують оборотність запасів і дозволяють приймати обґрунтовані стратегічні рішення.

Практичне значення для малого бізнесу:

Ефективне управління товарними запасами відіграє вирішальну роль у стабільному функціонуванні малого підприємства, дозволяючи не лише підвищити загальну ефективність операцій, а й зміцнити конкурентні позиції на ринку. Системний підхід до управління залишками забезпечує низку важливих переваг:

Скорочення витрат на зберігання — за рахунок оптимізації обсягів замовлень і зменшення надлишкових товарних запасів знижується потреба в оренді або утриманні великих складських площ, зменшуються витрати на охорону, кондиціонування, страхування та утримання персоналу складу.

Зниження обігового капіталу, закріпленого у залишках — кошти, які раніше були «заморожені» у вигляді неходових товарів, можуть бути спрямовані на інші напрямки: інвестиції в розвиток, маркетинг, підвищення заробітних плат або розширення асортименту.

Підвищення рівня обслуговування клієнтів — системне поповнення запасів і точне прогнозування попиту дозволяє гарантувати постійну наявність найпопулярніших товарів. Це сприяє формуванню лояльності серед клієнтів, зменшенню кількості втрачених замовлень та підвищенню повторних покупок.

Автоматизація процесів і підвищення оперативності — використання ІТ-систем і алгоритмів прогнозування забезпечує швидке оновлення даних, генерацію звітів і аналітики, а також зменшує ймовірність людської помилки. Менеджмент може оперативно приймати рішення щодо формування замовлень, коригування обсягів постачання чи зміни цінової політики.

Гнучкість у реагуванні на зміни ринку — адаптивна система дозволяє миттєво реагувати на сезонні коливання, зміну попиту або логістичні збої. Це зменшує ризики втрат і підвищує життєздатність бізнесу навіть в умовах кризи або нестабільності.

У сукупності всі ці чинники забезпечують не лише економію ресурсів, а й створюють додану вартість для кінцевого споживача, формуючи репутацію підприємства як надійного постачальника. Таким чином, грамотне управління запасами є не просто логістичною функцією, а стратегічним елементом конкурентної переваги малого бізнесу.

Сучасне управління товарними запасами потребує не лише базових логістичних операцій, а повноцінного стратегічного підходу, що включає глибоку аналітику, інтеграцію прогнозних моделей, автоматизацію рутинних процесів,

моніторинг у реальному часі та впровадження інструментів підтримки прийняття рішень.

Такий підхід дозволяє підприємству стати адаптивним до ринкових коливань, ефективно реагувати на нестабільний попит, знижувати вплив людського чинника та досягати синергії між різними функціями бізнесу — закупівлею, логістикою, фінансами, маркетингом. У цифрову епоху управління запасами перетворюється на складний багатовимірний процес, де використовуються дані з різних джерел: CRM, ERP, онлайн-продажів, аналітики ринку, що дозволяє забезпечити максимальну точність, знизити операційні витрати та підвищити конкурентоспроможність підприємства на локальному та глобальному ринках.

1.3 Використання методів машинного навчання та нейромереж у логістичних завданнях

Машинне навчання (ML) і штучні нейронні мережі (ШНМ) дедалі ширше застосовуються в галузі логістики, оскільки вони забезпечують принципово новий рівень точності, ефективності та адаптивності у прийнятті рішень. Зокрема, ці технології активно впроваджуються у завдання прогнозування попиту, управління запасами, маршрутизації постачання, а також у сфери обслуговування клієнтів, оцінки ризиків і виявлення нестандартних ситуацій. Їхня ключова перевага полягає в здатності обробляти великі масиви різнотипних даних, виявляти закономірності, які не очевидні для людини, та адаптуватися до нових сценаріїв без потреби ручного втручання.

Застосування ML і ШНМ у логістиці стало відповіддю на зростання складності сучасних ланцюгів постачання, які охоплюють багатоетапні процеси, численні джерела ризику, постійну змінність попиту та необхідність оперативного реагування на зміни в ринковому середовищі.

Завдяки гнучкості моделей машинного навчання підприємства мають змогу не лише знижувати витрати, а й підвищувати рівень обслуговування, забезпечуючи точне й своєчасне поповнення запасів, оптимальні графіки доставки та прогнозування вузьких місць у ланцюгах постачання. У результаті, ML і ШНМ

стають критичними компонентами цифрової трансформації логістичних систем і одним із головних рушіїв інновацій у сфері розумного управління запасами та транспортними потоками.

Основні напрями застосування ML і ШНМ у логістиці:

Прогнозування попиту — машинне навчання дозволяє будувати точніші прогнози на основі великої кількості факторів: історії продажів, сезонності, цінової політики, маркетингових кампаній, зовнішніх подій (погода, інфляція). Моделі, такі як лінійна регресія, дерева рішень, нейронні мережі (MLP, LSTM), забезпечують високу точність завдяки виявленню складних взаємозв'язків між змінними.

Управління запасами — ML-моделі можуть динамічно визначати оптимальний обсяг замовлення, точки поповнення, рівень страхового запасу залежно від змін попиту. Вони враховують історичні тренди, поточні продажі та поведінку клієнтів, що дозволяє скоротити втрати через дефіцит або надлишок товару.

Оптимізація постачання і маршрутів — за допомогою алгоритмів кластеризації та нейромереж логістичні компанії оптимізують доставку, враховуючи фактори трафіку, відстані, часові вікна.

Алгоритми Reinforcement Learning дозволяють самонавчальним системам оптимізувати рішення на основі зворотного зв'язку в реальному часі.

Виявлення аномалій та шахрайства — моделі класифікації й автоенкодери використовуються для виявлення підозрілих транзакцій, збоїв у ланцюгах постачання, нестандартної поведінки клієнтів або партнерів.

Це важливо для попередження ризиків та забезпечення прозорості в логістичних процесах.

Планування персоналу та ресурсів — з урахуванням прогнозів попиту та обсягів поставок системи ML допомагають визначити оптимальну кількість працівників, змін, транспорту чи складських ресурсів, необхідних у певний період.

Типи нейронних мереж, що використовуються в логістиці:

MLP (Multi-layer Perceptron) — це одна з найпростіших, але водночас універсальних архітектур штучних нейронних мереж. Вона складається з кількох шарів нейронів (вхідного, одного або декількох прихованих та вихідного), де кожен нейрон з'єднаний з усіма нейронами наступного шару. MLP добре підходить для вирішення завдань регресії (наприклад, передбачення обсягу продажів) або класифікації (наприклад, визначення, чи буде товар дефіцитним).

Завдяки своїй простоті і гнучкості, MLP активно використовується у логістичних системах, що мають обмежену обчислювальну потужність, зокрема у малому бізнесі.

LSTM (Long Short-Term Memory) — спеціалізована архітектура рекурентних нейронних мереж (RNN), яка розроблена для роботи з послідовними даними та часовими рядами.

Вона дозволяє враховувати попередні значення, що особливо актуально для прогнозування попиту на основі щоденної, тижневої або місячної історії продажів. LSTM здатна утримувати довгострокові залежності між подіями та краще справляється з трендами та сезонними патернами у логістичних даних.

Її ефективність підтверджена в системах автоматичного формування замовлень, передбачення затримок постачання, адаптивного управління запасами.

CNN (Convolutional Neural Networks) — хоч традиційно застосовуються для обробки зображень, у логістиці вони використовуються в задачах розпізнавання візуального стану товару (наприклад, пошкодження упаковки), автоматичного зчитування маркування, QR-кодів або штрих-кодів, а також для аналізу відеопотоків з камер складу для моніторингу роботи персоналу, контролю заповненості полиць, виявлення порушень у зберіганні продукції.

CNN дозволяють значно підвищити точність візуального контролю та автоматизувати багато рутинних перевірок.

Практичне впровадження для малого бізнесу:

У сучасному цифровому середовищі навіть малі підприємства можуть інтегрувати інструменти машинного навчання завдяки широкому спектру

відкритих бібліотек (наприклад, scikit-learn, Keras, TensorFlow) та доступу до хмарних обчислювальних ресурсів (Google Colab, AWS, Azure). Це дозволяє створювати повноцінні аналітичні рішення без необхідності купувати дорогі сервери або наймати великий штат програмістів. Наприклад, на основі Python можна реалізувати мікросервіс на Flask чи Django, який щодня обробляє CSV-файл із даними продажів, генерує прогноз попиту за допомогою навченої моделі й формує автоматизовані рекомендації щодо обсягів закупівель для кожного товару.

Такі рішення можуть бути інтегровані у вебінтерфейс підприємства, дозволяючи керівникам бачити прогнози у вигляді графіків або таблиць, з можливістю експорту у PDF або Excel. Для компаній, які не мають власної ІТ-команди, можливе використання готових шаблонів із відкритих репозиторіїв, адаптація під власні потреби й поступове навчання співробітників основам роботи з даними.

Крім того, з'являються спеціалізовані low-code платформи (як-от Google AutoML чи DataRobot), які дають змогу запускати моделі без глибоких знань програмування.

Переваги впровадження нейромереж у логістиці:

- Автоматизація процесів прийняття рішень — замість ручного аналізу Excel-файлів, система самостійно розраховує обсяг поповнення товарів, пріоритизує позиції та повідомляє про ризики дефіциту;
- Зниження помилок у прогнозах — нейромережі краще враховують сезонність, тренди та сплески попиту, надаючи точніші прогнози у порівнянні зі стандартними статистичними підходами;
- Гнучкість і адаптивність до змін ринку — моделі можуть перенавчатися на нових даних, адаптуючись до змін у поведінці споживачів або макроекономічних умов;
- Покращення обслуговування клієнтів — наявність прогнозованого запасу мінімізує ймовірність ситуацій, коли необхідного товару немає в наявності;

- Скорочення витрат на зберігання, транспорт і закупівлі — точне планування дає змогу зменшити надлишкові замовлення, уникати штрафів за прострочку або затримки доставки.

Виклики:

- Потреба у якісних і структурованих даних — без чистих і повних записів про продажі, залишки, дати постачань, модель не зможе коректно навчитися;
- Складність вибору та налаштування моделі — для кожного типу товару або бізнесу підходить різна архітектура, тому потрібне експериментування;
- Потреба в технічній експертизі — навіть із low-code рішень часто потрібен фахівець, який розуміє логіку моделі, може її адаптувати, оновити або проаналізувати її похибку;
- Пояснюваність результатів моделей — деякі складні моделі працюють як «чорні скриньки», що ускладнює пояснення керівництву або партнерам, чому саме запропоновано ті чи інші прогнози.

Незважаючи на виклики, використання нейромереж у логістиці є стратегічно виправданим навіть для невеликих компаній. Це не лише інструмент автоматизації, а й потужний спосіб підвищення конкурентоспроможності, оскільки дозволяє швидше адаптуватися до ринку, приймати обґрунтовані рішення та зменшити витрати без втрати якості.

Загалом, використання машинного навчання (ML) і нейронних мереж у логістиці вже давно перестало бути просто технологічним трендом — сьогодні це практичний, перевірений інструмент досягнення реальної конкурентної переваги. Ці технології дозволяють не лише автоматизувати рутинні процеси, але й робити логістику більш гнучкою, адаптивною до зовнішніх коливань, а головне — проактивною. Сучасні логістичні платформи на базі ML можуть не лише аналізувати дані, але й передбачати події, оптимізувати маршрути, прогнозувати навантаження на склади та мінімізувати ризики простоїв або затримок. Особливо значущим є впровадження таких рішень у малому бізнесі, який часто обмежений у ресурсах, але водночас потребує оперативності й точності. Завдяки відкритим технологіям, як-от безкоштовні фреймворки, хмарні сервіси та low-code

платформи, ці інструменти стають доступними навіть для компаній без великого ІТ-відділу. Малі підприємства отримують змогу впроваджувати персоналізовані системи прогнозування, зменшувати залишки на складі, швидше реагувати на поведінку клієнтів і динаміку попиту.

Машинне навчання у логістиці трансформує спосіб, у який підприємства приймають рішення: від інтуїтивного — до даних, від реактивного — до прогнозного, від ручного — до автоматизованого. Це не лише підвищує ефективність бізнесу, а й сприяє його сталому розвитку в умовах зростаючої конкуренції, ринкової турбулентності та інформаційного перенавантаження..

