Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)
Кафедра Інформатики
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

ДОСЛІДЖЕННЯ МЕТОДІВ АНАЛІЗУ ДАНИХ ДЛЯ РЕАЛІЗАЦІЇ УНІВЕРСАЛЬНОЇ СИСТЕМИ АНАЛІЗУ РИНКУ
(тема)

Виконав:
студент 2 курсу, групи ІНФм-19-1
Тесленко Д. В.
(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки
(код і повна назва спеціальності)

Освітня програма Інформатика
(повна назва освітньої програми)

Керівник проф. Кузьомін О. Я.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ Кобилін О.А.
(підпис) (прізвище, ініціали)

2020 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Освітня програма Інформатика
(повна назва освітньої програми)

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

ЗАВДАННЯ
НА АТЕСТАЦІЙНУ РОБОТУ

студентові Тесленку Дмитру Вікторовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів аналізу даних для реалізації універсальної системи аналізу ринку

затверджена наказом по університету від «_23_» жовтня 2020 року № 1428Ст.

2. Термін подання студентом роботи до екзаменаційної комісії _4_ грудня 2020 р.

3. Вихідні дані до роботи Види ринків та їх функції, перелік використовуваних програмних засобів: Intellij IDEA, теоретичні відомості про аналіз даних у предметній області ринку.

4. Перелік питань, що потрібно опрацювати в роботі
1. Аналіз предметної області ринку та постановка задачі дослідження
2. Аналіз методів аналізу даних
3. Формування вимог до системи
4. Проектування системної реалізації
5. Реалізація прототипу системи

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Комп'ютерна презентація

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на атестаційну роботу	23.10.2020	
2	Аналіз завдання, підбір літератури	23.10.20-26.10.20	
3	Аналіз літератури з досліджуваної проблеми	26.10.20-02.11.20	
4	Аналіз технічних засобів для реалізації	02.11.20-06.11.20	
5	Проектування системи	06.11.20-10.11.20	
6	Програмна реалізація	10.11.20-16.11.20	
7	Оформлення пояснювальної записки	16.11.20-25.11.20	
8	Перевірка на плагіат	25.11.20	
9	Рецензування	30.11.20	
10	Підготовка презентації та доповіді	30.11.20-04.12.20	
11	Занесення роботи в електронний архів	04.12.20	
12	Попередній захист атестаційної роботи	07.12.20	

Дата видачі завдання 23 листопада 2020 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Кузьомін О. Я.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до атестаційної роботи: 58 с., 5 табл., 16 рис., 40 джерел.

ЕЛЕКТРОННИЙ РИНОК, ФУНКЦІЇ РИНКУ, АНАЛІЗ ДАНИХ, ДЕРЕВА РІШЕНЬ, JAVA, SPRING FRAMEWORK.

Метою дослідження є отримання позитивного впливу на роботу першої функції ринку та налагодження ефективної комунікації між покупцем та продавцем.

Об’єктом дослідження є підбір найбільш ефективного методу аналізу даних в рамках програмної реалізації, що буде відповідати вимогам мети дослідження.

Задачею дослідження є проектування системи, яка за допомогою обробки даних з різних типів ринку буде надавати корисну інформацію користувачу та задовольняти поставлену мету дослідження.

У результаті роботи спроектована універсальна система аналізу даних ринку та реалізований її прототип.

E-MARKET, MARKET FUNCTIONS, DATA ANALYSIS, DECISION TREES, JAVA, SPRING FRAMEWORK.

The goal of research is focused on getting a positive influence on first of the market functions and setting-up an effective communication between buyer and seller.

The object of research is a selection of most efficient data analysis method within program realization restrictions, which is going to satisfy the goal of the research.

The task of research is focused on designing a system, which with use of data processing from different markets will give out useful information to end–user and will satisfy set goal of the research.

As a result of work, a universal market data analysis system was designed and it’s working prototype was implemented.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

B2B – Business-to-Business – Бізнес-до-Бізнесу

B2C – Business-to-Customer – Бізнес-до-Споживача

B2G – Business-to-Government – Бізнес-до-Уряду

G2G – Government-to-Government – Уряд-до–Уряду

G2B – Government-to-Business – Уряд-до-Бізнесу

G2C – Government-to-Customer – Уряд-до-Споживача

C2G – Customer-to-Government – Споживач-до-Уряду

C2B – Customer-to-Business – Споживач-до-Бізнесу

C2C – Customer-to-Customer – Споживач-до-Споживача

CorQ – Quantitative Correlation – кількісне співвідношення

CorC – Categorial Correlation – категоріальне співвідношення

SumQ – Quantitative Summarization – кількісне узагальнення

SumC – Categorial Summarization – категоріальне узагальнення

ЕР – електронний ринок

EFA – Exploratory Factor Analysis – розвідувальний факторний аналіз

CFA – Confirmatory Factor Analysis – підтверджуючий факторний аналіз

SEM – Structural Equation Modeling – моделювання структурних рівнянь

CFI – Comparative Fit Index – порівняльний показник відповідності

RMSEA – Root Mean Square Error of Approximation – середньоквадратична похибка наближення

IQ – Intelligence quotient – коефіцієнт інтелекту

POJO – Plain Old Java Object – «Простий старий Java-об’єкт»

IDE – Integrated Development Environment – інтегроване середовище розробки

MVC – Model-view-controller – Модель-представлення-контролер

ВСТУП

Інтернет, мобільні телефони та електронна комерція впливають на динаміку всіх галузей ринку. Традиційний бізнес знаходиться в стані коливання, і за таких обставин, спритні та інноваційні компанії почали використовувати ці технології для прийняття нової моделі ведення бізнесу.

Сьогодні, основна увага приділяється управління відносинами з клієнтами. Компанії більше фокусуються на зростанні кількості клієнтів та їх утриманні, замість пошуку нових. Вони складають бази даних про окремих клієнтів, щоб краще їх розуміти та створювати індивідуальні пропозиції та повідомлення. Тепер корпоративні монологи заміняються діалогами з клієнтами та вдосконалюються методи вимірювання прибутковості та вартості споживача [1].

Ці задачі вимагають реалізацію швидких, потужних та зручних програмних рішень, які б налагодили Інтернет-спілкування з кінцевим користувачем. Великий об’єм даних з цих додатків, не дозволяє компаніям повністю спиратися на обмежену кількість персоналу аналітиків, тому вони потребують більш ефективні та розвинуті системи аналізу даних, що дозволять їм формувати нові маркетингові стратегії.

Кожне підприємство намагається створити свої окремі системи, не спираючись на сторонні реалізації, для збереження якомога більшої кількості внутрішньої інформації, що в свою чергу позбавляє споживача багатьох можливостей оцінки ринку та товарів на ньому.

Це утворює проблему, рішенням якої є створення універсальної системи оцінки ринка, що буде знаходитися у вільному доступі та розширить низку можливостей споживача, а також встановить рівні умови у його використанні.

1 ТЕОРЕТИЧНИЙ РОЗДІЛ

1.1 Аналіз предметної області

1.1.1 Поняття ринку в економіці

Поняття «ринок» є неоднозначним терміном, що цитується професіоналами в економіці по-різному, в залежності від різноманітних факторів. Але найбільш загальним є визначення, що ринок являє собою склад систем, установ, процедур або соціальних відносин, в разі яких сторони беруть участь в обміні [2].

Ринки займають основну роль в розвитку економіки, сприяючи обміну інформації, товарів та послуг. В цьому процесі, вони створюють економічну цінність для покупців, продавців, посередників ринку та для суспільства в цілому. За останню декаду спостерігається різке та неминуче зростання ролі інформаційних технологій, як на традиційних ринках, так і на доволі нещодавно сформованих електронних ринках (ЕР) [3].

1.1.2 Електронна комерція: класифікації та історія

Поняття електронної комерції визначає, що вона являє собою процес обміну ділової інформації, підтримки ділових відносин та ведення бізнес-операцій за допомогою телекомунікаційних мереж [4]. Що є дуже схожим з загальним визначенням ринку, але з фокусом на використання більш новітніх технологій.

Загалом, електронна комерція є актуальною серед декількох підприємств (B2B) або між підприємством та споживачами (B2C) [5], але Інтернет дозволяє охоплювати більший спектр потенційної комерційної діяльності та обміну інформації, а також надає шанси для утворення нових ЕР систем, які можна побачити у таблиці 1.1.

Таблиця 1.1 – Класифікація систем електронної комерції

	Держава	Бізнес	Споживач
Держава	G2G (узгодження)	G2B (інформація)	G2C (інформація)
Бізнес	B2G (електронні держзакупівлі)	B2B (електронна комерція, електронні ринки)	B2C (електронна комерція)
Споживач	C2G (дотримання податків)	C2B (порівняння цін, електронні ринки)	C2C (електронні ринки)

Більша частина з цих ЕР була сформована в Інтернеті ще у середини 1999 року. Однак концепція ЕР датується серединою 1940-х років, коли вперше була задокументована система під назвою «Selevision», що застосовувалась для створення віддаленого ринку продажу цитрусів у Флориді за телефоном. Проте, більш реальні розробки ЕР почались лише у кінці 1970-х років, завдяки реалізації першого онлайн ринку на основі комп’ютерних технологій [6].

Протягом своєї історії, основна функція ЕР полегшення торгових операцій для покупців та продавців – залишилася незмінною. Та лише інновації технологій стали основним середовищем для змін ринкового механізму, починаючи від аналогових телефонних систем, до цифрових комп’ютерних мереж [7].

1.1.3 Основні функції ринку

Ринок, як електронний так і інші, має три основні функції, зазначені в рисунку 1.1: підбір покупців і продавців; сприяння обміну інформації, товарів, послуг та платежів, пов’язаних з ринковим операціями; та надання

інституційної інфраструктури, такої як законодавча або нормативна база, що забезпечує ефективне функціонування ринку [8].

У сучасній економіці, перші дві функції забезпечують посередники, тоді як інституційна інфраструктура, як правило, є провінцією уряду. Основані в Інтернеті ЕР використовують інформаційні технології для виконання вищевказаних функцій з підвищеною ефективністю, що призводить до зменшення операційних затрат та більш продуктивного ринку загалом.

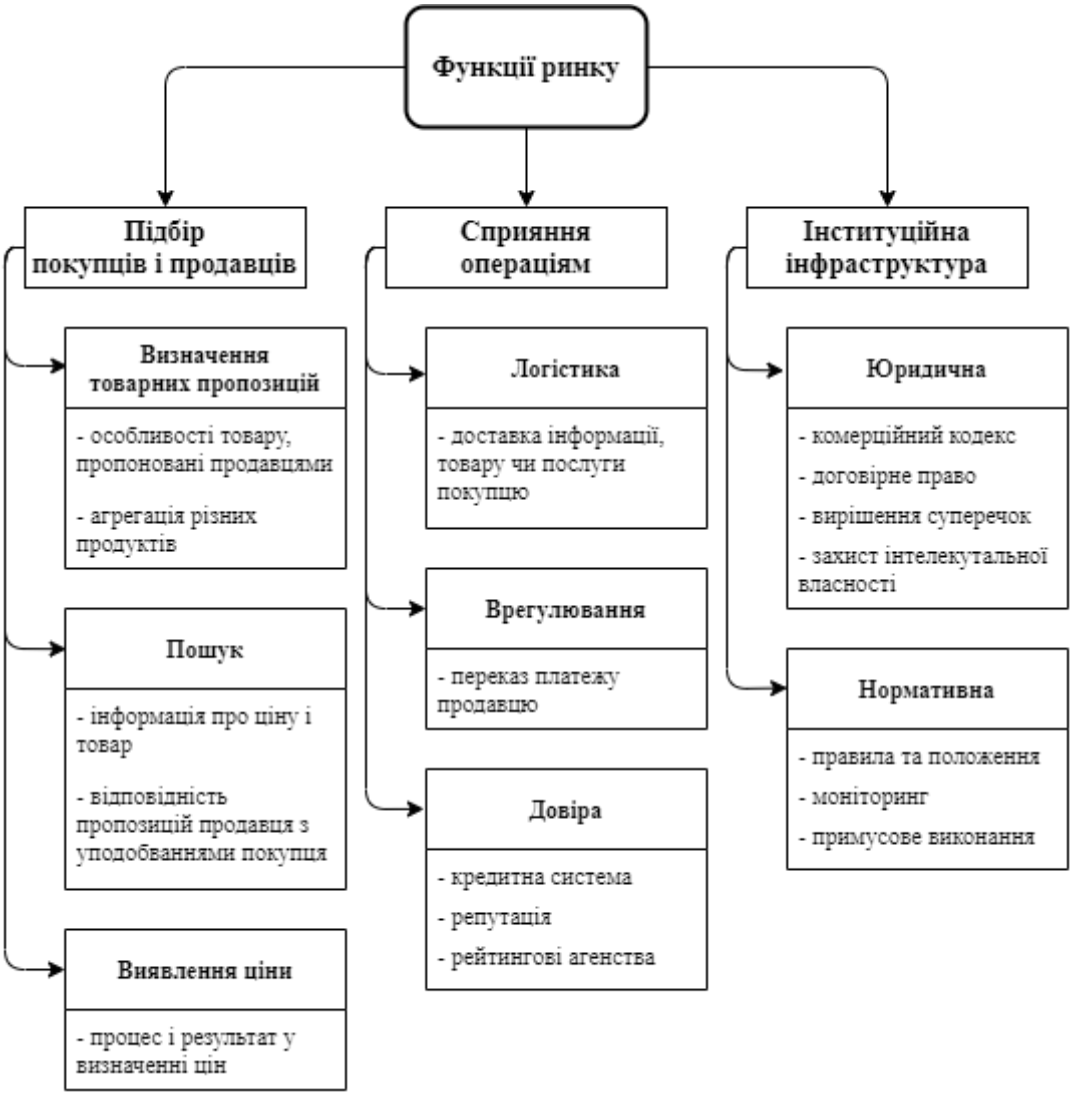

Рисунок 1.1 – Функції ринку

Ринки «очищаються» шляхом узгодження попиту та пропозицій. Цей процес має три основні компоненти: визначення товарних пропозицій, пошук та виявлення ціни.

