

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**Факультет інформаційних технологій**

Кафедра технологій управління

Спеціальність 122 «Комп'ютерні науки»

Освітня програма «Інформаційна аналітика та впливи»

**КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА**

на тему:

«Інформаційний аналіз та управління проектом розробки веб сервісу  
планування подорожей»

**Студента 2-го курсу групи ІАВ-21**

Руденко Юрій Олегович

*(прізвище, ім'я, по батькові)*

**Науковий керівник:**

Осауленко Ігор

Анатолійович

*(прізвище, ім'я, по*

---

*(підпис студента)*

*(дата)*

*(підпис)*

**Попередній захист:**

---

*(Висновок: "До захисту в Екзаменаційній комісії")*

Завідувач кафедри  
технологій управління

Морозов В.В.

---

*(підпис)*

*(прізвище, ініціали)*

*(дата)*

**Київ – 2023**

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ  
ТАРАСА ШЕВЧЕНКА**

**Факультет інформаційних технологій**

Кафедра технологій управління

Освітній рівень Магістр

Спеціальність 122 Комп'ютерні науки

Освітня програма Інформаційна аналітика та впливи

**ЗАТВЕРДЖУЮ**  
Завідувач кафедри  
професор Морозов В.В.

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ року

**ЗАВДАННЯ  
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студента: Юрія РУДЕНКА

Група: ІАВ-21

**1. Тема кваліфікаційної роботи:** «Інформаційний аналіз та управління проектом розробки веб сервісу для планування подорожей»

Затверджена протоколом від №5 від 08.12.2022 р.

**2. Строк подання студентом готової роботи** - 16.05.2023 р.

**3. Цільова установка та вихідні дані до роботи:** аналіз інструментів та методів для аналізу і класифікації даних також аналіз інструментів та методів управління проектами; аналіз організаційною структурою проекту; оцінка завантаженості ресурсів та управління їми; оцінка методів класифікації даних, складання дій для розпізнання та вирішення аналітичних подій; результатом роботи є розроблений сервіс аналізу з наявним переліком статистичних даних і функціоналом їх оновлення.

**4. Зміст роботи:**

Аналіз предметної області, обґрунтування цінності нового проекту, аналіз методів машинного аналізу; опис методологій управління проектами та вибір однієї для планування проекту; оцінка альтернатив проекту; PEST-аналіз проекту та аналіз впливу ринку; оцінка актуальних технологій для використання у сервісі планування ; розробка системи аналізу; визначення необхідного функціоналу для аналізу; встановлення зв'язків між даними; фіксування змін та реакція на зміну проаналізованих ресурсів; визначення актуальних підходів розробки.

### **5. Перелік графічного матеріалу (слайдів):**

Аналіз предметної області, аналіз існуючих проектів, формалізація ідеї, зацікавлені сторони та їхні впливи, обмеження проекту, Діаграма Гантта, зведені дані про план проекту, організаційна структура, базовий план та реакція на зміни, ризики проекту.

### **6. Календарний план виконання роботи**

№ з/ п	Назва частин роботи	План виконання роботи
	Вибір теми кваліфікаційної роботи	08.12.22
	Затвердження теми дипломної роботи та призначення наукового керівника	08.12.22
	Формування переліку нормативних матеріалів, літератури з проблематики дипломної роботи	01.03.23-07.03.23
	Розробка плану дипломної роботи і його погодження з науковим керівником	07.03.2023
	Написання I розділу дипломної роботи «Аналіз проблем, пошук існуючих рішень та вибір підходу до управління проекту»	14.03.23-28.03.23
	Написання II розділу дипломної роботи «Опис та розробка концепції проекту»	28.03.23-12.04.23

	Написання III розділу дипломної роботи «Аналіз інструментів та методів управління проектом»	12.04.23-26.04.23
	Написання IV розділу дипломної роботи «Вибір програмних засобів та проектування веб сервісу»	26.04.23-08.05.23
	Підготовка висновків і пропозицій	08.05.23-09.05.23
	Оформлення кваліфікаційної роботи	09.05.23-14.05.23
	Передача кваліфікаційної роботи рецензенту для рецензування, перевірка на антиплагіат	19.05.2023
	Передача кваліфікаційної роботи науковому керівникові	16.05.2023
	Попередній захист кваліфікаційної роботи	17.05.2023
14.	Захист кваліфікаційної роботи	23.05.2023

Дата видачі завдання 08.12.2022.

Керівник роботи

Осауленко Ігор Анатолійович

---

*(підпис)*

Завдання прийняв до виконання студент групи ІАВ-21

Руденко Юрій Олегович

---

*(підпис)*

## ЗМІСТ

АНОТАЦІЯ.....	6
ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ ТА ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ.....	12
1. Опис предметної області.....	12
2. Результати дослідження оточення ІТ проєкту .....	13
3. PEST аналіз .....	18
4. Машинне навчання, перелік основних задач .....	27
5. Методи аналізу .....	32
6. Аналіз систем машинного навчання .....	38
7. Аналіз методологій управління проєктами.....	41
1.7.1 Waterfall .....	41
1.7.2 Agile .....	42
1.7.3 Scrum.....	43
1.7.4 Kanban.....	44
1.7.5 Вибір методології для управління проєктом.....	45
РОЗДІЛ 2. ОПИС ТА РОЗРОБКА КОНЦЕПЦІЇ ПРОЄКТУ .....	46
1. Обґрунтування вибору технологій для вирішення поставлених задач .....	46
2. Аналіз переваг вибраних методів.....	52
3. Вибір методів для аналізу даних машинним навчанням .....	59
4. Метод аналізу Random Forest .....	68
5. Мета, цілі та завдання проєкту .....	71
РОЗДІЛ 3. АНАЛІЗ ІНСТРУМЕНТІВ ТА МЕТОДІВ УПРАВЛІННЯ ПРОЄКТУ.....	75
1. Управління змістом проєкту.....	75
2. Управління ресурсами.....	78
3. Управління ризиками.....	82

4. Управління якістю.....	84
РОЗДІЛ 4. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА ПРОЕКТУВАННЯ ВЕБ СЕРВІСУ.....	88
1. Проектування архітектури системи.....	88
2. Моделювання поведінки системи.....	90
3. Frontend-розробка.....	97
4. Backend-розробка.....	99
5. Взаємодія з користувачем.....	104
ВИСНОВКИ.....	112
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	115

## АНОТАЦІЯ

кваліфікаційної роботи магістра на тему

### **«Інформаційний аналіз та управління проектом розробки веб сервісу планування подорожей»**

Студент: Юрій РУДЕНКО

Науковий керівник: Ігор ОСАУЛЕНКО

Рік захисту – 2023

*Темою роботи* було обрано «Інформаційний аналіз та управління проектом розробки веб сервісу планування подорожей». Предметною областю туристична галузь в Україні та в усьому світі, розробка веб сервісу для планування подорожей.

*Метою* роботи є розробка концепції аналітичної складової сервісу планування на основі аналітичних даних що до користувачів.

*Ціль* – створення нового веб продукту на ринку в Україні для вирішення проблем в туристичній галузі після наслідків COVID-19 та війни; вихід веб продукту на світовий ринок для конкуренції з схожими продуктами; отримання прибутку.

*Предметом дослідження* є предметна область та середовище ІТ проекту, а саме: підбір технологій розробки, управління аналітикою, створення плану розробки відповідно до запланованих методів взаємодії з користувачами.

*Наукова новизна* полягає у плануванні та створенні зовсім нового проекту для України – веб сервісу для планування подорожей, який буде складатись із веб сайту.

Магістерська робота складається зі вступу, основної частини, що включає чотири розділи, висновків та списку використаних джерел.

*У першому розділі* проводиться аналіз та дослідження предметної області. В розділі описуються результати досліджень оточення ІТ проекту, результати різних досліджень.

*У другому розділі було виконано опис та розробку концепції проекту. Визначені проблеми та цілі, які можуть бути вирішені цим проектом. Були чітко сформульовані мета, цілі та завдання проекту, а також створено опис продукту проекту і спроектовано життєвий цикл проекту.*

*У третьому розділі були проаналізовані інструменти та методи управління проектом. Розглянута організаційна структура компанії і створена організаційна структура проекту. Також розглянуто управління змістом проекту, управління ресурсами, ризиками та якістю проекту.*

*В четвертому розділі було описано вибрані програмні засоби та проектування веб-сервісу. Були розглянуті етапи моделювання та бази даних. Детально розглянуто модель бази даних веб-сервісу, її структуру та залежності. Також були розглянуті методи розробки програмного продукту: клієнтської та сервісної частин.*

Робота містить 117 сторінок з додатками, 26 рисунків та 10 таблиць.

**Ключові слова:** *аналіз машинного навчання, веб сервіси, концепція проекту, управління змістом, управління якістю, управління ризиками.*

## ВСТУП

Світова пандемія COVID-19 сильно пошкодила туристичну галузь, викликавши непередбачувані наслідки для зайнятості та бізнесу. Туризм був одним з перших секторів, який серйозно постраждав від цієї кризи, оскільки заходи, вжиті для боротьби з поширенням вірусу, призвели до практично повного припинення подорожей по всьому світу. З врахуванням поточних обмежень на міжнародні поїздки та глобальної економічної рецесії, цей сектор також може стати одним із останніх, що відновиться. Ця ситуація має наслідки не лише для туристичної галузі, але й для багатьох інших сфер, які залежать від туризму та отримують підтримку від нього.

Складно переоцінити важливість обробки і аналізу даних в сучасному світі. У різних областях, таких як медицина, промисловість, енергетика та інших накопичується величезна кількість даних, часто представлених в складному для обробки і аналізу людиною форматі.

Мотивацією для дослідження послужив той факт, що в даний час росте інтерес наукової спільноти та бізнесу до технологій і методів машинного навчання і аналізу даних. Також, спостерігається зростання кількості різних програмних реалізацій існуючих алгоритмів і підходів [1].

Але сьогодні, щоб почати вирішувати будь-яку реальну задачу машинного навчання (Machine Learning, ML), необхідно мати досить високу компетенцією і розбиратися не тільки в математичному апараті, але і в відповідних програмних продуктах. Зараз набір технологій представлений у вигляді великої кількості різних продуктів, найчастіше несумісних між собою, навчитися користуватися яким в якому досить складно.

На зорі бурхливого розвитку технологій і методів машинного навчання кожен дослідник самостійно розробляв собі програмне забезпечення, яке брало на себе рутинні операції по первинній обробці даних і пошуку.

*Актуальність проекту* в тому, що, незважаючи на війну в Україні, можна створювати туристичні веб сервіси, які будуть працювати на увесь світ та бути

актуальними, оскільки розквіт туристичної сфери в пост ковідний період ще попереду.

*Темою роботи* було обрано «Аналіз розробки веб сервісу планування подорожей». Предметною областю є туристична сфера в Україні та світі, планування індивідуальних подорожей.

*Метою* роботи є аналіз предметної області, проблеми середовища ІТ проекту, пошук рішень для вирішення проблем, огляд організаційної структури сервісу та аналіз даних відповідно до кластерного методу аналізу також пошук рішень для управління проектом.

Основними завданнями дослідження є:

- аналіз предметної області та дослідження середовища ІТ проекту; проведення маркетингових досліджень; формування висновків;
- розгляд технологій аналізу та вибір потрібного варіанту для розробки проекту;
- проектування функцій навчання і взаємодії з даними моделі проекту, аналіз результатів;
- визначення вимог зацікавлених сторін до якості продукту проекту;
- розробка плану проекту
- розробка показників для контролю якості проекту;
- підготовка доповіді та презентації про результати виконаної роботи;

*Об'єктом дослідження* є процеси розробки веб сервісу на основі аналітичних даних. А також, постановка цілей та задач управління даним проектом.

*Предметом дослідження* є дослідження та аналіз предметної області та середовища ІТ проекту, процеси розробки сервісу, а саме: аналіз наявних методів збереження даних, їх аналізу і взаємодії з користувачем.

*Методи дослідження:* аналіз стану туристичної галузі, обробка вхідної інформації що до статистики подорожей і вподобань. Формування ідеї проекту. Визначення сильних та слабких сторін. Визначення потреб ринку та

користувачів. Використано метод критичного шляху при планування проекту. Для отримання висновків використано аналіз та синтез всіх отриманих результатів дослідження.

*Наукова новизна* полягає у плануванні та створенні зовсім нового проекту для України – веб сервісу для планування подорожей, який буде складатись із веб сайту.

*Практичне значення* полягає у розробленому плані проекту, що не має аналогів на ринку. В проекті враховано недоліки схожих працюючих проектів. Проаналізовано впливи зацікавлених сторін, щоб вистроїти ефективну комунікацію та взаємодію з ними.

# РОЗДІЛ 1. АНАЛІЗ ТА ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1. Опис предметної області

Світова пандемія COVID-19 сильно пошкодила туристичну галузь, викликавши непередбачувані наслідки для зайнятості та бізнесу. Туризм був одним з перших секторів, який серйозно постраждав від цієї кризи, оскільки заходи, вжиті для боротьби з поширенням вірусу, призвели до практично повного припинення подорожей по всьому світу. З врахуванням поточних обмежень на міжнародні поїздки та глобальної економічної рецесії, цей сектор також може стати одним із останніх, що відновиться. Ця ситуація має наслідки не лише для туристичної галузі, але й для багатьох інших сфер, які залежать від туризму та отримують підтримку від нього.

Навіть при відомій стійкості туристичної економіки до попередніх криз, нинішній масштаб і повністю комплексний економічний і медичний характер поточної кризи означає, що шлях до відновлення є дуже невизначеним. Незважаючи на деяке відновлення міжнародної туристичної активності, його масштаби залишаються обмеженими. У багатьох країнах внутрішній туризм відновився, але він може лише частково замінити втрати в туризмі, пов'язані з міжнародними подорожами.

Прогнозується, що повне відновлення міжнародного туризму досягне кінця 2021 року, а можливо й займе кілька років. Ця інформація відповідає останнім прогнозам інших організацій. Згідно з оцінками Всесвітньої туристичної організації ООН (UNWTO), кількість міжнародних туристів у перші вісім місяців 2020 року знизилася на 70%, що відповідає втратам експортних надходжень від міжнародного туризму вісім разів більшими, ніж у 2009 році (під час світової фінансової кризи). UNWTO прогнозує, що міжнародні прибутки зменшаться на майже 70%, і очікується відновлення до попереднього рівня не раніше 2023 року.

Пандемія має серйозні економічні та соціальні наслідки для багатьох людей, місць і підприємств, а також для загальної економіки. Туризм відіграє

важливу роль у генерації іноземної валюти, підтримці робочих місць та бізнесу, сприяє регіональному розвитку і підтримує місцеві громади. До початку пандемії цей сектор у середньому становив 4,4% ВВП, 6,9% зайнятості та 21,5% експорту послуг у країнах Організації економічного співробітництва і розвитку (OECD), а також 6,5% світового експорту, згідно з даними Всесвітньої організації торгівлі. Непрямий вплив туризму також є значним, підсилюючи масштаб шоку для національної та місцевої економіки.

## **2. Результати дослідження оточення ІТ проєкту**

За останніми даними з "UNWTO World Tourism Barometer", кількість міжнародних туристів в січні-липні 2022 року зросла майже втричі (+172%) порівняно з тим же періодом 2021 року. Це свідчить про відновлення сектору на майже 60% від рівня до пандемії. Стабільне відновлення вказує на високий накопичений попит на міжнародні подорожі, а також на послаблення або скасування обмежень щодо міжнародних поїздок в багатьох країнах (на 19 вересня 2022 року 86 країн не мали обмежень, пов'язаних з COVID-19).

У цей період близько 474 мільйонів туристів здійснили міжнародні подорожі, порівняно з 175 мільйонами за той же період 2021 року. Оцінки також показують, що у червні та липні 2022 року було зареєстровано 207 мільйонів міжнародних прибуттів, що вдвічі більше, ніж у ці два місяці минулого року. Ці два місяці складають 44% від загальної кількості прибуттів, зареєстрованих протягом перших семи місяців 2022 року. Європа прийняла 309 мільйонів з цих прибуттів, що становить 65% від загальної кількості.

В січні-липні 2022 року Європа та Близький Схід показали найшвидше відновлення, досягнувши 74% та 76% від рівня 2019 року відповідно, що становить значне покращення. Кількість міжнародних прибуттів до Європи майже втричі перевищила результати перших семи місяців 2021 року (+190%). В цьому сприяли сильний внутрішньорегіональний попит та подорожі зі Сполучених Штатів. Особливо помітне покращення спостерігалось в червні (-21% порівняно з 2019 роком) та липні (-16%), коли кількість прибуттів підвищилася до близько 85% від рівня 2019 року. Ці результати також були

сприятливими завдяки скасуванню обмежень на подорожі в багатьох напрямках (на 19 вересня 2022 року 44 країни Європи не мали обмежень, пов'язаних з COVID-19).

У період січень-липень 2022 року Близький Схід відзначився значним зростанням кількості міжнародних прибуттів. Порівняно з аналогічним періодом минулого року, кількість прибуттів зростає майже в чотири рази (+287%). Особливо вражаючими були результати у липні, коли кількість прибулих перевищила рівень до пандемії (+3%). Саудівська Аравія зіграла значну роль у цьому зростанні, з показниками, що перевищують 121%, завдяки паломництву Хадж. Це свідчить про відновлення туристичної активності в регіоні та позитивні наслідки від проведення важливих релігійних подій.

В січні-липні 2022 року Америка та Африка також відзначилися вражаючим зростанням кількості міжнародних прибуттів порівняно з 2021 роком. У Америці зареєстровано зростання на 103%, досягнувши 65% від рівня 2019 року. У свою чергу, Африка відзначилася зростанням на 171%, досягнувши 60% від рівня 2019 року. У регіоні Азії та Тихоокеанському спостерігалось збільшення кількості прибулих більш ніж удвічі (+165%) протягом перших семи місяців 2022 року. Однак вони залишилися на 86% нижче рівня 2019 року через деякі обмеження на міжнародні поїздки.

У січні-липні 2022 року кілька субрегіонів відзначилися значним відновленням до рівня прибуттів до пандемії. Південна Середземноморська Європа зазнала зниження лише на 15% порівняно з 2019 роком, Карибський басейн на 18%, а Центральна Америка на 20%. Західна Європа (-26%) і Північна Європа (-27%) також показали сильні результати у відновленні. У липні кількість прибулих наблизилася до рівня до пандемії в Карибському басейні (-5%), Південній і Середземноморській Європі (-6%) і Центральній Америці (-8%). Ці дані свідчать про позитивну тенденцію відновлення туристичних потоків у цих регіонах, але вони все ще не досягли повного відновлення до рівня до пандемії.

Дані за перші п'ять-сім місяців 2022 року показали, що деякі напрямки мають кількість міжнародних прибуттів, що перевищує рівень до пандемії. Ці напрямки включають:

1. Віргінські острови США: зростання на 32% порівняно з 2019 роком.
2. Албанія: зростання на 19%.
3. Сен-Мартен: зростання на 15%.
4. Ефіопія та Гондурас: обидві країни зростали на 13%.
5. Андорра: зростання на 10%.
6. Пуерто-Ріко: зростання на 7%.
7. Об'єднані Арабські Емірати та Домініканська Республіка: обидві країни зростали на 3%.
8. Сан-Марино та Сальвадор: обидві країни зростали на 1%.
9. Кюрасао: зростання на 0%.

Ці дані свідчать про позитивний тренд у відновленні туристичних потоків в цих напрямках, приближаючись до рівня до пандемії.

Дані про надходження від міжнародного туризму за перші п'ять-сім місяців 2022 року вказують на кілька напрямків, які перевищили попередні рівні до пандемії. Серед цих напрямків є:

1. Сербія: зростання на 73%.
2. Судан: зростання на 64%.
3. Румунія: зростання на 43%.
4. Албанія: зростання на 32%.
5. Північна Македонія: зростання на 24%.
6. Пакистан: зростання на 18%.
7. Туреччина, Бангладеш і Латвія: всі зростали на 12%.
8. Мексика і Португалія: обидві країни зростали на 8%.
9. Кенія: зростання на 5%.
10. Колумбія: зростання на 2%.

Ці дані свідчать про успішне відновлення туристичних потоків в цих напрямках, що дозволило їм перевищити попередні рівні до пандемії.

Дійсно, дані про витрати на виїзний туризм з основних вихідних ринків свідчать про поступове відновлення. Наприклад, витрати з Франції зросли до -12% у січні-липні 2022 року порівняно з 2019 роком, а витрати з Німеччини збільшилися до -14%. У Італії витрати на міжнародний туризм склали -23%, а у Сполучених Штатах -26%.

Також варто зазначити, що міжнародні пасажирські авіап перевезення показали стабільні показники відновлення. За січень-липень 2022 року обсяг перевезень збільшився на 234%, хоча він залишається на рівні 45% нижче, ніж у 2019 році. У липні відновлення досягло приблизно 70% від рівня трафіку перед пандемією. Ці дані надані Міжнародною асоціацією повітряного транспорту (IATA) і свідчать про позитивні тенденції в авіаційній галузі.

Так, зростання попиту в міжнародному туризмі також створило серйозні проблеми з роботою та робочою силою в туристичних компаніях та інфраструктурі, зокрема в аеропортах. Підвищений попит на послуги туризму може приводити до перевантаження та перенапруження персоналу та ресурсів, що може вплинути на якість обслуговування та загальний досвід туристів.

Економічна ситуація, загострена конфліктом між Російською Федерацією та Україною, також становить значний ризик для відновлення міжнародного туризму. Геополітичні напруги можуть вплинути на безпеку та стабільність в регіоні, що може знизити інтерес туристів до відвідування певних місць.

Крім того, інші фактори, такі як підвищення процентних ставок, зростання цін на енергоносії та продукти харчування, а також перспективи глобальної рецесії, які були зазначені Світовим банком, можуть стати головною загрозою для відновлення міжнародного туризму в середині 2022 і 2023 років. Ці економічні фактори можуть вплинути на покупередження та споживчу активність туристів, що може призвести до зниження попиту на подорожі.

Останній індекс довіри Всесвітньої організації туризму (UNWTO) та тенденції в бронюванні також вказують на можливе сповільнення відновлення міжнародного туризму. Ці показники відображають більш обережний прогноз та ознаки повільнішого зростання, що свідчить про потенційний ризик для подальшого відновлення галузі.

Оцінка 125 балів за шкалою від 0 до 200, дана Панеллю експертів з туризму UNWTO для періоду травень-серпень 2022 року, відповідає позитивним очікуванням, які були висловлені групою в травневому опитуванні за той самий 4-місячний період. Це означає, що експерти сподіваються на позитивне відновлення туристичної галузі в цей період, і їх очікування підтверджуються даними та показниками.

Перспективи на решту 2022 року у туристичній галузі є стримано оптимістичними, згідно з оцінками експертів з туризму. Оцінка періоду вересень-грудень 2022 року склала 111 балів, що нижче попередніх чотирьох місяців, що свідчить про зниження рівня довіри. Близько половини експертів (47%) очікують позитивних перспектив на цей період, 24% не очікують значних змін, а 28% вважають, що ситуація може погіршитися.

Однак, експерти виявляють більшу впевненість щодо 2023 року, прогнозуючи покращення показників туризму. За даними опитування, 65% експертів сподіваються на покращення в 2023 році порівняно з 2022 роком.

Варто враховувати, що прогнози туристичної галузі можуть залежати від різних факторів, таких як економічна ситуація, геополітичні події, епідеміологічна обстановка та інші зовнішні чинники, які можуть вплинути на подорожі та поведінку туристів.

Так, економічне середовище є важливим фактором, який впливає на перспективи повернення до допандемічного рівня в туристичній галузі. Невизначеність і зростання економічних чинників, таких як інфляція і ціни на енергоносії, можуть створювати труднощі для відновлення міжнародного туризму.

За даними опитування, близько 61% експертів прогнозують потенційне повернення міжнародних прибутків до рівня 2019 року у 2024 році або пізніше. Тимчасом, кількість експертів, які вказують на повернення до рівня до пандемії в 2023 році, зменшилася до 27% порівняно з попереднім опитуванням.

Зростання інфляції та цін на нафту призводять до підвищення витрат на транспорт та проживання, що може обмежити покупку спроможність туристів та їх здатність витратити гроші на подорожі. Ці чинники можуть впливати на споживачів та їх здатність до заощаджень, що може утруднити відновлення туристичної галузі до допандемічного рівня.

### 3. PEST аналіз

Методологія PEST-аналізу дозволяє оцінити вплив зовнішнього середовища на функціонування і розвиток туризму в Україні.

Таблиця 1.1

#### Характер та ступінь впливу політичних факторів на компанію

Фактори впливу політичного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору (max = 3 б.)			Середній бал
		Експерт	Експерт	Експерт	
Кількісні та якісні обмеження на імпорт	-	1	1	1	-1
Бюрократизація і рівень корупції	-	1	1	1	-1
Стійкість політичної влади та існуючого	+	2	2	1	+1,67
Податкова політика держави	-	2	1	1	-1,33
Вірогідність розвитку військових дій в країні	-	3	3	3	-2

Тенденції до регулювання або	-	1	2	2	-1,67
Антимонопольне та трудове законодавство	+	2	1	3	+2
Майбутнє і поточне законодавство, що регулює правила роботи	+	2	1	2	+1,67
Підтримка інноваційних компаній з боку держави	+	3	3	3	+3
Лібералізація зовнішньоекономічного співробітництва	+	2	3	3	+2,67

Сприятливі зовнішні умови сприяють розвитку туризму шляхом стимулювання регіональної економіки, зайнятості та зростання платоспроможного попиту. Однак, на шляху до розвитку туризму стоять обмеження, такі як недостатній рівень розвитку споживчого ринку та незначна частка платних послуг у валовому регіональному продукті.

