Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____Комп'ютерних наук_____
(повна назва)

Кафедра_____Системотехніки_____
(повна назва)

# КВАЛІФІКАЦІЙНА РОБОТА
# Пояснювальна записка

_____другий (магістерський)_____
(тема)

Дослідження застосування методів аналізу даних в системах електронної комерції
(тема)

Виконав:
здобувач групи __СПРм-20-1___
_____Гайдар М.І._____
(прізвище, ініціали)

Спеціальності _122 Комп'ютерні науки_
(код і повна назва спеціальності)

Тип програми __освітньо-професійна_____
Освітня програма _ОПП Системне

проектування_____
( повна назва освітньої програми)

Керівник ___професор Калита Н.І._____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри системотехніки _____Гребеннік І.В._____
(підпис) (прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет_____Комп'ютерних наук_____

Кафедра_____Системотехніки_____

Рівень вищої освіти_____другий (магістерський)_____

Спеціальність_____122 Комп'ютерні науки_____

Тип програми _____освітньо-професійна _____
(код і повна назва)

Освітня програма_____ОПП_____Системне проектування_____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
«____»_____ 20 ___ р.

# ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові_____Гайдару Максиму Ігоровичу_____
(прізвище, ім'я, по батькові)

1. Тема роботи ____Дослідження застосування методів аналізу даних в системах електронної комерції_____
_____
затверджена наказом по університету від __08__ листопада___2021 р. № __1664 Ст___
2. Термін подання студентом роботи до екзаменаційної комісії _13_ __грудня__ 2021 р.
3. Вихідні дані до роботи _Розробити серверну і клієнтську частини вебдодатку аналізу даних у системах електронної комерції. Серверна частина є реалізацією бази даних, розробленої для СУБД MySQL. Клієнтська частина має забезпечувати виконання таких бізнес-функцій. Бізнес-функції для незареєстрованих користувачів: перегляд наявних товарів, перегляд інформації за обраним товаром, підбір товарів у кошик, реєстрація та авторизація на сайті. Бізнес-функції для зареєстрованих користувачів: підбір товарів у кошик; оформлення замовлення; перегляд замовлень; вхід до системи зі статусом; Бізнес-функції для адміністраторів: вхід у систему зі статусом «admin»; додавання та редагування інформації про товари; перегляд звітів по результатам аналізу даних отриманих в процесі роботи системи.Операційна система Windows XP або вище, MacOS8 або вище, програмне забезпечення: програмний пакет MySQL Workbench, CASE-засіб All Fusion Data Modeler (ERWin), All Fusion Process Modeler (BPWin), IBM Rational Rose.
4. Перелік питань, що потрібно опрацювати в роботі провести аналіз предметної області та визначити процеси, які вимагають автоматизації і спосіб у якому вона буде реалізована; сформулювати вимоги до інформаційної системи; створити функціональну модель інформаційної системи («ТО-ВЕ»), використовуючи стандарт IDEF0; провести моделювання діаграм потоків даних використовуючи стандарт DFD; провести функціональне моделювання, визначити і уточнити вимоги до інформаційної системи; провести логічне та фізичне моделювання БД з використанням стандарту IDEF1X; обґрунтувати вибір платформи СУБД для реалізації БД; провести UML-моделювання

проектованої клієнтської частини інформаційної системи, створивши діаграму прецедентів (Use Case Diagram), діаграму послідовності дій (Sequence Diagram), діаграму станів (Statechart Diagram), діаграму активності (Activity Diagram) і діаграму класів (Class Diagram); провести аналіз і виділити основні бізнес-процеси, що виконуються на стороні клієнтської і серверної частин інформаційної системи; реалізувати фізичну модель БД для обраної платформи СУБД, створивши серверну частину інформаційної системи; реалізувати посилальну цілісність даних, а також одну з функцій бізнес-процесу на стороні серверної частини інформаційної системи; реалізувати один або кілька бізнес-процесів на стороні клієнтської частини інформаційної системи, розробивши інтерфейс доступу до БД; розробити відповідно до ГОСТ 34.69890 експлуатаційний документ «Керівництво користувача»; підготувати відповідно ГОСТ 19.401-78 програмний документ «Текст програми».

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) _схема організаційної структури підприємства; схеми функціональної структури розроблюваної системи у нотаціях IDEF0, DFD, UML; схеми алгоритмів роботи окремих компонентів системи; логічна та фізична модель БД у нотації IDEF1X;

**КАЛЕНДАРНИЙ ПЛАН**

| № | Назва етапів роботи | Терміни виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | Огляд та аналіз предметної галузі | 13.09.2021 – 19.09.2021 | |
| 2 | Аналіз існуючих системи електронної комерції | 20.09.2021 – 23.09.2021 | |
| 3 | Огляд методів аналізу даних та їх оцінок | 24.09.2021 – 25.09.2021 | |
| 4 | Вибір методів прогнозування | 26.09.2021 – 30.09.2021 | |
| 5 | Аналіз побудованих моделей для вибору даних для аналізу | 02.10.2021 – 9.09.2021 | |
| 6 | Логічне та фізичне проектування БД системи електронної комерції | 11.10.2021 – 16.10.2021 | |
| 7 | Проектування моделей прогнозування та їх оцінка | 17.10.2021 – 24.10.2021 | |
| 8 | Розробка програмно-апаратної частини компонентів системи із прогнозуванням даних | 25.10.2021 – 31.10.2021 | |
| 9 | Розробка клієнтської сторони інтерфейсу | 1.11.2021 – 15.11.2021 | |
| 10 | Підготовка пояснювальної записки та її додатків | 16.1.2021 – 10.12.2021 | |

Дата видачі завдання _____ _____20__ р.

Студент _____
(підпис)

Керівник роботи _____ _____
(підпис) (посада, прізвище, ініціали)

# РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: _75_ с., _2_ табл., _13_ рис., _2_ додатки, _23_ джерел інформації

БАЗА ДАНИХ, СУБД, MYSQL, ЕЛЕКТРОННА КОМЕРЦІЯ, АНАЛІЗ ДАНИХ, C#, ASP, MVC, ML, РЕГРЕСІЯ, ЧАСОВІ РЯДИ

Об’єктом досліджень у кваліфікаційній роботі є аналіз даних в системах електронної комерції.

Предметом дослідження є методи аналізу даних для вирішення задач прогнозування в системах електронної комерції

Мета кваліфікаційної роботи - дослідити методи аналізу даних в системах електронної комерції та реалізувати функції прогнозування на прикладі конкретної системи.

Методи дослідження: системний підхід, методи структурного аналізу і моделювання реляційних баз даних, регресійний аналіз, методи часових рядів, методи проектування веб-застосунків, подієвого об’єктно-орієнтованого програмування.

В роботі проведено аналіз предметної області, що має відношення до методів аналізу даних та їх прогнозування. Для визначення і уточнення вимог до розроблюваного додатку було проведено моделювання потоків даних (з відповідністю до стандарту DFD), логічне та фізичне моделювання (з відповідністю до стандарту IDEF1X). Розроблено UML-діаграми.

Проведено порівняльний аналіз різних методів прогнозування даних і обрано найкращі за оцінками похибки. Розроблені тригери БД для підтримки актуальної інформації у системі. По результатам аналізу методів прогнозування, обрані методи були реалізовані у системі електронної комерції.

Галузь застосування – отримання прогнозу результатів роботи системи електронної комерції для побудови стратегії розвитку і роботи.

# ABSTRACT

Explanatory note to the qualification work: _75_p., _2_ table., _13_ fig., _2_ appendices, _23_ sources of information

DATABASE, DBMS, MYSQL, ELECTRONIC COMMERCE, DATA ANALYSIS, C #, ASP, MVC, ML, REGRESSION, TIME SERIES

The object of research in the qualification work is the analysis of data in e-commerce systems.

The subject of research is data analysis methods for solving forecasting problems in e-commerce systems

The purpose of the qualification work is to investigate the methods of data analysis in e-commerce systems and to implement forecasting functions on the example of a specific system.

Research methods: systems approach, methods of structural analysis and modeling of relational databases, regression analysis, time series methods, methods of designing web applications, event object-oriented programming.

The paper analyzes the subject area related to the methods of data analysis and forecasting. To determine and clarify the requirements for the developed application, data flow modeling (in accordance with the DFD standard), logical and physical modeling (in accordance with the IDEF1X standard) were performed. UML charts developed.

A comparative analysis of different data forecasting methods was performed and the best error estimates were selected. Developed database triggers to support current information in the system. According to the results of the analysis of forecasting methods, the selected methods were implemented in the e-commerce system.

Field of application - obtaining a forecast of the results of the e-commerce system to build a strategy for development and operation.

ЗМІСТ

# ВСТУП

В сучасному світі величезна кількість бізнесів і підприємств знаходить своїх клієнтів у мережі інтернет, через те що вона стає все більш розвитою середою для комунікації і демонстрації свого продукту або послуги користувачу.

Але комунікація не обмежується лише демонстрацією продукту, адже із сучасним рівнем розвитку Інтернет надає можливість ще й замовити товар або послугу і отримати їх онлайн. Такий вид взаємодії клієнта і постачальнику продукту називається електронною комерцією

Електронна комерція сприяє розвитку інших, більш простих видів придбання товарів в інтернеті, що все сильніше полегшує взаємодію користувачів і бізнесу. Завдяки цьому збільшується і клієнтська база таких видів придбання товарів і послуг. Адже дійсно, навіщо витрачати час, відвідуючи реальні магазини із супутніми проблемами на кшталт не зручного місце розташування або відсутності товару на складі, коли можна зробити замовлення онлайн за значно менший час.

Із зростанням кількості користувачів, постають питання їх утримання і пристосування бізнесу або підприємства до їх вподобань. А для цього потрібно аналізувати великі обсяги даних цих користувачі і знаходити шляхи оптимізації роботи систем.

Саме в цьому і полягає актуальність теми «Методи аналізу даних у системах електронної комерції», адже у сьогоднішніх умовах розробка системи з продажу товарів або послуг є лише половиною вирішення проблеми оптимізації процесів реалізації. Маючи на руках величезний обсяг даних отриманих в результаті роботи системи, можливо не лише оптимізувати роботу задля покращення продуктивності або більшої конкурентоспроможності, але навіть прогнозувати обсяги продажів або інші параметри системи.

Кваліфікаційна робота направлена на поглиблення, упорядкування і узагальнення знань з процесів електронної комерції, веб-систем, дослідження доступних методів аналізу даних, проектування та реалізація конкретної системи електронної комерції із використанням різних методів аналізу даних на основі отриманих знань.

Об’єктом дослідження є аналіз даних в системах електронної комерції.

Предметом дослідження є методи аналізу даних для вирішення задач прогнозування в системах електронної комерції.

Мета кваліфікаційної роботи - дослідити методи аналізу даних в системах електронної комерції та реалізувати функції прогнозування на прикладі конкретної системи.

# 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Дослідження поняття «Аналіз даних»

Предметною областю даної кваліфікаційної роботи є дослідження методів аналізу даних у системі електронної комерції.

Для початку необхідно визначитися, що собою являють складові цієї теми.

Аналіз даних – це досить широке поняття, визначення якого дуже різниться в залежності від предметної області та галузі використання. На сьогодні існує велика кількість його дефініцій.

У найбільш загальному використанні, аналіз даних – це обстеження та дослідження, які пов’язані із обрахуванням багатовимірних даних систем, які в свою чергу мають велику кількість змінних [1].

Під час аналізу даних дослідник виконує сукупність дій з метою формування деяких уявлень щодо характерів явищ або інформації, що описується цими даними.

Аналіз даних не можна розглядати лише тільки як обробку інформації після її збору. Аналіз даних - це перш за все засоби перевірки гіпотез і вирішення завдань експериментатора.

Наприклад, аналіз результатів роботи користувачів із системою полягає у знаходженні за загальною сумою замовлень для зареєстрованого користувача розміру знижки і виділенню із даних про замовлення категорії товару який найбільш часто зустрічається. Після цього дані про знижку підкріплюються до даних користувача, а інформація про відповідність користувача і його улюбленої категорії зберігається у системі.

Або ж після оформлення замовлення система аналізує усі виконані замовлення із урахуванням нового і виділяє обмежену кількість товарів які найчастіше купують користувачі. Дані про це зберігаються у системі і використовуються при показі списку товарів.

Також, однією із складових аналізу даних можна вважити прогнозування даних, адже по суті результатами прогнозу є початкові дані оброблені методом певним методом прогнозування.

Усі ці дані дозволяють власникам системи точніше розуміти, що їм потрібно змінювати у роботі системи, а що навпаки не варто, які товари закупати, а від яких варто позбавитися, планувати обсяги закупівлі для оптимізації роботи системи.

У випадку систем електронної комерції аналіз даних дозволяє швидко реагувати на вподобання споживачів і змінюватися відносно них, або взагалі керувати ними. Наприклад, проаналізувавши дані покупок користувача, можна при наступному замовлення рекомендувати до його товарів схожі або підходящі товари.

Відоме протиріччя між обмеженими пізнавальними здібностями людини і складністю та мало вивченістю світу який нас оточує, змушує нас полегшувати завдання аналізу даних використовуючи моделі і моделювання, тим самим спрощуючи вивчення об'єктів, що нас цікавлять, подій і систем.

Модель в традиційному розумінні являє собою результат відображення однієї структури (вивченої) на іншу (маловивчену). Так, відображаючи фізичну систему (об'єкт) на математичну (наприклад, математичний апарат рівнянь), отримаємо фізико-математичну модель системи, або математичну модель фізичної системи. Будь-яка модель будується і досліджується при певних припущеннях, гіпотезах. Робиться це звичайно з допомогою математичних методів [2].

В випадку аналізу даних в системах електронної комерції актуальною є задача прогнозування продажів на майбутні періоди часу на основі даних про продажі за попередні місяці.

## 1.2 Огляд поняття Data mining

Сьогодні за для того аби отримати потрібну інформацію з величезною «купи» сховищ даних використовують те  що називають інтелектуальним аналізом даних - Data Mining, методи аналізу даних якого дозволяють приймати стратегічно важливі рішення у галузі електронної комерції і не лише тут.

Data Mining завдяки широті можливостей, які він відкриває, знайшов широке застосування в науці: його використовують як відмінний метод дослідження. Однак в бізнесі він грає не меншу роль: те, що допомагає науці, рухати людство по шляху до прогресу, дозволяє бізнесу збільшувати прибуток

і кількість лояльних клієнтів. Методи аналізу даних Data Mining в бізнесі дозволяють:

- сегментувати клієнтів;
- прогнозувати продажу;
- проводити аналітику складських запасів;
- приймати рішення про індивідуальні знижки для клієнтів;
- залучати нових клієнтів.

Сьогодні відомі статистичні та кібернетичні методи аналізу даних в Data Mining. Перші базуються на вже накопичених знаннях і даних, другі - на різних математичних підходах.

Статистичні методи аналізу даних Data Mining включають в себе: аналіз вихідних даних, багатомірний статистичний аналіз, аналіз зв'язків і аналіз часових рядів. Кібернетичні методи Data Mining об'єднують підходи, засновані на математиці і застосуванні штучного інтелекту. Ось деякі методи інтелектуального аналізу даних:

- кластеризація - пошук і об'єднання схожих структур і об'єктів (слово «кластер» в перекладі означає скупчення або гроно), не допомагає робити висновки, лише знаходить і об'єднує об'єкти із загальними властивостями;

- алгоритм k-середніх (k-means, швидкий кластерний аналіз) - допомагає визначити гіпотези щодо кількості кластерів, значення k може залежати від раніше проведених досліджень, припущень або навіть інтуїції,

- байєсовські мережі - графічні структури, що представляють ймовірні стосунки між величезним масивом змінних, служать для створення імовірнісного виведення на основі цих змінних,

- штучні нейронні мережі - дуже популярна останнім часом галузь, але перш ніж скористатися нейронною мережею, аналітик її повинен «навчити» (від того, наскільки правильно, вірно і точно буде навчена мережа, залежить її ефективність у вирішенні тих чи інших завдань).

Завдання саме інтелектуального аналізу даних за способом навчання можна розділити на навчання:

- без вчителя (коли машині пропонується будь-який обсяг даних для аналізу і вона самостійно шукає закономірності - наприклад, та залежність продажів певного набору аксесуарів в залежності від куплених товарів);

- і з учителем (коли машині пропонується навчальна вибірка даних, вона на ньому навчається і далі застосовує модель до інших даних);

Крім того, за метою завдання аналізу можна розділити на описові (лише констатують ту чи іншу закономірність і взаємозв'язок) і прогнозувальну (як зрозуміло з назви, не тільки знаходять певні закономірності, а й прогнозують їх наступ в майбутньому).

Отже, аналізуючи усе вище сказане, можна сформулювати задачі аналізу даних [3]:

- задача класифікації: для кожної змінної в обсязі даних призначається категорія або клас, будь-яке значення може належати до певній категорії. Найпростіший приклад: будь-який продукт в продуктовому магазині можна віднести до овочів / фруктів, бакалії, молочної або м'ясної продукції. Єдина складність - кількість класів потрібно визначити заздалегідь;

- задача регресійного аналізу: дуже схожа на класифікацію, але тут досліджується множина значень з безперервного діапазону. В ході дослідження (рішення задачі) проходить пошук шаблонів для визначення цього числового значення;

- задача прогнозування: система на основі послідовностей значень, які вже є, прогнозує нові значення (в тому числі шукає і будує взаємозв'язки). При цьому можуть враховуватися різні фактори - пора року, місяць. Наприклад, зростання продажів канцтоварів перед 1 вересня.

- задача кластеризації (або сегментації): розподіл об'єктів за групами (на відміну від класифікації число груп і їх параметрів може бути заздалегідь не відомо).

- задача визначення взаємозв'язків (знаходження наборів даних серед інших таких же наборів, які зустрічаються частіше - так, наприклад, можна дізнатися, що найбільш регулярні комбінацію продуктів в замовленні і відображати такі товари поруч).

- задача аналізу відхилень: дозволяє визначити події, які невластиві даному набору даних. Так, наприклад, виявляються шахрайські дії в банках.