Визначення товарних пропозицій. Ринки надають продавцям інформацію про попит, що дозволяє їм правильно використовувати економічні ресурси, такі як капітал, технології, працю, та розробляти продукцію за характеристиками, що відповідають потребам покупців. Також, продавці визначають графік товарних пропозицій для максимізації їх прибутку на основі:

– інформації про попит покупця;

– вартості вхідних матеріалів;

– наявності технологій для виробництва та розповсюдження;

– операційних витрат на управління, поширення та оплату виробництва.

Пошук. Покупці обирають свої покупки з доступних пропозицій товарів зважаючи на такі фактори, як ціна та характеристики товару. При отриманні та обробці цієї інформації, покупці несуть «пошукові витрати». Ці витрати включають у себе втрату часу на пошук, а також пов’язані з цим можливі оплати за транспортування, телефонні дзвінки, Інтернет, журнали та інше. Так само продавці можуть нести втрати при пошуку цільових покупців, використовуючи дослідження ринку, рекламу та дзвінки для продажу.

Виявлення ціни. Ключовою функцією в нашій економічній системі є виявлення цін, тобто процес визначення ціни, за якої попит і пропозиція «звільняються» та відбувається обмін. Для певних ринків, таких як фінансові, це є їх основною функцією. Ринки можуть використовувати низку механізмів для виявлення ціни. Наприклад, деякі фінансові ринки використовують один або декілька типів аукціонів «ринок викликів» – аукціон на відкритті торгового дня в Нью-Йоркській фондовій біржі, де ставки приймаються до певного часу, а обмін відбувається при відкритті ринку. Так формується перша ціна, що надалі повідомляється до ринку в цілому та починає

торгівлю. Інші ринки, такі як традиційний автосалон, використовують переговори між покупцем і продавцем для досягнення оптимальної ціни. Але, в таких ринках як типовий універмаг, продавці орудують твердими пропозиціями, які покупці можуть взяти або залишити.

Поведінка покупців, продавців та посередників мотивована бажанням максимізувати приватну користь. Правильна реалізація цих трьох компонентів не тільки полегшує їх взаємовідносини, але також призводить до більш ефективного розподілу виробничих ресурсів, що є основним критерієм успішної економіки [9]. Тому, розглядання наступних функцій ринку не має значення без налаштування ефективного процесу підбору покупців і продавців.

1.1.4 Аналіз даних в галузі ринку

Термін «Аналіз даних» використовується досить давно, ще до початка процвітання комп'ютерної ери, як розширення математичної статистики, починаючи з розробок кластерного аналізу та інших багатовимірних методів, до встановлення таких понять, як «дослідницький» та «підтверджуючий» аналіз даних в статистиці [10].

Можливим визначенням аналізу даних можна назвати процес обчислення різних зведень та похідних значень з наданого набору даних. Крім того, цей процес може стати розумнішим, якщо автоматизувати міркування професійних аналітиків даних та використовувати підходи розроблені в сфері штучного інтелекту [11].

Загалом, цей термін застосовується як парасолька для охоплення всіх різних видів діяльності, включаючи ринкову, з акцентом на математичну статистику та її розширення.

Одним з популярних прикладів використання аналізу даних в галузі ринку є Добування Даних (англ. Data Mining). Це дисципліна для пошуку

цікавих закономірностей в базах даних, що вважається частиною процесу «відкриття знань» (англ. knowledge discovery), де знання представляють собою набір прийомів для отримання кількісних формул та категоріальних постанов, що використовуються для визначення зв’язку між різними ознаками. З цього процесу можна винести різноманітні корисні факти, які можуть бути застосовані у маркетингових стратегіях бізнесу [12].

Дослідження 2012 року показують, що компанії, які використовують аналітику для прийняття рішень, вигідніші на 6%, ніж ті, що її не використовують [13]. Але аналіз даних можна застосовувати не тільки як систему для збагачення продавця, а й як інструмент для інформування та покращення досвіду покупця. Можливість аналізу даних впливати на обидві сторони, робить цей процес ідеальним варіантом для ефективної реалізації першої функції ринку.

1.2 Огляд методів аналізу даних

Аналіз даних розширює «знання» за допомогою знаходження в даних співвідношень між ознаками або узагальнень сутностей або ознак. В свою чергу ознаки поділяються на кількісні та категоріальні. Комбінація цих двох основ утворює чотири основні групи методів: CorQ, CorC, SumQ, та SumC, що формують ядро аналізу даних. Слід зазначити, що в даний час переважає різна категоризація задач, пов’язаних з аналізом даних, одна з класичної математичної статистики, з ухилом на математично облікові моделі, а інша з машинного навчання добування даних – система, що концентрується на вивченні категорій об’єктів та залишає такі проблеми, як кількісне узагальнення [14].

Задача співвідношення або узагальнення зазвичай містить у собі п’ять інгредієнтів:

– збір математичних структур, що знаходяться в даних;

– модель обчислення, що співвідносить дані та математичні структури;

– критерій для оцінки відповідності даних та структур;

– метод оптимізації критерію;

– візуалізацію результатів.

Де математичні структури приймають одну з форм:

– лінійне поєднання ознак;

– нейронна мережа, що трансформує набір вхідних ознак у набір цільових ознак;

– дерево рішень, побудованого за сукупністю ознак;

– кластер сутностей;

– розділення набору сутностей на ряд кластерів, що не перекриваються.

Після вибору типу математичної структури, що буде використовуватися, її параметри можна дізнатися з даних.

Надалі, метод відповідності спирається на модель обчислення, що включає в себе функцію оцінюючу адекватність математичної структури – критерій, який вимірює або відхилення від цілі (яке слід мінімізувати), або схожість з нею (яку слід максимізувати).

Доступні на сьогодні обчислювані методи для оптимізації критерію охоплюють три основні групи:

– глобальна оптимізація, тобто пошук найкращого можливого рішення, здійсненого як для лінійних кількісних, так і для простих дискретних структур;

– локальне вдосконалення за допомогою таких загальних підходів, як градієнтний підйом і спуск, почергова оптимізація та пошук найближчих сусідів;

– натхненні природою підходи, що залучають сукупність допустимих рішень та їх ітеративну еволюцію – підхід, що включає відносно недавні досягнення у обчислювальних можливостях, таких як генетичні алгоритми, еволюційні алгоритми, оптимізація рою часток.

Слід зазначити, що в даний час не існує систематизованого опису всіх можливих комбінацій задач, типів даних, математичних структур, критеріїв та методів відповідності. Тому, наступне розподілення, вказане у таблиці 1.2, є прикладом найбільш вивчених проблем та методів їх рішень для кожної з чотирьох груп аналізу даних.

Таблиця 1.2 – Приклад методів аналізу даних для кожної з груп методів

Узагальнення	Кількісне	Аналіз основних компонентів
	Категоріальне	Кластерний аналіз
Співвідношення	Кількісне	Регресійний аналіз
	Категоріальне	Контрольована класифікація

Чотири підходи зазначені вище утворилися в різних фреймворках та зазвичай не вважаються пов’язаними між собою. Однак, вони пов’язані в контексті аналізу даних. Кластерний аналіз, наприклад, є найбільш популярним серед них, та застосовується в самих різноманітних галузях [15]. Але, інші методи, що переважно використовуються в окремих сферах, також є частиною більшої картини, та, в залежності від різних характеристик даних, можуть бути використані вже в нових галузях.

1.3 Постановка задачі дослідження

Сучасні реалізації електронного ринку, такі як онлайн сайти купівлі/продажу (http://amazon.com) або системи аукціонного типу (http://ebay.com), пропонують покупцю цілу низку можливостей для їх використання, що включають в себе розширений смарт-пошуку з фільтрами та порадами, історію покупок, індивідуальні пропозиції й таке інше, та в той же час продовжують неспинно розвиватися, хоча вже займають всесвітньо лідируючі позиції в галузі. Всі ці платформи містять всередині складні

модернізовані системи аналізу даних, що збирають та обробляють масивну кількість корисної інформації, і незважаючи на зручні функції, що отримає користувач, це є тільки малим відсотком від даних, що використовують для свого збагачення продавці або власники.

В свою чергу, інформування покупців ідентичною інформацією створює рівні умови на платформі. Користувач отримує змогу самостійного аналізу того чи іншого ринку, а також можливість слідкування за його трендами, збільшенням або зменшенням попиту на різні товари, активність інших покупців та багато іншого. Це призводить не лише до більш ефективних покупок зі сторони користувача, а також до більшої зацікавленості в ринку в цілому. Ця зацікавленість збільшує кількість продажів та підбурює продавців на конкуренцію між собою, що призводить до створення кращих продуктів та зручніших цін на них.

Виходячи з зазначеного вище, можемо виділити об’єкт, мету та задачу дослідження.

Об’єктом дослідження є підбір найбільш ефективного методу аналізу даних в рамках програмної реалізації, що буде відповідати вимогам мети дослідження.

Метою дослідження є отримання позитивного впливу на роботу першої функції ринку та налагодження ефективної комунікації між покупцем та продавцем.

Задачею дослідження є проектування системи, яка за допомогою обробки даних з різних типів ринку буде надавати корисну інформацію користувачу та задовольняти поставлену мету дослідження.

2 ПРОЕКТНИЙ РОЗДІЛ

2.1 Опис проведеного дослідження

2.1.1 Формування вимог до дослідження

Першочерговою задачею роботи є вибір найбільш ефективного для майбутньої програмної реалізації методу аналізу даних. Для цього зробимо перелік можливих методів аналізу та обробки, розглянемо приклади їх використання в різних галузях та визначимо наскільки кожен з них відповідає задачі дослідження.

Будь-які методи обробки даних так чи інакше використовуються для структурування та аналізу існуючої інформації. Завдань з аналізу інформації багато, проте ми будемо розглядати лише методи, які ефективно працюють для вирішення задачі щодо структурування даних з достатньої кількістю різнорідних параметрів.

2.1.2 Кластерний аналіз

Кластерний аналіз – це мистецтво пошуку груп у даних. Щоб побачити, що мається на увазі під цим, розглянемо рисунок 2.1. Це графік з восьми об’єктів, на якому вимірювалися дві змінні: вага об’єкта (weight, kg) на горизонтальній осі, а його висота (height, cm) на вертикальній. Оскільки цей приклад містить лише дві змінні, ми можемо дослідити його просто дивлячись на нього візуально.

У цьому невеликому наборі даних чітко виділяються дві різні групи об’єктів, а саме {*TIN*, *TAL*, *KIM*, *ILA*} та (*LIE*, *JAC*, *PET*, *LEO*}. Такі групи називаються кластерами, і виявлення їх є метою кластерного аналізу [16]. По суті, задачею є формування груп таким чином, щоб об’єкти, які знаходяться в

одній групі були схожі між собою, тоді як об’єкти в різних групах були несхожі, наскільки це можливо [17].