Висновок до розділу 1

У першому розділі було розглянуто теоретичні основи прогнозування попиту та управління запасами з акцентом на потреби малого бізнесу. Проаналізовано сучасні підходи до прогнозування, серед яких — статистичні методи, причинно-наслідкові моделі, експертні оцінки, машинне навчання та гібридні рішення. Особливу увагу приділено ролі цифрових технологій, які забезпечують точніший аналіз, адаптацію до змін ринку та зниження людського чинника у прийнятті рішень.

Розкрито ключові проблеми управління товарними запасами, зокрема нестабільність попиту, обмеженість ресурсів, складнощі з плануванням та відсутність автоматизації. Наведено класичні та сучасні методи управління запасами, включно з моделлю EOQ, класифікацією ABC/XYZ, JIT та системами безперервного моніторингу. Доведено, що інтеграція алгоритмів прогнозування у системи управління дозволяє значно підвищити ефективність логістичних рішень.

Особливо цінною є роль методів машинного навчання та штучних нейронних мереж, які дають змогу моделювати складні закономірності у попиті, автоматизувати рішення щодо поповнення товарів та забезпечити адаптивне реагування на зміни зовнішнього середовища. Завдяки доступності відкритих технологій навіть малі підприємства можуть впроваджувати ці інструменти у свої логістичні процеси.

Сформоване у розділі теоретичне підґрунтя підтверджує актуальність застосування інтелектуальних моделей у сфері прогнозування попиту й управління запасами та слугує основою для подальшого проектування прикладної інформаційної системи, що буде описано в наступному розділі.

2 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

2.1 Характеристика процесів управління запасами в малому бізнесі

Управління запасами є одним із ключових елементів у системі логістики малого бізнесу, що охоплює широкий спектр організаційних, аналітичних і операційних дій. Ефективність цього процесу безпосередньо впливає на прибутковість компанії, рівень обслуговування клієнтів, своєчасність виконання замовлень та здатність підприємства швидко адаптуватися до змін ринку. У сфері малого бізнесу, де ресурси обмежені, а попит може бути нестабільним, управління запасами має бути максимально точним, гнучким та прогнозованим.

Невеликі компанії зазвичай мають обмежені бюджети на закупівлю товарів, обмежений складський простір та менший досвід у використанні автоматизованих систем. У таких умовах навіть незначні помилки в управлінні запасами можуть призвести до суттєвих збитків. Надлишкові запаси займають складські площі, призводять до перевитрат на зберігання, втрат через псування або застарівання продукції. З іншого боку, дефіцит запасів може призвести до втрати клієнтів і зниження лояльності споживачів.

Основна мета управління запасами полягає в тому, щоб забезпечити наявність потрібного товару в потрібній кількості, в потрібному місці та у потрібний момент часу. Цей баланс вимагає постійного моніторингу попиту, наявності товарів, строків поставок, зміни сезонності та поведінкових факторів споживачів. У процесі управління запасами важливо враховувати вартість замовлення, ціну утримання запасу, тривалість циклу поповнення та ймовірність непередбачуваних ситуацій (наприклад, зриви постачання чи стрибки попиту).

Застосування сучасних інформаційних систем, зокрема тих, що базуються на штучному інтелекті, дозволяє малому бізнесу виходити за межі традиційного ручного обліку та переходити до проактивного управління запасами.

Такі підходи дають змогу не лише знижувати витрати, а й приймати більш обґрунтовані управлінські рішення, базуючись на аналітиці, прогнозах і візуалізації тенденцій. Типовий процес управління запасами в малому бізнесі включає наступні

етапи, кожен з яких відіграє важливу роль у забезпеченні безперебійного функціонування підприємства та уникненні надлишкових витрат:

Формування асортименту — визначення переліку товарів, які необхідно постійно тримати в наявності, залежно від попиту, сезонності, частоти продажів та специфіки бізнесу. Малий бізнес зазвичай не має можливості утримувати великий складський запас, тому формування асортименту має бути максимально виваженим і точним. Важливо враховувати як історичні дані про попит, так і поведінкові характеристики споживачів.

Облік залишків — ведення точної та актуальної бази даних товарів, що зберігаються на складі, із зазначенням їх кількості, дати надходження, вартості, терміну придатності та обігу. Наявність якісного обліку дозволяє уникати ситуацій, коли товари зникають зі складу, псуються або дублюються через некоректний запис.

Моніторинг продажів — постійний аналіз статистики реалізації дозволяє відстежити, які товари продаються найкраще, в який період відбувається зростання чи спад попиту, та які зовнішні чинники можуть впливати на динаміку продажів. Це створює основу для прийняття зважених рішень щодо обсягів закупівлі та строків поповнення.

Планування закупівель — формування замовлень на поповнення запасів на основі об'єктивних даних. У малому бізнесі помилки на цьому етапі часто спричиняють заморожування обігових коштів або виникнення дефіциту. Доцільно враховувати прогнозовані обсяги продажів, мінімальний необхідний рівень запасу (safety stock) та період постачання.

Реалізація поповнення — здійснення закупівлі відповідно до сформованого плану. На цьому етапі важливо обирати надійних постачальників, враховувати строки доставки, можливість затримок, варіативність умов постачання (мінімальна партія, знижки тощо). У разі використання автоматизованих систем закупівлі може запускатися напівавтоматично при досягненні критичного рівня запасу.

Контроль залишків — регулярна перевірка відповідності фактичних залишків записам у базі, виявлення розбіжностей, списання товарів з вичерпаним терміном придатності, формування звітності. У поєднанні з аналітичними інструментами контроль залишків дозволяє виявити товари з низькою обіговістю, сформувати «мертвий запас» та вжити відповідних заходів щодо його реалізації або оптимізації асортименту.

У малому бізнесі ці процеси часто виконуються вручну або в напівавтоматизованому режимі, наприклад, з використанням електронних таблиць (Excel), простих облікових програм або обмежених онлайн-сервісів. Таке управління має низку обмежень, оскільки покладається на людський фактор, не завжди дозволяє оперативно відстежувати зміни в попиті і має низьку адаптивність до зовнішніх змін.

Основні проблеми, що виникають у процесі управління запасами в умовах малого бізнесу, включають:

- Відсутність прогнозування попиту або його ненадійність. Більшість підприємців покладається на інтуїцію або дані минулих місяців без урахування сезонності, рекламних кампаній чи змін ринку.
- Реактивне, а не проактивне управління. Закупівлі часто здійснюються лише після того, як товари фактично закінчуються. Це створює ризики зриву продажів або затримки обслуговування клієнтів.
- Високий рівень заморожених коштів. Надмірні запаси, закуплені "про запас", займають складські площі, виводять кошти з обороту і часто не реалізуються через зміну попиту.
- Складність обліку змін зовнішніх факторів. Традиційні методи не враховують сезонні коливання, маркетингові активності, поведінкові тренди споживачів чи зміну цінової політики.
- Відсутність сповіщень і автоматичного контролю. Малий бізнес рідко використовує системи, які автоматично повідомляють про досягнення критичних залишків або ризики втрати товару через термін придатності.

У сукупності ці фактори знижують загальну ефективність бізнесу, призводять до втрати прибутку, погіршення рівня обслуговування клієнтів, зростання витрат на зберігання, логістику та компенсацію браку. Підприємства стають менш гнучкими у реагуванні на ринкові зміни та частіше втрачають конкурентні позиції. У таких умовах особливої актуальності набуває впровадження інтелектуальних систем управління запасами, які ґрунтуються на аналізі історичних даних продажів, поведінкових шаблонів споживачів та прогнозуванні майбутнього попиту за допомогою методів машинного навчання та штучних нейронних мереж. Такі рішення дозволяють малим підприємствам вперше отримати переваги, які раніше були доступні лише великим корпораціям із розгалуженою ERP-інфраструктурою.

Ефективне управління запасами в малому бізнесі потребує впровадження сучасних інструментів автоматизації, які не лише полегшують рутинні облікові операції, але й забезпечують багаторівневий аналітичний підхід до прийняття рішень. До таких інструментів належать системи, що інтегрують функції обліку, прогнозування, візуалізації даних та управління логістикою. Автоматизація дозволяє встановлювати оптимальні рівні замовлень, визначати точки поповнення, враховувати сезонність і динаміку продажів, формувати звіти для прийняття стратегічних рішень. Використання інтелектуальних моделей прогнозування попиту — зокрема штучних нейронних мереж — дозволяє адаптувати бізнес до ринкових змін, передбачати сплески попиту, зменшити втрати через непродані запаси та скоротити витрати на зберігання. Така система формує основу для прийняття проактивних, а не реактивних управлінських рішень, що є критично важливим у конкурентному середовищі малого бізнесу. Завдяки впровадженню подібних підходів підприємство отримує змогу оптимізувати ланцюг постачання, покращити фінансові показники та зміцнити свою ринкову позицію навіть за обмежених ресурсів.

2.2 Задача прогнозування попиту та критерії ефективності

У контексті управління товарними запасами ключовою задачею є точне прогнозування попиту, що дозволяє мінімізувати витрати на зберігання, уникнути дефіциту, оптимізувати обіг товару та забезпечити стабільність бізнес-процесів. Точність прогнозу на пряму впливає на здатність малого підприємства своєчасно реагувати на зміни в структурі попиту, запобігати надмірному накопиченню нереалізованого товару та уникати втрати потенційного доходу через нестачу товару. На відміну від великих корпорацій, де можуть бути задіяні цілі аналітичні відділи та дорогі ERP-системи, малий бізнес часто має обмежені можливості для складної аналітики. Тому завдання прогнозування попиту повинно вирішуватись ефективно, швидко та за допомогою адаптивних інструментів. Саме в цьому контексті важливим є застосування технологій штучного інтелекту та машинного навчання, що дозволяють знаходити приховані закономірності у даних, враховувати багатофакторний вплив та динаміку поведінки споживачів. З огляду на нестабільність ринку, постійні коливання попиту, мінливу поведінку клієнтів, вплив акцій, маркетингових кампаній, сезонності, погодних умов, святкових періодів та навіть макроекономічних факторів — традиційні методи стають недостатніми. Вони або не враховують зовнішні змінні, або швидко втрачають актуальність при різких змінах.

Машинне навчання, зокрема нейронні мережі, забезпечують здатність адаптувати модель до нових даних, самооновлюватися з урахуванням нових патернів попиту і таким чином покращувати точність прогнозів у режимі реального часу.

Прогнозування попиту в рамках автоматизованої системи на основі штучного інтелекту не є просто допоміжним інструментом, а стає основним елементом прийняття рішень, що дозволяє підвищити ефективність усього ланцюга постачання, покращити планування закупівель, підвищити рівень обслуговування клієнтів та скоротити витрати малого бізнесу.

У рамках даної кваліфікаційної роботи задача прогнозування формалізується як задача регресійного моделювання, що передбачає побудову моделі, здатної

передбачити числове значення — обсяг попиту — на основі вхідних параметрів. Такий підхід дозволяє не лише отримати конкретне прогнозне значення, але й аналізувати вплив окремих змінних на результат, адаптуючи систему до реальних умов бізнесу.

Вхідні дані:

- Історичні дані про продажі (дата транзакції, кількість реалізованих одиниць, ціна продажу, знижки);
- Параметри товару (назва, категорія, бренд, артикул, поточний запас, точка продажу);
- Зовнішні чинники (день тижня, місяць, сезонність, святкові дні, наявність акцій чи розпродажів, погодні умови);
- Додаткові фактори (кількість переглядів товару, поведінкові патерни покупців, географічна зона попиту).

Ці дані подаються у вигляді структурованого CSV-файлу або інтерактивної веб-форми. Після цього вони проходять етапи очищення, нормалізації та кодування категоріальних ознак для подачі в нейронну мережу.

Вихідні дані:

- Прогнозований обсяг попиту на визначений період (наприклад, на 7, 14 або 30 днів уперед). Модель здатна масштабувати прогноз як по окремих товарах, так і по групах чи категоріях, залежно від вибраного рівня агрегації даних.

Прогнозування здійснюється за допомогою штучної нейронної мережі типу «багатошаровий персептрон» (MLP), реалізованої на основі бібліотеки Keras. Така модель є ефективною для регресійних завдань із числовими виходами та багатьма вхідними параметрами, зокрема в умовах складної багатофакторної залежності. Кожен шар мережі виконує трансформацію даних через вагові коефіцієнти та функції активації, що дозволяє моделі враховувати як лінійні, так і складні нелінійні залежності. Навчання відбувається на основі помилки передбачення, яка мінімізується за допомогою методу зворотного поширення похибки (backpropagation).