З точки зору типу даних, які використовуються для аналізу, сьогодні можна виділити технології:

- Data Mining;

- Text Mining (текстовий аналіз);

- Visual Mining (візуальний аналіз);

- OLAP (online analytical processing, аналітична обробка у реальному часі);

- аналіз процесів (Process Mining);

- аналіз Web-ресурсів (Web mining);

- і аналіз в режимі реального часу (Real-Time Data Mining).

З огляду різноманіття зазначених технологій, їх типів даних та представлення типів даних, для них застосовуються спеціальних програмні рішення, сервіси і т.д. Зараз інтенсивно розробляються програмні рішення для автоматизації процесу аналізу - це вимога ринку: грамотний, корисний і зрозумілий аналіз звітів і даних необхідний вже не тільки великим компаніям-лідерам ринку. Невеликі і середні бізнеси теж прагнуть аналізувати і прогнозувати, але при цьому є значно меншими бюджетами та меншим пулом спеціалістів (а іноді і взагалі відсутністю фахівців в області IT або статистики).

## 1.3 Огляд методів аналізу даних

Аналітичний підхід до моделювання базується на тому, що дослідник при вивченні системи відштовхується від моделі. У цьому випадку він виходячи з певних експертних міркувань вибирає потрібну модель. Як правило, це теоретична модель, закон, відома залежність, представлена найчастіше у функціональному вигляді (наприклад, рівняння, що зв'язує вихідний параметр у з вхідними впливами). Варіювання вхідних параметрів на виході дасть результат, який моделює поведінку системи в різних умовах.

По виду моделювання моделі поділяють на:

- емпіричні – ті, які отримані на основі емпіричних фактів, залежностей;
- теоретичні – такі моделі отримані на основі математичних законів;
- змішані, напівемпіричні – вони отримані одночасно і на основі емпіричних залежностей і математичних законів.

Нерідко теоретичні моделі з'являються з емпіричних, наприклад, багато законів фізики спочатку були отримані з емпіричних даних.

Як правило, для аналізу даних використовуються різні математичні методи.

З точки зору інформаційного підходу до аналізу даних, крім моделі, присутні ще три важливі складові: експерт, гіпотеза і аналітик.

Експерт - є ключовою фігурою в процесі аналізу даних. По-справжньому ефективні аналітичні рішення можна отримати не на основі одних лише комп'ютерних програм, а в результаті поєднання кращого з того, що видають людина і комп'ютер. Експерт висуває гіпотези (припущення) і для перевірки їх достовірності або переглядає якісь вибірки різними способами, або будує певні моделі.

Аналітик грає роль спільного знаменника між експертами, тобто є сполучною ланкою між фахівцями різних рівнів і областей. Він збирає у експертів різні гіпотези, висуває вимоги до даних, перевіряє гіпотези і разом з експертами аналізує отримані результати. Аналітик повинен володіти системними знаннями, так як крім завдань аналізу на його плечі часто лягають технічні питання, пов'язані з базами даних, інтеграцією з джерелами даних і продуктивністю. Тому в подальшому головною особливістю в аналізі даних ми будемо вважати аналітика, припускаючи, що він тісно співпрацює з експертами предметних областей.

Перед тим як перейти до огляду безпосередньо методів аналізу даних, необхідно проаналізувати на які категорії аналіз даних умовно поділяють. Поступово збільшуючи складність методів та кількість операції необхідних для їх реалізації опишемо категорії методів аналізу даних.

1. Описовий аналіз – опис наявних даних. Методи описового аналізу є початковою ланкою перед будь-яким аналітичним процесом, і вони описують, що наявні данні відображають. Для цього використовуються методи маніпулювання, упорядкування та інтерпретації сирих даних з різних джерел, щоб перетворити їх на релевантну інформацію для заданого процесу.

Виконання описового аналізу має велике значення, оскільки це допомагає виділити із потоку інформації важливу . Важливо зазначити, що сам по собі цей аналіз не проводить ніякої роботи із даними для більш складних вихідних даних типу прогнозування, або виявлення залежностей, але він дозволяє організувати дані та підготувати їх для подальшого аналізу.

2. Дослідницький аналіз – дослідження зв’язків між даними. Головною метою дослідницького аналізу є дослідження. Після упорядкування даних, ми маємо лише їх нормалізованих вид і можемо починати дослідження

на предмет взаємозв’язків між ними. Після процесу дослідження, дослідницький аналіз дає змогу виявити зв’язки та сформулювати гіпотези та моделі для вирішення конкретних проблем. Типовою сферою застосування дослідницького аналізу є інтелектуальний аналіз даних.

3. Діагностичний аналіз – аналіз причин отримання даних. Один з найпотужніших видів аналізу даних. Аналітика даних отриманих діагностичними методами аналізу даних надає аналітикам і власникам бізнесу можливість отримати чітке контекстне розуміння причин отриманих даних, або результатів роботи. Знаючи причини та процеси при яких були отримані вихідні данні, є можливість точно визначити шляхи вирішення проблем чи проблеми.

Створений для надання прямих відповідей на конкретні запитання, це один із найважливіших у світі методів дослідження, серед інших ключових організаційних функцій, таких як аналітика роздрібної торгівлі.

4. Прогнозний аналіз – прогнозування даних. Методи прогнозування дозволяють побудувати моделі які здатні із заданою точністю прогнозувати данні або результати роботи. Для цього вони використовують результати роботи вищезазначених методів описового, дослідницького та діагностичного аналізу, а також машинного навчання (ML) та штучного інтелекту (AI). Таким чином, відкривається можливість виявляти майбутні тенденції, потенційні проблеми чи неефективність, зв’язки та прогалини у даних.

Завдяки прогнозному аналізу з’являється перспектива розгортки та розробки ініціатив, які покращать різноманітні операційні процеси і допоможуть отримати важливу перевагу в конкурентній боротьбі. Маючи розуміння, чому тенденція, закономірність або подія відбулися на основі даних, є можливість розробки обґрунтованого прогнозу того, як події можуть розвиватися в певних сферах бізнесу.

5. Рекомендаційний аналіз – опис процесів прогнозування. Методи рекомендаційного аналізу даних переходять від прогнозного аналізу таким чином, що вони спрямовані на використання моделей або тенденцій для розробки практичних бізнес-стратегій.

Заглиблюючись у рекомендаційний аналіз, точка зору переноситься у процеси споживання даних, беручи добре впорядковані набори візуальних даних і використовуючи їх як потужне рішення для вирішення виникаючих

проблем у ряді ключових сфер бізнесу, включаючи досвід роботи з клієнтами, маркетинг, логістична аналітика, продажі, HR, фінанси, тощо.

В наш час розрізняють два основні підходи до методів аналізу даних, які зазвичай позначають як «статистичний» і «гуманітарний».

Статистичні або кількісні методи використовуються в основному в галузях наук, які вивчають закономірності, які проявляються в структурі, динаміці та взаємозв'язках соціально-економічних подій (економіка та ін.).

Гуманітарні або якісні методи використовуються в основному в галузях наук, які вивчають закономірності функціонування і розвитку суспільства (політологія та ін.). Необхідно відзначити, що найчастіше ці методи використовуються в одній і тій же роботі одночасно. Крім того, використання цих методів здійснюється в одному і тому ж порядку:

- визначення показників і порядку їх збору;
- збір показників;
- зведення та, при необхідності, угруповання показників;
- обробка показників.

Необхідно мати на увазі, що статистичний або кількісний аналіз дозволяє виявити як динаміку одного показника (одномірна модель), двох показників (бінарна модель), а й одночасно декількох показників (багатофакторний аналіз), що дозволяє виявити вплив прихованих чинників, а також модель колійного аналізу , що дозволяє визначити послідовність прямого або непрямого впливу одного показника на інший.

Основними статистичними методами аналізу даних є [4]:

- кореляційний аналіз;
- регресійний аналіз;
- канонічний аналіз;
- метод порівняння середніх;
- частотний аналіз;
- метод сполучення;
- аналіз відповідностей;
- кластерний аналіз;
- дискримінантний аналіз;
- факторний аналіз;
- багатовимірне ранжування;

- дерева класифікації;
- аналіз головних компонент і класифікація;
- часові ряди;
- карти контролю якості;
- методи аналізу вживаності;
- нейронні мережі;
- планування експериментів;
- моделювання структурними рівняннями.

Гуманітарні або якісні методи передбачають включене спостереження (відкрите і приховане), інтенсивне інтерв'ювання (глибинні індивідуальне і групові інтерв'ю) і т.д.

В даній роботі, беручи до уваги кількісну природу досліджуваних даних, фокус буде саме на статистичному аналізі даних.

Більш детально розглянемо кожен з методів статистичного аналізу даних:

### 1.3.1 Кластерний аналіз

Кластеризація – це процес групування наборів елементів даних таким чином, що ці елементи більше схожі (за певною метрикою) один на одного, ніж на елементи в інших групах. Оскільки при кластеризації немає цільової змінної, метод часто використовується для пошуку невидимих закономірностей у даних. Цей підхід також використовується для надання додаткового контексту тенденції або набору даних.

На сучасному етапі кластеризація часто виступає першим кроком при аналізі даних. Після виділення схожих груп застосовуються інші методи, для кожної групи будується окрема модель.

Кластеризація набуває цінність тоді, коли вона виступає одним з етапів аналізу даних, побудови закінченого аналітичного рішення. Аналітику часто легше виділити групи схожих об'єктів, вивчити їх особливості і побудувати для кожної групи окрему модель, ніж створювати одну загальну модель на всіх даних.

Завдання кластеризації відноситься до класу задач навчання без учителя.

Етапи кластерного аналізу:

1. Відбір вибірки для кластеризації.

2. Визначення множини змінних, за якими будуть оцінюватися об'єкти у вибірці, тобто простору ознак.

3. Приведення всіх даних до єдиної шкали (нормування, стандартизація).

4. Обчислення значень тієї чи іншої міри подібності (або відмінності) між об'єктами.

5. Застосування методу кластерного аналізу для створення груп схожих об'єктів.

6. Перевірка достовірності результатів кластерного рішення.

Більшість алгоритмів кластеризації припускають порівняння об'єктів між собою на основі певної міри близькості (подібності). Мірою близькості називається величина, що має межу і зростаюча зі збільшенням близькості об'єктів. Міри схожості «винаходяться» за спеціальними правилами, а вибір конкретних заходів залежить від завдання, а також від шкали вимірювань. В якості запобіжного близькості для числових атрибутів дуже часто використовується евклідова відстань, що обчислюється за формулою:

$$D(x, y) = \sqrt{\sum_i (x - y)^2}$$

де D – евклідова відстань, x і y числові атрибути.

Прикладом алгоритму кластеризації є K-means.

Кластеризація K означає тип неконтрольованого навчання, яке використовується, коли у вас є дані без міток (тобто дані без визначених категорій або груп). Метою цього алгоритму є пошук груп у даних із кількістю груп, представлених змінною K. Алгоритм працює ітеративно, щоб присвоїти кожну точку даних одній із груп K на основі наданих функцій. Точки даних групуються на основі подібності функцій. Результатами алгоритму кластеризації K-середніх є:

- центроїди кластерів K, які можна використовувати для позначення нових даних;
- мітки для навчальних даних (кожна точка даних призначена для одного кластера).

Замість того, щоб визначати групи перед переглядом даних, кластеризація дозволяє знайти та проаналізувати групи, які утворилися органічно.

Кожен центроїд кластера - це набір значень властивостей, які визначають результуючі групи. Аналіз ваг об’єктів центроїда можна використати для якісної інтерпретації того, яку групу представляє кожен кластер [5].

З точки зору бізнесу маркетологи за допомогою методів кластерного аналізу мають змогу групувати клієнтів у кластери на основі, наприклад, демографічних показників, купівельної поведінки, грошової вартості або будь-якого іншого фактора, який може бути релевантним для певної предметної області, що наддасть змогу оптимізувати зусилля та надати клієнтам найкращий сервіс на їх потреби.

### 1.3.2 Когортний аналіз

Цей тип методу аналізу даних використовує дані роботи системи для вивчення та порівняння визначеного сегмента поведінки користувачів, який потім можна згрупувати з іншими з подібними характеристиками. Використовуючи цю методологію аналізу даних, можна отримати глибоке уявлення про потреби споживачів або чітке розуміння ширшої цільової групи.

Когортний аналіз є корисним під час аналізу даних в маркетингу, оскільки він дозволяє зрозуміти вплив рекламних акції або різних впливів на певні групи клієнтів.

Корисним інструментом для початку виконання методу когортного аналізу є Google Analytics.

### 1.3.3 Регресійний аналіз

Регресійний аналіз використовує дані роботи системи, щоб зрозуміти, як значення певної залежної змінної впливає, коли одна (у випадку лінійної регресії) або більше незалежних змінних (множинна регресія) змінюються або залишаються незмінними. Маючи розуміння щодо взаємозв’язку кожної

змінної та процесів їх зміни у минулому, є можливість передбачити можливі результати та приймати кращі бізнес-рішення в майбутньому.

Логістична регресія - це простий та більш ефективний метод вирішення бінарних та лінійних задач класифікації.

Це класифікаційна модель, яку дуже легко реалізувати і вона досягає дуже хороших показників за допомогою лінійно роздільних класів. Він є широко використовуваним алгоритмом класифікації в промисловості.

Модель логістичної регресії, як і персептрон, є статистичним методом для двійкової класифікації, який можна узагальнити на багатокласову класифікацію.

Лінійна регресія корисна для пошуку зв'язку між двома безперервними змінними. Одна з них - це предиктор або незалежна змінна, а інша - відповідь або залежна змінна. Він шукає статистичні відносини, але не детерміновані відносини. Взаємозв’язок між двома змінними називається детермінованим, якщо одна змінна може бути точно виражена іншою.

Наприклад, при проведенні регресійного аналізу продажів у 2019 році було виявлено, що такі змінні, як якість продукції, дизайн магазину, обслуговування клієнтів, маркетингові кампанії та канали продажів, впливають на загальний результат. Тепер є можливість використовувати регресію, щоб проаналізувати, які з цих змінних змінилися або з’явилися нові протягом 2020 року. Таким чином, з’являється розуміння, які незалежні змінні вплинули на загальну ефективність залежної змінної - річного обсягу продажів.

Основними методами регресійного аналізу є [6]:

- SDCA (Stochastic Dual Coordinate Ascent) - стохастичний метод подвійних координат;
- моделі регресії Пуассона;
- дерева регресії.

Стохастичний градієнтний спуск (SGD) став популярним методом вирішення великомасштабних проблем оптимізації контрольованих моделей машинного навчання, через їх гарні теоретичні гарантії. У той час як схожий метод подвійних координат (DCA) був реалізований у різних пакетах програмного забезпечення, йому досі бракувало хорошого аналізу схожості. По суті, SDCA – це реалізація звичайної логістичної регресії із удосконаленим

алгоритмом оптимізації заснованому на стохастичному методі подвійних координат.

Регресія Пуассона подібна до звичайної множинної регресії, за винятком того, що залежна (Y) змінна є спостережуваним числом, яке має розподіл Пуассона. Таким чином, можливими значеннями Y є цілі невід'ємні числа: 0, 1, 2, 3 тощо. Передбачається, що великі числа зустрічаються не часто. Отже, регресія Пуассона подібна до логістичної регресії, яка також має дискретну множину результатів. Однак результати не обмежуються конкретними значеннями, як це відбувається в логістичній регресії.

Одним із прикладів відповідного застосування регресії Пуассона є дослідження того, як кількість колоній бактерій пов'язана з різними умовами навколишнього середовища та розведеннями. Інший приклад – кількість відмов для певної машини при різних умовах експлуатації. Ще одним прикладом є дуже важливі статистичні дані щодо дитячої смертності або захворюваності на рак серед груп з різною демографічною групою.

Загальна ідея метода дерев регресії полягає в тому, що простір предикторів сегментується на декілька простих областей.

Для того, щоб зробити прогноз для даного спостереження, ми зазвичай використовуємо середнє значення навчальних даних у області, до якої воно належить. Оскільки набір правил розділу, що використовуються для сегментації простору предикторів, можна підсумувати за допомогою дерева, такі підходи називаються методами дерева рішень.

Лінійна регресія — це глобальна модель, де існує єдина прогнозна формула, що зберігається на всьому просторі даних. Коли дані мають багато функцій, які взаємодіють складним, нелінійним чином, побудова єдиної глобальної моделі може бути дуже складною і перевантаженою, коли це все ж вдасться.

Альтернативний підхід до нелінійної регресії полягає в тому, щоб поділити простір на менші області, де взаємодія більш керована. Після цього знову виконується процес розбиття на підрозділи — це називається рекурсивним розбиттям — поки, нарешті, модель не прийде до таких кусків простору, які настільки ручні, що відкривається можливість підігнати до них прості моделі. Таким чином, глобальна модель складається з двох частин: одна

— це лише рекурсивне розділення, інша — проста модель для кожної частини розділу.

Дерева регресії використовують дерева для представлення рекурсивного розділу. Кожен з кінцевих вузлів або листків дерева представляє частину розділу і асоційовану із ним просту модель, яка застосовується тільки в цій частині. Точка x належить листку, якщо x потрапляє у відповідну частину розбиття. Щоб зрозуміти, в якій частині модель знаходиться, вона починає з кореневого вузла дерева і задає послідовність запитань про функції. Внутрішні вузли позначені запитаннями, а краї або гілки між ними позначені відповідями.

Яке питання буде поставлено далі, залежить від відповідей на попередні запитання. У класичній версії кожне запитання відноситься лише до одного атрибута і має відповідь так чи ні, наприклад, «Чи більше 20 продажів у день?» або «Чи є жанр == ПРАВДА?»

Необхідно відмітити, що не всі змінні мають бути одного типу; деякі можуть бути безперервними, деякі можуть бути дискретними, але впорядкованими, деякі можуть бути категоричними тощо.

Ці методи прості та корисні для інтерпретації.

### 1.3.4. Нейронні мережі

Нейронна мережа є основою для інтелектуальних алгоритмів машинного навчання. Це одна з форм комп’ютерної аналітики, яка керується даними і намагається зрозуміти, як людський мозок обробляв би ідеї щодо взаємозв’язків у вхідних даних та передбачає значення. Нейронні мережі навчаються на окремих наборах даних, а це означає, що вони еволюціонують і розвиваються з часом.

