НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Навчально-науковий інститут атомної та теплової енергетики
Кафедра цифрових технологій в енергетиці

"На правах рукопису"
УДК 004.65

«До захисту допущено»
Завідувач кафедри
_____ Наталія АУШЕВА
"___"_____2022р.

Магістерська дисертація

зі спеціальності - 122 Комп'ютерні науки
за освітньо-професійною програмою магістерської підготовки -
Комп'ютерний моніторинг та геометричне моделювання процесів і систем

на тему Статистичний аналіз різнорідних масивів даних у системі санітарно-гігієнічного моніторингу

Виконав: студент 2 курсу, групи ТР-13мп
Левицький Ілля Вячеславович _____
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник доцент к.т.н Полягушко Л. Г. _____
(посада, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Рецензент _____ _____
(посада, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

Київ - 2022

Національний технічний університет України
“ Київський політехнічний інститут ім. Ігоря Сікорського”

Навчально-науковий інститут атомної та теплової енергетики
Кафедри цифрових технологій в енергетиці
Рівень вищої освіти другий, магістерський
За освітньою програмою "Комп’ютерний моніторинг та геометричне моделювання процесів і систем"
Спеціальності 122 Комп’ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Наталія АУШЕВА
(підпис)
«____» _____ 2022р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ

Левицькому Іллі Вячеславовичу
(прізвище, ім’я, по батькові)

1. Тема дисертації Статистичний аналіз різнорідних масивів даних у системі санітарно-гігієнічного моніторингу

Науковий керівник Полягушко Любов Григорівна, к.т.н
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від “07”листопада 2022 року № 4067-с

2. Строк подання студентом дисертації 7 грудня 2022року

3. Вихідні дані до роботи мова програмування python3, платформа NodeJS, фреймворк Express, бібліотека Pandas, бібліотека NumPy, бібліотека SqlConnector, бібліотека MatplotLib, бібліотека Seaborn, бібліотека React, каскадна таблиця стилів CSS3, мова розмітки HTML5, система керування базами даних MySQL

4. Перелік питань, які потрібно розробити
1) Отримати та перетворити вхідні дані у швидкий робочий формат для подальшого аналізу
2) Визначити оптимальні методи математичної статистики для проведення статистичного аналізу
3) Розробити програмне забезпечення
4) Описати методику роботи користувачів з системою
5) Розробити стартап-проект

5. Ілюстративний матеріал містить Актуальність теми дослідження, об’єкт, мета та завдання, об’єкт та предмет дослідження, аналіз аналогічних методів та систем, методи дослідження, загальна концепція, структура модулів, алгоритм, схема інформаційних потоків, USE CASE, клієнтська частина, серверна частина, база даних, початок роботи з

системою, експериментальне дослідження результатів, практичне значення, висновки

6. Перелік публікацій Полягушко Л.Г, Левицький І.В., Костриця С.В. Еталонне тестування та аналіз реляційних баз даних моніторингових систем з великими обсягами даних. III Міжнародна студентська конференція "Розвиток суспільства та науки в умовах цифрової трансформації", Луцьк, Україна, 16 грудня 2022. Полягушко Л.Г, Левицький І.В., Костриця С.В. Методи оптимізації пошукових запитів та покращення швидкодії реляційних баз даних. Вісник Вінницького політехнічного інституту (до друку)

7. Дата видачі завдання «15» вересня 2022р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Вирішення організаційних питань початку практики: оформлення документації, встановлення каналів комунікації з керівником дипломної роботи	01.09.22р.- 07.09.22р.	
2.	Підготовка матеріалів, що були розроблені за час навчання	01.09.22р.- 14.09.22р.	
3.	Побудова моделей, розробка методів та алгоритмів програмного забезпечення	14.09.22р.- 11.10.22р.	
4.	Демонстрація програмного продукту керівнику. Виправлення помилок	12.10.22р.- 20.10.22р.	
5.	Захист програмного продукту комісії. Виправлення помилок.	21.10.22р.- 25.10.22р.	
6.	Залік	26.10.22р.	
7.	Розроблення стартап-проекту	27.10.22р.- 15.11.22р.	
8.	Підготовка магістерської дисертації до передзахисту та передача її керівнику. Постановка доповіді	16.11.22р.- 20.11.22р.	
9.	Передзахист магістерської дисертації комісії. Перевірка нормоконтролю	21.11.22р.- 30.11.22р.	
10.	Подання документів на кафедру	05.12.22р.	
11.	Захист	12.12.22р.- 18.12.22р.	

Студент _____ Левицький І. В.
(підпис) (прізвище та ініціали)

Науковий керівник _____ Полягушко Л. Г.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Магістерська дисертація за темою “Статистичний аналіз різнорідних масивів даних у системі санітарно-гігієнічного моніторингу” виконана студентом кафедри цифрових технологій в енергетиці НН ІАТЕ Левицьким Іллею Вячеславовичем із спеціальності 122 “Комп’ютерні науки” за освітньо-професійною програмою “ Комп’ютерний моніторинг та геометричне моделювання процесів та систем” та складається зі: вступу; постановки задачі статистичного аналізу різнорідних масивів даних у системі санітарно-гігієнічного моніторингу; аналізу наявних програмних продуктів, математичних підходів та видів візуалізації для статистичного аналізу даних; засобів розробки; опису програмної реалізації; стартапу; висновків до розділів; загальних висновків; списку використаних джерел, який налічує 40 джерел та 1 додаток. Загальний обсяг роботи 101 сторінка.

Актуальність теми. Здоров’я є одним з найважливіших параметрів високого рівня життя населення та сталого розвитку держави. Навколишнє середовище є одним з факторів, що впливає на здоров’я населення [41], а статистичний аналіз у системі медико-санітарного моніторингу є зарекомендованим методом для обробки даних, їх аналізу та подальшого приведення до систематизованого вигляду. Тому актуальною задачею є використання статистичного аналізу у системі медично-санітарного моніторингу для відслідковування чинників впливу на населення.

Метою роботи є створення програмного забезпечення, що збирає медичні та екологічні дані у системі «Комплексного Еко-Енерго-Економічного моніторингу» (КЕЕЕМ), аналізує їх на віддаленому сервері, а результати обчислень повертає користувачу у вигляді матриці або графіку.

Завдання роботи:

1. Обрати формат готового відображення, формули для обробки даних, тип даних та засоби розробки.
2. Отримати дані з віддаленої бази даних та перетворити їх у формат для найшвидшого опрацювання на сервері.

3. Розробити програмне забезпечення для статистичного аналізу на основі отриманих даних, отриманий результат візуалізувати для користувача.

Об'єкт дослідження. Методи статистичного аналізу масивів даних, отриманих з системи моніторингу.

Предмет дослідження. Статистичний аналіз даних у системі медико-санітарного моніторингу.

Методи дослідження. Метод визначення коефіцієнтів кореляційної матриці, метод множинної вибірки за допомогою коефіцієнту кореляції та напівкореляції за формулами Пірсона та Спірмена.

Практичне значення одержаних результатів. Розроблене програмне забезпечення є інтегрованим у існуючу систему КЕЕЕМ для полегшення моніторингу ситуації здоров'я населення, що пов'язане з навколишніми чинниками. Також ПЗ покликане полегшити прийняття рішень для відповідних експертів, а також скоротити кількість розбіжностей, що виникають між спостереженнями (фактами) та припущенням експерта (очікуваними результатами).

Апробація результатів дисертації.

Полягушко Л.Г, Левицький І.В., Кострица С.В. Еталонне тестування та аналіз реляційних баз даних моніторингових систем з великими обсягами даних. III Міжнародна студентська конференція "Розвиток суспільства та науки в умовах цифрової трансформації", Луцьк, Україна, 16 грудня 2022.

Левицький І.В., Полягушко Л.Г. Единий реєстр наукових статей. V International Scientific and Practical Conference "Topical Issues of Modern Science, Society and Education", Харків, Україна, 28 листопада 2021.

Ключові слова. Реляційна база даних, статистичний аналіз даних, Комплексний Еко-Енерго-Економічний моніторинг, кореляційна матриця, формула кореляційного коефіцієнта Пірсона, формула Спірмена, бібліотека NumPy, бібліотека Pandas, бібліотека візуалізації Seaborn, бібліотека візуалізації MatplotLib.

ABSTRACT

The master's dissertation on the topic "Statistical analysis of heterogeneous data sets in the system of sanitary and hygienic monitoring" was completed by the student of the Department of Digital Technologies in Energy of NN IATE Levytskyi Illia Viacheslavovych majoring in 122 "Computer Sciences" under the educational and professional program "Computer Monitoring and Geometric modeling of processes and systems" and consists of: introduction; task setting for statistical analysis of heterogeneous data sets in the system of sanitary and hygienic monitoring; analysis of available software products, mathematical approaches and types of visualization for statistical data analysis; development tools; description of software implementation; startup; conclusions to sections; general conclusions; of the list of used sources, which includes 40 sources and 1 appendix. The total volume of work is 101 pages.

Actuality of theme. Health is one of the most important parameters of a high standard of living of the population and sustainable development of the state. The environment is one of the factors affecting the health of the population [1], and statistical analysis in the system of medical and sanitary monitoring is a recommended method for processing data, analyzing them and further bringing them to a systematized form. Therefore, it is an urgent task to use statistical analysis in the system of medical and sanitary monitoring to monitor the factors affecting the population.

The purpose of the work is to create software that collects medical and environmental data in the system of "Complex Eco-Energy-Economic Monitoring" (KEEEM), analyzes them on a remote server, and returns the results of calculations to the user in the form of a matrix or graphic.

Job tasks:

1. Choose the display format, formulas for data processing, data type and development tools.

2. Get data from a remote database and convert it into a format for the fastest processing on the server.

3. Develop software for statistical analysis based on the obtained data, visualize the obtained result for the user.

Object of study. Methods of statistical analysis of data arrays obtained from the monitoring system.

Subject of study. Statistical analysis of data in the medical and sanitary monitoring system.

Research methods. The method of determining the coefficients of the correlation matrix, the method of multiple sampling using the correlation coefficient and semi-correlation according to the Pearson and Spearman formulas.

Practical significance of the obtained results. The developed software is integrated into the existing KEEEM system to facilitate the monitoring of the population's health situation related to environmental factors. Also, the software is designed to facilitate decision-making for relevant experts, as well as to reduce the number of discrepancies between observations (facts) and the expert's assumption (expected results).

Approbation of the results of the dissertation.

Polyagushko L.H., Levitskyi I.V., Kostrytsia S.V. Benchmark testing and analysis of relational databases of monitoring systems with large volumes of data. III International Student Conference "Development of Society and Science in the Conditions of Digital Transformation", Lutsk, Ukraine, December 16, 2022.

Levitskyi I.V., Polyagushko L.H. Unified register of scientific articles. V International Scientific and Practical Conference "Topical Issues of Modern Science, Society and Education", Kharkiv, Ukraine, November 28, 2021.