*Таблиця 1.2*

#### **Характер та ступінь впливу економічних факторів на компанію**

<b>Фактори впливу економічних</b>	<b>Характер впливу</b>	<b>Оцінка експертів ступеня впливу фактору (max = 3 б.)</b>	<b>Середній бал</b>
-----------------------------------	------------------------	---	---------------------

<b>факторів</b>	<b>фактору</b>	Експерт	Експерт	Експерт	
Рівень інфляції	-	2	3	3	-2,67
Курси основних валют	-	3	2	3	-2,67
Рівень наявних доходів населення	+	2	2	2	+2
Ступінь глобалізації та відкритості економіки	+	2	1	3	+2
Монетарна та фіскальна політика	+	1	1	2	+1,33
Рівень розвитку підприємництва та бізнес середовища	+	2	2	1	+1,67
Цінова конкуренція з боку зарубіжних компаній	-	2	1	1	-1,33
Інвестиційний клімат в галузі	+	3	2	2	+2,33

*Таблиця 1.3*

**Характер та ступінь впливу соціально-культурних факторів на компанію**

Фактори впливу соціально-культурних факторів	Характер впливу фактору	Оцінка експертів ступеня впливу фактору (max = 3 б.)			Середній бал
		Експерт	Експерт	Експерт	
Рівень підготовки молодих спеціалістів в галузі	+	2	2	2	+2
Вимоги до якості продукції та рівня сервісу	+	2	1	1	+1,33
Рівень міграції та імміграційні настрої	-	3	2	3	-2,67
Культура формування заощаджень і кредитування	+	2	2	1	+1,67
Розвиток релігії та інших вірувань	+	1	2	3	+2
Ставлення до імпортованих товарів і	+	2	2	1	+1,67
Спосіб життя і звички споживання	+	2	1	1	+1,33
Темпи росту населення	+	2	1	1	+1,33

Таблиця 1.4

**Характер та ступінь впливу технологічних факторів на компанію**

Фактори впливу технологічних	Характер впливу	Оцінка експертів ступеня впливу фактору (max = 3 б.)	Середній бал
------------------------------	-----------------	--	--------------

<b>факторів</b>	<b>фактору</b>	Експерт	Експерт	Експерт	
Рівень інновації та технологічного розвитку галузі	+	1	1	1	+1
Ступінь використання, впровадження та передачі технологій	+	2	1	3	+2
Доступ до новітніх технологій	+	2	1	2	+1,67
Витрати на дослідження та розробки	-	3	3	2	-2,67
Виробництва якісно нової продукції (розвиток конкурентних	-	1	2	3	-2
Законодавство в галузі технологічного оснащення галузі	+	2	3	2	+2,33

Проаналізуємо вплив факторів, які спричиняють найбільший вплив на діяльність компанії

*Таблиця 1.5*

**Фактори, які спричиняють найбільший вплив на діяльність компанії за результатами оцінювання PEST-аналізу**

<b>Політичні фактори</b>	<b>Вага</b>	<b>Економічні фактори</b>	<b>Вага</b>
--------------------------	-------------	---------------------------	-------------

Вірогідність розвитку військових дій в країні	-2	Курси основних валют	-2,67
Підтримка інноваційних компаній з боку держави	+2,67	Інвестиційний клімат в галузі	+2,33
<b>Соціально-культурні</b>	<b>Вага</b>	<b>Технологічні фактори</b>	<b>Вага</b>
Рівень міграції та імміграційні настрої	-2,67	Витрати на дослідження та розробки	-2,67
Рівень підготовки молодих спеціалістів в галузі	+2	Законодавство в галузі технологічного оснащення галузі	+2,33

Після проведення аналізу PEST-аналізу і врахування факторів, які мають найбільший вплив на діяльність компанії, можна зробити наступні висновки. Підтримка інноваційних компаній з боку держави та сприятливий інвестиційний клімат в галузі мають позитивний вплив на туристичну галузь. Проте, існують соціальні проблеми, такі як низький рівень доходів населення, обумовлений високими курсами основних валют, воєнним конфліктом у країні, а також низька споживча активність та негативна динаміка в показниках населення та стані здоров'я, особливо в сучасний час.

*Таблиця 1.6*

#### **Підсумки здійсненого PEST -аналізу для компанії**

<b>Фактори</b>	<b>Зміни в галузі</b>	<b>Зміни в організації</b>	<b>Дії</b>
----------------	-----------------------	----------------------------	------------

<b>Політичні</b>	<ol style="list-style-type: none"> <li>1. Падіння кількості туристів.</li> <li>2. Підтримка інноваційних компаній з боку держави</li> </ol>	<ol style="list-style-type: none"> <li>1. Створення бонусної програми або програми знижок.</li> <li>2. Збільшення капіталізації компанії.</li> </ol>	<ol style="list-style-type: none"> <li>1. Заохочення суспільства до туризму всередині України для дослідження власної країни.</li> <li>2. Розширення співпраці з туристичними компаніями</li> </ol>
<b>Економічні</b>	<ol style="list-style-type: none"> <li>1. Падіння кількості туристів.</li> <li>2. Інвестиційний клімат в галузі</li> </ol>	<ol style="list-style-type: none"> <li>1. Зосередження розвитку проекту на міжнародні ринки.</li> <li>2. Збільшення коштів на рекламні кампанії.</li> </ol>	<ol style="list-style-type: none"> <li>1. Створення бонусної програми або програми знижок.</li> <li>2. Розширення функціоналу веб сервісу.</li> <li>3. Заохочення туристів до використання веб сервісу.</li> </ol>
<b>Соціально-культурні</b>	<ol style="list-style-type: none"> <li>1. Падіння попиту сервісу для мігрантів.</li> </ol>	<ol style="list-style-type: none"> <li>1. Створення плану для залучення мігрантів до сервісу.</li> </ol>	<ol style="list-style-type: none"> <li>1. Цільова робота з мігрантами.</li> <li>2. Реферальна програма для активних користувачів.</li> </ol>

*Продовження табл. 1.6*

	2. Збільшення фінансів у туристів.	2. Популяризація сервісу за рахунок активних користувачів.	
<b>Технологічні</b>	1. Падіння попиту на сервіс через конкуренцію. 2. Використання нових технологій у туризмі.	1. Аналіз нових конкурентів. 2. Покращення сервісу за рахунок розвитку туризму в державі.	1. Зміни в роботі, функціоналі веб сервісу для конкурування на ринку. 2. Створення нового функціоналу для використання технологічного оснащення в сфері туризму.

Результати дослідження показують, що аналіз впливу зовнішнього середовища є важливим процесом для розробників стратегічного планування, оскільки він дозволяє контролювати зовнішні чинники, що впливають на підприємство. Це дозволяє виявити можливості та загрози, які можуть вплинути на інтереси компанії.

Лише через детальний аналіз зовнішнього середовища можна бути впевненим у майбутній успіх і контролювати його вплив на поточну діяльність проекту та компанії в цілому. Це підкреслює необхідність постійного і уважного спостереження за змінами в навколишньому середовищі, використовуючи сучасні методи та підходи.

Отже, компанія повинна бути готовою використовувати сучасні методи і прийоми для постійного та уважного вивчення свого оточення і враховувати його вплив на свою діяльність і стратегію розвитку.

#### 4. Машинне навчання, перелік основних задач

Поняття аналізу даних можна визначити наступним кількома способами:

1. Сукупність дій, здійснюваних дослідником у процесі вивчення отриманих тим або іншим способом даних з метою формування певних уявлень про характер явища, що описується цими даними. Дослідник намагається дані згорнути, скоротити їхню кількість, і втратити при цьому якнайменше корисної інформації, потенційно в них закладеної. Робиться це звичайно, за допомогою математичних методів.

2. Процес вивчення статистичних даних (пошуку закономірностей статистичних, закономірностей у середньому) за допомогою математичних методів, що не передбачають імовірнісної моделі досліджуваного явища. Ці методи не задовольняють строгим математичним вимогам. Вибір найкращого методу тут майже завжди спирається на евристичні міркування, що не підлягають формалізації. Тому проблема обґрунтування одержуваних висновків тут вимагає особливої уваги. Виділення методів аналізу даних обумовлено потребами ряду наук, у яких, з одного боку, велика потреба пошуку статистичних закономірностей, а, з іншого, - припущення, що лежать в основі ймовірнісно-статистичних методів, розроблених спеціально для вирішення таких задач.

3. Ототожнюється з поняттям прикладної статистики, що розуміється як наукова дисципліна, що розробляє й систематизує поняття, прийоми, математичні методи й моделі, призначені для організації збору, стандартного запису, систематизації й обробки статистичних даних з метою їхнього зручного подання, інтерпретації й одержання наукових і практичних висновків.

4. Процедури пошуку статистичних закономірностей (згортки інформації) що не зводяться до застосування формальних алгоритмів. В основі лежить комплексне використання математико-статистичних методів і методів аналізу даних у розумінні другого визначення з опорою на кілька методологічних принципів.

Серйозне місце в аналізі даних починає займати глибинний аналіз даних (Data Mining), процес пошуку різних закономірностей у величезному наборі фактичних даних [2].

Data Mining є процесом виділення цінних та практично корисних знань з великих обсягів сирих даних, які раніше були невідомі або непередбачувані. Ці знання можуть бути інтерпретовані та використані для прийняття рішень у різних сферах людської діяльності.

Переважає більшість класичних процедур мають час виконання, квадратичне або навіть кубічне за об'ємом вихідних даних. При кількості об'єктів, що перевершує кілька десятків тисяч, вони працюють неприйнятно повільно навіть на найсучасніших комп'ютерах. За останні десятиліття значні зусилля в області Data Mining були спрямовані на створення спеціалізованих алгоритмів, здатних виконувати ті ж задачі без істотної втрати точності [2].

Машинне навчання виникло на стику прикладної статистики, оптимізації, дискретного аналізу, і за останні 30 років оформилося в самостійну математичну дисципліну. Методи машинного навчання становлять основу згаданої вище ще молодшої дисципліни — інтелектуального аналізу даних.

В загальному розрізняють два типи навчання. Навчання за прецедентами, або індуктивне навчання, засноване на виявленні загальних закономірностей за частковими емпіричними даними. Дедуктивне навчання передбачає формалізацію знань експертів та їх перенесення в комп'ютер у вигляді бази знань. Дедуктивне навчання прийнято відносити до області експертних систем.

Загальну постановку задачі навчання за прецедентами можна сформулювати наступним чином [4]. Дано скінченну множину прецедентів (об'єктів, ситуацій), по кожному з яких зібрані (виміряні) деякі дані. Дані про прецедент називають також його описом. Сукупність всіх наявних описів прецедентів називається навчальною вибіркою. Потрібно за цими частковими даними виявити загальні залежності, закономірності, взаємозв'язки, властиві не тільки цій конкретній вибірці, але взагалі всім прецедентам, у тому числі тим, які ще не спостерігалися [5].



Рисунок 1.1 – Карта машинного навчання

Найпоширенішим способом опису прецедентів є ознаковий опис. Фіксується сукупність  $n$  показників, що вимірюються у всіх прецедентів. Якщо всі  $n$  показники числові, то ознакові описи являють собою числові вектори

розмірності  $n$ . Можливі й більш складні випадки, коли прецеденти описуються часовими рядами або сигналами, зображеннями, відеорядами, текстами, попарними відношеннями подібності або інтенсивності взаємодії, тощо.

Основні стандартні типи задач навчання за прецедентами [6]:

1. Навчання з учителем (supervised learning) — найпоширеніший випадок. Кожен прецедент являє собою пару «об'єкт, відповідь». Потрібно знайти функціональну залежність відповідей від опису об'єктів і побудувати алгоритм, що приймає на вході опис об'єкту і видає відповідь на виході. Функціонал якості звичайно визначається як середня помилка відповідей, виданих алгоритмом, за всіма об'єктами вибірки. До нього входять:

1. задача класифікації (classification) відрізняється тим, що множина припустимих відповідей скінченна. Її називають мітками класів (class label). Клас — це множина всіх об'єктів з даним значенням мітки;
2. задача регресії (regression) відрізняється тим, що припустимою відповіддю є дійсне число або числовий вектор;
3. задача прогнозування (forecasting) відрізняється тим, що об'єктами найчастіше є відрізки часових рядів, що обриваються в той момент, коли потрібно зробити прогноз на майбутнє. Для вирішення задач прогнозування часто вдається пристосувати методи регресії або класифікації, причому в другому випадку мова йде скоріше про задачі прийняття рішень.

2. Навчання без учителя (unsupervised learning). У цьому випадку відповіді не задаються, і потрібно шукати залежності між об'єктами. Сюди належить:

1. задача кластеризації (clustering) полягає в тому, щоб згрупувати об'єкти в кластери, використовуючи дані про попарну подібність об'єктів. Показники якості можуть визначатися по-різному, наприклад, як відношення середніх міжкластерних і внутрішньокластерних відстаней;
2. задача пошуку асоціативних правил (association rules learning). Вхідні дані представляються у вигляді ознакових описів. Потрібно знайти такі набори ознак, і такі значення цих ознак, які особливо часто (невипадково часто) зустрічаються в ознакових описах об'єктів.

3. Часткове навчання (semi-supervised learning) займає проміжне положення між навчанням з учителем і без учителя. У цьому підході кожен навчальний прецедент складається з пари "об'єкт, відповідь", проте відомі відповіді тільки для обмеженої кількості прецедентів. Прикладом прикладної задачі, де може застосовуватися часткове навчання, є автоматична рубрикація великої кількості текстів, де деякі з них вже класифіковані за рубриками;
4. Трансдуктивне навчання (transductive learning) відрізняється від стандартного навчання тим, що його метою є прогнозування для нових, невідомих тестових прецедентів на основі часткових даних з навчальної вибірки. У цьому випадку не потрібно знаходити загальну закономірність, оскільки немає планів отримати нові тестові дані. Замість цього, можна покращити якість прогнозування шляхом аналізу всієї тестової вибірки в цілому, наприклад, шляхом її кластеризації. У багатьох додатках трансдуктивне навчання практично не відрізняється від часткового навчання;
5. Навчання з підкріпленням (reinforcement learning). Роль об'єктів грають пари «ситуація, ухвалене рішення», відповідями є значення показника якості, що характеризує правильність ухвалених рішень (реакцію середовища). Як і у задачах прогнозування, тут суттєву роль грає фактор часу. Приклади прикладних задач: формування інвестиційних стратегій, автоматичне керування технологічними процесами, самонавчання роботів, тощо;
6. Динамічне навчання (online learning) може бути як навчанням з учителем, так і без учителя. Специфіка в тому, що прецеденти надходять потоком. Потрібно негайно ухвалювати рішення щодо кожного прецеденту й одночасно доучувати модель залежності з урахуванням нових прецедентів. Як і у задачах прогнозування, тут істотну роль грає фактор часу;
7. Активне навчання (active learning) відрізняється тим, що має можливість самостійно призначати наступний прецедент, який стане відомий;
8. Метанавчання (meta-learning або learning-to-learn) відрізняється тим, що прецедентами є раніше вирішені задачі навчання. Потрібно визначити, які з евристик, що у них використовуються, працюють більш ефективно. Кінцева

мета - забезпечити постійне автоматичне вдосконалення алгоритму навчання з часом.

## 5. Методи аналізу

Кластеризація використовується для групування об'єктів в сукупності в однорідні кластери або класи. Уявімо дані вибірки як точки в ознаковому просторі, тоді метою кластеризації є визначення "згущень точок" або існуючих структур. Кластеризація є описовим процесом, що дозволяє провести розвідувальний аналіз і вивчити "структуру даних", але не робить статистичних висновків.

Синонімами терміну «кластеризація» є «автоматична класифікація», «навчання без учителя» й «таксономія».

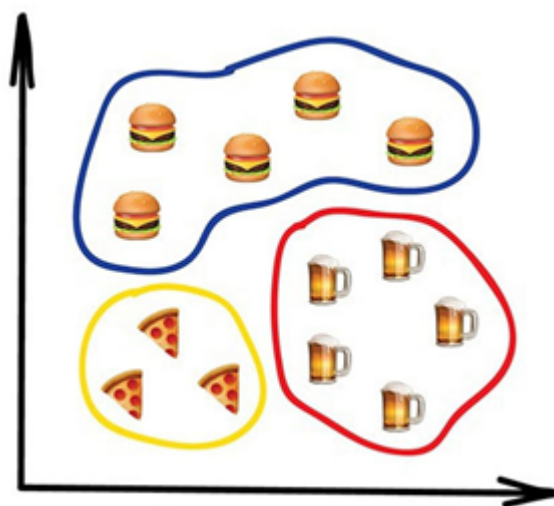


Рисунок 1.2 – Задача кластеризації

Означення терміна "кластер" може варіюватись залежно від конкретного дослідження: кожне дослідження може мати свої власні "кластери". Переклад терміну "cluster" може бути "скупченням" або "гроном". Кластер можна описати як групу об'єктів, що мають спільні властивості. Дві характеристики кластера можуть бути:

1. Внутрішня однорідність: об'єкти всередині кластера схожі між собою за деякими ознаками.
2. Зовнішня ізолюваність: кластери мають відмінність від інших кластерів та інших об'єктів поза кластером.

Найбільше застосування кластеризація спочатку одержала в таких науках як біологія, антропологія, психологія. Для рішення економічних задач кластеризація тривалий час мало використалася через специфіку економічних даних та явищ.

На сьогоднішній день розроблено більше сотні алгоритмів кластеризації.

Процес кластеризації залежить від обраного методу й майже завжди є ітеративним. Процес кластеризації може включати проведення багатьох експериментів, де варіюються різні параметри, такі як міра відстані, тип стандартизації змінних, кількість кластерів і т.д. Однак, ці експерименти не повинні бути самоціллю, оскільки основною метою кластеризації є отримання значущої інформації про структуру аналізованих даних. Отримані результати потребують подальшої інтерпретації, дослідження та вивчення характеристик об'єктів з метою точного опису сформованих кластерів.

Кластерний аналіз застосовується в різних областях. Він корисний, коли потрібно швидко класифікувати велику кількість інформації.

Так, у медицині кластеризація використовується для діагностування захворювань, вибору стратегій лікування. У сфері маркетингу, одним з використань кластеризації є сегментація конкурентів і споживачів для більш ефективного визначення їх потреб і поведінки. У менеджменті, прикладами задачі кластеризації можуть бути розподіл персоналу на різні групи залежно від їх навичок і компетенцій, класифікація споживачів і постачальників для оптимального управління взаєминами, а також виявлення подібних виробничих ситуацій, що можуть призвести до виникнення дефектів або недоліків. У соціології, задача кластеризації полягає у розбитті респондентів на однорідні групи для подальшого дослідження їхніх думок, уподобань або соціальних зв'язків [7].

Задача класифікації подібна із задачею кластеризації, є її логічним продовженням. Класифікація є найбільш простою та одночасно найбільш часто розв'язуваною задачею Data Mining. Через широке застосування задач класифікації важливо чітко розуміти сутність цього поняття. Нижче наведено кілька визначень:

1. Класифікація - це процес розподілу об'єктів або прикладів на дискретні категорії або класи на основі їхніх властивостей або ознак.

2. Класифікація - це завдання машинного навчання, в якому модель навчається передбачати категорію або клас об'єкта на основі його ознак або характеристик.

3. Класифікація - це процес встановлення відповідності між об'єктами і певними позначками або категоріями, що дозволяє відрізнити однорідні об'єкти від різноманітних.

4. Класифікація - це аналітичний метод, за допомогою якого об'єкти групуються в категорії або класи на основі схожості їхніх характеристик або спільних властивостей.

Ці визначення допомагають уявити класифікацію як процес організації об'єктів у відповідні групи або класи залежно від їхніх характеристик і мети аналізу.

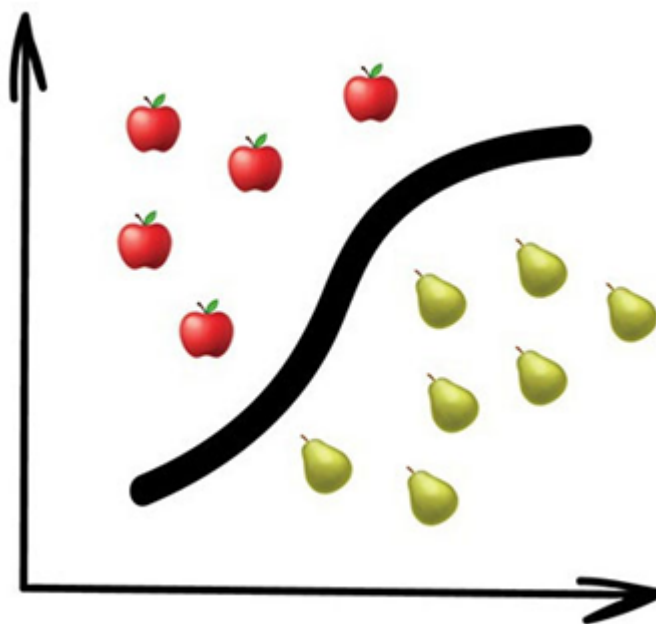


Рисунок 1.3 – Задача класифікації

Класифікація належить до стратегії навчання з учителем (supervised learning), котру також називають контрольованим або керованим навчанням.

Задачею класифікації часто називають прогнозування категоріальної залежної змінної (тобто залежної змінної, що є категорією) на основі вибірки безперервних та/або категоріальних змінних.

Наприклад, можна передбачити, хто із клієнтів фірми є потенційним покупцем певного товару, а хто – ні, хто скористається послугою фірми, а хто – ні, тощо. Цей тип задач відноситься до задач бінарної класифікації, у них залежна змінна може приймати лише два значення (наприклад, так чи ні, 0 або 1).

Інший варіант класифікації виникає, коли залежна змінна може належати до певного набору попередньо визначених класів. Наприклад, це може бути потрібно, коли треба передбачити, яку марку автомобіля збирається придбати клієнт. У таких ситуаціях враховується набір класів для залежної змінної.

Класифікація може бути одномірною, коли здійснюється за однією ознакою, або багатовимірною, коли використовуються дві чи більше ознаки для класифікації.

Багатомірна класифікація була розроблена біологами для вирішення проблеми дискримінації та класифікації організмів. Одним з ранніх досліджень у цьому напрямку є робота Рональда Фішера (1930 р.), в якій організми були розділені на підвиди на основі вимірювання їх фізичних параметрів. Цей підхід дозволяє використовувати не лише одну ознаку, а кілька фізичних параметрів для здійснення класифікації.

Метою процесу класифікації є побудова моделі, яка використовує прогнозуючі атрибути як вхідні параметри і передбачає значення залежного атрибута. Класифікація включає розподіл множини об'єктів на класи за певним критерієм. Класифікатором називається сутність, яка визначає, до якого з певних класів належить об'єкт з урахуванням вектора ознак.

Для математичного класифікаційного підходу необхідно мати формальний опис об'єкта, з яким можна оперувати, використовуючи математичні методи класифікації.

Набір вхідних даних (або вибірку даних) розбивають на дві множини: навчальну й тестову. Навчальна множина – множина, що включає дані, що використовуються для навчання (конструювання) моделі. Така множина містить вхідні й вихідні (цільові) значення прикладів. Вихідні значення призначені для навчання моделі. Тестова множина також містить вхідні й вихідні значення

прикладів. Тут значення цільової змінної використовується для перевірки працездатності моделі.

Процес класифікації складається з трьох етапів: конструювання моделі, оцінка якості моделі та її використання.

На етапі конструювання моделі визначається опис множини визначених класів. Кожен приклад набору даних відноситься до одного з цих визначених класів. У цей період використовується навчальна множина, на основі якої будується модель. Отримана модель може бути представлена класифікаційними правилами, деревом рішень або математичною формулою. Ця модель використовується для подальшої класифікації нових прикладів даних.

Слід зазначити, що оптимальний алгоритм — це той, який покаже себе найкращим чином при перевірці якості класифікації на тестовій вибірці. Найбільш часто використовуваними способами формування тестової вибірки є: перехресна перевірка (cross-validation) і відкладена вибірка. Перехресна перевірка (Cross-validation) є процедурою, яка використовується для оцінки точності класифікації на тестовій множині даних, яка також називається крос-перевірочною множиною. У цій процедурі точність класифікації тестової множини порівнюється з точністю класифікації навчальної множини. Якщо класифікація тестової множини дає приблизно такі ж результати точності, як класифікація навчальної множини, то вважається, що дана модель пройшла крос-перевірку.

Для розподілу вибірки на навчальну та тестову множини часто використовується метод поділу у певній пропорції. Наприклад, можна обрати дві третини даних для навчальної множини і одну третину даних для тестової множини. Цей підхід особливо корисний для вибірок з великою кількістю прикладів. Якщо ж вибірка має малі об'єми, рекомендується застосовувати спеціальні методи, при використанні яких навчальні й тестова вибірки можуть частково перетинатися.

