

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“ ____ ” _____ 2024 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи моніторингу
роботи соціального сервісу з використанням методів
кластерного аналізу”

Виконав здобувач вищої освіти
II курсу, групи КН-23М
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Ширяєв О.І.
« ____ » _____ 2024 р.

Керівник проекту
кандидат технічних наук
_____ Смірнова Т.В.
« ____ » _____ 2024 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Рівень вищої освіти магістр
Галузь знань 12 "Інформаційні технології"
Спеціальність 122 "Комп'ютерні науки"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерні науки"

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 6 » вересня 2024 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Ширяєву Олегу Ігоровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу

2. Керівник роботи Смірнова Тетяна Віталіївна, канд. техн. наук

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 18-13 від 07.08.2024 року

3. Строк подання студентом роботи до захисту 2.12.2024 р.

4. Мета та завдання випускної кваліфікаційної роботи: Метою розробки є дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу

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

1. Призначення та область використання.

6. Наукова новизна.

2. Перегляд аналогічних існуючих систем.

7. Маркетингове та економічне обґрунтування ІТ-проєкту.

3. Опис і обґрунтування проектних рішень.

8. Заходи з охорони праці та техніки безпеки.

4. Етапи програмування системи.

9. Висновки.

5. Впровадження системи в промислову експлуатацію

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Наукова новизна 1 аркуш

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

Показники економічної ефективності 1 аркуш

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Доренська А.О.	05.10.2024	14.11.2024
Охорона праці	Марченко К.М., к.т.н., доцент	06.10.2024	16.11.2024

7. Дата видачі завдання « 6 » вересня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2024 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2024 р.	
3.	Розробка моделі компонента	20.10.2024 р.	
4.	Розробка структур даних	25.10.2024 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2024 р.	
6.	Програмування алгоритмів	10.11.2024 р.	
7.	Розрахунок економічної ефективності	13.11.2024 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2024 р.	
9.	Оформлення ПЗ	17.11.2024 р.	
10.	Попередній захист роботи	2.12.2024 р.	

Дата видачі завдання
« 6 » вересня 2024 р.

Підпис керівника

(прізвище та ініціали)Завдання прийнято до виконання
« 6 » вересня 2024 р.

Підпис здобувача

(прізвище та ініціали)

АНОТАЦІЯ

Ширяєв О.І. Дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2024.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Метою розробки є дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Об'єктом дослідження є процес моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Предметом дослідження є методи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Методи дослідження базуються на методах кластерного аналізу, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі PHP фреймворк Yii2.

Ключові слова: Комп'ютерні науки, кластерний аналіз

ABSTRACT

Shiryaev O.I. Research and software implementation of the social service monitoring system using cluster analysis methods. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2024.

In this graduation thesis for the second (master's) level of higher education, software was developed, which is intended for the system of monitoring the work of social services using cluster analysis methods.

The purpose of the development is the research and software implementation of the social service monitoring system using cluster analysis methods.

The object of the research is the process of monitoring the work of the social service using cluster analysis methods.

The subject of the study is methods of monitoring the work of social services using methods of cluster analysis.

Research methods are based on methods of cluster analysis, methods of mathematical statistics, methods of software development.

The result of the work is the software implementation of the social service monitoring system using cluster analysis methods.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on a PC with Windows 10/11 OS.

The program was developed in the PHP environment of the Yii2 framework.

Keywords: Computer science, cluster analysis

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	6
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	9
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	9
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	32
2.3 Розгорнута постановка завдання	37
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	39
3.1 Опис функціонування системи	39
3.2 Розробка структурної схеми.....	43
3.3 Розробка функціональної схеми	46
3.4 Розробка діаграми процесів.....	49
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	51
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	51
4.2 Захист розробленого програмного забезпечення.....	62
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	67
6 НАУКОВА НОВИЗНА	71

					ВКРМ-122.24.0020.00.00.ПЗ			
Вим	Арк.	№ докум.	Підп.	Дата	Дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу	Літ.	Аркуш	Аркушів
Розроб.	Ширяєв О.І.					М	1	97
Перев.	Смірнова Т.В.							
Н.контр.	Коваленко А.С.					ЦНТУ КН-23М		
Затв.	Смірнов О.А.							

7	МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ	72
7.1	Визначення цільової аудиторії кінцевого готового продукту	72
7.2	Оцінка привабливості шляхом застосування методів експертних оцінок ...	73
7.3	Вибір методу оцінки вартості ПЗ	74
7.4	Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості.....	75
7.5	Пропозиція алгоритму просування проєкту розробки ПЗ	76
7.6	Оптимізація каналів збуту та шляхів реалізації ПЗ	77
7.7	Визначення ключових факторів успіху конкретного проєкту.....	78
8	ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	80
8.1	Вступ.....	80
8.2	Аналіз умов праці на робочому місці ІТ-фахівця	81
8.3	Пропозиції щодо підвищення працездатності ІТ-фахівців.....	84
8.4	Розрахункова частина	86
8.5	Висновки до розділу.....	88
9	ОСНОВНІ ВИСНОВКИ.....	89
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	91

КБПЗ-2024

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ЕОМ	– електрона обчислювальна машина
КС	– комп'ютерна система
КТЗ	– комплекс технічних засобів
МЕТОЗ	– методичне забезпечення
ОС	– операційна система
ОРЗ	– організаційне забезпечення
ПЗ	– програмне забезпечення
ПК	– персональний комп'ютер
ПП	– програмний продукт
ППЗ	– прикладне програмне забезпечення
ППП	– пакет прикладних програм
БД	– база даних
DLL	– бібліотека динамічної компоновки

КБПЗ-2024

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

ВСТУП

Актуальність теми. В сучасному світі люди вже не можуть уявити свого життя без інтернету та соціальних сервісів. Одним із видів соціальних сервісів є веб-сервіси, які надають можливість користувачам задати питання або відповісти на вже задані питання. Таким чином формується соціальна мережа, учасником якої може стати кожен. У таких системах кожен користувач стикався з необхідністю швидко знайти необхідну інформацію. Тому використання «інтелектуального» групування результатів при пошуку необхідної інформації надає можливість вирішення проблеми релевантного та швидкого пошуку в надзвичайно великих обсягах інформації.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.
- Дослідження системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.
- Програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Об'єктом дослідження є процес моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Предметом дослідження є методи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Методи дослідження базуються на методах кластерного аналізу, методах математичної статистики, методах розробки програмного забезпечення.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

– Розроблено вітчизняний продукт моніторингу роботи соціального сервісу з використанням методів кластерного аналізу, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2024 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Розроблена система призначена для обміну знаннями між користувачами. Користувачі мають змогу поставити запитання або відповісти на вже поставлені запитання. Методи кластерного аналізу виконують функцію групування результатів при пошуку інформації, надаючи користувачу можливість швидкої навігації, вибору свідомо більш релевантної підмножини і виключення свідомо менш релевантного – що підвищує зручність і простоту використання інтерфейсу в порівнянні з виведенням у вигляді простого сортованого за релевантністю списку.

Розв'язувана кластеризацією проблема очевидна: якщо користувачеві хочеться переглядати тільки сторінки певного типу, не витрачаючи часу на перегляд інших, він має можливість відразу ж вибрати потрібну групу (кластер). Час пошуку при цьому скорочується, а якість зростає.

Програмні рішення для даної програми не потребують для своєї роботи саме персонального комп'ютера. Веб-сервісами можна користуватись з будь-яких доступних користувачеві пристроїв, які зазвичай людина має змогу використовувати і мати з собою там, де їй потрібно.

Програмне забезпечення, що розробляється може застосовуватися, як для індивідуальних потреб, так для роботи і освіти.

1.2 Область застосування

Область застосування даного програмного продукту – класифікація великої кількості інформації. Використані методи кластерного аналізу також можна застосувати в таких галузях:

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

- інформатика;
- аналіз даних;
- вилучення і пошук інформації;
- групування і розпізнавання об'єктів;
- маркетинг (задача сегментації конкурентів і споживачів);
- соціологія (розбиття респондентів на однорідні групи);
- медицина (лікування захворювань або їх симптомів, кластеризація захворювань, а також таксономія пацієнтів, препаратів і т.д).

Динаміка сучасного життя наполегливо вимагає переходу на дистанційну освіту в бізнесі, в вузах і навіть школах. Переваги дистанційної освіти очевидні – це економить багато часу і не відриває від виробничого процесу.

Якщо ж учень не має можливості на даний момент отримати потрібну інформацію від вчителя, то він може отримати її за допомогою веб-сервісу, що розробляється, від інших компетентних користувачів, які також можуть виправити помилки або ж підказати шляхи вирішення проблеми.

Веб-сервіс, що розробляється, може використовувати широке коло користувачів, які самостійно бажають обмінюватись знаннями і досвідом. Користувачі можуть знайти тут корисну інформацію для себе і зекономити час пошуку потрібних даних.

Розроблюваний соціальний сервіс, представлений в магістерській роботі, може знайти своє застосування в сфері освіти, як джерело додаткової інформації та обміну знаннями.

Отже, програмне забезпечення програмне забезпечення моніторингу роботи соціального сервісу з використанням методів кластерного аналізу може використовуватись студентами і учнями та широким колом користувачів для покращення і обміну знаннями.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

Методи кластерного аналізу виконують функцію групування результатів при пошуку інформації, надаючи користувачу можливість швидкої навігації, вибору свідомо більш релевантної підмножини і виключення свідомо менш релевантного – що підвищує зручність і простоту використання інтерфейсу в порівнянні з виведенням у вигляді простого сортованого за релевантністю списку.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ – 2024

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Предметом розробки магістерського проекту (ДП) є механізми роботи соціального сервісу, в умовах швидкого вибору і сортування посилань на веб-сторінки відповідно до функцій значущості інформаційних ресурсів та релевантності документа запиту.

Існують такі аналоги:

- Stack Overflow.
- Quora.
- Question.

Перераховані вище веб-сервіси в даний час використовують модель вільного обміну знаннями. В якості оплати вони пропонують збільшення репутації користувачам.

Stack Overflow – популярна система питань і відповідей про програмування. Є частиною Stack Exchange Network (Рисунок 2.1).

В системі користувачі ставлять питання, на які можуть відповідати інші користувачі. Також є можливість коментувати питання і відповіді. Існує система оцінювання питань та відповідей, а також система рейтингу та нагород користувачів. Репутація зростає коли інші голосують за питання користувача, відповіді та редагування. По мірі отримання репутації користувачам будуть доступні нові привілеї, наприклад, можливість голосувати, коментувати і навіть редагувати повідомлення інших учасників.[1]

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

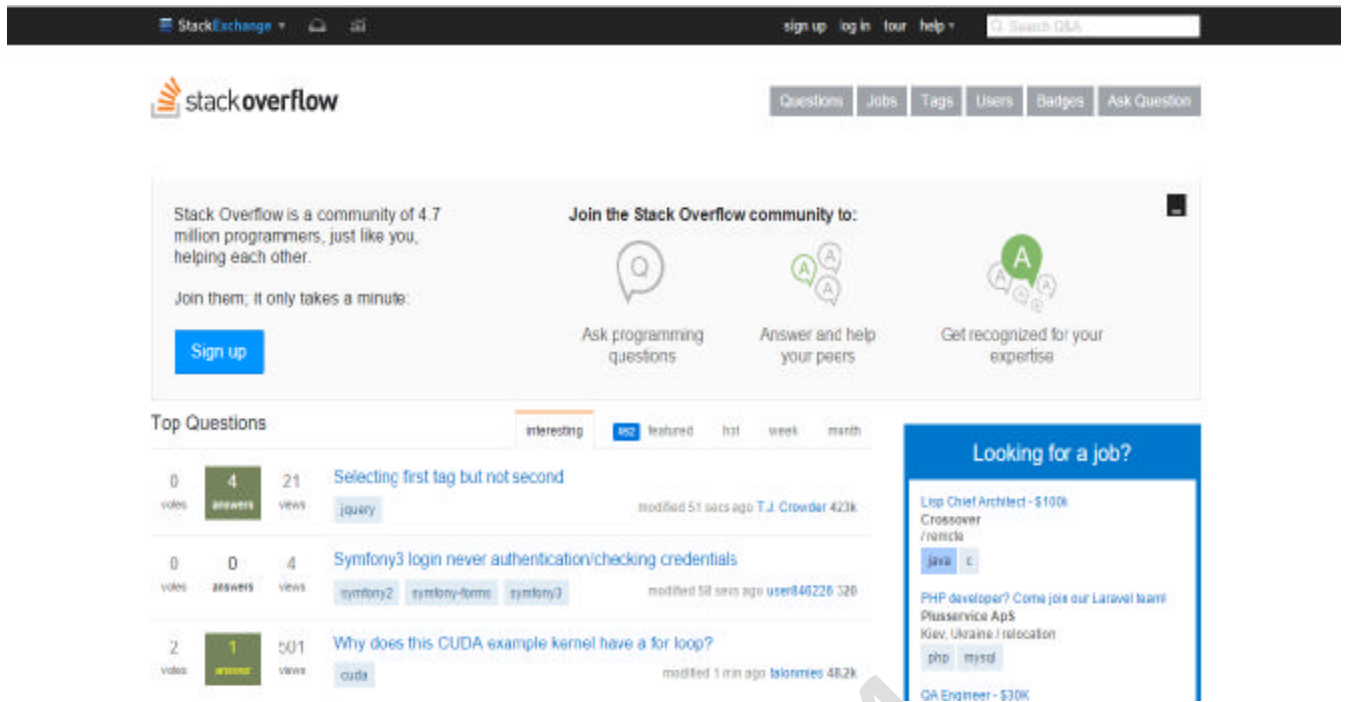


Рисунок 2.1 – Скріншот головної сторінки веб-сервісу Stack Overflow

До недоліків даного веб-сервісу можна віднести:

- відсутність безпосереднього завантаження зображень (користувачам потрібно завантажувати зображення на сторонній хостинг);
- відсутня реалізація приватних повідомлень між користувачами;
- незареєстровані користувачі не можуть задати питання.

Quora – соціальний сервіс, на якому користувачі можуть ставити відкриті для аудиторії питання та отримати на них відповіді, а також функціонує як база знань: архів запитань-відповідей, що об'єднані навколо тематичних кіл (рисунок 2.2).

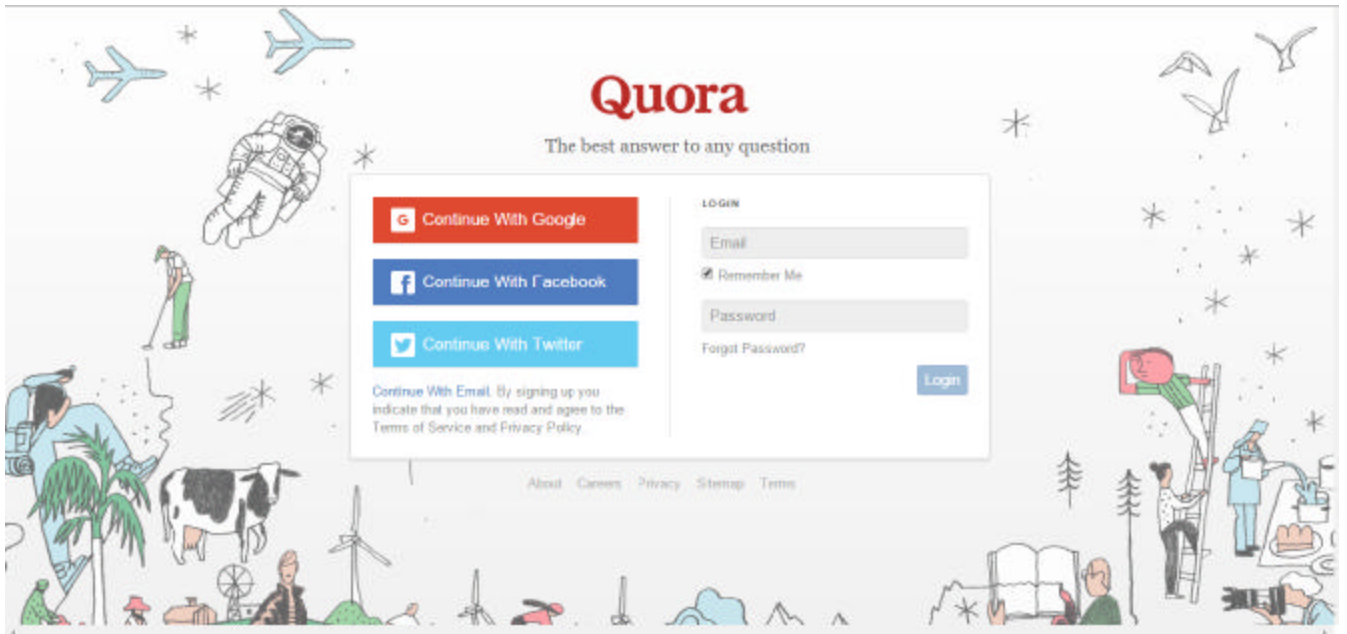


Рисунок 2.2 – Скріншот головної сторінки веб-сервісу Quora

На відміну від великої кількості подібного роду сервісів, Quora дозволяє відстежувати питання і відповіді своїх друзів з Facebook або Twitter. Користувачі можуть підписуватися на цікаві для них теми і відповіді на окремі питання, отримуючи таким чином поновлення, наприклад, по електронній пошті. На сторінках Quora у верхній частині розміщена форма для пошуку тем і питань.

Головні недоліки розглянутого веб-сервісу:

- контент генерується практично виключно англійською мовою, тому є доступним не для всіх україномовних користувачів;
- незареєстровані користувачі не мають можливості переглянути контент веб-сервісу.

Quintura Search – візуальна пошукова система.

Quintura Search створена як надбудова для традиційних пошукових систем. Quintura Search дозволяє вам вибрати кращі результати серед знайдених обраної користувачем пошукової системи шляхом маніпулювання контекстом на візуальній карті, яка показує, ті слова, які найбільш тісно пов'язані з запитом.

Особливість Quintura – візуальна «хмара», або карта, слів-асоціацій між зображеннями або документами, яка є зручним навігатором по ним. Використовуючи принцип Quintura, коли пошук здійснюється одним клацанням миші, користувач бачить слова на карті і клацає мишкою по ним, швидко знаходячи необхідні зображення і документи.

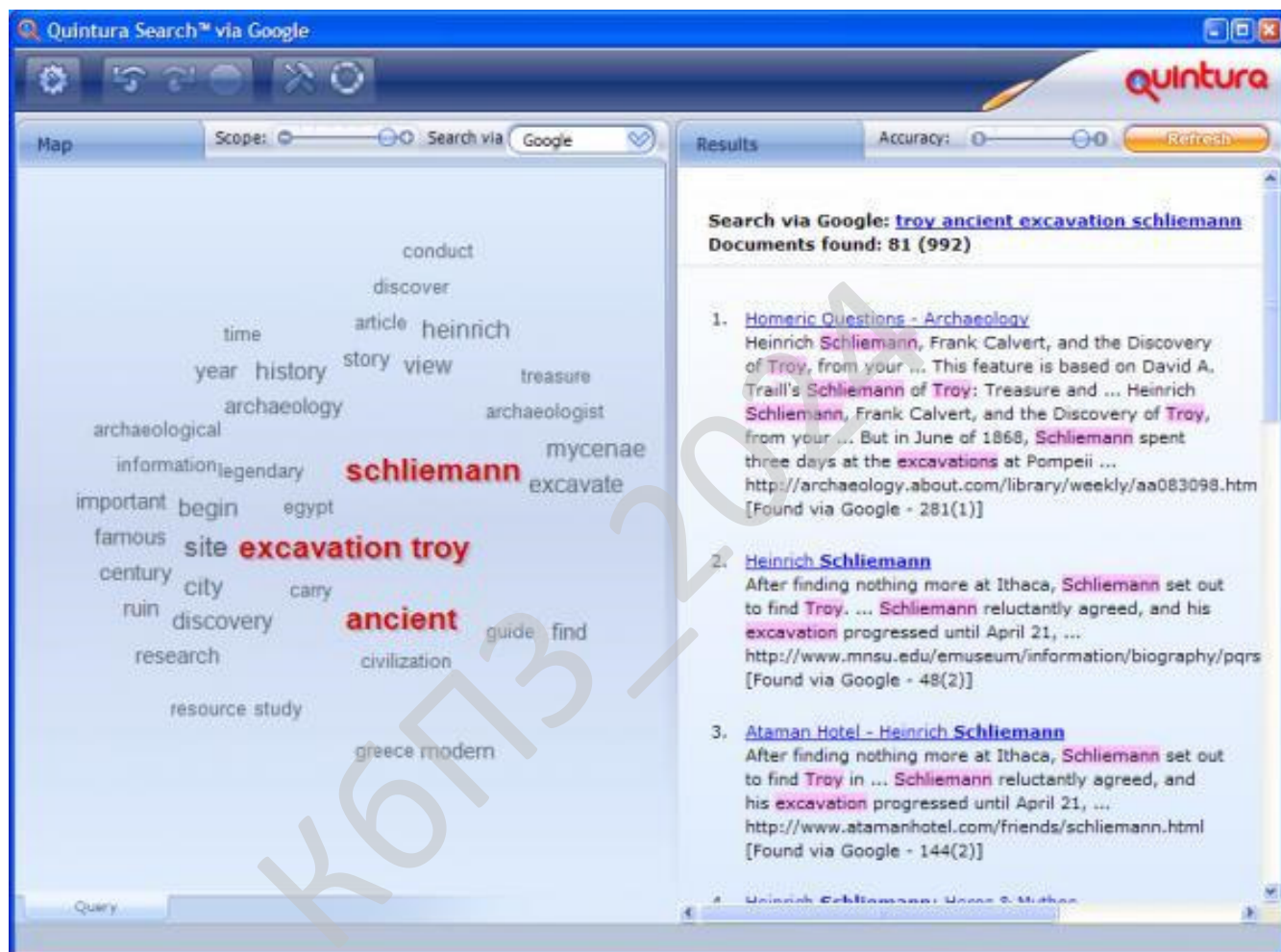


Рисунок 2.3 – Скріншот Quintura Search

Традиційні пошукові системи, як правило, «байдужі» до предмету пошуку користувача в Інтернеті, включаючи пошук по картинках. На відміну від них, Quintura пропонує слова-асоціації у вигляді візуальної хмари. Ці слова допомагають користувачеві спілкуватися з пошуковою системою на одній мові.

– Групування результатів пошуку по пошуковим системам, в яких були знайдені результати пошуку.

– Групування результатів пошуку по доменних зонах (наприклад.org,.com і ін.).

Результати кластеризації по найбільш поширеним в результатах пошуку ключовим словам представляються у вигляді списку пунктів меню, за якими можливе пересортування результатів видачі. При відображенні кластери упорядковуються за статистикою знайдених слів. Є можливість вказати обмеження області пошуку відповідним розділом (наприклад, новини, зображення, вікіпедія, блоги, робота, магазини і держава). Можна вказати обмеження на число аналізованих результатів (100, 200, 500) і на час реакції = системи (5, 10, 30 секунд). Пошук з урахуванням словозміни (облік морфології) реалізований тільки для англійської мови. Пошук за енциклопедіями реалізований тільки на основі використання метапошуку по англійській частині енциклопедії wikipedia, навіть якщо запит користувача введений російською мовою. Метапошукова система не використовує ні синтаксичного, ні семантичного аналізу текстів. Пошук проводиться по всім словам запиту користувача. Реалізований кластерний аналіз не дозволяє отримувати точні результати пошуку при введенні запиту користувача на розмовній мові. Це обумовлено відсутністю синтаксичного і семантичного аналізу, які дозволяють виділити головну суть в запиті користувача і визначити вимоги до смислового змісту тексту відповіді.

Отже, метапошукова система vivisimo (clusty) належить до класу метапошукових систем, що здійснюють кластеризацію результатів пошуку існуючих пошукових систем.