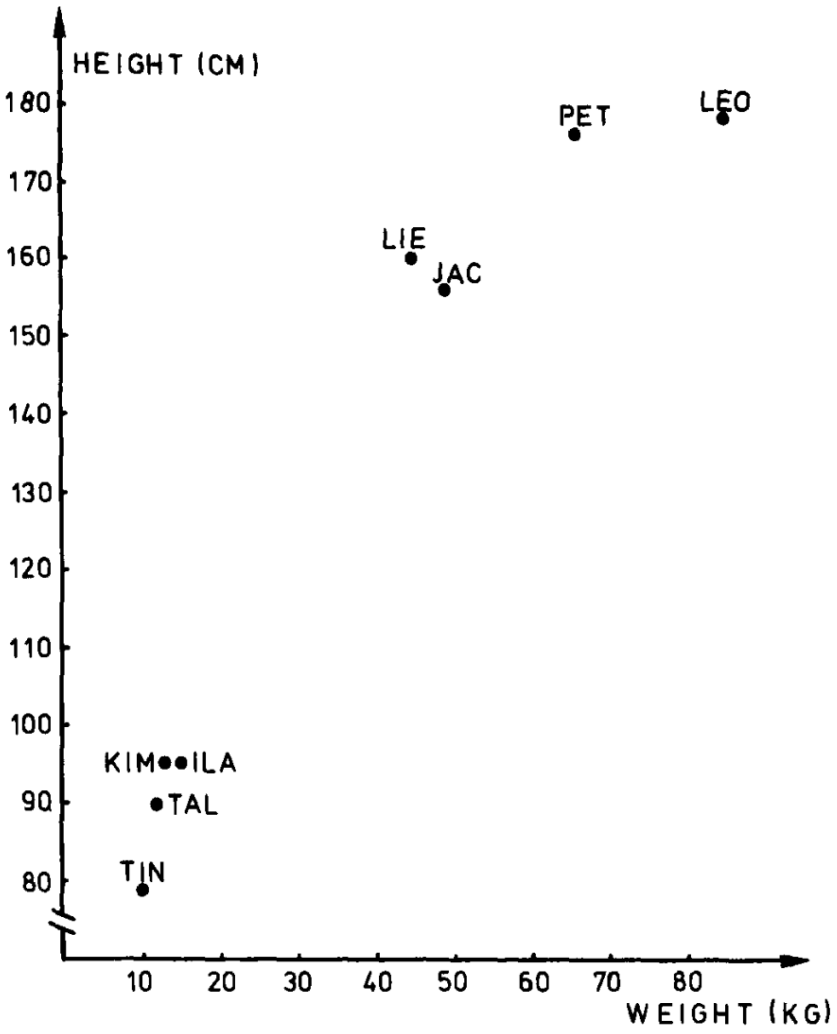

Рисунок 2.1 – Графік кластеризації з вісьма об’єктами

Наведемо ще один приклад: при сегментації ринку можна кластеризувати споживачів за двома параметрами – ціни і якості. Припустимо, компанія – виробник автомобілів провела опитування споживачів, в якому задавала два питання: «За яку ціну Ви готові купити автомобіль?» і «Оцініть якість автомобіля *X* за 50-бальною шкалою». Приклад даних отриманих в результаті опитування можна побачити в таблиці 2.1.

Таблиця 2.1 – Приклад результату опитування

Номер учасника опитування	Ціна, тис. $	Якість автомобіля X
1	27	19
2	11	46
3	25	15
4	36	27
5	35	25
6	10	43
7	11	44
8	36	24
9	26	14
10	26	14
11	9	45
12	33	23
13	27	16
14	10	47

Якщо подивитися на діаграму розсіювання «ціни – якості», представлену на рисунку 2.2, то відразу буде видно групи споживачів:

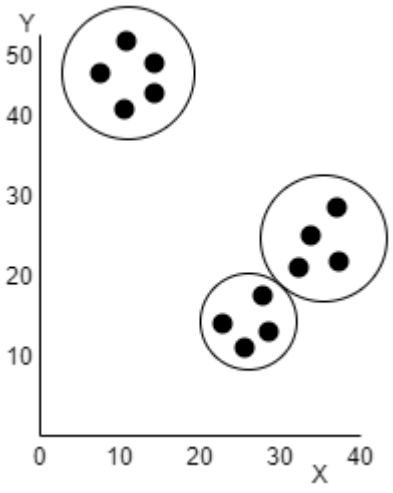

Рисунок 2.2 – Співвідношення ціни – якості

Володіючи цією інформацією, кожній групі споживачів можна запропонувати саме те, що необхідно саме цій групі, і за рахунок цього збільшити рівень продажів компанії.

Зрозуміло, в реальному житті кластери, помітні оком, зустрічаються нечасто, набагато частіше бувають ситуації, коли всі результуючі параметри змішуються в одну «купу». Цей ефект можна побачити на рисунку 2.3.

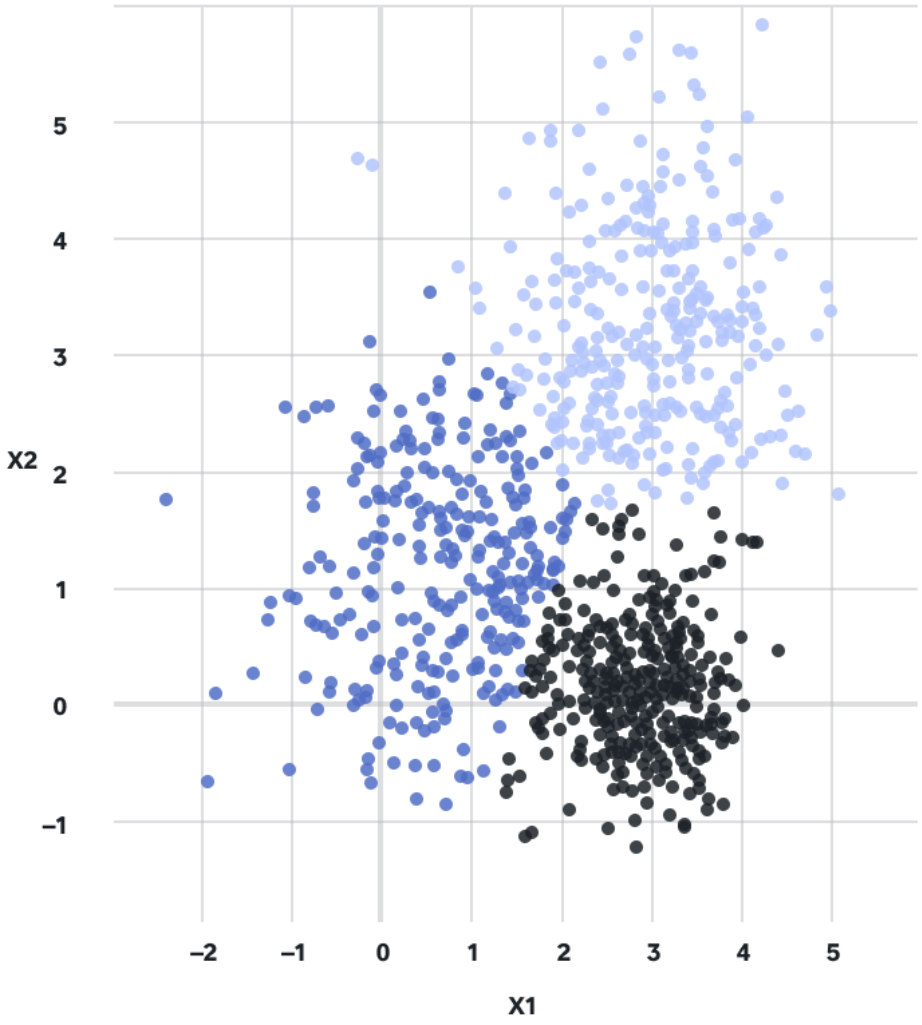

Рисунок 2.3 – Приклад реалістичної діаграми розсіювання

Особливо часто це зустрічається, коли аналізованих параметрів було не два, а кілька десятків (кластерний аналіз не обмежує число аналізованих параметрів, тому можна розглядати всю проблему комплексно) [18].

У минулому кластеризація зазвичай виконувалась суб’єктивно, спираючись на сприйняття та судження дослідника. У прикладі відображеному на рисунку 2.1 були використовувані очі та мозок людини, які дуже добре підходить для класифікації до трьох вимірів. Однак необхідність класифікувати випадки більш трьох вимірів та майбутні стандарти об’єктивності сучасної науки породили так звані автоматичні процедури класифікації.

За останні 30 років була розроблена велика кількість алгоритмів та комп’ютерних програм для кластерного аналізу. Є дві причини розвитку такої різноманітності методів. По-перше, автоматична класифікація є доволі молодою науковою дисципліною. Це можна побачити з тисяч статей розкиданих у різноманітних періодичних виданнях (переважно журнали статистики, біології, психометрія, інформатика та маркетинг). Друга, основна причина різноманітності алгоритмів полягає в тому, що не існує загального визначення кластера, і насправді їх декілька видів: сферичні кластери, витягнуті кластери, лінійні кластери тощо [19]. Більш того, різні програми використовують різні типи даних, такі як безперервні змінні, дискретні змінні, подібності та несхожості. Тому потрібні різні методи кластеризації, щоб пристосуватися до типу програми та типу шуканих кластерів.

2.1.3 Розвідувальний факторний аналіз

Потреби психологів, що шукали вправний та акуратний опис інтелектуальних здібностей людини, привели до розвитку факторних аналітичних методів. Гальтон, вчений протягом 19-20 століть, заклав основи факторної аналітики шляхом розробки кількісних методів для визначення взаємозалежності між двома змінними. Карл Пірсон був першим, хто чітко визначив факторний аналіз. А у 1902 році Макдоннелл першим хто

опублікував застосування факторного аналізу у порівнянні фізичних характеристик між 3000 злочинцями та 1000 студентами Кембриджа [20].

Факторний аналіз можна охарактеризувати як впорядковане спрощення взаємопов'язаних вимірювань. Традиційно факторний аналіз був використаний для дослідження можливої базової структури набору взаємопов'язаних змінних без нав'язування будь-яких заздалегідь продуманих структур результату [21]. Виконуючи розвідувальний факторний аналіз (EFA) можна визначити кількість конструкцій та основну структуру факторів.

Основні особливості:

– це метод змінної зменшення, який визначає кількість прихованих конструкцій та основну структуру факторів набору змінних [22];

– висуває гіпотезу основної конструкції, змінної, яка не вимірюється безпосередньо;

– оцінює фактори, що впливають на реакцію спостережуваних змінних;

– дозволяє описати та визначити кількість прихованих конструкцій (факторів);

– включає унікальні фактори, похибка через ненадійність вимірювань.

Задачі аналізу:

– допомогти слідчому визначити кількість прихованих конструкцій, що лежать в основі набору елементів (змінних);

– забезпечити засіб пояснення варіацій між змінними (елементами) за допомогою кількох новостворених змінних (фактори);

– для визначення змісту або значення факторів, наприклад, прихованих конструкцій.

Обмеження методу:

– кореляції, основи факторного аналізу, описують взаємозв'язки, але причинних висновків не можна робити не лише з кореляцій;

– надійність вимірювального підходу;

– обсяг вибірки (більша вибірка – більша кореляція): мінімальна кількість випадків для отримання достовірних результатів – це більше 100 спостережень і в 5 разів більше факторів. Оскільки деякі випробувані можуть підійти не під кожен фактор, бажана більша вибірка [23]. Наприклад, 30 факторів потребують щонайменше 150 випадків (5 * 30);

– змінні можуть бути конкретними для вибірки, наприклад, унікальна якість, якою володіє група, не є узагальненою для населення;

– ненормальний розподіл даних.

2.1.4 Підтверджуючий факторний аналіз

CFA дозволяє досліднику перевірити гіпотезу про існування зв’язку між спостережуваними змінними та їхніми прихованим конструкціями. Дослідник використовує знання теорії, емпіричне дослідження або те й інше, апріорі обумовлює модель взаємозв’язків, а потім перевіряє гіпотезу статистично [24].

На використання CFA можуть впливати:

– вимога достатнього обсягу вибірки (наприклад, 5-20 випадків на оцінку параметра);

– вимірювальні прилади (системи);

– багатовимірна нормальність;

– ідентифікація параметрів;

– відсутні дані;

– інтерпретація індексів придатності моделі [25].

Запропонований підхід використання CFA складається з наступних процесів:

– переглядання відповідної теоретичної та дослідницької літератури для підтримки специфікації моделі;

– вказання моделі (наприклад, діаграма, рівняння);

– визначення ідентифікації моделі (наприклад, чи можна знайти унікальні значення для оцінки параметрів, кількість ступенів свободи та інше);

– збирання даних;

– проведення попереднього описового статистичного аналізу (наприклад, масштабування, виявлення відсутніх даних, проблеми колінеарності, виявлення відхилень);

– оцінка параметрів моделі;

– оцінка відповідності моделі;

– представлення та інтерпретування результатів.

Традиційні статистичні методи зазвичай використовують один статистичний тест для визначення значимості аналізу [26]. Однак моделювання структурних рівнянь (SEM), зокрема CFA, спирається на кілька статистичних тестів для визначення адекватності моделі, що відповідає даним. Тест *chi*-квадрат вказує на величину різниці між очікуваною та спостережуваною матрицями коваріації. Значення *chi*-квадрат, близьке до нуля, вказує на незначну різницю між цими матрицями. Крім того, рівень вірогідності повинен бути більше 0,05, коли *chi*-квадрат наближається до нуля.

Порівняльний показник відповідності (CFI) дорівнює функції невідповідності, скоригованій для обсягу вибірки. CFI коливається від 0 до 1, більше значення вказує на кращу відповідність моделі. Прийнятна відповідність моделі позначається значенням CFI 0,90 або більше [27].

Середньоквадратична похибка наближення (RMSEA) пов’язана із залишковим значенням у моделі. Значення RMSEA варіюються від 0 до 1, де менше значення вказує на кращу відповідність моделі. Прийнятна відповідність моделі позначається значенням RMSEA 0,06 або менше.

Якщо відповідність моделі прийнятна, оцінюються параметри. Відношення кожної оцінки параметра до його стандартної похибки розподіляється як *z*-статистика і є значною на рівні 0,05, якщо її значення

перевищує 1,96, а на рівні 0,01, якщо значення перевищує 2,56. Нестандартизовані оцінки параметрів зберігають інформацію про масштабування змінних та можуть бути інтерпретовані лише з посиланням на шкали змінних. Стандартизовані оцінки параметрів є перетвореннями нестандартизованих оцінок, що усувають масштабування та можуть бути використані для неформальних порівнянь параметрів у всій моделі. Стандартизовані оцінки відносяться до оцінок розміру ефекту.