Після побудови моделі можна перейти до використання її для класифікації нових або невідомих значень. Для оцінки правильності та точності моделі використовується тестова множина, де відомі значення порівнюються з

результатами класифікації, здійсненої за допомогою побудованої моделі. Рівень точності визначається як відсоток правильно класифікованих прикладів у тестовій множині. Важливо, щоб тестова множина не залежала від навчальної множини. Якщо модель показує задовільну точність, її можна використовувати для класифікації нових прикладів, чиї класи є невідомими.

Основні характерні риси задач класифікації та кластеризації зазначені в таблиці 1.1.

Таблиця 1.7 - Порівняння класифікації та кластеризації

Характеристика	Класифікація	Кластеризація
Стратегія	Навчання з учителем	Навчання без учителя
Наявність мітки класу	Навчальна множина супроводжується міткою, що вказує клас, до якого відноситься спостереження	Мітки класу навчальної множини невідомі
Підстава для класифікації	Нові дані класифікуються на підставі навчальної множини	Дано множину даних з метою встановлення існування класів або кластерів даних

Для класифікації використовуються різні алгоритми. Основні з них:

- класифікація за допомогою дерев рішень;
- бассівська (наївна) класифікація;
- класифікація за допомогою штучних нейронних мереж;
- класифікація методом опорних векторів;
- статистичні методи, зокрема, лінійна регресія;
- класифікація за допомогою методу найближчого сусіда;
- класифікація за допомогою генетичних алгоритмів.

Оцінювання методів варто проводити, виходячи з наступних характеристик: швидкість, робастність, інтерпретуємість, надійність.

Швидкість характеризує час, що потрібен на створення моделі та її використання.

Робастність, тобто стійкість до яких-небудь порушень вихідних передумов, означає можливість роботи із зашумленими даними й пропущеними значеннями в даних.

Інтерпретуємість забезпечує можливість розуміння моделі аналітиком.

Надійність методів класифікації передбачає можливість роботи цих методів при наявності в наборі даних шумів і викидів [8].

## **6. Аналіз систем машинного навчання**

На сьогоднішній день, існує кілька систем які забезпечують можливість користувачам застосовувати для аналізу даних алгоритми машинного навчання. Зробимо короткий огляд цих систем:

1. Amazon Machine Learning (компанія Amazon) - керований сервіс для побудови моделей машинного навчання (ML) і створення прогнозів, який дозволяє розробляти різноманітні масштабовані інтелектуальні програми [2]. Як відомо, компанія Amazon один з лідерів з надання AWS (Amazon Web Services - хмарні послуги). Розробникам компанії вдалося створити досить потужну систему для аналізу даних і машинного навчання, що підтримує багато сучасних мов. Сервіс використовує потужні алгоритми машинного навчання спільно з інтерактивними візуальними інструментами, що дозволяє з легкістю створювати, оцінювати і розгортати моделі машинного навчання. Після того як модель створена, можна досліджувати її сильні і слабкі сторони і налаштувати продуктивність відповідно до бізнес-цілям. Консоль управління AWS і API надають інструменти візуалізації даних і моделей, а також майстрів, які допомагають створювати моделі машинного навчання, визначати їх якість і виконувати тонке налаштування прогнозів відповідно до потреб

додатків. З недоліків можна виділити високу ціну і закритий код, складність налагодження робочого оточення, орієнтованість на API (в поточному контексті - можливість звертатися до обчислювальних ресурсів віддалено) і на використання серверів компанії.

2. Azure Machine Learning (компанія Microsoft) - хмарний сервіс для виконання завдань прогнозувальної аналітики, за допомогою якого можна легко створювати моделі і інтегрувати їх в промислові рішення [16]. Студія машинного навчання - це місце, де перетинаються обробка і аналіз даних, прогнозна аналітика, хмарні ресурси і дані клієнта. Система дозволяє створювати свої проекти, використовувати вбудовані алгоритми, а також завантажувати свої алгоритми на різних мовах програмування. Дозволяє робити аналіз результатів, збір і зберігання результатів прогнозування з можливістю спільного доступу до них, побудова прогнозних моделей на основі великих даних (Big Data). Сервіс являє собою один з найпростіших інструментів для використання машинного навчання, що прибирає вхідний бар'єр для всіх, хто вирішує використовувати його для своїх потреб. Ніяке програмування не потрібно, виконується тільки візуальне з'єднання наборів даних і модулів, щоб створити модель прогнозувальної аналітики. З недоліків можна виділити досить високу ціну, слабкі можливості налагодження і закрити реалізацію.
3. Google Cloud Machine Learning (компанія Google) [9]. Компанія Google створила по-справжньому "сучасний" продукт, здатний інтегруватися в усі сучасні технології та використовувати самі передові розробки в області машинного навчання. Google Cloud Machine Learning надає можливість, починаючи від отримання і обробки інформації і до прогнозування. Дані для тренування моделей (процедури, що вимагає неабияких комп'ютерних ресурсів) розробники можуть брати з таких інструментів, як Google Cloud Dataflow, Google BigQuery, Google Cloud Dataproc, Google Cloud Storage і Google Cloud Datalab. Як результат, тепер будь-який додаток може скористатися перевагами тих же технологій машинного навчання, якими користуються багато сервісів Google.

Незважаючи на свої переваги, система володіє зайвою складністю, перевантаженістю і високою ціною, укупі з закритим кодом.

У таблиці 1.2 представлені основні відмінності системи, що проектувалась від аналогів.

Таблиця 1.8 - Основні відмінності розробленої системи від аналогів

	Пропріетарна ліцензія	Програмна розширюваність	Швидкий старт	Baseline	Простота установки і розгортання
Amazon Machine Learning	+	+	-	-	+
Azure Machine Learning	+	+	-	-	-
Google Cloud Machine Learning	+	-	-	-	-
PHP-ML	-	+	+	+	+

В ході дослідження були розглянуті ці існуючі рішення.

У всіх подібних систем є недоліки і обмеження. Основними з них є:

- складне налаштування (довгий старт);
- пропріетарна ліцензія;
- не можна змінювати поведінку системи на свій розсуд;
- деякі з них заточені тільки під нейронні мережі.

В даному розділі було розглянуто суть машинного навчання та перелік основних завдань. Також було дано визначення поняття та розглянута

методологія аналізу даних, була розглянута задача класифікації та кластеризації. Розглянуто огляд існуючих аналогів.

## **7. Аналіз методологій управління проектами**

Методологія управління проектами визначає набір принципів і практик, які використовуються для керування проектами з метою досягнення їх оптимальної продуктивності.

Ефективне управління проектами є важливим для організацій і команд, але для досягнення успіху важливо правильно вибрати методологію управління проектами, яка відповідає типу команди, характеристикам проекту, організації та поставленим цілям.

Існує кілька методологій управління проектами, і вибір конкретної залежить від потреб і характеристик вашого проекту.

### **1. Waterfall**

Методологія Waterfall є класичним підходом до управління проектами, де завдання та етапи виконуються послідовно, один за одним, і кожен етап повинен бути завершений перед початком наступного.

У методології Waterfall зазвичай виконуються наступні етапи управління проектом:

1. Вимоги.
2. Аналіз.
3. Дизайн.
4. Розробка.
5. Тестування.
6. Розгортання та обслуговування.

Проект прогресує вперед у лінійному порядку, подібно до справжнього водоспаду.

Але, як і в справжньому водоспаді, цей підхід може бути ризикованим. Оскільки все планується заздалегідь, існує багато можливостей для помилок,

якщо очікування не відповідають реальності. Крім того, немає можливості повернутися до попереднього етапу після його завершення (уявіть, що спроба протистояти потоку води вгору водоспаду — це не дуже ефективно).

**Waterfall методологія управління проектами відповідає таким умовам:**

- 1. Кінцева мета проекту чітко визначена і не підлягає змінам.**
- 2. Зацікавлені сторони мають чітке уявлення про свої вимоги і ці вимоги не змінюються.**
- 3. Проект має лінійну та передбачувану структуру (тобто не передбачається значних змін).**
- 4. Ви працюєте в регульованій галузі, де необхідне детальне відстеження проекту або документація.**
- 5. Можливо, на середині проекту вам доведеться залучити нових учасників і навчити їх працювати швидко.**

## **2. Agile**

Гнучка методологія управління проектами виникла внаслідок розчарування у традиційних лінійних підходах до управління проектами, які не могли пристосовуватись до змін під час його виконання. Відсутність можливості перегляду та внесення змін до проекту по ходу роботи викликала невдоволення. У зв'язку з цим, увага почала переходити до більш ітеративних моделей, які дозволяли командам переглядати проект та вносити зміни за потреби.

Концепція гнучкого управління проектами привела до розвитку кількох конкретних підфреймворків і методологій, таких як Scrum, Kanban і Lean. Однак, що їх об'єднує? Основними принципами гнучкої методології управління проектами є:

- 1. Співпраця:** акцент робиться на командній співпраці, взаємодії та залученні зацікавлених сторін протягом усього процесу проекту.

2. Швидкість: принципи гнучкого управління спрямовані на ефективну та швидку реалізацію проектів шляхом коротких ітерацій та доставки цінності з ранніх стадій проекту.

3. Відкритість до змін, заснована на даних: гнучкі методології визнають необхідність адаптації та змін в проекті, орієнтуючись на дані та сприймаючи зміни як норму.

Ці принципи сприяють більш гнучкому, пристосовуваному та ефективному управлінню проектами в порівнянні з традиційними лінійними методологіями.

Таким чином, гнучкі методології управління проектами зазвичай передбачають виконання проекту у короткі етапи з частим тестуванням, переоцінкою та адаптацією. Багато гнучких методів дозволяють додавати роботу до резерву, який команди можуть опрацювати на кожному етапі або циклі. Керівники проектів або власники продукту визначають пріоритетність резерву, щоб команди знали, на чому зосередитися в першу чергу.

#### **Методологія гнучкого управління проектами (Agile) підходить, якщо:**

1. Ваш проект може бути змінений.
2. Ви не маєте чіткого уявлення про остаточне рішення з самого початку.
3. Вам потрібно працювати швидко, і важливіше бачити швидкий прогрес, ніж досягати ідеальних результатів.
4. Ваші зацікавлені сторони або клієнти потребують (або бажають) бути залученими на кожному етапі проекту.

### **3. Scrum**

Scrum є структурою управління проектами, яка сприяє ефективній командній співпраці над складними проектами розробки продуктів і програмного забезпечення. Це добра новина, оскільки scrum легко зрозуміти. Однак, є й погана новина: його важко впровадити.

Методологія scrum покладає акцент на командну роботу в управлінні проектами. Вона сприяє встановленню підзвітності та досягненню чітко визначеної мети за допомогою ітеративного прогресу. Scrum є частиною

гнучкого розробки програмного забезпечення, де команди практикують гнучкість. Назва походить від виду спорту регбі, де сутички відіграють роль форми, де кожен гравець має свою певну роль, але всі працюють разом для швидкого досягнення стратегії.

Scrum є структурою, що складається з цінностей, ролей, подій і артефактів. Ці компоненти працюють разом, щоб створити гнучку методологію управління проектами, яка допомагає командам ефективно керувати своєю роботою. Основна мета структури Scrum полягає у спрощенні. Вона не є традиційною методологією управління проектами, але виступає як основа для розробки продуктів і програмного забезпечення.

#### **4. Kanban**

Kanban є ще одним методом гнучкого управління проектами. Цей популярний метод економічного керування робочим процесом використовується для визначення, управління та вдосконалення послуг, що забезпечують роботу знань. Він сприяє візуалізації роботи, підвищенню ефективності та постійному вдосконаленню. Робота представлена на дошках Kanban, що дозволяє оптимізувати роботу в кількох командах і виконувати навіть найскладніші проекти в єдиному середовищі. Однак, Kanban - це не просто використання карток для керування своєчасною доставкою. Структура Kanban розроблена для допомоги командам у зменшенні вузьких місць, підвищенні ефективності, поліпшенні якості та збільшенні продуктивності. Канбан ґрунтується на чотирьох принципах і шести основних практиках.

**Принципи методології Канбан можна сформулювати наступним чином:**

1. Почніть зараз і зосередьтеся на поточних завданнях. Ретельно розберіться у діючих процесах, визначте, що працює ефективно, а що потребує вдосконалення.

2. Використовуйте поступовий підхід до змін. Спостерігайте, як повільно вносити зміни до процесів з плином часу, уникайте радикальних трансформацій.

3. Зберігайте наявні ролі. Канбан не пропагує власних унікальних ролей, як це роблять інші фреймворки, наприклад, Scrum. Він акцентує увагу на співробітниках, що вже працюють у команді і мають свої ролі.

4. Сприяйте розвитку лідерства. Підтримуйте стимулювання інновацій та ідей для поліпшення на всіх рівнях. Поширюйте концепцію кожного співробітника як лідера, незалежно від його посади або ролі.

### **5. Вибір методології для управління проектом**

Waterfall методологія управління проектами була обрана для керування всім проектом через свою здатність чітко та послідовно виконувати всі етапи розробки. Оскільки проект стосується розробки веб-сервісу, важливо визначити всі вимоги та процеси з самого початку, і передавати їх інженерам для розробки. Внесення змін у вимоги пізніше може бути недоцільним або дуже витратним.

Вибір Waterfall методології допоможе оцінити бюджет проекту та фіксувати його на початковому етапі. Також це дозволить встановити чіткі терміни, до яких проект буде адаптуватися.

## РОЗДІЛ 2. ОПИС ТА ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЙ

### 1. Обґрунтування вибору технологій для вирішення поставлених задач

Розроблений сервіс повинен мати функціонал для роботи з аналізом даних, тому для створення безперебійно швидкодіючого інтернет ресурсу обов'язково необхідно вибрати які технології використовувати для реалізації тому перед початком розробки вирішив дослідити наявні інструменти та порівняти їх сильні і слабкі сторони, для того, щоб у проекті використати максимально ефективні технології.

#### 1.1. Оцінка можливостей баз даних

PostgreSQL - це безкоштовна об'єктно-реляційна система управління базами даних (СКБД). Вона доступна на багатьох UNIX-подібних платформах, таких як AIX, різні BSD-системи, HP-UX, IRIX, Linux, macOS, Solaris / OpenSolaris, Tru64, QNX, а також для Microsoft Windows.

Основні переваги PostgreSQL включають:

1. Високопродуктивність і надійність механізмів транзакцій і реплікації: PostgreSQL забезпечує міцну підтримку транзакцій, що дозволяє забезпечити цілісність даних і виконувати операції паралельно. Вона також має розширені можливості реплікації для забезпечення високої доступності і надійності системи.

2. Розширювана система вбудованих мов програмування: PostgreSQL підтримує кілька вбудованих мов програмування, таких як PL/pgSQL, PL/Perl, PL/Python і PL/Tcl, що дозволяє розширювати можливості бази даних. Крім цього, є підтримка інших мов, таких як PL/Java, PL/PHP, PL/Py, PL/R, PL/Ruby, PL/Scheme, PL/sh і PL/V8, а також можливість використання C-сумісних модулів.

3. Спадкування: PostgreSQL підтримує концепцію спадкування, що дозволяє створювати таблиці, які успадковують властивості від інших таблиць. Це спрощує організацію даних і забезпечує більш гнучкі можливості управління структурою бази даних.

4. Легка розширюваність: PostgreSQL може бути легко розширена шляхом створення власних типів даних, функцій, операторів та індексів. Це дозволяє адаптувати базу даних під конкретні потреби проекту і виконувати складні операції.

Однією з вагомих переваг PostgreSQL є його спільнота та розробка. Відмінність полягає в тому, що PostgreSQL не контролюється однією компанією, а є проектом з відкритим кодом, в розробці якого беруть участь численні люди та компанії. Це дозволяє внести найновіші досягнення і забезпечити постійний розвиток та підтримку СКБД.

PostgreSQL, який написаний на мові C, поширюється у вигляді набору текстових файлів з вихідним кодом. Для його встановлення потрібно скомпілювати файли на власному комп'ютері і скопіювати їх у відповідний каталог. Вся процедура детально описана в документації.

PostgreSQL є широко використовуваною системою керування базами даних з відкритим вихідним кодом. Вона була розроблена як прототип в Каліфорнійському університеті Берклі в 1987 році під назвою POSTGRES і згодом була активно розширена і вдосконалена. Третя версія POSTGRES, яка включала підтримку одночасної роботи з декількома сховищами даних, вдосконалені механізми запитів та систему внутрішніх правил, знайшла застосування у великих проектах, таких як фінансовий аналіз, моніторинг функціональності потоків, база даних відстеження астероїдів, система медичної інформації та географічні системи. POSTGRES також використовувався як навчальний інструмент в кількох університетах.

Зрозуміло, що подальший розвиток та підтримка вимагають великих зусиль та часу для дослідження баз даних. Офіційний проект у Берклі був зупинений на версії 4.2. Проте в 1994 році Andrew Yu і Jolly Chen розробили інтерпретатор мови SQL, вдосконалили вихідний код і опублікували свою реалізацію в Інтернеті під назвою Postgres95. У 1996 році програмний продукт було перейменовано на PostgreSQL з початковою версією 6.0. З того часу група експертів з баз даних, що добровільно долучилися до проекту, займається його підтримкою та розвитком.

Вільна реляційна система управління базами даних . Розробку та підтримку MySQL здійснює корпорація Oracle.

MySQL є ідеальним рішенням для малих і середніх додатків. Ця система управління базами даних (СУБД) відзначається своєю гнучкістю завдяки підтримці різноманітних типів таблиць. Користувачі можуть вибирати між таблицями типу MyISAM, які підтримують повнотекстовий пошук, та таблицями InnoDB, які дозволяють використовувати транзакції на рівні окремих записів. Така різноманітність дозволяє адаптувати MySQL до конкретних потреб проекту та забезпечує більшу гнучкість в роботі з даними.

Завдяки своїй відкритій архітектурі та GPL-ліцензуванню, СУБД MySQL постійно поповнюється новими типами таблиць. MySQL можна встановити на багато різних платформ, включаючи AIX, BSDi, FreeBSD, HP-UX, Linux, macOS, NetBSD, OpenBSD, OS/2 Warp, SGI IRIX, Solaris, SunOS, SCO OpenServer, UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Server 2003, WinCE, Windows Vista, Windows 7 і Windows 10.

У MySQL є різноманітні API та конектори для таких мов програмування, як Delphi, C, C++, Eiffel, Java, Lisp, Perl, PHP, Python, Ruby, Smalltalk, Component Pascal і Tcl. Також надаються бібліотеки для мови .NET та підтримка ODBC за допомогою драйвера MyODBC.

MongoDB - це система управління базами даних з відкритим вихідним кодом, яка працює у документо-орієнтованому стилі і не вимагає опису схеми таблиць. MongoDB займає проміжне положення між швидкими та масштабованими системами, які працюють з даними у форматі ключ/значення, і реляційними СУБД. Вона надає зручність у формуванні запитів та функціональні можливості.

MongoDB використовує JSON-подібний формат для зберігання документів і має гнучку мову для формування запитів. Вона підтримує створення індексів для різних атрибутів документів, ефективно зберігання великих бінарних об'єктів, журналювання операцій для змін і додавання даних у базу даних. MongoDB може працювати за парадигмою Map/Reduce і підтримує реплікацію і створення відмовостійких конфігурацій.

У MongoDB є вбудовані механізми для забезпечення шардування, що дозволяють розподілити набір даних між серверами на основі певного ключа. Поєднавши шардування з реплікацією даних, можна створити горизонтально масштабований кластер зберігання, у якому відсутня єдина точка відмови (будь-який збій вузла не призводить до зупинки роботи бази даних). MongoDB підтримує автоматичне відновлення після збою та перенесення навантаження з вузла, що вийшов з ладу. Розширення кластера або перетворення одного сервера на кластер можна здійснити без зупинки роботи бази даних шляхом додавання нових машин.

Redis є розподіленим сховищем пар ключ-значення, яке зберігає дані в оперативній пам'яті і може забезпечити довговічність зберігання за бажанням користувача. Це програмне забезпечення з відкритим вихідним кодом, написане на мові ANSI C, і розробляється компанією VMware. Сирцеві тексти проекту поширюються під ліцензією BSD.

Redis надає подібні до Memcached функціональні можливості для зберігання даних у форматі ключ-значення, але з розширеною підтримкою структурованих даних, таких як списки, хеші і множини. На відміну від Memcached, Redis

забезпечує постійне зберігання даних на диску і гарантує збереження бази даних навіть у випадку аварійного завершення роботи. Клієнтські бібліотеки доступні для багатьох популярних мов програмування, включаючи Perl, Python, PHP, Java, Ruby і Tcl.

Redis підтримує транзакції, що дозволяють виконати групу команд як єдину операцію, забезпечуючи непорушеність і послідовність виконання заданого набору команд. В разі проблем транзакції дозволяють відкотити зміни. Всі дані повністю кешуються в оперативній пам'яті, що сприяє досягненню високої продуктивності.

## **1.2 Вибір мови програмування**

При виборі мови програмування для виконання поставлених задач аналізу, можна розглянути PHP як один з варіантів. PHP є скриптовою мовою програмування, спеціально створеною для генерації HTML-сторінок на стороні веб-сервера. Вона є однією з найпоширеніших мов, використовуваних у сфері веб-розробки, разом з Java, .NET, Perl, Python і Ruby. Більшість хостинг-провайдерів підтримують PHP, що забезпечує його широку доступність для проектів.

У випадку PHP, код інтерпретується веб-сервером і перетворюється в HTML-код, який передається на сторону клієнта. На відміну від скриптової мови JavaScript, PHP-код не відображається користувачеві, оскільки браузер отримує готовий HTML-код. Це має позитивний вплив на безпеку, але може знизити інтерактивність сторінок.

Однією з особливостей PHP є його наявність інтерфейсів до багатьох баз даних. У PHP вбудовані бібліотеки для роботи з такими базами даних, як MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase. Завдяки стандарту відкритого інтерфейсу зв'язку з базами даних (ODBC), можна підключатися до будь-яких баз даних, для яких існують драйвери.

Мова програмування PHP є традиційною і знайомою для багатьох програмістів, які працюють в різних областях. Вона включає багато

конструкцій, які запозичені з мови C і Perl. Код PHP схожий на той, який можна знайти в типових програмах, написаних на мовах C або Pascal. PHP є відкритим програмним забезпеченням, що означає наявність його сирцевого коду та безкоштовне використання.

Мова PHP є високорівневою, інтерпретованою та динамічною мовою загального призначення. Вона була створена для обробки HTML-сторінок на стороні сервера, але розширила свої можливості і знайшла застосування в різних областях розробки. PHP є ефективною мовою, і більшість PHP-сценаріїв, особливо невеликих розмірів, обробляються швидше, ніж аналогічні програми, написані на Perl.

Perl, з свого боку, є мовою програмування загального призначення, яка спочатку була розроблена як інструмент для обробки текстових даних. Однак вона широко використовується для вирішення різноманітних завдань, включаючи системне адміністрування, веб-розробку, розробку мережевого програмного забезпечення та програмного забезпечення з графічним інтерфейсом користувача.

Мова надає перевагу практичному (простота у використанні, ефективність, повнота) над гарним (крихітність, елегантність, мінімалістичність). Мо-ва має багато можливостей, включно з підтримкою кількох парадигм програмування (процедурне програмування, об'єктно-орієнтоване програмування а також функціональне програмування), управлінням пам'яттю, вбудованою підтримкою системи обробки текстів і великим набором сторонніх модулів.

Структура Perl має загальні риси, які походять від мови C. Perl є процедурною мовою змінних, виразів присвоювання, блоків коду, окремих фігурних дужок, керуючих структур і функцій.

Однак Perl також запозичує деякі риси з мов командного рядка UNIX. Усі змінні в Perl позначаються провідними символами, які точно вказують тип даних цієї змінної у даному контексті (наприклад, скаляр, масив, хеш). Ці символи дозволяють змінним бути інтерпольованими в рядках. Perl має велику

кількість вбудованих функцій, які надають інструменти, часто використовувані для програмування командного рядка, наприклад, сортування або виклику системних служб.

Perl також запозичує масиви з мови Lisp, регулярні вирази з AWK і sed, а також хеші з AWK. Регулярні вирази в Perl значно спрощують виконання багатьох завдань з розбору, обробки тексту та маніпулювання даними.

У Perl 5 була додана підтримка складних типів даних, першокласних функцій і об'єктної моделі. Об'єктна модель включає посилання, пакети, виклики методів від класу, змінні з лексичним областями видимості і директиви компілятора. Одним з найважливіших покращень, внесених у Perl 5, була можливість використовувати код у вигляді "пакетів" або модулів для повторного використання. Всі версії Perl автоматично виконують типізацію даних і контроль пам'яті. Інтерпретатор знає типи та запити пам'яті кожного об'єкта в програмі, розподіляє і звільняє пам'ять, враховуючи посилання. Автоматичне перетворення даних з одного типу в інший, наприклад, з числа в рядок, відбувається під час виконання. Невірні перетворення типів даних призводять до фатальних помилок.