Система vivisimo (clusty) за рахунок використання кластерного аналізу має характеристики пошуку, що відрізняються від традиційних пошукових систем (можливість виділення слів, що часто зустрічаються разом зі словами пошукового запиту). Однак використання тільки кластерного аналізу не дає істотних переваг.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		14

Технологія кластерного аналізу має особливості, що дозволяють на деяких запитах користувачів отримати кращу якість пошуку в порівнянні з традиційними пошуковими системами. Однак, на інших запитах якість пошуку у систем кластерного аналізу гірше, ніж у традиційних пошукових систем. Це просто системи одного рівня, але дещо відрізняються в частині подання інформації користувачу.

Якісне підвищення ефективності пошукових систем можливо тільки за рахунок використання синтаксичного і семантичного аналізу тексту, а існуючі пошукові системи (як традиційні, так і кластерні) цього не реалізують.

Недоліками є:

- пошукова структура не дає обширні результати;
- обмежений синтаксис;
- найбільша пошукова система Google не використовується.

Кластерний аналіз

Кластерний аналіз – це техніка, яка дозволяє групувати схожі інвестиції на основі їхніх профілів ризику та прибутку. Роблячи це, ми можемо визначити оптимальний розподіл портфеля, який максимізує наш очікуваний прибуток для певного рівня ризику або мінімізує наш ризик для даного рівня прибутку. Однак виконання кластерного аналізу не є тривіальним завданням, оскільки вимагає вибору правильного методу, правильної кількості кластерів і правильних змінних для розгляду. У цьому розділі ми розглянемо деякі з найкращих інструментів і програмного забезпечення, які можуть допомогти нам виконувати кластерний аналіз ефективно та результативно.

Деякі з найкращих інструментів і програмного забезпечення для виконання кластерного аналізу:

1. R: R – це безкоштовна мова програмування з відкритим кодом, яка широко використовується для статистичного аналізу та візуалізації даних. У R є багатий набір пакетів, які підтримують різні типи кластерного аналізу, наприклад ієрархічну кластеризацію, кластеризацію k-середніх, нечітку кластеризацію та

кластеризацію на основі моделі. R також дозволяє нам створювати інтерактивні графіки та інформаційні панелі, які можуть допомогти нам досліджувати та повідомляти результати кластерного аналізу. Наприклад, ми можемо використовувати пакет factoextra для візуалізації кластерів і їх центроїдів, пакет NbClust для визначення оптимальної кількості кластерів і пакет clusterCrit для оцінки якості кластерів.

2. Python: Python – це ще одна безкоштовна мова програмування з відкритим вихідним кодом, популярна для науки про дані та машинного навчання. Python також має широкий спектр бібліотек, які підтримують кластерний аналіз, таких як scikit-learn, scipy, pandas і matplotlib. Python пропонує простий та інтуїтивно зрозумілий синтаксис, який полегшує реалізацію та налаштування алгоритмів кластерного аналізу. Наприклад, ми можемо використовувати клас KMeans від scikit-learn для кластеризації k-means, функцію дендрограми від scipy для побудови ієрархічного дерева кластеризації та функцію pairplot від pandas для візуалізації парних зв'язків між змінними.

3. Excel: Excel – це програма для роботи з електронними таблицями, яка широко використовується для аналізу даних і звітності. Excel має вбудовану функцію під назвою Data Analysis ToolPak, яка дозволяє нам виконувати базовий кластерний аналіз, наприклад кластеризацію k-середніх та ієрархічну кластеризацію. Excel також має різноманітні надбудови та розширення, які можуть покращити його можливості кластерного аналізу, наприклад XLSTAT, Solver та Analytic Solver. Excel – це зручний і знайомий інструмент, який може допомогти нам швидко й легко виконати кластерний аналіз. Наприклад, ми можемо використовувати опцію Cluster Analysis з Data Analysis ToolPak, щоб виконати кластеризацію k-середніх, і тип діаграми Clustered Column, щоб візуалізувати кластери та їхні розміри.

Рейтинговий аналіз

Аналіз ранжування вартості – це техніка, яка допомагає оцінити майбутні витрати на проект, продукт або послугу шляхом порівняння різних альтернатив

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

на основі їх відносної економічної ефективності. Інструменти та програмне забезпечення для ранжирування витрат – це програми, які полегшують цей процес, надаючи такі функції, як збір даних, обчислення, візуалізація та звітність. У цьому розділі ми розглянемо деякі з найкращих інструментів і програмного забезпечення для проведення аналізу рейтингу витрат і те, як вони можуть допомогти вам приймати кращі рішення.

Деякі з найкращих інструментів і програмного забезпечення для проведення аналізу рейтингу витрат:

1. CostRank: CostRank – це веб-інструмент, який дозволяє проводити аналіз рейтингу витрат для будь-якого типу проекту чи сценарію. Ви можете створювати власні критерії та альтернативи, призначати ваги та бали, а також генерувати коефіцієнти економічної ефективності та рейтинги. CostRank також надає інтерактивні діаграми та таблиці, які допомагають візуалізувати та порівняти результати. Ви можете експортувати дані та звіти в різні формати, наприклад Excel, PDF або CSV. CostRank простий у використанні та має зручний інтерфейс. Ви можете підписатися на безкоштовну пробну версію або вибрати план передплати, який відповідає вашим потребам. Наприклад, ви можете використовувати CostRank, щоб порівняти економічну ефективність різних джерел енергії для вашого будинку чи підприємства.

2. CostRanker: CostRanker – це програмне забезпечення, яке допомагає виконувати аналіз рейтингу витрат для інженерних і будівельних проектів. Ви можете ввести дані проекту, такі як витрати, переваги, ризики та невизначеності, і CostRanker розрахує коефіцієнти ефективності та рейтинги для кожної альтернативи. CostRanker також забезпечує аналіз чутливості та моделювання методом Монте-Карло для врахування мінливості та невизначеності ваших даних. CostRanker має графічний інтерфейс користувача, який дозволяє створювати та редагувати ваші моделі, а також переглядати та експортувати результати. CostRanker сумісний з операційними системами Windows і Linux. Ви можете завантажити безкоштовну демо-версію або придбати ліцензію на повну версію.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

Наприклад, ви можете використовувати CostRanker, щоб порівняти економічну ефективність різних варіантів конструкції моста чи дамби.

3. CostRanking: CostRanking – це хмарна платформа, яка дає змогу виконувати аналіз рейтингу витрат на охорону здоров'я та втручання в охорону здоров'я. Ви можете завантажити свої дані, такі як витрати, результати та характеристики популяції, а CostRanking виконає аналіз і створить коефіцієнти ефективності та рейтинги для кожного втручання. CostRanking також надає інструменти для аналізу сценаріїв, аналізу впливу на бюджет та аналізу цінності інформації. CostRanking має веб-інтерфейс, який дозволяє вам отримувати доступ до своїх проектів і керувати ними, а також ділитися та співпрацювати з іншими. CostRanking безкоштовний для академічних і некомерційних користувачів і має модель оплати за використання для комерційних користувачів. Наприклад, ви можете використовувати CostRanking, щоб порівняти економічну ефективність різних стратегій вакцинації або програм скринінгу.

Симуляція сценарію витрат

Одним із найважливіших аспектів моделювання сценарію витрат є вибір правильних інструментів і програмного забезпечення для виконання аналізу. На ринку доступно багато варіантів, кожен зі своїми функціями, перевагами та обмеженнями. У цьому розділі ми розглянемо деякі з найпопулярніших і широко використовуваних інструментів і програмного забезпечення для моделювання сценарію витрат і порівняємо їх сильні та слабкі сторони. Ми також надамо кілька прикладів того, як ці інструменти та програмне забезпечення можна застосувати до різних типів сценаріїв витрат.

Деякі з доступних інструментів і програмного забезпечення для моделювання сценарію витрат:

1. Microsoft Excel: це програма для роботи з електронними таблицями, яку можна використовувати для створення й обробки даних, виконання обчислень і створення діаграм і графіків. Excel є одним із найпоширеніших і доступних інструментів для моделювання сценарію витрат, оскільки він простий у

використанні, гнучкий і сумісний з іншими програмами. Excel може обробляти як прості, так і складні сценарії витрат залежно від рівня знань і налаштувань користувача. Однак Excel також має деякі недоліки, такі як обмежена масштабованість, безпека та точність. Excel може не впоратися з великими обсягами даних, складними формулами або кількома сценаріями без шкоди для продуктивності чи надійності. Excel також вимагає ручного введення та перевірки даних, що може збільшити ризик помилок і невідповідностей.

2. Кришталева куля: це програмна надбудова для Excel, яка розширює його можливості для моделювання сценарію витрат. Crystal Ball дозволяє користувачам виконувати моделювання методом Монте-Карло, аналіз чутливості, оптимізацію та прогнозування на основі даних Excel. Crystal Ball може допомогти користувачам врахувати невизначеність, мінливість і ризик у своїх сценаріях витрат, а також досліджувати різні результати та альтернативи. Crystal Ball також надає графічні та статистичні інструменти для візуалізації та передачі результатів моделювання. Crystal Ball – це потужний і зручний інструмент для моделювання сценарію витрат, але він також має деякі обмеження, наприклад високу вартість, залежність від Excel і проблеми сумісності з іншими програмами чи платформами.

3. @RISK: це ще одна програмна надбудова для Excel, яка дає змогу користувачам виконувати аналіз ризиків і моделювати дані Excel. @RISK використовує моделювання Монте-Карло для створення розподілу ймовірностей можливих результатів для будь-якої ситуації, яка включає невизначеність. @RISK може допомогти користувачам кількісно оцінити ризики та керувати ними, оцінити альтернативи та оптимізувати рішення в їхніх сценаріях витрат. @RISK також пропонує ряд графічних і статистичних інструментів для відображення та аналізу результатів моделювання. @RISK – це всеосяжний і універсальний інструмент для моделювання сценарію витрат, але він також має деякі труднощі, такі як крута крива навчання, висока вартість і залежність від Excel.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

4. Simul8: це окреме програмне забезпечення, яке дозволяє користувачам створювати та запускати моделі симуляції дискретних подій будь-якого процесу чи системи. Simul8 може допомогти користувачам змоделювати поведінку та продуктивність сценаріїв витрат за різних умов і припущень, а також перевірити вплив змін і вдосконалень. Simul8 може обробляти складні та динамічні сценарії витрат, такі як ті, що включають черги, ресурси, графіки або мережі. Simul8 також надає низку графічних та аналітичних інструментів для візуалізації та звітування про результати моделювання. Simul8 – це гнучкий і надійний інструмент для моделювання сценарію витрат, але він також має деякі недоліки, такі як висока вартість, крута крива навчання та обмежена інтеграція з іншими програмами чи джерелами даних.

Чутливість до вартості

Інструменти моделювання чутливості до витрат відіграють вирішальну роль в оцінці впливу змін вартості на результати проекту. Ці інструменти та програмне забезпечення надають цінну інформацію з різних точок зору, дозволяючи організаціям приймати обґрунтовані рішення. Ось деякі з доступних інструментів для виконання моделювання чутливості до витрат:

1. Моделювання за методом Монте-Карло: цей широко використовуваний метод передбачає виконання кількох моделювань з різними вхідними витратами для аналізу діапазону можливих результатів. Призначаючи розподіли ймовірностей змінним витрат, організації можуть оцінити ймовірність досягнення конкретних результатів проекту.

2. Аналіз чутливості: цей інструмент допомагає визначити ключові чинники витрат та їхній вплив на результати проекту. Змінюючи окремі витрати, зберігаючи інші постійними, організації можуть зрозуміти чутливість результатів проекту до змін конкретних факторів витрат.

3. Діаграми Tornado: діаграми Tornado візуально представляють результати аналізу чутливості шляхом ранжування важливості різних чинників

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

витрат. Довжина смужок вказує на величину впливу, дозволяючи особам, які приймають рішення, зосередитися на найбільш впливових факторах.

4. Аналіз сценаріїв: цей інструмент передбачає створення різних сценаріїв шляхом коригування змінних витрат для вивчення різних ситуацій «що-якщо». Порівнюючи результати різних сценаріїв, організації можуть отримати уявлення про потенційний вплив варіацій вартості на успіх проекту.

5. Деревя рішення: Деревя рішення забезпечують структурований підхід до аналізу чутливості витрат. Намітивши різні шляхи прийняття рішень і відповідні витрати, організації можуть оцінити потенційні результати та прийняти обґрунтовані рішення на основі міркувань витрат.

6. Аналіз витрат і вигод: цей інструмент порівнює витрати та вигоди різних варіантів або сценаріїв проекту. За допомогою кількісного визначення витрат і оцінки відповідних вигод організації можуть оцінити фінансову життєздатність різних стратегій, чутливих до витрат.

Важливо зазначити, що ці інструменти не є вичерпними, і організації можуть використовувати їх комбінацію на основі своїх конкретних потреб і вимог. Приклади їх застосування включають оцінку впливу коливань вартості матеріалів на будівельні проекти, аналіз чутливості виробничих витрат до оптимізації ланцюжка поставок або оцінку фінансових наслідків коливань цін у розробці продукту.

Аналіз чутливості у вартості

Аналіз за допомогою моделювання структури витрат виконання аналізу чутливості при моделюванні структури витрат вимагає використання відповідних інструментів і програмного забезпечення. Ці інструменти можуть полегшити складні обчислення, імовірнісний аналіз і моделювання сценаріїв. Ось деякі інструменти та програмне забезпечення, які зазвичай використовуються для проведення аналізу чутливості:

1. Microsoft Excel: Microsoft Excel є широко використовуваним інструментом для виконання аналізу чутливості. Він пропонує різні вбудовані

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

функції та функції, які дозволяють користувачам проводити аналіз чутливості, наприклад таблиці даних, диспетчер сценаріїв і пошук цілі.

2. Програмне забезпечення для моделювання за методом Монте-Карло: програмне забезпечення для моделювання за методом Монте-Карло, таке як @RISK і Crystal Ball, надає розширені можливості для імовірнісного аналізу. Ці інструменти дозволяють користувачам запускати кілька моделювань із випадково згенерованими значеннями для змінних і оцінювати розподіл ймовірностей результатів.

3. Інструменти візуалізації даних: інструменти візуалізації даних, такі як Tableau та Power BI, можна використовувати для візуалізації результатів аналізу чутливості. Ці інструменти дозволяють користувачам створювати інтерактивні інформаційні панелі та діаграми, які покращують розуміння та інтерпретацію аналізу.

4. Програмне забезпечення для моделювання витрат: програмне забезпечення для моделювання витрат, таке як Costimator і CostX, надає спеціалізовані можливості для моделювання структури витрат і аналізу чутливості. Ці інструменти пропонують розширені функції для моделювання складних структур витрат, проведення аналізу "що, якщо" та оцінки чутливості.

5. Статистичне програмне забезпечення: статистичне програмне забезпечення, таке як R і SAS, можна використовувати для розширеного статистичного аналізу та моделювання чутливості. Ці інструменти забезпечують широкий спектр статистичних функцій і методів, які можуть підвищити точність і надійність аналізу.

Вибираючи інструменти та програмне забезпечення для виконання аналізу чутливості, враховуйте такі фактори, як складність структури витрат, специфічні вимоги до аналізу та наявні ресурси. Виберіть інструменти, які пропонують необхідні можливості та сумісні з вашими існуючими системами та робочими процесами. Крім того, враховуйте простоту використання, масштабованість і підтримку, яку надають постачальники програмного забезпечення.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

часовій шкалі проекту можна моделювати тривалість завдання як трикутний розподіл і імітувати дату завершення проекту.

– Приклад: уявіть собі оцінку проекту розвідки нафти. Моделюючи ціни на нафту, витрати на буріння та запаси, ви можете оцінити ймовірність досягнення позитивної чистої приведеної вартості (NPV).

3. Python (NumPy, Pandas і Matplotlib):

– Розуміння: Python – це універсальна мова програмування для аналізу даних і моделювання.

– Деталі: бібліотеки NumPy і Pandas дозволяють маніпулювати даними та створювати спеціальні сценарії аналізу чутливості. Ви можете запускати моделювання за методом Монте-Карло, діаграми торнадо та односторонній аналіз чутливості. Візуалізуйте результати за допомогою Matplotlib. Наприклад, у моделі ланцюга постачання змінюйте прогнози попиту та спостерігайте за впливом на рівні запасів.

– Приклад: припустімо, ви оптимізуєте бюджет маркетингової кампанії. Змінюючи розподіл між каналами (наприклад, соціальні мережі, телебачення, друковані видання), ви можете оцінити чутливість кампанії до різних сценаріїв витрат.

4. Кришталева куля (Оракул):

– Інсайт: Crystal Ball – це надбудова Excel, спеціально розроблена для моделювання за методом Монте-Карло.

– Деталі: він повністю інтегрується з Excel і надає такі розширені функції, як кореляційне моделювання, прогнозування часових рядів і аналіз сценаріїв. Ви можете визначити вхідні розподіли, кореляції та припущення. Для інвестицій у нерухомість моделюйте дохід від оренди, підвищення вартості власності та відсоткові ставки, щоб оцінити чутливість проекту.

– Приклад: аналізуючи фармацевтичний науково-дослідний проект, ви можете змоделювати показники успіху клінічних випробувань, витрати на

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

розробку та дати закінчення терміну дії патенту, щоб оцінити чисту поточну вартість проекту.

5. R (чутливість, ggplot2):

– Insight: R є потужною мовою статистичного програмування.

– Деталі: Пакет 'sensitivity' в R пропонує інструменти для глобального та локального аналізу чутливості. Ви можете досліджувати важливість параметрів, будувати діаграми торнадо та виконувати аналіз чутливості на основі дисперсії (наприклад, індекси Соболя). Використовуйте ggplot2 для візуалізації. Для кліматичної моделі оцініть чутливість прогнозів температури до сценаріїв викидів парникових газів.

– Приклад: під час оптимізації фінансового портфеля змінюйте ваги активів і оцініть вплив на доходи з поправкою на ризик за допомогою R.

Пам'ятайте, що вибір інструменту залежить від конкретного контексту, наявності даних і досвіду. Аналіз чутливості покращує процес прийняття рішень, виявляючи критичні фактори та керуючи управлінням ризиками.

Найкращі практики для виконання кластерного аналізу

Кластерний аналіз

1. Підготовка та дослідження даних:

– Зрозумійте свої дані: перш ніж занурюватися в кластерний аналіз, уважно вивчіть свій набір даних. Враховуйте його розмір, розмірність і характер змінних (безперервні, категоричні або змішані). Візуалізуйте розподіли, кореляції та відсутні значення.

– Вибір функцій і масштабування: виберіть відповідні функції для кластеризації. видалити нерелевантні або зайві змінні. Стандартизуйте або нормалізуйте функції, щоб забезпечити однакову важливість під час обчислення відстані.

– Обробка відсутніх даних: приписуйте відсутні значення за допомогою відповідних методів (середнє значення, медіана або вдосконалені методи). Відсутні дані можуть суттєво вплинути на результати кластеру.

– Виявлення викидів: ідентифікуйте та обробляйте викиди. Викиди можуть спотворювати межі кластера та впливати на центроїди.

2. Вибір правильної метрики відстані:

– Евклідова відстань: зазвичай використовується для безперервних змінних. Вимірює прямолінійну відстань між точками.

– Відстань Манхеттен (квартал міста): підходить для категорійних або порядкових даних. Враховує лише горизонтальні та вертикальні рухи.

– Косинусне подібність: корисно для тексту або даних великої розмірності. Вимірює подібність кутів між векторами.

– Вибір метрики: враховуйте природу даних і контекст проблеми. Експериментуйте з різними показниками, щоб знайти найбільш відповідний.

3. Вибір кількості кластерів:

– Метод ліктя: побудуйте графік суми квадратів у кластері (WCSS) проти кількості кластерів. Шукайте точку «ліктя», де зниження WCSS сповільнюється.

– Оцінка силуету: виміряйте, наскільки добре кожна точка даних відповідає власному кластеру порівняно з іншими кластерами. Вищі показники силуету вказують на краще розділення.

– Знання предметної області: іноді експертиза предметної області може керувати вибором кластерів.

4. Вибір алгоритму:

– K-Means: популярний і ефективний. Добре працює для сферичних кластерів. Розумно ініціалізуйте центроїди, щоб уникнути локальних мінімумів.

– Ієрархічна кластеризація: будує деревоподібну структуру кластерів. Агломеративний (знизу вгору) або роздільний (зверху вниз) підходи.

– DBSCAN: Алгоритм на основі щільності. Автоматично виявляє кластери різної форми та обробляє шум.

– Моделі суміші Гауса (GMM): ймовірнісний підхід. Припускає, що дані надходять із суміші розподілів Гауса.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

– Порівняння алгоритмів: експериментуйте з різними алгоритмами та оцінюйте їх продуктивність за допомогою внутрішніх (наприклад, оцінка силуету) і зовнішніх (наприклад, мітки істинності землі) метрик перевірки.

5. Інтерпретація результатів:

– Профілі кластерів: Вивчіть характеристики кожного кластера. Що їх визначає? Чи мають вони значення?

– Візуалізації: використовуйте діаграми розсіювання, теплові карти або графіки паралельних координат для візуалізації кластерів.

– Перевірка: оцінка якості кластерів за допомогою зовнішньої перевірки (скоригований індекс Ренда, індекс Фаулкса-Маллоуза) або внутрішньої перевірки (оцінка силуету, індекс Дейвіса-Болдіна).

6. Приклад реального світу:

– Уявіть собі аналіз даних клієнтів для платформи електронної комерції. Кластеризація допомагає ідентифікувати такі сегменти, як «Ентузіасти, які багато витрачають на технології», « Покупці, які час від часу» та «Шукачі знижок». Маркетингові стратегії можна адаптувати відповідним чином.

Пам'ятайте, що кластерний аналіз – це і мистецтво, і наука. Експериментуйте, повторюйте та підтверджуйте свої результати. Дотримуючись цих передових методів, ви демістифікуєте процес і розкриєте цінну інформацію зі своїх даних.

Етапи виконання кластерного аналізу

Кластерний аналіз – це потужна техніка, яка використовується в різних сферах для виявлення подібностей і групування точок даних на основі їхніх характеристик. Використовуючи інструменти кількісного аналізу, кластерний аналіз дозволяє нам виявляти приховані закономірності та отримувати цінну інформацію зі складних наборів даних. Незалежно від того, чи є ви фахівцем із обробки даних, дослідником ринку чи просто зацікавленим в дослідженні закономірностей у своїх даних, розуміння кроків, пов'язаних із виконанням кластерного аналізу, може бути надзвичайно корисним.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

1. Визначте мету: перш ніж занурюватися в будь-який аналіз, дуже важливо чітко визначити мету дослідження. Чого ви намагаєтеся досягти за допомогою кластерного аналізу? Хочете сегментувати клієнтів на основі їх купівельної поведінки чи класифікувати документи на основі їх вмісту? Визначення мети керуватиме подальшими кроками та гарантуватиме, що аналіз узгоджується з вашими цілями.

2. Підготовка даних. Якість ваших результатів значною мірою залежить від якості ваших даних. Тому важливо попередньо обробити та очистити набір даних перед виконанням кластерного аналізу. Це може включати обробку відсутніх значень, нормалізацію змінних або перетворення спотворених розподілів. Забезпечуючи цілісність даних, ви можете мінімізувати зміщення та отримати точніші кластери.

3. Вибір змінних: Кластерний аналіз вимагає вибору відповідних змінних, які відображають суть подібності між спостереженнями. Вибір відповідних змінних залежить від характеру вашого набору даних і питання дослідження. Наприклад, якщо ви прагнете об'єднати клієнтів на основі їх демографічних показників, такі змінні, як вік, дохід і місцезнаходження, будуть актуальними. Важливо знайти баланс між включенням достатньої кількості змінних для значущої кластеризації та уникненням надмірного шуму.

4. Вибір міри відстані: міра відстані кількісно визначає різницю між двома спостереженнями в наборі даних. Зазвичай використовувані міри відстані включають евклідову відстань, манхеттенську відстань і косинусну подібність. Вибір вимірювання відстані залежить від типу даних і основних припущень вашого аналізу. Наприклад, евклідова відстань підходить для неперервних змінних, тоді як косинус подібність часто використовується для текстової кластеризації.

5. Вибір алгоритму кластеризації. Існує багато алгоритмів кластеризації, кожен зі своїми сильними та слабкими сторонами. Популярні алгоритми включають k-середні, ієрархічну кластеризацію та DBSCAN. Вибір відповідного

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

алгоритму залежить від таких факторів, як розмір набору даних, бажана кількість кластерів і форма кластерів, які ви очікуєте знайти.