Keywords. Relational database, statistical data analysis, Complex Eco-Energy-Economic monitoring, correlation matrix, Pearson's correlation coefficient formula, Spearman's formula, NumPy library, Pandas library, Seaborn visualization library, MatplotLib visualization library.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ 10
ВСТУП 11
1 ПОСТАНОВКА ЗАДАЧІ СТАТИСТИЧНОГО АНАЛІЗУ РІЗНОРІДНИХ МАСИВІВ ДАНИХ У СИСТЕМІ САНІТАРНО-ГІГІЄНІЧНОГО МОНІТОРИНГУ 14
1.1 Призначення програмного продукту 14
1.2 Потенційні користувачі 17
1.3 Вхідні та вихідні дані програми 18
1.4 Задача статистичного аналізу різнорідних масивів даних у системі санітарно-гігієнічного моніторингу 21
2 АНАЛІЗ НАЯВНИХ ПРОГРАМНИХ ПРОДУКТІВ, МАТЕМАТИЧНИХ ПІДХОДІВ ТА ВИДІВ ВІЗУАЛІЗАЦІЇ ДЛЯ СТАТИСТИЧНОГО АНАЛІЗУ ДАНИХ 22
2.1 Наявні рішення схожих проблем методами SAS 22
2.1.1 Загальна інформація про можливості SAS 22
2.1.2 Вступ до статистичного аналізу стану здоров’я методами SAS 23
2.1.3 Робота з даними у CSV-форматі 24
2.1.4 Практичний приклад застосування пакету аналізу SAS 27
2.2 Наявні рішення схожих проблем методами IBM SPSS Statistics 30
2.2.1 Загальна інформація про можливості SPSS Statistics 30
2.2.2 Робота з даними, отриманими мовою запитів SQL 33
2.2.3 Практичний приклад застосування IBM SPSS Statistics 34
2.3 Аналіз математичних методів встановлення коефіцієнту кореляції 38
2.3.1 Формула Пірсона 38
2.3.2 Формула Спірмена 38
2.4 Аналіз та оцінка методів візуалізації результатів, отриманих шляхом статистичного аналізу 40

2.4.1 Діаграма розсіювання 40
2.4.2 Матриця кореляції 41
3 ЗАСОБИ РОЗРОБКИ 43
3.1 Мова програмування Python та бібліотеки мови, що використовуються при розробці ПЗ 44
3.2 Формат вхідних даних 51
3.3 Система керування базами даних 51
3.4 Інструменти для інтеграції ПЗ 54
3.5 Редактори 55
4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ТА РОБОТИ СИСТЕМИ 59
4.1 Частина ПЗ, що забезпечує отримання та перетворення даних 59
4.2 Частина ПЗ, що забезпечує статистичний аналіз та візуалізацію оброблених даних 62
4.3 Робота користувача з системою 66
5 СТАРТАП-ПРОЄКТ 70
5.1 Опис ідеї стартапу 70
5.2 Технологічний аудит ідеї проекту 72
5.3 Аналіз ринкових можливостей запуску стартап-проекту 73
5.4 Розробка ринкової стратегії проекту 81
5.5 Розробка маркетингової компанії проекту 83
ВИСНОВКИ 88
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 89
ДОДАТОК А 93

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧКИ

КЕЕЕМ	Комплекс еколого-економіко-енергетичного моніторингу
СКБД	Система керування базами даних
ПЗ	Програмне забезпечення
Фреймворк	Інфраструктура програмних рішень, що полегшує розробку складних систем
SQL	(англ. Structured Query Language) – структурована мова запитів
HTML	Мова гіпертекстової розмітки
CSS	(англ. Cascading Style Sheet) — каскадна таблиця стилів
NodeJS	Платформа для розробки рішень на основі JS
NumPy	Програмна математична бібліотека python
Pandas	Програмна математично-наукова бібліотека python
MatplotLib	Бібліотека візуалізації python
Seaborn	Покращена оболонка бібліотеки MatplotLib

ВСТУП

Здоров’я є одним з найважливіших параметрів високого рівня життя населення та сталого розвитку держави [1]. Навколишнє середовище є одним з факторів, що впливає на здоров’я населення, тому для прийняття правильних та своєчасних рішень дуже важливо проводити багатовекторній статистичний аналіз екологічних та медичних показників регіону [2].

Для того, щоб експертам можна було відслідковувати екологічний стан відповідальної області у режимі реального часу була створена система «Комплексний Еко-Енерго-Економічний Моніторинг». А для збору інформації у системі, її аналізу та візуалізації результатів актуальним стало створення програмного забезпечення, що вирішує дані задачі.

Готове програмне забезпечення є кросплатформеним, оскільки розроблялося під існуючу систему Еко-Енерго-Економічного моніторингу, використовує обґрунтовані формули при аналізі та обробці інформації та має зрозумілий вигляд оброблених даних для користувача.

Для кращого розуміння яким має бути готовий продукт, треба, в першу чергу, зрозуміти як, за рекомендаціями, повинен виглядати дизайн користувача:

— оброблені дані програмного забезпечення повинні знаходитися у віджеті статистичного аналізу по областям України, оскільки віджети передбачають персоналізацію під завдання, відповідним до змісту розташування (медик аналізує вплив чинників на здоров’я людини по областям країни) та помітним для користувача;

— віджет повинен узагальнювати основну особливість програмного забезпечення та мати фільтри для вибору користувача (окрім області), для обмеження відображення інформації потрібно ввести елементи вибору окрім випадаючого меню;

— результати обробки та аналізу даних потрібно виводити тільки за запитом користувача, щоб не перенавантажувати сервер – для цього краще обрати окреме діалогове вікно виведення.

Для того, щоб працювати з даними для початку необхідно їх отримати та перетворити. Дані, що необхідно отримати знаходять у базі даних системи Комплексного Еко-Енерго-Економічного Моніторингу, запит, що надходить до бази даних формується вибором користувачі у інтерфейсі застосунка, а саме, у зазначеному раніше, діалоговому вікні. Після того, як користувач обирає параметри та натискає кнопку «Відобразити», формується SQL-запит до бази даних, розроблене програмне забезпечення отримує вхідні дані, перетворює їх у формат для швидкого опрацювання та передає до модулю обробки даних.

Для розроблення програмного забезпечення, що виконує статистичний аналіз в залежності від вхідних даних були проаналізовані методи аналізу залежності ризиків розвитку злоякісних новоутворень від оточуючої середи по області, в тому числі, забрудненості води та концентрації зважених частинок у повітрі [29]. Отриманий набір даних був візуалізований, зображення з візуалізацією є відповіддю на запит користувача.

Об’єктом дослідження є методи статистичного аналізу масивів даних, отриманих з бази даних системи моніторингу.

Предметом дослідження є створення програмного забезпечення, що демонструє результати статистичного аналізу даних у системі медико-санітарного моніторингу.

Методами дослідження для досягнення поставленої мети є визначення коефіцієнтів кореляційної матриці – даний метод використовувався для виявлення гами стохастичних зв’язків між випадковими величинами (є універсальним методом, тобто ігнорує унікальні випадки), також у роботі використовувалися методи множинної вибірки за допомогою коефіцієнту кореляції та напівкореляції за формулами Пірсона та Спірмена, особливо корисні для знаходження парної та непарної кореляції для даних у областях з невеликою кількістю випадків захворювань.

Розроблене програмне забезпечення є інтегрованим open-source проектом (що дозволить обслуговувати та змінювати функціонал застосунку) у існуючу систему

КЕЕЕМ для полегшення моніторингу ситуації здоров’я населення, що пов’язане з навколишніми чинниками. Також ПЗ покликане полегшити прийняття рішень для відповідних експертів і скоротити кількість розбіжностей, що виникають між спостереженнями (фактами) та припущенням експерта (очікуваними результатами).

У результаті визначений майбутній функціонал, алгоритм та методи дослідження для створення програмного забезпечення для статистичного аналізу даних у медико-санітарній частині системи Комплексного Еко-Енерго-Економічного моніторингу. Даний проект є гнучкий, тому при розширенні функціоналу та методів дослідження може бути збільшеним до вимог замовника проекту.

1 ПОСТАНОВКА ЗАДАЧІ СТАТИСТИЧНОГО АНАЛІЗУ РІЗНОРІДНИХ МАСИВІВ ДАНИХ У СИСТЕМІ САНІТАРНО-ГІГІЄНІЧНОГО МОНІТОРИНГУ

Завданням роботи є створення програмного продукту, що проводить статистичний аналіз на основі зібраних даних у медико-санітарній частині системи Комплексного Еко-Енерго-Економічного моніторингу. Розроблений модуль дасть змогу користувачеві (експерту у медико-санітарній частині) отримати доступ до підготовленого статистичного аналізу по заданим параметрам у області, що вплине на швидкість прийняття рішень користувача.

1.1 Призначення програмного продукту

Призначення статистичного аналізу даних у системі медико-санітарного моніторингу полягає у розширенні існуючого функціоналу системи Комплексного Еко-Енерго-Економічного Моніторингу. На сьогоднішній день існує багато розроблених інформаційних систем, що займаються моніторингом довкілля, в тому числі радіаційним, атмосферним, літосферним, економічним та моніторингом здоров’я людини. Враховуючи, концепцію сталого інноваційного розвитку, яким, в тому числі, займається ООН [41], постало питання створення комплексної системи, що могла би відповідати всім вимогам у одному веб-додатку.

Система КЕЕЕМ реалізує дану задачу з можливістю використання веб-застосунку різними експертами:

— Екологом;

— економістом;

— юристом;

— аналітиком (адміністратором);

— медиком.

На рисунку 1.1 показана загальна схема функціонування проекту:

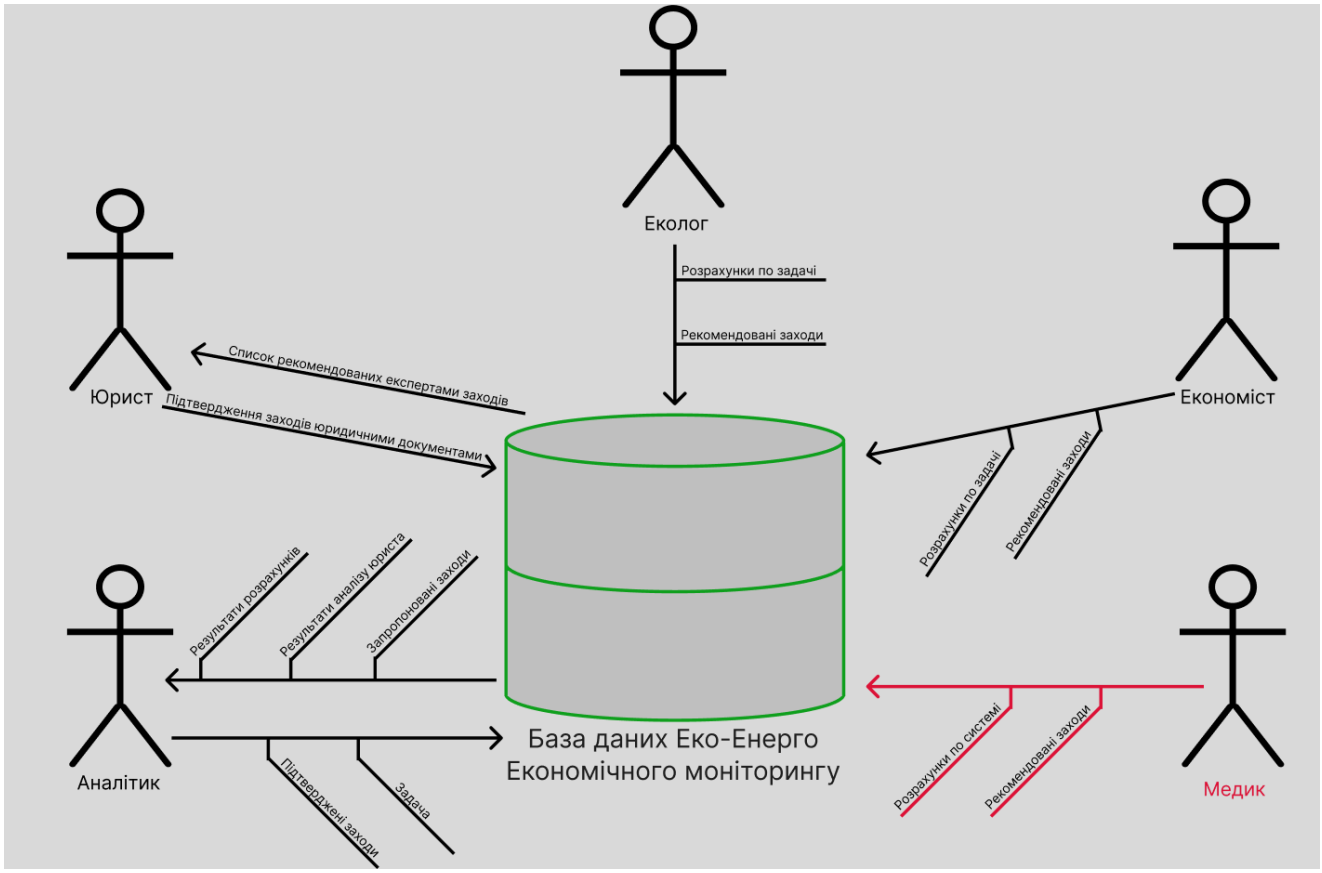

Рисунок 1.1 – Загальна схема системи Комплексного Еко-Енерго-Економічного моніторингу

Схема ілюструє такі поняття як заходи, задачі та розрахунки по задачі. Щоб краще зрозуміти схему, варто розглянути ці терміни більш детально.