Результатом роботи моделі є наближене значення майбутнього попиту на конкретний товар, яке з високою точністю дозволяє приймати обґрунтовані рішення щодо управління запасами, поповнення складу або початку рекламної кампанії.

Приклад структури моделі:

```
from keras.models import Sequential
from keras.layers import Dense

model = Sequential()
model.add(Dense(64, input_dim=5, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(1, activation='linear'))
model.compile(loss='mean_squared_error', optimizer='adam')
```

Така модель отримує на вхід нормалізовані числові ознаки (наприклад, день тижня, ціна, наявність акції у вигляді 0/1) та повертає прогнозований попит у вигляді кількості одиниць товару.

Критерії ефективності прогнозування:

- Середньоквадратична помилка (MSE): показує середній розмір відхилення прогнозу від фактичного значення.
- MAE (Mean Absolute Error): середня абсолютна помилка між передбаченим і реальним попитом.
- MAPE (Mean Absolute Percentage Error): середнє відсоткове відхилення, корисне для порівняння ефективності моделі на різних наборах товарів.
- R^2 (коефіцієнт детермінації): відображає ступінь, у якій модель пояснює варіації у вихідних даних.

Алгоритм дій системи(рис.2.1):

Завантаження або введення даних. Користувач має можливість завантажити CSV-файл з історичними даними про продажі, який містить такі поля, як дата, товар, кількість реалізованих одиниць, ціна, тип акції тощо.

Альтернативно дані можна вводити вручну через інтерактивну вебформу. Це забезпечує гнучкість у роботі з даними, дозволяючи оновлювати або вводити нову інформацію безпосередньо через інтерфейс.

Попередня обробка даних. Завантажені або введені дані проходять етап очищення — видаляються порожні рядки, некоректні значення, дублікати. Далі виконується нормалізація числових даних (наприклад, ціни, кількості), перетворення категоріальних змінних у числовий формат (one-hot encoding), а також формуються нові ознаки (наприклад, індекс сезонності, день тижня, час до свята). Це критично важливо для коректної роботи моделі.

Передача даних до моделі та прогноз. Підготовлені дані передаються в попередньо навчену модель машинного навчання, яка виконує прогноз майбутнього попиту на основі заданого інтервалу (7, 14 або 30 днів). Модель обчислює очікуваний попит для кожної товарної позиції та зберігає результат у базі даних.

Формування візуальних та текстових звітів. Після отримання результатів прогнозу система автоматично формує графіки динаміки попиту, діаграми порівняння «фактичне vs прогнозоване», табличні аналітичні звіти, а також формує фінальний PDF-документ із ключовими результатами прогнозу. Це дозволяє користувачу швидко оцінити ситуацію та прийняти обґрунтоване рішення.

Контроль рівня запасів та сповіщення. Якщо модель виявляє, що прогнозований попит перевищує наявний запас або ж наближається до мінімального критичного рівня, система автоматично генерує сповіщення для користувача. Це може бути повідомлення в інтерфейсі, email або PDF-звіт із попередженням. Метою є забезпечення своєчасного поповнення запасів та запобігання дефіциту.

Задача прогнозування попиту в цій роботі розглядається як багатовимірна аналітична задача, яка поєднує в собі елементи статистичного аналізу, машинного навчання, нейронних мереж, логістичного планування, а також обробки великих обсягів даних (Big Data). Її розв'язання включає не лише побудову математичних моделей, а й інтеграцію аналітичних механізмів у процес управління запасами, що

дозволяє формувати адаптивну стратегію реагування на зміни попиту. Прогнозування враховує широке коло факторів: історичні тренди продажів, сезонність, поведінку споживачів, маркетингові заходи, погодні умови, події, а також неочікувані зовнішні впливи (економічні кризи, пандемії тощо). Такий багатофакторний підхід значно підвищує точність прогнозу і дозволяє ефективно керувати як обсягом закупівель, так і рівнем запасів на складі.

В умовах обмежених фінансових, кадрових і часових ресурсів малого бізнесу, подібна система набуває особливої цінності. Вона дозволяє мінімізувати витрати на зберігання, уникнути надлишку нереалізованої продукції, зменшити втрати від дефіциту, своєчасно поповнювати товарні запаси, а також приймати обґрунтовані стратегічні рішення. Більше того, системи прогнозування можуть бути інтегровані з модулями автоматичного замовлення товарів або CRM-системами, що забезпечує ще вищий рівень автоматизації та ефективності бізнес-процесів малого підприємства.

Розробка системи прогнозування попиту та управління запасами потребує ретельного вибору технологічного стеку, який би забезпечував оптимальний баланс між продуктивністю, гнучкістю, масштабованістю, економічною доцільністю та доступністю впровадження для умов малого бізнесу. У сучасних умовах, коли підприємства прагнуть автоматизувати процеси управління запасами та оптимізувати логістику, важливо інтегрувати інструменти, які забезпечують взаємодію між аналітичними модулями, інтерфейсною частиною та базами даних з мінімальними витратами.

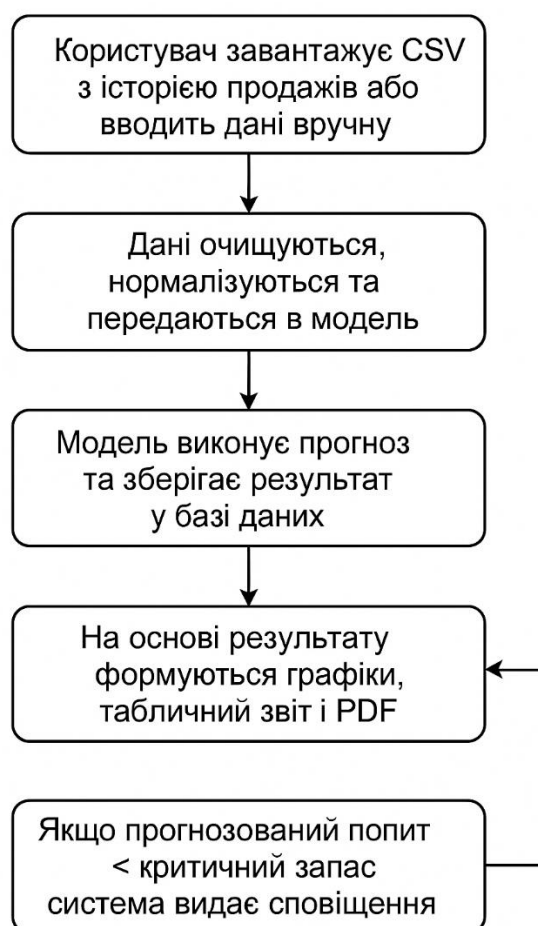


Рис. 2.1 Алгоритм дій системи

2.3 Огляд існуючих рішень та обґрунтування вибору інструментів реалізації

У цьому підрозділі представлено огляд технологій, які використовуються для вирішення подібних завдань, зокрема порівняння підходів із застосуванням Python, PHP, JavaScript, MySQL та допоміжних фреймворків. Обґрунтування вибору конкретних інструментів базується на їх сумісності, простоті інтеграції, підтримці машинного навчання, можливості масштабування, безпеці, продуктивності та зручності розгортання на локальних і віддалених серверах. Особливу увагу приділено організації взаємодії між модулями прогнозування (Python), клієнтською частиною (PHP/HTML/JS), візуалізацією результатів (Chart.js), асинхронною обробкою запитів (AJAX), та структурованим зберіганням даних (MySQL).

Аналіз існуючих рішень

Серед сучасних підходів до реалізації систем прогнозування попиту та управління запасами умовно виділяють два основні напрями:

Моно-архітектурні рішення — побудовані на єдиному середовищі виконання, де вся логіка, обробка даних, аналітика та представлення інтерфейсу реалізуються за допомогою одного технологічного стеку. Наприклад, системи, що повністю реалізовані на Python (з використанням Flask або Django), де і серверна логіка, і API, і HTML-інтерфейс створюються всередині одного середовища. Подібний підхід забезпечує швидкість розробки MVP (мінімально життєздатного продукту), зменшує складність початкового розгортання, але має обмежену масштабованість, менш гнучку структуру та гіршу ізольованість модулів, що ускладнює подальший розвиток.

Розподілені архітектури — побудовані на розділенні функціональних блоків системи. Кожен компонент (інтерфейс, аналітичне ядро, база даних, логіка обробки) виконується окремо і взаємодіє з іншими через стандартизовані інтерфейси (наприклад, REST API). У нашій системі застосовано саме цей підхід: клієнтська частина реалізована на PHP/HTML/JS, аналітична модель — на Python з Flask, дані зберігаються у MySQL, а взаємодія між модулями відбувається через HTTP-запити. Це забезпечує гнучкість оновлення окремих компонентів, можливість паралельного розроблення, масштабування кожного блоку незалежно (наприклад, розміщення ML-моделі на окремому сервері з GPU), спрощення обслуговування, а також інтеграцію з іншими сервісами (наприклад, Telegram-ботами, мобільними додатками або CRM).

Важливо зазначити, що розподілена архітектура також дозволяє легко реалізувати додаткові можливості: кешування запитів, логування, систему сповіщень або бізнес-аналітику на основі даних, які накопичуються в базі. Крім того, кожен модуль може бути реалізований окремими фахівцями своєї галузі (наприклад, ML-інженер відповідає лише за прогноз, а веброзробник — за інтерфейс), що сприяє підвищенню якості розробки.

Обґрунтування вибору мови PHP для клієнтської частини

PHP було обрано як основу для клієнтської частини проєкту з кількох вагомих причин, які враховують вимоги малого бізнесу до вартості розробки, легкості розгортання та сумісності з доступними хостингами. PHP — це мова сценаріїв загального призначення, яка активно використовується для розробки динамічних вебзастосунків. Її відкритий код, велика кількість навчальних матеріалів та документації, а також широка підтримка серед хостингових компаній робить її особливо привабливою для невеликих підприємств без великого технічного штату.

Основні переваги PHP, що вплинули на його вибір:

- велика та активна спільнота, що дозволяє легко знаходити рішення для типових задач;
- відсутність ліцензійних витрат та повна відкритість технології;
- підтримка великою кількістю хостинг-провайдерів (включаючи можливість безкоштовного хостингу для MVP);
- простота інтеграції з MySQL, MariaDB та іншими реляційними СКБД;
- наявність вбудованих механізмів для обробки форм, завантаження файлів, сесій, cookies, захисту від XSS/SQL injection;
- сумісність з фронтенд-бібліотеками (наприклад, AJAX/jQuery для асинхронних запитів);
- можливість використання популярних фреймворків, як-от Laravel або Symfony, для масштабування у майбутньому.

У рамках даного проєкту PHP використовується для наступних цілей:

- створення вебформ для введення параметрів прогнозу;
- передача введених даних на сервер обробки (Flask на Python) через API;
- обробка відповіді від модуля машинного навчання та відображення результату на сторінці;
- зберігання результатів у базі даних MySQL;
- адміністрування записів через окремий інтерфейс (у разі потреби);

- забезпечення взаємодії з іншими сервісами, наприклад, Telegram-ботом чи мобільним застосунком через REST API.

PHP в поєднанні з HTML/CSS та JavaScript створює надійну основу для побудови динамічного, доступного та інтуїтивного інтерфейсу користувача, що критично важливо для систем прогнозування в умовах малого бізнесу, де зручність і швидкість прийняття рішень напряду впливають на ефективність діяльності.

Приклад обробки запиту з форми та взаємодії з API:

```
<?php
$data = [
    'area' => $_POST['area'],
    'rooms' => $_POST['rooms']
];
$ch = curl_init('http://localhost:5000/predict');
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($data));
curl_setopt($ch, CURLOPT_HTTPHEADER, ['Content-Type:
application/json']);
$response = curl_exec($ch);
curl_close($ch);
$result = json_decode($response, true);
echo "<p>Попит: " . $result['demand'] . " одиниць</p>";
?>
```

Використання Python для аналітичного ядра

Мова програмування Python була обрана як основа для реалізації аналітичного ядра системи через її потужні можливості у сфері аналізу даних, машинного навчання та швидкої інтеграції з іншими компонентами. Python є лідером у сфері Data Science завдяки своїй простоті, гнучкості та багатству відкритих бібліотек, які дозволяють реалізовувати як прості моделі, так і складні нейронні мережі.