Впровадження інструментів і програмного забезпечення для тестування А/В

Впровадження та тестування

Одним із найважливіших етапів А/В-тестування є вибір правильних інструментів і програмного забезпечення для проведення експериментів. На ринку доступно багато варіантів, кожен зі своїми особливостями, перевагами та недоліками. У цьому розділі ми вивчимо деякі фактори, які слід враховувати, вибираючи інструмент або програмне забезпечення для тестування А/В для своїх РРС-оголошень і цільових сторінок. Ми також розглянемо деякі з найпопулярніших і широко використовуваних платформ і те, як вони можуть допомогти вам досягти ваших цілей тестування.

Ось деякі з аспектів, на які слід звернути увагу під час оцінки інструменту або програмного забезпечення для тестування А/В:

1. Простота використання: вам потрібен інструмент, який легко налаштувати, використовувати та керувати ним. Ви не хочете витратити надто багато часу та зусиль на технічні деталі чи усунення несправностей. Вам потрібен інструмент, який має зручний інтерфейс, чіткі інструкції та корисну підтримку. Вам також потрібен інструмент, який добре інтегрується з вашими існуючими платформами, такими як веб-сайт, аналітика, CRM тощо.

2. Функціональність: вам потрібен інструмент, який зможе задовольнити ваші потреби та вимоги до тестування. Вам потрібен інструмент, який може створювати та запускати різні типи тестів, наприклад спліт-тести, багатоваріантні тести, тести персоналізації тощо. Вам потрібен інструмент, який може тестувати різні елементи ваших РРС-оголошень і цільових сторінок, як-от заголовки, зображення, копія, кольори, кнопки, форми тощо. Вам потрібен інструмент, який може вимірювати та звітувати про ключові показники та результати ваших тестів, наприклад конверсії, дохід, ROI тощо.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

3. Надійність: вам потрібен надійний і точний інструмент. Вам потрібен інструмент, який може обробляти великі обсяги трафіку та даних без шкоди для якості чи дійсності ваших тестів. Вам потрібен інструмент, який може гарантувати, що ваші тести є статистично значущими та що на ваші результати не впливають зовнішні фактори, такі як сезонність, випадкові коливання або зміни поведінки користувачів. Вам потрібен інструмент, який може захистити ваші дані та конфіденційність і відповідати відповідним законам і нормам.

4. Вартість: вам потрібен інструмент, який відповідає вашому бюджету та пропонує хороше співвідношення ціни та якості. Вам потрібен інструмент із прозорою та гнучкою моделлю ціноутворення, яка відповідає вашим потребам і частоті тестування. Вам потрібен інструмент, який не має прихованих комісій чи зборів і пропонує безкоштовну пробну версію або гарантію повернення грошей. Вам потрібен інструмент, який має позитивну рентабельність інвестицій і який може допомогти вам збільшити конверсії та дохід.

Нижче наведено кілька прикладів інструментів і програмного забезпечення для тестування А/В, які можна використовувати для своїх РРС-оголошень і цільових сторінок:

– Google Optimize: це безкоштовний інструмент від Google, який дозволяє створювати та запускати тести А/В, багатоваріантні тести та тести персоналізації на вашому веб-сайті. Він інтегрується з Google Analytics і Google Ads, тому ви можете легко відстежувати та вимірювати результати тестування та продуктивність. Він також має візуальний редактор, який дозволяє вносити зміни на ваші веб-сторінки без кодування.

– Optimizely: це провідна платформа для А/В-тестування та експериментів, яка пропонує низку функцій і рішень для різних галузей і випадків використання. Це дозволяє тестувати та оптимізувати ваші РРС-оголошення, цільові сторінки, веб-сторінки, мобільні програми, кампанії електронною поштою тощо. Він має потужний механізм аналітики, який надає інформацію в режимі реального часу та

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

дієві рекомендації. Він також має надійну систему безпеки та відповідності, яка гарантує захист ваших даних і конфіденційності.

– VWO: це комплексна платформа для A/B-тестування та оптимізації коефіцієнта конверсії, яка допомагає створювати та проводити експерименти на основі даних на вашому веб-сайті та цільових сторінках. Це дозволяє тестувати та покращувати різні аспекти вашого досвіду користувача, такі як дизайн, вміст, функціональність і персоналізація. Він має розумний та інтуїтивно зрозумілий інтерфейс, який полегшує використання та керування тестами. Він також має набір інструментів, які допоможуть вам проаналізувати та зрозуміти вашу поведінку та відгуки користувачів.

Використання фінансових інструментів і програмного забезпечення для аналізу витрат

У епоху цифрових технологій компанії мають доступ до широкого спектру фінансових інструментів і програмного забезпечення, які можуть оптимізувати процес аналізу витрат. Ці інструменти можуть допомогти підприємствам відстежувати витрати, аналізувати тенденції витрат і визначати сфери потенційної економії.

Одним із популярних інструментів для аналізу витрат є програмне забезпечення для бюджетування та прогнозування. Це програмне забезпечення дозволяє підприємствам створювати детальні бюджети, відстежувати фактичні витрати та порівнювати їх із запланованими сумами. Маючи дані про витрати в режимі реального часу, підприємства можуть визначати будь-які відхилення та негайно вживати коригувальних заходів. Крім того, ці інструменти часто надають можливості прогнозування, що дозволяє підприємствам прогнозувати майбутні витрати та приймати обґрунтовані рішення на основі очікуваних витрат.

Іншим корисним інструментом для аналізу витрат є програмне забезпечення для керування витратами. Це програмне забезпечення автоматизує процес відстеження витрат, полегшуючи для компаній моніторинг і контроль витрат. Він може отримувати дані про витрати з різних джерел, таких як кредитні

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

картки та квитанції, і надавати докладні звіти для аналізу. Використовуючи програмне забезпечення для керування витратами, підприємства можуть зменшити адміністративний тягар відстеження витрат і отримати цінну інформацію про моделі витрат.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Технічним завданням магістерського проекту передбачена розробка ПЗ моніторингу роботи соціального сервісу з використанням методів кластерного аналізу. Розробити необхідно систему, яку можна буде використовувати на різних веб-переглядачах і пристроях з різним розширенням екрану, тобто – багатофункціональну і адаптивну.

PHP (англ. PHP: Hypertext Preprocessor – PHP: гіпертекстовий препроцесор), попередня назва: Personal Home Page Tools – скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. PHP – проект відкритого програмного забезпечення.

PHP інтерпретується веб-сервером у HTML-код, який передається на сторону клієнта. На відміну від скриптової мови JavaScript, користувач не бачить PHP-коду, бо браузер отримує готовий html-код. Це є перевага з точки зору безпеки, але погіршує інтерактивність сторінок. Але ніщо не забороняє використовувати PHP для генерування і JavaScript-кодів які виконуються вже на стороні клієнта. PHP – мова, код якої можна вбудовувати безпосередньо в html-код сторінок, які, у свою чергу, будуть коректно оброблені PHP-інтерпретатором. Обробник PHP просто починає виконувати код після відкриваючого тегу (<?php) і продовжує виконання до того моменту, поки не зустрине закриваючий тег (?>).

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

Велика різноманітність функцій PHP дає можливість уникати написання багаторядкових функцій, призначених для користувача, як це відбувається в C або Pascal.

В PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase.

Мова PHP здаватиметься знайомою програмістам, що працюють в різних областях. Багато конструкцій мови запозичені з C, Perl. Код PHP дуже схожий на той, який зустрічається в типових програмах на C або Pascal. Це помітно знижує початкові зусилля при вивченні PHP. PHP – мова, що поєднує переваги Perl і C і спеціально спрямована на роботу в Інтернеті, мова з універсальним і зрозумілим синтаксисом. І хоча PHP є досить молодого мовою, вона здобула таку популярність серед web-програмістів, що в наш час є мало не найпопулярнішою мовою для створення веб-застосунків (скриптів). [3]

Ефективність є дуже важливим чинником у програмуванні для середовищ розрахованих на багато користувачів, до яких належить і web. Важливою перевагою PHP є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl. Проте хоч би що робили розробники PHP, виконавчі файли, отримані за допомогою компіляції, працюватимуть значно швидше – в десятки, а іноді і в сотні разів. Але продуктивність PHP достатня для створення цілком серйозних веб-застосунків.

Yii2 – це високопродуктивний та швидкодіючий веб-фреймворк, написаний на PHP, реалізує парадигму модель-вид-контролер. Yii – скорочення від «Yes It Is!».

Головними перевагами і можливостями фреймворку є:

- Реалізація архітектурного шаблону Модель-вид-контролер.
- Інтерфейси DAO та Active Record для роботи з базами даних.
- Підтримка інтернаціоналізації.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

- Кешування сторінок та окремих фрагментів.
- Перехоплення та обробка помилок.
- Введення та валідація веб-форм.
- Генерація базового PHP-коду для CRUD-операцій.
- Використання AJAX та інтеграція з jQuery.
- Міграції бази даних.
- Можливість підключення сторонніх бібліотек[6].

Спільнота розробників використовуючих фреймворк активно розвивається та розроблює все більше нових корисних компонентів, які легко можна підключити і використовувати при роботі з фреймворком.

Перевага Yii над іншими фреймворками полягає у ефективності, широких можливостях та якісній документації. Yii з самого початку дуже ретельно спроектований для відповідності всім вимогам при розробці серйозних веб-додатків. Дякуючи своїй легковагості та наявності просунутих засобів кешування, Yii особливо підходить для розробки додатків із великим потоком трафіку, таких як портали, форуми, системи управління контентом (CMS), системи електронної комерції та ін.

Подібно більшості інших PHP-фреймворків, Yii – це MVC-фреймворк.

MVC (або Модель-вид-контролер, англ. Model-view-controller, MVC) – архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення. Цей шаблон поділяє систему на три частини: модель даних, вигляд даних та керування.

Мета шаблону – гнучкий дизайн програмного забезпечення, який повинен полегшувати подальші зміни чи розширення програм, а також надавати можливість повторного використання окремих компонентів програми. Крім того використання цього шаблону у великих системах призводить до певної впорядкованості їх структури і робить їх зрозумілішими завдяки зменшенню складності.

Модель (model) – це екземпляр класу CModel або класу, успадкованого від нього. Модель використовується для зберігання даних або застосовних до них бізнес-правил.

Представлення – це PHP-скрипт, який складається переважно з елементів користувацького інтерфейсу. Він може охоплювати вирази PHP, проте рекомендується, щоб ці вирази не змінювали дані і залишалися відносно простими. Згідно концепції розділення логіки та представлення, більша частина коду логіки повинна бути розміщена у контролері або моделі, а не у скрипті представлення.

Контролер (controller) – це екземпляр класу CController або успадкованого від нього класу. Він створюється об'єктом додатку тоді, коли користувач робить відповідний запит. При запуску контролер виконує відповідну дію, що зазвичай передбачає створення відповідних моделей і рендеринг необхідних представлень. У найпростішому випадку дія – це метод класу контролера, назва якого починається на action.

Модель інкапсулює ядро даних і основний функціонал з їх обробки. Також компонент Модель не залежить від процесу введення або виведення даних. Компонент виводу Представлення може мати декілька взаємопов'язаних областей, наприклад, різні таблиці і поля форм, в яких відображається інформація. У функції Контролера входить моніторинг за подіями, що виникають в результаті дій користувача (зміна положення курсора миші, натиснення кнопки або введення даних в текстове поле).

Зареєстровані події транслюються в різні запити, що спрямовуються компонентам Моделі або об'єктам, відповідальним за відображення даних. Відокремлення моделі від вигляду даних дозволяє незалежно використовувати різні компоненти для відображення інформації. Таким чином, якщо користувач через Контролер внесе зміни до Моделі даних, то інформація, подана одним або декількома візуальними компонентами, буде автоматично відкорегована відповідно до змін, що відбулися.[4]

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

Для розробки програмного забезпечення було обрано середовище розробки JetBrains PhpStorm – крос-платформове інтегроване середовище розробки для PHP, яке розробляється компанією JetBrains на основі платформи IntelliJ IDEA.

PhpStorm являє собою інтелектуальний редактор для PHP, HTML і JavaScript з можливостями аналізу коду на льоту, запобігання помилок у сирцевому коді і автоматизованими засобами рефакторинга для PHP і JavaScript. Автодоповнення коду в PhpStorm підтримує специфікацію PHP 5.3, 5.4 та 5.5 (сучасні і традиційні проекти), включаючи генератори, співпрограми, простори імен, замикання, типажі і синтаксис коротких масивів. Присутній повноцінний SQL-редактор з можливістю редагування отриманих результатів запитів.

PhpStorm розроблений на основі платформи IntelliJ IDEA, написаної на Java. Користувачі можуть розширити функціональність середовища розробки за рахунок установки плагінів, розроблених для платформи IntelliJ, або написавши власні плагіни.

PhpStorm надає багатий і інтелектуальний редактор коду для PHP з підсвічуванням коду, розширеною конфігурацією форматування коду, перевіркою наявності помилок на льоту і розумним автодоповненням.

Основними перевагами PhpStorm є:

- Підтримка PHP 5.3, 5.4 та 5.5, включаючи генератори, співпрограми, простори імен, замикання, типажі, синтаксис коротких масивів, доступ до члена класу при інстанціюванні, розіменування масиву при виклику функції, бінарні літерали, вираження в статичних виклики тощо. PhpStorm може використовуватися як для сучасних, так і для традиційних проектів на PHP.
- Автодоповнення коду фіналізують класи, методи, імена змінних, ключові слова PHP, а також широко використовувані імена полів і змінних залежно від їхнього типу.
- Підтримка стандартів оформлення коду (PSR1/PSR2, Drupal, Symfony2, Zend).

- Підтримка PHPDoc. PhpStorm надає відповідне автодоповнення коду, засноване на анотаціях @property, @method і @var.
- Детектор дубльованого коду.
- PHP Code Sniffer (phpcs), котрий перевіряє код на льоту
- Рефакторинги (перейменування, введення змінної/константи/поля, вбудовування змінної).
- Підтримка редагування шаблонів Smarty (підсвічування синтаксичних помилок, автодоповнення функцій і атрибутів Smarty, автоматична вставка парних дужок, лапок і закриваючих тегів тощо)
- MVC подання для фреймворків Symfony2 і Yii
- Розпізнавання коду, запакованого в PHAR-архіви.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускні кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

КБПЗ_2024

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Технічним завданням магістерського проекту передбачена розробка програмного забезпечення моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Кластерним підходом до виявлення структури в базах даних без міток є кластерний аналіз. У кластерному аналізі передбачається, що випадки або одиниці можуть бути згруповані у значущі групи, які називаються «кластерами», на основі даних спостереження. Кластерний аналіз використовувався в природничих науках у ранніх спробах класифікації видів і отримав назву чисельної таксономії.

Неієрархічний

Основною формою кластерного аналізу є неієрархічний. У неієрархічному кластерному аналізі ми просто намагаємося сформувати групи, але немає порядку між групами. Наприклад, при порівнянні загальних показників смертності та народжуваності для країн світу часто виділяються три групи: країни з високим рівнем народжуваності та високим рівнем смертності (країни, що розвиваються, позначені «1»), країни з високим рівнем народжуваності і низькі загальні показники смертності (країни, що проходять через «демографічний перехід», позначені «3»), і країни з низьким загальним рівнем народжуваності та зростаючими загальними показниками смертності (розвинені країни, позначені «2»; зростання загального рівня смертності спричинено старіння населення).

Ієрархічний

На відміну від неієрархічного кластерного аналізу, ієрархічний кластерний аналіз формує кластери ітеративно, шляхом послідовного приєднання або розбиття груп. Є два типи: розбіжний, який починається з усього набору даних в

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

одній великій групі, а потім послідовно розбиває його на менші групи, поки кожне спостереження не стане окремою групою; і агломераційний, у якому кожне спостереження починається з власної групи, а групи послідовно об'єднуються в пари, поки в кінці кожне спостереження не буде в одній великій групі. Розділові методи потребують обчислень і мають обмежене застосування в соціальних науках. Агломераційні методи реалізовані в багатьох стандартних програмних пакетах.

В ієрархічному кластерному аналізі виникає проблема, коли два (або більше) спостереження поміщено в групу: якщо я порівнюю нове спостереження з групою, чи вибираю спостереження (у групі), яке найближче до мого нового спостереження., чи я вибираю спостереження (у групі), яке є найдалшим від мого спостереження, чи я вибираю якусь середню точку (скажімо, середнє значення) для порівняння з моїм новим спостереженням? Перший варіант відомий як одиночне з'єднання, другий – як повне з'єднання, а третій – як середнє з'єднання.

Під час проходження передмагістерської практики був проведений аналіз повсякденного використання користувачами подібних веб-сервісів, який показав, що такі програми найчастіше використовуються в тих випадках, коли потрібно вирішити якусь проблему і отримати компетентні відгуки та інформацію від експертів в даній сфері.

Для того щоб реалізувати поставлену в магістерській роботі задачу потрібно розглянути структурну та функціональну схеми, а також діаграму процесів, які дадуть наочне уявлення про принципи функціонування вузлів програмного забезпечення та взаємодію між ними.

Веб-сервіс, що підлягає розробці, повинен виконувати всі вищеназвані функції. Розроблене ПЗ повинне бути компактним і зручним у користуванні. Це планується досягти за рахунок використання Yii 2 і всіх його прикладних засобів.

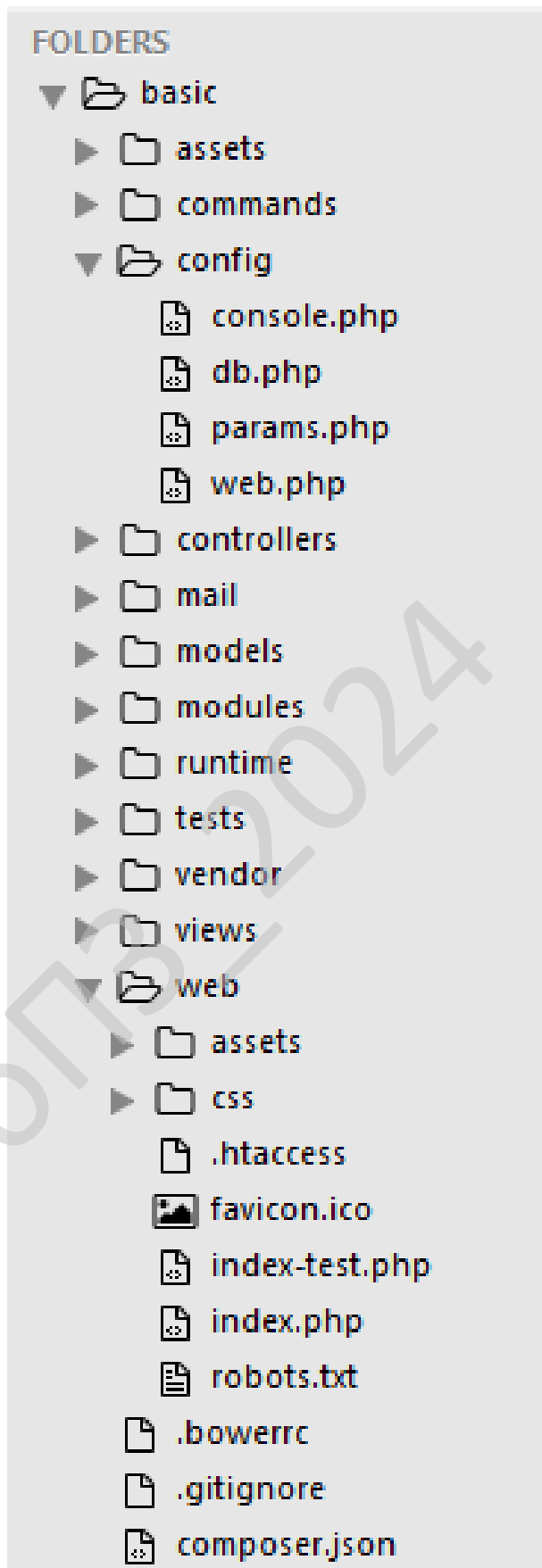


Рисунок 3.1 – Структура проекту в Yii2

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

Ієрархічна багаторівнева побудова системи дозволить подолати складність розв'язання проблеми масштабування системи при подальшій розробці. Структура проекту в Yii 2 зображена на рисунку 3.1

Система, що розробляється, буде побудована використовуючи шаблон проектування Модель-Представлення-Контролер (MVC, Model-View-Controller), який широко використовується в веб-програмуванні.

MVC спрямований на відділення бізнес-логіки від користувальницького інтерфейсу, надає можливість легко змінювати окремі частини додатку, не вносячи змін до інших. Моделі представляють структури даних, тобто таблиці у БД, які використовуються у додатку. Ці класи дозволяють працювати з БД у стилі парадигми програмування ООП. В розроблюваному програмному забезпеченні міститься 4 таблиці (де зберігаються дані про користувачів, запитання, коментарі, теги тощо).

Загалом функціонування системи має 2 основні складові: введення, зміна, зберігання та виведення інформації (різноманітні дані про користувачів, запитання, коментарі, теги) та, власне, механізм групування результатів при пошуку інформації, надаючи користувачу можливість швидкої навігації по результатам пошукової видачі.

Всі дані зосереджено в таблицях бази даних, які пов'язані між собою та об'єднані в єдину систему. Робота з базою даних реалізована за допомогою СУБД MySQL та вбудованих в Yii інструментів для роботи з БД.

Yii Active Record (AR) реалізує перероблений підхід об'єктно-реляційного відображення (ORM) і спрощує роботу з базами даних. Представляючи таблицю бази даних як клас, а рядки таблиці як екземпляри класу, Yii AR позбавляє від необхідності написання SQL-виразів, пов'язаних з операціями CRUD (створення, читання, оновлення та видалення). В результаті, можна працювати із більшою об'єктно-орієнтованістю. Наприклад, наступний код реалізує збереження даних в БД під час процедури реєстрації.

```
public function signup(){  
    $user = new User();
```

```

$user->email = $this->email;
$user->setPassword($this->password);
return $user->save();
}

```

Оперування та перегляд даних здійснюється за допомогою інтерфейсу користувача у вигляді форм, кнопок та полів пошуку. Реалізовано також функцію пошуку за різними критеріями. Результати пошукової видачі зображаються у вигляді категорій, які формуються в результаті використання методів кластеризації, а саме алгоритму k-means.

3.2 Розробка структурної схеми

Кластерний (або класифікаційний) аналіз упорядковує місця, види або змінні в групи (кластери), у яких подібність усередині групи є максимальною, а між групами – мінімізованою. Кластерний аналіз є явним способом пошуку структури в даних (наприклад, визначення регіональної структури). Кластерний аналіз накладає групову структуру, навіть якщо існує безперервна структура даних (наприклад, відповідь уздовж градієнта). Кластерний аналіз може бути особливо корисним при розробці схем регіональної класифікації (блокування) об'єктів і встановленні еталонних умов (Gerritsen et al., 2000).

Кластерний аналіз (класифікація) зазвичай використовується для:

1. Надати інформацію про збіг видів (внутрішня структура даних).
2. Встановити типи спільнот для описових досліджень (синтаксономія та відображення).
3. Виявлення зв'язків між спільнотами та середовищем шляхом аналізу груп, сформованих кластерним аналізом щодо змінних середовища (зовнішній аналіз).

Кластерний аналіз може бути розділовим або агломеративним методом. Методи розділення починаються з усіх об'єктів в одній групі, а потім послідовно ділять кожну групу на дві менші групи, доки не буде виконано якесь «правило зупинки». Агломераційні методи починаються з окремих об'єктів, які

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

об'єднуються в групи шляхом колекції об'єктів або групуються в більші групи, поки всі об'єкти не потраплять в одну групу. Кластеризація також може бути ієрархічною або неієрархічною. Методи ієрархічної кластеризації впорядковують групи в ієрархічну систему, яка визначає зв'язки (подібності) між групами (рис. 13). Неієрархічні методи поділяють дані на групи без визначення зв'язків між групами.