Задача представляє із себе будь-яку екологічну ситуацію (у тому числі здоров'я населення [1]), яка потребує розгляду. Задачі можуть поділятися на екстрені чи планові, в залежності від ситуації. В загальному випадку задачі моніторингу є плановими і розраховані на прийняття рішень в залежності від результату зібраних даних та накопиченої статистики.

Заходи уособлюють собою певні дії, що направлені на покращення екологічної ситуації або вирішення екстрених задач. Як показано на схемі, усі заходи потребують підтвердження експерта-юриста.

Ресурс – це людські або грошові ресурси або матеріальні речі, що необхідні для вирішення поставленої задачі.

Як бачимо на рисунку 1.1 актор «Медик» та його гілка дій виокремлені червоним – тут і надалі червоний колір на загальних схемах буде означати робочу область у загальній системі КЕЕЕМ. Медик спирається на зібрані дані протягом певного проміжку часу, обирає параметри і проводить розрахунки.

У даному випадку мета модулю статистичного аналізу у медико-санітарній частині полягає у полегшенні роботи медика-експерта та пришвидшенні прийняття рішень. Загальна схема взаємодії експерта з модулем статистичного аналізу різнорідних масивів даних у системі санітарно-гігієнічного моніторингу показана на рисунку 1.2.

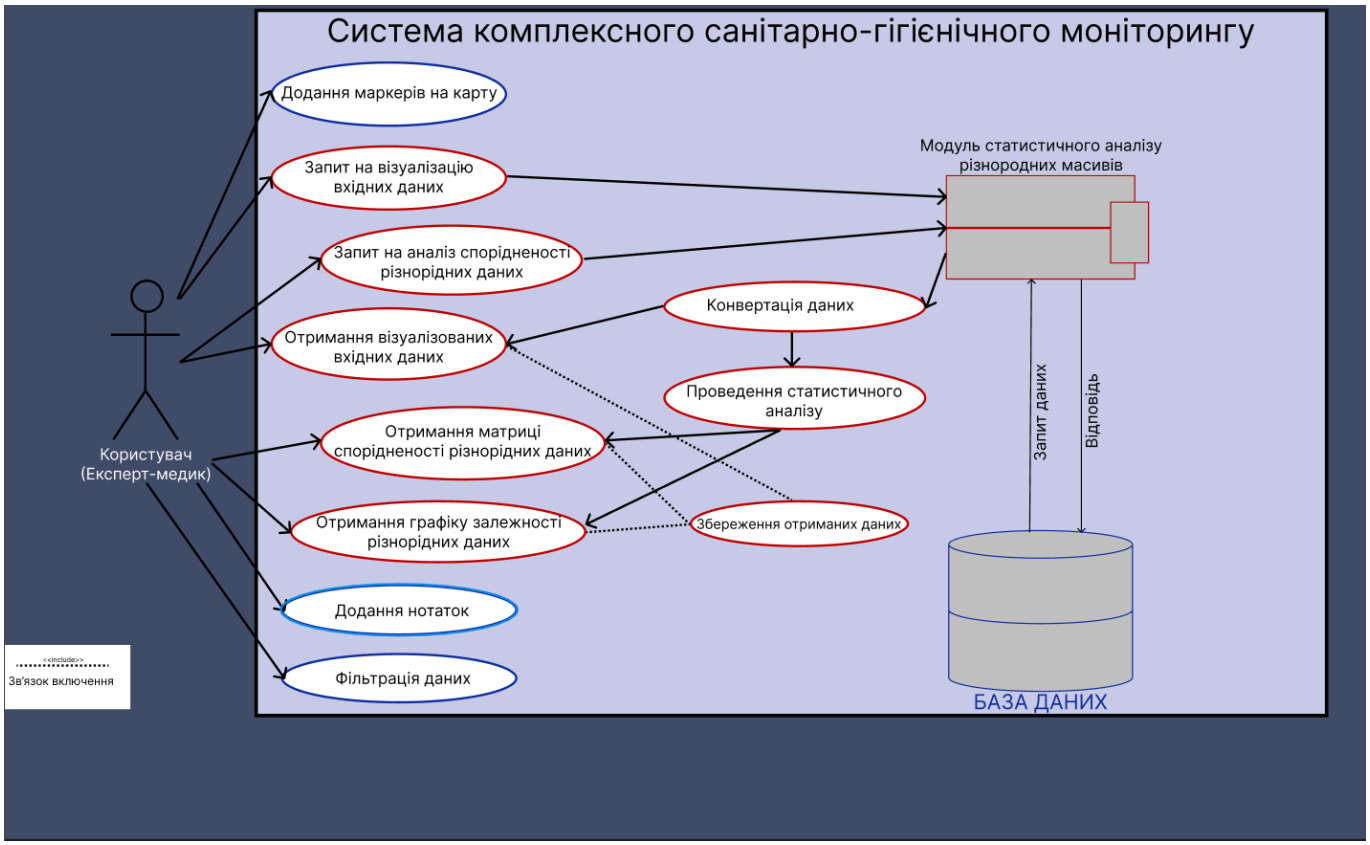

Рисунок 1.2 – Схема роботи експерта-медика з базою даних через модуль статистичного аналізу даних у медико-санітарній системі

На схемі взаємодії показана велика кількість елементів, для кращого розуміння розглянемо детальніше ключові з них.

База даних – база даних КЕЕЕМ, до неї надходить запит даних, що обмежується модулем статистичного аналізу, а від неї надходить відповідь у вигляді необхідних

даних. Під обмеженими даними мається на увазі тільки ті дані, що відносяться до медико-санітарної частини.

Модуль статистичного аналізу різнорідних масивів – розроблене ПЗ, що покликане спростити роботу експерту-медику і має на схемі наступні функції:

— конвертація даних – під конвертацією даних мається на увазі перетворення вхідних даних до робочого формату, тобто прибирання зайвого і перетворення їх у формат для пришвидшеної обробки;

— проведення статистичного аналізу – основна частина програмного забезпечення, виконує обробку даних;

— збереження отриманих даних – можливість зберегти отримані дані користувачеві;

— створення візуалізованих вхідних даних – ПЗ забезпечує візуалізацію вхідних даних;

— створення візуалізованої матриці спорідненості різнорідних даних – ПЗ забезпечує візуалізацію оброблених даних;

— створення графіку залежності різнорідних даних – ПЗ створює графік залежності різнорідних даних.

Також користувач може фільтрувати дані через користувацький інтерфейс, чим робити запити до бази даних більш специфічними.

1.2 Потенційні користувачі

До потенційних користувачів програмного продукту та зацікавлених у результатах статистичного аналізу у системах медико-санітарного моніторингу у першу чергу варто віднести державні медичні установи, дослідників-медиків та інші медичні заклади та установи.

Державні медичні установи можуть мати різну мотивацію у використанні системи, наприклад, дане ПЗ актуальне для спостереження – інформацію за минулі роки можуть використовувати для передбачення кількості випадків захворювань у

наступному, враховуючи зменшення або приріст нових захворювань. А щоб спростити завдання медику та знизити вірогідність помилки при формуванні висновків у ПЗ є візуалізація кореляції вхідних медичних даних та подання базових даних у графіках.

Дослідникам, насамперед, буде корисним використання модуля статистичного аналізу для знаходження певного патерну захворюваності по чинникам чи по регіонам для використання у науково-дослідницьких роботах.

Програмне забезпечення передбачене, насамперед у використанні в якості складової частини системи КЕЕЕМ, але, враховуючи, що ПЗ має відкритий вихідний код, інші розробники можуть використовувати його на базі своїх програм моніторингу чи готових даних.

1.3 Вхідні та вихідні дані програми

Вхідні дані – це дані, що завдяки зовнішнім API видобуваються у базу даних системи і використовуються при побудові обчислень на їх базі:

— результати підрахунків;

— характеристика об’єкта;

— міста чи області;

— викиди в атмосферу;

— середовище, оточення;

— документ події;

— ресурс події;

— формули;

— ГДК;

— залежності на мапі;

— медична статистика;

— очікуваний результат;

— значення параметра;

— POI;

— точки полігону;

— ресурси.

Результати підрахунків визначаються як результат виконання математичних формул (у таблиці формули зазначені за їх id), а також містять інформацію про експерта, що провів останні розрахунки та опис за його бажанням.

Викиди в атмосферу містять у собі точний день вимірів, результати вимірів – середні та максимальні. а також одиниці вимірювання.

Дані про середу та оточення містять у собі рівні вимірювання, всього є декілька видів середовищ:

— атмосфера;

— питна вода;

— технічна вода;

— ґрунт;

— господарський ґрунт;

— поверхневі води;

— стратосфера.

У кожного середовища є свій унікальний ідентифікатор, у цій же таблиці є види оточення, що прив’язані до своєї середи, наприклад, до питної води відносяться водопровідна вода, вода с колодязів та каптажів джерел, фасована з пунктів розливу та вода з бюветів.

У таблиці подій знаходяться різні події, що впливають на екологічний та економічний стан оточуючої середи. Для того, щоб подія була розглянута аналітиком, необхідно, щоб інший експерт та юрист підтвердили цю подію зі своїх кабінетів.

Документ події зберігає у собі ідентифікатор події та код нормативного документу, що прив’язаний до неї.

Ресурс події зберігає у собі ідентифікатор події та ідентифікатор базового вичерпного ресурсу. який та чи інша подія у регіоні використала.

Таблиця формул зберігає у собі формулу та її опис, а також одиниці вимірювання і ідентифікатор експертів, що можуть її використовувати.

Таблиця ГДК визначає гранично допустимий коефіцієнт.

Залежності на мапі визначають зв’язок об’єкта на мапі, що був встановлений експертом та його табличними значеннями.

Медична статистика містить у собі посилання на формулу підрахунку, назву хвороби(поки що враховуються тільки злоякісні утворення та туберкульоз), посилання на регіон та медичний код.

Очікуваний результат містить у собі дані про тип викиду в атмосферу, ідентифікатор середи, значення та дату запису.

Значення параметру описує параметр, ідентифікатор розрахунку, самого параметра та експерта, що провів розрахунок, а також посилання на формулу.

POI представляють собою точки збору інформації про навколишнє середовище та містять інформацію про точне місцерозташування датчиків, опис (зазвичай з адресою) та тип власника (приватний або державний).

Точки полігону визначають координати вершин полігону, що був нанесений експертом на мапі.

Регіон визначає назву міста або області, ідентифікатор та ідентифікатор відповідного полігону.

Таблиця ресурсів визначає назву ресурсу, його тип (опис), одиницю вимірювання та ціну.

Із зазначених вище вхідних даних для статистичного аналізу використовувалися: міста, а саме назви та ідентифікатори регіонів; викиди в атмосферу по регіонам; кількість хворих на усі хвороби по регіонам; якість питної води по регіонам.