Основні бібліотеки, що були використані в рамках цього проєкту:

- pandas — для імпорту, очищення та агрегації табличних даних. Використовувалась для попередньої обробки CSV-файлів із історією продажів, нормалізації значень, а також агрегації по періодах (тиждень, місяць).
- NumPy — для роботи з числовими масивами, векторизації операцій, забезпечення швидкої обробки великих обсягів числових даних без необхідності циклів.
- scikit-learn — для створення та навчання моделей машинного навчання. У проєкті застосовано як прості моделі (лінійна регресія), так і методи оптимізації, крос-валідації, вибору ознак. У перспективі можна легко замінити модель на Random Forest або SVM без зміни структури API.
- Flask — легкий вебфреймворк, який дозволяє розгорнути аналітичне ядро як RESTful сервіс. Він отримує HTTP POST-запити від PHP-сервісу, виконує прогноз і повертає JSON-відповідь. Flask є зручним для швидкого прототипування та легко розгортається на сервері або в Docker-контейнері.
- joblib — для серіалізації (збереження) навчених моделей, що дозволяє швидко завантажувати їх у пам'ять під час старту сервісу без необхідності повторного навчання.
- matplotlib / seaborn (у рамках тестування) — для візуалізації трендів, перевірки якості навчання моделі, графічного представлення помилок прогнозування.

Також додатково можуть застосовуватись:

- xgboost або lightgbm — для побудови більш точних моделей у майбутньому;
- keras / tensorflow — у разі переходу на глибокі нейронні мережі;
- scipy / statsmodels — для статистичного аналізу часових рядів та побудови ARIMA-моделей.

Python відіграє ключову роль у створенні точного, адаптивного та масштабованого прогнозного ядра, що може надалі розвиватися в напрямку більш складних моделей машинного навчання або штучного інтелекту.

Приклад коду Flask-сервісу:

```
from flask import Flask, request, jsonify
import joblib

model = joblib.load('model.pkl')
app = Flask(__name__)
@app.route('/predict', methods=['POST'])
def predict():
    data = request.get_json()
    area = data['area']
    rooms = data['rooms']
    prediction = model.predict([[area, rooms]])
    return jsonify({'demand': int(prediction[0])})
app.run()
```

База даних — MySQL

Для збереження історії запитів, даних про товари, користувачів та прогнозів було використано систему керування базами даних MySQL. Це одна з найпоширеніших у світі реляційних СУБД, яка забезпечує швидке зберігання, пошук і обробку структурованих даних. Було створено базу даних та схему зв'язків (рис.2.2) для системи. Вибір MySQL зумовлений її безкоштовною ліцензією, активною спільнотою, підтримкою широкого набору інструментів адміністрування (phpMyAdmin, MySQL Workbench), а також сумісністю з мовами PHP і Python.

Основні переваги MySQL:

- підтримка транзакцій та індексації;
- висока продуктивність при роботі з середніми та великими обсягами даних;
- гнучкість у структурі таблиць і типах даних;
- підтримка зовнішніх ключів і цілісності даних;
- можливість масштабування за рахунок реплікації і кластеризації.

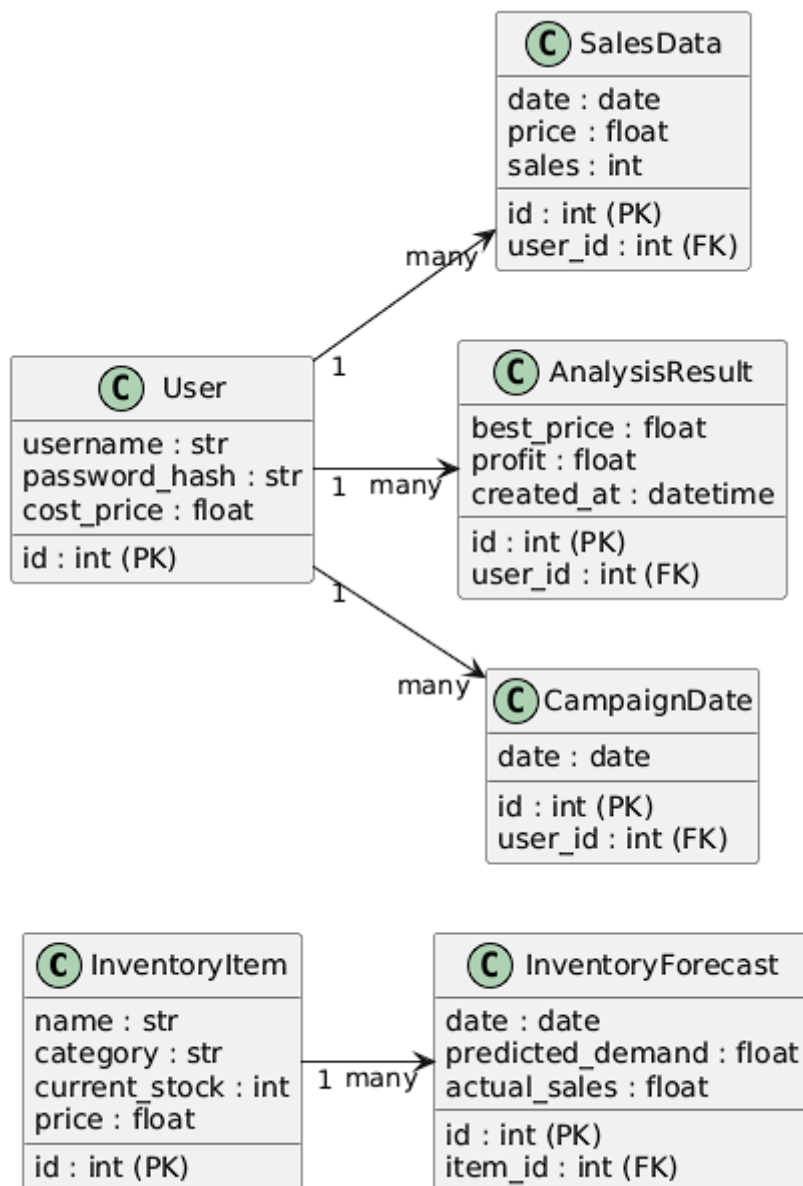


Рис 2.2 Схеми зв'язків бази даних

У межах проєкту створено такі таблиці:

- User - зберігає дані авторизованих користувачів(таб. 2.1);

Таблиця 2.1

User

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор користувача
username	String(64)	Ім'я користувача (логін), має бути унікальним
password_hash	String(128)	Хешований пароль
cost_price	Float	Собівартість товару (використовується у прогнозі прибутку)

- SalesData - містить історію продажів для подальшого прогнозування прибутку (таб. 2.2);

Таблиця 2.2

SalesData

Поле	Тип	Опис
id	Integer, PK	Унікальний запис
user_id	Integer, FK	Користувач, якому належать ці дані
date	Date	Дата продажу
price	Float	Ціна товару у цей день
sales	Integer	Кількість продажів

- AnalysisResult - результати аналітики для оптимізації ціни (таб. 2.3);

Таблиця 2.3

AnalysisResult

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор
user_id	Integer, FK	Ідентифікатор користувача
best_price	Float	Оптимальна ціна за результатами моделі
profit	Float	Очікуваний прибуток
created_at	DateTime	Дата та час проведення аналізу

- CampaignDate - зберігає дати рекламних кампаній (акцій) (таб. 2.4);

Таблиця 2.4

CampaignDate

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор
user_id	Integer, FK	Ідентифікатор користувача
date	Date	Дата проведення акції

- InventoryItem - зберігає дані про товари в запасі(таб. 2.5);

Таблиця 2.5

InventoryItem

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор
name	String(100)	Назва товару
category	String(100)	Категорія товару
current_stock	Integer	Поточний залишок на складі
price	Float	Ціна товару

- InventoryForecast - містить прогнози попиту на кожен товар (таб. 2.6);

Таблиця 2.6

InventoryForecast

Поле	Тип	Опис
id	Integer, PK	Унікальний запис
item_id	Integer, FK	Ідентифікатор товару
date	Date	Дата прогнозу
predicted_demand	Float	Прогнозована кількість продажів
actual_sales	Float (nullable)	Реальні продажі (для порівняння з прогнозом)

Для оптимізації запитів застосовано індексацію по полях user_id, product_id, created_at. Також реалізовано зовнішні ключі для зв'язку requests.user_id -> users.id та requests.location_id -> locations.id, що забезпечує логічну цілісність даних.

У перспективі база даних може бути перенесена до хмарного середовища (наприклад, Amazon RDS, Google Cloud SQL), що забезпечить додаткові можливості для резервного копіювання, відновлення після збоїв і горизонтального масштабування.

Фрагмент PHP-коду для читання з бази даних:

```
$mysqli = new mysqli("localhost", "user", "pass", "dbname");
$result = $mysqli->query("SELECT * FROM products");
while($row = $result->fetch_assoc()) {
    echo $row['name'] . " – " . $row['stock'] . "<br>";
}
```

Додаткові технології

- HTML/CSS/JavaScript — для побудови інтерфейсу користувача. У тому числі використано Bootstrap для адаптивності.
- AJAX/jQuery — для асинхронної взаємодії без перезавантаження сторінки.

Наприклад:

```
$("#predictBtn").click(function() {
    $.post("/predict.php", $("#form").serialize(), function(data)
{
    $("#result").html(data);
    });
});
```

Chart.js — для побудови графіків зміни попиту у часі за результатами прогнозу.

Обраний стек технологій дозволяє створити ефективну, модульну та масштабовану систему, яка легко адаптується під потреби малого бізнесу як на етапі розробки, так і в процесі експлуатації. Кожна з використаних технологій виконує чітко визначену роль у загальній архітектурі системи:

PHP забезпечує обробку взаємодії з користувачем, зокрема прийом і перевірку форм, асинхронні запити до серверного ядра, інтеграцію з базою даних MySQL, а також побудову сторінок результатів. Крім того, PHP є зручним інструментом для швидкого оновлення і масштабування інтерфейсної частини, з можливістю інтеграції фреймворків Laravel або Symfony. Python реалізує прогнозне ядро системи з використанням машинного навчання, обробки великих обсягів даних (CSV, бази), побудови моделей, що враховують сезонність, категорії, поведінкові характеристики. Аналітична модель працює у вигляді окремого Flask-сервісу, що дозволяє масштабувати її незалежно від основного сайту або розміщувати на віддалених серверах (включаючи GPU/Cloud-інфраструктуру). JavaScript/HTML — відповідають за динамічну, інтерактивну та дружню до користувача частину інтерфейсу. Реалізовано валідацію форм, динамічну підстановку результатів прогнозу, графіки зміни попиту за допомогою Chart.js, а

також мобільну адаптацію через Bootstrap. Використання jQuery дозволяє мінімізувати час відповіді і підвищити зручність у роботі з формами без перезавантаження сторінок. MySQL — використовується як реляційна база даних для зберігання всіх ключових компонентів: історії запитів, профілів користувачів, довідників товарів, географічної інформації, результатів прогнозів, журналів помилок тощо. Її структура дозволяє організувати логічну зв'язність даних і забезпечити швидкий доступ до необхідної інформації.

Всі технології є відкритими та мають активні спільноти підтримки. Це дозволяє не лише мінімізувати витрати на впровадження, але й забезпечити легке оновлення, масштабування, додавання нових модулів у майбутньому (наприклад, мобільного застосунку, Telegram-бота або CRM-інтеграції). Саме комбінація цих технологій забезпечує повну функціональність системи: від зручного введення даних до аналітичної обробки та збереження результатів у структурованій базі, придатній для подальшої бізнес-аналітики або побудови звітності.

Висновок до розділу 2

У другому розділі було здійснено глибокий аналіз програмно-технологічних аспектів розробки системи прогнозування попиту та управління товарними запасами. Проведено огляд існуючих архітектурних рішень і методів реалізації, обґрунтовано доцільність використання розподіленої архітектури, що забезпечує масштабованість, модульність та адаптивність системи до потреб малого бізнесу.

Обраний стек технологій — PHP, Python (Flask), MySQL, HTML/CSS/JavaScript з підтримкою бібліотек (Chart.js, jQuery, Bootstrap) — дозволив побудувати ефективну, гнучку і функціональну систему. PHP був використаний для побудови клієнтської частини, Python — для реалізації аналітичного ядра на основі моделей машинного навчання, MySQL — для надійного збереження та обробки даних, а JavaScript і HTML — для створення інтуїтивно зрозумілого та адаптивного інтерфейсу користувача.