Типовою сферою застосування нейронних мереж є прогнозний аналіз даних.

### 1.3.5 Прогнозування часових рядів

Прогнозування часових рядів — це процес аналізу даних часових рядів із використанням статистичних даних та моделювання для прогнозування та надання інформації для прийняття стратегічних рішень.

Не завжди методи часових рядів дають точний прогноз, і ймовірність прогнозів сильно відрізняється, особливо коли аналіз проводиться на змінних які часто коливаються, а також через факторів поза контролю. Зазвичай, чим коректніші і повніші ми маємо дані, тим точнішими можуть бути прогнози.

Хоча прогнозування і «прогноз» загалом означають одне й те саме, є помітна відмінність. У деяких галузях прогнозування може стосуватися даних на певний момент у майбутньому, тоді як прогноз стосується майбутніх даних загалом. Прогнозування рядами часто використовується в поєднанні з аналізом часових рядів. Аналіз часових рядів включає розробку моделей для кращого розуміння даних. Аналіз може надати причини результатів, які він генерує. Потім прогнозування робить наступний крок щодо того, що робити з цими знаннями та передбачуваними екстраполяціями того, що може статися в майбутньому.

Аналіз часових рядів надає комплекс методів для кращого розуміння набору даних.

Основними компонентами часового ряду є:

1. Рівень - базове значення для ряду, якби це була пряма лінія.
2. Тренд - необов’язковий і часто лінійно зростаюча або спадаюча поведінка ряду у часі.
3. Сезонність - необов'язкові повторювані шаблони або цикли поведінки у часі.
4. Шум - додаткова мінливість у спостереженнях, яку не можна пояснити моделлю.

Усі часові ряди мають рівень, більшість із них мають шум, а тенденція та сезонність є необов’язковими.

Основними ознаками багатьох часових рядів є тенденції та сезонні коливання. Іншою важливою особливістю більшості часових рядів є те, що спостереження, близькі один до одного в часі, мають тенденцію корелювати.

Можна вважати, що ці складові компоненти певним чином поєднуються, щоб забезпечити спостережуваний часовий ряд.

Про ці компоненти можна робити припущення як за поведінкою, так і за тим, як вони поєднуються, що дозволяє моделювати їх за допомогою традиційних статистичних методів [7].

## 1.4 Огляд способів оцінки моделей прогнозування

Прогнозне моделювання — це процес розробки моделі з використанням історичних даних для прогнозування нових даних, які не відомі на поточний момент.

Прогнозне моделювання можна описати як математичну задачу апроксимації функції відображення (f) від вхідних змінних (x) до вихідних змінних (y). Це називається проблемою апроксимації функції.

Завдання алгоритму моделювання полягає в тому, щоб знайти найкращу функцію відображення, яку можливо, враховуючи наявний час і ресурси.

Виходячи з цього постає питання, як розрахувати точність отриманої моделі прогнозування.

Точність є мірою класифікації, а не регресії. Неможливо обчислити точність для регресійної моделі. Працездатність або продуктивність регресійної моделі мають бути передані, як похибка в прогнозуванні.

Якщо прогнозувати числове значення, наприклад зріст або суму в доларах, ми не хочемо знати, чи точно модель передбачила значення (на практиці це може бути надзвичайно складно); замість цього ми хочемо знати, наскільки прогнози були близькими до реальних очікуваних значень.

Похибка в прогнозуванні відображає саме це і підсумовує в середньому, наскільки близькими були прогнози до їх очікуваних значень.

Є три показники похибок, які зазвичай використовуються для оцінки та звітності про ефективність регресійної моделі; вони є:

середня квадратична похибка (MSE);

середньоквадратична похибка (RMSE);

середня абсолютна похибка (MAE).

Існує багато інших показників оцінки регресії, але три вищезазначених є найбільш часто використовуваними.

Mean Squared Error, або MSE скорочено, є популярним способом оцінки похибки для моделей регресії.

Це також важлива функція аналізу втрат для алгоритмів придатності або оптимізації з використанням методу найменших квадратів для вирішення задачі регресії. Тут «найменші квадрати» відносяться до мінімізації середньоквадратичної похибки між прогнозами та очікуваними значеннями.

MSE розраховується як середнє значення квадратів різниць між прогнозованими та очікуваними цільовими значеннями в наборі даних:

$$MSE = \frac{1}{N} * \sum_{i=1}^{N} (y_i - \widetilde{y_i})^2 ,$$

де $y_i$ – реальні данні,

$\widetilde{y_i}$ – прогнозовані дані,

N – кількість спостережень.

Піднесення в квадрат також має ефект роздування або збільшення великих похибок. Тобто, чим більша різниця між прогнозованим і очікуваним значеннями, тим більшою буде отримана в квадраті позитивна похибка. Це призводить до ефекту «покарання» моделей за більші похибки, коли MSE використовується як функція втрат.

Середньоквадратична похибка, або RMSE, є розширенням середньоквадратичної похибки:

$$RMSE = \sqrt{\frac{1}{N} * \sum_{i=1}^{N} (y_i - \widetilde{y_i})^2} ,$$

де $y_i$ – реальні данні,

$\widetilde{y_i}$ – прогнозовані дані,

N – кількість спостережень.

Важливо, що обчислюється квадратний корінь похибки, що означає, що значення RMSE є такими ж, як вихідні значення цільового значення, яке прогнозується.

Наприклад, якщо цільова змінна має одиниці вимірювання «долари», то оцінка похибки RMSE також матиме одиницю «долари», а не «долари в квадраті», як MSE.

Таким чином, може бути поширеним використання MSE для навчання регресійної моделі прогнозування та використання RMSE для оцінки та звітування про її ефективність.

Середня абсолютна похибка, або MAE, є популярним показником, оскільки, як і RMSE, одиниці оцінки похибки збігаються з одиницями цільового значення, яке прогнозується:

$$MAE = \frac{1}{N} * \sum_{i=1}^{N} |y_i - \widetilde{y_i}| ,$$

де $y_i$ – реальні данні,

$\widetilde{y_i}$ – прогнозовані дані,

N – кількість спостережень.

На відміну від RMSE, зміни MAE є лінійними і, отже, інтуїтивно зрозумілими. Тобто, MSE і RMSE карають моделі за більші похибки сильніше, ніж за менші, роздуваючи або збільшуючи середню оцінку похибки. Це пов’язано з квадратом значення похибки. MAE не надає більшої чи меншої ваги різним типам помилок, а замість цього оцінки збільшуються лінійно зі збільшенням похибки.

Як видно з назви, оцінка MAE розраховується як середнє значення абсолютної похибки. Таким чином, різниця між очікуваним і прогнозованим значенням може бути додатною або негативною і змушена бути позитивною при розрахунку MAE.

# 2. ОГЛЯД МЕТОДІВ ТА ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ СИСТЕМ

## 2.1 Опис CASE засобів моделювання інформаційних систем

Перед початком проектування та розробки інформаційної системи необхідно визначитися із усіма інформаційними технологіями, які будуть використані під час процесів проектування системи та безпосередньо її розробки.

Перш за все, для побудови моделі, яка спростить процес реалізації веб додатку, будуть використані CASE засоби системного і функціонального моделювання AllFusion Process Modeler [8]. Дана інформаційна технологія дозволяє будувати моделі системи у нотаціях IDEF0 і DFD. Для моделювання діаграм Use Case є зручним використати програмний засіб IBM Rational Rose. Під час побудови логічної і фізчиної моделі, використовуються інформаційні технології AllFusion Data Modeler і MySQL Workbench відповідно.

Для формулювання функціональних вимог до вебсайту автоматизації процесів електронної комерції і аналізу даних у ній, необхідно розробити функціональну модель даної системи за допомогою стандарту IDEF0.

Стандарт IDEF0 (Integrated Definition Function Modeling) використовується для створення функціональних моделей, що відображають структуроване зображення функцій виробничої системи або середовища, а також інформації і об’єктів, що зв’язують ці функції.

З погляду розробника IDEF0 – це методологія функціонального моделювання, що дозволяє описати бізнес-процеси системи у вигляді ієрархічної системи взаємозалежних функцій.

Стандарт Data Flow Diagrams (DFD) входить до методології графічного структурного аналізу. Згідно стандарту DFD створюється моделі потоків даних систем, що досліджуються або розробляються [9].

Діаграми потоків даних представляють, яким чином кожний функціональний блок (процес) системи оброблює вхідні дані у вихідні, зберігає або використовує дані зі сховищ даних, показують відношення між цими процесами. DFD-діаграми використовуються як доповнення до функціональної моделі IDEF0 для опису даних документообігу і обробки інформації.

DFD - це нотація, призначена для моделювання інформаційний систем з точки зору зберігання, обробки і передачі даних.

Діаграми потоків даних системи, що створюється за допомогою стандарту DFD, може бути використані:

- для аналізу функцій існуючої системи та визначення вимог до її доробки, у тому числі структури даних (бази даних);

- для визначення функціональних вимог до розроблювальної системи;

- для визначення вимог до структури даних (бази даних), видів та типів даних, що потрібні для виконання функцій системою, що розробляється.

Поряд із функціональним моделюванням IDEF0 і DFD, створюються Use Case діаграми за допомогою інформаційної технології IBM Rational Rose яка дозволяює моделювати розроблювану систему за допомогою діаграм варіантів використання, діаграм класів, діаграм послідовності дій і діаграми станів [10].

Діаграми варіантів використання служать для проведення ітераційного циклу загальної постановки завдання разом із замовником.

Варіант використання являє собою послідовність дій, виконуваних системою у відповідь на подію, що ініціюється деяким зовнішнім об'єктом (дійовою особою). Варіант використання описує типову взаємодію між користувачем і системою.

У найпростішому випадку варіант використання визначається в процесі обговорення з користувачем тих функцій, які він хотів реалізувати.

Ці діаграми є основою для досягнення взаєморозуміння між програмістами-професіоналами, які розробляють проект, і замовниками проекту.

Метою побудови діаграми варіантів використання є визначення повного сценарію або окремої частини поведінки сутності не беручи до уваги її внутрішню будову.

Діаграма класів є центральною ланкою методології об'єктно-орієнтованого аналізу і проектування. Вона показує класи і їхні зв’язки, тим самим представляючи логічний аспект проекту.

На стадії аналізу діаграми класів використовуються, щоб виділити загальні ролі і обов'язки об'єктів (сутностей), що забезпечують необхідну

поведінку системи, на стадії проектування - щоб передати структуру класів, які формують архітектуру системи.

Діаграма класів визначає ієрархію об'єктів системи і різні статистичні зв'язку, які існує між ними [11].

Діаграми послідовності визначають часову послідовність порядок, вид і тип переданих повідомлень, які мають місце в рамках варіанту використання.

На діаграмі послідовності взаємодія зображується у вигляді двовимірної схеми: вертикальна (час) і горизонтальна (об'єкти, які беруть участь у взаємодії). Істотною є тільки послідовність повідомлень, проте часова вісь може служити реальною метрикою виміру активності об'єкта.

- рекурсивне повідомлення - повідомлення самому собі.

Діаграми стану (Statechart) є засобом опису поведінки (статичних станів) систем. Вони визначають всі відомі стани, в яких може перебувати об'єкт, а також процес зміни стану об'єкта в результаті впливу деяких подій.

У поведінці об'єкта в системі можна виділити події, які відображаються переходами, і діяльності, які відображаються станами. Події пов'язані з переходами і розглядаються як миттєві і неперервні.

Діяльності пов'язані з станами і можуть тривати досить довго. Діяльність може бути перервана в результаті настання деякої події.

Подія - це те, що викликає перехід з одного стану в інший.

## 2.2 Технології проектування бази даних

Інформація, яка необхідна для вирішення задач в ІС акумулюється, оброблюється та збергіається у базах даних.

На основі створених моделей будується логічна модель бази даних.

Логічне моделювання даних проводиться з використанням одного з CASE-засобів «Allfusion ErWin Data Modeler» [12]. Даний засіб дозволяє проводити логічне і фізичне моделювання структури даних систем з використанням ER-діаграм (ER, «Entity-Relationship» – «сутність-зв’язок») у нотації стандарту IDEF1X и нотації Мартіна;

Логічна структура бази даних описує всі її об'єкти, їх поведінку і взаємодію один з одним [13].

Фізичне моделювання може проводитися і у засобі «Allfusion ErWin Data Modeler» формуючи фізичну модель на основі логічної схеми БД, але більш зручним є використання СУБД MySQL Workbench.

Фізична структура бази даних описує кількість файлів даних і журналу транзакцій, з яких складається база даних, їх первинний і поточний розмір, положення на диску, ім'я, розширення, крок збільшення і деякі інші параметри. Ці параметри необхідні тільки для правильного сприйняття сервером бази даних. Для користувачів, що працюють з базою даних, в переважній більшості випадків її фізична структура не має значення.

Якщо на фізичному рівні розглядаються структури, використовувані для зберігання різної інформації, то на логічному рівні необхідно розглядати об'єкти, які можна створювати в базі даних, а також різні властивості, які впливають на роботу сервера з базою даних. Під об'єктами тут розуміється не тільки власне об'єкт, яким є таблиця, уявлення, збережена процедура, але також і користувачі, ролі, повнотекстові каталоги. До логічного рівня відносяться і права доступу користувачів і ролей бази даних до створених в ній об'єктів.

## 2.3 Технології для реалізації інтерфейсу розроблюваної системи

Розробка інтерфейсу клієнтської частини системи буде відбуватися на основі Razor уявлень із застосуванням мови тегів для написання гіпертекстових документів html.

Razor - це назва механізму візуалізації, який був введений Microsoft в версії MVC 3 і перероблений у версії MVC 4.

Механізм візуалізації обробляє контент ASP.NET і шукає інструкції, які зазвичай вставляють динамічний контент в висновок, що відправляється браузеру. У Microsoft підтримуються два механізму візуалізації: механізм ASPX, що працює з дескрипторами <% і%>, які були основною опорою розробки ASP.NET протягом багатьох років, і механізм Razor, що має справу з областями контенту, які позначені за допомогою символу @ [14].

## 2.4 Технології для реалізації інформаційної системи

Для реалізації веб додатку аналізу даних у системі електронної комерції було вирішено використовувати мову програмування C# із застосуванням шаблону розробку веб додатків MVC (Model-View-Controller).

Суть цього шаблону полягає у чіткому розмежуванні відповідальності за різні частини системи і їх функції між трьома компонентами: моделлю, уявленнями і контролерами [15].

Моделі відповідають за верифікацію даних, їх передачу між уявленнями і контролерами. Також модель формує логіку системи: те як ми працюємо і оброблюємо дані.

Представлення забезпечують представлення отриманих даних у виді зрозумілому користувачу. Це може бути просто шаблон який відображує дані, а може бути код який іще і оброблює дані перед їх демонстрацією.

Контролери відповідають за роботу усього додатку управляючи запитами користувачів і визиваючи відповідні методи і функції.

## 2.5 Технології аналізу даних у системі

Основною задачею кваліфікаційної роботи є дослідження методів аналізу даних у системах електронної комерції у вигляді реалізації методів прогнозування обсягів продажів.

Для реалізації веб-застосунку із заданими функціями було вирішено використовувати найбільш оптимальний варіант, який задовольняє потребам, а саме безкоштовну бібліотеку машинного навчання від Microsoft – ML.NET.

ML.NET – це бібліотека машинного навчання (ML), розроблена для розширення екосистеми .NET за допомогою можливостей машинного навчання. Ця бібліотека забезпечує просту та зрозумілу інтеграцію з існуючими моделями машинного навчання і пропонує велику кількість інструментів для створення необхідних моделей.

Машинне навчання — це велика галузь досліджень, яка використовує штучний інтелект. У той час, коли розвиток штучного інтелекту загалом спрямований на те, щоб дати комп’ютерам можливість вдавати людські здібності, завдання ML – дозволити комп’ютерам краще поратися з певними

завданнями, при отриманні більшої матеріалу (даних) для аналізу певних можливостей. Так, комп’ютер «вчиться», в процесі виконання поставлених завдань [16].

Центральною частиною ML.NET є модель машинного навчання. Модель визначає кроки, необхідні для перетворення вхідних даних у прогнозовані дані. За допомогою ML.NET можна навчати спеціальну модель, вказавши алгоритм, або імпортувати попередньо навчені моделі TensorFlow і ONNX.

При наявності моделі, існує можливість додавати її до програми, щоб робити прогнози.

# 3. ПОСТАНОВКА ЗАДАЧІ

## 3.1 Призначення розробки обраної інформаційної системи

Постановку задачі на розробку методи аналізу даних сформулюємо на прикладі конкретної системи електронної комерції, а саме інтернет магазину продажу відеоігор.

Підприємство займається зберіганням, реалізацією та обліком відеоігор у вигляді електронних цифрових ключів. Цифровий ключ - це серійний номер, комбінація з 13, 15, 18, або 25 букв і цифр, який отримує покупець після успішної оплати продукту.

Задачами підприємства є:

- забезпечення користувачів і клієнтів необхідним інструментарієм для перегляду і вибору товарів та оформлення замовлення;
- впровадження та реалізація маркетингових засобів, стратегій які мають за мету збільшення кількості повторних звернень, середньої суми замовлення, рекомендацій;
- отримання детальної інформації щодо прогнозів продажів товарів, розрахованих із застосуванням методів аналізу даних;
- отримання прибутку від реалізації товарів.

Функціями підприємства є:

- надання списку товарів із їх описом та ціною;
- фільтрування товарів за категоріями;
- оформлення замовлення;
- надання програми лояльності.

До організаційної структури входить відділ продажу. Задачі і функції даного підрозділу:

- організація облік продажу товару;
- відслідковування темпів продажів;
- збільшення прибутків;
- введення промоакцій, знижок, персоналізованих пропозицій.

Для визначення області діяльності яка підлягає автоматизації роботи необхідно провести аналіз трудомісткості робіт в підприємстві, а саме у його підрозділах.

Основні бізнес функції реалізуються відділом продажів, тому інформаційну систему доцільно реалізувати у вигляді вебресурсу який буде мати ті ж самі функції і виконувати ті ж самі задачі.