Статистичну значущість кластерів можна оцінити за допомогою таких методів, як дискримінантний аналіз, SIMPROF, ANOSIM (Clarke and Gorley, 2006) і MRPP (McCune and Mefford, 1999). Ці методи перевіряють гіпотезу про відсутність різниці між двома або більше групами об'єктів. SIMPROF, ANOSIM і MRPP особливо підходять для аналізу даних спільноти, оскільки вони є непараметричними методами, які не припускають, що дані є багатовимірними нормальними або дисперсіями однорідними. Усі ці методи вимагають, щоб групи (кластери) були визначені апріорі. Отже, ці методи не слід використовувати для перевірки значущості класифікації, отриманої з кластерного аналізу. Статистичний аналіз апріорних кластерів та аналіз видів-індикаторів може бути корисним у розробці регіональних еталонних ділянок та/або для блокування дизайну (тобто визначення груп ділянок, які мають подібні спільноти) програми моніторингу. Аналіз видів-індикаторів (Dufrene and Legendre, 1997; McCune and Mefford, 1999) можна використовувати для ідентифікації видів, характерних для конкретних кластерів. Для перевірки стабільності кластерного аналізу були розроблені такі методології, як «відбійний ніж», тобто ці методології оцінюють, чи призводить підхід до кластеризації до статистично значущих кластерів, які є економними і на які відносно не впливає видалення сайту з аналізу (Omernik, 2004).).

Прикладами алгоритмів кластерного аналізу є:

- Ієрархічні алгоритми.
- Алгоритм К-середніх.
- Мінімальне покриваюче дерево.

- Метод найближчого сусіда.
- Алгоритми нечіткої кластеризації.
- Застосування нейронних мереж.
- Генетичні алгоритми.

В ході розробки магістерського проекту було прийняте рішення застосувати для кластеризації результатів пошуку алгоритм k-Means. Даний алгоритм простий у розробці та є доволі швидким.

Структурна схема зображена на рисунку 3.2 відображає основні вузли програмного забезпечення та їх взаємодію між собою. Веб-сервіс, що розробляється, буде побудований використовуючи шаблон проектування MVC.

Структурна схема відображає основні блоки системи, що розробляється та їх взаємодію між собою. Головний модуль являє собою головну сторінку веб-сервісу, з якого при натисненні відповідних кнопок можна виконувати різні операції. Основними структурними складовими системи, що розробляється є:

- Головна сторінка
- Реєстрація і авторизація;
- Профіль користувача (історія активності користувача та налаштування)
- Запитання (створення, редагування, коментарі, помічення тегами)
- Інформація про автора

Сформуємо та розглянемо структурну схему побудови системи, яка показана на рисунку 3.2

При розробці концепції побудови майбутньої системи та шляхів її реалізації основний наголос робився на швидкості, зрозумілості, легкості використання.



Рисунок 3.2 – Структурна схема веб-сервісу

3.3 Розробка функціональної схеми

Функціональна схема містить інформацію про способи реалізації пристроєм заданих функцій. За такою схемою можна визначити, як здійснюються перетворення і які для цього необхідні функціональні елементи. Дана схема розробляється на основі структурної схеми для кожного блоку, в результаті з окремих функціональних елементів складається загальна функціональна схема програмного забезпечення.

ПЗ, що підлягає розробці, має забезпечувати виконання всіх функцій, що закладені в системі.

Таким чином, визначивши структуру майбутньої системи та функції, які будуть виконувати її складові, маємо всі необхідні дані для побудови функціональної схеми системи, яка представлена на рисунку 3.3.

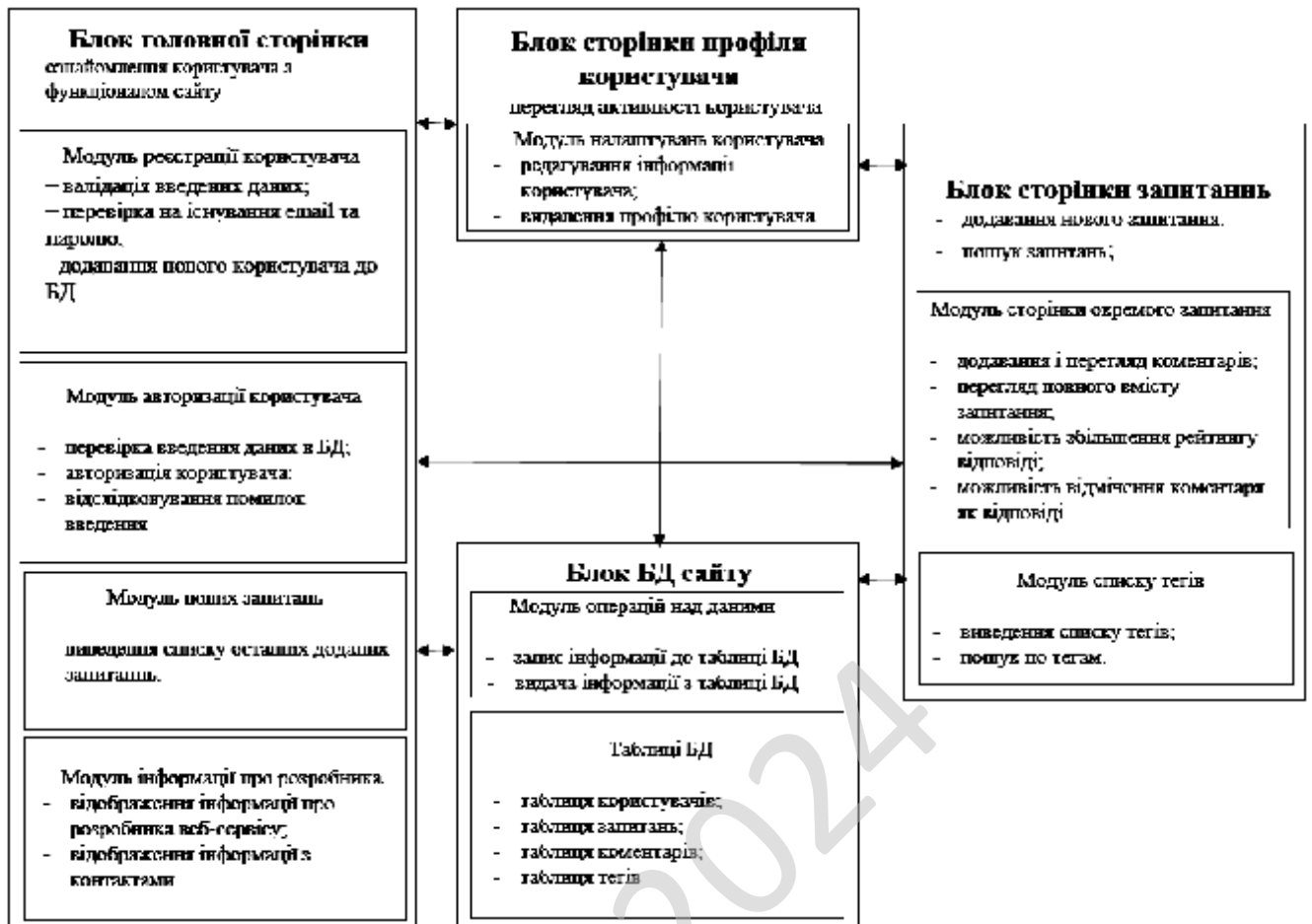


Рисунок 3.3 – Функціональна схема системи

Серед блоків схеми перераховані функції, які виконує програма, що гарно ілюструє її функціональність.

До складу головної програми входить чотири основні блоки системи:

- блок головної сторінки;
- блок сторінки профіля користувача;
- блок сторінки запитань;
- блок бази даних сайту.

Блок головної сторінки забезпечує ознайомлення користувача з функціоналом веб-сервісу.

Блок головної сторінки містить в собі такі модулі:

- модуль реєстрації користувачів;
- модуль авторизації користувача;
- модуль, який містить нові запитання;
- модуль інформації про розробника.

Модуль реєстрації користувачів має забезпечувати можливість зареєструватись. На сторінці має відображатись форма для введення email та паролю. Після того як користувач введе і вони пройдуть перевірку на коректність, інформація користувача буде збережена в БД.

Модуль авторизації користувачів повинен здійснювати процедуру встановлення належності користувачеві інформації в системі залежно від логіна і пароля користувача, а також керування рівнями та засобами доступу до веб-сервісу. Також необхідно реалізувати валідацію введених в форму даних.

Модуль інформації про розробника забезпечує виведення даних про розробника, а також форму зворотнього зв'язку. Форма зворотнього зв'язку потрібна, користувачі могли звернутись з своїми пропозиціями і скаргами до розробників.

Модуль нових запитань здійснює відображення останніх доданих запитань.

Блок сторінки профілю користувача надає можливість перегляду активності користувача, а саме: інформація про користувача, запитання, відповіді та коментарі, які він задавав, час коли користувач останній раз з'являвся на сайті та теги.

Також має бути реалізована можливість редагувати основну інформацію на сторінці свого профілю та функція видалення свого профілю.

Блок сторінки запитань складається з таких модулів:

- модуль сторінки окремого запитання;
- модуль виведення списку тегів.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

Модуль сторінки окремого запитання виконує такі функції: додавання і перегляд коментарів, перегляд повного вмісту запитання, можливість збільшення рейтингу відповіді, можливість відмічення коментарю як відповіді.

Модуль списку тегів: виведення списку тегів, пошук по тегах.

Користувацький інтерфейс повинен бути розроблений на дуже простому рівні та бути інтуїтивно зрозумілим, що спростить використання розробленого програмного забезпечення.

3.4 Розробка діаграми процесів

Для визначення ієрархії процесів в системі та їх взаємовпливу, розроблена і наводиться на рисунку 3.4 діаграма процесів.

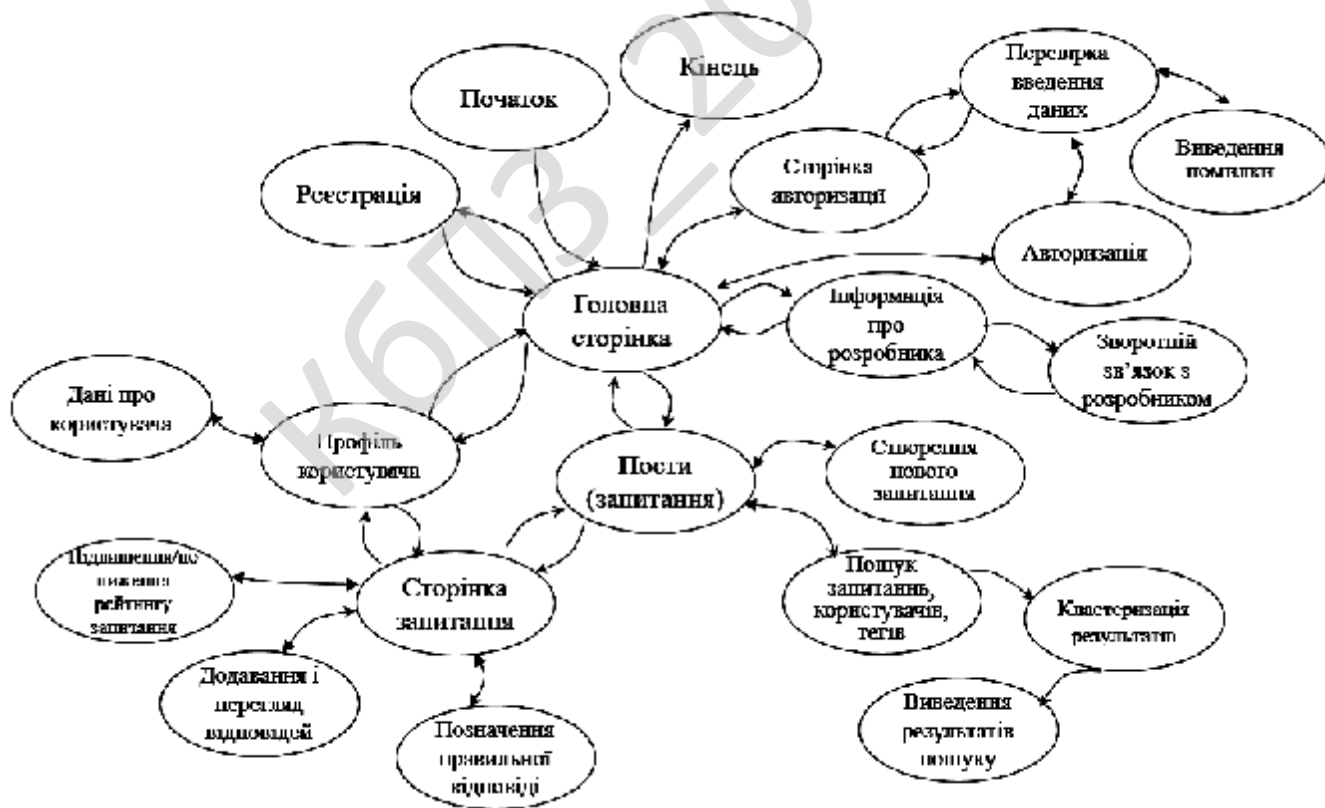


Рисунок 3.4 – Діаграма взаємодії процесів в системі

На початку роботи системи запускається головна сторінка веб-сервісу, на якій відображаються нові нещодавно додані повідомлення (запитання). Щоб задати запитання користувачеві потрібно зареєструватись або пройти авторизацію, тобто ввести email та пароль. Після перевірки введених даних, якщо все вірно користувача буде допущено в систему під своїм логіном. Якщо ж є помилки буде виведено повідомлення про помилку. Після того як користувач авторизується він отримує можливість задавати питання і відповідати на запитання інших користувачів, підвищувати та понижувати рейтинг запитань.

У темі з запитанням, користувач, що створив тему може позначити відповідь як правильну, що полегшує іншим користувачам пошук при виникненні схожої проблеми. Також користувач може здійснити пошук по вже існуючим запитанням та тегам. Результати пошуку будуть виведені по

категоріям, які утворюються завдяки використанню методів кластерного аналізу.

Таким чином, фактично була закінчена теоретична побудова структури системи, яка підлягає розробці. Визначені основні алгоритми ПЗ системи та означені шляхи їх реалізації та є всі необхідні дані для розробки програмного забезпечення системи.

Отже, щоб наочно представити принципи функціонування вузлів програмного забезпечення та взаємодію між ними було реалізовано структурну та функціональну схеми, а також діаграму процесів. Архітектура майбутньої системи побудована з використанням шаблону проектування MVC. Для розробки було обрано такі інструменти як: фреймворк Yii2, PHP 5.4, середовище розробки PHPStorm.

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

На рисунку 4.1 наведено блок-схему основної програми. Її робота складається з виконання наступних кроків.

В розділі пояснюється підхід до реалізації програмного забезпечення моніторингу роботи соціального сервісу з використанням методів кластерного аналізу, наведено алгоритми роботи програми, розглянуто принципи роботи об'єктів та їх призначення.

Програма складається з декількох типів сторінок, що підключені до головного модуля програми та викликаються з нього.

База даних програмного забезпечення, що розроблялася, складається з 4 таблиць. Структура таблиць БД зображена на рисунку 4.1.

Для встановлення з'єднання із базою даних необхідно створити екземпляр класу Connection і активувати його. Додаткова інформація необхідна для підключення до БД (хост, порт, ім'я користувача, пароль тощо) вказана у DSN. У разі виникнення помилки в процесі з'єднання з БД, буде викликано виключення (наприклад, невірний DSN або неправильні ім'я користувача/пароль).

Підключення до БД виглядає таким чином:

```
<?php
return [
    'class' => 'yii\db\Connection',
    'dsn' => 'mysql:host=localhost;dbname=database',
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',
];
```

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

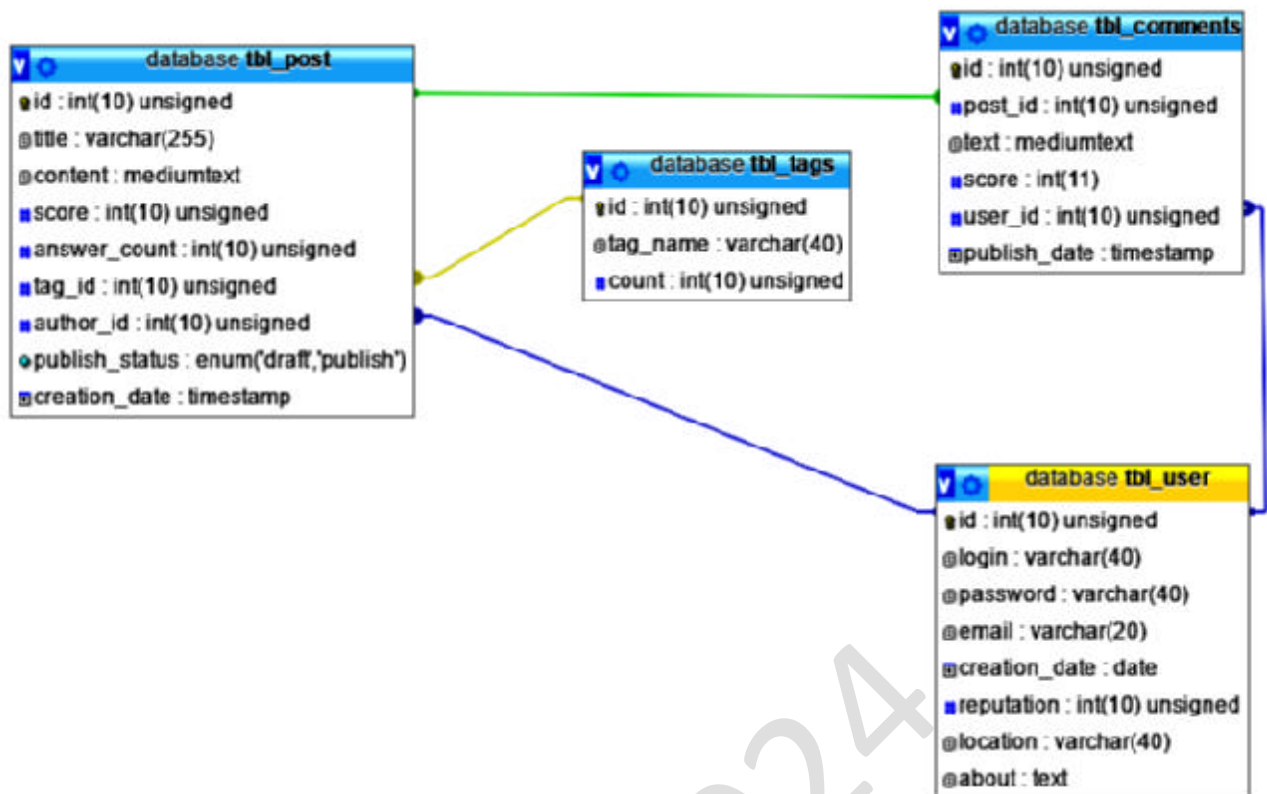


Рисунок 4.1 – Схема бази даних

Таблиця “tbl_user” відображає користувача. Вона містить у собі такі поля:

- id: унікальний ідентифікатор;
- name: ім'я, нік або назва організації зареєстрованої на сайті;
- login: логін користувача;
- password: пароль від аккаунту;
- creation_date: дата створення профіля;
- reputation: репутація;
- location: країна;
- about: інформація про користувача.

Логін та пароль є обов'язковими полями. Це контролюється методами rules у моделях User, Signup та Login, зв'язаних з користувачем. Наприклад, реалізацію методу rules моделі Login наведено нижче.

```

public function rules()
{
    return [
        [['email','password'],'required'],
        ['email','email'],
        ['password','string','min'=>2,'max'=>10],
        ['password','validatePassword']//функція валідації паролю
    ];
}

```

Нижче на рисунку 4.2 наведено блок-схему головного функціонального модуля програми, який описує основні стадії таких операцій як авторизація, реєстрація та створення нового запитання.

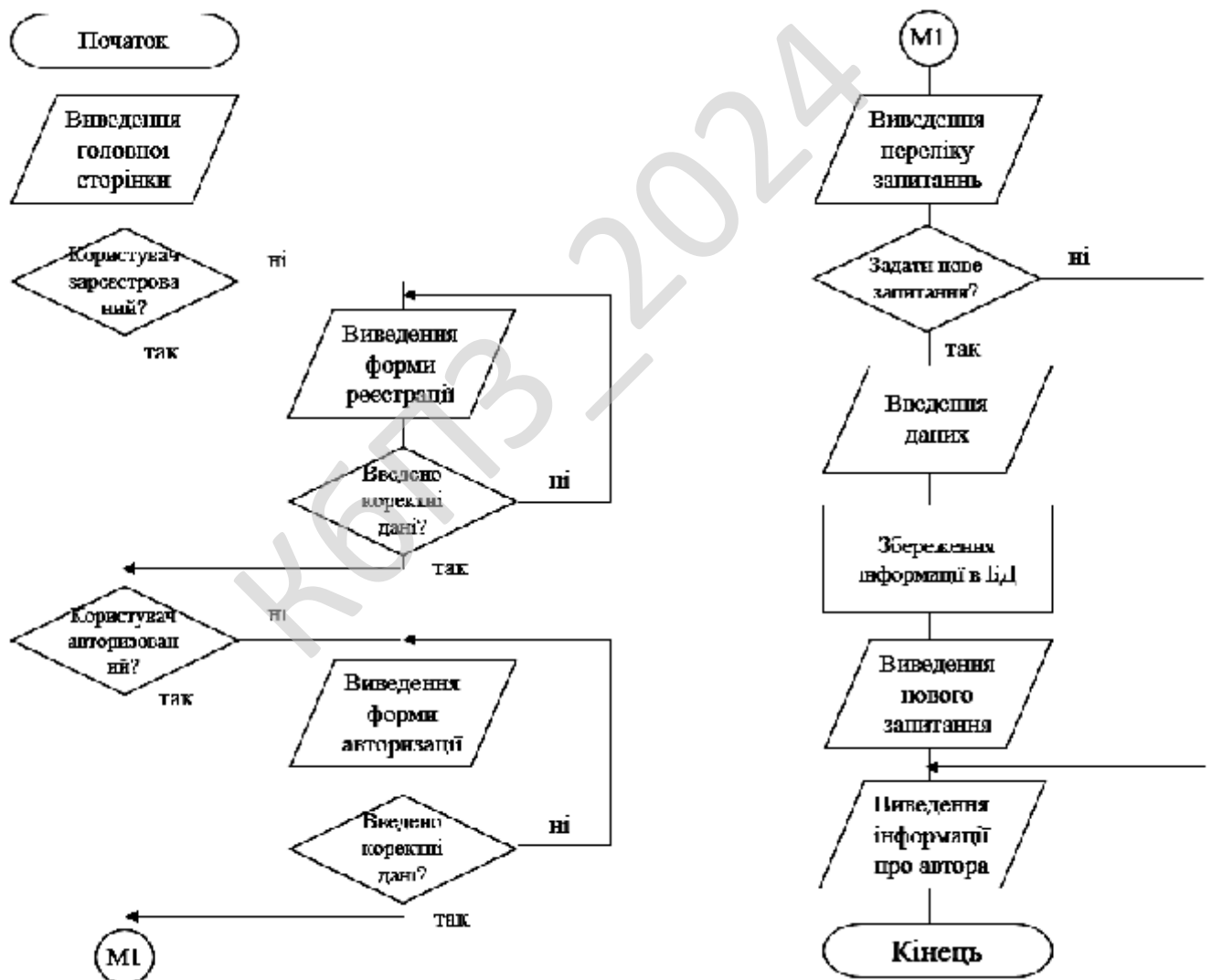


Рисунок 4.2 – Блок-схема основної програми

Під час розробки програми було створено сім класів-моделей чотири з яких розширюють клас Active Record і зв'язані з таблицями у базі даних. Це класи: Post, Comment, Tag, User. Вони представляють собою сутності постів (тем з запитанням), коментарів, тегів та користувачів відповідно. Класи Signup та Login є моделями-прошарками для класу User і створені для розділення відповідальності з метою виокремлення коду, який відповідає за автентифікацію та реєстрацію користувачів. Вони виконують функції фільтрів, валідації та запису у БД. Наприклад, метод, що відправляє по електронній пошті повідомлення на вказану адресу, використовуючи інформацію форм моделі ContactForm. виглядає наступним чином:

```
public function contact($email)
{
    if ($this->validate()) {
        Yii::$app->mailer->compose()
            ->setTo($email)
            ->setFrom([$this->email => $this->name])
            ->setSubject($this->subject)
            ->setTextBody($this->body)
            ->send();

        return true;
    }
    return false;
}
```

Форма зворотного зв'язку зображена на рисунку 4.3.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54

Зворотній зв'язок

Якщо у вас є питання, будь ласка, заповніть наступну форму, щоб зв'язатися з нами. Дякуємо!