Якщо при CFA виявляється неприйнятна відповідність моделі, замість цього може бути застосований EFA [28].

CFA та EFA мають свої подібності:

– обидва методи базуються на лінійних статистичних моделях;

– статистичні тести, пов’язані з обома методами, є дійсними, якщо виконуються певні припущення;

– обидва методи передбачають нормальний розподіл;

– обидва включають виміряні змінні та приховані конструкції.

Але, в той же час, є і відмінності, наприклад, CFA вимагає специфікації:

– моделі апріорі;

– кількості факторів;

– які елементи навантажують кожен фактор;

– моделі, підтриманої теорією або попередніми дослідженнями;

– явної помилки.

Коли EFA:

– визначає факторну структуру (модель);

– пояснює максимальну величину відхилення.

Загалом, CFA та EFA – це потужні статистичні методи, які можуть бути застосовані в багатьох сферах, де потрібне «вимірювання» даних, наприклад шкала задоволеності, ставлення до здоров’я, анкета обслуговування клієнтів.

2.1.5 Нейронні мережі

У 1943 році була опублікована робота МакКаллока і Пітса, в якій вперше було висунуто поняття нейронних мереж як інструменту для аналізу даних та була запропонована модель штучного нейрона [29].

У роботі зазначено передбачення, що моделюючи нейронну структуру мозку, можливо наблизитися до реалізації штучного інтелекту. Вже у той час здавалося, що використовуючи модель мозку людини, а саме його особливих біологічних клітин – нейронів, можна буде побудувати мережу, що дозволить вирішувати складні завдання, які ми вирішуємо кожного дня.

З того моменту, інтерес до нейронних мереж періодично зростав та спадав. Це було обумовлено успіхом нових відкриттів та досліджень в цій області. Зараз же, нейронні мережі приваблюють все більше і більше дослідників та можуть бути застосовані майже до будь-якої області діяльності. Це значно сприяло їх розвитку як популярного інструменту для аналізу даних.

На думку фахівців Дослідницького центру IBM [30], нейронні мережі можна класифікувати за типом завдання, а саме наступним чином:

– класифікація образів – до відомого застосування відноситься розпізнавання букв, мови, класифікація сигналу електрокардіограми, клітин крові і таке інше;

– кластеризація – застосовується для вилучення знань, стиснення даних і дослідження їх властивостей;

– апроксимація функцій – популярним прикладом є шумозаглушення при прийомі сигналу будь-якої природи, незалежно від трансльованої інформації;

– передбачення – основним прикладом є передбачення цін на фондовій біржі або прогноз погоди;

– оптимізація – наприклад, призначення працівників до роботи за рядом вмінь та інших факторів;

– асоціативна пам’ять – асоціативні з’єднання становляться доступними за вказівкою заданого змісту, та можуть бути сформовані навіть по частковому входу або спотвореному змісту, найкращим прикладом є мультимедійні інформаційні бази даних;

– управління – прикладом є системи оптимального управління двигуном, рульове керуванням кораблів, літаків та інше.

Загалом, схема процесу аналізу даних за допомогою нейронних мереж складається з 5 етапів:

– вибір типології мережі – на даний момент існує 9 типів мереж [31], на цьому етапі підбирається найбільш відповідний під клас задачі тип мережі;

– експериментальний підбір характеристик мережі – після вибору типу необхідно підібрати структуру мережі, а саме кількість нейронів, їх ваги, взаємозв’язки і таке інше;

– експериментальний підбір параметрів навчання – експериментально визначаються параметри навчання: максимальний час навчання, кількість даних, максимально припустима похибка і тому подібне;

– навчання мережі – за допомогою навчальної вибірки проводиться навчання мережі, передбачається, що вибірка містить в собі достатньо повну інформацію, що зможе охарактеризувати дані в цілому;

– перевірка адекватності навчання – проводиться аналіз отриманих результатів на даних, які не входили в навчальну вибірку, здійснюється ручний контроль результатів роботи нейронної мережі.

Для опису роботи нейронної мережі припустимо, що на вхід даються набори чисел та для кожного з них відомо значення функції, яке вона має для даного набору. Як приклад, зазначмо, що значенням є курс обміну деякої валюти на майбутній день. У цьому випадку вхід – рівень цього курсу та рівень деяких інших фінансових показників за, скажімо, останній місяць. Для іншого прикладу, зазначмо вхідний вектор – характеристики позичальника банку, а результат – чи виконав він умови повернення кредиту.

В обох цих випадках мова йде про історичні дані. Але, надалі були пред’явлені нові дані: значення фінансових показників на поточний день та інформація про вже нового клієнта, який звернувся з таким самим проханням надання кредиту. Результат тепер невідомий, і потрібно його приблизно визначити, а саме: яким буде курс обміну завтра, та чи є перспективним для банку цей новий клієнт.

В цій ситуації нейронна мережа виконує наступні функції. За елементарна операцію, якою вона оброблює дані, береться «зважена» сума вхідних величин (тобто сума, взята з деякими коефіцієнтами, які називаються вагами). Потім отримана величина перетворюється за допомогою нелінійної монотонної функції (функції активації) так, щоб отримане в результаті значення знаходилось в інтервалі від 0 до 1 [32]. Ця конструкція називається штучним нейроном. Сама мережа складається з багатьох таких нейронів, та потрібно зазначити, що частина з них обробляє безпосередньо вхідні дані (перший шар нейронів), а інші – сигнали, отримані на виході з нейронів першого шару і т.д., і, нарешті, є єдиний вихідний нейрон, який і видає результат. Також ваги, відповідні різним нейронам, та самі параметри функцій активації можуть змінюватися незалежно одні від одного. Обробляючи історичні або, як правильно їх називати, навчальні дані, і змінюючи при цьому ваги, мережа працює найкращим чином для того, щоб пристосувати свій вихідний сигнал до вже відомого результату. Цей процес називається навчанням нейронної мережі. Після того як воно закінчено, на вхід можна давати нові дані та отримувати новий прогноз.

Головним мінусом нейронних мереж є те, що процес навчання і процес прийняття рішень мережі є абсолютно неконтрольованим. Іншими словами, нейронна мережа працює за принципом «чорного ящику», на вході якого подаються дані, а на виході виходить результат. Як саме працює всередині себе нейронна мережа, зрозуміти неможливо, оскільки тестові дані аналізовані в момент навчання нейронної мережі обробляються автоматично за допомогою мінімізації помилки та змін внутрішні параметрів (ваг). І

незважаючи на те, що отримати значення ваг в навченій мережі можливо, ніяких пояснень щодо сенсу цих ролей та важливості ваг в отриманні результату немає.

2.1.6 Дерева рішень

Перші згадки дерев рішень з’явилися в роботах Ховленда і Ханта кінця 50-х років XX сторіччя, але основний розвиток цієї ідеї почався після опублікування книги «Experiments in Induction» [33] в 1966 році.

Дерева рішень – це спосіб представлення правил в ієрархічній послідовній структурі, яка дає можливість зробити логічне співвідношення об’єкту або ситуації на вході, з одним або декількома вихідними вузлами. Під одним правилом мається на увазі логічна конструкція, представлена у вигляді «якщо ... то». Розглянемо наступну задачу: необхідно побудувати вирішальне правило для визначення можливості видачі кредиту фізичної особі. Приклад такого дерева рішень зазначений на рисунку 2.3.

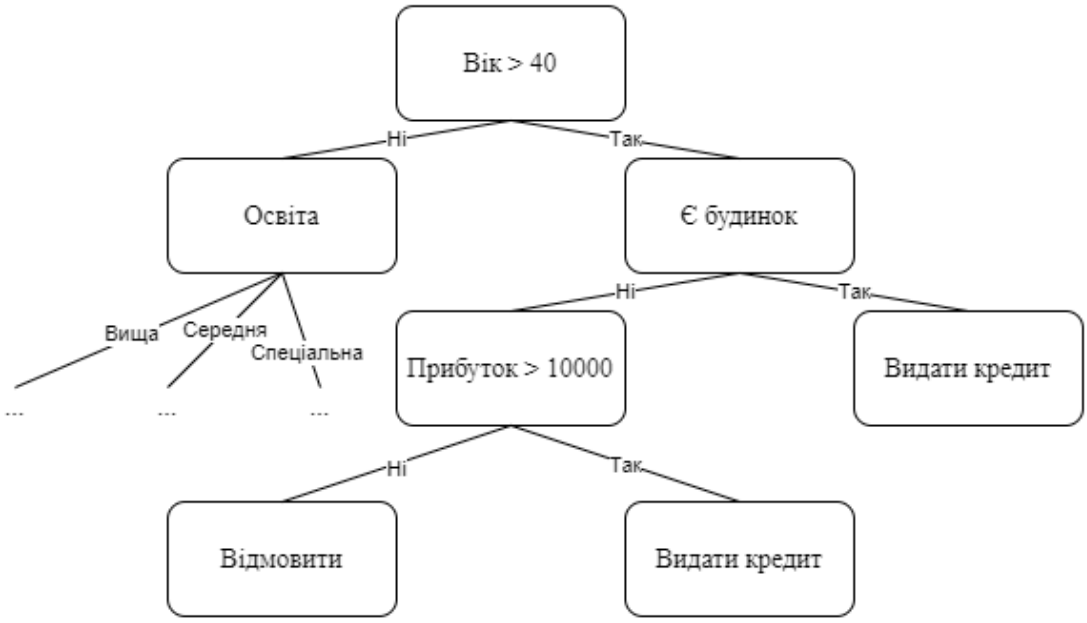

Рисунок 2.3 – Чітке дерево рішень

У цьому прикладі наведено так зване чітке дерево рішень. Але існує й інший тип дерев рішень, що представляє неменший інтерес, а саме ймовірні дерева рішень. Приклад такого дерева можна побачити на рисунку 2.4, в якому кожен параметр прийняття рішення входить в результуюче рішення з певною ймовірністю.

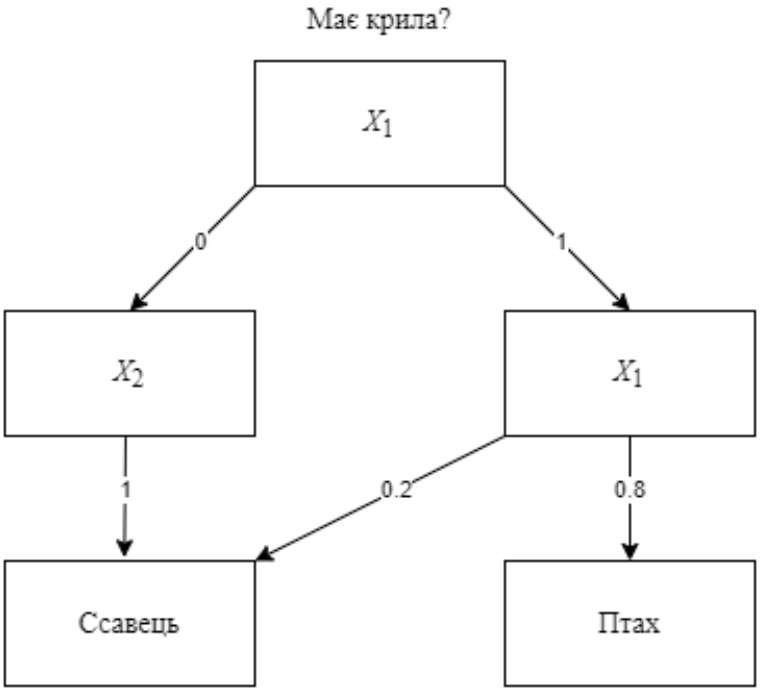

Рисунок 2.4 – Ймовірне дерево рішень

Метод дерев рішень може допомогти при прийнятті складного рішення, на яке впливають десятки параметрів.

Дерева рішень широко застосовуються в багатьох областях діяльності:

– банківська справа – оцінка кредитоспроможності клієнтів банку;
– промисловість – контроль за якістю продукції (виявлення дефектів);
– медицина – діагностика різних захворювань;
– молекулярна біологія – аналіз будови амінокислот;
– консалтинг.

І це далеко не повний список областей, де можна використовувати дерева рішень. Багато потенційних можливостей застосування цього інструменту ще не були досліджені.

2.1.7 Регресійний аналіз

Регресійний аналіз – це концептуально простий метод дослідження функціональних зв'язків між змінними. Наприклад, оцінювач нерухомості хоче визначити ціну продажу будинку за вибраними фізичними характеристиками будівлі та податками на неї. У той час соціальний дослідник може побажати визначити, чи є споживання сигарет пов'язаним з різними соціально-економічними та демографічними змінними, такими як вік, освіта, дохід та ціна сигарет. Такі відношення виражаються в формі рівняння або моделі, що з'єднує відповідь або залежну змінну та одну або кілька пояснювальних чи передбачуваних змінних [34]. У прикладі дослідника, змінною відповіді є споживання сигарет (вимірюється числом пачок сигарет, проданих у певному городі на душу населення протягом року), а пояснювальні або передбачувані змінні – це різні соціально-економічні та демографічні факти. У той час, у прикладі оцінки нерухомості змінною відповіді є ціна будинку, а пояснювальні змінні – це характеристики будівлі та податки, сплачені за неї.