Така інформаційна система спростить взаємодію клієнтів і самого підприємства, автоматизує процеси обліку і реалізації товарів та полегшить процедуру формування звітів з продажу, які у майбутньому можуть використовуватися для аналізу стратегій розвитку підприємства.

Щодо функції аналізу даних, то вона має бути реалізована на окремій сторінці.

Розглянемо одну із задач аналізу даних, а саме прогнозування продажів з такими вхідними даними.

Кожна гра описується кортежом параметрів

$$G_i = < A, K_m, C_p\ > ,$$

де А – атрибути гри які є її описом (назва, виробник);

$K_m$ – жанр до якого гра відноситься (категорія), $m = \overline{1, M}$;

$$K_m = \{G_i,\ i = \overline{1, I_m}\};$$

$C_p$ – діапазон вартостей гри, $p = \overline{1, P}$, наприклад не дорогі, середньої вартості, дорогі.

Інформація щодо продажів описується кортежом параметрів

$$Q_j = < D_j, K_m, C_p, n_j\ >,$$

де $D_j$ – дата продажів;

$n_j$ – кількість проданих копій у $j$ день.

З метою прогнозування продажів на фіксований період часу, необхідно дослідити методи прогнозування даних у вигляді різних моделей регресії і моделі часових рядів, порівняти їх та імплементувати їх у систему електронної комерції для генерації звітів прогнозу.

Отже, інформаційна система розроблюється для прогнозування обсягів продажу та аналізу даних, отриманих під час процесів продажу товарів або послуг, і демонстрації результатів цього аналізу власнику. На основі цих даних власник може приймати рішення щодо подальшого розвитку системи.

## 3.2 Мета впровадження методів аналізу даних у систему електронної комерції

Метою створення моделей прогнозування з боку власника бізнесу є:

1. Отримання інформації щодо можливих аномалій у продажах для планування закупівлі товарів.
2. Отримання інформації про зміні у кількості можливих продажів при зміні цінових діапазонів.
3. Спрощення процесів звітності і відстежування результатів роботи системи шляхом переносу звітності з паперових носіїв на електронні.
4. Збільшення нових користувачі шляхом передбачення їх вподобань.
5. Аналіз даних роботи системи для подальшого формування стратегії розвитку бізнесу.

## 3.3 Функціональні можливості розроблюваної системи

Основними функціональними можливостями системи є:

1. Перегляд результатів прогнозу на певний період часу використовуючи модель регресії по всім категоріям і ціновим діапазонам.
2. Перегляд результатів прогнозу на певний період часу використовуючи модель часового ряду по всім категоріям і ціновим діапазонам.
3. Перегляд результатів прогнозу на певний період часу використовуючи обидві моделі прогнозування за певною категорією і по всім ціновим діапазонам.
4. Перегляд результатів прогнозу на певний період часу використовуючи обидві моделі прогнозування по всім категоріям і за певним ціновим діапазонам.

5. Перегляд результатів прогнозу на певний період часу використовуючи обидві моделі прогнозування за певною категорією і за певним ціновим діапазонам.

## 3.4 Основні не функціональні вимоги до системи

Не функціональними вимогами до розроблюваної системи є:

### 3.4.1 Інтерфейс користувача:

1. Інтерфейс має бути інтуїтивно зрозумілим.
2. Інтерфейси усіх основних сторінок повинні відображатися при мінімальному співвідношенні 1024 х 768, для того щоб система адекватно працювала на усіх, навіть не дуже сучасних приладах, для збільшення охоплення користувачів.
3. Інтерфейс головної сторінки повинен постійно оновлюватися підтримуючи актуальну інформацію про наявність товарів і вміст кошика клієнта.

### 3.4.2 Вимоги до браузерів.

Система повинна працювати на усіх основних браузерах сьогодення: Google Chrome версії 83, Opera версії 66, Mozila Firefox версії 76, Safari версії 13.0, Microsoft Edge версії 81.0.

### 3.4.3 Вимоги до продуктивності:

1. Система повинна стабільно працювати із глибиною історії не менше року, адже для нормальної роботи системи рекомендацій та побудови більш менш адекватної моделі прогнозування необхідна велика кількість даних про замовлення.
2. Аналізуючи навантаження схожих систем електронної комерції можна зробити висновок, що розроблюваний веб додаток має витримувати навантаження не менше ніж 300 користувачів одночасно;
3. Система має відображати форми інтерфейсу користувачів не більше ніж за 1-2 секунди. Більший час може відлякати клієнтів.

4. База даних має бути спроектована таким чином, аби різні система витримувала навантаження великої кількості користувачів і водночас не сповільнювався процес пошуку даних у ній.

5. Система має аналізувати дані отримані під час роботи системи після кожного виконаного замовлення і підтримувати актуальну інформацію про результати аналізу даних у БД.

## 3.5 Основні вимоги до безпеки

Вимоги до безпеки системи:

1. Доступ до БД має бути організовано за допомогою системи авторизації.

2. Адміністратор не має прямого доступу до БД.

3. Користувач який не має статусу адміністратора не може перейти у панель адміністратора.

4. Результати моделювання не повинні завантажуватись у систему напряму і зберігатися там, а лише використовувати дані про замовлення які зберігаються у БД

# 4 МАТЕМАТИЧНИЙ ОПИС АЛГОРИТМІВ ПРОГНОЗУВАННЯ

## 4.1 Алгоритм методу SDCA

Ми розглядаємо таку загальну задачу оптимізації, пов'язану з регуляризованою мінімізацією втрат лінійних предикторів [17]:

Нехай $x_1, \ldots, x_n$ — вектори в двомірному просторі, нехай $\varphi_1, \ldots, \varphi_n$ — послідовність скалярних опуклих функцій, $\lambda > 0$ — параметр регуляризаціїі і нехай $w_1, \ldots, w_n$ – набір коефіцієнтів регресії. Мета — знайти $\min_w P(w)$, де:

$$\mathrm{P}(w) = \left[\frac{1}{n}\sum_{i=1}^{n}\varphi_i(w^T x_i) + \frac{\lambda}{2}\|w\|^2\right]. \quad (4.1)$$

де у якості $\varphi_i(a)$ може бути застосована:

$$\varphi_i(a) = \log\left(1 + \exp(-y_i a)\right),$$

для отримання логістичної регресії;

$$\varphi_i(a) = (a - y_i)^2,$$

для отримання гребеневої регресії;

$$\varphi_i(a) = |a - y_i|,$$

для отримання регресії з абсолютним значенням.

Простим підходом для вирішення цієї задачі задачіє стохастичний градієнтний спуск (SGD). Час виконання алгоритму не залежить від n і тому є сприятливим, коли n дуже велике. Однак підхід SGD має кілька недоліків:

- він не має чіткої умови зупинки;
- він має тенденцію бути занадто агресивним на початку процесу оптимізації, особливо коли параметр λ дуже малий;
- в той час як SGD досягає помірної точності досить швидко, його зближення стає досить повільним, коли нас цікавить більш точні рішення.

Альтернативним підходом є двокоординатне сходження (DCA), яке вирішує задачу двоїстості формули (4.1):

Зокрема, для кожного і нехай $\varphi_i^*$ є опукла спряжена $\varphi_i$, а саме $\varphi_i^*(u) = max_z(zu - \varphi_i(z))$. Проблема двоїстості набуває вигляду $\max\limits_{\alpha} D(\alpha)$, де:

$$D(\alpha) = \left[\frac{1}{n}\sum_{i=1}^{n} -\varphi_i(-\alpha_i) - \frac{\lambda}{2}\left\|\frac{1}{\lambda n}\sum_{i=1}^{n}\alpha_i x_i\right\|^2\right]. \quad (4.2)$$

Двоїсте цільове значення у формулі (4.2) має різну подвійну змінну, пов'язану з кожним прикладом у навчальному наборі даних. На кожній ітерації DCA двоїсте цільове значення оптимізується щодо однієї двоїстої змінної, тоді як решта двоїстих змінних зберігається в такті.

Якщо визначити:

$$\mathrm{w}(\alpha) = \frac{1}{\lambda n}\sum_{i=1}^{n}\alpha_i x_i,$$

то відомо, що $\mathrm{w}(\alpha^*) = \mathrm{w}^*$, де α – оптимальний розв'язок формули (4.2).

Ми зосередимось на стохастичній версії DCA, скорочено SDCA, в якій під час кожного раунду ми вибираємо, яку подвійну координату випадково оптимізувати.

Алгоритм роботи метода SDCA:

1. Ініціалізувати $\mathrm{w}^{(0)} = w\left(\alpha^{(0)}\right)$.
2. Запустити цикл довжиною у *t=1,2,3,…,T* , де T - кількість ітерацій, $T_0$ – випадкове значення від 1 до T, найкраще T/2:

   Визначити $\Delta\alpha_i$ яке б максимізувало

   $$-\varphi_i^*(-\alpha_i^{(t-1)} + \Delta\alpha_i)) - \frac{\lambda \mathrm{n}}{2}\left\|w^{(t-1)} + (\lambda \mathrm{n})^{-1}\Delta\alpha_i x_i\right\|^2;$$

   $$\alpha^{(t)} = \alpha^{(t-1)} + \Delta\alpha_i e_i;$$

   $$w^{(t)} = w^{(t-1)} + (\lambda \mathrm{n})^{-1}\Delta\alpha_i x_i.$$

3. Обчислити середнє значення:

$$\overline{\alpha} = \frac{1}{T - T_0} \sum_{i=T_0+1}^{T} \alpha^{(t-1)};$$

$$\overline{w} = w(\overline{\alpha}) = \frac{1}{T - T_0} \sum_{i=T_0+1}^{T} w^{(t-1)}.$$

4. Обчислити випадкове значення:

$$\overline{\alpha} = \alpha^{(t)} \text{ і } \overline{w} = w^{(t)} \text{ де } t \in T_0 + 1, \dots, T.$$

5. На вихід подати $\overline{w}$.

4.2 Алгоритм методу регресії Пуассона

Розподіл Пуассона моделює ймовірність подій *y* (тобто невдачі, смерті, продажі т.д) за формулою [18]:

$$\Pr(Y = y | \mu, t) = \frac{e^{-\mu}\mu^{y}}{y!}, y = 0,1, \dots$$

Необхідно зазначити, що розподіл Пуассона задається одним параметром $\mu$. Це середня частота рідкісної події на одиницю експозиції. Експозицією може бути час, простір, відстань, площа, обсяг або чисельність. Оскільки експозиція часто є періодом часу, для позначення експозиції використовується символ t. Якщо значення експозиції не вказано, вважається, що воно дорівнює часу.

Параметр μ можна інтерпретувати як ризик нового виникнення події протягом певного періоду експозиції *t*. Ймовірність подій *y* тоді визначається як:

$$\Pr(Y = y|\,\mu, t) = \frac{e^{-\mu t}(\mu t)^y}{y!}, y = 0,1, \ldots$$

Розподіл Пуассона має властивість, що його середнє значення і дисперсія рівні.

У регресії Пуассона припускається, що частота впливу Пуассона μ визначається набором k регресора змінних X. Вираз, що зв'язує ці величини:

$$\mu = t \exp\left(\beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k\right)$$

Часто $X_1 \equiv 1$ і $\beta_1$ називають перерізом. Коефіцієнти регресії $\beta_1, \beta_{2,}, \ldots, \beta_k$, є невідомими параметрами, які отримуються з набору даних. Їхні значення позначені як $b_1, b_{2,}, \ldots, b_k$.

Використовуючи ці позначення, фундаментальна модель регресії Пуассона для спостереження $i$-ого значення записується як:

$$\Pr(Y_i = y_i|\,\mu_i, t_i) = \frac{e^{-\mu_i t_i}(\mu_i t_i)^{y_i}}{y_i!}, \text{де } \mu_i = t_i \mu\left(\grave{x_i}\beta\right)$$
$$= t_i \exp\left(\beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_k X_{ki}\right)$$

Тобто для заданого набору значень змінних регресора результат відповідає розподілу Пуассона.

Коефіцієнти регресії оцінюються за допомогою методу максимальної правдоподібності. Логарифм функції імовірності:

$$\ln[L(y, \beta)] = \sum_{i=1}^{n} y_i \ln\left[t_i \mu\left(\grave{x_i}\beta\right)\right] - \sum_{i=1}^{n} t_i \mu\left(\grave{x_i}\beta\right) - \sum_{i=1}^{n} \ln\left(y_i!\right)$$

Відзначмо, що деякі пакети моделювання ігнорують останній елемент формули, оскільки він не включає параметри регресії.

Рівняння правдоподібності можуть бути сформовані, взявши похідні щодо кожного коефіцієнта регресії та встановивши результат рівним нулю. Це призводить до набору нелінійних рівнянь, які не допускають розв’язку в закритій формі.

Таким чином, необхідно використовувати ітераційний алгоритм для знаходження набору коефіцієнтів регресії, що максимізують логарифмічну ймовірність. Використовуючи метод ітераційнозважених найменших квадратів, рішення можна знайти за п’ять або шість ітерацій. Однак алгоритм вимагає повного проходження даних на кожній ітерації, тому він відносно повільний для задач з великою кількістю рядків. З сучасними комп’ютерами це стає все меншою проблемою.

Застосовуючи звичайну теорію максимальної правдоподібності, асимптотичний розподіл оцінок максимальної правдоподібності (MLE) є багатовимірною нормою. Тобто:

$$\hat{\beta} \sim N\left(\beta, \beta V_{\hat{\beta}}\right), \text{де}$$

$$V_{\hat{\beta}} = \left(\sum_{i=1}^{n} \mu_i x_i x_i^{`}\right)^{-1}$$

Відомо, що в моделі Пуассона середнє значення та дисперсія рівні. На практиці, дані рідко відповідають цьому правилу. Зазвичай дисперсія більша за середнє — ця ситуація називається наддисперсією. Збільшення дисперсії представлено в моделі постійним кратним дисперсійно-коваріаційної матриці φ. Тобто:

$$V_{\hat{\beta}} = \varphi\left(\sum_{i=1}^{n} \mu_i x_i x_i^{`}\right)^{-1}, \text{де}$$

$$\hat{\varphi} = \frac{1}{n-k} \sum_{i=1}^{n} \frac{(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}$$

## 4.3 Алгоритм методу дерев регресії

Евристика відіграє вирішальну роль при побудові дерев рішень у визначенні як ефективності класифікації, так і обчислювальної вартості. Більшість сучасних алгоритмів навчання дерева рішень використовують евристику на основі (не)чистості, яка по суті вимірює чистоту результуючих

підмножин після застосування атрибута розщеплення для розділення навчальних даних. Прибуток інформації, визначений таким чином, широко використовується в якості стандартної евристики:

$$IG(S,X) = Ent(S) - \sum_x \frac{|S_x|}{|S|} Ent(S_x) \quad (4.3)$$

де S – набір даних навчання,

X – набір атрибутів, x – їхні значення,

$S_x$ - є підмножиною S, що складається з екземплярів X=*x*,

$$Ent(S) = -\sum_{i=1}^{|C|} P_S(c_i) \log P_S(c_i) ,$$

де $P_S(c_i)$ - оцінюється за відсотком належності екземплярів $c_i$ у $S$,

|C| - кількість класів.

Необхідно зазначити, що побудова дерев — це рекурсивний процес розбиття навчальних даних, а S — навчальні дані, пов'язані з поточним вузлом. Тоді $P_S(c_i)$ насправді є $P_S(c_i|x_p)$ для всіх навчальних даних, де $X_p$ — це набір атрибутів уздовж шляху від поточного вузла до кореня, який називається атрибутами шляху, а $x_p$— це призначення значень до змінних в $X_p$.

У процесі побудови дерева кожен атрибут-кандидат (атрибути, яких немає в $X_p$) досліджується за допомогою Формули 1, і атрибут з найбільшим прибутком інформації вибирається як атрибут розщеплення. Найбільш трудомісткою частиною цього процесу є оцінка є $P_S(c_i|x_p, x)$ для обчислення $Ent(S_x)$. Він повинен пройти через кожен екземпляр в $S_x$, для кожного з яких він перебирає кожен атрибут- кандидат X.

Алгоритм побудови швидкого дерева рішень FT(П,S), де П - набір атрибутів-кандидатів, S - набір позначених екземплярів [19]:

1. Якщо (S чистий або порожній) або (П порожній), повернути T.

2. Обчислити $P_S(c_i)$ на S для кожного класу $c_i$.

3. Для кожного атрибута X в П, обчислити IG(S, X) на основі формули (4.3)

4. Використовуючи атрибут $X_{max}$ з найвищим IG для кореня розбити S на непересічні підмножини $S_x$.

6. Для всіх значень x із $X_{max}$ обчислити $T_x$ = FT(П-$X_{max}$, $S_x$) і додати $T_x$ як дочірній елемент $X_{max}$.

7. Повернути T.

## 4.4 Алгоритм методу прогнозування часових рядів

Сингулярний спектральний аналіз (SSA) – це метод аналізу та прогнозування часових рядів, що поєднує в собі елементи класичного аналізу часових рядів, багатовимірної статистики, багатовимірної геометрії, динамічних систем та обробки сигналів [20].

SSA має на меті розкласти вихідний ряд на суму невеликої кількості інтерпретованих компонентів, таких як повільно змінний тренд, коливальні компоненти та «безструктурний» шум. Він заснований на розкладанні за сингулярними значеннями конкретної матриці, побудованої за часовими рядами. Для часового ряду не потрібно припускати ні параметричну модель, ні умови типу стаціонарності; це робить SSA технікою без моделі.

Нехай $x_1, x_{2,}, \dots, x_N$ — часовий довжини N. Для довжини вікна L ($1 < L < N$) будуються вектори з відставанням $X_i = (x_i \dots, x_{i+L-1})^T$, i = 1, 2, …, K =N−L+1, і розміщуються в матрицю X = $(x_{i+j-1})_{i,j=1}^{L,K}$ = [ $X_1, \dots, X_K$ ]. Ця матриця має розмір L × K і її часто називають «матрицю траєкторії». Вона має вигляд матриці Ганкеля, що свідчить про те, що всі елементи вздовж діагоналі рівні.