Ім'я:

Ім'я: cannot be blank

Електронна пошта:

Тема:

Текст повідомлення:


Код верифікації: 

Рисунок 4.3 – Форма зворотного зв'язку

Всі теги:

<input type="button" value="memcache x 9"/>	<input type="button" value="arc x 9"/>	<input type="button" value="графіки x 9"/>
<input type="button" value="routes x 9"/>	<input type="button" value="com x 9"/>	<input type="button" value="microsoft x 9"/>
<input type="button" value="kde x 9"/>	<input type="button" value="gae x 9"/>	<input type="button" value="технології x 9"/>
<input type="button" value="Функції x 9"/>	<input type="button" value="jar x 9"/>	<input type="button" value="safari x 9"/>
<input type="button" value="сторінка x 9"/>	<input type="button" value="books x 9"/>	<input type="button" value="system x 9"/>
<input type="button" value="читання x 9"/>	<input type="button" value="логіка x 9"/>	<input type="button" value="сервіси x 9"/>
<input type="button" value="управління x 9"/>	<input type="button" value="обробка x 9"/>	<input type="button" value="zend_auth x 9"/>
<input type="button" value="трафік x 9"/>	<input type="button" value="localhost x 9"/>	<input type="button" value="антивірус x 9"/>
<input type="button" value="patterns x 9"/>	<input type="button" value="баннер x 9"/>	<input type="button" value="webservice x 9"/>
<input type="button" value="jqueryui x 9"/>	<input type="button" value="jplayer x 9"/>	<input type="button" value="оплата x 9"/>
<input type="button" value="webstorm x 9"/>	<input type="button" value="іконки x 9"/>	<input type="button" value="скріншот x 9"/>
<input type="button" value="pear x 9"/>	<input type="button" value="xmlhttprequest x 9"/>	<input type="button" value="multiple x 9"/>
<input type="button" value="бібліотеки x 9"/>	<input type="button" value="база-даних x 9"/>	<input type="button" value="многоязичність x 9"/>
<input type="button" value="library x 9"/>	<input type="button" value="сервіс x 9"/>	<input type="button" value="gem x 9"/>
<input type="button" value="locale x 9"/>	<input type="button" value="photo x 9"/>	<input type="button" value="локалізація x 9"/>

← Назад 6 7 8 9 10 11 12 13 14 15 Вперед →

Рисунок 4.4 – Сторінка, що відображає список тегів

На рисунку 4.4 зображено сторінку тегів. Реалізацію методу rules моделі Tag наведено нижче.

```
public function rules()  
{  
    return [  
        [['tag_name', 'count'], 'required'],  
        [['count'], 'integer'],  
        [['tag_name'], 'string', 'max' => 40],  
    ];  
}
```

Блок-схема зображена на рисунку 4.4 відображає аналіз заголовків і тексту запитань і створення кластерів, які містять запитання з подібним вмістом використовуючи методи попередньої обробки.

Основними етапами кластерного аналізу результатів пошуку є:

- отримання пошукового запиту користувача і пошук інформації;
- обробка отриманих даних;
- синтаксичний аналіз вхідних даних (токенізація);
- видалення стоп-слів;
- приведення слів до основи (стеммінг)
- застосування алгоритму виділення кластерів;
- формування кластерів і їх виведення на веб-сторінку у вигляді посилань, описаних за ключовими словами, що містять набір пошукових результатів, що відносяться до даного кластеру.

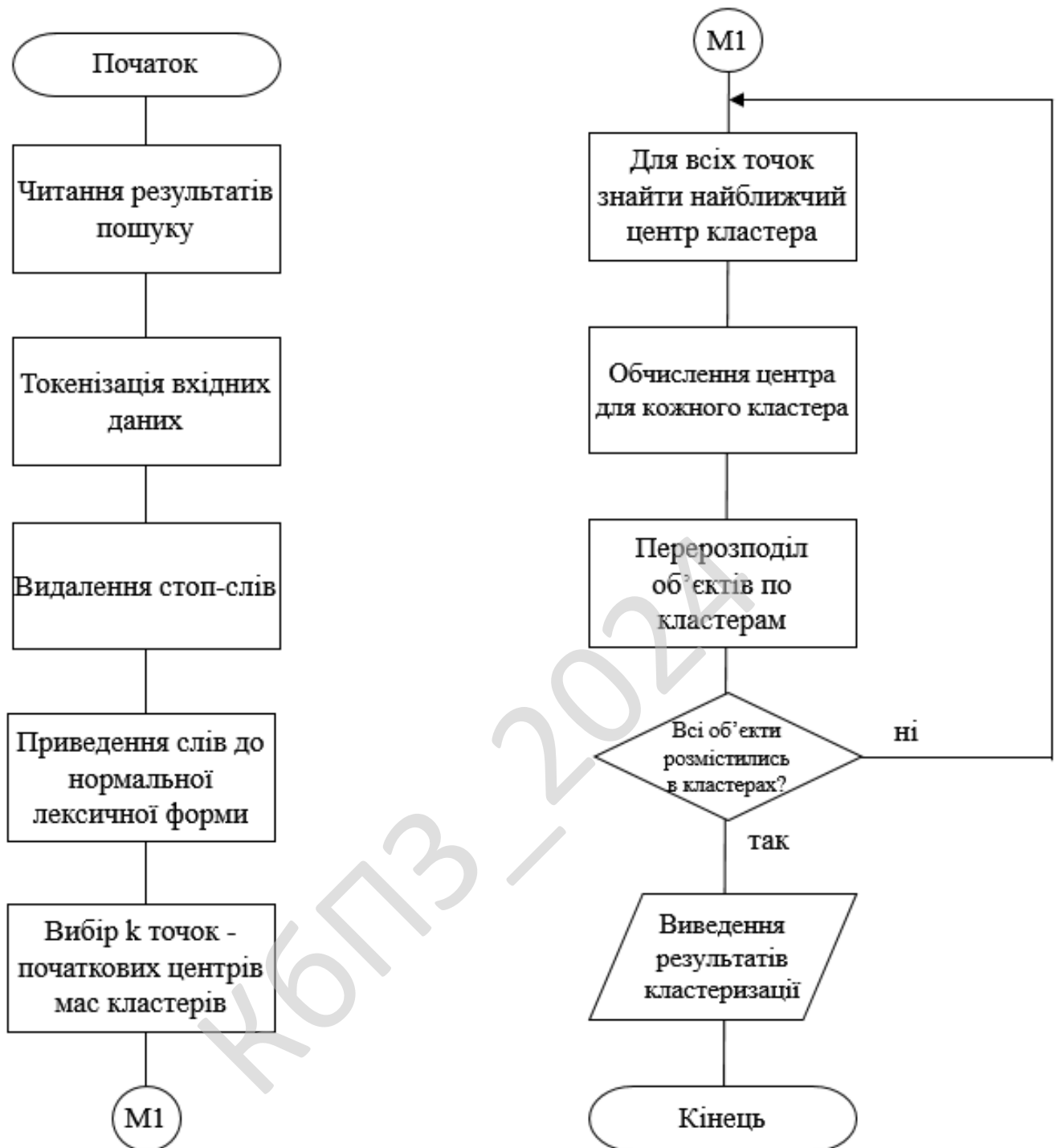


Рисунок 4.4 – Блок-схема підпрограми

Спочатку необхідно виділити вектор характеристик, що містить інформацію про словниковий склад кожного поста (запитання). Розмірність цього вектора дорівнює кількості слів в словнику. Тоді відстань між текстами можна ввести як відстань між векторами, що відповідають цим текстам.

$$tf(t, d) = \frac{n_i}{\sum_k n_k}, \quad (4.1)$$

де n_i є число входжень слова в текст, а в знаменнику – загальне число слів в даному тексті.

Для поліпшення роботи методу кластеризації було здійснено також переробку текстів. По-перше, приведення слів до своєї початкової лексичної форми і видалення всіх знаків пунктуації. Для скорочення слів до основи шляхом відкидання допоміжних частин таких як закінчення чи суфікс було використано стеммер Портера. По-друге, було здійснено видалення з тексту слів, що зустрічаються в ньому малу кількість разів, а також слів, що зустрічаються в більшості текстів (стоп-слова), використовуючи словник стоп-слів.

Алгоритм k-means є простим повторюваним алгоритмом кластеризації, який розділяє певний набір даних на задане користувачем число кластерів, k. Алгоритм простий для реалізації і запуску, відносно швидкий, легко адаптується і поширений на практиці. Це історично один з найважливіших алгоритмів інтелектуального аналізу даних.

Мета методу – розділити n спостережень на k кластерів так, щоб кожне спостереження належало до кластера з найближчим до нього середнім значенням. Метод базується на мінімізації суми квадратів відстаней між кожним спостереженням та центром його кластера, тобто функції:

$$\sum_{i=1}^N d(x_i, m_j(x_i))^2, \quad (4.2)$$

де d – метрика, x_i – i-ий об'єкт даних, а $m_j(x_i)$ – центр кластера, якому на j-ій ітерації приписаний елемент x_i . Нижче наведено код обчислення Евклідової відстані, що визначає міру наближення кластерів один до одного.

```
protected function calculateDistance($point_a, $point_b)
{
    $distance = 0;
    for ($i = 0, $count = count($point_a); $i < $count; $i++) {
        $difference = $point_a[$i] - $point_b[$i];
```

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

```

        $distance += pow($difference, 2);
    }
    return $distance;
}

```

Принцип алгоритму полягає в пошуку центрів кластерів та наборів елементів кожного кластера при наявності деякої функції $\Phi(\circ)$, що виражає якість поточного розбиття множини на k кластерів, коли сумарне квадратичне відхилення елементів кластерів від центрів цих кластерів буде найменшим:

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2, \quad (4.3)$$

де k – число кластерів, S_i – отримані кластери, $i = 1, 2, \dots, k$, μ_i – центри мас векторів $x_j \in S_i$.

В початковий момент роботи алгоритму довільним чином обираються центри кластерів. Далі для кожного елемента множини ітеративно обраховується відстань від центрів з приєднанням кожного елемента до кластера з найближчим центром. Обчислення найближчого центру виконує функція `calculateClosestCentroid()`, код якої наведено нижче.

```

protected function calculateClosestCentroid(array $observation, array
$centroids)
{
    $centroid_distance = [];
    foreach ($centroids as $centroid) {
        array_push($centroid_distance,
$this>calculateDistance($observation, $centroid));
    }
    asort($centroid_distance);
    $centroid_distance = array_keys($centroid_distance);
    return array_shift($centroid_distance);
}

```

Для кожного з отриманих кластерів обчислюються нові значення центрів. Функція `calculateCentroids()` реалізує обчислення значень нових центрів.

```

protected function calculateCentroids(array $clustered_data)
{
    $centroids = [];
    foreach ($clustered_data as $cluster) {

```

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

```

    $cluster_sum = array_fill(0, count(current($cluster)), 0);
    foreach ($cluster as $observation) {
        foreach ($observation as $key => $value) {
            $cluster_sum[$key] += $value;
        }
    }
    $centroid = array_fill(0, count(current($cluster)), 0);
    foreach ($cluster_sum as $key => $value) {
        $centroid[$key] = $value / count($cluster);
    }
    array_push($centroids, $centroid);
}
return $centroids;
}

```

При цьому виконується мінімізація функції $\Phi(^{\circ})$, після чого повторюється процедура перерозподілу елементів між кластерами.

Отже, реалізація алгоритму k-середніх складається з таких кроків:

- вибрати k інформаційних точок в якості центрів кластерів поки не завершиться процес зміни центрів кластерів;
- зіставити кожен інформаційну точку з кластером, відстань до центра якого мінімальна;
- переконатися, що в кожному кластері міститься хоча б одна точка. Для цього кожний порожній кластер потрібно доповнити довільною точкою, що розташована «далеко» від центра кластера;
- центр кожного кластера замінити середнім від елементів кластера;
- кінець.

На рисунку 4.5 зображено результати кластеризації при пошуку слова «веб-сервіс»

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

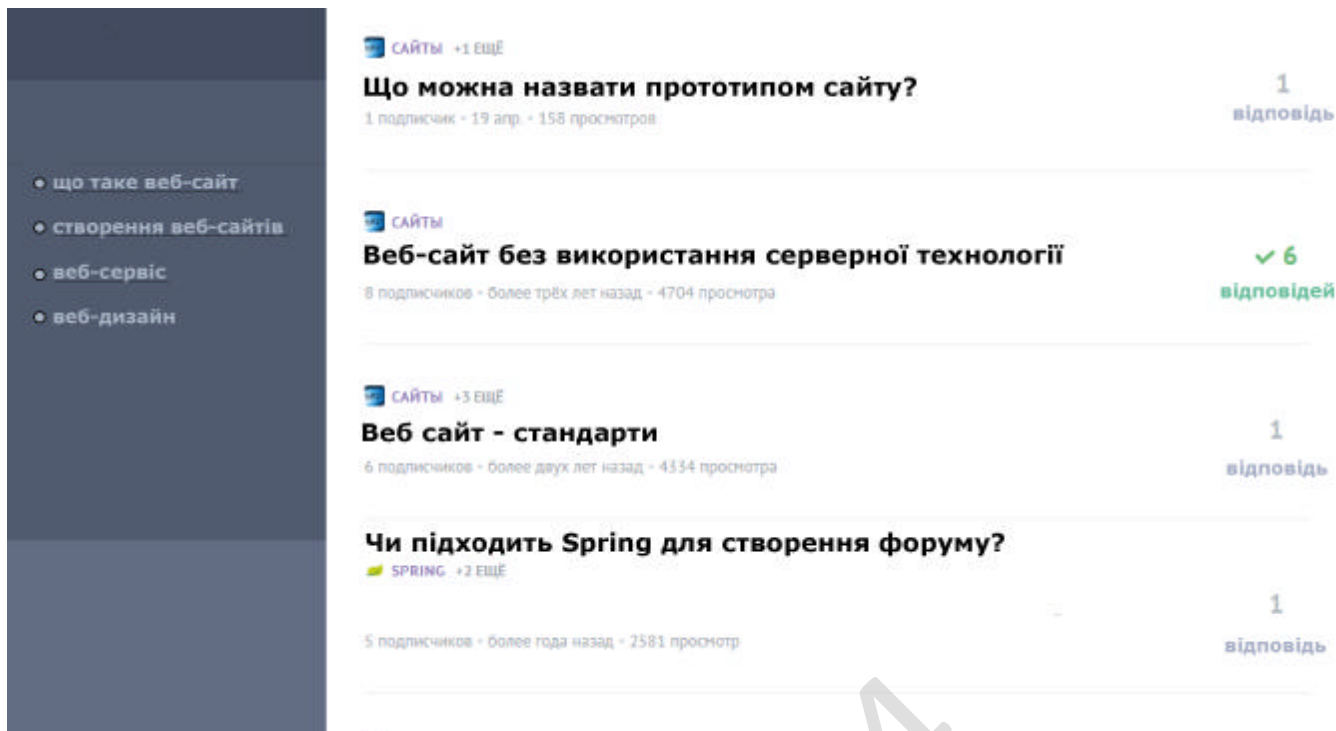


Рисунок 4.5 – Результати кластеризації

З алгоритму видно, що кожна ітерація потребує $N * K$ порівнянь, які визначають тимчасову складність однієї ітерації. Число ітерацій, необхідних для збіжності, змінюється і може залежати від N . Відповідно, чим більше точок в безлічі (N), тим довше буде працювати алгоритм.

До обмежень алгоритму включають також чутливість до ініціалізації і визначення значення K .

Незважаючи на всі недоліки, алгоритм K -середніх залишається найбільш широко використовуваним алгоритмом кластеризації на практиці.