Позначимо змінну відповіді як Y, а набір передбачуваних змінних як $X_1, X_2, ..., X_p$, де p позначає кількість передбачуваних змінних. Повноцінну залежність між Y і $X_1, X_2, ..., X_p$ можна апроксимувати регресійною моделлю:

$$Y = f(X_1, X_2, ..., X_p) + \varepsilon, \tag{2.1}$$

де ε – випадкова помилка, що представляє невідповідність наближення.

Це враховує неможливість моделі точно відповідати даним. Функція $f(X_1, X_2, ..., X_p)$ описує взаємозв'язок між Y та $X_1, X_2, ..., X_p$. Прикладом є лінійна регресійна модель [35]:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p + \varepsilon, \quad (2.2)$$

де $\beta_0, \beta_1, ... \beta_p$ – це параметри регресії або коефіцієнти, що є невідомими константами які визначаються з даних.

Регресійний аналіз має безліч областей застосування. Частковий список містив би економіку, фінанси, бізнес, право, метеорологію, медицину, біологію, хімію, інженерію, фізику, освіту, спорт, історію, соціологію та психологію.

2.1.8 Дискримінантний аналіз

Дискримінаційний аналіз – це статистичний прийом, який дозволяє досліднику вивчати відмінності між двома або більше групами об'єктів відносно кількох змінних одночасно [36].

У соціальних науках існує широкий спектр ситуацій, в яких ця техніка може бути корисною. Розглянемо, наприклад, дослідницьку групу, якій було доручено вивчити результати захоплення об'єкту терористами із залученням заручників. В основному, вони хочуть знати, які елементи ситуації могли б передбачити безпечне звільнення заручників, не виконуючи вимоги терористів. Вони висувають гіпотезу про те, що існують декілька змінних, що можуть бути визначниками безпечного звільнення заручників без поранень або страт. Серед цих змінних: кількість терористів, сила їхньої підтримки у місцевому населенні, чи є вони незалежною групою або членами більшої бойової організації, тон їхньої риторики, тип і кількість зброї,

співвідношення терористів до заручники тощо. Вивчаючи попередні випадки, коли влада відмовлялась задовольнити вимоги терористів, дослідники сподіваються визначити:

– які, якщо такі є, з цих змінних є корисними для прогнозування долі заручників;

– як ці змінні можна об’єднати в математичне рівняння для прогнозування найбільш вірогідного результату;

– точність отриманого рівняння.

Дискримінаційний аналіз може надати необхідні докази. Якщо минулі випадки безпечного звільнення заручників дійсно відрізняються за досліджуваними змінними від тих, в яких заручники були поранені, тоді рівняння прогнозування буде корисним для органів влади, які вирішують проблему терористичної діяльності.

Інші сфери, в яких ця техніка була вигідно застосована, включають тестування на працевлаштування персоналу, поіменний аналіз законодавчих органів, психологічне тестування дітей, наслідки лікування, економічні відмінності між географічними регіонами, прогнозування поведінки виборців та багато іншого [37].

Основними передумовами аналізу є існування двох або більше груп, які відрізняються за кількома змінними, а також факт того, що ці змінні можуть бути виміряні на рівні інтервалу або співвідношення. Тоді дискримінантний аналіз може допомогти проаналізувати відмінності між групами та забезпечить засіб для віднесення (класифікації) будь-якого випадку до найбільш схожої групи.

2.1.9 Кореляційний аналіз

Звичайний канонічний кореляційний аналіз досліджує ступінь взаємозв’язку між двома наборами змінних. Фактично, аналіз здійснюється

шляхом первинного зведення оцінок кожного об’єкту дослідження щодо змінних у кожному наборі, що надалі включаються до єдиної складової змінної. Проста або двовимірна кореляція між двома складеними оцінками (по одній для кожного з двох наборів змінних) є канонічною кореляцією [38].

Однак складені оцінки обчислюються з урахуванням особливих обмежень. Вони виводяться, щоб максимізувати взаємозв’язок між двома наборами змінних, які вони представляють. Ця «оптимізація» виконується шляхом «зважування» даних кожного об’єкту, а потім підсумовування зважених балів у кожному наборі змінних. Ваги можуть бути як негативними, так і позитивними числами, та просто помножуються на бали для кожного об’єкта. Ці ваги називаються функціональними коефіцієнтами [39] і є такими ж, як бета-ваги в регресійному аналізі або коефіцієнти закономірності в факторному аналізі. Композити, отримані з використанням цих «найкращих можливих» ваг, називаються варіативними балами. Квадратний канонічний коефіцієнт кореляції вказує на частку дисперсії, яку два композити, отримані з двох наборів змінних, ділять лінійно.

У найзагальнішому вигляді, прийняття гіпотези про наявність кореляції означає, що зміна значення змінної *А* станеться одночасно з пропорційною зміною значення *Б*: якщо обидві змінні зростають, то кореляція позитивна; якщо одна змінна зростає, а друга зменшується – кореляція негативна [40].

При вивченні кореляцій намагаються встановити, чи існує якийсь зв’язок між двома показниками в одній вибірці (наприклад, між ростом і вагою дітей або між рівнем IQ і шкільною успішністю) або між двома різними вибірками (наприклад, при порівнянні пар близнюків), і якщо цей зв’язок існує, то чи супроводжується він збільшення одного показника зростанням (позитивна кореляція) або зменшенням (негативна кореляція) іншого.

2.1.10 Обґрунтування вибору метода

Розглянувши більшість популярних методів аналізу даних, можемо визначити з них найбільш відповідний для програмної реалізації системи роботи.

Для цього будемо спиратися на факт того, що система повинна бути універсальною – легко працювати з даними з різних типів ринків. Це свідчить про те, що незважаючи на варіативну кількість вхідних параметрів, вони повинні бути змістовно трансформовані та зменшені до більш узагальненого вигляду.

Також, обробка цих параметрів повинна бути максимально прозорою та зрозумілою для споживача, бо неясність обробки даних (наприклад в нейронних мережах) збільшує поріг входження та зменшує кількість кінцевих користувачів системи. Крім цього, завдяки трансформації параметрів до більш узагальненого виду, аналіз пріоритетів та корисності кожного з них стає нерелевантним.

З зазначеного вище, можем зробити висновок, що метод чіткого дерева рішень є найбільш відповідним до умов реалізації системи, а саме: працює з невеликою кількістю параметрів, процес обробки є зрозумілим для користувача та сам аналіз не проводить оцінку параметрів, та дає на вихід легкі для сприйняття результати.

2.2 Формування вимог до системи

Загалом, існує дві основні групи, на які можна поділити вимоги до інформаційної системи, а саме функціональні та нефункціональні вимоги.

До функціональних вимог відносяться наступні вимоги:

– система повинна бути реалізована у вигляді сервісу для можливості розгортання на сервері;

– система повинна виконувати свої функції завдяки певному запиту на URL-адресу сервера;

– система повинна мати можливість працювати з різними базами даних ринків одночасно;

– система повинна застосовувати API з запитами для отримання інформації з баз даних підключених ринків;

– система повинна трансформувати отримані дані в узагальнені об’єкти;

– система повинна реалізовувати метод чіткого дерева рішень для аналізу отриманих даних;

– система повинна надавати користувачу можливість задавати свої вузли дерева рішень;

– система повинна надавати користувачу інформацію про купівлю/продаж товарів на ринку;

– система повинна надавати користувачу детальний опис товару;

– система повинна надавати користувачу інформацію про інших користувачів ринку;

– система повинна мати функціонал обробки помилок з виведенням зрозумілого повідомлення про проблему;

– система повинна мати можливість легкого масштабування для підключення нових API;

Також, до інформаційної системи висунуто ряд вимог нефункціонального характеру, які здебільшого базуються на стабільності та швидкості:

– швидкість відгуку – система має опрацьовувати запити користувача зі швидкістю часу відгуку не більше 3 секунд;

– надійність роботи – система не повинна зупиняти свою роботу у буть-якому разі та має обробляти будь-які запити та видавати відповідні повідомлення помилок у разі невірної інформації;

– багатопоточність – система повинна підтримувати можливість обробки та виконання своїх функцій при запитах з декількох машин одночасно.

2.3 Вибір інструментів розробки

Виходячи з переліку сформованих вимог до системи, для розробки робочого прототипу програмної реалізації були вибрані такі інструменти: мова програмування Java, Spring Framework та середа розробки Intellij Idea.

Програмування на Java є зручним та гнучким, що робить його очевидним вибором мови програмування для розробників веб-додатків та експертів з управління програмами. Під гнучкістю в цьому випадку мається на увазі те, що програма, розроблена в її системі кодування, може безперебійно працювати в будь-якій операційній системі, незалежно від системи, в якій вона була розроблена спочатку.

Існує багато мов програмування, але Java домінує галузь з популярності вже протягом десятків років і хоча Python зайняв перше місце за кількістю застосування в проектах у 2018 році та зрушив Java з трону лідерства, більшість заснованих проектів до цього все ще використовують Java.

Частина того, чому Java настільки популярна полягає в тому, що вона є об’єктно-орієнтованою. Простіше кажучи, об’єктно-орієнтована мова кодування спрощує розробку програмного забезпечення, розбиваючи процес виконання на невеликі, прості в обробці фрагменти. Складні проблеми кодування, існуючі в C, C++ та інших мовах, важко зустріти при програмуванні на Java. Крім того, об’єктно-орієнтовані мови, такі як Java, надають програмістам більшу модульність та легкий для розуміння прагматичний підхід. У той час Java API надає розробникам тисячі класів та близько 50 ключових слів для роботи. Це робить його універсальним та

пристосованим до якомога більшої кількості ідей програмування, які може мати програміст.

Тому, враховуючи одну з вимог системи, а саме легке розширення за допомогою підключення додаткових API, Java є ідеальним вибором мови реалізації, завдяки своєму широкому розповсюдженню у різних галузях.

Також, одним з інструментів розробки був вибраний Spring Framework. Spring – це найпопулярніший механізм розробки додатків для корпоративної Java. Мільйони розробників по всьому світу використовують цей фреймворк для створення високопродуктивного та багаторазового коду.

Spring framework – це платформа Java з відкритим кодом, написана Родом Джонсоном і вперше випущена за ліцензією Apache 2.0 у червні 2003 року.

Основні особливості Spring Framework можна використовувати при розробці будь-яких додатків Java, але є розширення для створення веб-додатків поверх платформи Java EE. Мета фреймворку полягає в спрощенні розробки сервісів та сприянні використання передової практики програмування на основі POJO, що допомагає реалізувати висунуті вимоги до системи.

Для середи розробки системи було вибрано IntelliJ IDEA – це IDE на основі Java, яке широко використовується як компаніями-розробниками програмного забезпечення, так і індивідуальними розробниками. Воно заповнено високопродуктивними інструментами та можливостями, і багато з цих функцій працюють «з коробки», наприклад, підтримка Gradle, Maven, STS тощо. Програмне забезпечення спрямоване на підвищення продуктивності, надаючи надзвичайно інтуїтивну синтаксичну допомогу в коді, яка крім Java підтримує всі фреймворки та мови. IntelliJ IDEA аналізує код, шукаючи зв'язки між символами у всіх файлах проекту, використовуючи цю інформацію, воно надає допомогу в глибокому кодуванні, швидку навігацію та розумний аналіз помилок.

2.4 Проектування системної реалізації

У процесі виконання атестаційної роботи були враховані вимоги до системи та були спроектовані основні узагальненні сутності. Структуру сутностей та зв’язки між ними можна побачити на рисунку 2.5.