Стовпці $X_j$ з X, які розглядаються як вектори, лежать у L-вимірному просторі. Розкладання матриці $X * X^T$ на сингулярні значення дає набір L власних значень і власних векторів. Деяка комбінація певного числа $l < L$ цих власних векторів визначає $l$-вимірний підпростір. L-вимірні дані $\{X_1, \dots, X_K\}$ потім проектуються на цей $l$-вимірний підпростір, і наступне усереднення по діагоналях дає деяку матрицю Ганкеля $\overline{X}$, яка розглядається як наближення до X. Ряд, відновлений з $\overline{X}$, задовольняє деяку лінійну рекурентну формулу, яка може бути використана для прогнозування.

# 5. РОЗРОБКА АРХІТЕКТУРИ ТА КОМПОНЕНТІВ СИСТЕМИ

## 5.1 Розробка системних вимог

Проаналізувавши постановку задачі на розробку системи, можна сказати, що система призначена для прогнозування обсягів продажів системи електронної комерції за допомогою методів регресії і часових рядів.

Виходячи з цього і мети створення даної системи, потрібно сформулювати системні вимоги до розроблюваної системи.

Системні вимоги – це вимоги які висуваються до прикладної програмної системи до середовища її використання. Прикладами таких вимог можуть бути: певна частота процесора, кількість оперативної пам’яті, операційна система на стороні клієнта.

Отже, системні вимоги до вебсайту прогнозування обсягів продажу системи електронної комерції:

а) серверна частина інформаційної системи має бути реалізована у вигляді бази даних, доступ до якої організовується використовуючи системи автентифікації для забезпечення безпеки даних і яка б підтримувала високе навантаження великої кількості підключень. Для цього необхідно використати СУБД;

б) клієнтська частина інформаційної системи має бути реалізована у вигляді вебсторінок написаних на HTML для їх завантаження користувачам;

в) інтерфейс доступу має бути організовано за допомогою завантаження вебсторінок браузером Opera, Mozila, Chrome і т.д.

## 5.2 Розробка моделі потоків даних

Після формулювання системних вимог до системи, наступним кроком є створення діаграми потоків на основі стандарту DFD для отримання уявлення про те, як має виглядати логічна та фізична моделі системи.

На рис. 5.1 представлена концептуальна DFD діаграма, яка показує модель потоків даних у вебсайті автоматизації процесів електронної комерції та прогнозування продажів.

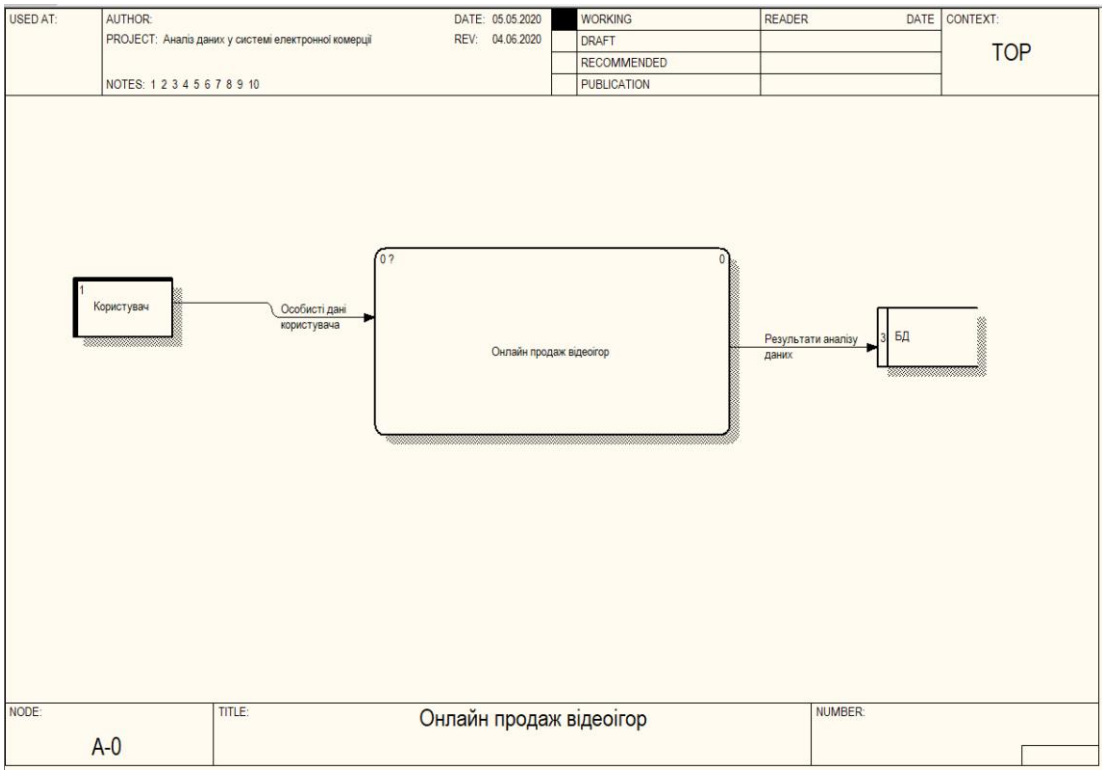



Рис. 5.1 – Концептуальна діаграма потоків даних інформаційної системи.

На ній зображено загальний вид моделі із дугами:

- у якості входу «Особисті дані Користувачів» які передає зовнішня сутність «Користувач»;
- сховищами даних є «База Даних» у яку заносяться про результати аналізу роботи системи;
- зовнішньою сутністю є «Користувач» який передає у систему дані про себе.;
- виходом є звіти про аналіз результатів роботи ІС, що представляють із себе звіти про продажі за певні періоди і записи у базу даних результатів аналізу даних.

На рисунку 5.2 представлена декомпозиція концептуальної діаграми потоків даних, яка уточнює процес роботи моделі потоків даних системи електронної комерції і аналізів даних у ній.

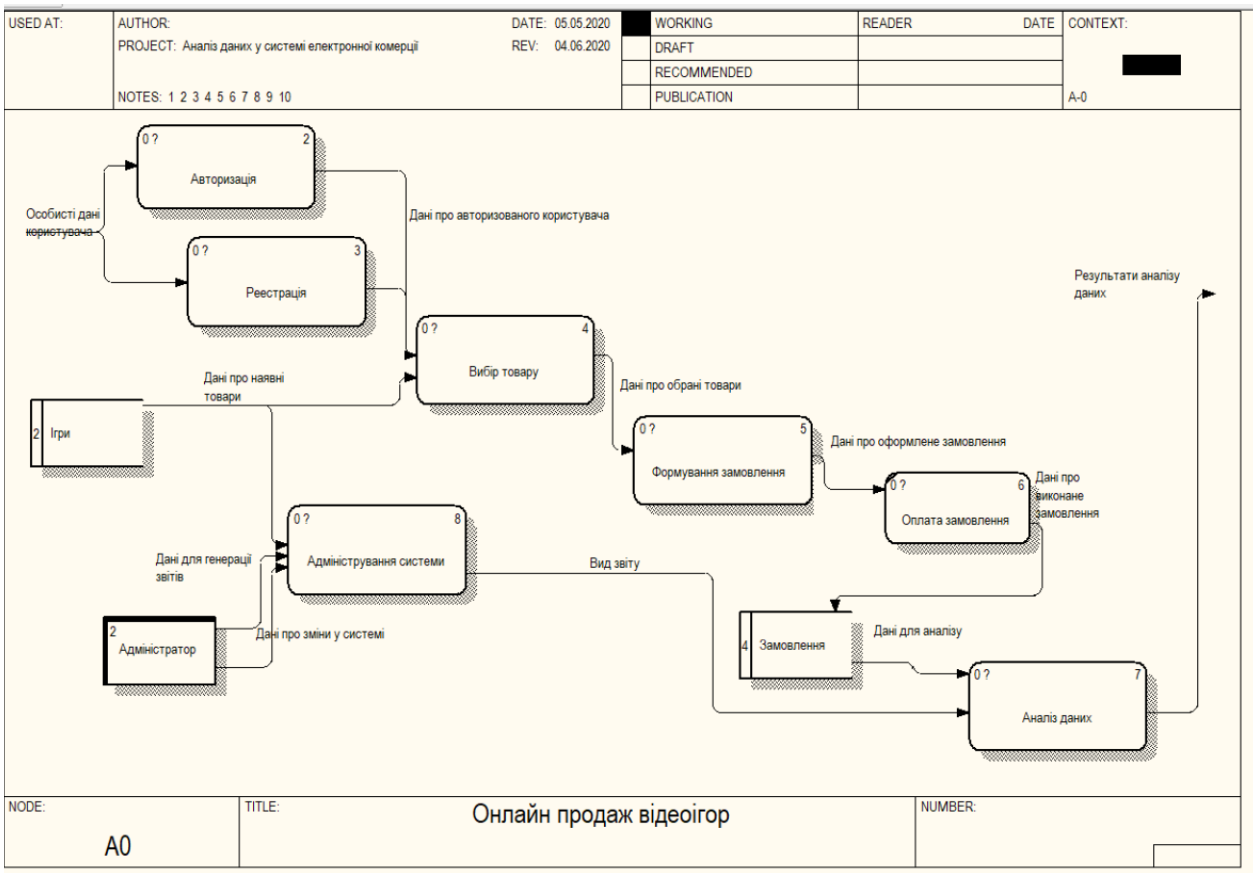



Рис. 5.2 – Декомпозиція концептуальної діаграми потоків даних.

На ній зображено: процеси авторизації, реєстрації, вибору товару, формування замовлення, оплати замовлення, адміністрування системи і аналізу даних; зовнішня сутність «Адміністратор» яка передає дані необхідні для формування звітів про роботу системи і зміни даних про товари; дуги які відповідають даним про авторизованого користувача, даним про обрані товари, даним про сформоване замовлення, даним про виконане замовлення, даним про наявні товари, інформації необхідної для формування звітів, даним про зміни у системі і результати роботи адміністратора.

Аналізуючи дану декомпозицію можна дійти висновку, що для реалізації процесів необхідна сутність «Товари» у базі даних, атрибути якої («productId», «Name», «category», «description», «price», «ImageData») дозволяють отримувати дані із таблиці БД для подальшої роботи із ними.

На рисунку 5.3 представлена декомпозиція процесу «Адміністрування системи». Цей процес необхідний для організації роботи системи та обліку товарів. Він дозволяє адміністраторам редагувати список товарів змінюючи інформацію про окремі товари або додаючи нові і формувати звіти за певні періоди. На діаграмі показані потоки даних які задіяні у роботі цього процесу.

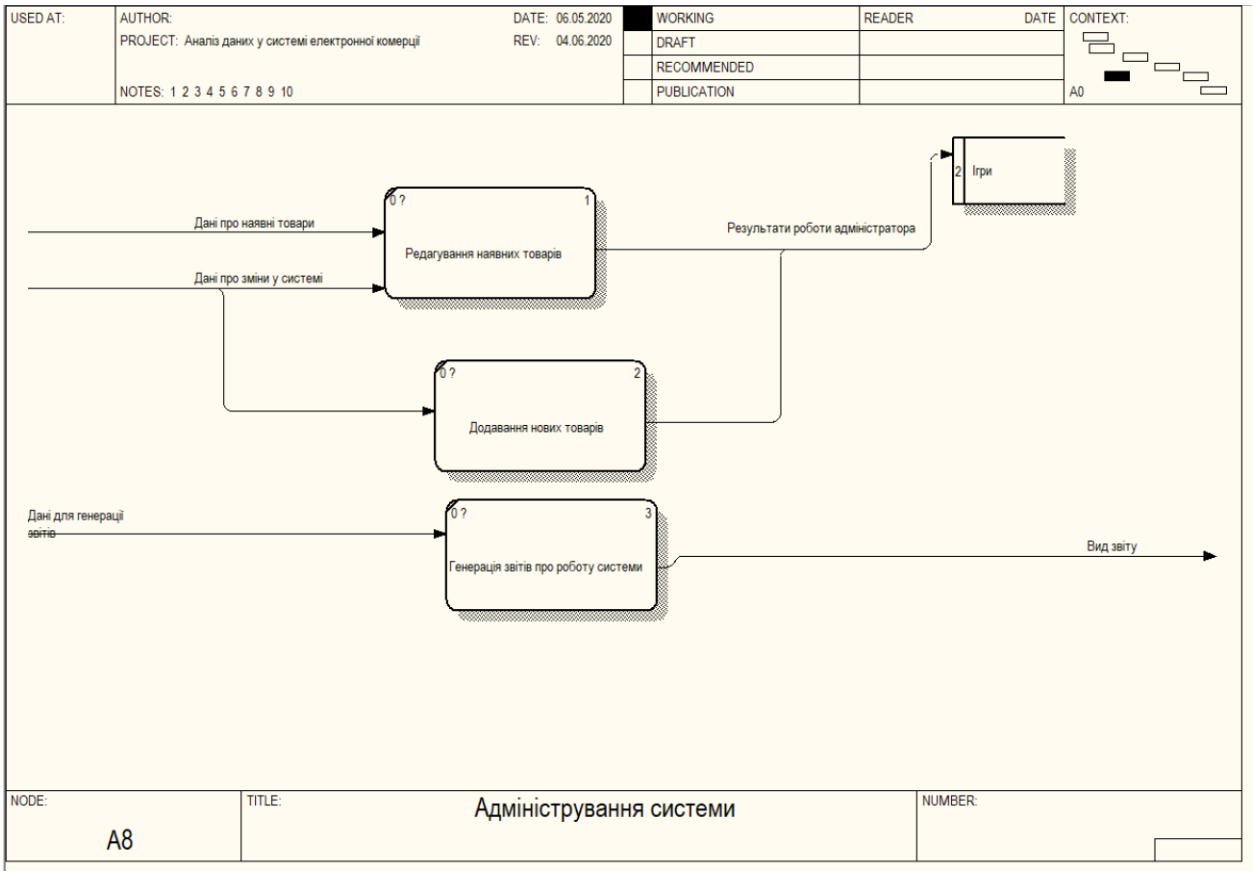



Рис. 5.3 – Декомпозиція процесу «Адміністрування системи».

На ній зображено: функціональні блоки додавання нового товару, редагування інформації про уже існуючі записи про товари і формування звітів по датам; дуги які відповідають даним про результати адміністрування системи, що містять у собі успішні результати редагування бази товарів і звіти про роботу системи.

На рисунку 5.4 представлена декомпозиція процесу «Формування замовлення». Цей процес необхідний для компіляції усіх товарів із корзини користувача, відображення результату користувачу, надання можливості зміни вмісту замовлення, і підтвердження його для наступної обробки.

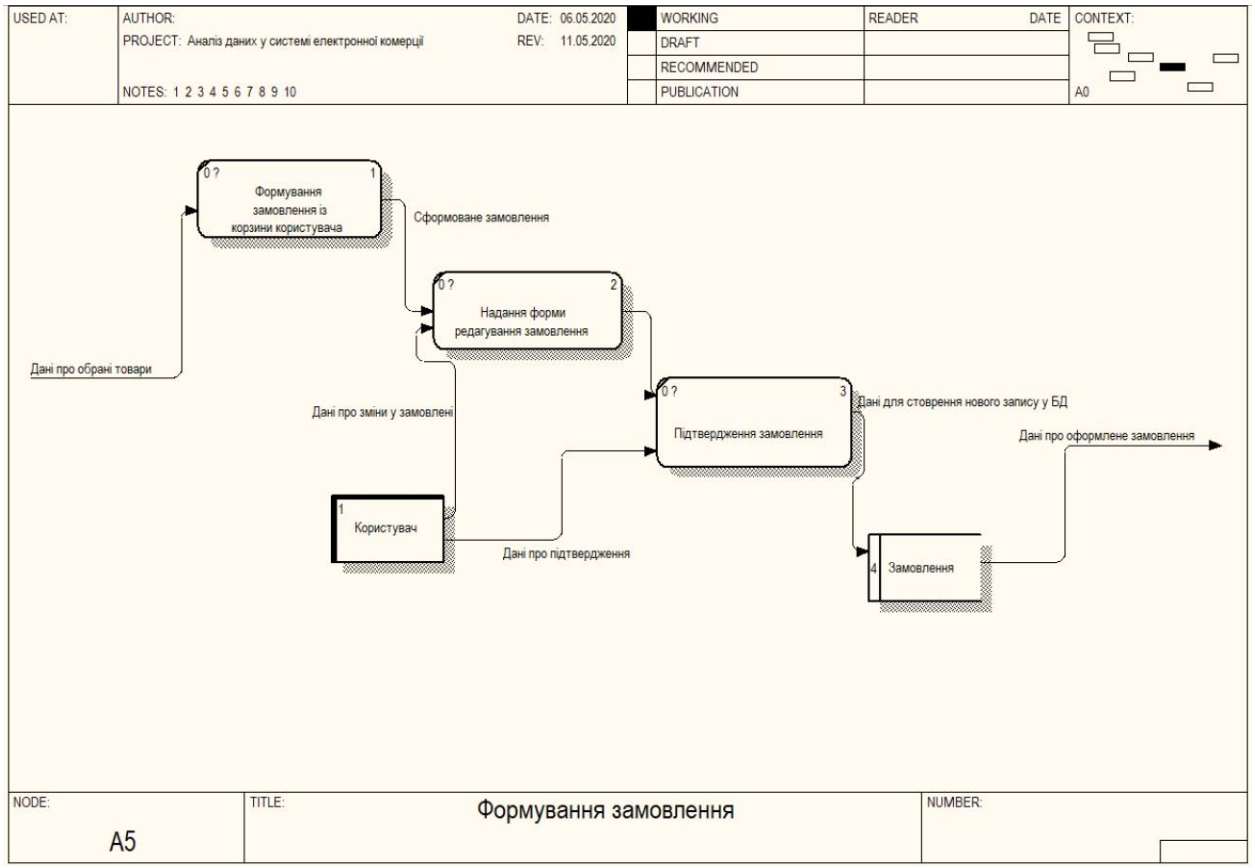



Рис. 5.4 – Декомпозиція процесу «Формування замовлення».

На ній зображено: функціональні блоки формування замовлення із корзини користувача, надання форми редагування даних про замовлення користувачем (видалення товару із корзини), підтвердження замовлення користувачем; дуги які відповідають даним, що містять результати формування замовлення із корзини, дані про відредаговане замовлення, даним про зміни, даним про оформлене замовлення, даним про замовлення які передаються у сховище даних «Замовлення»; зовнішня сутність «Користувач» яке передає дані необхідні для редагування замовлення і його підтвердження; сховище даних «Замовлення» яке відповідає таблиці у базі даних у якій міститься інформація про усі замовлення.