1.4 Задача статистичного аналізу різнорідних масивів даних у системі санітарно-гігієнічного моніторингу

Головною метою статистичного аналізу різнорідних масивів даних у медико-санітарній частині системи КЕЕЕМ є подання обробленої інформації у вичерпній формі – у графіках або таблицях для зменшення похибки висновків експерта системи. Також другорядною метою є об’єднання вхідних даних і подання їх у зручному для користувача вигляді.

До головної та другорядних задач, які має вирішувати ПЗ відносяться:

— візуалізація вхідних даних;

— встановлення оптимальних формул для визначення функцій залежності або кореляції споріднених даних;

— визначення форми візуалізації створених даних;

— визначення дизайну остаточних графіків діаграм;

— інтеграція програмного продукту у систему КЕЕЕМ.

Загалом, задача полягає у створенні програмного забезпечення, що допомагає аналізувати вхідну інформацію медичного характеру в залежності від різних чинників на базі вже існуючого застосунку та візуалізувати результати.

Висновки до розділу.

Призначення програмного продукту полягає у покращенні існуючої системи моніторингу модулем статистичного аналізу, щоб користувач, що є експертом-медиком робив висновки на базі вхідних та оброблених даних. Вхідні дані представляють собою захворювання, викиди в атмосферу та стан інших чинників в областях.

2 АНАЛІЗ НАЯВНИХ ПРОГРАМНИХ ПРОДУКТІВ, МАТЕМАТИЧНИХ ПІДХОДІВ ТА ВИДІВ ВІЗУАЛІЗАЦІЇ ДЛЯ СТАТИСТИЧНОГО АНАЛІЗУ ДАНИХ

На сьогоднішній день проблема сталого розвитку суспільства стоїть особливо гостро на питанні здоров'я та оточуючого середовища [1]. Проведений аналіз показав, що в Україні бракує спеціалізованих систем для статистичного аналізу в сфері КЕЕЕМ.

Ідентично схожі рішення, зазвичай, виконуються за державним замовленням у багатьох країнах світу та інтегруються в готові потужні системи [1], тому варто розглядати готові шаблони систем [4] та системи, що побудовані на програмному забезпеченні з широким набором можливостей та багатим вибором інструментів [5]. До таких потужних систем для статистичного аналізу і не тільки можна віднести [6] IBM SPSS та SAS.

2.1 Наявні рішення схожих проблем методами SAS

SAS (раніше "Statistical Analysis System") — комплекс статистичного програмного забезпечення, розроблений SAS Institute для управління даними, передової аналітики, багатовимірного аналізу, бізнес-досліджень, розслідування кримінальних справ та прогнозованої аналітики [4].

2.1.1 Загальна інформація про можливості SAS

Технологічні рішення SAS допомагають задовольнити потреби організацій практично в кожній галузі, незалежно від розміру. Сфери для використання SAS включають:

— банківську – у систему входить ПЗ для виявлення шахрайства;

— торгову – із вбудованими інструментами планування попиту виробники та роздрібні торгівці можуть зберігати полиці з товаром наперед;

— урядову – завдяки аналітичним інструментам потреб громадян;

— медичну – інструментарій передбачає ефективну аналітику системи охорони здоров’я, що дозволяє покращити догляд та врятувати життя.

SAS забезпечує провідні рішення з управління даними та вирішення задач в усьому життєвому циклі аналітики. Як стверджують розробники [4] – їх технологія була роками в процесі розробки та доопрацювань та забезпечує рішення, що базуються на досвіді клієнтів. Базові функції SAS включають:

— доступ до даних, підготовку та оцінку даних та розбиття даних на підгрупи, якщо можливо;

— досконалі потокові кроки та управління інформацією;

— візуалізацію та звітність;

— розмовний штучний інтелект і чатботи;

— статистику;

— машинне навчання та глибинне навчання;

— оптимізацію;

— економетрику;

— розгортання та моніторинг моделі.

SAS є потужним інструментом для загальної аналітики та її візуалізації [4].

2.1.2 Вступ до статистичного аналізу стану здоров’я методами SAS

Використання та значення даних зі спостережень як натуралістичних реальних випробувань, реєстрів пацієнтів та аналізів охорони здоров'я стверджує, що аналіз баз даних зріс в останні роки [7]. Спостережувані дослідження виробляють реальну систему даних про те, як методи лікування або стратегії працюють на практиці, що є критично важливим для споживачів, зацікавлених у фактичній практиці інформації. Такі дані зараз широко використовуються дослідниками і керівниками медичних закладів для оцінки стратегій лікування і прийняття стратегічних рішень [8]. Однак, методи і стандарти статистичного аналізу гірше розвинуті для даних зі спостережень в порівнянні з випадковими (рандомізованими) клінічними випробуваннями. Аналіз

якості даних зі спостережень є більш складним через такі проблеми, як упередження відбору. Неякісні аналізи і обмежений досвід з такими даними призвели до відсутності оптимального використання і навіть недовіри до такої роботи. Кілька дослідницьких груп, визнаючи ці аналітичні та звітні питання, починають надавати загальне керівництво щодо підвищення якості таких аналізів. Однак досі залишається відсутньою практична детальна інструкція щодо впровадження такої методології [9].

Варто заглибитися у дослідження на основі спостережень проти експериментальних досліджень. Коли дослідник розробляє та проводить експериментальне дослідження, призначення впливу (або лікування) знаходиться під контролем дослідника та виконується випадковим чином. Випадкове призначення використовує ймовірність для формування груп лікування, які, як очікується, будуть порівнянними. Дослідження без цього аспекту називаються дослідженнями на основі спостережень або неекспериментальними дослідженнями [9]. У дослідженнях за участю людей експериментальні дослідження зазвичай називаються випадковими (рандомізованими) контрольованими дослідженнями (РКД).

2.1.3 Робота з даними у CSV-форматі

Щоб імпортувати дані, розділені комою, до програмного забезпечення SAS, можна використати одну з двох команд: ‘proc sql’ або ‘proc import’. Приклад наведено на рисунку 2.1.

```
proc import
        datafile=nof1
        out=WORK.ndata(
                rename=(VAR1=Patient VAR2=Treatment VAR3=Cycle VAR4=Pair VAR5=Y)
        )
        dbms=dlm
        replace;
    datarow=2;
    delimiter=",";
    getnames=no;
run;
```

Рисунок 2.1 – Імпорт даних до середовища SAS

Параметр «datafile» приймає або шлях до файлу, укладений у лапки, або посилання на файл. Параметр «dbms» визначає файл із роздільниками, тобто спостереження розмежовуються деякими символами у вказаному файлі. Оскільки вказано параметр «replace», набір даних «WORK.ndata» замінюється, якщо він уже

існує. Дані зчитуються з файлу, починаючи з рядка номер 2, як зазначено в операторі ‘datarow’. Оператор 'delimiter' визначає символ розділювача у файлі, який надається параметру 'datafile'. У цьому випадку файл розділяється комами. Назви змінних не отримуються з файлу, отже оператор getnames = no.

На ранніх етапах статистичного аналізу необхідно зібрати інформацію про набір даних. Належний статистичний аналіз може бути неможливим без збору інформації про змінні, присутні в наборі даних. На цьому етапі аналітик може поставити деякі важливі запитання [9]. Які змінні присутні в наборі даних і що вони представляють? Як вимірювалися спостереження і в яких одиницях? Які змінні неперервні, а які категоріальні? Які рівні були виміряні для конкретної категоріальної змінної? Скільки спостережень доступно для певного рівня категоріальної змінної? Кількість можливих питань настільки величезна, що їх тут важко перелічити. Мова SQL може використовуватися в програмному забезпеченні SAS для виконання запитів до заданого набору даних і відповідей на деякі з цих запитань. Окрім цього, програмне забезпечення SAS надає кілька окремих процедур, які можна застосовувати в різноманітних ситуаціях. У цьому розділі представлено невелику кількість прикладів. Ці приклади – лише невелика частина можливого. Видано чимало підручників, які можуть допомогти в інших ситуаціях [9].

Набір даних можна роздрукувати для візуалізації в програмному забезпеченні SAS за допомогою процедур «print» або «sql». Процедура «sql» потужніша за процедуру «print», тому в прикладах, наведених у цьому розділі, використовується процедура «sql» [9]. Наступна програма друкує деякий набір даних NORMAL.ndata (рисунок 2.2):

```
title "First six observations of NORMAL.ndata";
proc sql outobs=6;
        select *
        from NORMAL.ndata;
quit;
```

Рисунок 2.2 – Вивід набору даних NORMAL.ndata

Перший оператор визначає заголовок для коду та його виведення. Зазвичай програма SAS складається з кількох викликів процедур, які послідовно надсилаються до програмного забезпечення SAS для інтерпретації, що призводить до спільного викладення результатів у вихідні дані програмного забезпечення SAS. У мові програмування SQL символ зірочки використовується для представлення всіх змінних, присутніх у даному наборі даних, тому всі змінні з «NORMAL.ndata» вибираються для друку. Параметр «outobs» у «proc sql» визначає кількість надрукованих спостережень. Щоб заощадити місце в цьому документі, потрібно лише шість спостережень [9].

Код виводить таблицю, представлену нижче. Рядок символів, який використовується в операторі 'title', використовується як підпис у таблиці 2.1.

Таблиця 2.1 – Вихідний датасет у вигляді таблиці

Patient	Treatment	Cycle	Period	Y
1	A	1	1	2394
1	B	1	2	2686
1	A	2	3	2515
1	B	2	4	2675
1	A	3	5	2583
1	B	3	6	2802

Підмножину даного набору даних можна легко надрукувати за допомогою SQL. Потрібні змінні можна вибрати та надрукувати, вказавши назви змінних через кому в операторі «select» (рисунок 2.3).

```
title "Observations from patients 11 and 12 in NORMAL.ndata";
proc sql;
        select *
        from NORMAL.ndata
        where Patient ge 11;
quit;
```

Рисунок 2.3 – Вибір підмножини набору даних

Працювати з даними SQL методами SAS є складним заняттям і може значно забирати час при використанні у специфічних системах, що мають бути гнучкими і при цьому мати достатню швидкодію.

2.1.4 Практичний приклад застосування пакету аналізу SAS

Дані, показані на рисунку нижче (рисунок 2.4), складаються з 89 випадків виживання.

Група 1			Група 2		
17	185	542	1	383	778
42	193	567	63	383	786
44	195	577	105	388	797
48	197	580	125	394	955
60	208	795	182	408	968
72	234	855	216	460	977
74	235	1174*	250	489	1245
95	254	1214	262	523	1271
103	307	1232*	301	524	1420
108	315	1366	301	535	1460*
122	401	1455*	342	562	1516*
144	445	1585*	354	569	1551
167	464	1622*	356	675	1690*
170	484	1626*	358	676	1694
183	528	1736*	380	748	

Рисунок 2.4 – Набір даних щодо лікування хворих на рак шлунку

Є шість значень на рядок, крім останнього рядка, він має п’ять. Перші три значення належать пацієнтам першої групи лікування, а решта – пацієнтам другої групи.

Наступний крок (рисунок 2.5) створює відповідний набір даних SAS.

```
data cancer;
   infile 'n:\handbook2\datasets\time.dat' expandtabs missover;
   do i = 1 to 6;
      input temp $ @;
      censor=(index(temp,'*')>0);

      temp=substr(temp,1,4);
      days=input(temp,4.);
      group=i>3;
      if days>0 then output;
   end;
   drop temp i;
run;
```

Рисунок 2.5 – Набір команд для перетворення вхідних даних в перехідний формат

Оператор infile надає повну назву шляху до файлу, що містить дані ASCII. Значення розділені табуляцією, тому використовується параметр expandtabs.

Параметр missover запобігає переходу SAS до нового рядка, якщо оператор введення містить більше змінних, ніж значень даних, як у випадку з останнім рядком. У цьому випадку змінна, для якої немає відповідних даних, встановлюється як відсутня [10].