Також продемонстровано приклади практичного використання кожної технології у вигляді фрагментів коду, що підтверджує реальну функціональність і взаємозв'язок між компонентами. Окрему увагу приділено безперервності роботи

системи, її здатності до інтеграції з іншими сервісами (API, хмарні БД, мобільні платформи) і можливості подальшого вдосконалення (впровадження глибокого навчання, геоаналітики, CRM тощо).

Реалізована технічна база забезпечує всі необхідні передумови для створення повноцінної аналітичної системи підтримки прийняття рішень у сфері управління запасами, що особливо актуально в умовах обмежених ресурсів малого бізнесу.

3 РЕАЛІЗАЦІЯ СИСТЕМИ ПРОГНОЗУВАННЯ ПОПИТУ

3.1 Архітектура та компоненти веб-застосунку

Цей підрозділ описує архітектурну модель реалізованого застосунку прогнозування попиту на товари, яка побудована на модульному принципі та складається з чотирьох основних логічних рівнів: клієнтського інтерфейсу (frontend), серверної логіки (backend), аналітичного модуля (на Flask) та бази даних (MySQL/PostgreSQL). Кожен із цих рівнів має чітко визначену зону відповідальності, виконує конкретні функції та взаємодіє з іншими компонентами через HTTP-запити, REST API або засоби міжпроцесної передачі даних (наприклад, JSON-обмін). Клієнтська частина (frontend) реалізована з використанням HTML5, CSS3, JavaScript, фреймворку Bootstrap для адаптивної верстки та бібліотеки Chart.js для побудови інтерактивних графіків. Вона відповідає за введення вихідних даних (товари, запаси, CSV), виведення результатів прогнозування у зручному візуальному форматі, а також за формування запитів до серверної частини через AJAX/Fetch API. Серверна частина (backend) побудована на PHP 8, забезпечує обробку запитів із клієнта, валідацію даних, збереження у базу, а також ініціацію запитів до аналітичного ядра. Вона відповідає за логіку маршрутизації, контроль доступу, інтеграцію з системою автентифікації користувачів, облік історії запитів та формування відповідей у HTML або JSON. Аналітичний модуль, реалізований на Flask (Python), є окремим сервісом, що виконує прогнозування попиту на основі моделей машинного навчання. Він включає підсистеми для підготовки даних, виклику ML/NN-моделі (наприклад, Keras, scikit-learn), обробки результату та генерації фінального PDF-звіту з використанням бібліотеки ReportLab. Flask-сервер має власне REST API, що дозволяє інтегрувати його з іншими зовнішніми системами або переносити на окремі хостинги.

База даних (MySQL або PostgreSQL) зберігає всі структуровані дані, зокрема інформацію про товари, користувачів, історію продажів, результати прогнозування, критичні рівні залишків, а також дані рекламних кампаній і собівартості. У базі реалізовані зовнішні ключі, що забезпечують цілісність між

сутностями, а також передбачена можливість створення індексів для пришвидшення пошуку та фільтрації.

Завдяки такій багаторівневій архітектурі система легко масштабована, розширювана й може бути адаптована до нових функціональних вимог без порушення цілісності її структури.

На рис. 3.1 зображено загальну архітектуру додатку.

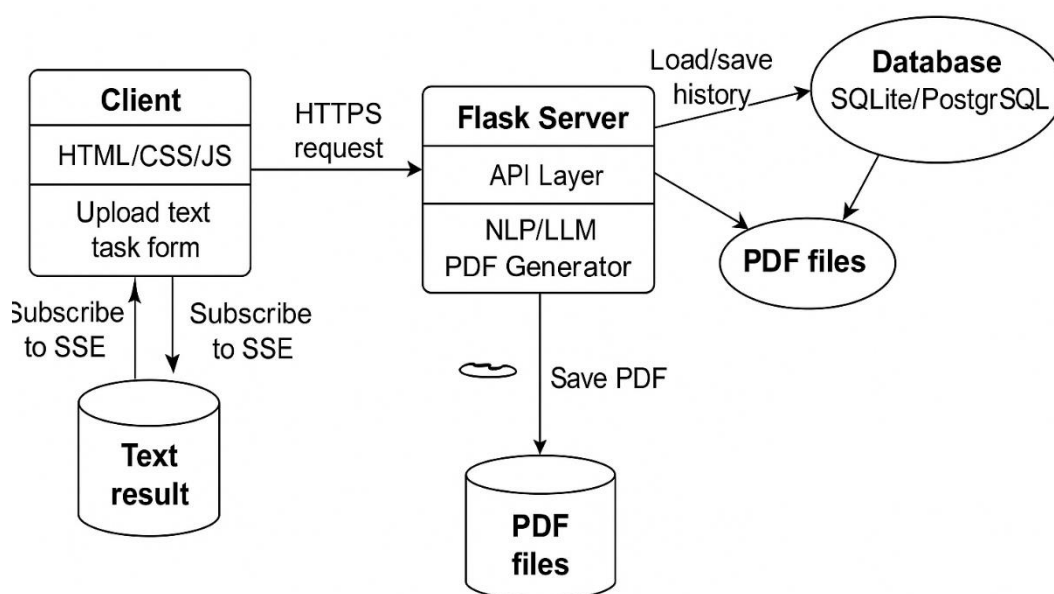


Рис. 3.1 – Архітектура веб-застосунку для прогнозування попиту

Основний потік даних:

1. Користувач взаємодіє з інтерфейсом вебзастосунку, створеним на HTML/CSS/JS. Він має змогу вручну додати нову товарну позицію, оновити наявний запас або імпортувати великі масиви даних за допомогою CSV-файлів. Кожне введення значення проходить базову перевірку на клієнтському рівні (наприклад, через JavaScript-валидацію).

2. Після підтвердження форми дані передаються на сервер через AJAX-запит. PHP-сервер приймає запит, додатково перевіряє його на коректність, зберігає дані в базу (MySQL/PostgreSQL), а також формує запит до аналітичного ядра — окремого сервісу Flask. Це відбувається через HTTP POST-запит із JSON-структурою, яка містить характеристики товару, історію продажів, сезонність, поточний залишок тощо.

3. Flask-сервер приймає запит і активує машинне навчання. Дані проходять через шар попередньої обробки (нормалізація, видалення пропусків), після чого передаються в модель (наприклад, Keras або RandomForest). Модель прогнозує попит на основі заданих параметрів, враховуючи також зовнішні впливи — маркетингові кампанії, сезонність, цінові коливання. Результатом є числове значення прогнозованого попиту на обраний період.

4. Отриманий результат повертається назад до PHP-сервера у форматі JSON. Він може бути відразу виведений у браузері користувача (у вигляді повідомлення або графіка через Chart.js), а також — на вимогу — збережений у форматі PDF (через Flask-генератор звітів ReportLab). Крім цього, PHP записує прогноз у базу даних для ведення історії.

5. Уся історія запитів користувача, разом із параметрами, прогнозами, датами й збереженими PDF-звітами, зберігається у базі. Користувач має змогу у будь-який момент переглянути архів попередніх запитів, повторити прогноз на основі тих самих або змінених даних, а також експортувати статистику у зовнішній файл або таблицю. Таким чином, система забезпечує не лише одноразовий розрахунок, а повноцінне збереження аналітичної інформації для прийняття рішень у майбутньому.

Компоненти:

Клієнт: реалізований із використанням HTML5, CSS3, JavaScript та фреймворку Bootstrap, клієнтський інтерфейс є першою точкою взаємодії користувача із системою. Він включає адаптивну верстку, що дозволяє працювати як із ПК, так і з мобільних пристроїв. Для асинхронної передачі даних між клієнтом і сервером застосовується AJAX. Бібліотека Chart.js використовується для динамічного виводу графіків змін попиту, тенденцій продажів та візуалізації історичних даних. Усі форми включають валідацію даних, динамічне заповнення полів та інтеграцію з зовнішніми модулями, зокрема для імпорту CSV.

Сервер (PHP): серверна частина системи побудована на PHP 8 і виконує ключові функції обробки запитів із клієнта, маршрутизації, валідації, обробки помилок і обміну з базою даних. Також PHP ініціює запити до аналітичного ядра

(Flask), обробляє отриману відповідь, записує результати до БД і формує HTML або JSON-відповіді для фронтенду. Логування всіх подій, обробка сесій, автентифікація користувачів, авторизація дій та захист форм — реалізовані як окремі модулі. При необхідності PHP може бути масштабовано через фреймворки типу Laravel або Slim.

Аналітичне ядро (Python/Flask): є окремим мікросервісом, розгорнутим незалежно від основного вебсервера. Реалізоване на Flask, ядро приймає дані у форматі JSON, проводить попередню обробку (скалінг, нормалізація, one-hot encoding тощо), формує матриці ознак і передає їх у модель машинного навчання. Використовуються такі інструменти як scikit-learn (для регресійних моделей), Keras/TensorFlow (для нейронних мереж), а також xgboost для підвищення точності. Ядро повертає результати у форматі JSON або автоматично генерує PDF-звіт через бібліотеку ReportLab, з графіками, даними і поясненнями. Крім того, API підтримує версіювання моделей, логування помилок, окремий режим тестування та автоматичну перевірку точності моделей (cross-validation).

База даних (MySQL/PostgreSQL): відповідає за централізоване зберігання всієї інформації, включаючи профілі користувачів, товари, історію продажів, запити до прогнозування, результати моделей і лог-файли системи. Структура бази відповідає нормалізованій моделі 3NF, включає зовнішні ключі, унікальні індекси, тригери та обмеження цілісності. Передбачено збереження архівів прогнозів, реалізацію ролей користувачів, кешування даних та підготовку підключення до сторонніх джерел (API маркетплейсів, CRM). У подальшому можливе розширення на хмарні рішення (Amazon RDS, Google Cloud SQL) із підтримкою реплікації та автоматичного резервного копіювання.

Для демонстрації логіки об'єктної моделі застосунку використовується діаграма класів (рис. 3.2).

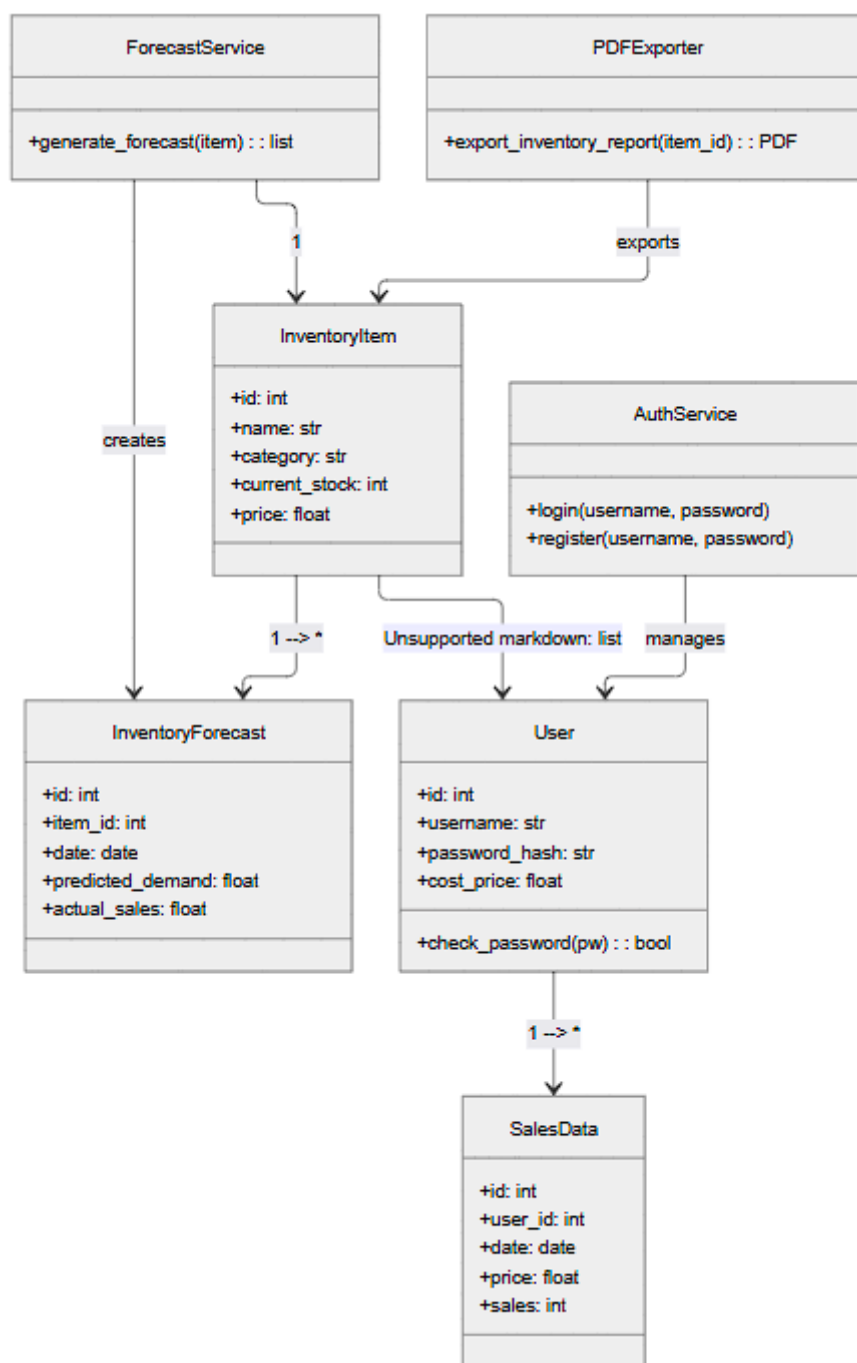


Рис. 3.2 Діаграма класів основної логіки застосунку

3.2 Реалізація прогнозування з використанням нейронної мережі

На етапі реалізації модуля прогнозування попиту було прийнято рішення використовувати модель штучної нейронної мережі (ШНМ) як найбільш ефективний та гнучкий інструмент для обробки складних, багатofакторних і нелінійних залежностей, що виникають у процесах формування попиту на товари. Нейронні мережі здатні адаптуватися до широкого спектра вхідних параметрів, автоматично виявляючи приховані закономірності, які часто не вдається охопити