Мова запитів SQL майже не використовувалась напряму і знайшла своє застосування тільки в запитах на створення таблиць. Наприклад, запит на створення таблиці користувачів виглядає наступним чином:

```
CREATE TABLE `tbl_user` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `login` varchar(40) NOT NULL,
  `password` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
```

```

`creation_date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
`about` text,
`reputation` int(10) unsigned,
`location` varchar(40,
PRIMARY KEY (`id`))
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Усі дії CRUD (create, read, update, delete) відбуваються методами класу Active record, функціональність якого була описана у минулих розділах.

4.2 Захист розробленого програмного забезпечення

Дані у програмному забезпеченні захищаю за допомогою NTRU. NTRUEncrypt (аббревіатура Nth-degree TRUncated polynomial ring або Number Theorists aRe Us) – це криптографічна система з відкритим ключем, що раніше називалася NTRU.

Криптосистема NTRUEncrypt, заснована на ґратчастій криптосистемі, створена як альтернатива RSA і криптосистемам на еліптичних кривих (ECC). Стійкість алгоритму забезпечується труднощами пошуку найкоротшого вектора ґрати, що більше стійка до атак, здійснюваних на квантових комп'ютерах. На відміну від своїх конкурентів RSA, ECC, Elgamal, алгоритм використовує операції над кільцем:

$$\mathbb{Z}[X]/(X^N - 1),$$

усічених багаточленів ступеня, що не перевершує $N - 1$:

$$\mathbf{a}(X) = \mathbf{a} = a_0 + a_1X + a_2X^2 + \dots + a_{N-2}X^{N-2} + a_{N-1}X^{N-1}.$$

Такий багаточлен можна також представити вектором:

$$\vec{\mathbf{a}}(X) = \vec{\mathbf{a}} = \sum_{i=0}^{N-1} a_i X^i = [a_0, a_1, a_2, \dots, a_{N-2}, a_{N-1}]$$

Як і будь-який молодий алгоритм, NTRUEncrypt погано вивчений, хоча й був офіційно затверджений для використання в сфері фінансів комітетом Accredited Standards Committee X9.[1]

						ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			62

Існує реалізація NTRUEncrypt з відкритим вихідним кодом.[2]

NTRUEncrypt, що споконвічно називався NTRU, був винайдений в 1996 році й представлений на конференціях CRYPTO, Конференція RSA, Eurocrypt. Причиною, що послужила початком розробки алгоритму в 1994 році, стала стаття [3], у якій говорилося про легкість злому існуючих алгоритмів на квантових комп'ютерах, які, як показало час, не за горами[4]. У цьому ж році, математики Jeffrey Hoffstein, Jill Pipher і Joseph H. Silverman, що розробили систему разом із засновником компанії NTRU Cryptosystems, Inc. (пізніше перейменованої в SecurityInnovation), Даніелем Лієманом (Daniel Lieman) запатентували свій винахід.[5]

Кільця усічених багаточленів

NTRU оперує над багаточленами ступеня не переважаючої $N - 1$:

$$\mathbf{a} = a_0 + a_1X + a_2X^2 + \dots + a_{N-2}X^{N-2} + a_{N-1}X^{N-1};$$

де коефіцієнти a_0, \dots, a_{N-1} – цілі числа. Щодо операцій додавання й множення за модулем багаточлена $X^N - 1$. Такі багаточлени утворюють кільце R , назване кільцем усічених багаточленів, що ізоморфно кільцю відносин:

$$\mathbb{Z}[X]/(X^N - 1).$$

NTRU використовує кільце усічених багаточленів R разом з діленням за модулем на взаємно прості числа p і q для зменшення коефіцієнтів багаточленів.

У роботі алгоритму також використовуються зворотні багаточлени в кільці усічених багаточленів. Слід зазначити, що не всякий багаточлен має зворотний, але якщо зворотний поліном існує, то його легко обчислити.[6][7]

Генерація відкритого ключа

Для передачі повідомлення від Аліси до Боба необхідні відкритий і закритий ключі. Відкритий знають як Боб, так і Аліса, закритий ключ знає тільки Боб, що він використовує для генерації відкритого ключа. Для цього Боб вибирає два «маленьких» поліноми $f, g \in R$. «Малість» поліномів мається на увазі в тому розумінні, що він маленький щодо довільного полінома за модулем q : у довільному поліномі коефіцієнти повинні бути приблизно рівномірно розподілені

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

за модулем q , а в малому поліномі вони багато менше q . Малість поліномів визначається за допомогою чисел df і dg :

Поліном f має df коефіцієнтів рівних «1» і $df-1$ коефіцієнтів рівних «-1», а інші – «0». У цьому випадку говорять, що:

$$\mathbf{f} \in \mathcal{L}_f$$

Поліном g має dg коефіцієнтів рівних «1» і стільки ж рівних «-1», інші – «0» У цьому випадку говорять, що:

$$\mathbf{g} \in \mathcal{L}_g$$

Причина, по якій поліноми вибираються саме таким чином, полягає в тому, що f , можливо, буде мати зворотний, а g – однозначно немає ($g(1) = 0$, а нульовий елемент не має зворотного).

Боб повинен зберігати ці поліноми в секреті. Далі Боб обчислює зворотні поліноми \mathbf{f}_p й \mathbf{f}_q , тобто такі, що:

$$\mathbf{f} \cdot \mathbf{f}_p \equiv 1 \pmod{p} \quad \text{й} \quad \mathbf{f} \cdot \mathbf{f}_q \equiv 1 \pmod{q}$$

Якщо f не має зворотного полінома, то Боб вибирає інший поліном f .

Секретний ключ – це пари $(\mathbf{f}, \mathbf{f}_p)$, а відкритий ключ h обчислюється за формулою:

$$\mathbf{h} = (p\mathbf{f}_q \cdot \mathbf{g}) \pmod{q}.$$

Приклад:

Для приклада візьмемо $df=4$, а $dg=3$. Тоді як поліноми можна вибрати

$$\begin{aligned} \mathbf{f} &= -1 + X + X^2 - X^4 + X^6 + X^9 - X^{10} \\ \mathbf{g} &= -1 + X^2 + X^3 + X^5 - X^8 - X^{10} \end{aligned}$$

Далі для полінома f шукаються зворотні поліноми за модулем $p=3$ і $q=32$:

$$\begin{aligned} \mathbf{f}_p &= 1 + 2X + 2X^3 + 2X^4 + X^5 + 2X^7 + X^8 + 2X^9 \\ \mathbf{f}_q &= 5 + 9X + 6X^2 + 16X^3 + 4X^4 + 15X^5 + 16X^6 + 22X^7 + 20X^8 + 18X^9 + 30X^{10} \end{aligned}$$

Заключним етапом є обчислення самого відкритого ключа h :

$$\mathbf{h} = (p\mathbf{f}_q \cdot \mathbf{g}) \pmod{32} = 8 + 25X + 22X^2 + 20X^3 + 12X^4 + 24X^5 + 15X^6 + 19X^7 + 12X^8 + 19X^9 + 16X^{10}.$$

Шифрування

Тепер, коли в Алісі є відкритий ключ, вона може відправити зашифроване повідомлення Бобові. Для цього повідомлення представити у вигляді полінома m з коефіцієнтами за модулем p , обраними з діапазону $(-p/2, p/2]$. Тобто m є «малим» поліномом за модулем q . Далі Алісі необхідно вибрати інший «малий» поліном r , що називається «сліпучої», обумовлений за допомогою числа dr :

Поліном r має dr коефіцієнтів рівних «1» і стільки ж рівних «-1», інші – «0». У цьому випадку говорять, що:

$$r \in \mathcal{L}_r.$$

Використовуючи ці поліноми, зашифроване повідомлення виходить за формулою:

$$e = (r \cdot h + m) \bmod q.$$

При цьому кожній, хто знає (або може обчислити) осліплюючий поліном r , зможе прочитати повідомлення m .

Приклад:

Припустимо, що Аліса хоче послати повідомлення, представлене у вигляді полінома:

$$m = -1 + X^3 - X^4 - X^8 + X^9 + X^{10},$$

і вибрала «осліплюючий» поліном, для якого $dr=3$:

$$r = -1 + X^2 + X^3 + X^4 - X^5 - X^7.$$

Тоді шифротекст e , готовий для передачі Бобові буде:

$$e = (r \cdot h + m) \bmod 32 = 14 + 11X + 26X^2 + 24X^3 + 14X^4 + 16X^5 + 30X^6 + 7X^7 + 25X^8 + 6X^9 + 19X^{10}$$

Розшифрування

Тепер, одержавши зашифроване повідомлення e , Боб може його розшифрувати, використовуючи свій секретний ключ. Спочатку він одержує новий проміжний поліном:

$$a = (f \cdot e) \bmod q.$$

Якщо розписати шифротекст, то одержимо ланцюжок:

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

$$\mathbf{a} = (\mathbf{f} \cdot \mathbf{e}) \bmod q = (\mathbf{f} \cdot (\mathbf{r} \cdot \mathbf{h} + \mathbf{m})) \bmod q = (\mathbf{f} \cdot (\mathbf{r} \cdot p\mathbf{f}_q \cdot \mathbf{g} + \mathbf{m})) \bmod q$$

і остаточно:

$$\mathbf{a} = (p\mathbf{r} \cdot \mathbf{g} + \mathbf{f} \cdot \mathbf{m}) \bmod q.$$

Після того, як Боб обчислив поліном \mathbf{a} за модулем q , він повинен вибрати його коефіцієнти з діапазону $(-q/2, q/2]$ і далі обчислити поліном \mathbf{b} , одержуваний з полінома \mathbf{a} приведенням за модулем p :

$$\mathbf{b} = \mathbf{a} \bmod p = (\mathbf{f} \cdot \mathbf{m}) \bmod p,$$

так як:

$$(p\mathbf{r} \cdot \mathbf{g}) \bmod p = 0.$$

Тепер, використовуючи другу половину секретного ключа й отриманий поліном \mathbf{b} , Боб може розшифрувати повідомлення:

$$\mathbf{c} = (\mathbf{f}_p \cdot \mathbf{b}) \bmod p.$$

Неважно бачити, що:

$$\mathbf{c} \equiv \mathbf{f}_p \cdot \mathbf{f} \cdot \mathbf{m} \equiv \mathbf{m} \pmod{p}.$$

У такий спосіб отриманий поліном \mathbf{c} дійсно є вихідним повідомленням \mathbf{m} .

Приклад:

Боб одержав від Аліси шифроване повідомлення \mathbf{e} :

$$\mathbf{e} = 14 + 11X + 26X^2 + 24X^3 + 14X^4 + 16X^5 + 30X^6 + 7X^7 + 25X^8 + 6X^9 + 19X^{10}$$

Використовуючи секретний ключ \mathbf{f} Боб одержує поліном \mathbf{a} :

$\mathbf{a} = \mathbf{f} \cdot \mathbf{e} \pmod{32} = 3 - 7X - 10X^2 - 11X^3 + 10X^4 + 7X^5 + 6X^6 + 7X^7 + 5X^8 - 3X^9 - 7X^{10} \pmod{32}$,
з коефіцієнтами, що належать проміжку $(-q/2, q/2]$. Далі перетворить поліном \mathbf{a} у поліном \mathbf{b} , зменшуючи коефіцієнти за модулем p .

$$\mathbf{b} = \mathbf{a} \pmod{3} = -X - X^2 + X^3 + X^4 + X^5 + X^7 - X^8 - X^{10} \pmod{3}$$

Заключний крок – перемноження полінома \mathbf{b} із другою половиною закритого ключа \mathbf{f}_p :

$$\mathbf{c} = \mathbf{f}_p \cdot \mathbf{b} = \mathbf{f}_p \cdot \mathbf{f} \cdot \mathbf{m} \pmod{3} = \mathbf{m} \pmod{3}$$

$$\mathbf{c} = -1 + X^3 - X^4 - X^8 + X^9 + X^{10}$$

Який є вихідним повідомленням, що передавала Аліса.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Згідно з ТЗ в процесі роботи над бакалаврською роботою було розроблене ПЗ моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Для успішного впровадження та інтеграції розробленого ПЗ в існуючу апаратну систему вона повинна задовольняти такі вимоги:

- встановлена ОС Windows або Linux;
- веб-сервер повинен відповідати необхідним вимогам для використання Yii2;
- PHP 5.4.0 та вище;
- встановлений браузер (Google Chrome, Opera, Mozilla Firefox);
- обсяг оперативної пам'яті комп'ютера повинен становити щонайменше 512 Мб.

Програмне забезпечення, що було розроблене в процесі виконання магістерської роботи, являє собою веб-сервіс, архітектура і структура якого детально описані в попередніх розділах пояснювальної записки і забезпечує вирішення актуальних в наш час задач. Щоб здійснити ефективне впровадження веб-сервісу в роботу була розроблена інструкція користувача з відповідними покроковими поясненнями з приводу експлуатації системи. Від якості методичного забезпечення залежить якість організації самого процесу функціонування програми та подальше виконання системою визначених функцій. Методичне забезпечення (МЕТОЗ) по використанню програми було розроблено з розрахунком на користувача, що має слабкі знання та навички роботи з програмним забезпеченням подібного класу та напрямку виконання завдань.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

Згідно з інструкцією користувача, для того, щоб створити обліковий запис необхідно вказати адресу електронної пошти та пароль у формі на сторінці реєстрації.

Рисунок 5.1 – Скріншот сторінки реєстрації

Для участі не потрібна реєстрація; можна переглядати матеріали веб-сервісу і в якості анонімного користувача. Але є функції, які неможливо зробити без реєстрації, наприклад, голосування або помітка відповіді правильною.

Як автор питання, користувач може виконати додаткову дію: прийняти відповідь, яка, на його думку, найкраще допоможе вирішити проблему.

Щоб позначити коментар як відповідь необхідно:

- вибрати одну відповідь, яка найкраще допоможе вирішити проблему;
- щоб відзначити відповідь як прийнятну, потрібно поставити прапорець в полі поруч з відповіддю, так, щоб його стан змінився з неактивного на позначений;

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

– у будь-який час можна змінити позначену відповідь або просто скасувати прийняття відповіді.

Методичне забезпечення сприяє виконанню плану впровадження, який передбачає організацію підготовки кадрів, експлуатаційного персоналу та аналіз функціонування системи після виконання робіт з її впровадження.

У плані впровадження зазначаються організації (підрозділи), які відповідають за виконання робіт. Тому для впровадження розробленого програмного забезпечення в експлуатацію потрібна взаємодія програміста-розробника з користувачем, ІТР по обслуговуванню технічного обладнання, керівників різних рівнів, підрозділів, які будуть його експлуатувати.

Сторінка користувача показана на рисунку 5.2.

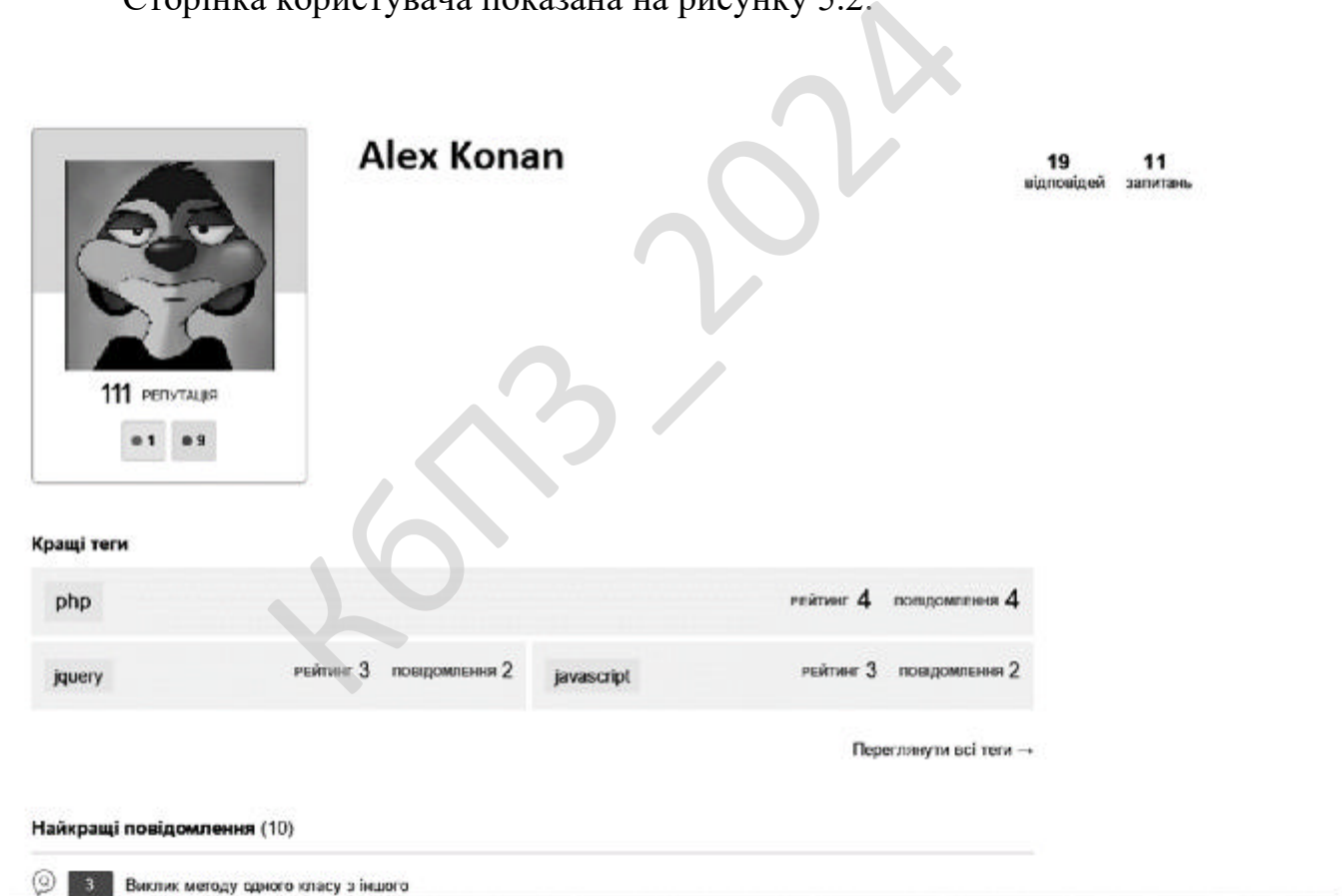


Рисунок 5.2 – Скріншот сторінки користувача

Тому лише виконання вимог та правил МЕТОЗ дозволить в короткий термін часу провести необхідні реорганізаційні зміни та успішно впровадити систему в промислову експлуатацію.

Отже, вказані мінімальні системні вимоги і розроблена інструкція користувача дає змогу легко користуватися програмним забезпеченням.

Простоту інтерфейсу досягнуто шляхом зображення на екрані інформації, мінімально необхідної для виконання користувачем чергового кроку завдання, розміщення і представлення елементів з урахуванням їх смислового значення і логічного взаємозв'язку.

КБПЗ_2024

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Метою розробки є дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Об'єктом дослідження є процес моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Предметом дослідження є методи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Методи дослідження базуються на методах кластерного аналізу, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

– Розроблено вітчизняний продукт моніторингу роботи соціального сервісу з використанням методів кластерного аналізу, який має більш широкі можливості, на відміну від існуючих аналогів.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та програмної реалізації системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу можуть бути цікавими широкої аудиторії (рисунок 7.1).

соціальні служби та неурядові організації	<ul style="list-style-type: none"> ці організації можуть використовувати систему для аналізу ефективності своїх програм, визначення груп користувачів з різними потребами та оптимізації ресурсів.
урядові структури та установи	<ul style="list-style-type: none"> уряди та муніципальні установи, що займаються соціальною політикою, можуть використовувати систему для покращення розподілу фінансування, аналізу результативності соціальних програм та ідентифікації пріоритетних соціальних груп.
комерційні компанії, які працюють у сфері соціальних досліджень	<ul style="list-style-type: none"> компанії з дослідницької сфери можуть використовувати цю систему для моніторингу соціальних сервісів, а також для аналізу поведінки користувачів та соціальних трендів, що допомагає адаптувати їхні продукти та послуги під потреби ринку.
аналітичні центри та дослідницькі інститути	<ul style="list-style-type: none"> центри, що займаються соціальними та демографічними дослідженнями, можуть використовувати систему для кластеризації даних про соціальні групи, аналізу регіональних відмінностей та тенденцій у соціальному обслуговуванні.
наукові установи та університети	<ul style="list-style-type: none"> дослідники та студенти, які працюють над проектами з кластерного аналізу та соціології, можуть використовувати результати для дослідження поведінкових патернів, соціальних трендів та адаптації сучасних методів кластеризації до соціальних даних.
розробники програмного забезпечення та консультанти з даних	<ul style="list-style-type: none"> фахівці з даних та аналітики можуть зацікавитися системою для використання в комерційних проектах, створення адаптованих рішень для клієнтів у соціальній сфері або розробки нових продуктів на основі отриманих інсайтів.
благодійні організації та фонди	<ul style="list-style-type: none"> благодійні фонди можуть використовувати кластерний аналіз для ефективного спрямування своїх зусиль та ресурсів на різні групи бенефіціарів, що дозволить їм краще зрозуміти потреби та особливості своєї аудиторії.

Рисунок 7.1 – Цільова аудиторія

Загалом, такі дослідження надають корисні інсайти для тих, хто займається соціальними програмами, моніторингом та аналізом потреб суспільства, а також впроваджує нові технології для покращення роботи соціальних сервісів.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для оцінки привабливості програмної реалізації системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу можна застосувати метод експертних оцінок. При цьому обирають групу експертів (наприклад, соціологів, фахівців з даних, представників соціальних служб) для оцінки системи за кількома критеріями.

Експерти встановлюють основні критерії для оцінки привабливості проекту: технічна ефективність (наскільки якісно реалізований функціонал кластерного аналізу), актуальність для користувачів (наскільки вирішує актуальні потреби соціального сервісу), економічна ефективність (потенціал для скорочення витрат на аналітику та моніторинг), зручність використання (наскільки легко використовувати систему для персоналу соціального сервісу), можливість інтеграції (наскільки легко впровадити систему в існуючу інфраструктуру). Кожен критерій оцінюється експертами за шкалою, наприклад, від 1 до 5, де: 1 – низька привабливість, 5 – висока привабливість.

Кожен експерт незалежно оцінює систему за визначеними критеріями. Результати групуємо в таблицю. 7.1. Далі підраховують середні оцінки для кожного критерію, щоб визначити загальну привабливість проекту. За даними таблиці, середній бал для всіх критеріїв становить 4.2, що свідчить про високу привабливість проекту. Проект має високі показники в таких критеріях, як актуальність, технічна ефективність та інтеграція, але можна розглянути шляхи підвищення економічної ефективності та зручності використання.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

Таблиця 7.1 – Зведені результати експертних оцінок

Критерій	Експерт 1	Експерт 2	Експерт 3	Середній бал
Технічна ефективність	4	5	4	4.33
Актуальність	5	4	5	4.67
Економічна ефективність	3	4	4	3.67
Зручність використання	4	4	3	3.67
Можливість інтеграції	5	5	4	4.67

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості програмної реалізації системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу найбільш підходящим методом є метод компонентної вартості.

Метод компонентної вартості (аналіз за функціональними модулями) застосовується, коли система складається з декількох функціональних модулів (наприклад, кластерний аналіз, обробка даних, зберігання даних, модулі звітності), можна розрахувати вартість окремо для кожного з них. Далі сумуються витрати на розробку всіх модулів, що дозволяє отримати загальну вартість. Переваги: точніша оцінка вартості для проектів зі складною архітектурою. Недоліки: потребує детального розподілу функцій і ресурсів, може бути трудомістким.

Метод компонентної вартості є оптимальним вибором, оскільки він дозволяє детально оцінити витрати на кожен елемент системи і підходить для складних проектів з багатьма функціями. Це дає можливість глибокого розуміння вартості кожного функціонального елемента, зокрема, складових частин, які забезпечують кластерний аналіз, збір та зберігання даних.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Економічна ефективність від впровадження системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу може бути продемонстрована через зниження витрат, оптимізацію робочих процесів та покращення якості обслуговування.

Таблиця 7.2 – Економічна ефективність від реалізації проєкту для клієнта

1. Зниження витрат на аналіз та обробку даних

До впровадження системи аналіз даних соціального сервісу виконувався вручну або з використанням менш ефективних інструментів, що займало значну частину робочого часу фахівців. Впровадження автоматизованої системи на основі кластерного аналізу скорочує час на аналіз даних, що, відповідно, знижує витрати на оплату праці.

Приклад розрахунку: Якщо до впровадження 10 аналітиків витрачали 100 годин на місяць на ручний аналіз, то із впровадженням системи цей час може скоротитися до 30 годин. За середньою зарплатою аналітика \$20/год, це зекономить: $(100-30) \times 10 \times 20 = 14,000$ доларів щомісяця

2. Оптимізація ресурсів завдяки кластерному аналізу

Кластерний аналіз допомагає соціальному сервісу сегментувати клієнтів за потребами, забезпечуючи точнішу таргетовану допомогу. Це знижує витрати на невідповідне обслуговування та дозволяє уникнути надмірних витрат.

Приклад розрахунку: Припустимо, що ефективність обслуговування клієнтів зросла на 15% завдяки точнішій сегментації. Якщо річний бюджет обслуговування складає 1 мільйон доларів, то економія складе: $1,000,000 \times 0.15 = 150,000$ доларів на рік



Рисунок 7.2 – Алгоритм просування проекту

Такий алгоритм дозволить ефективно розгорнути проект, залучити нових клієнтів, підвищити довіру та забезпечити довгострокове використання системи моніторингу на основі кластерного аналізу.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту та шляхів реалізації проекту системи моніторингу роботи соціального сервісу з використанням кластерного аналізу можна зосередитися на кількох ключових напрямках. Ось кілька стратегій для цього:

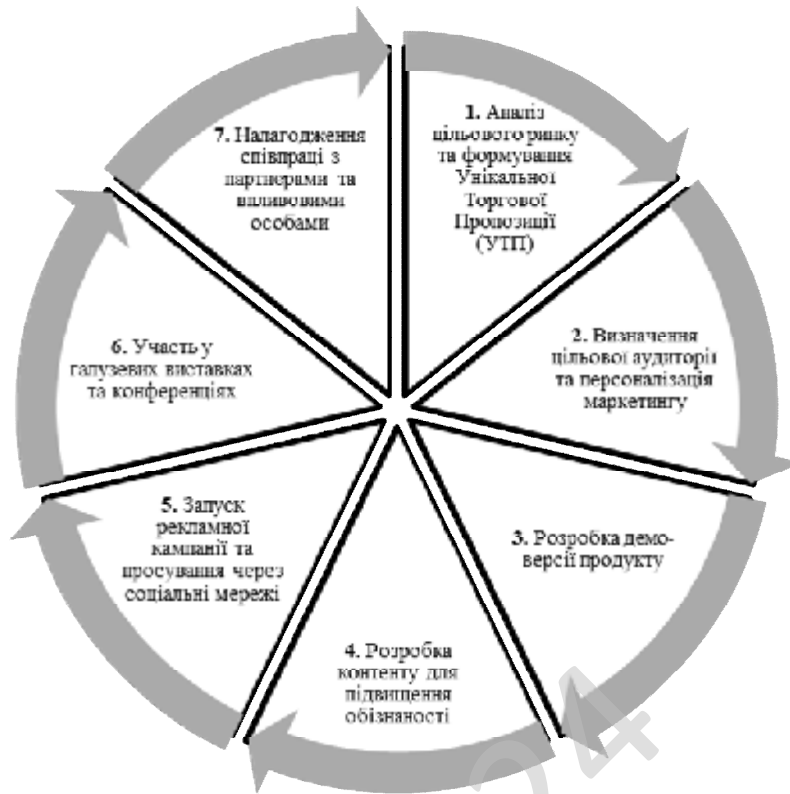


Рисунок 7.3 – Оптимізація каналів збуту

Ці підходи дозволять максимально ефективно охопити цільову аудиторію, підвищити рівень продажів та забезпечити задоволеність клієнтів системою моніторингу на основі кластерного аналізу.

7.7 Визначення ключових факторів успіху конкретного проєкту

Ключові фактори успіху проєкту програмної реалізації системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу представлені на рисунку 7.4.

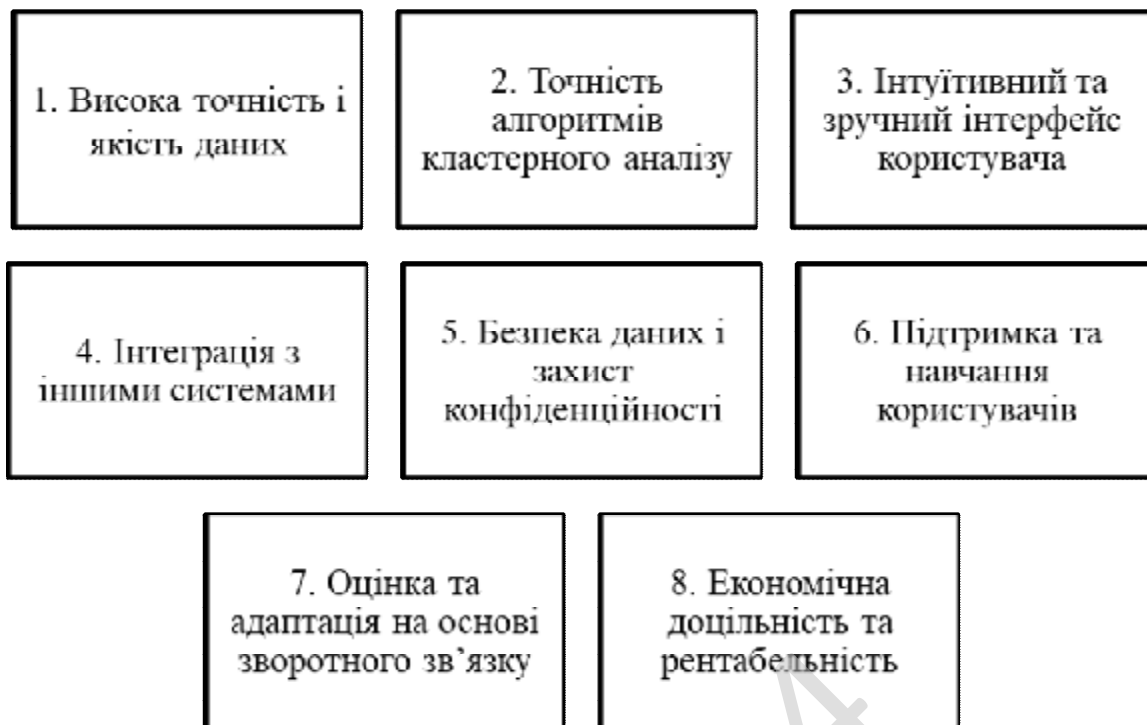


Рисунок 7.4 – Ключові фактори успіху проекту

Ці фактори сприяють успішному впровадженню та ефективному використанню системи, дозволяючи соціальним сервісам підвищити свою продуктивність, забезпечити надійну аналітику та ефективно моніторити свої процеси.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Згідно закону України “Про охорону праці” [3] кожна компанія впроваджує заходи з охорони праці. Реалізується трудові відносини з вживанням необхідних засобів з охорони праці та розробки відповідних документів:

- Інструкцій з охорони праці по кожній професії і загальні;
- Положення про охорону праці;
- Накази з охорони праці;
- Журнали реєстрації та інструктажу.

Роботодавець створює відділ який працює відповідно до типового положення, яку затверджується центральним органом виконавчої влади і забезпечує виконання вимог державної політики у сфері охорони праці.

За недотриманням вимог, керівники ІТ компаній можуть бути притягнуті до відповідальності, яка виглядає у виді накладання штрафу. Якщо в результаті порушення умов охорони праці є постраждалі працівники то керівні особи ІТ компаній притягуються до кримінальної відповідальності. Законом України “Про охорону праці” [3] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [5], яким затверджено нормативно-правовий акт з охорони праці НПАОП 0.00-7.15-18, «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		80