Читання та обробка даних відбувається в ітераційному циклі do. Оператор input зчитує одне значення в символьну змінну, temp. Символьна змінна використовується для обробки зірочок, які вказують на цензуровані значення, оскільки між числом і зірочкою немає пробілу. Символ @ у кінці утримує рядок для подальших даних, які зчитуються з нього.

Якщо temp містить зірочку, функція index визначає її позицію; якщо ні, результат нульовий. Відповідно встановлюється змінна censor. Функція substr приймає перші чотири символи temp, а функція input зчитує це в числову змінну days [10].

Якщо значення days більше нуля, у набір даних виводиться спостереження. Це призводить до виключення згенерованого відсутнього значення, оскільки останній рядок містить лише п’ять значень.

Нарешті, символьна змінна temp та змінна індексу циклу і видаляються з набору даних, оскільки вони більше не потрібні.

З таким складним кроком даних, як цей, було б розумно перевірити результуючий набір даних, наприклад, за допомогою proc print.

Proc lifetest можна використовувати для оцінки та порівняння функцій виживання двох груп пацієнтів, що показано на рисунку 2.6:

```
proc lifetest data=cancer plots=(s);
   time days*censor(1);
   strata group;
symbol1 l=1;
symbol2 l=3;
run;
```

Рисунок 2.6 – Приклад набору команд для створення графіків

Параметр plots=(s) у операторі proc вказує на побудову кривих виживання. Log survival (ls), log-log survival (lls), hazard (h) і PDF (p) — це інші функції, які можна побудувати, а також графік цензурованих значень за стратами (c). Можна вказати перелік графіків; наприклад, plots=(s,ls,lls).

У операторі time вказується змінна часу виживання, за якою йде зірочка та змінна цензури, а в дужках значення, що вказують на цензуроване спостереження. Цензурна змінна має бути числовою, з непропущеними значеннями як для цензурованих, так і для нецензурованих спостережень [10].

Два оператори symbol використовуються для визначення різних типів ліній для двох груп (рисунок 2.7).

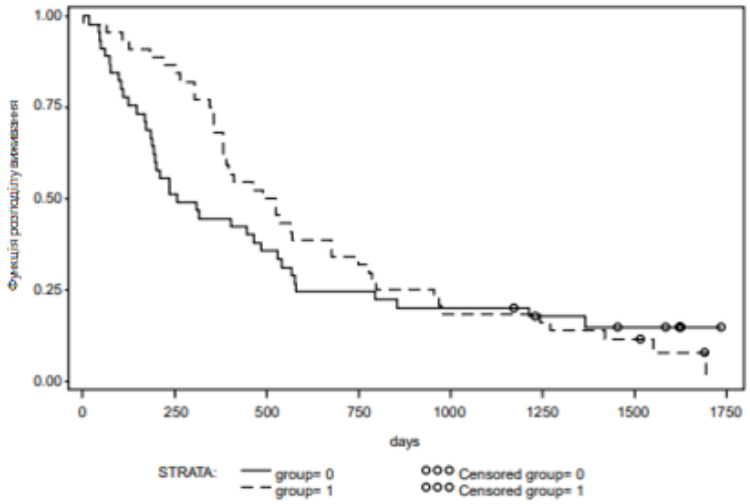

Рисунок 2.7 – Вихідний графік виконання написаної програми

ПЗ SAS є, безумовно, потужним універсальним застосунком для аналізу потреб користувача, проте не є достатньо гнучким для інтеграції у існуючу систему, не

передбачає кросплатформеність та є платним ПЗ. До того ж у даному програмному забезпеченні попередня обробка даних, навіть напряму з бази даних потребує великої кількості зусиль як розробника, так і користувача.

2.2 Наявні рішення схожих проблем методами IBM SPSS Statistics

SPSS Statistics – набір статистичного програмного забезпечення, розроблений IBM для управління даними, передової аналітики, багатовимірного аналізу, бізнес-передбачень і полегшення розслідувань. Long , що вироблена компанією SPSS Inc., була придбана IBM в 2009 році. Спочатку ПЗ називалося Statistical Package for the Social Sciences (SPSS), відображаючи оригінальний ринок, пізніше назва була замінена на Statistical Product and Service Solutions [11].

2.2.1 Загальна інформація про можливості SPSS Statistics

SPSS – це широко використовувана програма для статистичного аналізу в соціальних науках. Вона також використовується дослідниками ринку, дослідниками охорони здоров’я, дослідницькими компаніями, урядом, дослідниками освіти, маркетинговими організаціями, дата-майнерами та іншими.. Окрім статистичного аналізу, функціями базового програмного забезпечення є керування даними (вибір випадків, зміна форми файлів, створення похідних даних) і документування даних (словник метаданих зберігається у файлі даних) [11].

Багато функцій SPSS Statistics доступні через випадаючі меню або можуть бути запрограмовані за допомогою власної мови синтаксису команд 4GL. Програмування синтаксису команд має переваги відтворюваного виведення, спрощення повторюваних завдань і обробки складних маніпуляцій даними та аналізу. Крім того, деякі складні програми можуть бути запрограмовані лише в синтаксисі та не доступні через структуру меню. Інтерфейс випадаючого меню також генерує синтаксис команди: його можна відобразити у вихідних даних, хоча налаштування за замовчуванням потрібно змінити, щоб синтаксис був видимим для користувача. Їх

також можна вставити в синтаксичний файл за допомогою кнопки «вставити», наявної в кожному меню. Програми можна запускати в інтерактивному режимі або в автоматичному режимі, використовуючи додаткову програму Production Job Facility (рисунок 2.8).

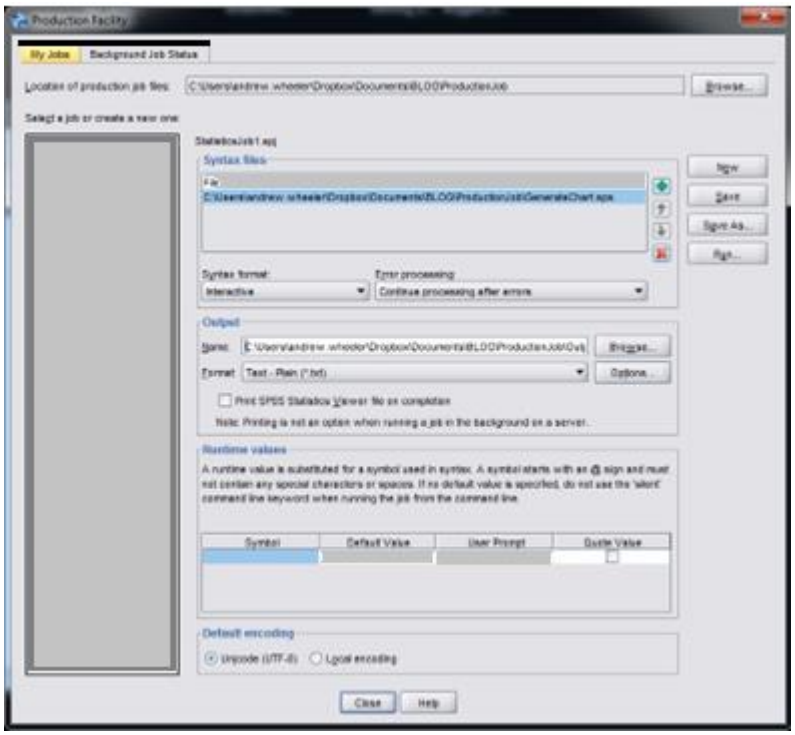

Рисунок 2.8 – Використання Production Job Facility

Крім того, мова «macro» може бути використана для написання підпрограм мови команд. Розширення програмованих можливостей Python може отримувати доступ до інформації в словнику даних і динамічно створювати програми синтаксису команд. Розширення можливостей програмування Python, представлене в SPSS 14, замінило менш функціональні «сценарії» SAX Basic для більшості цілей, хоча SaxBasic все ще є доступним і користується деякою популярністю. Крім того, розширення Python дозволяє SPSS запускати будь-яку статистику в пакеті безкоштовного програмного забезпечення R. Починаючи з версії 14 і далі, SPSS може керуватися зовнішньо за допомогою програми Python (рисунок 2.9) або VB.NET за допомогою наданих «плагінів» [12].

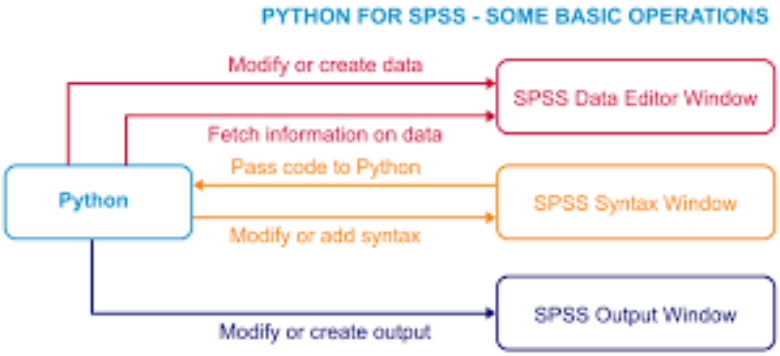

Рисунок 2.9 – Схема базового використання python3 у SPSS

SPSS Statistics накладає обмеження на внутрішню файлову структуру, типи даних, обробку даних і відповідні файли, що разом значно спрощує програмування. Набори даних SPSS мають двовимірну структуру таблиці, де рядки зазвичай представляють випадки (наприклад, окремих осіб або домогосподарства), а стовпці представляють вимірювання (наприклад, вік, стать або дохід домогосподарства). Уся обробка даних відбувається послідовно від випадку до випадку через файл (набір даних). Файли можна зіставляти один до одного та один до багатьох, але не багато до багатьох. На додаток до цієї структури та обробки за змінними, існує окремий сеанс матриці, де можна обробляти дані як матриці за допомогою операцій матриці (рисунок 2.10) та лінійної алгебри [13].

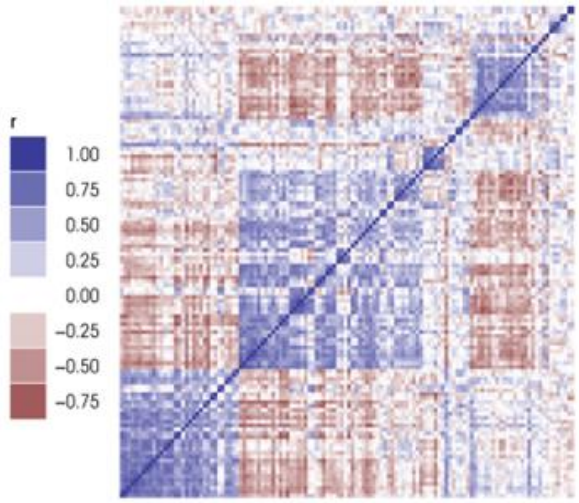

Рисунок 2.10 – Матриця кореляції засобами SPSS

Вже з опису та ілюстрацій до нього зрозуміло, що SPSS є потужним інструментом для збору, аналізу та візуалізації даних у великому розширені.

2.2.2 Робота з даними, отриманими мовою запитів SQL

Графічний інтерфейс користувача має два режими перегляду, які можна перемикати, натискаючи одну з двох вкладок у нижній лівій частині вікна статистики SPSS. «Перегляд даних» показує вигляд електронної таблиці випадків (рядки) і змінних (стовпці). На відміну від електронних таблиць, комірки даних можуть містити лише числа або текст, і в цих комірках не можна зберігати формули. «Перегляд змінних» відображає словник метаданих, де кожен рядок представляє змінну та відображає назву змінної, мітку змінної, мітку(-і) значення, ширину друку, тип вимірювання та багато інших характеристик. Комірки в обох поданнях можна редагувати вручну, визначаючи структуру файлу та дозволяючи вводити дані без використання синтаксису команд. Цього може бути достатньо для невеликих наборів даних. Великі набори даних, такі як статистичні опитування, частіше створюються в програмному забезпеченні для введення даних або вводяться під час особистого інтерв'ю за допомогою комп'ютера, шляхом сканування та використання програмного забезпечення оптичного розпізнавання символів і оптичних позначок, або шляхом прямого захоплення з онлайн-анкет. Потім ці набори даних зчитуються в SPSS [11].