Ruby є інтерпретованою, повністю об'єктно-орієнтованою мовою програмування з чіткою динамічною типізацією. Вона володіє високою ефективністю розробки програм і поєднує найкращі риси Perl, Java, Python, Smalltalk, Eiffel, Ada і Lisp. Синтаксис Ruby подібний до Perl, а об'єктно-орієнтований підхід нагадує мову програмування Smalltalk. Крім того, вона також запозичує деякі риси з мов програмування Python, Lisp, Dylan і CLU.

Інтерпретатор Ruby поширюється як вільне програмне забезпечення і має багатоплатформову реалізацію. Початковий код проекту поширюється під ліцензіями BSD ("2-clause BSD") і "Ruby", яка посилається на останню версію ліцензії GPL і повністю сумісна з GPLv3.

## **2. Аналіз переваг вибраних методів**

Серед перелічених можливих БД для використання мною була вибрана MySQL завдяки своїй швидкодії та інших параметрів перелічених далі.

MySQL дозволяє отримувати доступ до даних, що зберігаються у базах даних, та здійснювати з ними маніпуляції, забезпечуючи захист даних від пошкоджень і збереження їх цілісності. Вона також надає метадані, необхідні для опису даних. Основна відмінність між системами керування базами даних (СКБД) і реляційними системами керування базами даних (РСКБД) полягає в тому, що остання спеціалізується на роботі з реляційними базами даних, зберігаючи дані у вигляді таблиць і відношень між ними.

MySQL є вільною СКБД з відкритим вихідним кодом, створеною як альтернатива комерційним системам. На сьогоднішній день MySQL є однією з найпоширеніших СКБД і використовується в основному для створення динамічних веб-сторінок, оскільки має велику підтримку різноманітних мов програмування. MySQL надає багатий набір функціональних можливостей, що забезпечують безпечне середовище для зберігання, обслуговування та отримання даних.

Вона відрізняється високою швидкодією, стійкістю і простотою використання, була розроблена з метою покращення продуктивності обробки великих баз даних.

Вихідні коди сервера компілюються на багатьох платформах. Найповніші можливості сервера виявляються в UNIX-системах, де є підтримка багатоканальності, що підвищує продуктивність системи в цілому.

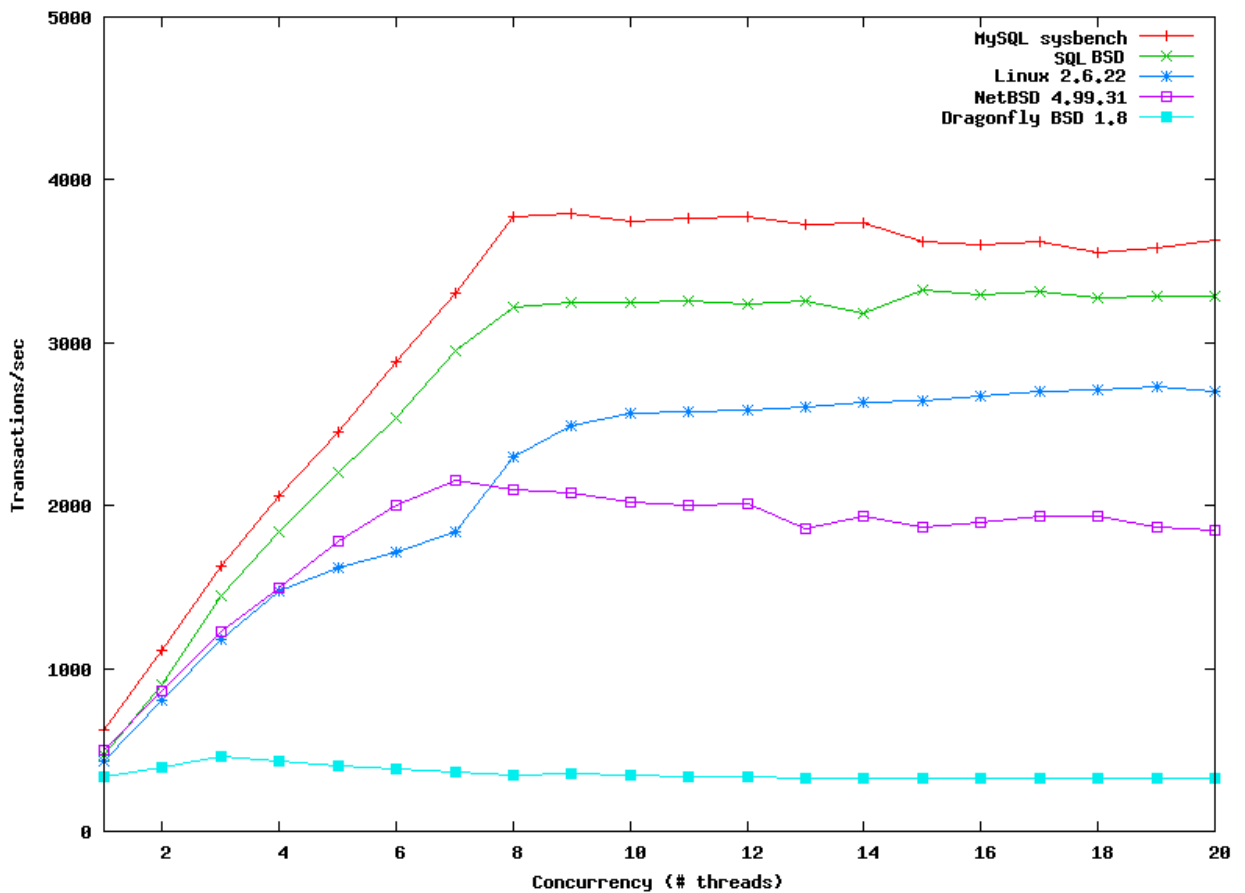


Рисунок 2.1. Швидкодія MySQL

MySQL є безкоштовним для некомерційного використання і має кілька переваг:

1. Простота у встановленні та використанні: MySQL має легкий процес встановлення і простий інтерфейс, що дозволяє швидко розпочати роботу з базами даних.
2. Підтримка багатьох користувачів: MySQL може одночасно обслуговувати необмежену кількість користувачів, що дозволяє багатьом людям працювати з базою даних одночасно.
3. Велика кількість рядків у таблицях: MySQL може зберігати значну кількість рядків у таблицях, і за документацією, навіть до 50 мільйонів записів.

4. Висока швидкість виконання команд: MySQL відомий своєю швидкодією та ефективністю виконання команд бази даних, що забезпечує швидкий доступ до даних.

5. Проста і ефективна система безпеки: MySQL має надійну систему безпеки, яка забезпечує захист ваших даних від несанкціонованого доступу та зберігання.

6. Масштабованість: MySQL може масштабуватися для роботи з великими базами даних. Згідно з документацією, деякі компанії, що використовують MySQL, успішно зберігають до 50 мільйонів записів у своїх базах даних.

MySQL є популярною системою керування базами даних, яка забезпечує широкий функціонал і використовується у багатьох проектах та компаніях.

Переносність: MySQL є переносною системою, оскільки вона може працювати на різних платформах, таких як Unix, Linux, Windows, OS/2, Solaris, Mac OS. Це дозволяє використовувати MySQL на різних операційних системах без проблем.

Зв'язаність: MySQL має мережеву структуру, що дозволяє кільком користувачам одночасно отримувати доступ до бази даних з будь-якої точки Інтернету. Для цього MySQL надає різні програмні інтерфейси додатків (API), такі як C, C++, Perl, PHP, Java, Python, які дозволяють з'єднатися з MySQL з додатків, написаних на цих мовах.

Безпека: MySQL має систему контролю доступу до даних і може забезпечувати шифрування даних при їх передаванні. Вбудована база даних MySQL містить таблиці, які дозволяють налаштовувати, які користувачі з якого домену мають доступ до яких таблиць і які команди можна виконувати. Крім того, паролі, збережені в базі даних, можуть бути зашифровані за допомогою функції `password()` в MySQL.

MySQL - це надійна система керування базами даних з широким функціоналом, яка працює на різних платформах і надає можливості для забезпечення безпеки даних та з'єднання з різними додатками.

Благодаря тому, що код MySQL є відкритим, можна додавати потрібні функції до пакету і розширювати його функціональність за необхідністю.

Внаслідок відкритості коду та безкоштовності програми, стабільності та надійності її роботи, сформувалася спільнота людей, які не лише підтримують MySQL, але й активно беруть участь у розвитку самого пакету і навчанні менш досвідчених людей роботі з ним. Існує велика кількість списків розсилки та конференцій, де можна отримати безкоштовну допомогу у будь-який час.

На сьогоднішній день існують версії програми для більшості поширених комп'ютерних платформ. Це свідчить про те, що вам не нав'язують певну операційну систему. Ви можете вибрати з чим працювати, наприклад, Linux або Windows, і навіть у разі зміни операційної системи ваші дані не втраяться, і не потрібні додаткові інструменти для їх перенесення.

При виборі мови програмування також важливим фактором стала ефективність, особливо при програмуванні для багатокористувацьких середовищ, зокрема для веб-розробки.

Однією з важливих переваг PHP є його "движок". В PHP це не компілятор або інтерпретатор, а транслюючий інтерпретатор. Цей "движок" PHP дозволяє обробляти сценарії з високою швидкістю.

Деякі оцінки свідчать, що більшість PHP-сценаріїв, особливо невеликих розмірів, оброблюються швидше, ніж аналогічні програми, написані на Perl.

PHP надає розробникам і адміністраторам гнучкі і ефективні засоби безпеки, які можна умовно поділити на дві категорії: засоби на рівні системи і засоби на рівні додатку.

У PHP реалізовані механізми безпеки, які перебувають під контролем адміністраторів. Правильне налаштування PHP забезпечує максимальну свободу дій і безпеку. PHP може працювати у так званому "безпечному режимі" (safe mode), який обмежує можливості використання PHP користувачами з деякими важливими параметрами. Наприклад, можна обмежити максимальний час виконання і використання пам'яті (неконтрольоване витрачання пам'яті негативно впливає на швидкодію сервера). Також, адміністратор може встановлювати обмеження на каталоги, в яких користувач може переглядати і виконувати PHP-сценарії, а також використовувати PHP-сценарії для перегляду конфіденційної інформації на сервері, наприклад, файлу "passwd".

Стандартний набір функцій PHP включає надійні механізми шифрування. PHP також сумісний з багатьма додатками від незалежних фірм, що дозволяє легко інтегрувати його з захищеними технологіями електронної комерції (e-commerce). Ще одна перевага полягає в тому, що вихідний код PHP-сценаріїв можна переглянути в браузері, оскільки сценарій компілюється перед його надсиланням за запитом користувача. Реалізація PHP на стороні сервера запобігає викраденню складних сценаріїв користувачами, навіть якщо вони знають достатньо для виконання команди "View Source".

Оскільки PHP є вбудовуваною (embedded) мовою, вона відзначається високою гнучкістю для задоволення потреб розробника. Хоча зазвичай PHP рекомендується використовувати в поєднанні з HTML, вона так само успішно інтегрується з JavaScript, WML, XML та іншими мовами. Крім того, добре структуровані PHP-додатки легко розширюються за потребою (хоча це стосується всіх основних мов програмування).

Відсутність проблем залежності від браузерів також є перевагою, оскільки PHP-сценарії повністю компілюються на стороні сервера перед передачею клієнту. Практично будь-які пристрої з браузерами, включаючи стільникові

телефони, електронні записники, пейджери і портативні комп'ютери, а також традиційні ПК, можуть отримувати PHP-сценарії. Програмісти, які працюють з допоміжними утилітами, можуть також запускати PHP у режимі командного рядка.

Оскільки PHP не залежить від конкретного веб-сервера, користувачі не обмежені вибором певного сервера (навіть якщо він їм незнайомий). PHP працює на різних серверах, таких як Apache, Microsoft IIS, Netscape Enterprise Server, Stronghold і Zeus. І оскільки ці сервери працюють на різних платформах, PHP є переносним і незалежним від платформи, і може функціонувати на UNIX, Solaris, FreeBSD, а також на платформах Windows 95/98/NT/2000/XP/2003.

### 3. Вибір методів для аналізу даних машинним навчанням

Для технологій обраних мною наявна велика кількість бібліотек для роботи з можливостями машинного навчання, серед яких PHP-ML - повноцінна бібліотека для роботи зі штучним інтелектом, розроблена командою PHP-AI.

Останнім часом PHP-ML стала однією з найпопулярніших на GitHub. Як кажуть самі творці, пакет є «свіжим підходом до машинного навчання на PHP». З вимог - встановлена версія інтерпретатора не нижче 7.0. Додати розробку до свого проекту можна через Composer.

- Прості та ефективні інструменти для пошуку даних та аналізу даних
- 20+ готових до використання алгоритмів
- Побудований із звичайним PHP
- Ніяких додаткових залежностей
- Відкритий комерційний доступ - ліцензія MIT

Для виконання поставленого завдання в обраній мною бібліотеці наявний метод MLPClassifier.

Ця мережна архітектура є найбільш поширеною в даний час. Вона була запропонована в 1986 році у роботі Rumelhart, McClelland і детально розглядається у багатьох підручниках з нейронних мереж. Кожен елемент мережі обчислює зважену суму своїх вхідних сигналів з урахуванням доданку корекції, а потім проходить цю активацію через передавальну функцію, що визначає вихідне значення елемента. Елементи мережі організовані в шарову топологію з однонаправленим потоком сигналу. Цю мережу можна розглядати як модель вхід-вихід, де ваги та порогові значення (зміщення) є параметрами моделі. Така мережа може моделювати функцію будь-якого рівня складності, і кількість шарів та кількість елементів у кожному шарі

визначають складність функції. Визначення кількості проміжних шарів і елементів у них є важливим аспектом при конструюванні MLP.

Кількість вхідних та вихідних елементів є унікальною для кожного завдання. Після визначення кількості шарів і елементів у кожному з них, наступним кроком є знаходження оптимальних значень ваг та порогів мережі, які мінімізують помилку прогнозу, що видається мережею. Для цього застосовуються спеціальні алгоритми навчання.

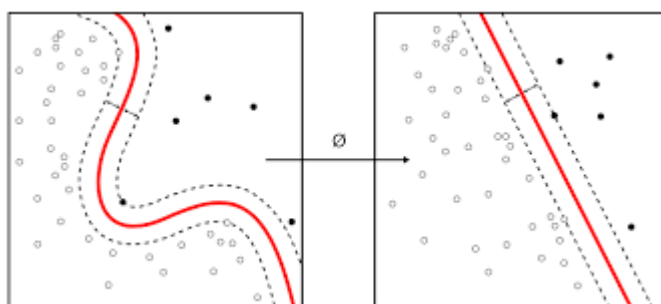


Рисунок 2.2. Класифікація даних

За допомогою зібраних історичних даних, ваги та порогові значення автоматично коригуються з метою мінімізації помилки. Цей процес фактично полягає в тому, що модель, реалізована мережею, пристосовується до наявних навчальних даних. Помилка для конкретної конфігурації мережі визначається шляхом пропуску всіх наявних спостережень через мережу та порівняння отриманих вихідних значень з бажаними (цільовими) значеннями. Усі ці різниці підсумовуються у функцію помилок, яка відображає величину помилки мережі. Зазвичай використовується опція суми квадратів помилок, де кожна помилка вихідного елемента для кожного спостереження обчислюється в квадраті і потім сумується. При використанні пакету RНР-ML, користувачеві надається середньоквадратична помилка (RMS) - це значення нормується на кількість спостережень та змінних, а потім обчислюється квадратний корінь. Це дуже ефективна міра помилки, усереднена за всіма навчальними даними та вихідними елементами.

У лінійному моделюванні можна використовувати алгоритми для визначення конфігурації моделі, яка мінімізує помилку до абсолютного

мінімуму. Однак, використання нейронних мереж, які надають більш широкі (нелінійні) можливості моделювання, має свою ціну. При коригуванні мережі для мінімізації помилки, ніколи не можна бути повністю впевненим, що досягнуто абсолютного мінімуму помилки.

У цьому контексті виникає поняття поверхні помилок, яке є дуже корисним. Кожній вазі і порозі мережі (тобто вільним параметрам моделі, яких у загальному маємо  $N$ ) відповідає одне вимірювання у багатовимірному просторі.  $N + 1$ -ше вимірювання відповідає помилці мережі. Для різних комбінацій ваг, відповідна помилка мережі може бути представлена точкою у  $N + 1$ -вимірному просторі, і всі такі точки утворюють поверхню - поверхню помилок. Мета навчання нейронної мережі полягає у знаходженні найнижчої точки на цій багатовимірній поверхні.

У разі лінійної моделі з сумою квадратів в якості опції помилок ця поверхня помилок буде являти собою параболоїд (квадрік) - гладку поверхню, схожу на частину поверхні сфери, з єдиним мінімумом. У такій ситуації локалізувати цей мінімум досить просто.

У випадку нейронних мереж поверхня помилок має складну структуру і включає різноманітні особливості, які можуть бути небажаними. Наприклад, вона може містити локальні мінімуми (точки, які є найнижчими у певній околиці, але не досягають глобального мінімуму), плоскі ділянки, сідлові точки та довгі вузькі яри.

Аналітичними методами неможливо точно визначити положення глобального мінімуму на поверхні помилок, тому процес навчання нейронної мережі полягає в експлорації цієї поверхні. Починаючи з випадкової початкової конфігурації ваг і порогів (точки на поверхні помилок), алгоритм навчання поступово шукає глобальний мінімум. Зазвичай це досягається обчисленням градієнту (нахилу) поверхні помилок у поточній точці, а потім використанням цієї інформації для спуску по схилу. В кінці кінців, алгоритм

зупиняється в нижній точці, яка може бути локальним мінімумом (або, якщо є удача, глобальним мінімумом).

Для навчання нейронної мережі широко використовується алгоритм зворотного поширення, який є простим у розумінні і має свої переваги у багатьох випадках. У цьому алгоритмі обчислюється вектор градієнта поверхні помилок, який вказує напрямок найкоротшого спуску по поверхні з поточної точки. Таким чином, роблячи невеликі кроки у цьому напрямку, помилка зменшується. Послідовність таких кроків, зменшуючи швидкість під час наближення до "дна", урешті-решт приводить до досягнення мінімуму. Однак існує виклик щодо вибору довжини кроків.

Якщо обрати велику довжину кроку, то збіжність алгоритму буде швидшою, але існує ризик перестрибнути через оптимальне рішення або відхилитися в неправильний напрямок, особливо якщо поверхня помилок має складну форму. Класичним прикладом цього явища є випадок, коли алгоритм повільно рухається по вузькому яру з крутими схилами, переходячи з одного боку на інший. З іншого боку, при маленькому кроці можна спіймати правильний напрямок, але це може вимагати багато ітерацій. На практиці, величина кроку обирається пропорційно крутизні схилу (зменшуючи швидкість поблизу мінімуму) з використанням певної константи, яку називають швидкістю навчання. Вибір правильної швидкості навчання залежить від конкретної задачі і, зазвичай, встановлюється на основі досвіду. Крім того, ця константа може змінюватись з часом, зменшуючись по мірі просування алгоритму.

Зазвичай цей алгоритм модифікується шляхом включення додаткового параметра імпульсу або інерції. Цей член сприяє просуванню в фіксованому напрямку, що дозволяє "прискорити" алгоритм, якщо було зроблено кілька послідовних кроків у тому ж напрямку. Це іноді допомагає уникнути локальних мінімумів та швидше проходити плоскі ділянки на поверхні помилок.

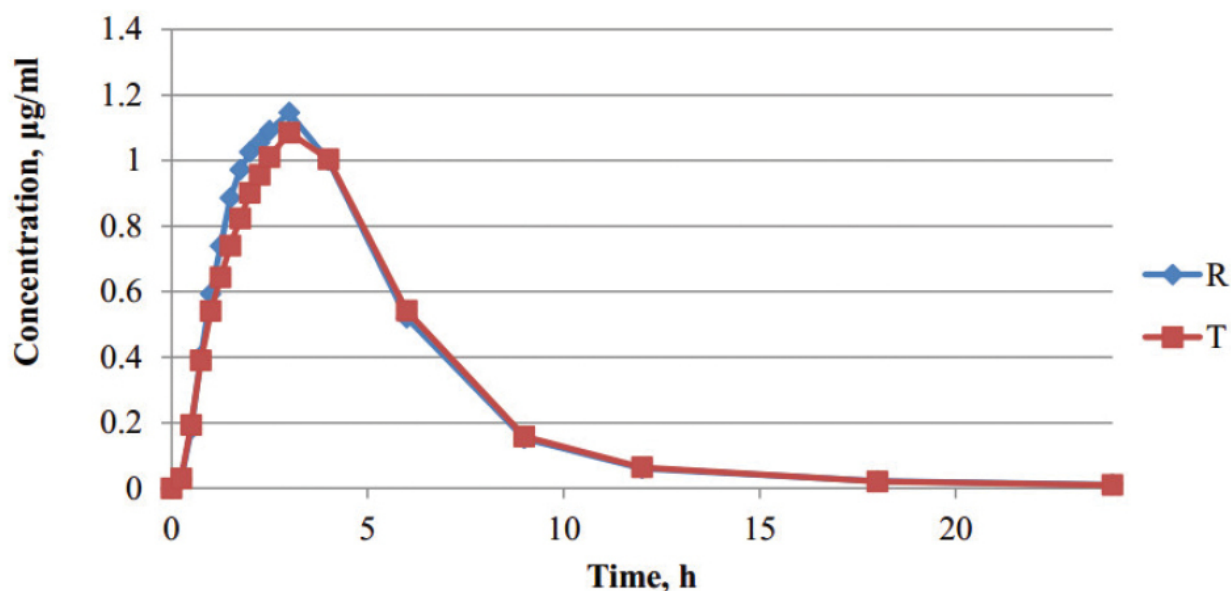


Рисунок 2.3. Візуалізація дії алгоритму

Таким чином, алгоритм навчання діє ітеративно і складається з кроків, які називаються епохами. На кожній епісі навчальні спостереження послідовно подаються на вхід мережі, а потім порівнюються вихідні значення мережі з цільовими значеннями, і обчислюється помилка. Значення помилки та градієнт поверхні помилок використовуються для коригування ваг. Цей процес повторюється знову і знову. Початкова конфігурація мережі вибирається випадковим чином, і процес навчання завершується, коли пройдено задану кількість епох, досягнуто певного рівня помилки або коли помилка перестає зменшуватися.

## 1. Алгоритм кластеризації k-means

Кластеризація методом k-середніх (k-means clustering) є популярним методом для групування об'єктів в однорідні кластери. Основна мета цього методу полягає в розділенні  $n$  спостережень на  $k$  кластерів таким чином, щоб кожне спостереження належало до кластера з найближчим середнім значенням. Цей метод базується на мінімізації суми квадратів відстаней між спостереженнями та центрами їх кластерів.

Основними перевагами методу k-середніх є його простота та швидкість виконання. Він особливо зручний для кластеризації великої кількості спостережень порівняно з ієрархічним кластерним аналізом, оскільки у методу k-середніх не виникає проблема перевантажених та менш наочних дендрограм.

Незважаючи на очевидні переваги, метод k-середніх також має певні недоліки. Результати класифікації сильно залежать від випадкових початкових позицій кластерних центрів. Алгоритм також чутливий до викидів, які можуть спотворити середнє значення. Крім того, кількість кластерів повинна бути передбачена дослідником заздалегідь.

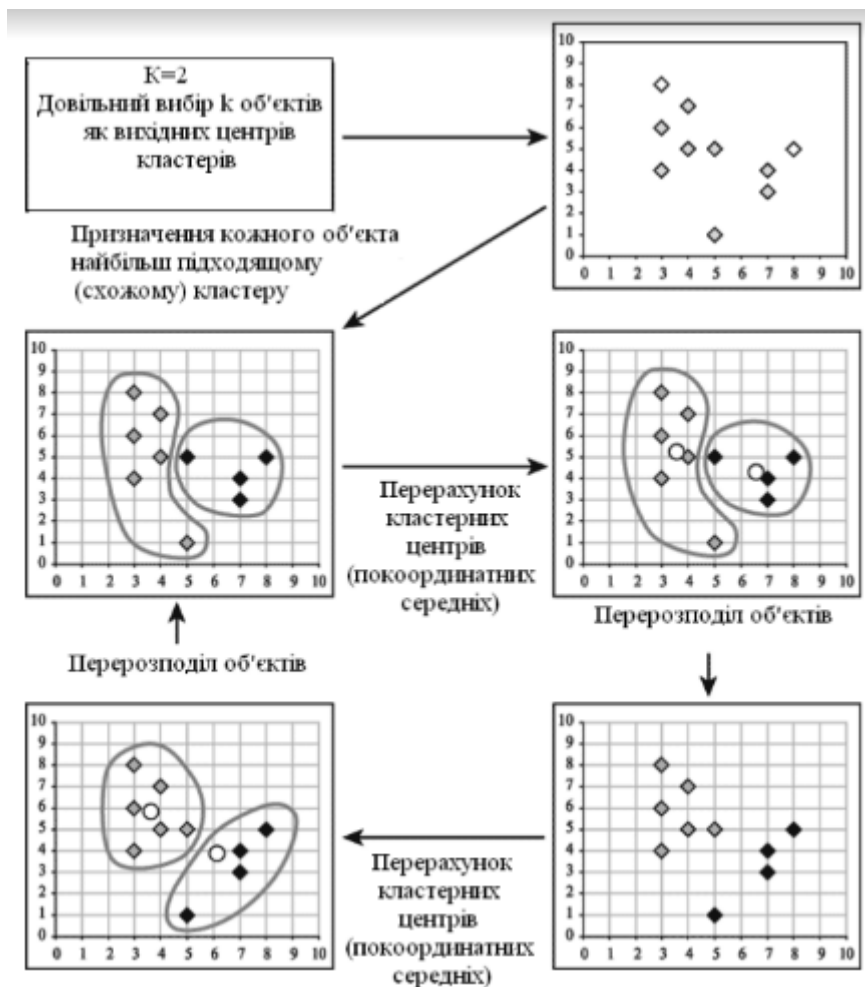


Рисунок 2.4 – Приклад роботи алгоритма k-середніх( $k=2$ )

## 2. Метод опорних векторів

Метод опорних векторів (Support Vector Machines, SVM) є одним з найважливіших і популярних методів машинного навчання, який був запропонований Вапніком у 1963 році. За останнє десятиліття метод SVM знайшов широке застосування в задачах класифікації, регресії та виявлення новизни. Його важливою властивістю є те, що визначення параметрів моделі зводиться до задачі випуклої оптимізації (convex optimization), що дозволяє знаходити глобальний оптимум.