Для реалізації процесу необхідна сутність «Замовлення», яка відповідатиме однойменному сховищу даних. Атрибути цієї сутності ( «OrderID», «fk_user_id», «date», «email», «totalsum») дозволяють отримувати дані із бази даних для подальшої роботи із ними.

На рисунку 5.5 представлена декомпозиція процесу «Аналізу даних». Цей процес необхідний для аналізу результатів роботи і подальшої зміни параметрів системи для покращення якості і потенційного збільшення

кількості замовлень шляхом персоналізованих рекомендацій користувачам і дисконтної системи, а також прогнозування обсягів продажів системи.

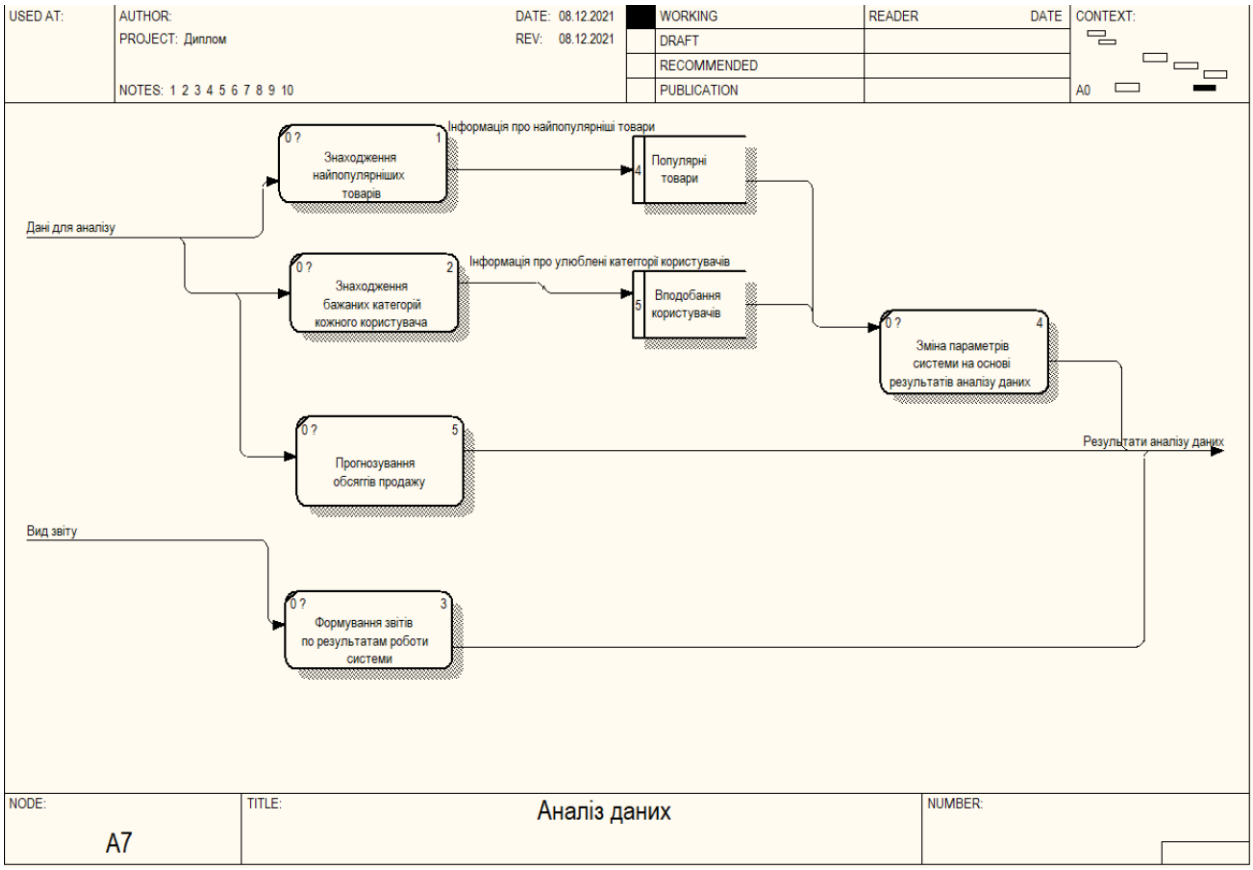



Рис. 5.5 – Декомпозиція процесу «Аналіз даних».

На ній зображено: функціональні блоки знаходження найпопулярніших товарів для виведення їх у списку товарів на головні сторінці, знаходження улюблених категорій товарів для кожного користувача із замовленням для виведення товарів із цих категорій користувачам при наступному використанні системи, зміна параметрів системи на основі результатів аналізу шляхом запису їх у БД; дуги що відповідають даним про найпопулярніші товари і даним про відповідність користувачів до їх улюблених товарів; сховища даних «Популярні товари» яке зберігає інформацію про найпопулярніші товари і «Вподобання користувачі» яка зберігає дані про відповідність користувачів то тих категорій товарів, які вони найчастіше купляють.

Для реалізації процесу необхідні сутності:

- «Популярні товари» атрибути якої «id», «gameId»;
- «Вподобання користувачів» атрибути якої «id»,«userid»,«category»

На рисунку 5.6 представлена декомпозиція процесу «Прогнозування даних». Цей процес необхідний для побудови моделей прогнозування для отримання даних про очікувані обсяги продажів на певний період часу.

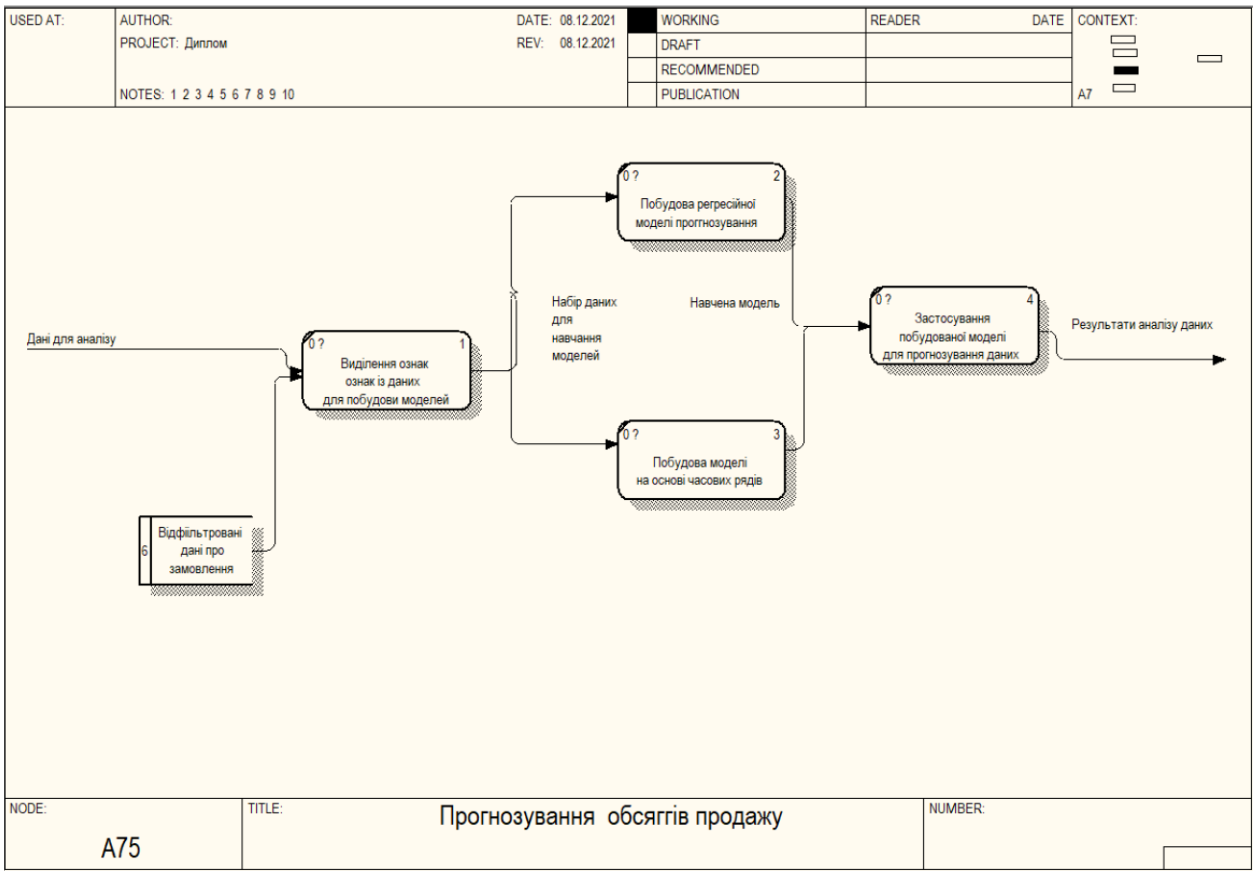



Рис. 5.6 – Декомпозиція процесу «Прогнозування даних».

На ній зображено: процеси виділення ознак з набору даних, побудова моделі прогнозування на основі методу регресії і її навчання, побудова моделі прогнозування на основі методу часових рядів і її навчання, використання побудованих моделей для прогнозування обсягів продажу; дуги що відповідають даним про параметри прогнозування, набір даних для навчання моделей, навчені моделі і результати аналізу даних; сховище даних «Відфільтровані дані про замовлення» яке містить дані про виконані замовлення по датам із інформацією про кількість проданих ігор, їх жанри і ціновий діапазон.

Для реалізації процесу необхідна сутність «Статистика» атрибути якої «Date», «Sales», «Quartal», «Category», «Price»;

В результаті побудови діаграми потоків даних була отримана модель потоків даних розроблюваної системи, яка більш детально показує функціональні вимоги до системи і полегшує процес безпосередньо розробки, шляхом визначення таблиць і їх складу, які потрібні у базі даних.

Після аналізу конкретного прикладу системи електронної комерції можна виділити сутності, які і увійдуть у логічну модель розроблюваної системи. Більш детально вони описані у таблиці 5.1.

Таблиця 5.1 – Сутності логічної моделі даних системи.

| № | Назва сутності | Назва атрибуту | Тип даних (домен) | Призначення |
|---|---|---|---|---|
| 1. | Games | GameId | Лічильник, ціле | Первинний ключ |
| | | Name | Масив символів на 255 байтів | Назва гри |
| | | Category | Масив символів на 255 байтів | Категорія до якої гра відноситься |
| | | Description | Масив символів на 500 байтів | Опис гри |
| | | Price | Десяткове число на 12 символів із 2 символами після коми | Ціна гри |
| | | Quantity | Ціле | Відповідає кількості наявного товару |
| | | Click | Ціле | Лічильник який відстежує кількість переходів на сторінку товару |
| | | AddToCart | Ціле | Лічильник який відстежує |

| | | | | |
|---|---|---|---|---|
| | | | | кількість додавань товару в корзину |
| | | ImageData | Масив бінарних значень на 500 символів | Поле для зберігання даних зображення |
| | | ImageMimeType | Масив символів на 50 байтів | Поле для зберігання даних зображення |
| 2. | Users | UserId | Лічильник, ціле | Первинний ключ |
| | | Name | Масив символів на 255 байтів | Ім’я користувача |
| | | Surname | Масив символів на 255 байтів | Фамілія користувача |
| | | Bday | Дата | Дата народження користувача |
| | | telephone | Масив символів на 255 байтів | Номер телефону |
| | | Login | Масив символів на 255 байтів | Логін для входу на сайт |
| | | Password | Масив символів на 255 байтів | Пароль для входу на сайт |
| | | Discount | Ціле | Розмір знижки |
| | | City | Текст | Місцезнаходження користувача |
| | | AdminStatus | Мале ціле | Статус користувача |
| 3. | Order | OrderID | Лічильник, ціле | Первинний ключ |
| | | Fk_user_id | Ціле | Вторинний ключ |
| | | Date | Дата | Дата оформлення замовлення |
| | | Email | Масив символів на 255 байтів | Електронна пошта куди відправляється замовлення |

| | | | | |
|---|---|---|---|---|
| | | TotalSum | Десяткове число на 16 символів із 2 символами після коми | Загальна сума замовлення. |
| 4. | OTGs | Fk_order_id | Ціле | Вторинний ключ |
| | | Fk_game_id | Ціле | Вторинний ключ |
| | | quantity | Ціле | Кількість замовленого товару |
| | | otgId | Лічильник, ціле | Первинний ключ |
| 5. | PopularGames | Id | Лічильник, ціле | Первинний ключ |
| | | Gameids | Ціле | Номер гри |
| 6. | UserFavs | UserFavId | Лічильник, ціле | Первинний ключ |
| | | Userid | Ціле | Ідентифікатори користувачів |
| | | CategoryName | Масив символів на 255 байтів | Назва категорії |
| 7. | Info | Date | Дата | День роботи системи |
| | | Sales | Цііле | Кількість проданих копій у цей день |
| | | Q | Ціле | Квартал |
| | | Categ | Ціле | Ідентифікатор жанру проданих ігор |
| | | P | Ціле | Ідентифікатор цінового діапазону проданих ігор |

5.3 Діаграма класів розроблюваної системи.

На рисунку 5.7 представлена діаграма класів для розроблюваної системи аналізу даних в системах електронної комерції.

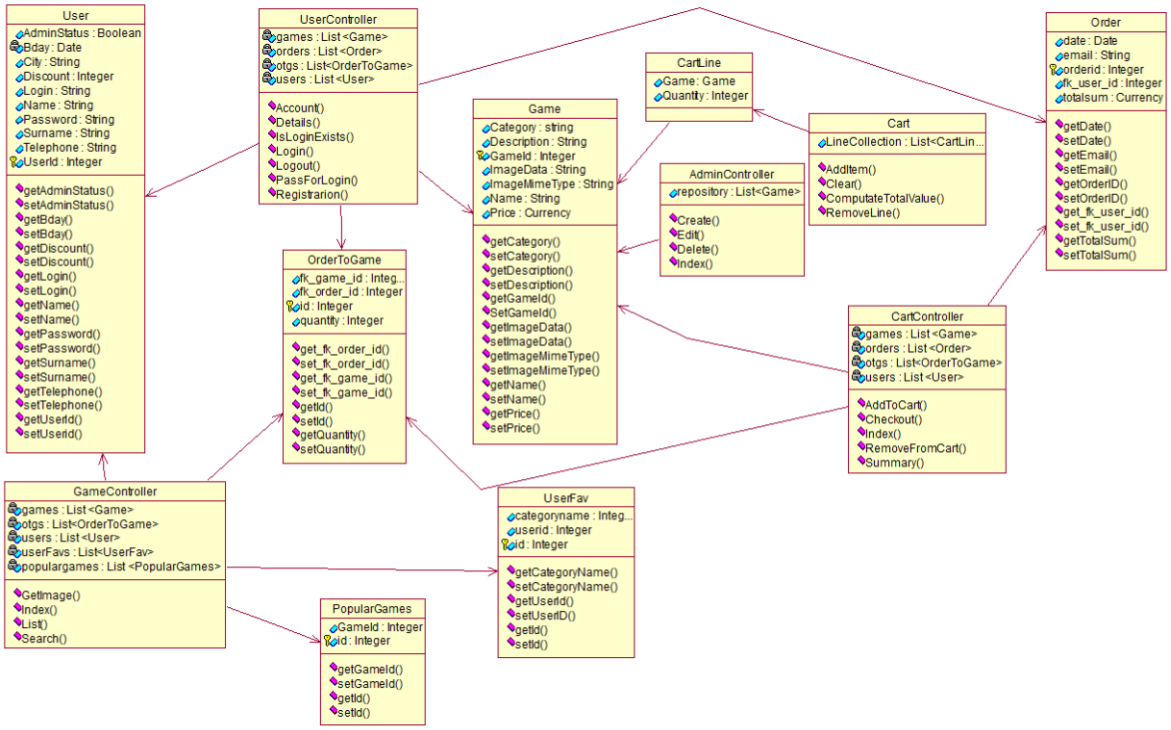



Рис. 5.7 – Діаграма класів ІС

На даній діаграмі зображені класи із їх атрибутами і методами які мають бути реалізовані під час безпосередньо розробки вебдодатку.

Основні класи які є сутностями-представленнями інформації із бази даних це:

- клас User, який відповідає таблиці «Users» у базі даних і містить у собі усю інформацію про користувачів;
- клас Order, який відповідає таблиці «Orders» у базі даних і містить у собі інформацію про усі замовлення оброблені системою;
- клас Game, який відповідає таблиці «Games» у базі даних і містить у собі інформацію про усі товари реалізацією яких займається розроблювана система електронної комерції;
- клас PopularGame, який представляє таблицю «PopularGames» бази даних і містить у собі результати аналізу даних наслідків роботи системи у вигляді переліку найбільш популярних ігор;

- клас UserFav, який відповідає таблиці «UserFavs» бази даних і містить у собі результати аналізу даних наслідків роботи системи у вигляді переліку відповідності користувачів і їх улюблених категорій, тобто категорій, які найчастіше фігурували у їх замовленнях;
- клас OrderToGame, який представляє таблицю «OTGs» бази даних і зберігає результати обробки і виконання замовлення, а саме інформацію про відповідність замовлення до його вмісту, тобто id ігор і їх кількість;
- клас Info, який представляє таблицю «Info» бази даних і зберігає дані про кількість продажів за певний період часу відповідно до жанру проданих ігор і їх цінового діапазону.