SPSS Statistics може читати та записувати дані з текстових файлів ASCII (включаючи ієрархічні файли), інших пакетів статистики, електронних таблиць і баз даних. SPSS Statistics може читати та записувати в зовнішні таблиці реляційної бази даних через ODBC і SQL (рисунок 2.11).

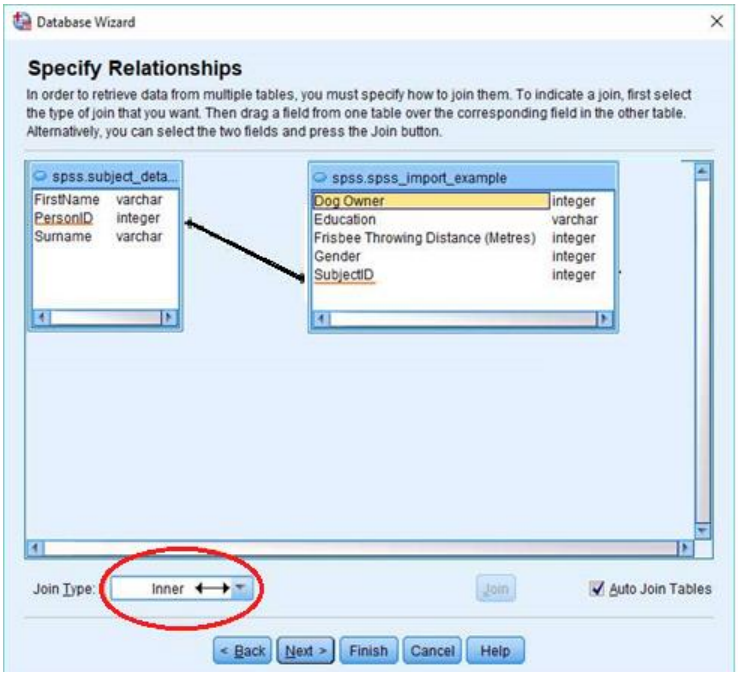

Рисунок 2.11 – Імпорт даних у SPSS з MySQL

Статистичні дані виводяться у власний формат файлу (файл *.spv із підтримкою зведених таблиць), для якого, окрім вбудованої програми перегляду, можна завантажити окрему програму для читання. Запатентований вихід можна експортувати в текстовий або Microsoft Word, PDF, Excel та інші формати. Крім того, вихідні дані можна отримати як дані (за допомогою команди OMS), як текст, текст із роздільниками, PDF, XLS, HTML, XML, набір даних SPSS або різноманітні формати графічних зображень (JPEG, PNG, BMP та EMF).

2.2.3 Практичний приклад застосування IBM SPSS Statistics

Забруднення повітря внаслідок швидкої індустріалізації та урбанізації стало четвертою за величиною загрозою глобальному здоров'ю людини після гіпертонії, харчових звичок і куріння. За оцінками Всесвітньої організації охорони здоров'я (ВООЗ), передчасна смертність через забруднене повітря сягає семи мільйонів, при цьому частка смертей через забруднене повітря на вулиці включає ішемічну хворобу серця – 40%, інсульт – 40%, хронічне обструктивне захворювання легень – 11 %, рак легенів — 6 %, гострі інфекції нижніх дихальних шляхів у дітей — 3 %. Крім того, екстремальні погодні явища та забруднення повітря впливають на інфекційні захворювання через воду, їжу, комах-переносників та гризунів. Основними

причинами забруднення повітря є тверді частинки (PM2,5, PM10), діоксид сірки (SO2) і оксиди азоту (NO2), які утворюють тверді частки (PM), безпосередньо забруднюючи повітря або перетворюючись на вторинні забруднювачі через хімічні реакції. в атмосфері. Зокрема, оскільки Міжнародне агентство з дослідження раку (IARC) класифікувало ПЧ як канцерогенну групу 1, інтерес до впливу ПЧ на здоров'я зростає. Повідомлялося, що тверді частки підвищують ризик респіраторних захворювань, таких як загострення астми та хронічне обструктивне захворювання легень, а також серцево-судинних захворювань, таких як нерегулярне серцебиття, судинна дисфункція та аритмія. Також повідомляється, що це пов'язано з гострою та хронічною передчасною смертю.

У цьому дослідженні застосовано техніку аналізу машинного навчання для прогнозування ризику твердих частинок. Типовими алгоритмами машинного навчання, які використовувалися в цьому дослідженні, були random forest, аналіз дерева рішень і багатошарова нейронна мережа. Крім того, для визначення зв'язків між незалежними змінними, які впливають на ризик твердих часток, було проведено асоціативний аналіз. Для аналізу асоціацій використовувався алгоритм апріорного принципу. Соціальні великі дані та інтелектуальний аналіз даних базуються на причинно-наслідковому зв'язку між емоційними ефектами PM. Результати дослідження могли б визначити фактори ризику та захисний фактор щодо проблеми ТЧ. Крива ROC (операційна характеристика приймача) і AUC (площа під кривою) використовувалися для оцінки моделей машинного навчання. IBM SPSS використовувався для аналізу дерева рішень, а R 3.4.2 використовувався для аналізу багаторівневої нейронної мережі випадкового лісу. , аналіз асоціацій та оцінка моделі.

Результати аналізу основних факторів, що впливають на емоції, пов'язані з твердими частинками, з використанням моделі випадкового лісу представлені на рисунку 2.12.

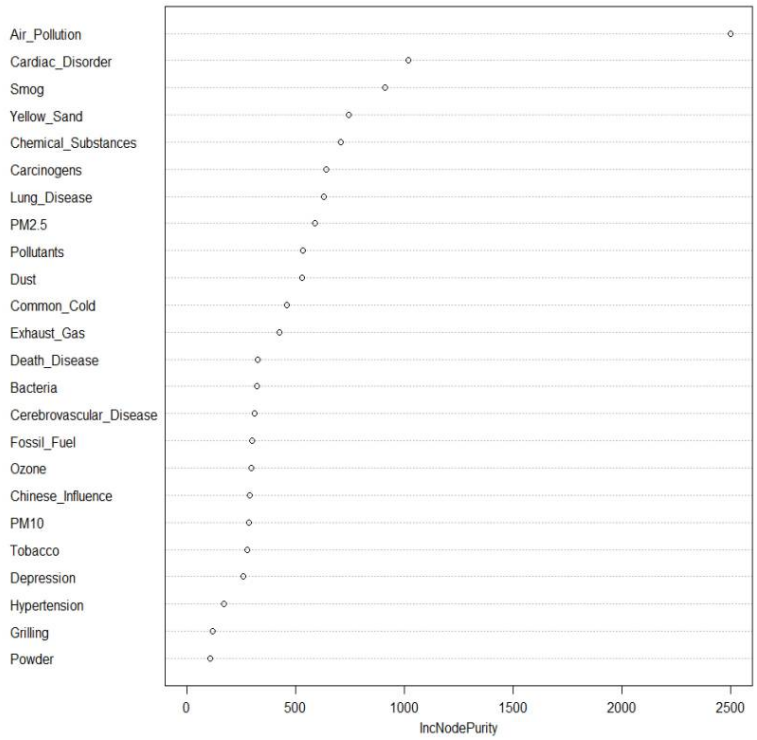

Рисунок 2.12 – Графік IncNodePurity проведений з використанням моделі випадкового лісу

Рисунок, який показує важливість (IncNodePurity) моделі випадкового лісу, вказує на те, що основний фактор, який має найбільший вплив на емоції, пов’язані з твердими частинками (важливий фактор, який класифікує нейтральні та негативні емоції), – це «забруднення повітря». За ним йдуть серцеві розлади, смог, жовтий пісок, хімічні речовини, канцерогени, захворювання легень, PM 2,5 і забруднюючі речовини [14].

Модель дерева рішень для прогнозування фактору ризику від твердих частинок показано на рисунку 2.13.

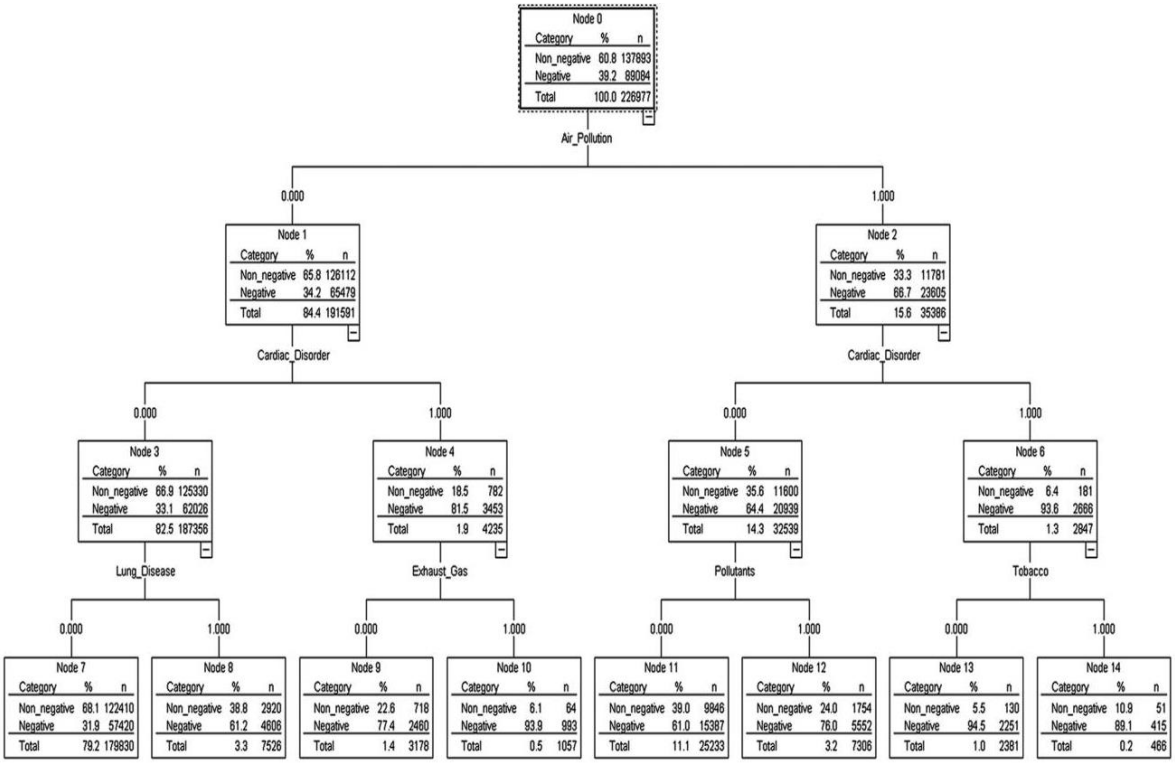

Рисунок 2.13 – Дерево рішень, побудоване методами IBM SPSS

Кореневе дерево у верхній частині структури дерева показує частоту залежної змінної без введених змінних-прогностиків (незалежних змінних). Співвідношення емоцій для твердих частинок кореневого вузла було 39,4% негативним і 60,6% невід'ємним. Оскільки причинно-захворювальний фактор у верхній частині під кореневим вузлом є фактором, який має найбільший вплив (дуже актуальний) на залежну змінну, вплив фактору «забруднення повітря» було виявлено найбільшим, тобто негативні емоції щодо твердих часток зросли з 38,2% до 69,7%, якщо онлайн-документ містив фактор забруднення повітря. Негативні емоції щодо твердих часток зросли з 65,7% до 92,6%, коли в документі були фактори забруднення повітря та серцевих захворювань [14].