Основна ідея класифікації за допомогою SVM полягає у визначенні поняття "поділ" (margin), який відображає мінімальну відстань від гіперплощини до екземплярів у навчальній вибірці. Головна мета полягає у знаходженні гіперплощини поділу, яка максимізує значення поділу. Розташування цієї гіперплощини визначається опорними векторами, які є підмножиною точок даних. Для створення потужних класифікаторів у 1992 році було запропоновано використання нелінійних ядер, що дозволяють перейти від скалярних добутків до більш складних функцій ядра [14]. Це дозволяє будувати нелінійні гіперплощини поділу, але при цьому не потребує безпосереднього доступу до початкових даних. До найпоширеніших ядер відносяться поліноміальне, радіальна базисна функція, радіальна базисна функція Гауса та сигмоїдне ядро. Ці ядра забезпечують гнучкість та ефективність методу SVM при розв'язанні різноманітних завдань машинного навчання.

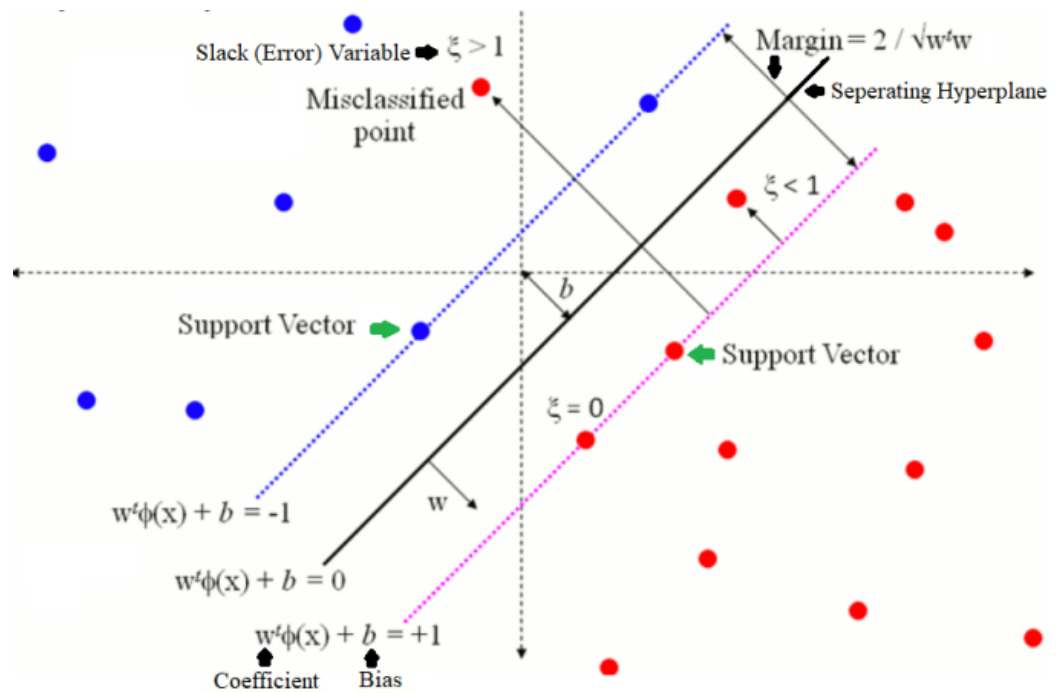


Рисунок 2.5 - Приклад роботи алгоритма SVM

Метод опорних векторів (SVM) має кілька переваг, таких як здатність збігатися до глобального мінімуму та ефективність при обмеженій кількості навчальних даних [15].

Проте, існують певні недоліки у методі:

1. Модель SVM визначає новий приклад як належний до певної категорії, що робить її неімовірнісною, а лише бінарним класифікатором. Це означає, що SVM не надає ймовірнісних оцінок для належності до кожної категорії.

2. Використання ядер дозволяє обробляти нелінійно роздільні дані, але не гарантує надійного узагальнення до векторів, що знаходяться далеко від навчальних даних. Це може призводити до неправильної класифікації випадкових векторів, які не зустрічалися під час навчання.

3. Процес навчання SVM є повільним і може бути обтяжливим для обробки великих обсягів даних. Це обмежує його застосування в задачах аналізу даних, де швидкість обробки є критичною.

Необхідно урахувати ці недоліки при використанні методу опорних векторів і за потреби розглядати альтернативні підходи для вирішення конкретних завдань машинного навчання.

### 3. Статистичний метод логістичної регресії

Логістична регресія є статистичним методом регресії, який використовується для моделювання залежності між векторною змінною та скалярним вихідним значенням. Цей метод є узагальненням лінійної регресії, де використовується softmax функція, і він застосовується, коли залежна змінна може приймати лише обмежену множину значень.

Для оцінки параметрів логістичної регресії часто використовується валідаційна вибірка. Зазвичай це відбувається за допомогою методу максимальної правдоподібності, де параметри обираються таким чином, щоб максимізувати значення функції правдоподібності для даної вибірки.

У такий спосіб логістична регресія дозволяє побудувати модель, яка враховує нелінійні залежності між змінними та може бути використана для класифікації або передбачення значень залежної змінної на основі вхідних даних [17].

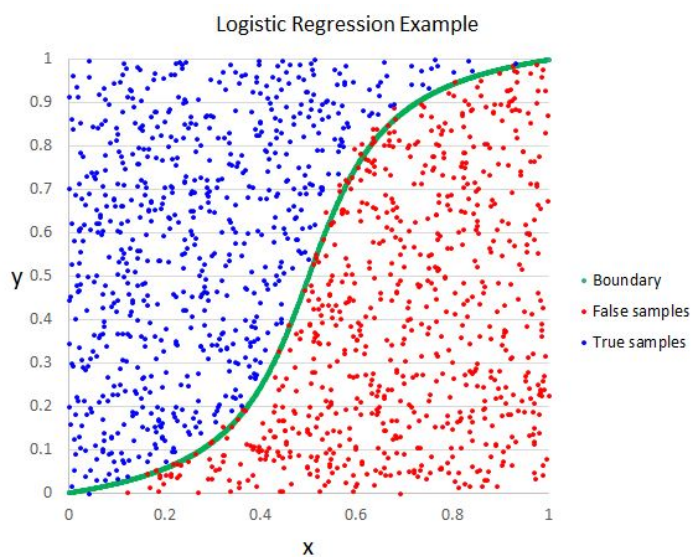


Рисунок 2.6 - Приклад роботи алгоритма логістична регресія

Логістична регресія має важливу перевагу порівняно з іншими моделями та алгоритмами - вона надає можливість оцінити результат у вигляді ймовірності належності до певного класу [18]. Це означає, що вона може дати відповідь про ймовірність належності об'єкта до конкретного класу, що є важливим для багатьох задач.

Проте варто зазначити, що логістична регресія має деякі недоліки. Один з них полягає в обмеженій здатності побудувати складну гіперплощину, що може призвести до менш точного розпізнавання об'єктів. У деяких випадках модель може мати обмежену спроможність адаптуватися до складних шаблонів та нелінійних залежностей між змінними. Тому в окремих ситуаціях існує потреба у більш потужних алгоритмах для досягнення високої точності класифікації.

#### **4. Метод аналізу Random Forest**

У випадку якщо одиночні моделі не задовольняють вимогам якості класифікації використовують ансамблеві моделі.

Унікальний текст:

Random forest є одним із популярних методів машинного навчання, що базується на використанні ансамблю дерев рішень [19]. Цей метод знаходить застосування у різних задачах, включаючи класифікацію, регресію та кластеризацію.

Дерева рішень у random forest будуються на основі навчальної вибірки з використанням концепції інформаційної ентропії. На кожному вузлі дерева обирається атрибут, що найбільш ефективно розбиває навчальну множину на підмножини з максимальною розрізненістю. Основним критерієм для вибору атрибуту є нормалізований приріст інформації. Атрибут з найбільшим нормалізованим приростом інформації обирається для прийняття рішення щодо поділу даних у вузлі дерева.

Більшість алгоритмів побудови дерев рішень є "жадібними", тобто на кожному вузлі обирається найкращий атрибут для поділу даних у поточний момент. Це призводить до того, що алгоритм не може повернутись назад та змінити вже обраний атрибут, що може обмежити досягнення оптимального розбиття.

Для навчання кожного дерева рішення в random forest використовується технологія "статистичного бутстрепа" (bootstrapping), коли навчальна вибірка розбивається на частини, які можуть містити спільні елементи. Це дозволяє уникнути великої кореляції між деревами рішень, що може виникнути, коли вони навчаються на одних і тих самих даних [20]. Такий підхід сприяє покращенню роботи ансамблю та забезпечує більш точні результати класифікації..

У зв'язку з великою чутливістю класифікатора до навчальних даних, навіть невелика зміна в навчальній вибірці може призвести до зміни поділу на підмножини. Це може вплинути на результати класифікації і призвести до зовсім інших висновків.

Однак, для зменшення чутливості та попередження перенавчання використовуються різні технології, серед яких прорідження (pruning) вже побудованого дерева є ефективним підходом.

Прорідження дерева полягає у видаленні частини гілок або листків, що не мають значущого внеску у класифікацію. Це дозволяє спростити модель і зменшити її складність, одночасно покращуючи її загальну здатність до узагальнення.

Прорідження може здійснюватися різними способами, наприклад, за допомогою критеріїв важливості атрибутів, які враховують значення атрибутів у розбитті дерева. Атрибути, які не мають значущого внеску, можуть бути відкинуті, що сприяє спрощенню моделі і зменшенню ризику перенавчання.

Застосування прорідження дерева є важливим кроком у поліпшенні якості класифікації та забезпеченні його стійкості до змін в навчальних даних. Це дозволяє отримати більш узагальнену модель, яка може ефективно працювати з новими невідомими даними.

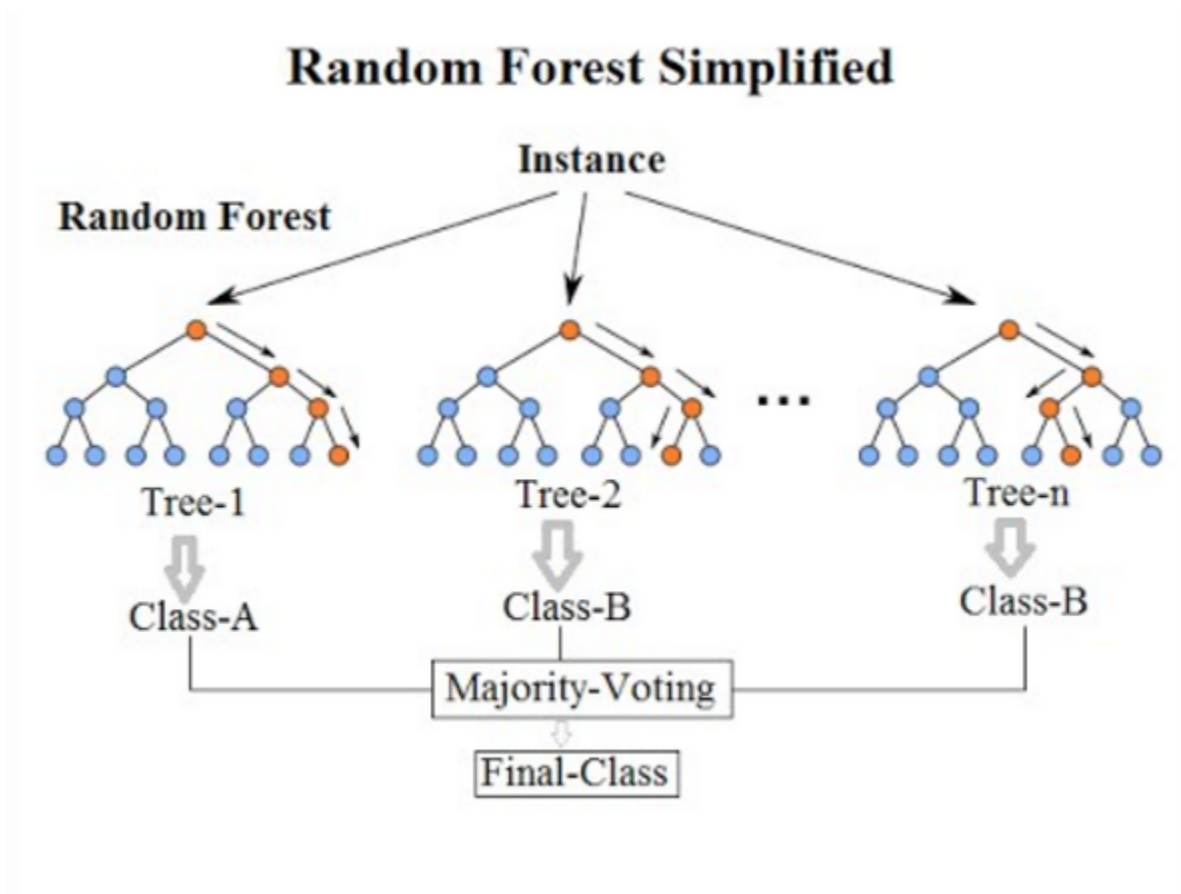


Рисунок 2.7 - Приклад роботи алгоритма Random Forest

Один з основних переваг алгоритму полягає у можливості оцінювання значущості ознак у моделі. Це дозволяє визначити, які атрибути мають найбільший вплив на результат класифікації або регресії. Таке знання може бути корисним для зрозуміння причинно-наслідкових зв'язків у даних та вибору найбільш інформативних ознак для подальшого аналізу.

Крім того, алгоритм Random Forest відомий своєю високою швидкістю навчання. Завдяки використанню ансамблю дерев рішень і паралельному обчисленню, цей метод може швидко працювати з великими обсягами даних, що робить його ефективним для великих проектів.

Ще однією перевагою є легка інтерпретація отриманої моделі. Зрозумілість та зручність пояснення результатів класифікації або регресії дають змогу аналізувати та використовувати модель навіть користувачам без глибоких знань у машинному навчанні.

Проте, алгоритм Random Forest має свої недоліки. Одним з них є його схильність до перенавчання, особливо в умовах багатьох рівнів шумів у даних. Це може призвести до занадто точного врахування випадкових шумів і зниження загальної здатності моделі до узагальнення на нові дані.

Крім того, через використання ансамблю дерев рішень, отримані моделі можуть мати великий розмір. Це може бути проблематичним у випадку обробки великих обсягів даних або в умовах обмеженої пам'яті. Необхідно враховувати цей аспект при використанні алгоритму Random Forest.

Незважаючи на недоліки, Random Forest залишається потужним і популярним методом машинного навчання, який знаходить застосування в різних сферах, включаючи класифікацію, регресію і кластеризацію даних.

В даному розділі було розглянуто перелік завдань які були поставлені при створенні сервісу та обґрунтовано вибір використаних технологій. Розглянуто основні алгоритми для вирішення задач класифікації. В якості базових були обрані саме ці алгоритми оскільки:

1. вони найпростіші в кодовій реалізації;
2. досить швидкі (крім Random Forest);
3. володіють можливістю для тонкого налаштування.

## **5. Мета, цілі та завдання проекту**

**Основною метою проекту є забезпечення збору і аналізу даних туристів з метою планування та детального формування подорожей за кордон. Проект має такі конкретні цілі:**

1. Забезпечити користувачам можливість внесення своїх даних. Це означає, що користувачі зможуть внести інформацію про свої туристичні плани, перевезення, проживання, харчування та інші деталі своєї подорожі.

2. Розробити веб-застосунок, який буде зручним для користувачів. Це означає створення інтуїтивно зрозумілого і простого у використанні інтерфейсу, який дозволить користувачам легко внести та переглянути свої дані.

3. Залучити до співпраці сервіси бронювання авіа- та залізничних квитків, житла, оренди автомобілів, ресторанного бізнесу та інших сфер, що пов'язані з подорожами. Це дозволить користувачам здійснювати бронювання та отримувати доступ до різноманітних послуг, пов'язаних з подорожами, безпосередньо через веб-сервіс.

4. Забезпечити веб-сервіс різноманітним функціоналом, який буде корисним та зручним для користувачів. Це може включати функції планування маршруту, рекомендації щодо визначних місць, відгуки та оцінки від інших користувачів, можливість спілкування з іншими туристами та інше.

5. Залучити нових користувачів до веб-сервісу шляхом проведення маркетингових активностей та просування проекту. Це може включати рекламні кампанії, участь у туристичних виставках та подіях, співпрацю з туристичними агентствами та інші заходи залучення цільової аудиторії.

6. Популяризувати веб-сервіс і забезпечити його визнання серед туристичної спільноти та інших зацікавлених осіб. Це може включати створення привабливого бренду, активну присутність у соціальних мережах, співпрацю зі впливовими особистостями в галузі туризму та інші маркетингові заходи.

Виконання цих цілей допоможе створити успішний веб-сервіс, який забезпечить користувачам зручність, інформаційну підтримку та можливість детального планування та формування своїх міжнародних подорожей.

**Завдання проекту включають наступні етапи:**

1. Створення концепції сервісу планування подорожей на основі рекомендацій і аналізу даних. Це передбачає визначення основних функціональних можливостей сервісу, аналіз ринку та конкурентів, визначення цільової аудиторії та її потреб.

2. Створення технічного завдання для розробки програмного продукту. В цьому етапі визначаються основні вимоги до функціональності, архітектури, інтерфейсу та інших аспектів розробки сервісу.

3. Залучення Front-end та Back-end розробників. Фронтенд розробники відповідають за розробку користувацького інтерфейсу, а бекенд розробники відповідають за розробку серверної частини та бази даних.

4. Узгодження дизайну сервісу для веб та мобільних версій. Дизайнери працюють над створенням привабливого та зручного дизайну, враховуючи вимоги до користувацького досвіду та бренду проекту.

5. Розробка програмного продукту. Команда розробників реалізує задуману функціональність та архітектуру, забезпечує взаємодію з базою даних та іншими сервісами.

6. Тестування отриманого продукту. Проводяться різні види тестування для перевірки функціональності, безпеки та відповідності вимогам.

7. Залучення юристів для затвердження правових відносин з користувачами. Юристи розробляють та переглядають угоди, політики конфіденційності та інші правові документи, що регулюють взаємовідносини з користувачами сервісу.

8. Залучення спеціалістів з маркетингу. Маркетологи розробляють стратегію маркетингу, визначають цільову аудиторію, проводять дослідження ринку та конкурентів, розробляють маркетингові акції та кампанії.

9. Розробка та запуск маркетингової кампанії для популяризації продукту. Використовуються різні канали комунікації та реклами для залучення нових користувачів та підвищення свідомості про сервіс.

10. Налагодження партнерських зв'язків із міжнародними туристичними компаніями та сервісами для подальшої співпраці. Укладаються угоди та співпрацюють з іншими гравцями туристичного ринку для покращення послуг та розширення можливостей для користувачів сервісу.

Виконання цих завдань допоможе реалізувати проект з розробки сервісу планування подорожей та досягти його успіху на ринку туристичних послуг.

## РОЗДІЛ 3 АНАЛІЗ ІНСТРУМЕНТІВ ТА МЕТОДІВ УПРАВЛІННЯ ПРОЕКТУ

### 1. Управління змістом проекту

Проект зазвичай передбачає координацію зусиль з метою створення чогось унікального, будь то продукт, сервіс або досягнення конкретного результату. Його завершення настає, коли поставлені цілі досягнуті, або коли реалізація проекту стає неможливою або непотрібною. Управління змістом проекту тісно пов'язане з управлінням термінами, вартістю та якістю проекту. Хоча менеджер проекту зазвичай відповідає за управління змістом, часом, вартістю, людськими ресурсами та іншими аспектами, іноді для керування змістом проекту залучають бізнес-аналітика.

Створення веб-додатку є неодмінною частиною добре організованого комплексного процесу, метою якого є надання продукту, що відповідає потребам користувачів. Цей процес включає кілька етапів, на яких необхідно правильно планувати, проектувати та реалізовувати продукт, враховуючи важливі аспекти.

Ми визначили 6 основних етапів, за якими працюємо з самого початку. Грамотне планування діяльності є ключовим фактором успіху, тому ми завжди дотримуємося перевірених методів і діємо ефективно, керуючись конкретним планом.

На кожному етапі розробки програмного продукту важливо точно розуміти масштаб проекту, його специфіку та поставлені цілі. Це надає нам базу знань для подальших кроків та дозволяє діяти ефективно, відповідно до очікувань.

Зрозуміло, що процес розробки завжди займає певний час, і кінцевий термін залежить від обсягу робіт та типу функціоналу.

Перед тим, як розробники приступають до написання першого рядка коду, ми проводимо стартову зустріч, на якій знайомимося з командою проекту, обговорюємо подальшу роботу, визначаємо правила комунікації та

методи керування проектом. Цілі проекту повинні включати такі фактори, як вартість, терміни та якість, і бути визначеними з урахуванням конкретних параметрів.

У процесі планування змісту проекту використовується договір або інша форма документу, який містить опис продукту проекту. Результатом планування змісту є Опис проекту, який стає документальною базою для подальших рішень щодо проекту та підтвердження або розширення загального розуміння проекту його учасниками.

1) Визначення змісту - це процес, який полягає у розбитті основних результатів проекту на менші, керовані компоненти, щоб сформувати базовий план проекту. Цей процес включає декомпозицію глобальної мети проекту, складання структури робіт проекту (WBS), визначення переліку робіт проекту, їх тривалостей та обмежень, а також встановлення зв'язків між роботами та розподілу ресурсів для виконання робіт. Головні цілі визначення змісту полягають у підвищенні точності оцінок вартості, тривалості робіт і ресурсів, формуванні базового плану для вимірювання виконання та управління проектом, а також у забезпеченні чіткого розподілу відповідальності. Вхідними даними для цього процесу є контракт, опис проекту і наявні ресурси, а вихідним результатом є базовий план проекту.

2) Підтвердження змісту - це процес, який включає формалізацію прийняття результатів виконання етапів та робіт проекту. Цей процес передбачає формальне приймання змісту проекту учасниками проекту, такими як інвестор, клієнт або замовник. Результатом процесу підтвердження змісту є документація, яка підтверджує факт того, що замовник (клієнт, інвестор) прийняв продукт або основні результати фази (етапу, роботи) проекту.

3) Управління зміною змісту - це процедура, яка визначає, як вносяться зміни до змісту проекту (базового плану). Зміни можуть бути спричинені зовнішніми подіями, помилками при визначенні змісту продукту або формуванні базового плану, змінами, що додають цінність результату проекту, а також реагуванням на ризики проекту шляхом переходу на

резервний план або план обходу. Для проектів, що виконуються за контрактом, управління зміною змісту проекту повинно відповідати умовам контракту. Управління зміною змісту має на меті забезпечити ефективне внесення змін і збереження контролю над змістом проекту.

Управління термінами проекту включає процеси, які забезпечують своєчасне виконання плану проекту. Основними підпроцесами управління термінами проекту є визначення складу робіт проекту, визначення взаємозв'язків робіт, оцінка тривалостей робіт, складання розкладу проекту та управління розкладом. Вхідними даними для цього процесу є контракт, структура робіт проекту (WBS) і опис проекту, а вихідним результатом є тимчасовий план-графік проекту (базовий план проекту). При реалізації управління розкладом на фазі виконання проекту головним завданням керівника проекту є раціональний вибір ресурсів (матеріальних, тимчасових, людських, фінансових) з метою запобігання зривам термінів проекту.

Управління вартістю проекту включає комплекс процесів, які гарантують виконання та завершення проекту в рамках установленого бюджету. Основними складовими цього підходу є наступні етапи:

- 1) Планування ресурсів: на цьому етапі визначаються необхідні ресурси, такі як людські ресурси, обладнання та матеріали, і встановлюється їхня кількість для успішного виконання проектних робіт. Планування ресурсів дозволяє точно визначити потреби проекту в ресурсах і розподілити їх ефективно.
- 2) Оцінка вартості: цей процес передбачає оцінку вартості необхідних ресурсів для виконання проектних робіт. Оцінка вартості може здійснюватися шляхом використання експертних оцінок, аналогічних проектів або параметричних оцінок на основі встановлених нормативів. Це допомагає визначити загальну вартість проекту та розподілити бюджет на окремі елементи.
- 3) Розробка бюджету: на цьому етапі створюється бюджет проекту, який включає вартісні оцінки для окремих робіт. Бюджет проекту складається з двох основних частин: кошторису надходжень, який визначає очікувані

прибутки від проекту, і кошторису витрат, який включає вартість ресурсів та розподіл бюджету на різні види витрат, такі як операційний бюджет та управлінський резерв.