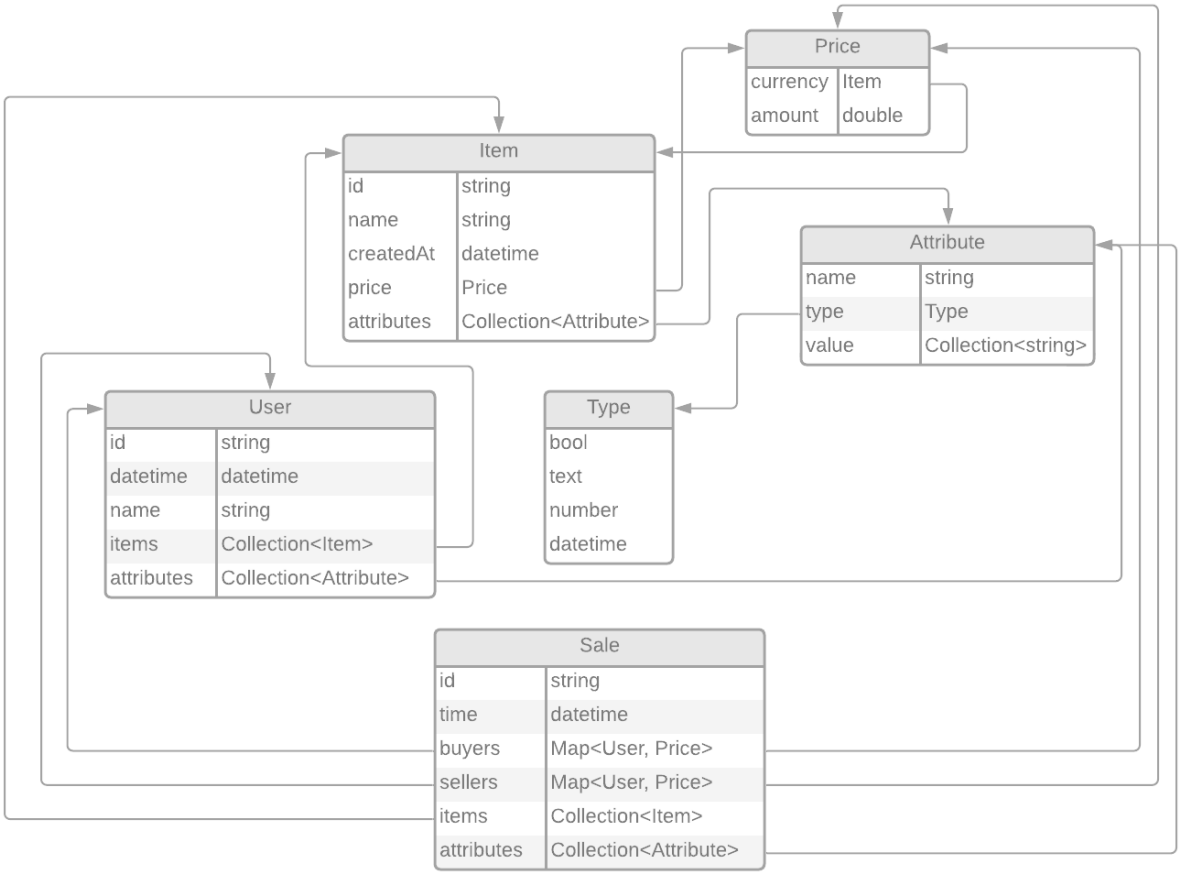

Рисунок 2.5 – Схема основних сутностей системи

Головною задачею у процесі проектування було запровадження до системи універсальності, а саме створення достатньо узагальнених сутностей (об’єктів) та функціоналу, задля того щоб спростити інтеграцію сторонніх API різних ринків. Для цього були висунуті три сутності, що описують основні елементи предметної області, а саме «Item», «User» та «Sale». Всі ці об’єкти мають поле «attributes» у вигляді колекції типу «Attribute», структуру якого можна побачити на рисунку 2.6.

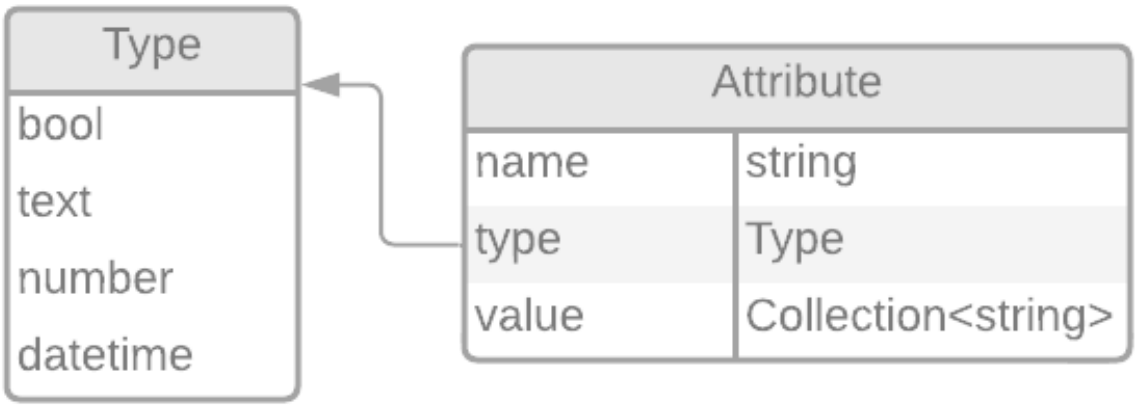

Рисунок 2.6 – Структура сутностей «Attribute» та «Type»

Цей об’єкт надає можливість додавання нових характеристик до сутностей, що його включають, та робить їх більш абстрактними загалом. Він складається з поля «name», що відповідає за назву атрибута, поля «type», яке описує тип даних характеристики та може бути одним з трьох видів: «bool» – логічний тип даних зі значенням true/false або 0/1, «text» – текстовий тип даних, «number» – числовий тип та «datetime» – тип дати та часу. А також об’єкт має змінну «value», яка містить значення атрибуту (одне або декілька).

Основна причина потреби цієї сутності полягає у проблемі того, що заздалегідь невідомо які саме ринкові API будуть застосовані в системі, а також невідома кількість та типи характеристик об’єктів у цих API. Для прикладу застосування цієї сутності припустимо, що API ринку повертає об’єкти двох товарів: мобільний телефон та молоко. У цьому випадку, у телефона і молока будуть різні по своєму типу та кількості характеристики. Але завдяки цій сутності, можливо легко перенести їх на узагальнену сутність «Item». Допустимо, що телефон має характеристику «колір» та значення «синій, червоний, чорний», а молоко має характеристику «дата виготовлення» зі значенням «20.05.2020», тоді для сутності «Attribute» значення змінних будуть заповнені як вказано в таблиці 2.2.

Таблиця 2.2 – Приклад заповненої сутності «Attribute»

Назва товару	name	type	value
Мобільний телефон	Колір	text	Синій, червоний, чорний
Молоко	Дата виготовлення	datetime	20.05.2020

Далі розглянемо одну з основних сутностей системи – «Item», структуру якої можна побачити на рисунку 2.7.

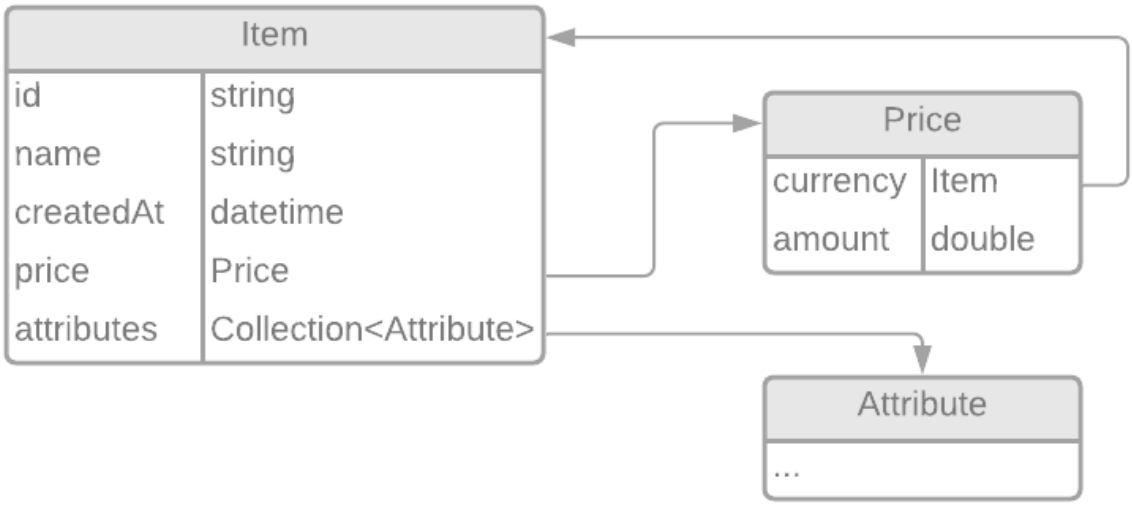

Рисунок 2.7 – Структура сутностей «Item» та «Price»

Цей об’єкт характеризує індивідуальний елемент ринку, наприклад товар. Він складається з поля «id», яке зазвичай визначається у різних ринкових API, поля «name», яке описує назву елементу, поля «createdAt», яке може описувати різноманітні часові характеристики ринкового елементу (наприклад дата пакування або час появи у магазині), поля «attributes», що було описано вище, та поля «price» – ціна елемента, яка в свою чергу також є окремим об’єктом, що складається з двох полів: «currency», яке само по собі є типом «Item» та «amount» – кількість елементів.

Щоб краще зрозуміти причину проектування ціни таким чином, приведемо приклад. Припустимо, що ми маємо об’єкт «Item» з заповненим полем «id» та полем «name» зі значенням «Гривня». Таким чином ми зможемо застосовувати цей об’єкт для заповнення сутності «Price» додатково вказав кількість гривень, що дозволить нам описати ціну товару.

Приклад заповненого об’єкта «Item» можемо побачити у таблиці 2.3.

Таблиця 2.3 – Приклад заповненої сутності «Item»

Назва поля	Значення
id	1
name	Мобільний телефон
createdAt	15.04.2020 13:44:53
price	Price (Гривня, 5400)
attributes	Attribute (Колір, text, Синій)

Наступна основна сутність – «User», яку можна побачити на рисунку 2.8. Вона описує користувача ринку, яким може виступати як фізична особа, так і підприємство, а також як у вигляді продавця, так і покупця. Об’єкт складається з поля «id» – ідентифікатор користувача, поля «datetime» часового типу, яке може бути інтерпретовано по-різному, залежно від потреб API, поля «name» – ім’я користувача, поля «items» – колекція об’єктів типу «Item», яка може використовуватися для відображення товарів, що продає користувач, та поля «attributes».

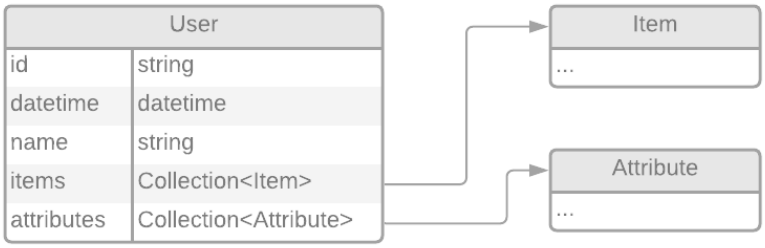

Рисунок 2.8 – Структура сутності «User»

Останньою основною сутністю системи є «Sale», відображена на рисунку 2.9. Це об’єкт, що характеризує головну операцію ринку – купівлю-продаж. Він складається з поля «id» – ідентифікатор операції, поля «time», яке відповідає за дату та час операції, полів «buyers» та «sellers», які характеризують залежність між покупцями/продавцями та їх прибутком/витратою, а також їх кількість. Бо, наприклад, товар може продавати одному покупцю пара продавців та ділити прибуток між собою особливим чином. Також в сутності існують поля «items» – товари, що були куплені/продані, та «attributes», для більш детального опису угоди.

Рисунок 2.9 – Структура сутності «Sale»

Також, в процесі проектування системи до неї була висунута наступна вимога: «система повинна надавати користувачу можливість задавати свої вузли дерева рішень», яка була реалізована за допомогою введення додаткових сутностей відображених на рисунку 2.10 та спеціалізованих процесорів. Сутність «Instruction» описує один вузол в дереві рішень та носить в собі результат обробки заданого «питання» в обох випадках: «так» чи «ні». Вона складається з поля «id» – ідентифікатор вузла, який задається користувачем для зв’язування з іншими вузлами, поля «name» – назва вузлу (в загальних випадках є «питанням»), поля «operation» – поле особливого типу «Operation», яке визначає яка саме операція обробки повинна виконуватися при роботі зі значенням інструкції (вузла), поля

«attributeName» – назва атрибуту сутності, яка буде оброблятися, поля «value» – значення яке буди використовуватися для обробки (не є обов’язковим у випадку операцій «SATISFY», «EXISTS» та «NOT_EXISTS»), поля «satisfyTree» – особливе поле, що використовується у операції «SATISFY», та містить у собі додаткове дерево рішень, поля «satisfyCondition» – поле, я якому вказується який саме результат з додаткового дерева рішень є вирішальним для задоволення операції «SATISFY», поля «dependentId» – ідентифікатор інструкції, від якої залежить даний вузол, поля «dependenceType» – який набір результатів батьківського вузла використовувати (0 – негативний, 1 – позитивний), поля «positiveResultSet» – набір результатів, який відповідає позитивній відповіді на «питання» та поля «negativeResultSet» – «негативний» набір результатів.