Також на діаграмі представленні класи-контролери які регулюють усю роботу системи і реалізують необхідні для функціонування системи бізнес-функції:

а) Клас UserController – регулює роботи системи із сутністю User і реалізує бізнес-функції які пов’язані із нею;
   1) Account() – функція входу в особистий кабінет і завантаження списку оформлених замовлень;
   2) Details() – функція перегляду детальної інформації про замовлення(назви продуктів і їх кількість);
   3) Login() – функція авторизації у системі;
   4) LogOut() – функція завершення сесії і виходу із системи;
   5) Registration() – функція реєстрації у системі;
   6) IsLoginExists() – функція перевірки існування введеного логіну користувачем для під час процесів авторизації чи реєстрації;

б) Клас GameController – відповідає за роботу системи із сутністю товарів які реалізує розроблювана система електронної комерції і втілює бізнес-функції пов’язані із ними;
   1) Index() – функція яка завантажує на вебсторінку детальну інформацію про обраний товар;
   2) List() – функція яка відповідає за бізнес-функцію завантаження сторінки із списком товарів із додатковою можливістю відображення списку за категоріями;

3) Search() – функція пошуку товарів за назвою і відображення списку відповідних товарів;

в) Клас CartController – регулює роботу системи із корзиною і замовленнями, реалізує функції необхідні для роботи;

1) AddToCart() – створює новий екземпляр класу «Cart», який містить інформацію про обраний користувачем товар і його кількість, і додає у нього дані про товари які обрав користувач;

2) Index() – відповідає за бізнес-функцію формування замовлення і завантаження його користувачу для редагування або подальшого підтвердження;
3) RemoveFromCart() – видаляє обраний товар із корзини;
4) Summary() – повертає значення загальної суми замовлення;
5) Checkout() – відповідає за бізнес функцію підтвердження замовлення користувачем і його виконання;

г) Клас AdminController – регулює роботу системи з точки зору адміністратора і реалізує бізнес-функції пов’язані із цим;

1) Index() – завантажує сторінку адміністратора із списком товарів для редагування;
2) Edit() – функція редагування інформації про наявні товари;
3) Create() – функція додавання нового товару до спису товарів;
4) Delete() – функція видалення обраного товару із списку товарів;
5) Predict() – функція прогнозування обсягів продажу на певний період часу.

Також на діаграмі зображені проміжні класи які виступають у ролі буферів між класами-сутностями бази даних і класами-контролерами які регулюють роботу системи:

а) Клас CartLine який є сутністю яка зберігає у собі дані про гру і її кількість, використовується у класі Cart;

б) Клас Cart який є реалізацією концепції корзини у якій зберігається інформація про обрані користувачем ігри і їх кількість. У якості атрибуту має список екземплярів класу CartLine і реалізує функції:

1) Функція AddItem() яка додає нову гру і її кількість у корзину і яку використовує функція AddToCart() класу CartController;

2) Функція Clear() яка повністю очищає корзину після оформлення замовлення;
3) Функція RemoveLine() яка видаляє один елемент із списку екземплярів класу CartLine під час виклику функції RemoveFromCart() класу CartController;
4) Функція ComputateTotalValue() яка обчислює вартість усього замовлення і використовується для підчас виклику функції Summary() класу CartController;

## 5.4 Логічне і фізичне моделювання даних

Логічна модель даних розроблюваної системи представлена на рисунку 5.8.

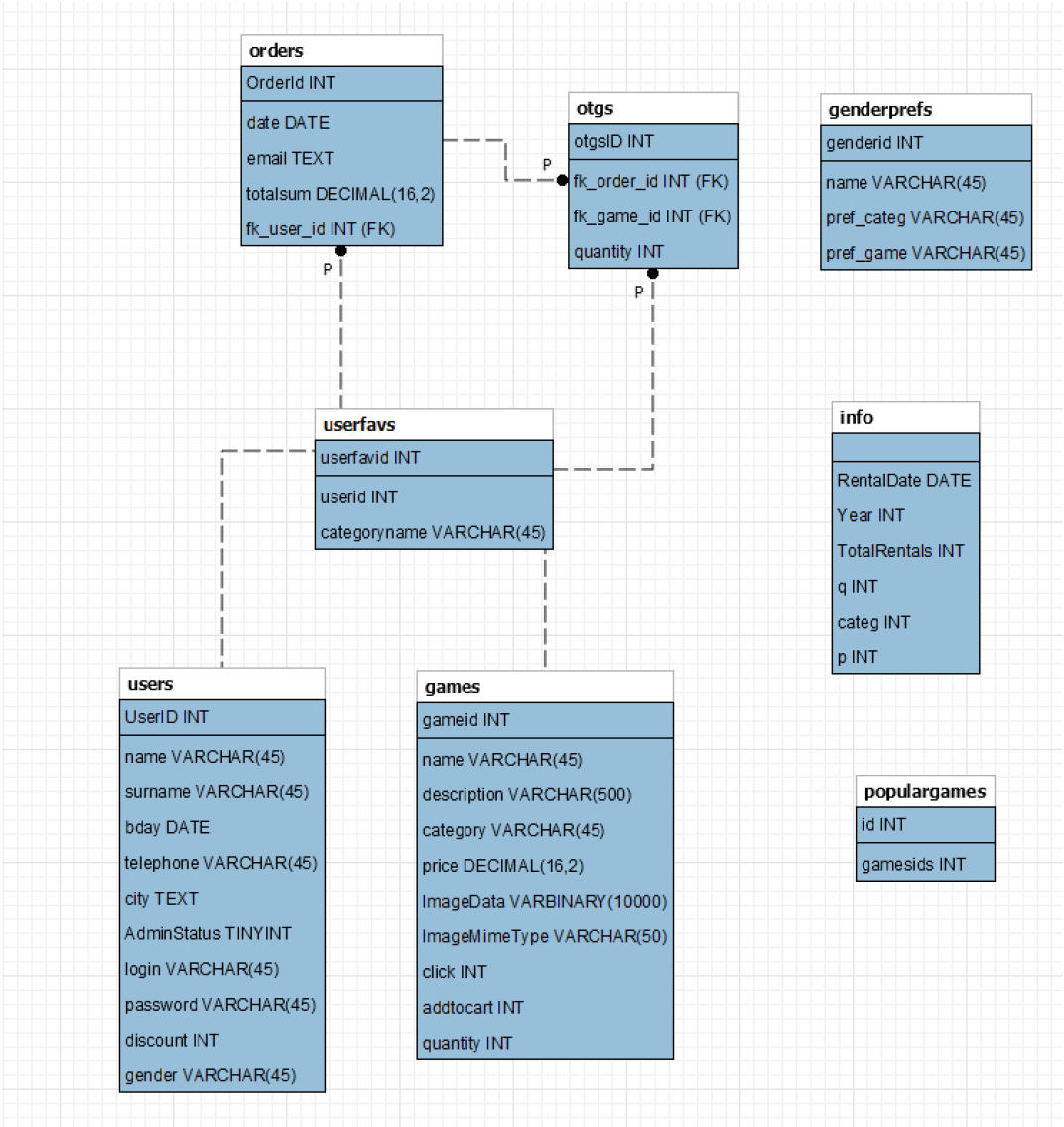



Рис. 5.8 – Логічна модель даних системи.

Наступним етапом проектування бази даних є саме її створення. Для формування фізичної моделі бази даних використовується функціонал програмного додатку MySQL Workbench, де на основі логічної моделі бази даних створюється фізична модель [21].

Тобто сутності змінюються на таблиці, атрибути сутностей на найменування «Поля» таблиць. Також полям додається інформація про типи даних.

В результаті проектування була отримана фізична модель бази даних розроблюваного вебсайту аналізу даних у системах електронної комерції.

Вона зображена на рисунку 5.9.

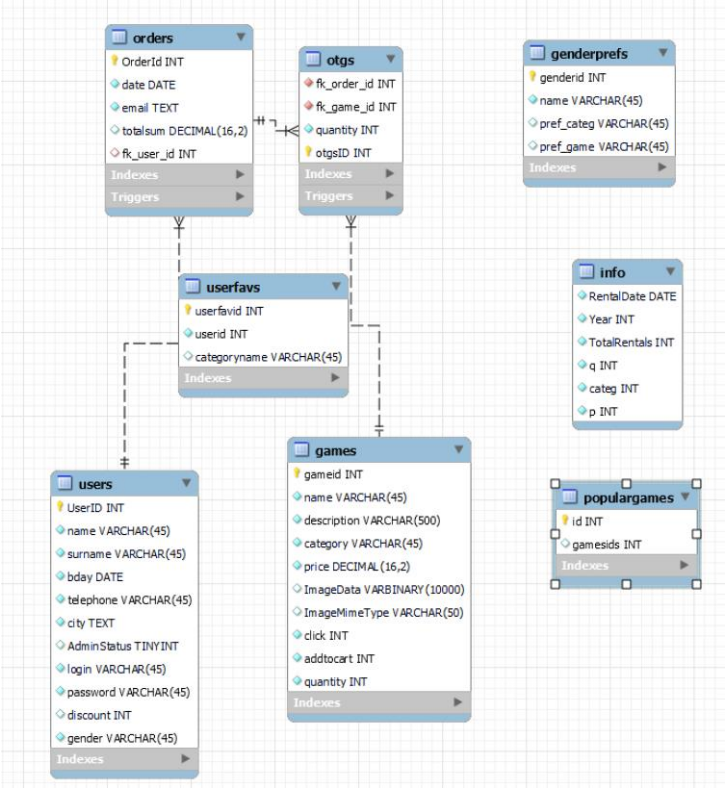



Рисунок 5.9 – Фізична модель бази даних

## 5.6 Розробка моделей прогнозування продажів

Ця проблема зосереджена на прогнозуванні товару на основі попередніх продажів. Щоб вирішити цю проблему, створено дві незалежні моделі машинного навчання, які використовують набір даних наведений у Таблиці 5.2.

Таблиця 5.2 – Структура набору даних для побудови та навчання моделей.

| Назва сутності | Назва атрибуту | Призначення |
|---|---|---|
| Info | Date | Дата продажів |
| | Sales | Кількість проданих копій у цей день |
| | Q | Квартал |
| | Categ | Ідентифікатор жанру проданих ігор |
| | P | Ідентифікатор цінового діапазону проданих ігор |

5.6.1 Побудова моделі прогнозування даних на основі методу часових рядів

Для побудови моделі прогнозування на основі методу часових рядів використовується методи бібліотеки Microsoft ML.Data.Transform.Timeseries.

Перш за все необхідно завантажити набір даних про минули продажі із інформацією про дати у програму.

Програмний код створення контексту для роботи із засобами машинного навчання і завантаження набору даних із БД:

```
string connectionString = 
"server=localhost;user=root;database=gamestore;port=3306;password=root
";

            MySqlConnection conn = new
MySqlConnection(connectionString);

            MLContext mlContext = new MLContext();


            DatabaseLoader loader =
mlContext.Data.CreateDatabaseLoader<ModelInput>();

            string query = "SELECT RentalDate, Year, TotalRentals, q,
categ, p FROM info";
```

```
            DatabaseSource dbSource = new
DatabaseSource(MySqlClientFactory.Instance,
                                    connectionString,
                                    query);

                IDataView dataView = loader.Load(dbSource);
```

Після цього моделі машинного навчання необхідно виділити ознаки, за якими вона, зрозуміє які дані треба використовувати як залежні змінні, а які як цільові.

Програмний код виділення ознак і запуску процесу тренування моделі:

```
var forecastingPipeline = mlContext.Forecasting.ForecastBySsa(
            outputColumnName: "ForecastedRentals",
            inputColumnName: "TotalRentals",
            windowSize: 63,
            seriesLength: 270,
            trainSize: 365,
            horizon: 63,
            confidenceLevel: 0.95f,
            confidenceLowerBoundColumn: "LowerBoundRentals",
            confidenceUpperBoundColumn: "UpperBoundRentals");

            SsaForecastingTransformer forecaster =
forecastingPipeline.Fit(firstYearData);
```

У коді виділення ознак також вказані параметри часового ряду який буде використовуватися для прогнозування, а саме:

- windowSize: 63 – довжина вікна ряду по якій буде будуватися матриця прогнозу, число 63 було обране бо прогноз будуватиметься на 7 днів, для кожної з трьох категорій і кожного з трьох цінових діапазона, тобто 7*3*3=63;
- seriesLength: 270 – довжина ряду який буде зберігатися у буфері для побудови прогнозу;
- trainSize: 365 – довжина часового ряду з початку спостереження для навчання моеделі, 365 бо дані для навчання містять інформацію про продажі за цілий рік;
- horizon: 63 – кількість прогнозованих даних.

Далі, необхідно згенерувати новий набір даних типу «Info», із даними дати на 7 днів наперед і без кількості проданих копій за період, для того аби на основі цієї структури і параметрів навчена модель спрогнозувала дані продажів.

Програмний код генерації нового набору даних відповідно до дати прогнозування:

```
DateTime buf = DateTime.Now;
            List<ModelInput> ListSSA = new List<ModelInput>();
            DateTime mark = buf.AddDays(7);
            int days = 7;
            do
            {
                buf = buf.AddDays(1);

                for (int i = 1; i < 4; i++)
                {

                    ListSSA.Add(new ModelInput
                    {
                        RentalDate = buf,
                        Year = 3,
                        TotalRentals = 0,
                        q = (buf.Month + 2) / 3,
                        categ = i,
                        p = 2
                    });

                }

            }
            while (buf.Date.CompareTo(mark.Date) != 0);

            List<ModelInput> predDataSSA = new List<ModelInput>();

            foreach (ModelInput row in ListSSA.ToList())
            {
                for (int i = 1; i < 4; i++)
                {
                    predDataSSA.Add(new ModelInput
                    {
                        RentalDate = row.RentalDate,
                        Year = row.Year,
                        TotalRentals = 0,
                        q = (row.RentalDate.Month + 2) / 3,
                        categ = row.categ,
                        p = i
                    });
```

```
                }
                  }
```

Після генерації набору даних можна переходити безпосередньо до отримання результатів прогнозу використовуючи навчену модель.

Програмний код прогнозування обсягу продажу:

```
var forecastEngine = forecaster.CreateTimeSeriesEngine<ModelInput,
ModelOutput>(mlContext);

            forecastEngine.CheckPoint(mlContext, modelPath);


            ModelOutput forecast = forecastEngine.Predict();

IEnumerable<double> forecastOutput =
            mlContext.Data.CreateEnumerable<ModelInput>(test,
reuseRowObject: false)
                .Take(days*9 + 1)
                .Select((ModelInput rental, int index) =>
                {
                    double estimate =
forecast.ForecastedRentals[index];
                    return Math.Round(estimate);
                });

            List<double> res = forecastOutput.ToList();

            for (int i = 0; i < predDataSSA.Count(); i++)
            {
                predDataSSA[i].TotalRentals = (float)res[i];
            }
```

Оцінити результати прогнозування можна за допомогою функції Evaluate(), яка відобразить MAE та RMSE похибки моделювання.

Програмний код функції оцінки моделі прогнозування:

```
static void Evaluate(IDataView test, ITransformer ml, MLContext cnt)
       {
            IDataView pred = ml.Transform(test);

            IEnumerable<float> real = cnt.Data.CreateEnumerable<ModelInput>(test,
true).Select(point => point.TotalRentals);

            IEnumerable<float> predict = cnt.Data.CreateEnumerable<ModelOutput>( pred,
true).Select(pred => pred.ForecastedRentals[0]);

            var metrics = real.Zip(predict, (realValue, predictedValue) => realValue -
predictedValue);
```

```
        var MAE = metrics.Average(error => Math.Abs(error));
        var RMSE = Math.Sqrt(metrics.Average(error => Math.Pow(error, 2)));

        Console.WriteLine("Метрики оцінки");
        Console.WriteLine("---------------------");
        Console.WriteLine($"MAE: {MAE:F3}");
        Console.WriteLine($"RMSE: {RMSE:F3}\n");
    }
```

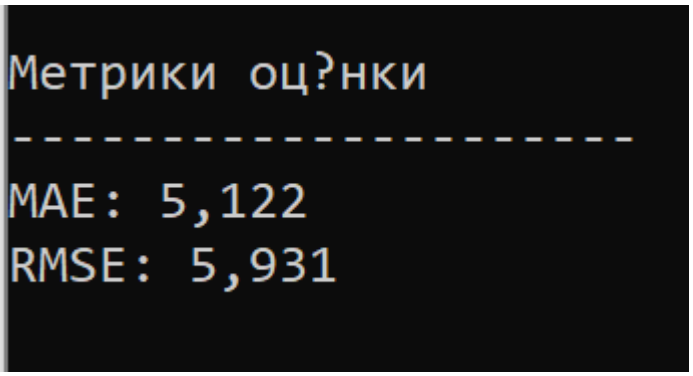



Рис. 4.10 – Результати обчислених метрик

Як відомо, RMSE показує оцінку у тих же одиницях що і цільове значення прогнозування, отже можна зробити висновок, що похибка прогнозування становитиме приблизно 6 одиниць, що є досить адекватним результатом для заданої предметної області, у якій судячи із даних про попередні продажі, їх кількість змінюється від 0 до 25 одиниць. Тобто модель помиляється приблизно на 20%.

### 5.6.2 Побудова моделі прогнозування даних на основі методів регресії

Для побудови моделей регресії використовуються методи бібліотеки машинного навчання ML.Context.Regression.

Перш за все необхідно завантажити набір даних про минули продажі із інформацією про дати у програму.

Програмний код створення контексту для роботи із засобами машинного навчання і завантаження набору даних із БД:

```
string connectionString =
"server=localhost;user=root;database=gamestore;port=3306;password=root
";
```

```
            MySqlConnection conn = new
MySqlConnection(connectionString);

            MLContext mlContext = new MLContext();


            DatabaseLoader loader =
mlContext.Data.CreateDatabaseLoader<ModelInput>();

query = "SELECT year(RentalDate) as yearr, month(RentalDate) as month,
dayofmonth(RentalDate) as day, Year, TotalRentals as Label, q, categ,
p FROM info";

            DatabaseSource dbSource1 = new
DatabaseSource(MySqlClientFactory.Instance,
                                connectionString,
                                query);

                IDataView trainingData = loaderFTT.Load(dbSource1);
```

Після цього моделі машинного навчання необхідно виділити ознаки, за якими вона зрозуміє які дані треба використовувати як залежні змінні, а які як цільові. Так як в даному випадку будується модель регресії, ознаки мають приймати дискретні значення, тобто поле дати необхідно розділити на складові року, місяця і дня.