Програмісти у процесі роботи мають негативний вплив на органи зору, а також мають значну розумову напругою і нервово-емоційне навантаження. Руки (суглоби пальців та м'язи рук) при роботі з клавіатурою мають теж істотне навантаження. До шкідливих факторів, які впливають на робітників галузі інформаційних технологій (ІТ) спеціалісти відносять високочастотні електромагнітні коливання (випромінювання) роботи апаратної частини ЕОМ та виділення шкідливих газів.

Ці шкідливі фактори можуть привести до професійних захворювань.

Розглянемо шкідливі чинники роботи програмістів керуючись наступними нормативно-правовими актами: «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98 [5], та «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» НПАОП 0.00-7.15-18.

Умови праці програміста включають наступні фактори:

- параметри повітряного середовища в приміщенні;
- вентиляція приміщення;
- освітлення приміщення;
- параметри повітряного середовища в приміщенні, тощо.

Щоб запропонувати заходи щодо зменшення негативного впливу комп'ютера на організм людини визначимо фактори, які можуть викликати професійне захворювання і впливають на працездатність програміста.

8.2 Аналіз умов праці на робочому місці ІТ-фахівця

На робочому місці ІТ-фахівця (або програміста) виникають небезпечні та шкідливі для безпечної життєдіяльності фактори:

- підвищений рівень шуму;
- несприятливі мікрокліматичні умови;
- недостатній рівень освітленості;

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		81

- шкідливі речовини;
- підвищений рівень електромагнітних випромінювань радіочастот;
- висока напруга електричної мережі;
- статична електрика та інші.

Робота програміста супроводжується також підвищеним ступенем напруженості трудового процесу. При систематичному впливі виробничих факторів, які не відповідають нормативним показникам, зростає рівень професійно зумовленої захворюваності працюючих та можуть виникнути професійні захворювання органів зору, руху, нервової системи. Таким чином, вивчення умов праці на робочому місці програміста є необхідною умовою запобігання негативних наслідків впливу небезпечних та шкідливих факторів. Робоче місце, добре пристосоване до трудової діяльності інженера, правильно і доцільно організоване, щодо простору, форми, розміру забезпечує йому зручне положення при роботі і високу продуктивність праці при найменшому фізичному і психічному напруженні.

Нормування параметрів проводиться в залежності від періоду року та категорії важкості виконуваних робіт. Для постійних робочих місць, якими є робочі місця ІТ-фахівців, встановлені оптимальні параметри мікроклімату, а за неможливості їх дотримання використовують допустимі параметри. Робота ІТ-фахівця за важкістю відноситься до Іа (роботи, що виконуються сидячи і не потребують фізичного напруження) та Іб (роботи, що виконуються сидячи, стоячи або пов'язані з ходінням та супроводжуються деяким фізичним напруженням) категорій. В таблиці 8.1. наведені оптимальні параметри мікроклімату в приміщеннях.

Виміряні за допомогою приладів температура та вологість у приміщеннях праці ІТ-фахівців повинні відповідати зазначеним у таблиці для теплого періоду року.

Слід зазначити, що для нормалізації параметрів мікроклімату слід використовувати у приміщеннях кондиціонування повітря, або забезпечити подачу

свіжого повітря системами вентиляції. Норми подачі свіжого повітря наведені у таблиці 8.2.

Таблиця 8.1 – Параметри мікроклімату для приміщень з ПК

Період року	Параметр мікроклімату	Величина
Холодний	Температура повітря в приміщенні; відносна вологість; швидкість руху повітря	22...24°C; 40... 60%; до 0,1 м/с
Теплий	Температура повітря в приміщенні; відносна вологість; швидкість руху повітря	23...25 °C; 40...60%; 0,1...0,2 м/с

Таблиця 8.2 – Норми подачі свіжого повітря в приміщення

Характеристика приміщення	Об'ємна витрата свіжого повітря, що подається в приміщення, м ³ на одну людину в годину
Об'єм до 20 м ³ на людину	Не менше 30
20... 40 м ³ на людину	Не менше 20
Більше 40 м ³ на людину	Може біти використана природна вентиляція

Створення сприятливих умов праці і правильне естетичне оформлення робочих місць на виробництві має велике значення як для полегшення праці, так і для підвищення його привабливості, позитивно впливає на продуктивність праці. Забарвлення приміщень і меблів повинні сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою. У службових приміщеннях, у яких виконується одноманітна розумова робота, що вимагає значної нервової напруги і великого зосередження, забарвлення повинно бути спокійних тонів – малонасичені відтінки холодного зеленого або блакитного кольорів.

При розробці оптимальних умов праці програміста необхідно враховувати освітленість. Рациональне освітлення робочого місця є одним з найважливіших факторів, що впливають на ефективність трудової діяльності людини, що попереджають травматизм і професійні захворювання. Правильно організоване освітлення створює сприятливі умови праці, підвищує працездатність і продуктивність праці. Освітлення на робочому місці програміста повинно бути таким, щоб працівник міг без напруги зору виконувати свою роботу. Стомлюваність органів зору залежить від ряду причин: недостатність освітленості; надмірна освітленість; неправильний напрям світла. Недостатність освітлення приводить до напруги зору, ослабляє увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає засліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань. [4]

8.3 Пропозиції щодо підвищення працездатності ІТ-фахівців

Поява та впровадження нових інформаційно-комунікаційних технологій зумовлює необхідність подальшого вдосконалення охорони праці фахівців ІТ-індустрії. Все це потребує розробки нових нормативно-правових актів з регламентації праці та відпочинку фахівців ІТ-індустрії і стандартів підприємств, центрів комп'ютерної техніки, центрів інформаційних технологій, сучасних комп'ютерних класів. Для підвищення розумової працездатності то зорової роботи повинна здійснюватися ергономічна оптимізація в рамках системи «оператор-термінал», яка сприятиме результативній фізичній та інтелектуальній працездатності і відновленню психосоматичного здоров'я фахівців ІТ-індустрії. Всі наведені заходи щодо вдосконалення охорони праці фахівців ІТ-індустрії повинні контролюватися службою охорони праці та комісією з охорони праці підприємства.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

Особливе значення у соціальному захисті цієї категорії працівників належить прийняття комплексного договору, який може забезпечити фахівців додатковими пільгами та компенсаціями.

Пропозиції щодо підвищення працездатності ІТ-фахівців, розіб'ємо на декілька категорій:

1. Середовище і розпорядок праці.

Для мінімізації негативних ефектів, що пов'язані з перевтомленням ІТ-фахівців, потрібно чітко прописати і реалізувати графік періодів праці-відпочинку, щоб фахівець міг можливість переключити увагу, дати можливість відпочити очам, мозку, елементарно, встати розім'яти ноги.

Також потрібно зробити максимально комфортними умови мікроклімату у офісному приміщенні, де працюють ІТ-фахівці. Мається на увазі встановлення і експлуатація, коли виникає необхідність, кондиціонерів, опалення, та системи вентиляції, задля попередження перегрівання, переохолодження ІТ-фахівців, і подальшої неможливості ними виконувати свої функції.

Також, за можливості, нами пропонується введення практики віддаленої праці ІТ-фахівцями, якщо роботодавець не може забезпечити оптимальні і безпечні умови в офісному приміщенні, або якщо фахівця вони не влаштовують із певних причин.

2. Фізичні і психоемоційні чинники.

Першим і найважливішим чинником, що впливає на працездатність ІТ-фахівців є робоче місце, і саме тому, роботодавець має забезпечити максимальний його комфорт і безпеку.

Гарантією цих факторів може слугувати сертифікація меблів, що використовуються на підприємстві ІТ-галузі. Тому нами пропонується закупівля тільки меблів, які пошли сертифікацію на відповідність. Під психоемоційними чинниками ми розуміємо гарне самопочуття фахівців, позитивний настрій, гарний психологічний клімат у колективі, тощо. Задля того, щоб психоемоційні чинники мали максимально позитивний ефект, керівництву слід поводити заходи, які

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		85

сприятимуть укріпленню і покращенню міжособистісних стосунків у колективі, таких як психологічні тренінги, тимбілдінг, спортивні змагання і естафети.

Також, сюди можна віднести розробку і впровадження системи мотивації працівників, як фінансової, так моральної і адміністративної.

8.4 Розрахункова частина

Для захисного штучного заземлення застосовуються вертикальні електроди: металевий куток $50 \cdot 50 \cdot 5$ мм., довжиною $L=3$ м., та горизонтальний електрод – металева полоса з перетином $50 \cdot 5$ мм. Напруга – 220/380 В. Розрахункова схема розташування заземлюючих електродів – по контуру (прямокутником).

Розрахунок проведемо за допустимим опором розтіканню струму заземлювача.

Початкові дані для розрахунку захисного заземлення: тип верхнього шару ґрунта – чорнозем, нижнього шару ґрунта – глина (питомий опір $\rho_2 = 40$ Ом·м). Умовна товщина верхнього шару ґрунта: $H=0,6$ м. Відстань між вертикальними заземлювачами (електродами) $A=3$ м. Глибина закладення горизонтального контуру заземлення $t=0,75$ м. Опір заземлювача, який нормується: $R_{3H} = 4$ Ом. Необхідно визначити необхідну кількість вертикальних заземлювачів та довжину полоси (горизонтального заземлювача) (рис. 8.1).

Виконуємо розрахунок.

Відстань від центра вертикального заземлювача до поверхні землі:

$$T=t+L/2=0,75 +3/2=2,25 \text{ м.}$$

Розрахунковий питомий опір ґрунта (з врахуванням того, що фактично вся конструкція заземлювача розташовується у нижньому шарі ґрунта):

$$\rho = \psi \rho_2 = 1,36 \cdot 40 = 54,5 \text{ Ом} \cdot \text{м.}$$

де $\psi = 1,36$ – табличне значення коефіцієнта сезонності для відповідної кліматичної зони у багат шаровому ґрунті [11];

$\rho_2 = 40$ Ом·м – табличне значення питомого опору нижнього шару ґрунта (глина) [11].

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		86

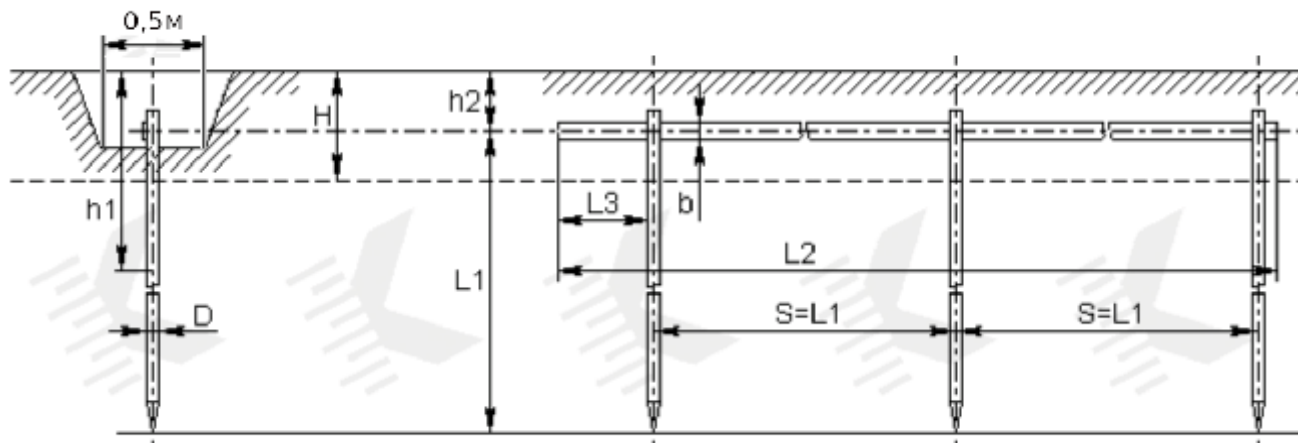


Рисунок 8.1 – Схема штучного заземлення

Еквівалентний діаметр вертикального електрода (кутка) [11]:

$$D_{\text{в}} = 0,95 \cdot K = 0,95 \cdot 50 = 47,5 \text{ мм.} = 0,0475 \text{ м.}$$

де $K = 50 \text{ мм}$ – розмір металевого кутка (задано).

Відношення $A/L = 3/3 = 1$.

Опір розтіканню електричного струму одного електрода вертикального заземлювача з урахуванням заглиблення заземлювача [11]:

$$R_0 = 0,366 \cdot (\rho/L) \cdot [\lg(2L/D_{\text{в}}) + (1/2) \cdot \lg((4T+L)/(4T-L))] = \\ = 0,366 \cdot (54,5/3) \cdot [\lg(2 \cdot 3/0,0475) + (1/2) \cdot \lg((4 \cdot 2,25+3)/(4 \cdot 2,25-3))] = 14,9 \text{ Ом.}$$

Визначаємо коефіцієнт екранування вертикальних електродів $K_{\text{ев}} = 0,53$ при орієнтовній кількості вертикальних електродів, яке дорівнює 5 [11].

Визначаємо необхідну кількість вертикальних електродів заземлювача (без врахування горизонтального заземлювача), при $R_{\text{ЗН}} = 4 \text{ Ом}$:

$$N = R_0 / (K_{\text{ев}} R_{\text{ЗН}}) = 14,9 / (0,53 \cdot 4) = 7,05 \approx 7 \text{ шт.}$$

Визначаємо довжину з'єднуючої полоси:

$$L_{\text{п}} = 1,05 \cdot A \cdot N = 1,05 \cdot 3 \cdot 7 = 18,5 \approx 18 \text{ м.}$$

Опір розтіканню електричного струму з'єднуючої полоси з урахуванням кліматичного коефіцієнта питомого опору ґрунта $K_{\text{п}}$ [11]:

$$R_{\text{п}} = 0,366 (\rho \cdot K_{\text{п}} / L_{\text{п}}) \lg(2 \cdot L_{\text{п}}^2 / (B \cdot t)) = \\ = 0,366 (40 \cdot 5 / 18) \cdot \lg((2 \cdot 18^2) / (0,05 \cdot 0,75)) = 20,1 \text{ Ом.}$$

де $K_{\text{п}} = 5$ – табличне значення кліматичного коефіцієнта питомого опору ґрунта для відповідної кліматичної зони для з'єднуючої полоси [11]:

$B = 50 \text{ мм} = 0,05 \text{ м.}$ - ширина з'єднуючої полоси (задана).

Загальний опір розтіканню електричного струму заземлювача [11]:

$$R = (R_0 \cdot R_{\Pi}) / (R_0 \cdot \eta_{\Pi} + N \cdot R_{\Pi} \cdot K_{\text{ев}}) = \\ = (14,9 \cdot 20,1) / (14,9 \cdot 0,55 + 7 \cdot 20,1 \cdot 0,53) = 3,56 \text{ Ом.}$$

де $\eta_{\Pi} = 0,55$ – табличне значення коефіцієнта екранування з'єднуючої полоси [11].

Умова $R \leq R_{\text{зн}}$ виконується ($3,56 \leq 4$).

Остаточна кількість вертикальних електродів дорівнює 7.

За потреби можна зменшити кількість електродів заземлювача, зменшивши загальний опір розтіканню електричного струму заземлювача методом зменшення питомого опору ґрунта, домішуючи у ґрунт безпосередньо навколи електродів заземлювача розчини солей NaCl, CaCl, сажу, соду, шлак або спеціальні суміші.

8.5 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому. З цих міркувань було здійснено аналіз приміщення, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Тільки повна усвідомленість працівника про можливі небезпеки, що можуть підстерігати його на робочому місці та дотримання вимог нормативних актів з питань охорони праці та відповідних рекомендацій фахівців, дозволять значною мірою знизити негативний вплив шкідливих та небезпечних факторів при роботі з комп'ютером на організм людини. Виконано розрахунок захисного штучного заземлення, як одного з ключових факторів безпеки програміста.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		88

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.
- Досліджена система моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.
- На основі отриманих результатів досліджень створена програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		89

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня РНР фреймворк Yii2. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм NTRUEncrypt.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проєкту.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		90

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ширяев О.І. Дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу // Збірник праць молодих науковців ЦНТУ. – Вип. 14. – Кропивницький: ЦНТУ, 2024.
2. Foreman J.W. Data Smart: Using Data Science to Transform Information into Insight 1st Edition. – Wiley, 2013. – 432 p.
3. Hurbans R. Grokking Artificial Intelligence Algorithms. – Manning, 2020. – 631 p.
4. Gusfield D. Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology 1st Edition. – Cambridge University Press, 2008. – 556 p.
5. Kotu V., Deshpande B. Data Science: Concepts and Practice. – Elsevier Science, 2018. – 953 p.
6. Knowledge Base A Complete Guide – 2021 Edition // The Art of Service – Knowledge Base Publishing, 2020. – 306 p.
7. Knuth D. The Art of Computer Programming, Vol. 1: Fundamental Algorithms, 3rd Edition 3rd Edition. – Addison-Wesley Professional, 2019. – 672 p.
8. Mattmann C. Machine Learning with TensorFlow, Second Edition. – Manning, 2020. – 1124 p.
9. Mueller J.P., Massaron L. Machine Learning For Dummies. – Wiley, 2016. – 714 p.
10. Teofili T. Deep Learning for Search. – Manning, 2019. – 695 p.
11. Rungta K. TensorFlow in 1 Day: Make your own Neural Network. – Publishdrive, 2019. – 587 p.
12. Weidman S. Deep Learning from Scratch: Building with Python from First Principles. – O'Reilly. 2019. – 252 p.

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		91

13. Rajasekaran S., Vijayalakshmi Pai G.A. Neural networks, fuzzy logic, and genetic algorithms: synthesis and applications (with cd-rom) Kindle Edition. – PHI, 2013. – 628 p.

14. Smirnov O., Fedorov E., Neskorođieva A., Neskorođieva T. «Intellectual Classification method of Gymnastic Elements Based on Combinations of Descriptive and Generative Approache». *CEUR Workshop Proceedings Volume 3664*, 2024, Pages 11-23.

15. Malyukov V., Bebishko B., Lakhno V., Smirnov O., Malyukova I., Mohylnyi H. «Managing the Purchase-Sale Process of Digital Currencies Under Fuzzy Conditions». *Lecture Notes in Networks and Systems*, 2023, 729 LNNS, pp. 104–112.

16. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.

17. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yanchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.

18. Smirnov, O., Karapetyan, A., Fedorov, E., «Creating Neural Network and Single Solution Human-Based Metaheuristic Methods of Solving the Traveling Salesman Problem». *CEUR Workshop Proceedings*, Volume 3312, 2022, pp. 47-58.

19. Smirnov, O., Neskorođieva, T., Fedorov, E., Rudakov, K., Neskorođieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings*, Volume 3187, 2022, pp. 1-12.

20. Smirnov O., Smirnova T., Anas M. Al-Oraiqat, Drieiev O., Polishchuk L., Sheroz Khan, Yassin M. Y. Hasan, Aladdein M. Amro, Hazim S. AlRawashdeh «Method for Determining Treated Metal Surface Quality Using Computer Vision Technology». *Sensors (Basel, Switzerland)* Volume 22, Issue 16, 6223, 2022.

21. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebishko, B. «Selection of a Rational Composition of Information Protection

Means Using a Genetic Algorithm». In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies, vol 131. 2023. Springer, Singapore. pp. 21-34.

22. Kuznetsov, A., Oleshko, I., Chernov, K., Bagmut, M., Smirnova, T. «Biometric authentication using convolutional neural networks». Lecture Notes in Networks and Systems. Volume 152, 2021, Pages 85-98.

23. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». SN Computer Science, Vol 2, 337, 2021. <https://doi.org/10.1007/s42979-021-00739-w>.

24. Smirnov O., Neskorođieva T., Fedorov E., Rymar P. «Neural Network Modeling Method of Transformations Data of Audit Production with Returnable Waste». CEUR Workshop Proceedings Volume 3101, 2021, Pages 192-207.

25. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

26. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

27. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

28. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and

Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

29. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.

30. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

31. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

32. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

33. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

34. Smirnov, O., Ulichev, O., Meleshko, Y., Khokh, V., Goncharenko, I. «Method of Choosing Objects for Informational Influence in Social Networks during Information Campaign Based on the Analytic Hierarchy Process». CEUR Workshop Proceedings, Vol 2588, P. 215-227, 2019.

35. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

36. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

37. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 353-358.

38. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.

39. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

40. Smirnov, S., Bulekbaeva, G., Kikvidze, O.G., Lakhno, V., Brzhanov, R., Tabylov, A. «Computer simulation in the MathCAD package of plastic deformation of the deposited layer on the flat surface of the part». Journal of Theoretical and Applied Information Technology Volume 97, Issue 20, 2019, Pages 2467-2484. (Scopus).

41. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

42. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». Сучасні інформаційні системи, 2023, том 7, № 2, С. 49-56.

43. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». Сучасні інформаційні системи. 2021. Т. 5, № 4. С. 79-95

44. Смірнов, О.А., Усік П.С., Полігенько О.О., Одарченко Р.С., Терещенко Л.Ю. «Інформаційна технологія та програмне забезпечення для підвищення ефективності планування підсистеми базових станцій стільникового зв'язку». Проблеми телекомунікацій. № 1(26). С. 83-96. 2020.

45. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у комп'ютерних системах операторів стільникового зв'язку» Вісник Черкаського державного технологічного університету. Технічні науки. №4. С. 103-110. 2020.

46. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.

47. Смірнов, С.А., Смірнова, Т.В., Минайленко, Р.М., Доренський, О.П., Сисоєнко С.В. «Хмарна автоматизована система інтелектуальної підтримки прийняття рішень для технологічних процесів». Вісник Черкаського державного технологічного університету. Технічні науки. №4, 2020, С. 84-92.

48. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральнуукраїнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.

49. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		96

50. О.А. Смірнов, Т.В. Смірнова, О.М. Дреєв, Є.К. Солових, «Методи оптимізації технологічних процесів відновлення сталевих покриттів», Shipbuilding & marine infrastructure / Суднобудування і морська інфраструктура № 1 (11). с. 48-57, 2019.

51. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

52. Смірнова Т.В., Дреєв О.М., Смірнов О.А. Експертна система оптимізації процесу відновлення та зміцнення поверхонь деталей типу «вал» електродуговим напиленням. Системи управління, навігації та зв'язку, № 2 (54). С. 149-154, 2019.

КБПЗ_2024

					ВКРМ-122.24.0020.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		97

Додаток А
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					ВКРМ-122.24.0020.00.00.ТЗ		
Вим.	Арк.	№ документа	Підпис	Дата			
Розробив	Ширяєв О.І.				Літ.	Аркуш	Аркушів
Перевірів	Смірнова Т.В.						
Н. Контр.	Коваленко А.С.				ЦНТУ КН-23М		
Затв.	Смірнов О.А.						
					Дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу		

1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 18-13 від 07.08.2024 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;

					ВКРМ-122.24.0020.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи моніторингу роботи соціального сервісу з використанням методів кластерного аналізу;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					ВКРМ-122.24.0020.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище PHP фреймворк Yii2.

					ВКРМ-122.24.0020.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати маркетингове та економічне обґрунтування ІТ-проєкту з урахуванням цін на 3 вересня 2024 року.

8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинні бути розглянуті пропозиції щодо підвищення працездатності ІТ-фахівців.

					ВКРМ-122.24.0020.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 97 аркушів.

10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Маркетингове та економічне обґрунтування ІТ-проєкту.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 02.12.2024 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 16.12.2024 р.

					ВКРМ-122.24.0020.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
другим (магістерським) рівнем вищої освіти
_____ Смірнова Т.В.

*Дослідження та програмна реалізація
системи моніторингу роботи соціального сервісу з використанням методів
кластерного аналізу*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 20

Літера: РП

Кропивницький – 2024 року

Файл PostController - контролер постів