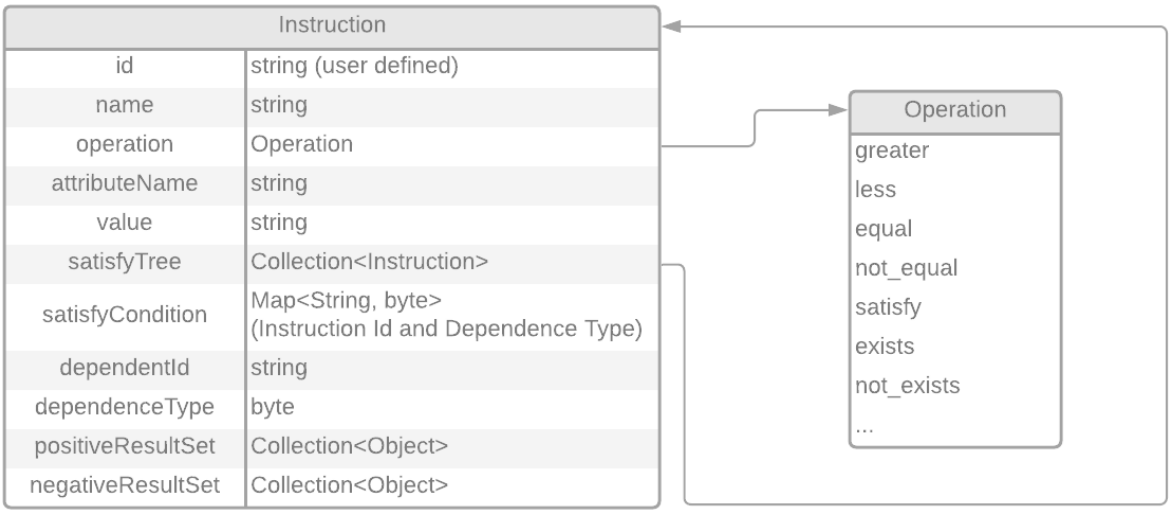

Рисунок 2.10 – Схема додаткових сутностей системи

Крім того, на рисунку 2.10 зазначений перелік «Operation». Для кожного пункту з цього переліку існує окремий процесор, який реалізує інтерфейс «AbstractOperationProcessor» та оброблює дані визначеним розробником способом. Стандартними операціями є:

– «GREATER/LESS» – чи повинно значення з даних бути більше або менше зазначеного в інструкції;

– «EQUAL/NOT_EQUAL» – чи повинно значення з даних дорівнювати або не дорівнювати значенню з інструкції;

– «EXISTS/NOT_EXISTS» – чи існує заданий в інструкції атрибут у даних;

– «SATISFY» – чи задовольняє вказаний атрибут сутності додаткове дерево умов.

Але розробник, який буде доповнювати систему новими API ринків може додати свої операції до цього переліку, розробивши відповідний процесор, або змінити функціонал існуючих процесорів використовуючи наслідування та перевизначення.

Узагальнену схему класів та роботи системи можна побачити на рисунку 2.11.

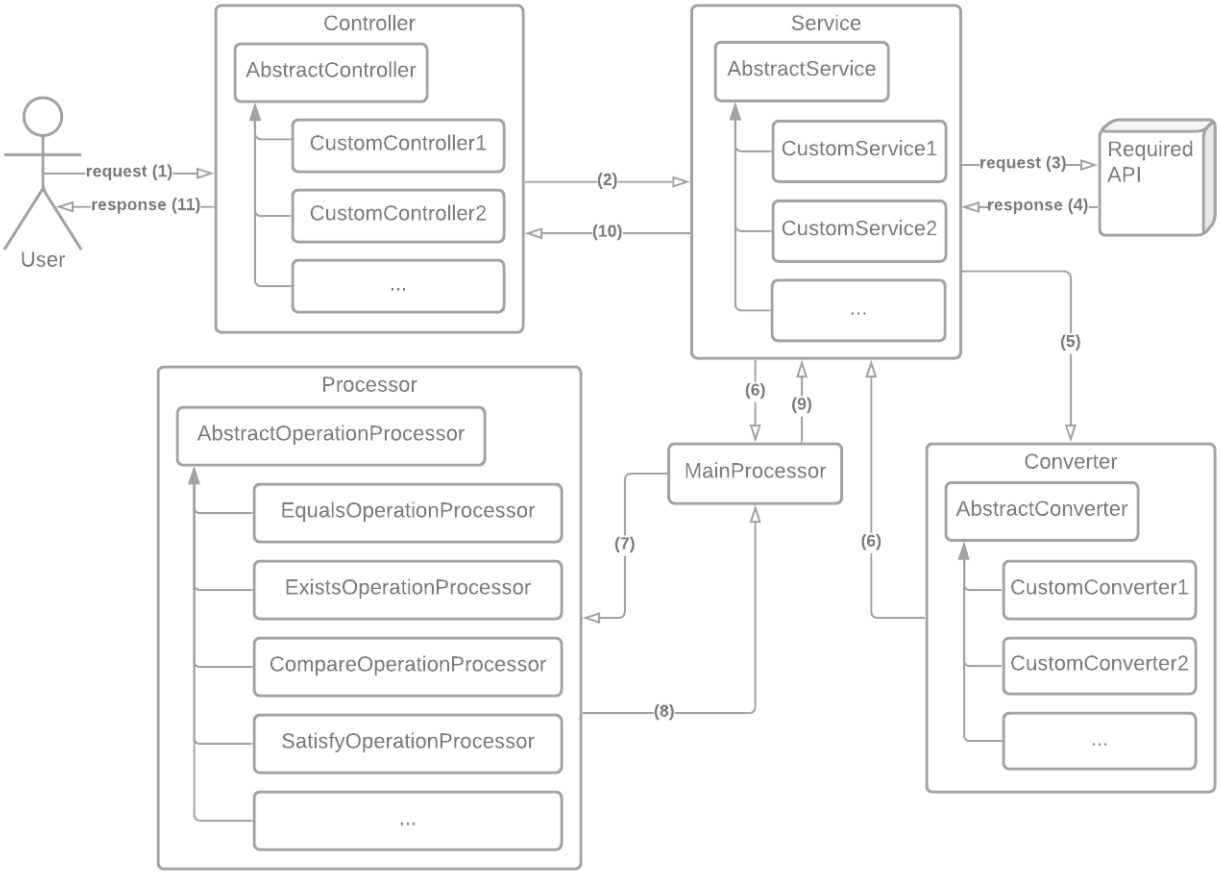

Рисунок 2.11 – Узагальнена схема роботи системи

3 ПРАКТИЧНИЙ РОЗДІЛ

3.1 Реалізація прототипу системи

Для наочного прикладу універсальності системи, був реалізований прототип з використанням API ринку комп’ютерної гри Path of Exile. Ця гра має у собі продуману систему обміну між гравцями, великий обсяг доступних для обміну предметів, зазначення різних типів валюти для купівлі та продаж, а також найголовніше – «живий» ринок аукціонного типу, повний контроль над цінами на якому мають самі гравці.

За допомогою IDE Intellij Idea та мови Java була реалізована система у вигляді сервісу за однією з варіацій шаблону MVC та використанням додаткового сервісного пласту (Service Layer). Також всі відповідні класи були реалізовані за правилами Spring Framework, використовуючи конфігурацію за допомогою анотацій. В результаті була сформована така структура пакетів:

– «config» – пакет з класами конфігурацій, який містить у собі клас «MainConfiguration» з настройкою RestTemplate для обробки та відправлення запитів та SwaggerUI для відображення інтерфейсу для роботи з запитами;

– «controller» – пакет пласту контролерів з внутрішнім пакетом «poe», який містить клас реалізації відповідного контролеру;

– «converter» – пакет з класами для трансформування об’єктів з API ринків у узагальнені об’єкти системи;

– «entity» – пакет пласту моделі з реалізацією усіх зазначених у проектуванні сутностей, а також окремі сутності «PoePrice» та «PoeResponse», виконані за правилами відповідного API;

– «processor» – пакет з класами процесорів, кожен з яких використовується в процесі обробки інструкцій у основному класі «MainProcessor»;

– «service» – пакет сервісного пласту для класів, що виконують задачу роботи з сторонніми API та зв’язування процесів контролеру, процесорів та конверторів.

Структуру пакетів та класи системи можна побачити на рисунку 3.1.

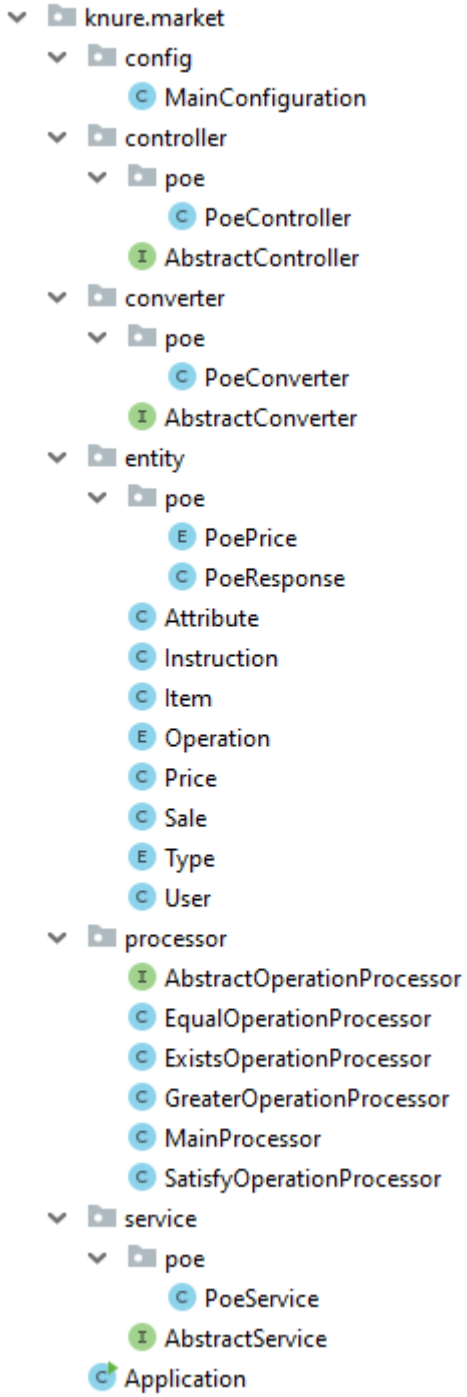

Рисунок 3.1 – Структура пакетів системи

3.2 Інструкція користувача

Так як система була реалізована у виді сервісу, для роботи з програмою користувач повинен перейти на URL http://[server-address]/swagger-ui.html (де server address – адрес сервера на якому розгорнута система) та за допомогою вбудованого інтерфейсу SwaggerUI заповнити об’єкт з інструкціями дерева для вибраного API ринку. Приклад інтерфейсу можна побачити на рисунку 3.2.

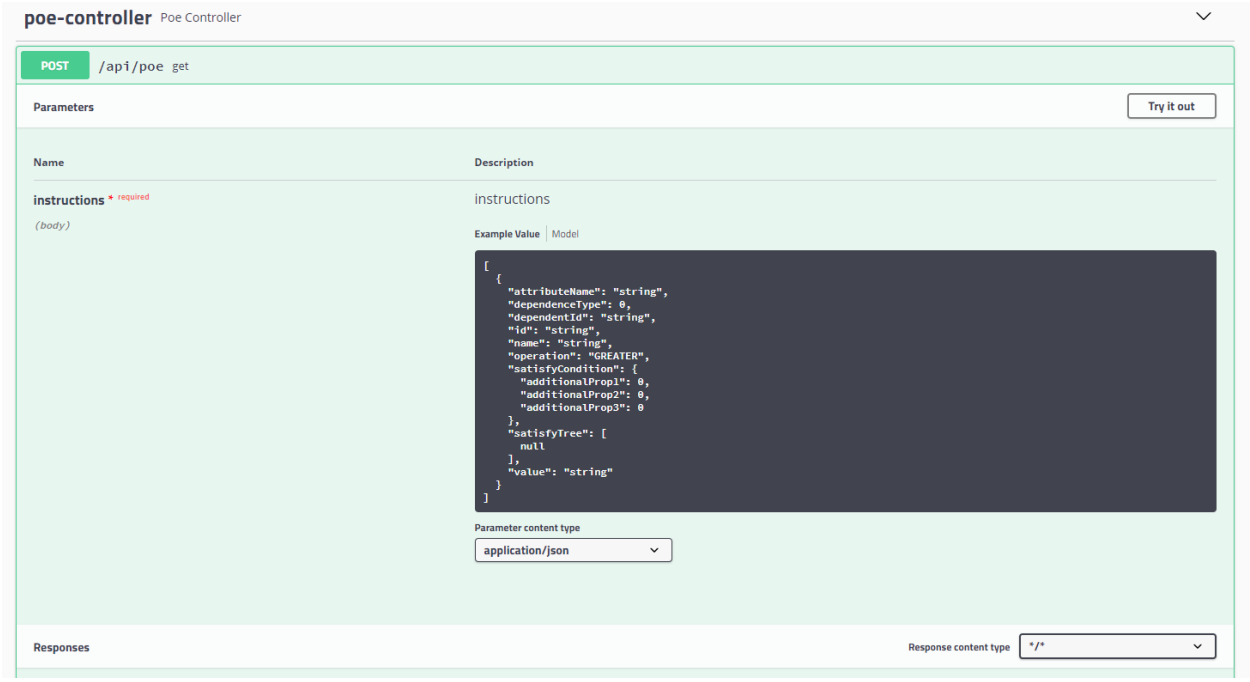

Рисунок 3.2 – Інтерфейс для роботи з запитами до системи

Після заповнення запиту до системи, користувач натискає кнопку «Execute» та отримує список результатів у вигляді об’єктів інструкцій, кожна з яких має відповідний набір даних, задовольняючий позитивний або негативний результат вузла. Таким чином, користувач може проаналізувати результати не тільки кінцевих вузлів дерева, але й усі інші результати, з якими працювала система.

3.3 Виконання програми

Для прикладу застосування було побудовано дерево рішень відображене на рисунку 3.3.