4) Управління вартістю: цей процес охоплює контроль та управління змінами в бюджеті проекту. Він включає в себе виявлення та аналіз змін у вартості проекту, прийняття рі

шень щодо внесення змін до бюджету та вжиття заходів для забезпечення виконання проекту в межах затвердженого бюджету.

Зазначені етапи допомагають ефективно управляти вартістю проекту, забезпечуючи оптимальне використання ресурсів та досягнення поставлених фінансових цілей. Результатом їх успішної реалізації є базовий план вартості, який стає частиною загального плану проекту та використовується як основа для моніторингу та контролю над вартістю проекту.

## **2. Управління ресурсами**

У проекті було залучено такі ресурси: люди: керівник проекту; веб-розробник, дизайнер, тестувальник;

Розглянемо завантаженість кожного з ресурсів на таблиці 3.1:

Табл. 3.1

Етап	Початок	Завершення	К і л ь к і с т ь витраченого часу
Аналіз предметної	01.01.2023	15.01.2023	15 дн
Маркетинговий аналіз	15.01.2023	01.02.2023	15 дн
Аналіз конкурентів	01.02.2023	15.02.2023	15 дн
Визначення ризиків	01.02.2023	15.02.2023	15 дн
Визначення витрат	15.02.2023	20.02.2023	4 дн
Пошук приміщення	20.02.2023	24.02.2023	4 дн
Підписання контракту	24.02.2023	25.02.2023	2 дн
Передача обов'язків	25.02.2023	26.02.2023	1 дн
Закупівля обладнання	26.02.2023	12.03.2023	14 дн
Алгоритм розроблено	12.03.2023	15.03.2023	3 дн
Визначення вимоги до постатчальників	16.03.2023	16.03.2023	1 дн
Розробка проектної документації	16.03.2023	20.03.2023	4 дн
Проект розроблено	20.03.2023	21.03.2023	1 дн
Розробка ПЗ	21.03.2023	30.02.2024	11 міс
Акт на розробку	21.03.2023	30.02.2024	11 міс
Створення проектного	01.03.2024	15.03.2024	15 днів
Тестування	01.03.2024	31.04.2024	60 днів
Презентація	20.05.2024	28.05.2024	8 днів
Розміщення сервісу на сервері	05.06.2024	05.08.2024	60 днів
Підписання АВР	10.08.2024	11.08.2025	1 дн

Управління термінами проекту включає ряд процесів, які спрямовані на забезпечення вчасного виконання плану проекту. Основними етапами управління термінами проекту є наступні:

1) Визначення складу робіт проекту: на цьому етапі визначаються конкретні операції, які потрібно виконати для досягнення запланованих результатів проекту. Це включає аналіз проекту, визначення завдань та розбиття їх на конкретні робочі пакети.

2) Визначення взаємозв'язків робіт: на цьому етапі встановлюються та документуються зв'язки між різними операціями проекту. Це допомагає встановити послідовність виконання робіт та зрозуміти, як одні завдання впливають на інші.

3) Оцінка тривалостей робіт: цей процес включає оцінку кількості робочих періодів, необхідних для виконання кожної окремої операції. Це може включати оцінку часу, необхідного для завершення завдання, та врахування резервного часу.

4) Складання розкладу проекту: на цьому етапі розробляється розклад виконання проекту з урахуванням взаємозв'язків робіт, їх тривалостей та потреби в ресурсах. Розклад проекту дозволяє визначити послідовність та тривалість робіт, розподілити ресурси та планувати виконання завдань.

5) Управління розкладом: цей процес охоплює контроль та управління змінами в розкладі проекту. Керівники проекту вживають заходів для уникнення порушень термінів проекту та реагують на непередбачені зміни, забезпечуючи вчасне виконання завдань.

Управління розкладом вимагає розумного розподілу ресурсів, таких як матеріальні, тимчасові, людські та фінансові, з метою уникнення затримок у виконанні проекту. До вхідних даних процесу входять контракт, розбиття

робіт на пакети (WBS) та опис проекту. Основним результатом цього процесу є тимчасовий план-графік проекту, який є базовим планом проекту та служить основою для моніторингу та контролю над термінами проекту..

Управління вартістю проекту включає низку процесів, спрямованих на забезпечення виконання та завершення проекту в рамках затвердженого бюджету. Основними етапами управління вартістю є наступні:

1) Планування ресурсів: на цьому етапі визначаються ресурси, такі як людські ресурси, обладнання та матеріали, а також їх кількість, необхідні для виконання робіт проекту. Це включає аналіз потреб проекту та розподіл ресурсів відповідно до завдань проекту.

2) Оцінка вартості: на цьому етапі проводиться оцінка вартості ресурсів, необхідних для виконання робіт проекту. Це включає оцінку фінансових витрат, пов'язаних з придбанням та використанням ресурсів проекту.

3) Розробка бюджету: на цьому етапі вартісні оцінки включаються до окремих робіт проекту, що дозволяє сформувати загальний бюджет проекту. Бюджет проекту включає в себе кошторис надходжень та кошторис витрат, які визначають вартість проекту.

4) Управління вартістю: на цьому етапі здійснюється контроль та управління змінами бюджету проекту. Керівники проекту вживають заходів для забезпечення фінансової стабільності проекту, виявлення та вирішення відхилень від бюджету.

Планування ресурсів та оцінка вартості виконуються на етапі ініціації проекту тимчасовою робочою групою. Вартісні оцінки можуть базуватися на експертних оцінках, аналогічних проектах або параметричних оцінках з врахуванням встановлених нормативів. Під час розробки бюджету проекту визначаються конкретні календарні терміни доходів та витрат, розраховуються фінансові резерви та визначаються джерела фінансування.

Бюджет проекту складається з двох складових: кошторису надходжень, який відображає очікувані доходи, та кошторисів витрат, які визначають вартість проекту. Кошторис витрат включає оперативний бюджет, що охоплює витрати

на ресурси, необхідні для виконання робіт проекту, управлінський резерв та бюджет непередбачуваних витрат.

Результатом розробки бюджету проекту є базовий план вартості, який представляється у формі таблиці або графіка. Затвердження базового плану вартості здійснює Генеральний директор компанії.

Управління вартістю проекту може базуватись на методі аналізу реалізованих обсягів. Для оцінки використовуються такі показники, як Плановий обсяг (ПО). Плановий обсяг представляє собою частину затвердженого бюджету проекту, яка була запланована для витрати протягом певного періоду.

Управління людськими ресурсами проекту включає ряд процесів, необхідних для найбільш ефективного використання людей, задіяних у проекті. Основними підпроцесами в управлінні людськими ресурсами є:

1) Організаційне планування: цей процес включає визначення, документування та призначення проектних ролей, встановлення відповідальності та відносин звітності в рамках проекту.

2) Призначення персоналу: на цьому етапі залучаються необхідні людські ресурси та призначаються на відповідні ролі для роботи над проектом.

3) Розвиток команди проекту: цей процес передбачає розвиток індивідуальних та групових компетенцій з метою поліпшення продуктивності і результативності проекту.

### **3. Управління ризиками**

У такому випадку проектному менеджеру доступні кілька варіантів розв'язання даної проблеми:

1) Перенести дедлайн виконання задачі, якщо це можливо.

2) Переконатися, що затримка не впливає на конфлікти ресурсів у майбутньому.

3) В разі об'ємної задачі розглянути можливість залучення додаткового спеціаліста.

4) Для виконання нескладних задач можна розглянути варіанти співпраці з фрілансерами або домовитися про овертайми для поточного спеціаліста.

5) Якщо можливо, проектний менеджер може попросити співробітника відпрацювати додаткові години або вихідні, щоб завершити задачу вчасно.

Управління інформаційною взаємодією в проекті включає ряд процесів, необхідних для своєчасної та якісної підготовки, збору, розподілу, зберігання та використання проектної інформації. Основними процесами в управлінні інформаційною взаємодією є:

1) Планування взаємодії: визначення потреб учасників проекту в інформації та взаємодії, включаючи кому і яка інформація потрібна, коли вона буде необхідна і яким чином надаватиметься.

2) Розподіл проектної інформації: своєчасне забезпечення учасників проекту необхідною інформацією.

3) Звітність щодо виконання: збір та поширення інформації щодо виконання плану проекту, включаючи звіти про поточний стан робіт та прогнозування.

4) Адміністративне завершення: підготовка, збір, організація зберігання та розповсюдження інформації з метою формалізації завершення етапу, фази або проекту в цілому.

Вхідними елементами бізнес-процесу управління інформаційною взаємодією в проекті є: технології інформаційної взаємодії в компанії, склад учасників проекту та план проекту. Основним документом, що розробляється у процесі, і регламентує процеси інформаційної взаємодії, є "План управління інформаційною взаємодією в проекті".

План управління інформаційною взаємодією в проекті повинен містити:

1) Опис інформації, яка підлягає поширенню, включаючи формат, зміст, рівень деталізації та угоду щодо термінології.

2) Структуру розповсюдження, яка конкретизує, кому потрібно передати відповідну інформацію (звіти про стан, звіти про виконання робіт та етапів, плани, технічна документація тощо) та якими методами буде здійснюватися поширення інформації різного типу.

3) Графік, що відображає часові рамки для кожного типу інформаційної взаємодії.

4) Методи доступу до інформації у проміжках між запланованими інформаційними взаємодіями.

#### **4. Управління якістю**

Управління якістю є невід'ємною складовою управління проектами, яка заслуговує на особливу увагу. Забезпечення якості є ключовою темою, яка повинна бути присутня протягом всього проекту. Це стосується як процесів планування і реалізації, так і моніторингу та контролю виконаної роботи.

Свідомість керівників проектів та учасників проекту щодо важливості якості зазвичай висока. Ніхто не заперечує значення якості продукту проекту. Від цього залежить задоволення замовника і досягнення очікуваних користей. Проте, на практиці управління якістю часто приділяється менше часу та уваги, порівняно з плануванням, моніторингом прогресу чи навіть управлінням ризиками. Це може бути пов'язано з відсутністю впроваджених передових практик, які дозволяють застосовувати методичний підхід до управління якістю з раціональним обсягом робіт та витрат.

В провідних стандартах управління якістю в проектах цей процес описується як планування, забезпечення та контроль якості. Але що означає це на практиці? Для того, щоб управління якістю приносило очікувані результати і було економічно ефективним, воно має бути чітко визначеним, вбудованим у виробничий процес і підтримуватися правильно підібраними

IT-інструментами. Особлива важливість цього питання проявляється у великих проектах, де традиційні "ручні" методи управління якістю стають недоцільними і неефективними.

На етапі створення та тестування бізнес-ідеї завжди важливо зосередитися на оптимізації витрат і грошового потоку, а не лише на очікуваному прибутку. Те, що може здатися перспективним на перший погляд, може не мати реального втілення, а кошти, витрачені безрезультатно, стануть втраченими. При виході на ринок з нашою ідеєю, ми володіємо брендом, обіцяємо клієнтам якість і цінність. Тому ми не можемо дозволити собі низькі показники якості, оскільки це призведе до розчарування клієнтів. Порушення обіцянки бренду - це щось, що клієнти не прощають. Тому на цьому етапі потрібно радикально змінити підхід до забезпечення якості додатку, і витрати на його розробку та підтримку значно зростають. В цьому контексті важливо розуміти, що доставка та обслуговування окремих модулів системи стає все дорожчою.

Загалом, варто зазначити, що ми можемо починати з меншої якості, коли маємо інноваційний проект і проектний менеджер не має повної впевненості в його успіху. Але, наприклад, коли ми автоматизуємо вже існуючий процес або замінюємо застарілу та неефективну систему на нову, ми повинні негайно забезпечити високу якість, оскільки тут немає часу на помилки.

Одним з ключових елементів контролю є проектний менеджер. Він об'єднує роботу команди в одне ціле, забезпечує просування проекту вперед і встановлює правильний темп. Ключова роль проектного менеджера полягає у розробці чітких характеристик продукції проекту, включаючи відповідальність за якість, критерії якості, спосіб досягнення якості і вимоги замовників та одержувачів щодо рівня якості. Проект повинен бути чітко визначений з самого початку. Специфікації включають початкові запити пропозицій, рекомендації, запити і зміни. Усі вищезазначені дії спрямовані на підготовку початкової проектною пропозиції. Приклади критеріїв включають

цільові терміни, основні функції, форму, продуктивність, терміни, простоту використання, вартість експлуатації та виробництва, доступність, точність, технологічність та безпеку.

Умови прийняття повинні бути домовлені між постачальником і клієнтом під час підготовки проекту, включаючи посилання на інші стандарти, що необхідно застосовувати під час реалізації проекту, послідовність управління дизайном, процедури управління змінами, графік управління конфігурацією та інструменти управління якістю.

Якість проекту визначається тим, наскільки створений продукт задовольняє потреби користувача. Один зі способів визначення цього полягає у здійсненні опитування клієнта до створення продукту та після його впровадження.

В сферу відповідальності за якісне планування проекту входить моніторинг надання відповідних рангів будь-якому аналізу якості, пропонування можливих тестувальників для окремих перевірок якості, перевірка знань осіб, зайнятих у перевірці якості, щодо їх обов'язків та належної підготовки, контроль якості випробувачів, тобто перевірка правильності їх відбору та отримання інформації про виконану роботу, контроль і моніторинг всіх коригувальних заходів, документування та звіт про організаційні правила та рекомендації щодо покращення, а при необхідності виконання функцій тестера якості.

Щоб зробити текст більш унікальним, можна додати деякі ідеї та розділи, наприклад:

- 
- - Значення ефективного контролю якості та впровадження стандартів у проекті для забезпечення високої якості продукту.

- - Важливість залучення експертів та
- 
- фахівців до процесу контролю якості та тестування продукту.
- - Ролі та відповідальність різних сторін (постачальників, клієнтів, управління проектом) у забезпеченні якості проекту.
- - Значення постійного вдосконалення та оновлення процесів контролю якості для відповідності змінюються потребам та вимогам ринку.
- Такі додаткові ідеї та розділи можуть допомогти зробити текст більш унікальним і додати нові аспекти до розглянутої теми.

## РОЗДІЛ 4. ВИБІР ПРОГРАМНИХ ЗАСОБІВ ТА РОЗРОБКА ВЕБ СЕРВІСУ

### 1. Проектування архітектури системи

Для Під час проектування мною була вибрана MVC архітектура системи. Принцип MVC (Model View Controller) - архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення. Цей шаблон передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд (інтерфейс користувача) та модуль керування.

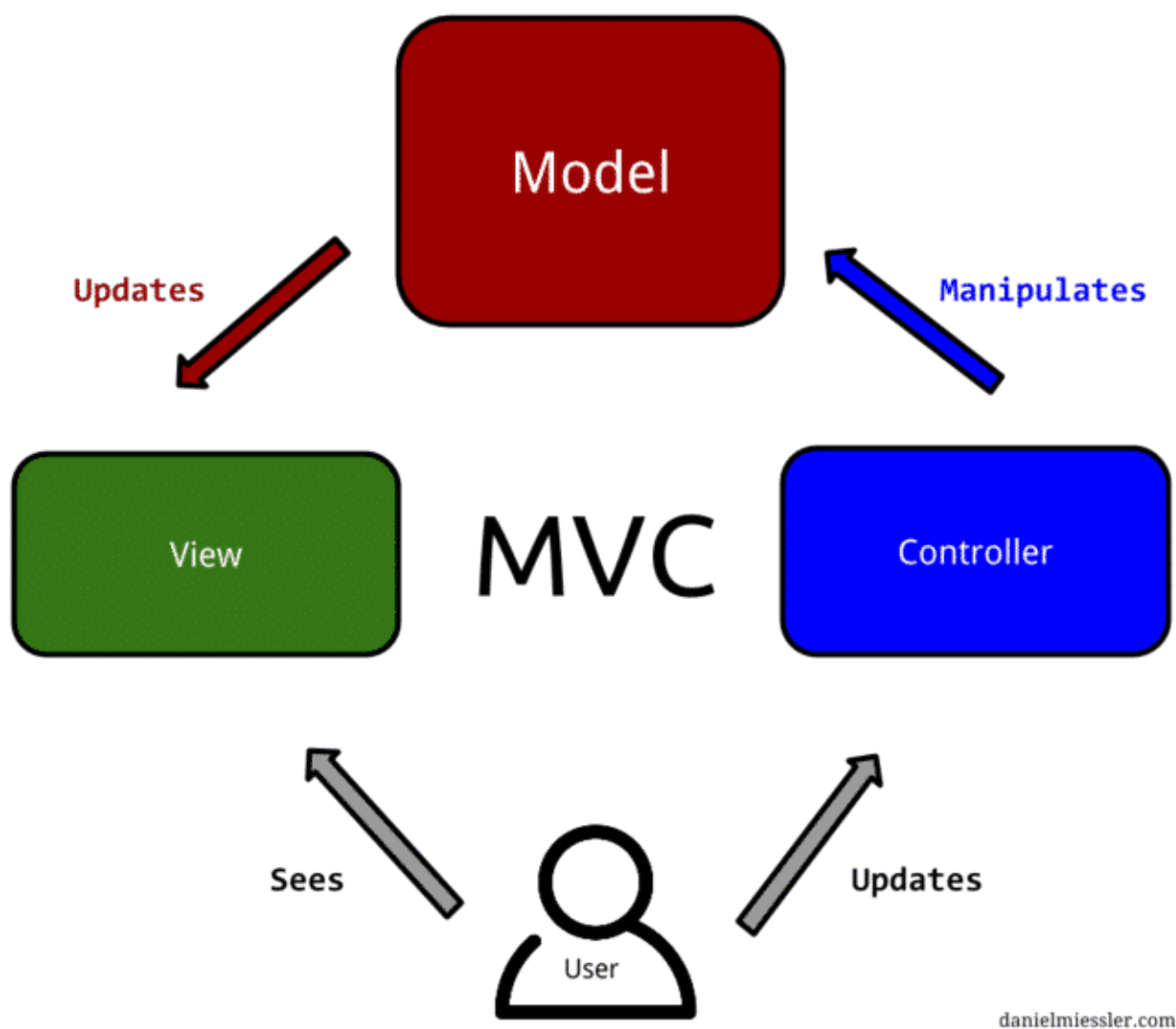


Рисунок 4.1 - Шаблон проектування MVC

В основі лежать:

1. Model - набір низькорівневих функцій, для роботи з БД.

2. View - файли зовнішнього вигляду. По суті це набір HTML файлів.
3. Controller - це сполучна ланка між моделями і видами. Він приймає запит, обробляє його і віддає користувачу згенеровану сторінку.

css	20.11.2019 1...	Папка файлів
img	23.11.2019 0...	Папка файлів
js	29.11.2019 2...	Папка файлів
Model	20.11.2019 1...	Папка файлів
Parse	20.11.2019 1...	Папка файлів
vendor	22.11.2019 2...	Папка файлів
View	10.12.2019 1...	Папка файлів
composer.json	11.11.2019 1...	Файл JSON
Controller.php	11.12.2019 0...	Файл PHP
index.php	24.11.2019 1...	Файл PHP
pick.model	10.12.2019 2...	Файл MODEL
user.model	10.12.2019 2...	Файл MODEL

Рисунок 4.2 – Вигляд файлової структури

Шаблон застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача.

Одними з основних елементів архітектури системи є:

- структура бази даних;
- взаємодія клієнта і сервера;
- спосіб зберігання моделей / даних / результатів;
- спосіб здійснення аналізу і обробки результатів.

Взаємодія клієнта і сервера побудована на використанні `mysql` запитів для передачі даних. Необхідно було узгодити які поля відповідають за кожен тип переданих значень. У підсумку вийшов зрозумілий і надійний інтерфейс програми, що дозволяє користувачеві гарантовано отримати необхідний результат.

На розробленому веб-ресурсі присутній функціонал навчання нейронної мережі на вхідних даних. Я використав формат JSON — це

текстовий формат обміну даними між комп'ютерами. JSON базується на тексті, може бути прочитаним людиною. Формат дає змогу описувати об'єкти та інші структури даних. Цей формат використовується переважно для передачі структурованої інформації через мережу (завдяки процесу, що називають серіалізацією).

Також необхідно було спроектувати модель зберігання навчених моделей, враховуючи вимогу зниження навантаження на обчислювальні ресурси сервера. Для цього було запропоновано підхід використання відбитка моделі, що представляє собою хеш-суму від даних і текстового коду алгоритму, завдяки якому не відбувається повторного навчання вже раніше навченого алгоритму.

## **2. Моделювання поведінки системи**

Діаграма варіантів використання є інструментом, що описує функціональні можливості системи та її призначення. Її розробка виконується з метою досягнення таких цілей:

1. Розробка початкової концептуальної моделі системи, яка в подальшому буде деталізована у логічні та фізичні моделі.
2. Визначення загальних вимог до функціональної поведінки проєктованої системи.
3. Встановлення загальних меж та контексту моделювання предметної області на початкових етапах проєктування системи.
4. Підготовка вихідної документації для забезпечення взаємодії розробників системи з її замовниками та користувачами.

Діаграма варіантів використання включає користувача системи та інші компоненти, такі як програмне забезпечення, сервери даних, база даних та аналітична система. На діаграмі зображені взаємодія та обмін інформацією між цими компонентами. Користувач, виступаючи як актор, виконує певні

операції та взаємодіє з системою. Він може ставити запити до системи, які обробляються, а також здійснювати операції, такі як задання початкових параметрів, запуск процесів обробки та аналізу даних.

З метою зробити текст більш унікальним та додати додаткові деталі, можна розширити його такими ідеями:

- Пояснення того, як саме взаємодіють компоненти системи та які процеси відбуваються в ході виконання операцій користувачем.

- Зазначення можливих типів операцій, які можуть бути виконані користувачем, таких як створення, редагування, видалення об'єктів або отримання звітів та результатів аналізу даних.

- Вказівка на важливість взаємодії між користувачем та системою для досягнення бажаних результатів та задоволення потреб користувача.

- Опис ролі серверів даних, бази даних та аналітичної системи у виконанні операцій та забезпеченні коректної обробки та аналізу даних.

Ці додаткові деталі та ідеї допоможуть розширити текст та зробити його більш унікальним.

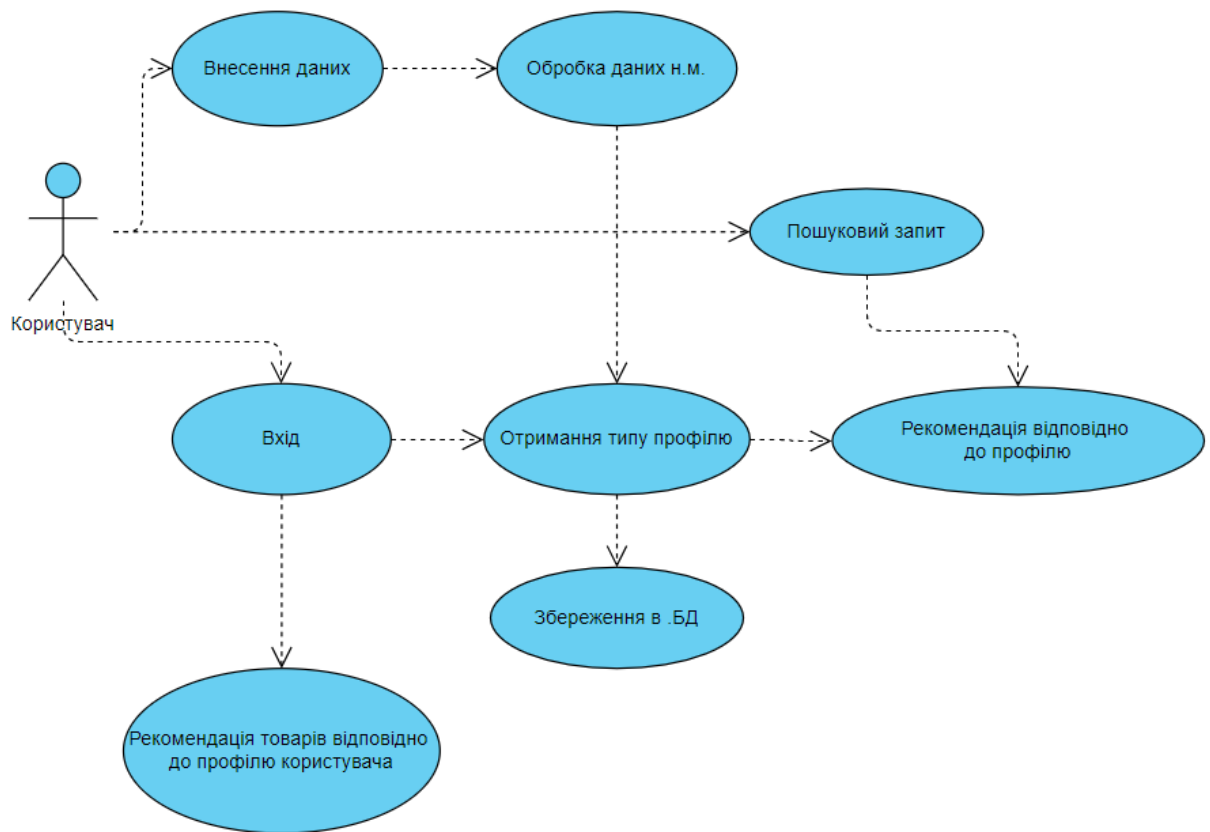


Рисунок 4.3 - Use case діаграма

Контекстна діаграма розробки системи показана на рисунку 4.4.