Програмний код виділення ознак:

```
var pipeline = mlContext.Transforms.Concatenate(
                "Features",
                nameof(ModelInput.yearr),
                nameof(ModelInput.month),
                nameof(ModelInput.day),
                nameof(ModelInput.Year),
                nameof(ModelInput.q),
                nameof(ModelInput.categ),
                nameof(ModelInput.p))
                // step 2: cache the data to speed up training
                      .AppendCacheCheckpoint(mlContext);
```

Так як бібліотека машинного навчання включає в себе декілька методів побудови моделей регресії, для дослідження було вирішено побудувати чотири тестові моделі різними методами. Навчання моделі проводилось на вибірці обсягом 700 екземплярів за період одного року. Якість моделей перевірялась на певну дату прогнозу. Приклад тестової вибірки наведено у таблиці 5.3. Показники похибок для кожного методу наведені на рисунку 4.11.

Таблиця 5.3 - Тестова вибірка даних для перевірки моделі прогнозування

| Дата продажів | Квартал | Жанр | Цінова категорія | Кількість продажів |
|---|---|---|---|---|
| 2020-08-12 | 4 | 1 | 1 | 17 |

Програмний код навчання моделей регресії методами SDCA, Poisson, FastTree, FastTreeTweedie та їх оцінки:

```
    (string Name, IEstimator<ITransformer> Learner)[] regressionLearners =
                    {
                        ("SDCA", mlContext.Regression.Trainers.Sdca()),
                        ("Poisson",
mlContext.Regression.Trainers.LbfgsPoissonRegression()),
                        ("FastTree", mlContext.Regression.Trainers.FastTree()),
                        ("FastTree Tweedie",
mlContext.Regression.Trainers.FastTreeTweedie())
                    };


                var results = new Table(TableConfiguration.Unicode(), "Learner",
"RMSE", "MSE", "MAE", "Prediction");


                foreach (var learner in regressionLearners)
                {
                    Console.WriteLine($"Навчання та оцінка моделі використовуючи
метод {learner.Name}");

                    var fullPipeline = pipeline.Append(learner.Learner);

                    var trainedModel = fullPipeline.Fit(trainingData);



                    var predictions = trainedModel.Transform(trainingData);
                    var metrics = mlContext.Regression.Evaluate(
                        data: predictions,
                        labelColumnName: "Label",
                        scoreColumnName: "Score");

                    var sample = new ModelInput
                    {
                        Year = 1,
                        yearr = 2020,
                        month = 12,
                        day = 8,
                        q = 4,
                        categ = 1,
                        p = 1

                    };
```

```
                var engine = mlContext.Model.CreatePredictionEngine<ModelInput,
DemandPrediction>(trainedModel);


                var prediction = engine.Predict(sample);

                results.AddRow(
                learner.Name,
                metrics.RootMeanSquaredError.ToString("0.##"),
                metrics.MeanSquaredError.ToString("0.##"),
                metrics.MeanAbsoluteError.ToString("0.##"),
                prediction.PredictedCount.ToString("0"));


            }
```

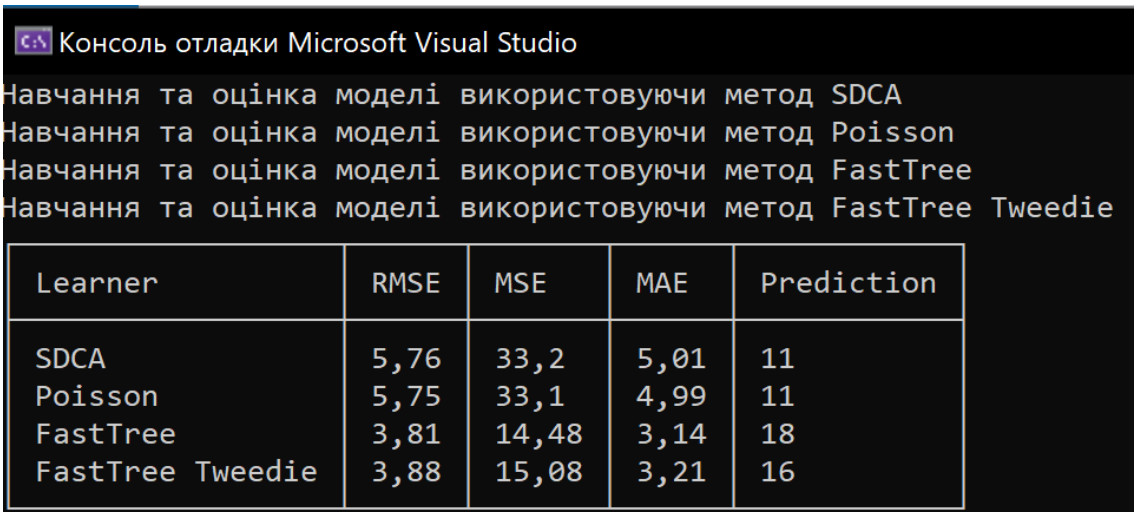



Рис. 4.11 – Результати оцінки побудованих моделей

Як видно на рисунку 4.11, найкращий результат показав метод FastTree: він має найменші значення похибок, але за результатами прогнозування FastTree дає завищенні значення, в той час як метод FastTreeTweedie занижує прогнозні значення. Тому на практиці, для побудови бізнес стратегії рекомендується застосовувати обидва методи.

## 5.7 Апробація методів прогнозування на прикладі системи продажів відеоігор

Після експериментального аналізу методів прогнозування, було обрано дві моделі на основі найкращих методів: метод прогнозування на основі часових рядів і метод прогнозування на основі FastTree.

Процес навчання моделей і отримання результатів було реалізовано на окремій сторінці звіту прогнозування.

Програмний код імплементації побудови моделей прогнозування на прикладі системи електронної комерції відеоігор наведено у Додатку Б.

На сторінці панелі адміністратора з’являється кнопка переходу на сторінку звітів прогнозування.

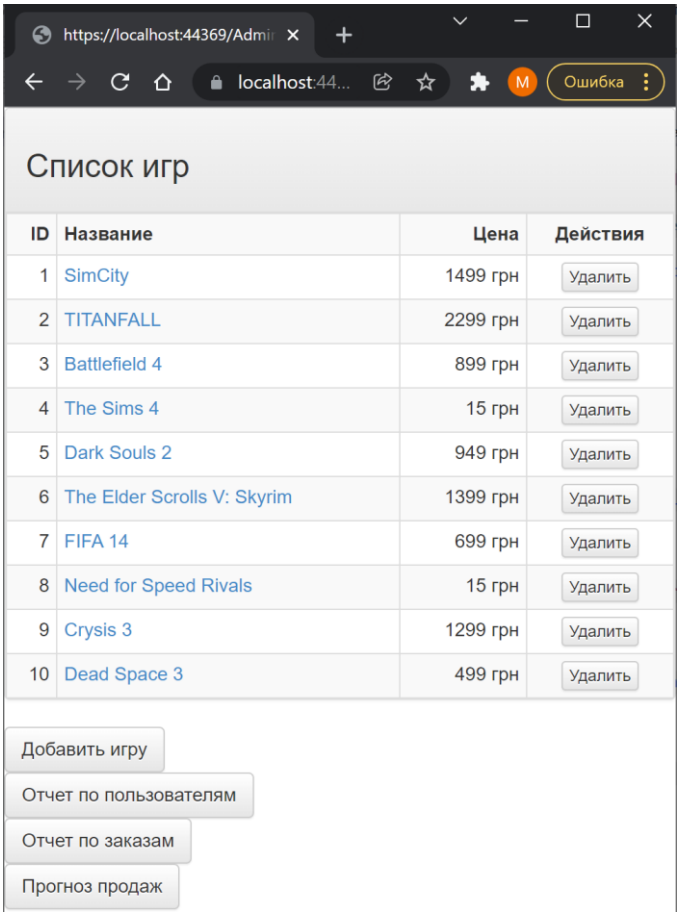



Рис. 4.12 – Сторінка панелі адміністратора.

При переході на сторінку, користувачу надається представлення для вибору параметрів прогнозу. Якщо спробувати отримати прогноз не вибравши жодного параметру, моделі спрогнозують дані для усіх можливих варіантів.

На рисунку 4.13 зображено результати прогнозування для обраної категорії «RPG» і цінового діапазону «до 500 грн.».

Отримавши результати прогнозування за допомогою двох найкращих методів прогнозування для обраної предметної області, власник бізнесу може робити висновки щодо рентабельності тієї чи іншої категорії товарів, а також планувати обсяги закупівлі, аби не залишитися із не реалізованими товарами закупленими по початковій ціні.

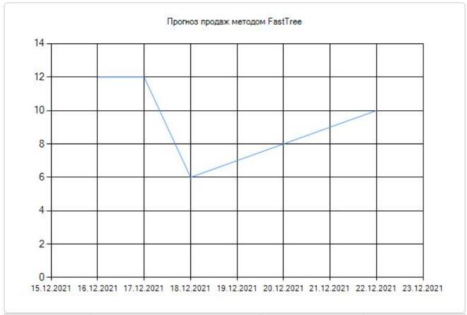
Як видно на графіках, результати прогнозування не монотонні графіки, а ламані. Це свідчить про  те, що і у методі регресії і у методі часових рядів, вихідні данні це дискретні величини, а не неперервні, і приймають конкретні значення відповідно до вхідних даних, а саме дат.

Прогнозирование продаж

RPG | До 500 грн. | Получить

Прогноз продаж на категорию RPG и ценовой диапазон До 500 грн.


| Дата | Ориентировочное кол-во продаж | Категория товаров | Ценовая категория |
|---|---|---|---|
| 16.12.2021 | 10 штук | RPG | До 500 грн. |
| 17.12.2021 | 14 штук | RPG | До 500 грн. |
| 18.12.2021 | 9 штук | RPG | До 500 грн. |
| 19.12.2021 | 10 штук | RPG | До 500 грн. |
| 20.12.2021 | 12 штук | RPG | До 500 грн. |
| 21.12.2021 | 8 штук | RPG | До 500 грн. |
| 22.12.2021 | 9 штук | RPG | До 500 грн. |




| Дата | Ориентировочное кол-во продаж | Категория товаров | Ценовая категория |
|---|---|---|---|
| 16.12.2021 | 12 штук | RPG | До 500 грн. |
| 17.12.2021 | 13 штук | RPG | До 500 грн. |
| 18.12.2021 | 7 штук | RPG | До 500 грн. |
| 19.12.2021 | 7 штук | RPG | До 500 грн. |
| 20.12.2021 | 8 штук | RPG | До 500 грн. |
| 21.12.2021 | 9 штук | RPG | До 500 грн. |
| 22.12.2021 | 10 штук | RPG | До 500 грн. |

Рис. 4.13 – Результати прогнозування обсягів продажів для категорії «RPG» і цінового діапазону «до 500 грн.»

## 5.8 Розробка SQL-запитів для організації роботи системи

Для підтримки процесу автоматичного аналізу даних, були розроблені функції і тригери які визивають ці функції після певних подій які відбуваються із таблицями бази даних [22].

Було реалізовано тригер «otgs_AFTER_INSERT» який відповідає за оновлення даних у таблиці «populargames» підтримуючі актуальну інформацію про найпопулярніші товари та після кожного замовлення оновлює інформацію про кількість проданих копій за день відповідно до жанру гри і

його діапазону для того аби при побудові нових моделей прогнозування вони постійно оновлювалися і доповнювалися новою інформацією для більш кращого прогнозування.

Програмний код реалізації тригеру наведено у Додатку Б.

У реалізації тригеру використано фунціонал СУБД MySQL який називається CURSOR. Ця конструкція дозволяє організувати складну обробку даних на стороні сервера звичним способом, а саме - рядок за рядком [23]. Обробка даних на стороні MySQL може скоротити час обробки даних, тому що не потрібно передавати дані з бази в програму і навпаки.

В даному випадку курсор отримує дані про три найпопулярніші товари із комбінації таблиць «Orders» і «OTGS» і рядок за рядком записує їх у таблицю «PopularGames».

# ВИСНОВКИ

Під час виконання кваліфікаційної роботи було розроблено проектні рішення, що направлені на розв’язання задачі прогнозування обсягів продажу веб додатку електронної комерції відеоігор.

В роботі виконано аналіз методів побудови прогнозів на основі існуючих даних, наведено математичний опис методів SDCA, регресії Пуассона, дерев прийняття рішень, моделей часових рядів. Сформульована постановка задачі на дослідження та розробку методу, який імплементується в інформаційну систему інтернет-магазину ігрових застосунків для вирішення задачі прогнозування обсягів продажів. На основі порівняння ефективності методів за допомогою метрик MAE, RMSE, MSE для прогнозування обрано два методи – дерев регресії та часових рядів.

Етапи проектування та розробки інтернет-магазину в роботі проілюстровані діаграмами потоків даних, класів та логічними і фізичними моделями БД. Функція прогнозування реалізована з використанням методів бібліотеки Microsoft.ML, яка містить велику кількість різних алгоритмів машинного навчання, що значно спрощує процес імплементації функцій у систему, адже не потрібно власноруч будувати різні моделі прогнозування, коли можна використати готові сервіси. Для адміністраторів інтернет-магазину реалізовано функції звітів про аналіз даних користувачів і замовлень, а також вибору параметрів прогнозування і перегляду їх результатів.

Порівнюючи реалізовану програмну систему із аналогами вже присутніми на ринку(як вітчизняному так і закордонному), можна дійти висновку, що система може конкурувати із ними завдяки відсутності у аналогів функціоналу прогнозування продажів імплементованого безпосередньо у систему.

Для розробки веб-системи «Інтернет-магазин» використані мова програмування C#, технологія Active Server Pages(ASP), техніка створення програмних додатків типу MVC ( Model-View-Controller), бібліотеки ML для використання методів машинного навчання і побудови моделей регресії та часових рядів.

База даних створена у СУБД MySQL, для маніпулювання даними використані засоби методів машинного навчання DataView.

Апробація розроблених методів прогнозування на даних інтернет магазину відеоігор показала, що реалізовані методи адекватно прогнозують обсяги продажів на майбутній період і показають, з поправкою на похибки, схожі прогнози.

# ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ


1. Електронна комерція. URL: https://pidruchniki.com/12090613/finansi/elektronna_komertsiya (дата звернення: 15.10.2021).
2. Електронна комерція як елемент системи світового господарства // У. О. Балик, М. В. Колісник – Львів: Національний університет “Львівська політехніка”, 2014. - № УДК: 339.1. – с. 11-19.
3. Класифікація сторінок в інтернеті. URL: http://scbali.com/ua/web-studiya/typy-saytiv.html (дата звернення: 20.10.2021).
4. Калита Н. И. Формирование базового набора тестовых сценариев для Web-базированных информационных систем / Н. И. Калита, М. А. Батурина // Проблемы информационных технологий. – 2010. – № 01(007). – С. 112-118..
5. Моделі великих даних для систем електронної комерції / А. Ю. Берко. – Львів: 2018. - УДК 004.652 – с. 37-42.
6. Sitnikov, D. QL-On-Hadoop Systems: Evaluting Performance of Polybase for Big Data Processing / Minukhin, S., Fedko, V., Sitnikov, D. // International Scientific- Practical Conference on Problems of Infocommunications Science and Technology, Proceedings. – 2019. – P. 591–594.
7. Коваленко А.І. Технології розробки корпоративних web-додатків [Електронне видання]: Конспект лекцій для студентів спеціальності 122 – «Комп’ютерні науки» – Харків: ХНУРЭ, 2019 – 120 с.
8. Що таке DFD (діаграми потоків даних).URL: https://habr.com/ru/company/trinion/blog/340064/ (дата звернення: 15.06.2020).
9. Використання засобів uml для прогнозування надійності програмного забезпечення на етапі його проектування / В.С. Яковина, Ю.І. Парфенюк - Національний університет “Львівська політехніка”, кафедра програмного забезпечення, 2013. - УДК 004.052; 004.415.2
10. Розробка uml діаграми варіантів використання. URL: https://studfile.net/preview/5200239/page:6/ (дата звернення: 27.07.2021).
11. Методичні вказівки до виконання лабораторної роботи “Розробка діаграми класів у середовищі Umbrello UML Modeller” : для студентів спеціальності 121 –
12. Буч Г., Джекобсон А., Рамбо Д. UML. Руководство пользователя. М.: ДМК Пресс, 2004.

13. Гарсіа-Моліна Г., Ульман Д., Уідом Д. Системи баз даних. Повний курс: Переклад з англ. – М.: Вільямс, 2003.
14. ASP.NET MVC 4. Разработка реальных Web-приложений с помощью ASP.NET MVC. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2013. – 432 с.
15. Приклади об'єктно-орієнтованого проектування. Паттерни проектування. / Є. Гамма, Р. Хелм, Р. Джонсон та інші; переклад з англійської А. Слінкіна. – СПб.: Пітер, 2001. – 368 с.
16. Інформаційні технології. Англійською та українською мовами. / Уклад. Хацько Н.Є., Гавриленко С.Ю. – Харків : НТУ «ХПІ», 2019. – 39 с.
17. Stochastic Dual Coordinate Ascent Methods for Regularized Loss Minimization. Shai Shalev-Shwartz, Tong Zhang. Journal of Machine Learning Research 14 (2013) p. 567-599
18. Poisson Regression. URL: https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Poisson_Regression.pdf (дата звернення: 20.09.2021).
19. Gehrke, J. E.; Ramakrishnan, R.; and Ganti, V. 2000. Rainforest – A framework for fast decision tree construction of large datasets. Data Mining and Knowledge Discovery 4:2/3:127–162.
20. Golyandina N. and Osipov E. (2007) The Caterpillar-SSA method for analysis of time series with missing values, J. of Statist. Plann. Inference, 137, 2642–2653.
21. Інфраструктура наукових інформаційних ресурсів і систем. Збірник обраних наукових статей. Праці Четвертого Всеросійського симпозіуму (С.-Петербург, 6-8 жовтня 2014 р.) Під ред. Е.В. Кудашева, В.А. Серебрякова. У 2-х тт. Т. 2. М .: ВЦ РАН, 2014.
22. Работа с MySQL, MS SQL Server и Oracle в примерах : практ. пособие. / С. С. Куликов. — Минск: БОФФ, 2016. — 556 с.ISBN 978-985-430-054-
23. Роб П., Коронел К. Системы баз данных: проектирование, реализация и управление : Пер. с англ. - СПб.:БХВ-Пстсрбург, 2004. – 1040.