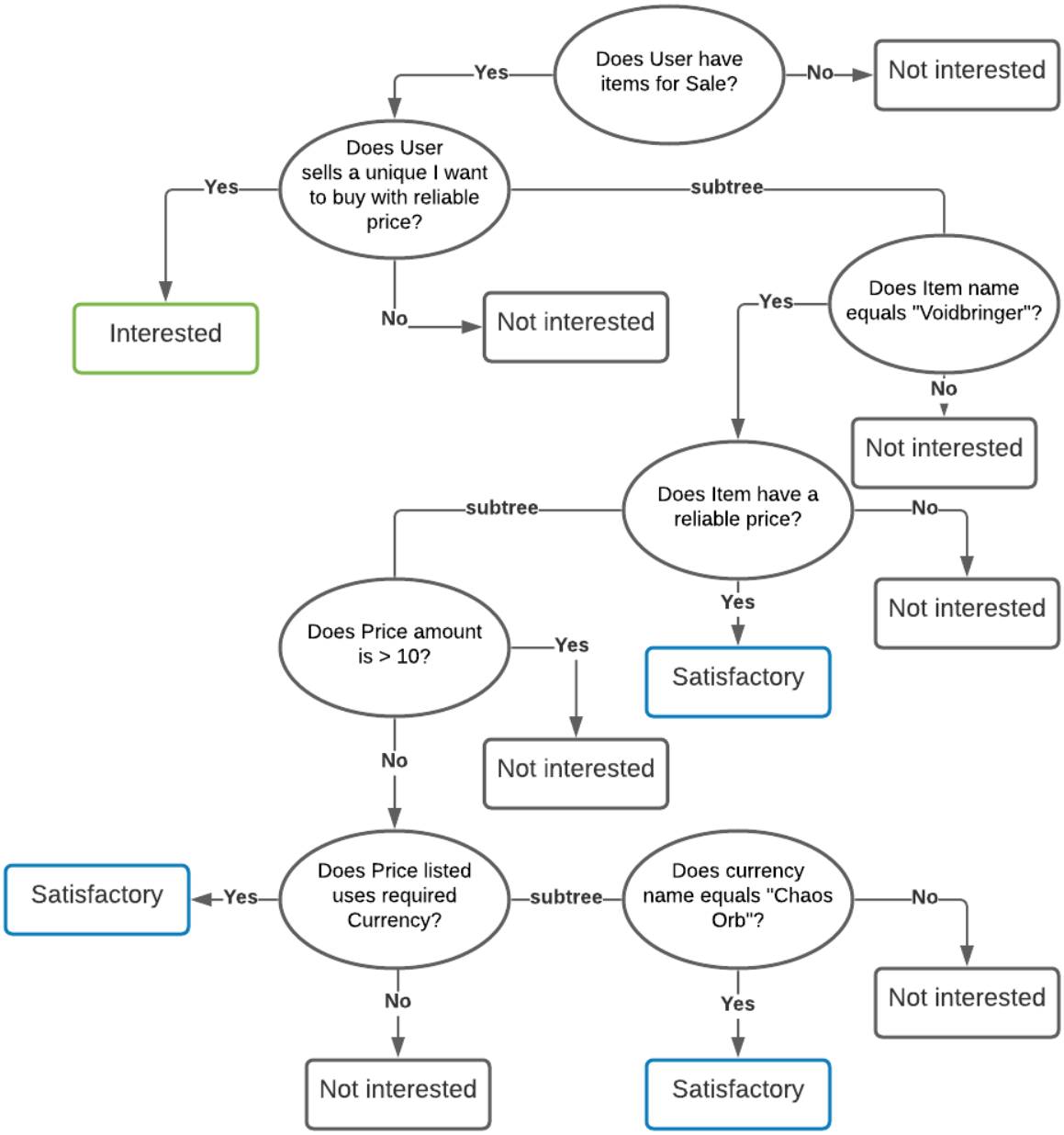

Рисунок 3.3 – Приклад дерева рішень

Розглянемо його більш детально. Користувач системи хоче купити ігровий предмет з назвою «Voidbringer», ціна якого зазначена в валюті під

назвою «Chaos Orb» та повинна бути менше 10. Для цього йому потрібно отримати список продавців, які задовольняють ці вимоги, щоб мати можливість зв’язатися з одним з них, використовуючи інформацію про продавця, та купити в нього цей предмет.

Користувач побудував дерево рішень, першим вузлом якого задає питання «Чи має користувач предмети на продаж?», якщо ні, цей користувач його не інтересує, якщо так – використовується наступний вузол з питанням «Чи користувач продає предмет який я хочу з потрібною ціною?», якщо ні, цей користувач його не інтересує, якщо так – користувач входить до кінцевого результату. Для відповіді на це питання користувач системи задав додаткове дерево рішень та умову виконання, при якій воно задовольнить поточний вузол. Додаткове дерево складається з двох вузлів з питаннями: «Чи має предмет назву «Voidbringer»?» та «Чи має предмет потрібну ціну?». Позитивний результат останнього задовольнить умову задану користувачем. Але цей вузол також складається з додаткового дерева рішень, вже з іншими питаннями. «Чи ціна вища за 10?», якщо так, цей предмет не інтересує користувача, якщо ні, то задається наступне питання «Чи ціна зазначена у потрібній валюті?». Цей вузол також містить додаткове дерево рішень з одним вузлом «Чи дорівнює назва валюти назві«Chaos Orb»?», позитивний результат якого задовольнить поточний вузол.

Після обробки цього дерева рішень, система повертає користувачу результат у вигляді списку кожної заданої їм інструкції та колекцій даних відібраних за її питанням в обох випадках.

З вибірки розміром у приблизно 1000 гравців, був знайдений продавець який задовольняє всі вимоги користувача системи. Це можна побачити на рисунку 3.4.

```
Name: Hauzilla
DateTime: null
Items:
Item#idd7cbfb8bf1ba88b494942671b5fff7e20e2ef51004834a861d75a0f75ee9d6b0
Name: Voidbringer
CreatedAt: null
Currency: Chaos Orb
Amount: 5.0
Attributes:
Name: lastCharacterName
Type: TEXT
Values: [Swift]
Name: stashName
Type: TEXT
Values: [5c]
Name: stashType
Type: TEXT
Values: [PremiumStash]
Name: league
Type: TEXT
Values: [Standard]
```

Рисунок 3.4 – Результат виконання програми

ВИСНОВКИ

В результаті виконання атестаційної роботи була досліджена предметна область електронного та звичайного ринків, висунута проблема та визначений об’єкт, мета та задача дослідження для її виправлення.

Було проведено дослідження різноманітних методів аналізу даних з наведенням їх прикладів та порівнянням, та вибраний найкращий метод, враховуючи галузь дослідження, поставлену мету та задачу.

Були сформовані вимоги до системи та вибрані інструменти для розробки, а саме мова програмування Java, Spring Framework, Intellij Idea.

Спираючись на висунуті вимоги, була спроектована універсальна система аналізу даних ринку та реалізований її прототип, використовуючи зазначені інструменти розробки. Система складається з узагальнених сутностей, що були сформовані спираючись на дослідження предметної області, та надає змогу користувачу використовувати наглядні дерева рішень для вирішення універсальних задач з аналізу даних ринків.

Усі поставлені задачі виконані в повному обсязі.

Розробка атестаційної роботи дала мені можливість заглибитися в роботу та проблеми ринкової галузі, винесла важливість аналізу даних та його практичне застосування, розширила знання мови програмування Java та дала додатковий досвід роботи з Spring Framework.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Lai, J. P. (2006). The Significance of E-Business and Knowledge-Based Customer Relationship in the E Marketplace Environment. *INTI Journal*, *2*(1), 552-559.

2. Swedberg, R. (1994). Markets as Social Structures. S. 255-282 in: NJ Smelser.

3. Bakos, J. Y. (1997). Reducing Buyer Search Costs: Implications for Electronic Marketplaces. *Management Science*, *43*(12), 1676-1692.

4. Vladimir, Z. (1996). Electronic commerce: structures and issues. *International journal of electronic commerce*, *1*(1), 3-23.

5. Timmers, P. (1998). Business models for electronic markets. *Electronic markets*, *8*(2), 3-8.

6. Bailey, J. P., & Bakos, Y. (1997). An exploratory study of the emerging role of electronic intermediaries. *International Journal of Electronic Commerce*, *1*(3), 7-20.

7. Segev, A., Gebauer, J., & Färber, F. (1999). Internet-based electronic markets. *Electronic Markets*, *9*(3), 138-146.

8. Bakos, Y. (1998). The emerging role of electronic marketplaces on the Internet. *Communications of the ACM*, *41*(8), 35-42.

9. Bakos, Y., & Brynjolfsson, E. (1999). Bundling information goods: Pricing, profits, and efficiency. *Management science*, *45*(12), 1613-1630.

10. Mirkin, B. (2013). *Mathematical classification and clustering* (Vol. 11). Springer Science & Business Media.

11. Johnsonbaugh, R., & Schaefer, M. (2004). *Algorithms*. Upper Saddle River, NJ: Pearson Education.

12. Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of the national academy of sciences*, *103*(23), 8577-8582.

13. Kuzomin, O., Tolmachova, T., & Astappiev, O. (2017). Analysis of Web user activity data. *International Journal of Information Models and Analyses*, *6*(2), 108-118.

14. Mirkin, B. (2011). *Core concepts in data analysis: summarization, correlation and visualization*. Springer Science & Business Media.

15. Kuzomin, O., Goloviy, N., & Dayoub, Y. (2008). Modeling of Effective Process of Network Maintaining Based on Statistical Data.

16. Kuzomin, O., & Lyashenko, V. (2008). Analysis of Spatial-temporal Dynamics in the System of Economic Security of Different Subjects of Economic Management.

17. Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis* (Vol. 344). John Wiley & Sons.

18. Kuzomin, O., & Lyashenko, V. (2006). Fuzzy set theory approach as the basis of analysis of financial flows in the economical security system.

19. Anderberg, M. R. (2014). *Cluster analysis for applications: probability and mathematical statistics: a series of monographs and textbooks* (Vol. 19). Academic press.

20. Priestman, M. (Ed.). (2003). *The Cambridge companion to crime fiction*. Cambridge University Press.

21. Child, D. (1990). *The essentials of factor analysis*. Cassell Educational.

22. Kuzomin, O., Lyashenko, V., Bulavina, E., & Torojev, A. (2005). Analysis of movement of financial flows of economical agents as the basis for designing the system of economical security (general conception). In *Third international conference «Information research, applications, and education* (pp. 27-30).

23. Kuzomin, O., & Lyashenko, V. (2009). Methods of comparative analysis of banks functioning: classic and new approaches. *Information Theories & Applications*, *16*(4), 384-396.

24. Kuzomin, O., & Lyashenko, V. (2007). Procedure of Formalization of the Indices of Banks' Stable Functioning in Comparative Estimates of Their Development.

25. Schumacker, R. E., & Lomax, R. G. (2004). *A beginner's guide to structural equation modeling*. Psychology press.

26. Hoyle, R. H. (1995). The structural equation modeling approach: Basic concepts and fundamental issues.

27. Hu, L. T., & Bentler, P. M. (1999). Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives. *Structural equation modeling: a multidisciplinary journal*, *6*(1), 1-55.

28. Kuzomin, O., & Lyashenko, V. (2008). Conceptual Foundations of Construction of the Models and Procedures for Prediction of the Avalanche-dangerous Situations Initiation.

29. McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, *5*(4), 115-133.

30. Kuzomin, O., Sorochan, M., Yanchevskiy, I., & Torojev, A. (2005). The use of situation representation when searching for solutions in computer aided design systems International Journal. *Information Theories & Applications.–Bulgaria*, *11*, 101-107.

31. Kuzomin, O., & Vasylenko, O. (2017). Methods and models for building a distributed mobile emergency monitoring system. *International Multidisciplinary Scientific GeoConference: SGEM*, *17*, 433-440.

32. Kuzomin, O., Lyashenko, V., Tkachenko, M., Ahmad, M. A., & Kots, H. (2016). Preventing of technogenic risks in the functioning of an industrial enterprise. *International Journal of Civil Engineering and Technology*, *7*(3), 262-270.

33. Hunt, E. B., Marin, J., & Stone, P. J. (1966). Experiments in induction.

34. Chatterjee, S., & Hadi, A. S. (2015). *Regression analysis by example*. John Wiley & Sons.

35. Kuzomin, O., Voronin, A., Ziatdinov, Y., Varlamov, I., Kirichenko, L., Radivilova, T., Baranovskyi, O., ... & Chebanyuk, O. (2017). Non-Linear Trade-off Scheme in Multicriteria Decision-Making Problems. *International Journal of Information Technologies & Knowledge, 11*(4), 3-22.

36. Klecka, W. R., Iversen, G. R., & Klecka, W. R. (1980). *Discriminant analysis* (Vol. 19). Sage.

37. Kuzomin, O., & Vasylenko, O. (2014). Data Loss Minimization in Situation's Centrums Data Bases. *International Journal" Information Technologies & Knowledge, 8*(2), 173-187.

38. Thompson, B. (1984). *Canonical correlation analysis: Uses and interpretation* (No. 47). Sage.

39. Kuzomin, O., Goloviy, N., & Dayoub, Y. (2008). Modeling of Effective Process of Network Maintaining Based on Statistical Data.

40. Kuzomin, O., & Lyashenko, V. (2011). Microsituation Concept in GMES Decision Support Systems.