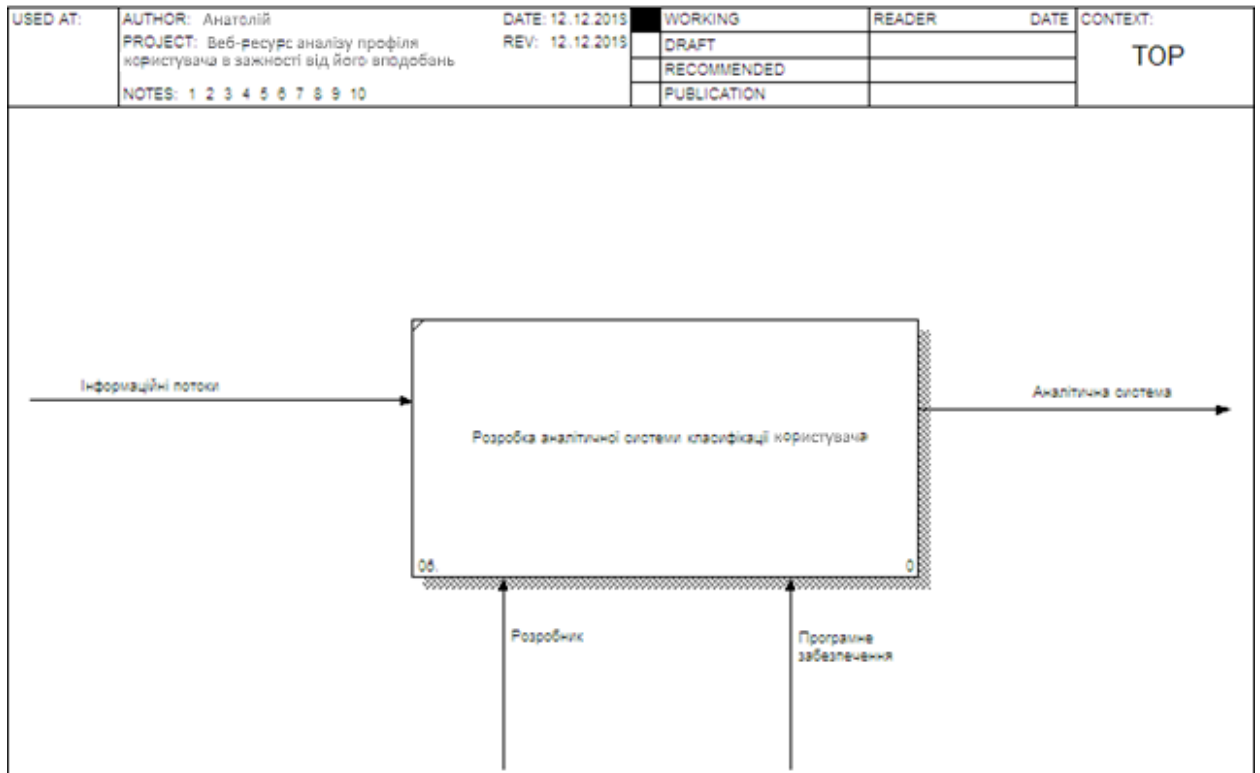


Рисунок 4.4 - Контекстна діаграма розробки системи

Таблиця 4.1 - Опис основних елементів контекстної діаграми

Назва стрілки	Опис	Тип
Інформаційні потоки	Вхідні дані	Input
Аналітична система оцінки	Результат розробки	Output
Розробник	Проектувальник системи	Mechanism
Програмне забезпечення	Середовища розробки програмного	Mechanism

Декомпозиція є науковим методом, що базується на розбитті складних завдань на серію менших і взаємопов'язаних підзадач. Цей метод дозволяє розглядати будь-яку систему як складну структуру, що складається з окремих

підсистем, які, у свою чергу, також можуть бути розділені на частини. Декомпозиція є ефективним способом розбиття великого завдання на більш прості елементи, які легше аналізувати та вирішувати.

Цей метод застосовується не тільки до матеріальних об'єктів, але й до процесів, явищ і понять. Він дозволяє зберегти структуру та взаємозв'язки між елементами системи, що сприяє більш ефективному розумінню та аналізу цієї системи. Крім того, декомпозиція дозволяє розподілити велике завдання між різними командами або особами, що сприяє більш ефективному виконанню проекту та досягненню кращих результатів.

Використання методу декомпозиції дозволяє спростити складні завдання, зробити їх більш керованими та досяжними, а також забезпечити кращу організацію роботи над ними. Цей метод є важливим інструментом у наукових, інженерних, технологічних та багатьох інших галузях, де потрібно ефективно розбити складність завдань та забезпечити кращу у п р а в л і н с ь к у с т р у к т у р у .

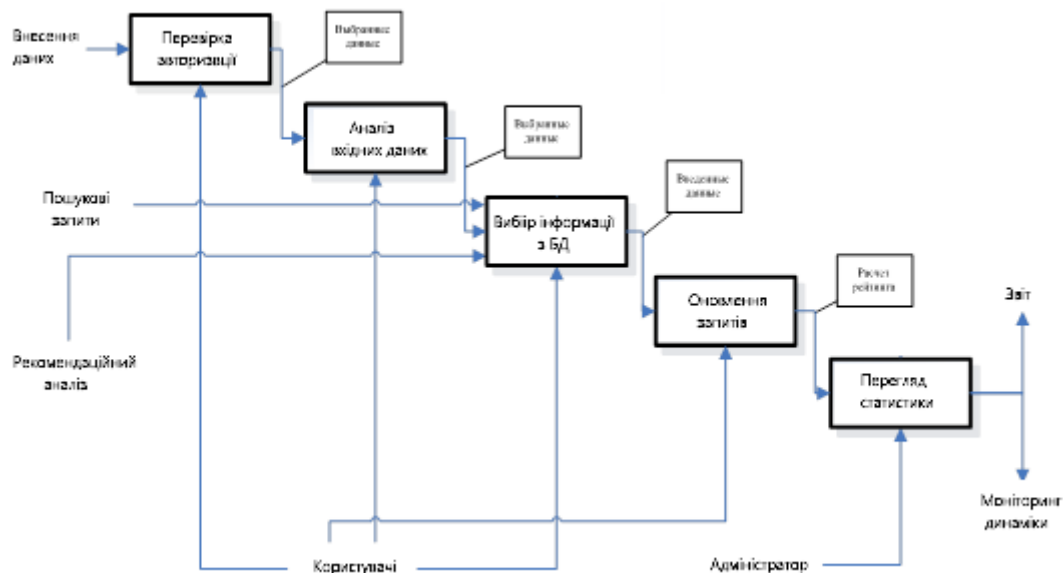


Рисунок 4.5 - Декомпозиція процесу розробки системи

Діаграма дерева вузлів відображає ієрархію робіт у моделі та дає змогу оглянути модель в цілому, але не надає інформацію про зв'язки між роботами.

Процес створення моделі робіт є ітеративним, що означає, що роботи можуть бути переміщені у дереві вузлів багато разів. Щоб уникнути плутанини та перевірити ефективність декомпозиції, необхідно створювати діаграму дерева вузлів після кожної зміни.

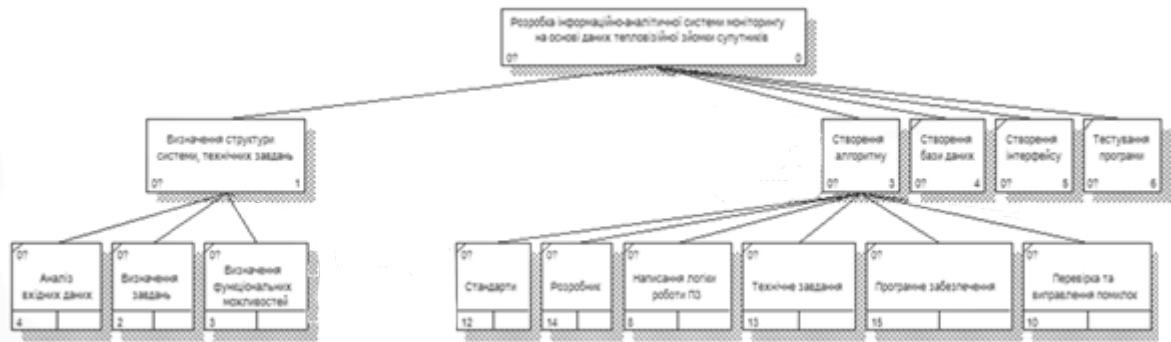


Рисунок 4.6 - Діаграма дерева вузлів

Для проведення аналізу профілю користувача було використано класифікатор MLPClassifier. Основний принцип його функціонування можна проілюструвати таким чином:

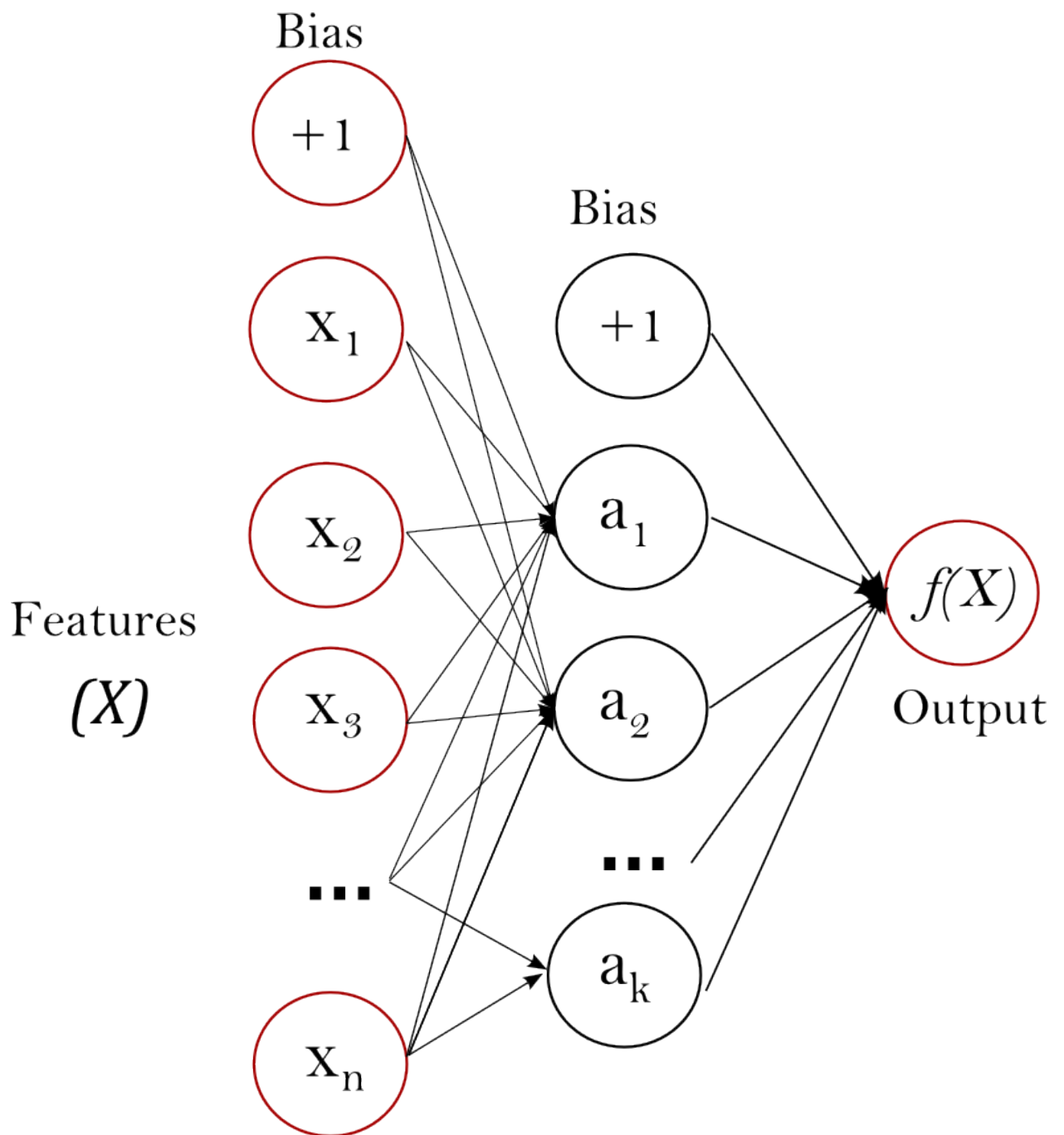


Рисунок 4.7 – Принцип роботи нейронів MLPClassifier

Малюнок демонструє типову структуру мережі з прямим передаванням сигналу. Нейрони організовані у шари з регулярним способом. Перший шар, який є вхідним, просто вводить значення вхідних змінних. Кожен нейрон з прихованого та вихідного шарів з'єднаний з усіма нейронами попереднього шару. Існують мережі, де нейрони мають зв'язки лише з обмеженою кількістю нейронів попереднього шару, але для більшості застосувань мережі

з повними зв'язками є більш ефективними. У моєму проекті було використано саме такий тип мережі з повною системою зв'язків.

### **3. Frontend-розробка**

Frontend-розробка фокусується на веб-стороні, яка спрямована на користувача. Розробники інтерфейсу забезпечують зручну взаємодію відвідувачів з веб-сайтами за допомогою програмування, дизайну та інших інструментів. Вони створюють спадні меню, макети та дизайн веб-сайтів.

Під час розробки UI/UX дизайну дотримуються основних принципів:

- 1. Ясність: Інтерфейс повинен бути зрозумілим завдяки мові, потоку, ієрархії та використанню зорових елементів, щоб уникнути недорозуміння.
- 2. Виразність: Надмірне уточнення та позначення елементів може роздути інтерфейс, що ускладнює його використання. Необхідно знайти баланс між прозорістю та зрозумілістю.
- 3. Знайомство: Важливо, щоб нові користувачі знайшли деякі елементи знайомими. Використання прикладів з реального життя може полегшити їх сприйняття.
- 4. Відповідність: Інтерфейс повинен надати користувачам належний зворотний зв'язок із тим, що відбувається та як обробляються їх ввідні дані.
- 5. Послідовність: Інтерфейс повинен узгоджуватися з програмою, щоб користувачі могли розпізнавати шаблони використання.
- 6. Естетика: Привабливий дизайн необов'язковий, але може зробити

використання програми більш приємним і задоволеним користувачам.

- 7. Продуктивність: Класний інтерфейс допомагає користувачеві бути ефективним за допомогою ярликів та зручного дизайну, що економить час.

Узагальнено, UX дизайн передбачає всебічний підхід до взаємодії користувача з інтерфейсом, який враховує багато аспектів, включаючи середовище користувача, тип пристрою та способи введення та відображення інформації.

Під час розробки інтерфейсу я надавав особливу увагу деталям, створюючи зручний і привабливий користувальницький інтерфейс, що оптимізує взаємодію з користувачем.

Інтерфейсні розробники використовують різні мови програмування, такі як JavaScript, HTML і CSS, для розробки веб-сайтів. Кожна мова має свою функціональність: HTML відповідає за структуру і вміст веб-сторінок, CSS забезпечує дизайн і стилізацію, а JavaScript додає інтерактивні можливості.

HTML є мовою, призначеною для створення структури веб-сторінок, що можуть бути переглянуті користувачами в Інтернеті. Вона дозволяє визначати заголовки, абзаци та вбудовувати медіафайли, такі як зображення та відео. HTML використовується за допомогою тегів, які визначають різні елементи, і результат зберігається як HTML-файл, що відображається у браузері.

CSS є мовою таблиць стилів, яка визначає зовнішній вигляд елементів HTML на веб-сторінці. Вона забезпечує варіанти дизайну, макету та адаптації до різних пристроїв з різними розмірами екранів. CSS дозволяє змінювати вигляд багатьох веб-сторінок одночасно шляхом взаємодії з елементами HTML.

Загалом, розробка інтерфейсу вимагає розуміння і використання різних мов інтернет-технологій для створення зручного, привабливого та функціонального веб-сайту.

Для встановлення зв'язку між HTML і CSS використовуються селектори. Селектори є частиною CSS-коду і визначають, на які елементи HTML будуть застосовуватися стилі CSS.

В оголошеннях ми визначаємо властивості та їх значення, які застосовуються до обраних селекторів. Властивості визначають розмір шрифту, колір, відступи та інші характеристики, а значення встановлюють конкретні налаштування цих властивостей.

JavaScript (JS) є однією з найпопулярніших мов для написання скриптів. Вона використовується для розробки динамічних веб-сторінок і надає широкий спектр технологій для зовнішньої та внутрішньої розробки.

JS покращує загальну інтерактивність веб-сайту, дозволяючи користувачам взаємодіяти з веб-сторінками. Він додає динамічну поведінку та спеціальні ефекти до веб-сторінки. JavaScript має потужний функціонал і може виконувати різноманітні завдання для веб-сайту.

JavaScript дозволяє створювати багатофункціональні компоненти інтерфейсу, такі як повзунки зображень, спливаючі вікна, мегаменю навігації, перевірки форм, вкладки, гармошки та багато іншого. Він також дозволяє виконувати складні операції, наприклад, реагувати на дії користувача і оновлювати веб-сторінку без перезавантаження.

Завдяки JavaScript веб-сторінки можуть мати додатковий функціонал і реагувати на дії користувачів, забезпечуючи динаміку і інтерактивність без необхідності перезавантаження сторінки.

#### **4. Backend-розробка**

Back-end розробники відповідають за серверну частину веб-сайтів і зосереджуються на технічних аспектах, що відбуваються у закулісся. Вони володіють спеціальними навичками, щоб створювати структуру веб-сайту і забезпечувати його загальну функціональність, яка використовується передньою частиною сайту.

Back-end розробники працюють з мовами програмування, базами даних і серверами, щоб забезпечити правильну роботу веб-сайту. Вони розробляють логіку програми, обробляють запити користувачів, зберігають і отримують дані з бази даних, забезпечують безпеку і захист інформації, а також виконують багато інших завдань, що допомагають забезпечити правильне функціонування веб-сайту.

Задачі back-end розробника включають створення і підтримку API (Application Programming Interface), яке дозволяє комунікацію між різними компонентами веб-сайту, розробку імплементації баз даних, що зберігають важливу інформацію для веб-сайту, роботу з серверами і налаштування хостингу, розробку інфраструктури для масштабування і оптимізації веб-сайту.

У своїй роботі back-end розробники часто використовують мови програмування, такі як Python, Java, Ruby, PHP і Node.js, а також фреймворки і інструменти для забезпечення ефективної розробки і підтримки веб-сайтів.

Back-end розробники грають важливу роль у створенні стабільних, безпечних і функціональних веб-сайтів. Вони співпрацюють з front-end розробниками і дизайнерами, щоб створити комплексний і високоякісний продукт для користувачів.

На головній сторінці відразу пропонується авторизуватися або зареєструватися і внести відповідні дані про себе для можливості аналізу і класифікації користувача.

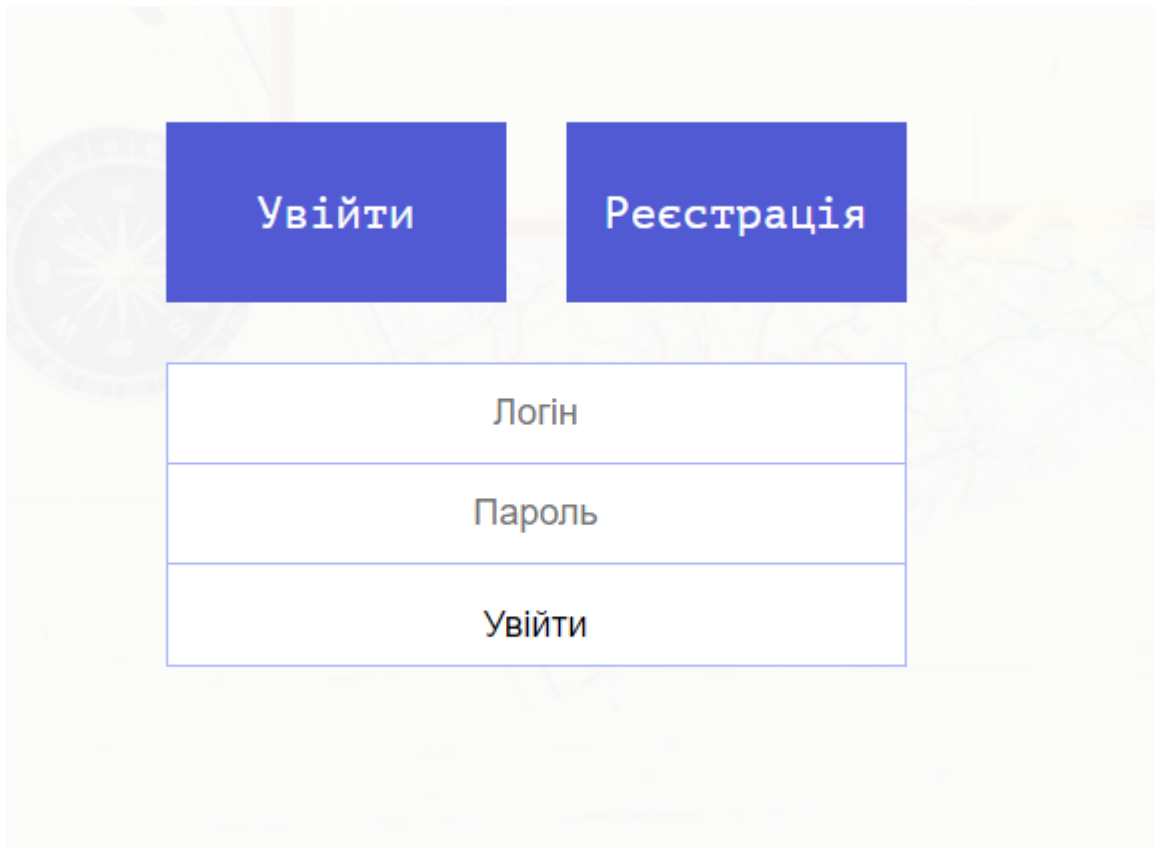


Рисунок 4.8. Головна сторінка

Для визначення типу користувача на основі машинного навчання потрібно передати певний набір даних які характеризують вподобання користувача. Також за необхідністю він може змінити ці дані.

<b>Увійти</b>	<b>Реєстрація</b>
<i>Стать</i> <input type="radio"/> Чоловіча <input type="radio"/> Жіноча	
<i>Сімейний статус</i> <input type="radio"/> Одружений/Заміжня <input type="radio"/> Ні	
<i>Наявність дітей</i> <input type="radio"/> Нема <input type="radio"/> 1-2 <input type="radio"/> 2+	
<i>Вік</i> <input type="radio"/> до 20 <input type="radio"/> 20-40 <input type="radio"/> 40+	
<i>Дохід (в тисячах)</i> <input type="radio"/> до 10 <input type="radio"/> 10-20 <input type="radio"/> 20+	
<i>Часто подорожуєте?</i> <input type="radio"/> Часто <input type="radio"/> Середне <input type="radio"/> Рідко	
Логін	
Пароль	
Реєстрація	

Рисунок 4.9. Форма для внесення користувацьких даних.

В адміністративній частині сайту знаходиться простий але зручний інтерфейс для статистики аналізу у вигляді графіків.