IBM SPSS Statistics є потужним універсальним застосунком для аналізу даних, що зустрічаються у медико-санітарній частині з можливістю налаштування візуалізації у багатьох випадках із врахуванням усіх факторів і майже повністю задовольняє мету роботи, проте є дороговартісним програмним забезпеченням, потребує професійних навичок від розробника і користувача, а гнучкість є доволі умовною.

2.3 Аналіз математичних методів встановлення коефіцієнту кореляції

Вхідні дані до програми є кількісними, найкращим визначенням відношень у вигляді, що буде зрозумілим користувачу є набір коефіцієнтів кореляції, що формують матрицю кореляції Вибір серед необхідних формул простий – формула Пірсона або формула Спірмена, оскільки кожна з них є швидким рішенням поставленої задачі

2.3.1 Формула Пірсона

Кореляція Пірсона може оцінити лінійне відношення між двома неперервними змінними (відношення лінійне тільки тоді, коли зміна однієї змінної пов'язана з пропорційною зміною в іншій змінній).

$$r_{xy} = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \bar{x})^2 \sum_{i=1}^{m}(y_i - \bar{y})^2}} = \frac{cov(x,y)}{\sqrt{s_x^2 s_y^2}},$$

Приклад використання випадку: Ми можемо використати кореляцію Пірсона для оцінки того, чи збільшення свинцю у питній воді приводить до збільшення захворюваності на рак ЖКТ. Коли не можна встановити жодного лінійного відношення, коефіцієнт Пірсона дає значення нуль, що свідчить про відсутність кореляції – це одна з причин чому даний метод не є універсальним для всіх можливих випадків.

2.3.2 Формула Спірмена

Коефіцієнт за Спірменом оцінює, наскільки добре можна описати відношення між двома змінними за допомогою монотонної функції.

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$$

Важливо: кореляція Спірмена може оцінити монотонне відношення між двома змінними — неперервним або ординальним і буде ґрунтуватися на ранжуванні значень для кожної змінної, а не на необроблених даних.

Порівняння коефіцієнтів Пірсона і Спірмена:

1. Основна відмінність між двома коефіцієнтами кореляції полягає в тому, що коефіцієнт Пірсона працює з лінійним зв'язком між двома змінними, тоді як коефіцієнт Спірмена працює і з монотонними зв'язками.

2. Ще одна відмінність полягає в тому, що коефіцієнт Пірсона працює з raw-data змінними, тоді як коефіцієнт Спірмена працює з впорядкованими змінними.

3. Ніяких наслідків від переходу методу статистичного аналізу на коефіцієнт кореляції Спірмена немає навіть якщо б дані виявилися ідеально лінійними. Але, з коефіцієнтом Пірсона було би все навпаки – при наявності монотонного зв’язку він був би проігнорований.

Висновки до розділу

У даному розділі був описаний порівняльний процес готових рішень, формул кореляції та графіків візуалізації для статистичного аналізу.

Із готових систем та ПЗ, що містять у собі пакети для статистичного аналізу були проаналізовані SAS та IBM SPSS Statistics. Обидві системи є універсальними, пропонують комплексне рішення поставленої задачі, проте обидві є дороговартісними, недостатньо гнучкими, а SAS неможливо інтегрувати у готову систему КЕЕЕМ.

Для знаходження методу визначення коефіцієнтів кореляційної матриці був проведений порівняльний аналіз двох популярних формул для визначення коефіцієнтів кореляції – Пірсона та Спірмена. В результаті порівняння було виявлено, що формула Спірмена має перевагу над стандартною формулою Пірсона при знаходженні нелінійних зв’язків, а формула Пірсона має більшу швидкодію у порівнянні з формулою Спірмена, тому використовувати формули варто в залежності від вхідних даних.

При аналізі існуючих графіків для візуалізації статистичного аналізу було виділено матрицю кореляції та діаграму розсіювання. При більш детальному розгляді

обох методів візуалізації у обох графіків було виявлено недоліки, при роботі з діаграмою розсіювання велика кількість даних, що міститься у БД КЕЕЕМ сильно спотворює відображення дійсності, для матриці кореляції велика кількість даних не є проблемою, проте значення, що сильно вибиваються із тенденції можуть бути проігноровані.

2.4 Аналіз та оцінка методів візуалізації результатів, отриманих шляхом статистичного аналізу

Для того, щоб визначити яку математичну формулу та інструменти для створення програмного забезпечення обрати – варто визначити кінцевий результат, для цього необхідно обрати діаграму, що визначала би міцність зв'язків різнорідних даних.

2.4.1 Діаграма розсіювання

На рисунку 2.14 представлено діаграму розсіювання, яку можна використовувати з кількома семантичними групами, які можуть допомогти добре зрозуміти графік.

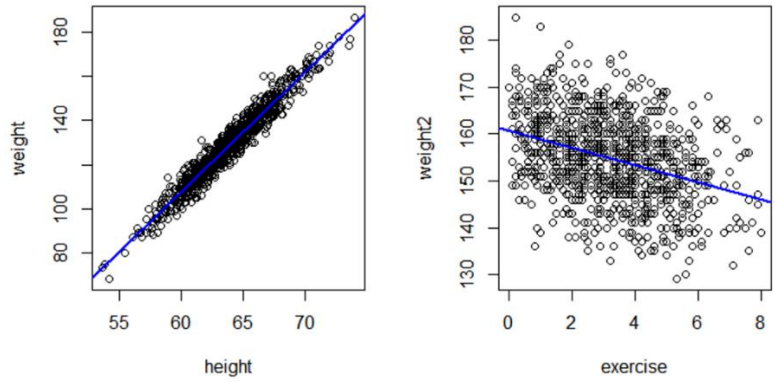

Рисунок 2.14 – Діаграма розсіювання

Вони можуть будувати двовимірну графіку, яку можна покращити шляхом відображення до трьох додаткових змінних із використанням семантики параметрів

відтінку, розміру та стилю. Усі параметри контролюють візуальну семантику, яка використовується для ідентифікації різних підмножин. Використання надлишкової семантики може допомогти зробити графіку більш доступною.

Переваги діаграм розсіювання:

— можуть відображати великі обсяги даних;

— найкращі для швидкого аналізу помірної кількості даних;

— можна легко побачити кореляцію між змінними та ефектами кластеризації.

Недоліки діаграм розсіювання:

— в деяких випадках дані лежать за межами графіку;

— точки діаграми можуть накладатися одна на одну частково і псувати дизайн;

— дискретність значень – трапляється, коли поле даних є категоричним;

— при великій кількості даних перегляд повної кількості даних унеможливлюється.

2.4.2 Матриця кореляції

Кореляційна матриця (рисунок 2.15) — таблиця, що показує коефіцієнти кореляції між змінними.

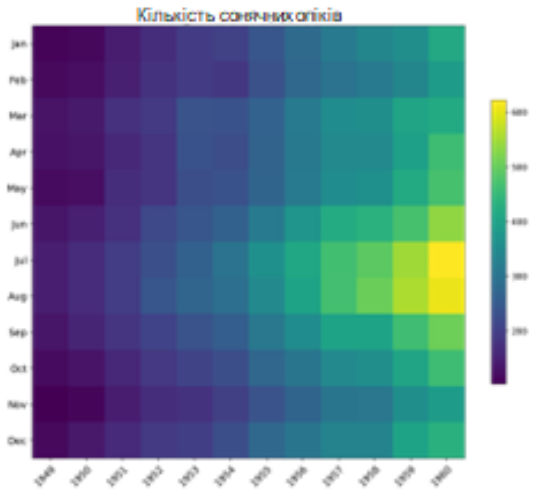

Рисунок 2.15 – Матриця кореляції

Кожна клітинка в таблиці показує кореляцію між двома змінними. Матриця кореляції використовується для узагальнення даних, як введення в більш досконалий аналіз, і як діагностична для розширених аналізів.

Переваги матриці кореляції:

— ні одна зміна не є маніпулятивною;

— можна обрати два методи збору даних;

— результати кореляційних досліджень є найбільш застосованими на практиці;

— починаючи з невеликих об’ємів даних можна побачити зв’язки кореляції;

— дослідники та експерти можуть самостійно встановити напрямок і силу кожного зв’язку;

— метод опитування гарно підходить під кореляційну матрицю;

— результати кореляційного дослідження легко класифікувати не зважаючи на обсяг вхідних даних.

Недоліки матриці кореляції:

— кореляційне дослідження є специфічним для зв’язків змінних;

— невідомо які змінні мають найбільший вплив;

— створення кореляційної матриці може бути трудомістким процесом;

— якість роботи аналітика може негативно вплинути на результати.

Серед методів знаходження взаємозв’язків матриця кореляції є найкращим інструментом візуалізації, хоча програє діаграмі розсіювання у виявлені втрачених від інших змінної.

3 ЗАСОБИ РОЗРОБКИ

Готове програмне забезпечення представляє із себе пакет скриптових застосунків, написаних мовою python3, є кросплатформеним та інтегрується до загальної системи KEEEM. Апаратне забезпечення, що використовувалося при розробці, наступне (рисунок 3.1):

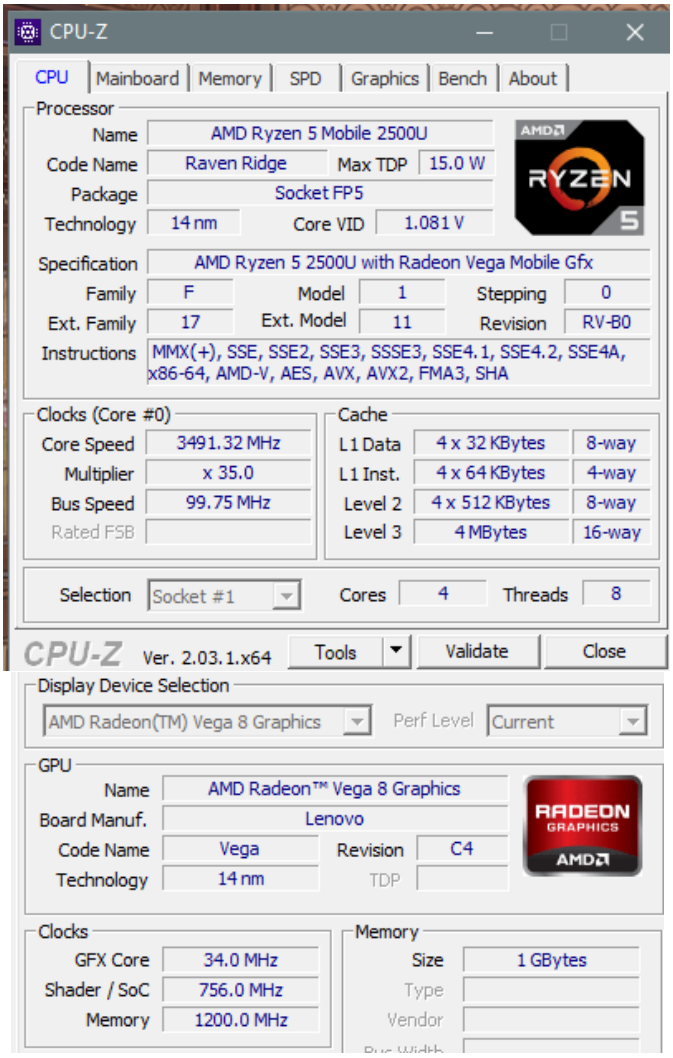

Рисунок 3.1 – Апаратне забезпечення, на якому здійснювалася розробка програмного забезпечення

Апаратне забезпечення, що використовувалося для розробки є доволі розповсюдженою конфігурацією, що дозволить розробникам краще орієнтуватися на засоби розробки та швидкодію системи.