традиційними статистичними методами, такими як лінійна регресія чи ARIMA. Це особливо актуально у випадках, коли поведінка споживачів формується під впливом неочевидних, динамічних факторів — сезонності, цінових змін, рекламних акцій, зовнішніх економічних умов, специфіки локального ринку тощо. Застосування ШНМ дає змогу будувати нелінійні моделі залежності між історичними даними (наприклад, минулими обсягами продажів), поточними характеристиками товару (розмір, тип, вартість, категорія), поведінковими індикаторами та контекстними змінними (день тижня, сезон, час доби), без необхідності вручну створювати складні математичні моделі. Крім того, нейронна мережа може бути донавчена або повторно адаптована до нових даних, що забезпечує гнучкість і стійкість у змінному бізнес-середовищі. Завдяки своїй архітектурі ШНМ дозволяє проводити прогнозування на основі як структурованих табличних даних, так і складніших сценаріїв із часовими рядами, що відкриває перспективи використання рекурентних або згорткових нейромереж у майбутньому.

Для реалізації було використано мову Python та бібліотеки глибокого навчання — зокрема Keras із бекендом TensorFlow. Навчання моделі здійснювалося на основі попередньо підготовленого CSV-файлу з історією продажів, що включав такі ознаки:

- назва товару (для класифікації категорій),
- тип товару (харчовий, промисловий, технічний тощо),
- категорія (наприклад, напої, побутова хімія, засоби догляду),
- день тижня (для врахування циклічності продажів),
- сезонність (квартал, місяць — впливає на сезонні товари),
- участь у маркетинговій кампанії (0/1 — наявність знижок, акцій),
- поточна ціна (для врахування еластичності попиту),
- попередній обсяг продажів (попит за останній період — тиждень, місяць).

Модель нейронної мережі має наступну архітектуру:

- Вхідний шар (InputLayer) — кількість нейронів відповідає кількості ознак (feature vector).
- Два прихованих шари (Dense) з функцією активації ReLU, кількість нейронів — 32 і 16 відповідно.
- Вихідний шар з одним нейроном та лінійною активацією (Linear), який повертає прогнозоване числове значення попиту.

```
from keras.models import Sequential
from keras.layers import Dense
model = Sequential()
model.add(Dense(32, input_dim=8, activation='relu'))
model.add(Dense(16, activation='relu'))
model.add(Dense(1, activation='linear'))
model.compile(loss='mse', optimizer='adam', metrics=['mae'])
```

Модель навчалась на 80% даних, що були випадковим чином вибрані з усього масиву підготовлених CSV-файлів, у той час як решта 20% використовувались як валідаційний набір. Такий поділ забезпечує незалежну оцінку ефективності моделі на прикладах, які вона раніше не бачила. В процесі навчання застосовувався критерій оптимізації «mean squared error» (MSE), що є стандартним для задач регресії, тоді як для фінального аналізу точності використовувався більш інтерпретований показник — середня абсолютна похибка (MAE). За підсумками навчання, модель досягла MAE на рівні 7–12% залежно від категорії товару, обсягу історичних даних та впливу сезонних коливань. Найбільшу точність було зафіксовано для товарів із стабільною частотою продажів (наприклад, базові продукти харчування), тоді як для сезонних товарів (новорічна продукція, літні аксесуари) похибка була вищою, але залишалася в межах, прийнятних для стратегічного й оперативного планування запасів.

Для підвищення стабільності, узагальнюваності та якості прогнозу було реалізовано низку допоміжних процедур:

нормалізація числових ознак (MinMaxScaler) — для приведення всіх числових значень до одного діапазону $[0,1]$, що дозволяє уникнути домінування окремих ознак при навчанні;

категоріальне кодування (OneHotEncoder) — перетворення текстових і номінальних полів (тип товару, день тижня, категорія) у числовий формат, зрозумілий моделі;

збереження моделі у форматі HDF5 (model.save) — дозволяє швидко завантажити модель після рестарту сервісу або переносу в інше середовище без повторного навчання;

реалізація циклу автоматичного перенавчання — при оновленні історичних даних або значному відхиленні фактичного попиту від прогнозованого, система може автоматично ініціювати перенавчання моделі без залучення розробника;

використання крос-валідації (k-fold cross-validation) — дає змогу уникнути перенавчання, дозволяючи оцінити стабільність результату на декількох незалежних підмножинах.

Прогнозна функція інтегрована у Flask-сервер, що функціонує як окремий мікросервіс, і викликається кожного разу, коли користувач у вебінтерфейсі надсилає запит із параметрами товару. Після обробки й прогнозування результат у вигляді числового значення попиту повертається у форматі JSON. Цей результат виводиться у вебінтерфейсі на окремому модулі «Результат прогнозу» та візуалізується за допомогою графіків. Крім цього, на запит користувача або за замовчуванням, система формує PDF-звіт, у якому міститься: ім'я товару, дата прогнозу, параметри, за якими він здійснювався, очікуване значення попиту, діаграма, а також службова інформація (номер запиту, користувач, ідентифікатор моделі). Цей звіт автоматично зберігається в базу даних і доступний у журналі запитів для подальшого аналізу або повторного використання. Використання нейронної мережі дозволило створити гнучку, точну та масштабовану модель прогнозування попиту, яка може бути легко вдосконалена та адаптована до змін у поведінці ринку та особливостей торгової активності. Така модель дозволяє враховувати не лише класичні параметри товару, а й складні залежності, пов'язані

з поведінкою споживача, регіональними особливостями, поточними ринковими тенденціями, результатами попередніх продажів і зміною обсягів запасів. Крім того, її модульність і здатність до донавчання забезпечують перспективу безперервного покращення системи на основі накопичених даних у реальному часі. Це особливо актуально для малого бізнесу, де правильне прогнозування попиту безпосередньо впливає на оптимізацію закупівель, мінімізацію втрат і підвищення рентабельності.

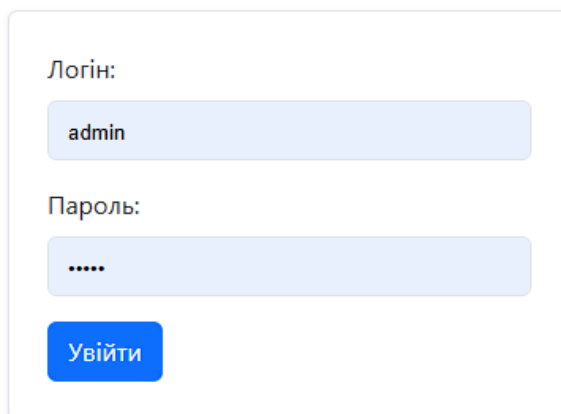
3.3 Інтерфейс керування запасами, сповіщення та генерація звітів

Важливою складовою системи є інтерфейс, з яким безпосередньо взаємодіє користувач у процесі виконання основних бізнес-операцій. Саме через інтерфейс відбувається вся робота із системою: введення даних, формування запитів на прогноз, отримання результатів, керування товарними залишками, генерація звітів і налаштування параметрів. Інтерфейс було розроблено з урахуванням принципів UX/UI-дизайну, що базуються на простоті, доступності, зрозумілості та мінімізації кількості дій для досягнення цільового результату. Особливу увагу приділено адаптивності до різних типів пристроїв — користувач може працювати з системою як з комп'ютера, так і з планшета або смартфона, що забезпечує мобільність і гнучкість у використанні. Кожен функціональний модуль системи реалізований у вигляді окремого екрану або розділу, що дозволяє розмежувати завдання за логічними блоками — авторизація, дашборд, управління запасами, історія продажів, аналітика, звітність, критичні залишки тощо. Це значно полегшує навчання нових користувачів і знижує ймовірність помилок.

Нижче подано приклади основних форм, вікон і таблиць, які реалізовані у вебзастосунку, з коротким описом функціонального призначення кожного з них.

Форма входу до системи (рис. 3.3). Сторінка дозволяє авторизованим користувачам отримати доступ до кабінету за логіном і паролем. Візуально інтерфейс простий, із чітко виділеними полями та кнопкою авторизації.

Вхід до системи



Логін:

Пароль:

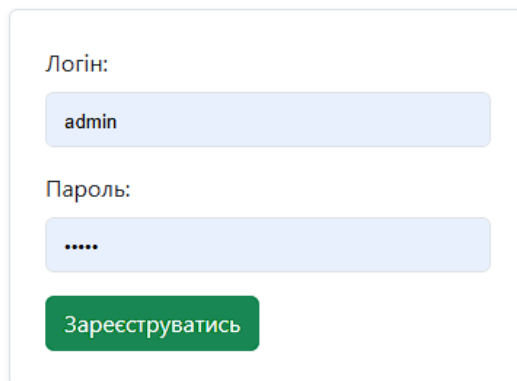
Увійти

Немає акаунту? [Зареєструйтесь](#)

Рис. 3.3 – Вхід користувача до системи

Форма реєстрації(рис. 3.4). Призначена для створення нового облікового запису. Передбачено лише базові поля (логін і пароль), що спрощує старт роботи.

Реєстрація нового користувача



Логін:

Пароль:

Зареєструватись

Вже маєш акаунт? [Увійти](#)

Рис. 3.4 – Реєстрація нового користувача

Головна сторінка — Дашборд (рис. 3.5). Після входу користувач потрапляє до дашборду, де відображено ключові функції: додавання продажів, імпорт CSV, аналіз цін, історію запитів. Також розміщено таблицю запасів, прогноз попиту й функції редагування.

Дашборд

Налаштування
Сповіщення
Вийти

Додати запис вручну

Дата:

Ціна:

Продажі:

Додати

Імпорт для аналізу цін

Виберіть файл
Файл не вибран

Імпортувати CSV

Аналіз цін

Запустити аналіз

Історія аналізів

Переглянути історію

Управління запасами

Завантажити CSV з запасами
+ Додати товар

Продажі (цінова стратегія)
Ваша собівартість: 30.0 грн

Рис. 3.5 – Головна сторінка користувача (Дашборд)

Налаштування профілю (рис. 3.6). Користувач може вказати собівартість товару, яка враховується у прогнозуванні прибутку та цінової оптимізації.

Налаштування профілю

Собівартість товару:

Зберегти

← Назад

Рис. 3.6 – Редагування собівартості товару

Управління запасами (рис. 3.7). Основна таблиця із товарами, категоріями, залишками, цінами та прогнозом попиту. У разі нестачі запасу система автоматично пропонує поповнення.

Управління запасами

Завантажити CSV з запасами

+ Додати товар

Товар	Категорія	Запас	Ціна	Прогноз попиту (на завтра)	Редагувати	Продажі	Графік	PDF
Молоко	Продукти	5	35.0 грн	22.0 Поповнити				
Хліб	Продукти	101	25.0 грн	32.0				
Яйця	Продукти	22	30.0 грн	—				

Рис. 3.7 – Управління запасами товарів

Історія продажів і ціноутворення(рис. 3.8). Відображає дані про попередні продажі: дата, ціна, кількість. Таблиця дозволяє проводити аналітику щодо зміни попиту залежно від вартості.