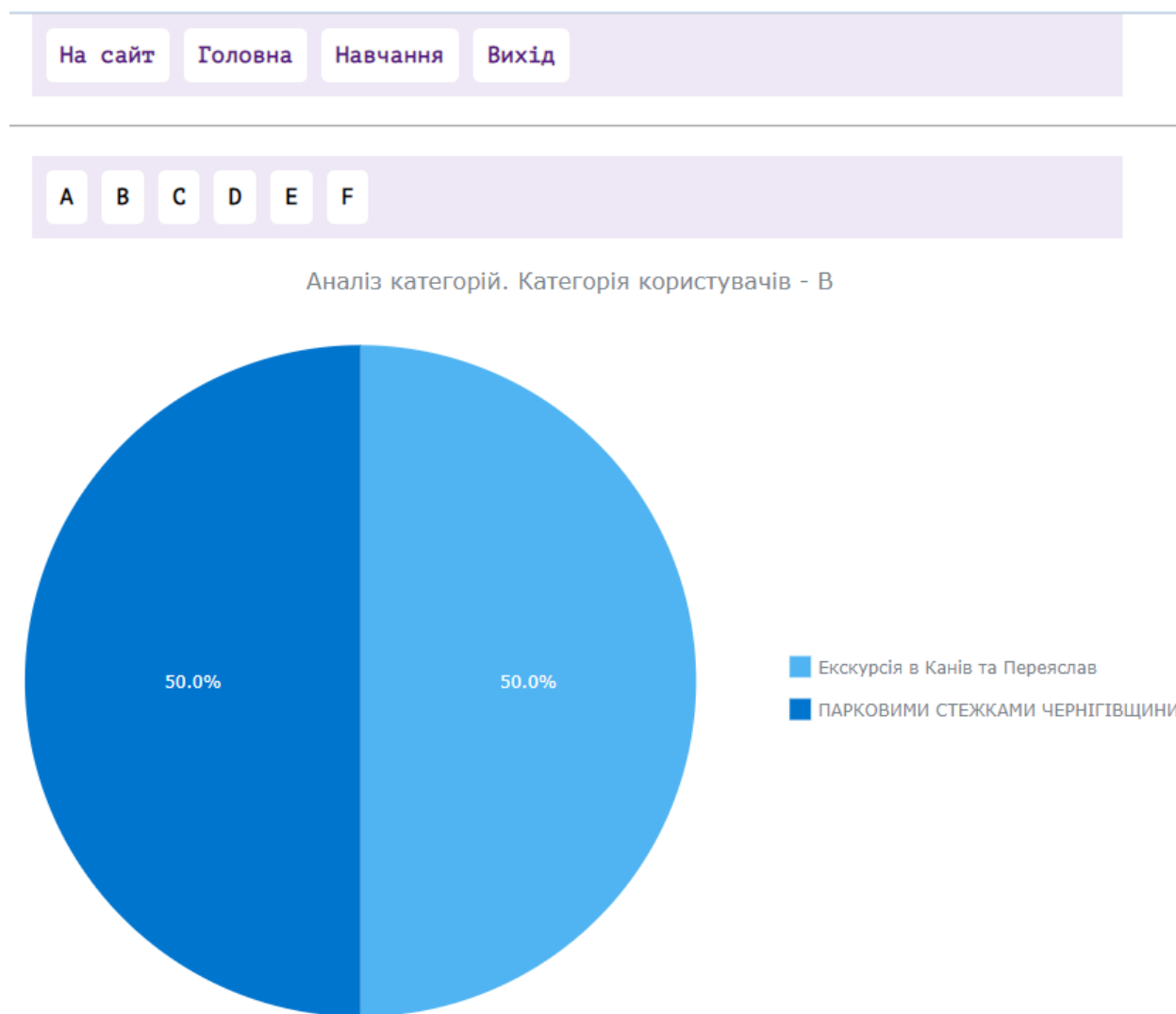


Рисунок 4.10. Статистика запитів по типам профілів користувачів

Також на сайті реалізована можливість навчання нейронної мережі по переданих користувачем даних у форматі json.

Фрагмент лістингу навчання нейронної мережі (Controller.php)

```
public function teachPickModel(){  
  
    $data = json_decode(trim($_POST['data']));
```

```

        $modelManager = new ModelManager();

        $mlp = new MLPClassifier(6, [2], ['i40', 'i20', 'so20', 'so10',
'sa10', 'sa5',]);

        $mlp->train($data->samples, $data->targets);

        $modelManager->saveToFile($mlp, $this->filepathPhone);

    }

    public function teachUserModel(){

        $data = json_decode(trim($_POST['data']));

        $modelManager = new ModelManager();

        $mlp = new MLPClassifier(6, [2], ['a', 'b', 'c', 'd', 'e',
'f',]);

        $mlp->train($data->samples, $data->targets);

        $modelManager->saveToFile($mlp, $this->filepathUser);

    }

```

## Аналіз даних на основі машинного навчання

### 5. Взаємодія з користувачем

На початку взаємодії користувача з сайтом йому пропонується надати додаткову інформацію про бажану подорож і відповідно до аналізу його категорії він отримує рекомендацію.

Дані для аналізу:

- Стать
- Вік
- Дохід
- Освіта
- Ціль
- Термін відпочинку

Заповніть форму для рекомендацій що до подорожі.

<p><i>Стать</i></p> <p><input checked="" type="radio"/> Чоловіча <input type="radio"/> Жіноча</p>
<p><i>Вік</i></p> <p><input type="radio"/> до 20 <input checked="" type="radio"/> 20-40 <input type="radio"/> 40+</p>
<p><i>Дохід (в тисячах)</i></p> <p><input type="radio"/> до 8 <input checked="" type="radio"/> 8-15 <input type="radio"/> 15+</p>
<p><i>Освіта</i></p> <p><input type="radio"/> Середня <input checked="" type="radio"/> Вища</p>
<p><i>Головне в подорожі</i></p> <p><input type="radio"/> Відпочинок <input type="radio"/> Лікування <input type="radio"/> Сімейний</p>
<p><i>Термін відпочинку</i></p> <p><input type="radio"/> 1-3 дні <input type="radio"/> 3-10 днів <input type="radio"/> 10+ днів</p>
<p>Підтвердити</p>

Рисунок 4.11. Головна сторінка

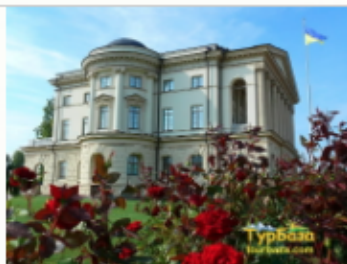
На основі цих даних відповідно до кластерного аналізу схожих категорій людей нейронна мережа на основі машинного навчання визначає до якого типу відноситься щойно зареєстрований користувач і рекомендує йому найпопулярніші варіанти відпочинку відповідно до його кластеру цікавості.

Ви можете допомогти в покращенні рекомендацій Показати



### ПАРКОВИМИ СТЕЖКАМИ ЧЕРНІГІВЩИНИ

Якщо захоплюєтесь архітектурою чи історією, або просто любите прогулятися гарним парком - ця екскурсія вихідного дня саме для Вас. Людина назавжди пов'язана з природою невидимим зв'язком, тому не дивно, що нас завжди притягують парки, ліси, сквери. Туроператор ТурБаза запрошує на прогулянку парковими стежками Чернігівщини, оцінити могутність та велич Української землі починаючи від сивокопаних гір та



### Гетьманська столиця Батурин

Якщо Вас цікавить героїчне минуле України, козацтва та інших періодів, саме для Вас ця захоплива історична мандрівка "Гетьманська столиця Батурин та Козелець". Церква Різдва Богородиці в Козельці та історична Гетьманська столиця - Батурин - це два дуже цікавих історичних місця в Україні. Одноденна екскурсія до цих місць може бути цікавим досвідом для відвідувачів, які цікавляться історією України



### Екскурсія в Канів та Переяслав

Канів і Переяслав - два міста, дві історії ... У Каневі вас чекають найцікавіші та унікальні історії, пов'язані з українським патріархом Т.Г. Шевченко. Переяслав - місто, яке було третім за величиною в Київській Русі, а сьогодні - один з найпопулярніших туристичних центрів Київської області.

**Обрати**

Рисунок 4.12. Покращення рекомендацій

Якщо пропозиція не відповідає інтересам користувача він може покращити методи навчання і надати додаткову інформацію що до своїх вподобань відповідно до своєї класифікації як користувача. Надалі для аналізу рекомендації будуть враховуватися ці дані.

Дані про відповіді зберігаються в базу даних при кожному запиті кожного користувача і сортуються відповідного до його приналежності певній категорії користувачів, тому користувачу відображаються завжди рекомендації відповідно до останніх побажань клієнтів, і він сам може впливати на рекомендації системи.

Також користувач може змінити налаштування свого профілю відповідно до зміни його життєвих обставин що надасть можливість проаналізувати його профіль знову і класифікатор може змінити отримані дані про користувача відповідно до вхідної інформації.

Редагувати дані

<p><i>Стать</i></p> <p><input checked="" type="radio"/> Чоловіча <input type="radio"/> Жіноча</p>
<p><i>Освіта</i></p> <p><input type="radio"/> Середня <input checked="" type="radio"/> Вища</p>
<p><i>Наявність дітей</i></p> <p><input type="radio"/> Нема <input checked="" type="radio"/> 1-2 <input type="radio"/> 2+</p>
<p><i>Вік</i></p> <p><input type="radio"/> до 20 <input checked="" type="radio"/> 20-40 <input type="radio"/> 40+</p>
<p><i>Дохід (в тисячах)</i></p> <p><input type="radio"/> до 8 <input checked="" type="radio"/> 8-15 <input type="radio"/> 15+</p>
<p><i>Часто здійснюєте покупки в інтернеті?</i></p> <p><input checked="" type="radio"/> Часто <input type="radio"/> Середне <input type="radio"/> Рідко</p>
<p>Змінити дані</p>

Рисунок 4.13. Зміна даних профілю

Після введення даних користувач отримує інформацію про можливі цікаві подорожі для нього на основі аналізу його профілю.



 <p><b>ПАРКОВИМИ СТЕЖКАМИ ЧЕРНІГІВЩИНИ</b></p> <p>Якщо захоплюєтесь архітектурою чи історією, або просто любите прогулятися гарним парком - ця екскурсія вихідного дня саме для Вас. Людина назавжди пов'язана з природою, невидимим зв'язком, тому не дивно, що нас завжди притягують парки, ліси, сквери. Туроператор ТурБаза запрошує на прогулянку парковими стежками Чернігівщини, оцінити могутність та велич Української землі починаючи від вічнозелених лісів та чагарників до парково-архітектурних ансамблів, що збереглися в повному обсязі до наших часів. Приєднуйтеся та переконайтеся!</p> <p><b>Обрати</b></p>	 <p><b>Гетьманська столиця Батурін</b></p> <p>Церква Різдва Богородиці в Козельці та історична Гетьманська столиця - Батурін - це два дуже цікавих історичних місця в Україні. Одноденна екскурсія до цих місць може бути цікавим досвідом для відвідувачів, які цікавляться історією України та культурними пам'ятками. Під час екскурсії до церкви Різдва Богородиці в Козельці та історичної Гетьманської столиці - Батуріну ви зможете насолодитися прекрасними пам'ятками історії та культури України, дізнатися багато цікавого про традиції та звичай наших предків та насолодитися красою цих місць.</p> <p><b>Обрати</b></p>	 <p><b>Екскурсія в Канів та Переяслав</b></p> <p>Канів і Переяслав - два міста, дві історії... У Каневі вас чекають найцікавіші та унікальні історії, пов'язані з українським патріархом Т.Г. Шевченком. Переяслав - місто, яке було третім за величиною в Київській Русі, а сьогодні - один з найпопулярніших туристичних центрів Київської області.</p> <p><b>Обрати</b></p>
 <p><b>ЕКСКУРСІЯ В ЗАМОК РАДОМИСЛЬ</b></p> <p>«Замок-музей Радомисль» - це архітектурна та історико-культурна перлина поліського краю. У 1612 році на цьому місці було збудовано «папірню» - укріплену фабрику з виробництва паперу для друкарні Києво-Печерської Лаври. Сьогодні «Замок Радомисль» приймає гостей та туристів, в ньому проводяться індивідуальні та колективні екскурсії, урочисті церемонії, мистецькі та культурні заходи, майстер класи, пленери, фестивалі та концерти. Замок отримав нове дихання, нове життя і дарує свою велич і енергію тисячоліття всім бажаючим... Все це і не тільки побачимо на власні очі в нашому одноденному прекрасному турі.</p> <p><b>Обрати</b></p>	 <p><b>Дивовижний БОГУСЛАВ і гора ТОТОХА</b></p> <p>Запрошуємо Вас поринути у невеличке та мальовниче містечко на Київщині - Богуслав. Місто розташовувалося на гранітних пагорбах вздовж річки Рось, що одразу притягує погляд на красу зелених берегів та висотних скель. Під час екскурсії ми відвідаємо музей Богуславщини, Свято-Троїцький храм, майстерню ручного ткацтва та пам'ятник Марусі Богуславі. А далі нас чекає відоме енергетичне місце сили гора Тотоха, яка вражає навіть екстрасенсів. Соди приїдять з усіх куточків України та й не тільки аби «зарядитися енергією». Що ж, спробуємо і ми) гора Т</p> <p><b>Обрати</b></p>	 <p><b>ВІДПУСТКА В КАРПАТАХ ТА ЗАКАРПАТТІ</b></p> <p>Де влітку найкраще відпочити? А головне, де буде не шумно та затішно. Відпочити від буднів в гамірному місті можна лише в неймовірних Карпатах та Закарпатті. Вони точно зачарують своїми краєвидами, чистим повітрям та подарують найкраще свято для душі. Душа просить відпочинку... Відпочинок на тиждень в Карпатах та Закарпатті взимку і влітку однаково гарний! Тут вам і європейський сервіс, і розваги на всі смаки, і ціни гаманець не кусають. Залиште всі справи вдома і відчуйте на собі магічний вплив карпатських гір. Дозвольте собі зануритись у неймовірну атмосферу українського Закарпаття. Поїхавши в тур в Карпати на тиждень, ви точно запам'ятаєте їх надовго.</p> <p><b>Обрати</b></p>

Рисунок 4.15. Варіанти рекомендацій

Навчені нейронні мережі зберігають результати свої алгоритмів у файлах і можуть використати їх коли відбувається запит про аналіз вхідних даних.

Фрагмент лістингу збереженої моделі нейронної мережі (pick.model)

```
O:34:"Phpml\Classification\MLPClassifier":8:{s:10:" * classes";a:6:
{i:0;s:3:"i40";i:1;s:3:"i20";i:2;s:4:"so20";i:3;s:4:"so10";i:4;s:4:"sa10";i:5
;s:3:"sa5";}s:21:" * activationFunction";N;s:18:" *
backpropagation";O:44:"Phpml\NeuralNetwork\Training\Backpropagation":3:
{s:58:" Phpml\NeuralNetwork\Training\Backpropagation learningRate";d:1;s:52:"
Phpml\NeuralNetwork\Training\Backpropagation sigmas";a:0:{s:56:"
```

```

Phpml\NeuralNetwork\Training\Backpropagation prevSigmas";a:0:{{}}s:68:"
Phpml\NeuralNetwork\Network\MultilayerPerceptron
inputLayerFeatures";i:6;s:62:"
Phpml\NeuralNetwork\Network\MultilayerPerceptron hiddenLayers";a:1:{i:0;i:2;}
s:62:" Phpml\NeuralNetwork\Network\MultilayerPerceptron
learningRate";d:1;s:60:" Phpml\NeuralNetwork\Network\MultilayerPerceptron
iterations";i:10000;s:9:" * layers";a:3:
{i:0;O:25:"Phpml\NeuralNetwork\Layer":1:{s:32:" Phpml\NeuralNetwork\Layer
nodes";a:7:{i:0;O:30:"Phpml\NeuralNetwork\Node\Input":1:{s:37:"
Phpml\NeuralNetwork\Node\Input input";d:0.3;}
i:1;O:30:"Phpml\NeuralNetwork\Node\Input":1:{s:37:"
Phpml\NeuralNetwork\Node\Input input";d:0.3;}
i:2;O:30:"Phpml\NeuralNetwork\Node\Input":1:{s:37:"
Phpml\NeuralNetwork\Node\Input input";d:0.3;}
i:3;O:30:"Phpml\NeuralNetwork\Node\Input":1:{s:37:"
Phpml\NeuralNetwork\Node\Input input";d:0.1;}
i:4;O:30:"Phpml\NeuralNetwork\Node\Input":1:{s:37:"
Phpml\NeuralNetwork\Node\Input input";d:0.1;}
i:5;O:30:"Phpml\NeuralNetwork\Node\Input":1:{s:37:"
Phpml\NeuralNetwork\Node\Input input";d:0.1;}
i:6;O:29:"Phpml\NeuralNetwork\Node\Bias":0:{{}}
i:1;O:25:"Phpml\NeuralNetwork\Layer":1:{s:32:" Phpml\NeuralNetwork\Layer
nodes";a:3:{i:0;O:31:"Phpml\NeuralNetwork\Node\Neuron":4:{s:11:" *
synapses";a:7:{i:0;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:17.532844338747168;s:7:" * node";r:22;}
i:1;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:19.532844338747154;s:7:" * node";r:24;}
i:2;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:19.532844338747154;s:7:" * node";r:26;}
i:3;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:9.37854761983922;s:7:" * d:-2.7523251159838935;}
i:5;O:31:"Phpml\NeuralNetwork\Node\Neuron":4:{s:11:" * synapses";a:3:
{i:0;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:11.96278462053462;s:7:" * node";r:37;}
i:1;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:-11.407951099643986;s:7:" * node";r:64;}
i:2;O:39:"Phpml\NeuralNetwork\Node\Neuron\Synapse":2:{s:9:" *
weight";d:-8.17527074679945;s:7:" * node";r:91;}}s:21:" *
activationFunction";r:105;s:9:" * output";d:0.9755271980683583;s:4:" *
z";d:3.685415665165058;}}}}}}

```

В адміністративній панелі сайту присутній функціонал навчання нейронної мережі статистичних даних відповідно до класифікації користувачі.

На сайт Головна Навчання Вихід

Додає JSON дані для навчання нейронної мережі з класифікації покупця.

Ок

Додає JSON дані для навчання нейронної мережі з рекомендації телефону.

Ок

Рисунок 4.16. Форма внесення JSON даних для навчання моделі

## ВИСНОВКИ

Як результат виконання кваліфікаційної роботи магістра, було розроблено систему аналізу даних і проаналізовано методи управління розробкою веб сервісу планування подорожей.

Було проведено аналіз сфери, пов'язаної з IT-проектом. Після пандемії COVID-19, глобальна економічна проблема у туристичній галузі залишається актуальною, і відновлення ринку займе кілька років. Для управління проектом була використана методологія Waterfall.

Розроблений програмний продукт є актуальним, для аналізу користувачів серед певних категорій людей.

В ході виконання магістерської роботи створено систему, яка складається з:

- БД, що містить в собі дані користувачів і проаналізовані дані;
- Веб-додаток аналізу профілів користувачів.
- Навчені моделі нейронної мережі PHP-ML які збурігаються в файлах

Створений програмний продукт дозволяє виконувати наступні функції:

- імпорт даних;
- аналіз вхідних даних;
- рекомендації подорожей;
- вивід статистичної інформації про користувачів.

Розроблено зручний інтерфейс, що оптимізує взаємодію користувача з інформаційною системою і дозволяє зручний доступ.

Впровадження створеної інформаційної системи автоматизує та спрощує аналіз даних.

Перевагою створеного ПЗ є оптимізація процесу аналізу статистичних даних та скорочення часу їх виконання з використанням зручного та легкого у використанні інтерфейсу.

Архітектуру інформаційної системи, яка агрегує підходи машинного навчання і аналізу даних, спроектована структура бази даних та клієнт-серверна взаємодія. Розроблено принцип зберігання навчених моделей і отриманих результатів.

Виконана програмна реалізація на мові програмування PHP, також були використані мова JavaScript, бібліотеки jQuery. Для інтерфейсу HTML, CSS.

В ході роботи на основі проведеного аналізу аналогів інформаційних систем для роботи з алгоритмами машинного навчання були сформульовані вимоги до проєктованої інформаційної системи, на основі яких розроблено:

1. Архітектуру інформаційної системи, яка агрегує підходи машинного навчання і аналізу даних, спроектована структура бази даних та клієнт-серверна взаємодія. Розроблено принцип зберігання навчених моделей і отриманих результатів класифікації.
2. Виконана програмна реалізація на мові програмування Python, також були використані мова JavaScript, фреймворки Flask і Angular.
3. Апробація системи була проведена з використанням двох наборів даних, кожен з яких ілюстрував вирішення однієї з типів задач класифікації. В результаті роботи системи для кожного з них був визначений оптимальний алгоритм на основі оцінки якості отриманих моделей.
4. Практичне застосування створеного аналітичного сервісу спрямоване на оптимізацію роботи науковців та дата-саєнтістів задіюючи мінімальні зусилля при роботі з алгоритмами для вирішення задач класифікації, підвищивши ефективність аналізу та час обробки даних.
5. Окрім того, дана аналітична система має перспективи розвитку, зокрема за рахунок додавання нових методів аналізу, наприклад, нейромережових моделей, а саме нейромережа – перцептрон за допомогою бібліотеки Keras. Також розширити функціонал системи можна за допомогою розробки класифікатора моделей. Принцип його роботи базуватиметься на зберіганні навчених моделей та їх

характеристик до бази даних. Вони можуть стати основою для створення класифікатора моделей, що підбиратиме тип алгоритму на основі характеристик масиву вхідних даних.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Управління проектами: процеси планування проектних дій: підручник / І.В. Чумаченко, В.В. Морозов, Н.В. Доценко, А.М. Чередніченко. – К.: Університет економіки та права «КРОК», 2014. – 670 с.
2. Бушуев С.Д. Креативні технології управління проектами і програмами: Монографія. /Бушуева Н.С., Бабаєв І.А., Яковенко В.Б., Гриша Е.В., Дзюба С.В., Войтенко А.С / – К.: «Саммит-Книга», 2010. –768 с.
3. Керівництво з управління інноваційними проектами та програмами Р2М: перекл. з англ.; під ред. С.Д. Бушуєва. – К.: Науковий світ, 2009. 173 с.
4. Semolic B., Steyn P. Project Excellence as part of Organizational Maturity// News Letter IPMA [Електронний ресурс]. — June 2013. — Volume 2, Issue 2. — Режим доступу: [www.ipma.ch/Documents/](http://www.ipma.ch/Documents/).
5. Бізнес-менеджмент: Навч. посіб. для студ. вищ. навч. закл. / Л.І. Федулова, В.Г. Федоренко, В.Ф. Гриньов, В.П. Сладкевич, В.Є. Воротін, А.Д. Чернявський, В.А. Коростельов, Л.С. Кобиляцький, В.Є. Скоцик, О.С. Курочкін; Міжрегіон. акад. упр. персоналом. – К.: Наук. світ, 2002. – 593 с.
6. Тімінський О. Г. Механізми створення системи захисту портфелю проектів в умовах сучасного агресивного проектного оточення / О. Г. Тімінський // Управління проектами та розвиток виробництва. - 2009. - № 2. - С. 56-60.
7. Морозов В. В. Модель впливу зовнішнього оточення на процес управління конфігурацією в проекті [Електронний ресурс] / В. В. Морозов, С. І. Рудницький // Управління розвитком складних систем. - 2013. - Вип. 16. - С. 46-52.
8. Бушуєв С. Д. Антикризове управління фінансовими установами в умовах турбулентності / С. Д. Бушуєв, Ю. Ф. Ярошенко // Управління розвитком складних систем. - 2013. - Вип. 15. - С. 5-10.

9. Навіщо нам UML [Електронний ресурс]. – URL: <https://habr.com/ru/post/458680/>
10. Діаграма прецедентів [Електронний ресурс] – URL: [https://uk.wikipedia.org/wiki/Діаграма\\_прецедентів](https://uk.wikipedia.org/wiki/Діаграма_прецедентів)
11. База даних [Електронний ресурс]. – URL: [https://uk.wikipedia.org/wiki/База\\_даних](https://uk.wikipedia.org/wiki/База_даних)  
Моделі баз даних [Електронний ресурс]. – URL: [https://uk.wikipedia.org/wiki/Моделі\\_баз\\_даних](https://uk.wikipedia.org/wiki/Моделі_баз_даних)
12. Ієрархічна модель даних [Електронний ресурс]. – URL: [https://uk.wikipedia.org/wiki/Ієрархічна\\_модель\\_даних](https://uk.wikipedia.org/wiki/Ієрархічна_модель_даних)
13. PostgreSQL [Електронний ресурс]. – URL: <https://uk.wikipedia.org/wiki/PostgreSQL>
14. Розробка з боку Frontend – що це таке і чим відрізняється від Backend [Електронний ресурс]. – URL: <https://dan-it.com.ua/uk/blog/rozrobka-z-boku-front-end-shho-ce-take-i-chim-vidriznjaietsja-vid-back-end/>
15. Bootstrap [Електронний ресурс]. – URL: <https://uk.wikipedia.org/wiki/Bootstrap>
16. Хто такі frontend- і backend-розробники [Електронний ресурс]. – URL: <https://te.itstep.org/blog/frontend-and-backend-developers>
17. Frontend і Backend розробка [Електронний ресурс]. – URL: <https://webtune.com.ua/statti/web-rozrobka/frontend-i-backend-rozrobka/#id2>
18. JSP [Електронний ресурс]. – URL: <https://uk.wikipedia.org/wiki/JSP>
19. Spring Framework [Електронний ресурс]. – URL: [https://uk.wikipedia.org/wiki/Spring\\_Framework](https://uk.wikipedia.org/wiki/Spring_Framework)
20. Fridlyand J. Dudoit S. Application of Resampling Methods to Estimate the Number of Clusters and to Improve the Accuracy of a Clustering Methods. — Stat. Berkely Tech. Report, 2001. — P. 600.
21. Google Machine Learning. — URL: <https://cloud.google.com/products/machine-learning> (online; accessed: 21.05.2017).

22. Granichin O. Volkovich Z. Toledano-Kitai D. Randomized Algorithms in Automatic Control and Data Mining. — USA, New York, 2015. — P. 275.
23. Hubert L. Arabie P. Comparing Partitions // Journal of Classification. — 1985. — P. 193—218.
24. Jain A. Dubes R. Algorithms for Clustering Data. — Prentice-Hall, Englewood Cliffs, 1988. — P. 320.
25. Kaggle. — URL: <https://www.kaggle.com/c/digit-recognizer> (online; accessed: 21.05.2017).