```
<?php

namespace app\controllers;

use Yii;
use app\models\Post;
use app\models\PostsSearch;
use yii\web\Controller;
use yii\web\NotFoundHttpException;
use yii\filters\VerbFilter;

/**
 * PostController імplementує CRUD дії для моделі Post.
 */
class PostController extends Controller
{

    public function behaviors()
    {
        return [
            'verbs' => [
                'class' => VerbFilter::className(),
                'actions' => [
                    'delete' => ['POST'],
                ],
            ],
        ];
    }

    /**
     * Список всіх моделей Post.
     */
    public function actionIndex()
    {
        $searchModel = new PostsSearch();
        $dataProvider = $searchModel->search(Yii::$app->request->queryParams);

        return $this->render('index', [
            'searchModel' => $searchModel,
            'dataProvider' => $dataProvider,
        ]);
    }

    /**
     * Відображення моделі Post.
     */
    public function actionView($id)
    {
        return $this->render('view', [
            'model' => $this->findModel($id),
        ]);
    }

    /**
     * Створення моделі нового поста.
     * Якщо створення пройшло успішно браузер переходить на сторінку 'view'
     */
    public function actionCreate()
    {
        $model = new Post();

        if ($model->load(Yii::$app->request->post()) && $model->save()) {
            return $this->redirect(['view', 'id' => $model->id]);
        } else {
            return $this->render('create', [
                'model' => $model,
            ]);
        }
    }
}
```

```
    }  
  }  
  
  /**  
   * Оновлення існуючої моделі Post.  
   */  
  public function actionUpdate($id)  
  {  
    $model = $this->findModel($id);  
  
    if ($model->load(Yii::$app->request->post()) && $model->save()) {  
      return $this->redirect(['view', 'id' => $model->id]);  
    } else {  
      return $this->render('update', [  
        'model' => $model,  
      ]);  
    }  
  }  
  
  /**  
   * Видалення моделі Post.  
   * Якщо видалення пройшло успішно браузер переходить на сторінку 'index'  
   */  
  public function actionDelete($id)  
  {  
    $this->findModel($id)->delete();  
  
    return $this->redirect(['index']);  
  }  
  
  /**  
   * Знаходить модель Post по ключу БД.  
   * якщо модель не знайдена генерується виключна ситуація  
   *NotFoundException  
   */  
  protected function findModel($id)  
  {  
    if (($model = Post::findOne($id)) !== null) {  
      return $model;  
    } else {  
      throw new NotFoundException('The requested page does not  
exist.');    }  
  }  
}
```

Файл SiteController.php - контролер сайту

```
<?php

namespace app\controllers;

use Yii;
use yii\filters\AccessControl;
use yii\web\Controller;
use yii\filters\VerbFilter;
use app\models\LoginForm;
use app\models\ContactForm;
use app\models\Signup;
use app\models>Login;

class SiteController extends Controller
{
    public function behaviors()
    {
        return [
            'access' => [
                'class' => AccessControl::className(),
                'only' => ['logout'],
                'rules' => [
                    [
                        'actions' => ['logout'],
                        'allow' => true,
                        'roles' => ['@'],
                    ],
                ],
            ],
            'verbs' => [
                'class' => VerbFilter::className(),
                'actions' => [
                    'logout' => ['post'],
                ],
            ],
        ];
    }

    public function actions()
    {
        return [
            'error' => [
                'class' => 'yii\web\ErrorAction',
            ],
            'captcha' => [
                'class' => 'yii\captcha\CaptchaAction',
                'fixedVerifyCode' => YII_ENV_TEST ? 'testme' : null,
            ],
        ];
    }

    public function actionIndex()
    {
        return $this->render('index');
    }

    //реєстрація користувача
    public function actionSignup(){
        $model = new Signup();
        if(isset($_POST['Signup'])){
            $model->attributes= Yii::$app->request->post('Signup');

            if($model->validate() && $model->signup()){
                return $this->redirect(['index']);
            }
        }
    }
}
```

```

    }
    return $this->render('signup', ['model'=>$model]);
}

//функція входу на свій акаунт
public function actionLogin(){
    $login_model = new Login();
    if(Yii::$app->request->post('Login')){
        $login_model->attributes = Yii::$app->request->post('Login');

        if($login_model->validate())
        {

            Yii::$app->user->login($login_model->getUser());
            return $this->goHome();
        }
    }
    return $this->render('login', ['login_model'=>$login_model]);
}

public function actionLogout()
{
    if(!Yii::$app->user->isGuest)
    {
        Yii::$app->user->logout();
        return $this->redirect(['login']);
    }
}

public function actionContact()
{
    $model = new ContactForm();
    if ($model->load(Yii::$app->request->post()) && $model->contact(Yii::$app->params['adminEmail'])) {
        Yii::$app->session->setFlash('contactFormSubmitted');

        return $this->refresh();
    }
    return $this->render('contact', [
        'model' => $model,
    ]);
}

public function actionAbout()
{
    return $this->render('about');
}
}

```

Файл UserController.php - контролер користувачів

```
<?php

namespace app\controllers;

use Yii;
use app\models\User;
use yii\data\ActiveDataProvider;
use yii\web\Controller;
use yii\web\NotFoundHttpException;
use yii\filters\VerbFilter;

/**
 * UserController імплементує CRUD події для User.
 */
class UserController extends Controller
{
    /**
     * @inheritdoc
     */
    public function behaviors()
    {
        return [
            'verbs' => [
                'class' => VerbFilter::className(),
                'actions' => [
                    'delete' => ['POST'],
                ],
            ],
        ];
    }

    /**
     * Список всіх моделей User.
     * @return mixed
     */
    public function actionIndex()
    {
        $dataProvider = new ActiveDataProvider([
            'query' => User::find(),
        ]);

        return $this->render('index', [
            'dataProvider' => $dataProvider,
        ]);
    }

    /**
     * Відображає модель User.
     * @param string $id
     * @return mixed
     */
    public function actionView($id)
    {
        return $this->render('view', [
            'model' => $this->findModel($id),
        ]);
    }

    /**
     * Створення нового користувача.
     * @return mixed
     */
    public function actionCreate()
    {
        $model = new User();

        if ($model->load(Yii::$app->request->post()) && $model->save()) {
```

```

        return $this->redirect(['view', 'id' => $model->id]);
    } else {
        return $this->render('create', [
            'model' => $model,
        ]);
    }
}

/**
 * Оновлення існуючої моделі User.
 * @param string $id
 * @return mixed
 */
public function actionUpdate($id)
{
    $model = $this->findModel($id);

    if ($model->load(Yii::$app->request->post()) && $model->save()) {
        return $this->redirect(['view', 'id' => $model->id]);
    } else {
        return $this->render('update', [
            'model' => $model,
        ]);
    }
}

/**
 * Видалення користувача .
 */
public function actionDelete($id)
{
    $this->findModel($id)->delete();

    return $this->redirect(['index']);
}

/**
 * Знаходить модель User по ключу БД.
 * If the model is not found, a 404 HTTP exception will be thrown.
 */
protected function findModel($id)
{
    if (($model = User::findOne($id)) !== null) {
        return $model;
    } else {
        throw new NotFoundHttpException('The requested page does not
exist.');
```

Class KMeans.php - реалізація алгоритму K-Means

```

<?php
use Exception;
class KMeans
{
    // ініціалізація поля даних
    protected $data;
    protected $clustered_data;
    // масив центроїдів
    protected $centroids;
    // масив відстаней центроїдів
    protected $centroid_distance;
    // доступні методи для кластеризації
    protected static $ACCEPTED_CLUSTERING_METHODS = [
        'random',
        'forgy',
    ];
    /**
     * ініціалізація
     *
     * @param $data array */
    public function __construct(array $data)
    {
        if (count($data) < 2) {
            throw new Exception('Data must have more than one row');
        }
        $this->data = $data;
    }
    /**
     * основна функція
     * @param $cluster_count integer кількість кластерів
     * @param $method string метод кластеризації
     * @return масив кластеризованих даних (getClusteredData)
     */
    public function cluster($cluster_count, $method = 'forgy')
    {
        if ($cluster_count < 2) {
            throw new Exception('Cluster count must be greater than 1');
        }
        if ($cluster_count > count($this->data)) {
            throw new Exception('Cluster count must be greater than the number
of data points');
        }
        if (!in_array($method, self::$ACCEPTED_CLUSTERING_METHODS)) {
            throw new Exception("Unrecognized method passed into cluster:
{$method}");
        }
        do {
            if (empty($centroids)) {
                $centroids = $this->getInitialCentroids($cluster_count,
$method);
            } else {
                $centroids = $this->calculateCentroids($this->clustered_data);
            }
            $new_clustered_data = array_fill(0, $cluster_count, []);
            foreach ($this->data as $observation) {
                $closest_centroid = $this->
>calculateClosestCentroid($observation, $centroids);
                array_push($new_clustered_data[$closest_centroid],
$observation);
            }
        } while ($this->assignmentConvergenceCheck((array) $this->
>clustered_data, $new_clustered_data) === false);
        $this->centroids = $centroids;

        return $this->getClusteredData();
    }
    /**

```

```

    * Отримання центроїдів
    *
    * @return array list of centroids
    */
public function getCentroids()
{
    if (empty($this->centroids)) {
        throw new Exception('Centroids have not been hydrated yet - run
cluster method first');
    }
    return $this->centroids;
}
/**
 * Отримання кластеризованих даних
 *
 * @return array multi-dimensional array of clustered data
 */
public function getClusteredData()
{
    if (empty($this->clustered_data)) {
        throw new Exception('Clustered data have not been hydrated yet - run
cluster method first');
    }
    return $this->clustered_data;
}
/**
 * Отримання відстаней центроїдів
 * will throw an exception if cluster has not been run yet
 *
 * @return array list
 */
/*
public function getCentroidDistance()
{
    if (empty($this->centroid_distance)) {
        throw new Exception('Centroid distance has not been hydrated yet -
run cluster method first');
    }
    return $this->centroid_distance;
}
*/
/**
 * Ініціалізація центроїдів
 *
 * @param $cluster_count integer кількість кластерів
 * @param $method string тип ініціалізації
 * @return array список центроїдів для ініціалізації
 */
protected function getInitialCentroids($cluster_count, $method)
{
    if ($method == 'forgy') {
        return $this->getForgyInitialization($cluster_count);
    }
    if ($method == 'random') {
        return $this->getRandomInitialization($cluster_count);
    }
}
/**
 * Ініціалізація випадковим чином
 *
 * @param $cluster_count integer кількість кластерів
 * @return array список
 */
protected function getRandomInitialization($cluster_count)
{
    $random_keys = array_rand($this->data, $cluster_count);
    $random_keys = array_flip($random_keys);
    return array_intersect_key($this->data, $random_keys);
}

```

```

/**
 *
 * @param $cluster_count integer кількість кластерів
 * @return array список
 */
protected function getForgeryInitialization($cluster_count)
{
    $data_range = $this->calculateRange($this->data);
    $random_points = [];
    for ($i = 0; $i < $cluster_count; $i++) {
        $random_points[$i] = array_fill(0, count($this->data), null);
        foreach ($data_range as $key => $range) {
            $random_points[$i][$key] = ($range['min'] + lcg_value() *
($range['max'] - $range['min']));
        }
    }
    return $random_points;
}
/**
 * Обчислити центроїди
 *
 * @param $clustered_data array масив кластеризованих даних
 * @return array список центроїдів
 */
protected function calculateCentroids(array $clustered_data)
{
    $centroids = [];
    foreach ($clustered_data as $cluster) {
        $cluster_sum = array_fill(0, count(current($cluster)), 0);
        foreach ($cluster as $observation) {
            foreach ($observation as $key => $value) {
                $cluster_sum[$key] += $value;
            }
        }
        $centroid = array_fill(0, count(current($cluster)), 0);
        foreach ($cluster_sum as $key => $value) {
            $centroid[$key] = $value / count($cluster);
        }
        array_push($centroids, $centroid);
    }
    return $centroids;
}
/**
 * Обчислити найближчу відстань до центроїда
 *
 * @param $observation array
 * @param $centroids array список центроїдів
 * @return integer
 */
protected function calculateClosestCentroid(array $observation, array
$centroids)
{
    $centroid_distance = [];
    foreach ($centroids as $centroid) {
        array_push($centroid_distance, $this-
>calculateDistance($observation, $centroid));
    }
    asort($centroid_distance);
    $centroid_distance = array_keys($centroid_distance);
    return array_shift($centroid_distance);
}
/**
 * Перевірити кластеризовані дані
 *
 * @param $clustered_data array the old holder of clustered_data
 * @param $new_clustered_data array new clustered_data to check
against

```

```

        * @return          boolean whether or not convergence has
occurred
        */
        protected function assignmentConvergenceCheck(array $clustered_data, array
$new_clustered_data)
        {
            if (empty($clustered_data)) {
                $this->clustered_data = $new_clustered_data;
                return false;
            }
            foreach ($clustered_data as $key => $cluster) {
                foreach ($cluster as $observation) {
                    if (!in_array($observation, $new_clustered_data[$key])) {
                        $this->clustered_data = $new_clustered_data;
                        return false;
                    }
                }
            }
            return true;
        }
    /**
     * Допоміжний метод обчислення розмірності даних
     *
     * @param $data array список векторів
     * @return array
     */
    protected function calculateRange($data)
    {
        $data_range = array_fill(0, count(current($data)), ['min' => null, 'max'
=> null]);
        foreach ($data as $observation) {
            $key = 0;
            foreach ($observation as $value) {
                if ($data_range[$key]['min'] === null ||
$data_range[$key]['min'] > $value) {
                    $data_range[$key]['min'] = $value;
                }
                if ($data_range[$key]['max'] === null ||
$data_range[$key]['max'] < $value) {
                    $data_range[$key]['max'] = $value;
                }
                $key++;
            }
        }
        return $data_range;
    }
    /**
     * Обчислення евклідової відстані
     *
     * @param $point_a список векторів a
     * @param $point_b список векторів b
     * @return відстань між векторами
     */
    protected function calculateDistance($point_a, $point_b)
    {
        $distance = 0;
        for ($i = 0, $count = count($point_a); $i < $count; $i++) {
            $difference = $point_a[$i] - $point_b[$i];
            $distance += pow($difference, 2);
        }
        return $distance;
    }
}

```

Login.php - файл авторизації

```

<?php
namespace app\models;
use yii\base\Model;
class Login extends Model
{
    public $email;
    public $password;
    public function rules()
    {
        return [
            [['email', 'password'], 'required'],
            ['email', 'email'],
            ['password', 'string', 'min'=>2, 'max'=>10],
            ['password', 'validatePassword'] //собственная функция для валидации
        ];
    }
    public function attributeLabels()
    {
        return [
            'email' => 'Електронна пошта:',
            'password' => 'Пароль:',
        ];
    }
    public function validatePassword($attribute, $params)
    {
        if (!$this->hasErrors()) // якщо немає помилок валідації
        {
            $user = $this->getUser(); // отримуємо користувача для порівняння
            if (!$user || !$user->validatePassword($this->password))
            {
                //якщо користувач не знайдений в БД
                $this->addError($attribute, 'Пароль или имейл введены неверно');
                //виведення помилки
            }
        }
    }
    public function getUser()
    {
        return User::findOne(['email'=>$this->email]); //отримати користувача по
        //введеному email
    }
}

```

Signup.php

```
<?php

namespace app\models;

use yii\base\Model;

/**
 *
 */
class Signup extends Model
{
    public $email;
    public $password;

    public function rules(){
        return[
            [['email','password'],'required'],
            [['email','email'],
            ['email','unique','targetClass'=>'app\models\User'],
            ['password','string','min'=>2,'max'=>10]

        ];
    }
    /**
     * @inheritdoc
     */
    public function attributeLabels()
    {
        return [
            'email' => 'Електронна пошта:',
            'password' => 'Пароль:',
        ];
    }
    public function signup(){
        $user = new User();
        $user->email = $this->email;
        $user->setPassword($this->password);

        return $user->save(); // true or false
    }
}
```

User.php - модель користувача

```

<?php

namespace app\models;

use Yii;
use yii\db\ActiveRecord;
use yii\web\IdentityInterface;

/**
 * Клас моделі для таблиці "tbl_user".
 *
 * @property string $id
 * @property string $login
 * @property string $password
 * @property string $email
 * @property string $creation_date
 * @property string $reputation
 * @property string $location
 * @property string $about
 *
 * @property TblComments[] $tblComments
 * @property TblPost[] $tblPosts
 */
class User extends ActiveRecord implements IdentityInterface
{

    public static function tableName()
    {
        return 'tbl_user';
    }
    public function rules()
    {
        return [
            [['password', 'email', 'creation_date'], 'required'],
            [['creation_date'], 'safe'],
            [['reputation'], 'integer'],
            [['about'], 'string'],
            [['login', 'location'], 'string', 'max' => 40],
            [['password'], 'string', 'max' => 10],
            [['email'], 'string', 'max' => 20],
        ];
    }

    public function attributeLabels()
    {
        return [
            'id' => 'ID',
            'login' => 'Login',
            'password' => 'Password',
            'email' => 'Email',
            'creation_date' => 'Creation Date',
            'reputation' => 'Reputation',
            'location' => 'Location',
            'about' => 'About',
        ];
    }

    public function setPassword($password){
        return $this->password = sha1($password);
    }
    public function setEmail($email){
        return $this->email = $email;
    }

    public function validatePassword($password)
    {

```

```
        return $this->password === sha1($password);
    }
    //=====
    public static function findIdentity($id)
    {
        return self::findOne($id);
    }
    public function getId()
    {
        return $this->id;
    }
    public static function findIdentityByAccessToken($token, $type = null)
    {
    }
    public function getAuthKey()
    {
    }
    public function validateAuthKey($authKey)
    {
    }

    public function getTblComments()
    {
        return $this->hasMany(TblComments::className(), ['user_id' => 'id']);
    }

    /**
     * @return \yii\db\ActiveQuery
     */
    public function getTblPosts()
    {
        return $this->hasMany(TblPost::className(), ['author_id' => 'id']);
    }
}
```

ContactUs.php - модель сторінки зворотнього зв'язку

```
<?php

namespace app\models;

use Yii;
use yii\base\Model;
class ContactForm extends Model
{
    public $name;
    public $email;
    public $subject;
    public $body;
    public $verifyCode;

    /**
     * @return масив правил валідації
     */
    public function rules()
    {
        return [
            // name, email, subject and body are required
            [['name', 'email', 'subject', 'body'], 'required'],
            // email has to be a valid email address
            ['email', 'email'],
            // verifyCode needs to be entered correctly
            ['verifyCode', 'captcha'],
        ];
    }

    /**
     * @return масив міток атрибутів
     */
    public function attributeLabels()
    {
        return [
            'name'=>'Імя:',
            'email'=>'Електронна пошта:',
            'subject'=>'Тема',
            'body'=>'Текст повідомлення',
            'verifyCode' => 'Код верифікації',
        ];
    }

    /**
     * Надсилає повідомлення
     * @param string $email
     * @return boolean пройшла модель валідацію чи ні
     */
    public function contact($email)
    {
        if ($this->validate()) {
            Yii::$app->mailer->compose()
                ->setTo($email)
                ->setFrom(['$this->email => $this->name'])
                ->setSubject($this->subject)
                ->setTextBody($this->body)
                ->send();

            return true;
        }
        return false;
    }
}
```

login.php - представлення сторінки входу на сайт

```

<?php

/* @var $this yii\web\View */
/* @var $form yii\bootstrap\ActiveForm */
/* @var $model app\models\LoginForm */

use yii\helpers\Html;
use yii\widgets\ActiveForm;

$this->title = 'Вхід на сайт';
$this->params['breadcrumbs'][] = $this->title;
?>
<div class="site-login">
    <h1><?= Html::encode($this->title) ?></h1>

    <p>Будь-ласка заповніть поля, щоб ввійти на сайт:</p>

    <?php $form = ActiveForm::begin([
        'id' => 'login-form',
        'options' => ['class' => 'form-horizontal'],
        'fieldConfig' => [
            'template' => "{label}\n<div class=\"col-lg-3\">{input}</div>\n<div
class=\"col-lg-8\">{error}</div>",
            'labelOptions' => ['class' => 'col-lg-1 control-label'],
        ],
    ]); ?>

    <?= $form->field($login_model, 'email')->textInput(['autofocus' =>
true]) ?>

    <?= $form->field($login_model, 'password')->passwordInput() ?>

    <div class="form-group">
        <div class="col-lg-offset-1 col-lg-11">
            <?= Html::submitButton('Ввійти', ['class' => 'btn btn-primary',
'name' => 'login-button']) ?>
        </div>
    </div>

    <?php ActiveForm::end(); ?>

</div>

```

Main.php -

```

<?php

/* @var $this \yii\web\View */
/* @var $content string */

use yii\helpers\Html;
use yii\bootstrap\Nav;
use yii\bootstrap\NavBar;
use yii\widgets\Breadcrumbs;
use app\assets\AppAsset;

AppAsset::register($this);
?>
<?php $this->beginPage() ?>
<!DOCTYPE html>
<html lang="<?= Yii::$app->language ?>">
<head>
    <meta charset="<?= Yii::$app->charset ?>">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <?= Html::csrfMetaTags() ?>
    <title><?= Html::encode($this->title) ?></title>
    <?php $this->head() ?>
</head>
<body>
<?php $this->beginBody() ?>

<div class="wrap">
    <?php
    NavBar::begin([
        'brandLabel' => '',
        'brandUrl' => Yii::$app->homeUrl,
        'options' => [
            'class' => 'navbar-inverse navbar-fixed-top',
        ],
    ]);
    echo Nav::widget([
        'options' => ['class' => 'navbar-nav navbar-right'],
        'items' => [
            ['label' => 'Головна', 'url' => ['/site/index']],
            ['label' => 'Про розробника', 'url' => ['/site/about']],
            ['label' => 'Зворотній зв'язок', 'url' => ['/site/contact']],
            Yii::$app->user->isGuest ? (
                ['label' => 'Ввійти', 'url' => ['/site/login']]
            ) : (
                <li>
                    . Html::beginForm(['/site/logout'], 'post')
                    . Html::submitButton(
                        'Вийти (' . Yii::$app->user->identity->email . ')',
                        ['class' => 'btn btn-link']
                    )
                    . Html::endForm()
                . </li>
            )
        ],
    ]);
    NavBar::end();
?>

    <div class="container">
        <?= Breadcrumbs::widget([
            'links' => isset($this->params['breadcrumbs']) ? $this-
>params['breadcrumbs'] : [],
        ]) ?>
        <?= $content ?>
    </div>
</div>

```

```
<footer class="footer">
  <div class="container">
    <p class="pull-left">&copy; Ширяев Олег Игоревич <?= date('Y') ?></p>
    <p class="pull-right"><?= Yii::powered() ?></p>
  </div>
</footer>

<?php $this->endBody() ?>
</body>
</html>
<?php $this->endPage() ?>
```

КБПЗ_2024

Post.php - модель Post

```

<?php
namespace app\models;
use Yii;
/**
 * Модель класу для таблиці "tbl_post".
 */
class Post extends \yii\db\ActiveRecord
{
    /**
     * @inheritdoc
     */
    public static function tableName()
    {
        return 'tbl_post';
    }
    public function rules()
    {
        return [
            [['title'], 'required'],
            [['content', 'publish_status'], 'string'],
            [['score', 'answer_count', 'tag_id', 'author_id'], 'integer'],
            [['creation_date'], 'safe'],
            [['title'], 'string', 'max' => 255],
            [['author_id'], 'exist', 'skipOnError' => true, 'targetClass' =>
                User::className(), 'targetAttribute' => ['author_id' => 'id']],
            [['tag_id'], 'exist', 'skipOnError' => true, 'targetClass' =>
                Tag::className(), 'targetAttribute' => ['tag_id' => 'id']],
        ];
    }
    public function attributeLabels()
    {
        return [
            'id' => 'ID',
            'title' => 'Title',
            'content' => 'Content',
            'score' => 'Score',
            'answer_count' => 'Answer Count',
            'tag_id' => 'Tag ID',
            'author_id' => 'Author ID',
            'publish_status' => 'Publish Status',
            'creation_date' => 'Creation Date',
        ];
    }
    /**
     * отримати коментарі
     */
    public function getTblComments()
    {
        return $this->hasMany(TblComments::className(), ['post_id' => 'id']);
    }
    /**
     * отримати автора
     */
    public function getAuthor()
    {
        return $this->hasOne(TblUser::className(), ['id' => 'author_id']);
    }
    /**
     * отримати тег
     */
    public function getTag()
    {
        return $this->hasOne(TblTags::className(), ['id' => 'tag_id']);
    }
}

```