Продажі (цінова стратегія)

Ваша собівартість: 30.0 грн

Дата	Ціна	Продажі	Дії
2025-05-19	70.0	20	
2025-05-16	124.0	27	
2024-03-29	140.0	396	
2024-03-28	48.0	325	
2024-03-27	110.0	50	
2024-03-26	104.0	53	
2024-03-25	84.0	112	
2024-03-24	54.0	264	
2024-03-23	68.0	170	
2024-03-22	80.0	129	
2024-03-21	99.0	64	

Рис. 3.8 – Журнал продажів товару

Критичні залишки(рис. 3.9). У разі, якщо прогноз попиту перевищує поточний запас, система сповіщає користувача. Це дозволяє своєчасно поповнити товарний запас.

Критичні запаси

Назад до дашборду

Товар	Категорія	Залишок	Прогноз попиту	Статус
Молоко	Продукти	5	22.0	Мало запасу

Рис. 3.9 – Повідомлення про критичні залишки

Історія аналізів(рис. 3.11). Усі результати аналітики зберігаються в окремому журналі. Можна переглянути дату, оптимальну ціну, прибуток і завантажити відповідний PDF-звіт.

Історія ваших аналізів

[← Назад до дашборду](#)

Дата	Оптимальна ціна	Очікуваний прибуток	
2025-05-20 16:22	65.0	7126.867523193359	Завантажити PDF
2025-05-20 10:58	66.0	7644.89794921875	Завантажити PDF
2025-05-20 10:01	66.0	7433.026611328125	Завантажити PDF
2025-05-20 09:49	66.0	7549.031799316406	Завантажити PDF
2025-05-19 16:34	67.0	7714.505645751953	Завантажити PDF
2025-05-17 14:11	65.0	7682.791061401367	Завантажити PDF

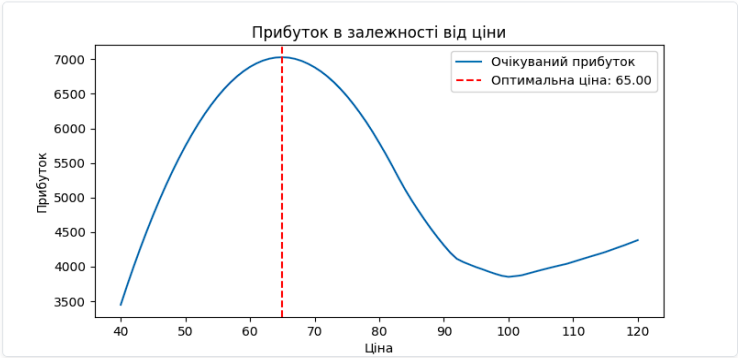
Рис. 3.11 – Журнал історії цінових аналізів

Аналіз цінової стратегії(рис. 3.10). Графік дозволяє визначити оптимальну ціну для максимізації прибутку. Червона лінія – рекомендована ціна, синя – очікуваний прибуток.

Результати оптимізації цінової стратегії

✓ Аналіз завершено. Результат збережено в історії аналізів.

На основі введених вами даних про ціни та продажі, система побудувала модель попиту та виконала оптимізацію прибутку. Нижче представлено графік залежності прибутку від ціни, де червоною лінією позначено найвигіднішу ціну.



Рекомендована ціна: 65 грн

Очікуваний прибуток: 7030.16 грн

[← Назад до дашборду](#)

[📄 Переглянути історію](#)

Рис. 3.10 – Графік оптимізації прибутку

Генерація PDF-звіту(рис. 3.12). Система формує фінальний звіт з ключовими результатами аналізу: дата, ціна, прибуток, а також графік у зручному форматі для друку або наради.

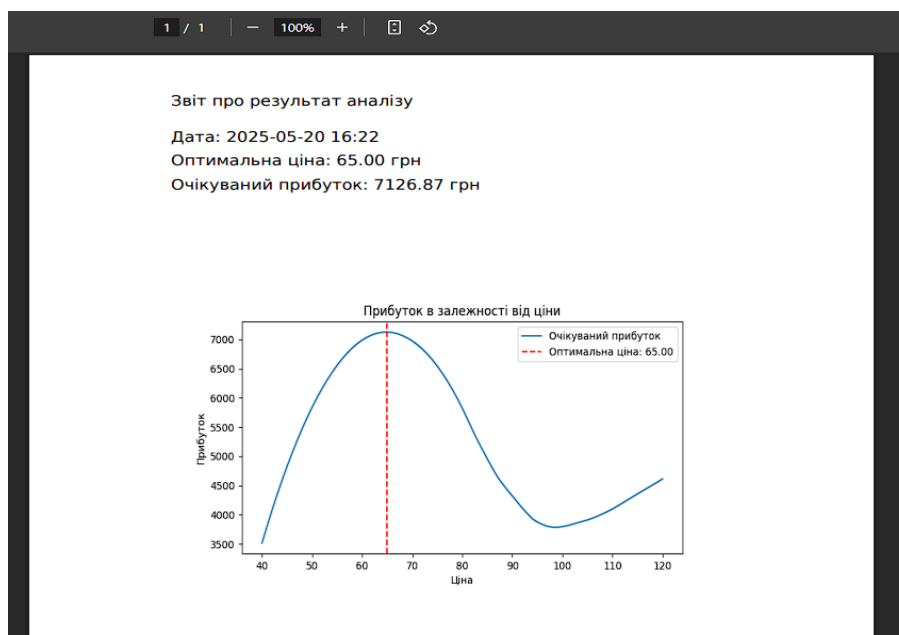


Рис. 3.12 – PDF-звіт результату аналізу

Загалом інтерфейс системи поєднує простоту використання, візуальну зрозумілість, логічну навігацію, адаптивність до різних пристроїв і чітке розділення функціональних блоків. Всі елементи інтерфейсу побудовані таким чином, щоб мінімізувати кількість дій, які має здійснити користувач для досягнення своєї мети — будь то додавання товару, запуск прогнозу, перегляд історії чи генерація аналітичного звіту.

Окремо варто відзначити інтеграцію механізмів візуалізації даних (графіки, статуси, підсвічування критичних значень), які дозволяють швидко орієнтуватися в ситуації з запасами й попитом. Завдяки цьому інтерфейс стає не лише засобом керування, а й інструментом підтримки прийняття рішень, що особливо актуально для малого бізнесу, де швидкість реагування і зручність мають ключове значення..

Висновок до розділу 3

У третьому розділі було детально розглянуто процес реалізації системи прогнозування попиту та керування товарними запасами, орієнтованої на потреби малого бізнесу. Було представлено архітектуру вебзастосунку, що базується на розподіленій моделі із чітким розділенням клієнтської частини, серверної логіки, аналітичного модуля на Flask і бази даних MySQL/PostgreSQL.

Така побудова забезпечує масштабованість, надійність і гнучкість при подальшому розвитку системи. Значну увагу приділено інтеграції модуля прогнозування попиту з використанням нейронної мережі, побудованої за допомогою бібліотеки Keras. Було описано її архітектуру, процес підготовки даних, використання нормалізації, категоріального кодування та крос-валідації. Результати навчання продемонстрували прийнятний рівень точності, що дозволяє ефективно використовувати модель для оперативного планування закупівель. Реалізовано повноцінний графічний інтерфейс для користувача, що охоплює всі ключові функції: додавання товарів, перегляд запасів, запуск прогнозу, аналіз цін, критичні залишки, формування звітів. Інтерфейс є адаптивним, інтуїтивно зрозумілим і дозволяє працювати як з ПК, так і з мобільних пристроїв. Розглянуто приклади форм, таблиць, графіків і PDF-звітів, що ілюструють функціональні можливості системи.

Розроблена система демонструє приклад практичного застосування штучного інтелекту в логістиці малого бізнесу, поєднуючи сучасні методи машинного навчання, веброзробку та бази даних. Вона не лише автоматизує частину рутинних завдань, але й дозволяє підприємцям приймати обґрунтовані рішення, мінімізувати втрати та покращити ефективність управління товарними запасами.

ВИСНОВКИ

У межах дипломної роботи було реалізовано систему прогнозування попиту на основі штучного інтелекту для оптимізації управління запасами в малому бізнесі з використанням сучасних вебтехнологій та методів машинного навчання. У ході дослідження було досягнуто наступних результатів:

Було проведено аналіз існуючих рішень у сфері прогнозування попиту та управління товарними запасами. Вивчено підходи до побудови систем штучного інтелекту для логістичних задач, а також сформульовано вимоги до майбутньої системи: точність прогнозу, простота в користуванні, автоматизація аналізу, інтеграція з базою даних і можливість формування звітів.

Було розроблено повноцінну архітектуру вебзастосунку, що включає: клієнтську частину на HTML/CSS/JS для взаємодії з користувачем, серверну частину на PHP для обробки запитів і роботи з БД, модуль аналітики на Python (Flask) з нейронною мережею на Keras, систему збереження історії в MySQL з логічними зв'язками між товарами, прогнозами, продажами й користувачами.

Було реалізовано повний цикл прогнозування попиту на основі нейронної мережі. Модель було навчено на історичних даних про продажі товарів із урахуванням ознак, таких як категорія, ціна, сезон, участь у маркетингових акціях. Результати прогнозу повертаються в реальному часі та відображаються у вигляді графіків і PDF-звітів. Система підтримує донавчання моделі та повторне використання даних. Розроблено інтуїтивно зрозумілий інтерфейс із повним функціоналом для малого бізнесу: додавання й редагування товарів, контроль запасів, виявлення критичних залишків, автоматичне формування звітів. Інтерфейс адаптований для роботи на різних пристроях та побудований з урахуванням логіки щоденного використання підприємцем. Результати тестування підтвердили стабільну роботу, точність прогнозування та зручність для кінцевого користувача.

Система дозволяє значно знизити втрати від надлишків або дефіциту товарів, оптимізувати закупівлі та підвищити рентабельність бізнесу.

Розроблений застосунок демонструє високу практичну цінність та може бути масштабований для використання в інших сферах торгівлі. Подальший розвиток системи передбачає мобільний додаток, інтеграцію з касовими системами, модулі аналітики прибутку й зв'язок із платформами електронної комерції.

ПЕРЕЛІК ПОСИЛАНЬ

1. Воронкова А. Е. Інформаційні системи в економіці: навч. посібник. — К.: Центр учбової літератури, 2021. — 376 с.
2. Глушков І. С., Виноградов В. В. Логістика: теорія та практика управління запасами. — Харків: ХНЕУ, 2020. — 245 с.
3. Мельник Л. Г., Іванов С. В. Прогнозування в системі управління підприємством: монографія. — Суми: ВТД «Університетська книга», 2022. — 312 с.
4. W3Schools - HTML Tutorial: [Електронний ресурс]. Режим доступу: <https://www.w3schools.com/html/>
5. W3Schools - PHP Tutorial: [Електронний ресурс]. Режим доступу: <https://www.w3schools.com/php/>
6. Bootstrap Documentation: [Електронний ресурс]. Режим доступу: <https://getbootstrap.com/docs/>
7. Chart.js – JavaScript Charting: [Електронний ресурс]. Режим доступу: <https://www.chartjs.org/>
8. Keras Documentation – Deep Learning with Python: [Електронний ресурс]. Режим доступу: <https://keras.io/>
9. TensorFlow – Machine Learning Platform: [Електронний ресурс]. Режим доступу: <https://www.tensorflow.org/>
10. Сковородко В. Г. Прогнозування економічних процесів: навчальний посібник. — Київ: КНЕУ, 2020. — 281 с.
11. Фархадов М. А. Штучний інтелект у прийнятті управлінських рішень. — Львів: Вид-во ЛНУ, 2019. — 198 с.
12. Scikit-learn – Machine Learning in Python: [Електронний ресурс]. Режим доступу: <https://scikit-learn.org/>
13. ReportLab – PDF Toolkit for Python: [Електронний ресурс]. Режим доступу: <https://www.reportlab.com/>
14. Pandas Documentation: [Електронний ресурс]. Режим доступу: <https://pandas.pydata.org/docs/>

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (ПРЕЗЕНТАЦІЯ)



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

Система прогнозування попиту на основі штучного інтелекту для оптимізації управління запасами в малому бізнесі

Виконав студент 4 курсу
групи ШІ-41

Югущий Олександр
Керівник роботи

К.п.н., доц., доцент кафедри ШІ ПІБ

Київ – 2024

ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Провести аналіз літературних джерел щодо методів прогнозування попиту та управління товарними запасами в малому бізнесі.
2. Дослідити існуючі програмні рішення та сервіси для прогнозування попиту, визначити їхні переваги та недоліки.
3. Оцінити можливості використання методів машинного навчання для моделювання попиту на товари.
4. Сформулювати функціональні та нефункціональні вимоги до веб-додатку для оптимізації управління запасами.
5. Спроектувати архітектуру системи з урахуванням:
 - реалізації веб-інтерфейсу для керування товарами;
 - завантаження CSV-файлів із даними;
 - модуля прогнозування попиту (нейронна мережа);
 - управління запасами (зміна, редагування, сповіщення);
 - формування PDF-звітів та виведення графіків.
6. Провести тестування системи, оцінити точність прогнозування та ефективність запропонованої моделі в умовах малого бізнесу.

3

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** – автоматизація процесу прогнозування попиту на товари у малому бізнесі шляхом розробки веб-додатку з використанням методів машинного навчання.
- **Об'єкт дослідження** – процес управління товарними запасами в умовах малих підприємств.
- **Предмет дослідження** – веб-додаток, який реалізує прогнозування попиту та формування рішень щодо поповнення запасів із використанням штучного інтелекту.

2

ВИМОГИ ДО ДОДАТКУ

Функціональні вимоги:

- Реєстрація та авторизація користувачів.
- Завантаження CSV-файлів із даними про запаси та продажі.
- Додавання нових товарів вручну.
- Оновлення запасів із фіксацією змін.
- Прогнозування попиту з використанням моделей штучного інтелекту.
- Візуалізація результатів у вигляді графіків (попит, продажі).
- Сповіщення про критичний рівень запасів.
- Генерація PDF-звітів за обраними товарами.
- Панель адміністратора для керування товарами, прогнозами та користувачами.

Нефункціональні вимоги:

- Підтримка захищених сесій.
- Зберігання даних у базі (SQLite або аналог).
- Висока швидкість при генерації прогнозів.
- Відповідність принципам зручного UX для швидкого прийняття рішень у бізнес-умовах.

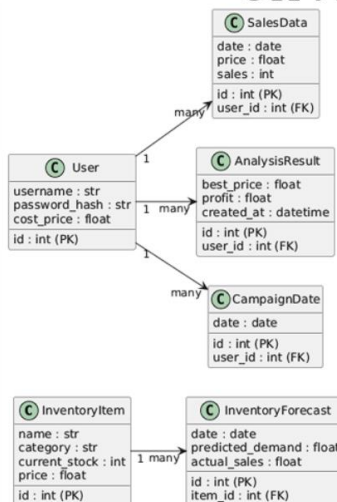
4

ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



5

СТРУКТУРА БАЗИ ДАНИХ



Таблиця User

Зберігає дані авторизованих користувачів.

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор користувача
username	String(64)	Ім'я користувача (логін), має бути унікальним
password_hash	String(128)	Хешований пароль
cost_price	Float	Собівартість товару (використовується у прогнозі прибутку)

Таблиця SalesData

Містить історію продажів для подальшого прогнозування прибутку.

Поле	Тип	Опис
id	Integer, PK	Унікальний запис
user_id	Integer, FK	Користувач, якому належать ці дані
date	Date	Дата продажу
price	Float	Ціна товару у цей день
sales	Integer	Кількість продажів

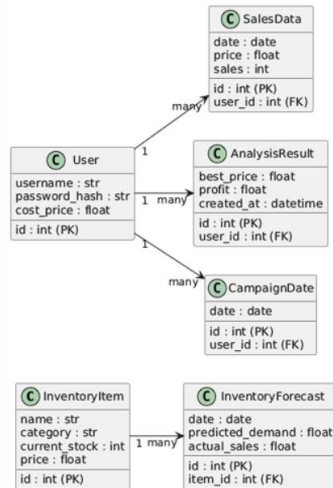
Таблиця AnalysisResult

Містить результати аналітики для оптимізації ціни.

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор
user_id	Integer, FK	Ідентифікатор користувача
best_price	Float	Оптиміальна ціна за результатами моделі
profit	Float	Очікуваний прибуток
created_at	DateTime	Дата та час проведення аналізу

6

СТРУКТУРА БАЗИ ДАННИХ



Таблиця CampaignDate

Зберігає дати рекламних кампаній (акцій).

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор
user_id	Integer, FK	Ідентифікатор користувача
date	Date	Дата проведення акції

Таблиця InventoryItem

Зберігає дані про товари в запасі.

Поле	Тип	Опис
id	Integer, PK	Унікальний ідентифікатор
name	String(100)	Назва товару
category	String(100)	Категорія товару
current_stock	Integer	Поточний залишок на складі
price	Float	Ціна товару

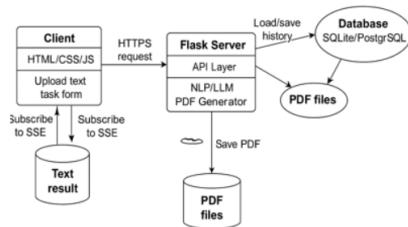
Таблиця InventoryForecast

Містить прогнози попиту на кожен товар.

Поле	Тип	Опис
id	Integer, PK	Унікальний запис
item_id	Integer, FK	Ідентифікатор товару
date	Date	Дата прогнозу
predicted_demand	Float	Прогнозована кількість продажів
actual_sales	Float (nullable)	Реальні продажі (для порівняння з прогнозом)

7

АРХІТЕКТУРА ДОДАТКУ



Основний потік даних:

- Користувач додає або оновлює товар/запас.
- Дані передаються серверу через форму або CSV.
- Нейромережа прогнозує попит.
- Результат відображається на дашборді та/або зберігається у PDF.
- Уся історія — у базі, прогноз можна повторити або експортувати.

Client (Клієнт):

- Інтерфейс, створений на основі HTML/CSS/JS + Bootstrap.
- Форми: додавання товарів, оновлення запасів, імпорту CSV.
- Отримує результати у вигляді графіків або PDF-звітів.
- Відображає сповіщення про критичний рівень запасів.
- Кнопки для запуску прогнозу, експорту та перегляду історії.

Server (Flask API):

API Layer:

- Обробка запитів: додавання товару, релегування, імпорту CSV.
- Авторизація користувачів.
- Взаємодія з базою даних через SQLAlchemy (ORM).

AI/ML Layer:

- Прогнозування попиту з використанням нейронної мережі (Keras, scikit-learn).
- Врахування сезонності, цінових змін, рекламних кампаній.

PDF Layer:

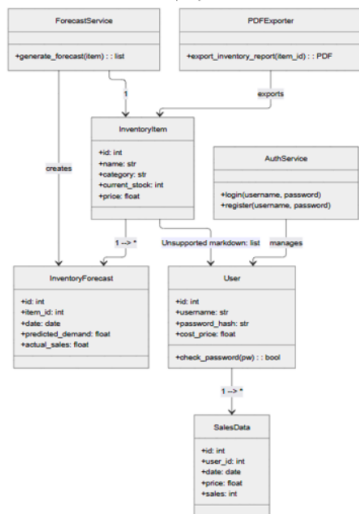
- Генерація фінального звіту за допомогою ReportLab.

Data Layer (База даних):

- SQLite (локальна) / PostgreSQL (на сервері).
- Таблиці: User, InventoryItem, InventoryForecast, SalesData, AnalysisResult.
- Зберігаються товари, залишки, історія прогнозів.

8

ДІАГРАМА КЛАСІВ ОСНОВНОЇ ЛОГІКИ



InventoryForecast

- id: int
 - item_id: FK → InventoryItem
 - date: date
 - predicted_demand: float
 - actual_sales: float
- Зберігає прогнози попиту.

InventoryItem

- id: int
 - name: str
 - category: str
 - current_stock: int
 - price: float
- Модель для управління товарними запасами.

ForecastService

- + generate_forecast(item: InventoryItem) → List[InventoryForecast]
- Використовує ML/NN (Keras) для прогнозу попиту.

User

- id: int
 - username: str
 - password_hash: str
 - cost_price: float
 - + check_password(password): bool
- Відповідає за зберігання облікових даних та собівартості.

SalesData

- id: int
 - user_id: FK → User
 - date: date
 - price: float
 - sales: int
- Історичні дані продажів для навчання моделі.

AuthService

- + login(username, password)
 - + register(username, password)
- Клас для обробки авторизації та реєстрації.

PDFExporter

- + export_inventory_report(item_id: int) → PDF
- Генерує звіт по запасам і прогнозах у PDF-форматі.

9

```

graph TD
    A([Головна сторінка]) --> B([Про застосунок])
    A --> C([Вхід])
    A --> D([Реєстрація])
  
```



Реєстрація

Дашборд ннз

12

Процес аналізу

Імпорт для аналізу цін

Вибрати файл

Файл не вибрано

Імпортувати CSV

Аналіз цін

Запустити аналіз

Історія аналізів

Переглянути історію

Запуск аналізу

Результати оптимізації цінової стратегії

Аналіз завершено. Результат збережено в історії аналізів.

На основі введених вами даних про ціни та продажі, система побудувала модель попиту та виконала оптимізацію прибутку. Нижче представлено графік залежності прибутку від ціни, де червоним лінійно позначено найвигіднішу ціну.

Прибуток в залежності від ціни

Рекомендована ціна: 66 грн
Очікуваний прибуток: 7644.9 грн

Назад до аналізу

Переглянути історію

Результат аналізу

13

Перегляд історії та PDF

Історія ваших аналізів

Назад до аналізу

Дата	Оптимальна ціна	Очікуваний прибуток	
2025-05-20 10:58	66.0	7644.8794621075	Завантажити PDF
2025-05-20 10:01	66.0	7433.02611328125	Завантажити PDF
2025-05-20 09:49	66.0	7548.831789319406	Завантажити PDF
2025-05-19 16:34	67.0	7714.33564571393	Завантажити PDF
2025-05-17 14:11	65.0	7682.793601401367	Завантажити PDF

Звіт про результат аналізу

Дата: 2025-05-20 10:58
Оптимальна ціна: 66.00 грн
Очікуваний прибуток: 7644.90 грн

Прибуток в залежності від ціни

Історія аналізів

PDF статистика

14

Запаси та графіки

Управління запасами

Завантажити CSV з запасами

Додати товар

Товар	Категорія	Запас	Ціна	Прогноз попиту (на завтра)	Редагувати	Продажі	Графік	PDF
Молоко	Продукти	5	35.0 грн	22.0 Повисити				
Хліб	Продукти	101	25.0 грн	32.0				
Яйця	Продукти	22	30.0 грн	—				

Запаси

Динаміка продажів "Молоко"

Оновити запас для "Молоко"

Змінити на (шт):

0

Поточний запас: 5
Новий запас буде: 5

Зберегти

Скасувати

Редагування запасів

Графік продаж

Редагування запасів

15

АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

1. Андрійчук І.Р., Шевченко С.М. Розробка клієнт-серверного застосунку соціальної мережі для розповсюдження фотографій з використанням фреймворків SPRING ТА NEXTJS. Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в ІКТ». Збірник тез. 24.04.2024, ДУІКТ, м.Київ. К.: ДУІКТ, 2024. С. 20-21.

16

ВИСНОВКИ

1. Проведено аналіз літературних джерел щодо сучасних методів прогнозування попиту та управління запасами у малому бізнесі.
2. Досліджено існуючі рішення (ERP/AI-системи) для управління складом, виявлено їхні обмеження та підтверджено актуальність застосування штучного інтелекту для автоматичного прогнозування попиту.
3. Сформульовано вимоги до веб-застосунку, що забезпечує завантаження історичних даних, формування моделей попиту та оптимізацію залишків.
4. Спроектовано архітектуру системи з поділом на клієнтську, серверну, базу даних та модельну частину.
5. Реалізовано веб-додаток на основі Flask із використанням нейромережі (Keras), що виконує прогноз попиту з урахуванням сезонності, дня тижня та рекламних акцій.
6. Додано функціонал управління товарами, ручного коригування запасів, автоматичних сповіщень, графіків та експорту PDF-звітів.
7. Проведено тестування роботи системи та точності прогнозів, що підтвердило відповідність поставленим вимогам і високу прикладну цінність рішення.

17