3.1 Мова програмування Python та бібліотеки мови, що використовуються при розробці ПЗ

Для розробки програмного забезпечення використовувалася мова програмування python3. Насамперед, вона була використана через такі фактори як кросплатформеність, тобто її зручно інтегрувати в уже існуючу систему, легкість розробки складних обчислювальних програм для обробки великих масивів даних та швидкого аналізу. Для написання коду використовувався редактор PyCharm Community (рисунок 3.2) – його безкоштовна версія цілком задовольняє потреби поставлених задач.

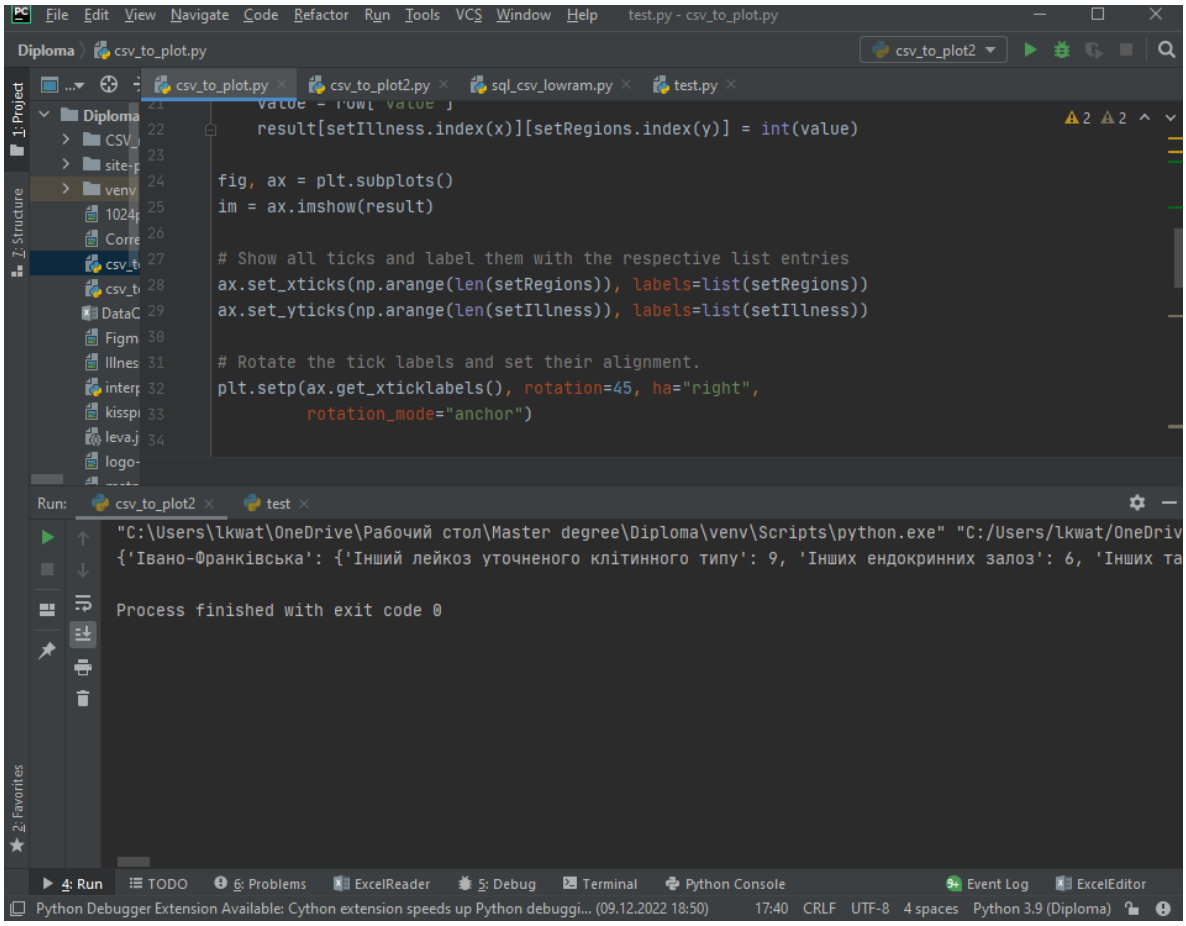

Рисунок 3.2 – Приклад використання редактора Pycharm у роботі

Для математичних прорахунків використовувалися, в основному, дві бібліотеки – Numpy та Pandas. Numpy (рисунок 3.3) написана на мові програмування C і має надзвичайну швидкодію для роботи з числами та важкими прорахунками.

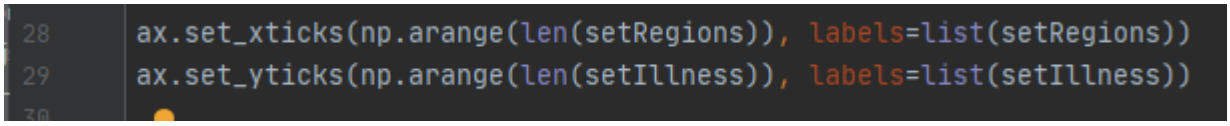

Рисунок 3.3 – Приклад використання бібліотеки NumPy при взаємодії з іншими бібліотеками

NumPy — бібліотека для мови програмування Python, що додає підтримку великих багатовимірних масивів і матриць, поряд з великою колекцією високорівневих математичних функцій для роботи на цих масивах. Предок PNumy, Numeric [15]. Так бачимо на рисунку 3.3, що функції даної бібліотеки використовуються для упорядкування даних у раніше створених numpy-масивах унікальних значень хвороб та регіонів.

Pandas (рисунок 3.4) є програмною бібліотекою, що написана на мові cython, частково спирається на NumPy і покликана, в першу чергу, маніпулювати великими чисельним даними, матрицями та у використанні для аналізу даних, в тому числі, статистичного [16].

Рисунок 3.4 – Приклад використання бібліотеки Pandas

Це вільне програмне забезпечення, випущене під трійковою ліцензією BSD. Назва походить від терміну «panel data», це термін економетрики для наборів даних, які включають спостереження протягом декількох періодів часу для одних і тих же об’єктів чи суб’єктів [16]. Головною причиною використання Pandas також стали DataFrame (рисунок 3.5), з яких легко завантажувати дані в бібліотеки візуалізації.

Рисунок 3.5 – Структура DataFrame

DataFrame використовуються для маніпуляцій табулярно структурованими даними. Pandas дозволяє імпортувати дані у DataFrame у форматах comma-separated values, JSON, Parquet, SQL database tables чи queries, та Microsoft Excel. Pandas дозволяє маніпулювати даними наступними операціями:

— merging – злиття (рисунок 3.6) об’єктів між собою [17];

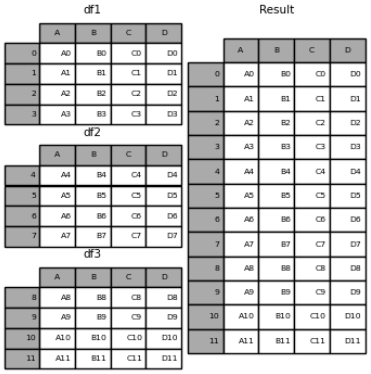

Рисунок 3.6 – Приклад злиття трьох DataFrame

— reshaping – змінення розміру (рисунок 3.7) чи транспонування об’єктів [17];

Рисунок 3.7 – Приклад зміни розміру через транспонування

— selecting – вибір значень з DataFrame;

— data cleaning – фільтрація значень, що повинні залишатися, бути обрані для подальшої роботи чи очищені [17];

— data wrangling (рисунок 3.8) – процес збору та перетворення необроблених даних в інший формат для кращого розуміння, прийняття рішень, доступу та аналізу за якнайменший час [18].

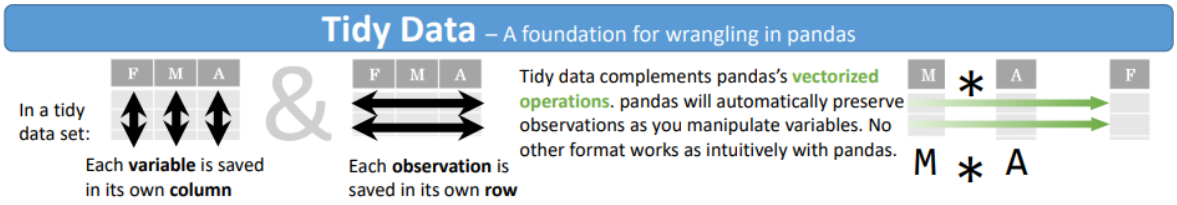

Рисунок 3.8 – Tidy Data, вона ж матриця моделі, що розроблена, щоб зберігати дані у охайному вигляді [19].

Розробка Pandas інтегрувала в Python багато порівнюваних особливостей роботи з DataFrames, які були вперше застосовані у мові програмування R [20].

Можливості DataFrame:

— підтримуються різні типи стовпців;

— розмір – динамічний;

— промарковані рядки і стовпчики;

— можливість виконання арифметичних операцій (рисунок 3.9) по рядкам і стовпчикам [16].

Operator	Pandas Method(s)
+	add()
-	sub(), subtract()
*	mul(), multiply()
/	truediv(), div(), divide()
//	floordiv()
%	mod()
**	pow()

Рисунок 3.9 – Приклад базових арифметичних операторів

Також важливим чинником для використання саме DataFrame стала його можливість оперувати з досконалою версією графічної бібліотеки matplotlib для відображення даних – Seaborn (рисунок 3.10).

```
plt.setp(ax.get_xticklabels(), rotation=45, ha="right",
         rotation_mode="anchor")


# Loop over data dimensions and create text annotations.
for i in range(len(setIllness)):
    for j in range(len(setRegions)):
        try:
            text = ax.text(j, i, result[i][j],
                           ha="center", va="center", color="w")
        except IndexError:
            break

ax.set_title("Dedicated illnesses for regions")
fig.tight_layout()
fig.canvas.manager.set_window_title('Illness x Region')
plt.show()
```

Рисунок 3.10 – Приклад використання MatplotLib з налаштуваннями Seaborn

Matplotlib — це бібліотека для візуалізації та графіків на мові програмування Python.

Вона надає об'єктно-орієнтований API для інтегрування сніппетів в застосунки з використанням загальних інструментів GUI, таких як Tkinter, wxPython, Qt або GTK. Існує також процедурний інтерфейс «pylab», заснований на вкладеній мові (як OpenGL), розроблений, щоб бути схожим на MATLAB. А вбудована SciPy використовує Matplotlib за замовчуванням. Matplotlib має всеосяжний і потужний API; будь-який атрибут фігури можна змінити як заманеться. Поєднання високого рівня інтерфейсу seaborn та глибокої налаштованості matplotlib дозволяє якнайшвидше досліджувати дані, так і створювати графіки, що можуть бути адаптовані до кінцевого продукту будь якої якості та рангу публікації [21].

Seaborn — бібліотека, в якій можна робити статистичні графіки на мові Python. Вона спирається на matplotlib і щільно вписується в структури даних pandas. Вона забезпечує високорівневий інтерфейс для малювання привабливої та інформативної статистичної графіки і є найкращою додатковою бібліотекою для візуалізації. Інтеграція Seaborn з matplotlib дозволяє використовувати його в багатьох середовищах, які підтримує matplotlib, включаючи пошуковий аналіз в веб-редакторах, взаємодію в режимі реального часу в GUI застосунках, і архівований вигляд вихідних даних в ряді растрових і векторних форматів. Для найкращого досвіду використання seaborn необхідно мати впевнені знання і щодо концепцій та основних положень бібліотеки matplotlib [22]. Тому варто використовувати бібліотеку seaborn разом із налаштуваннями matplotlib на більш глибокому шарі.

Seaborn передбачає наступний функціонал:

1. API, орієнтоване на датасети, щоб визначити зв’язок між змінними - немає універсального найкращого способу візуалізації даних. На різні питання найкраще відповідають різні візуальні інтерпритації [23].
2. автоматичне оцінювання та побудову графіків лінійної регресії;
3. підтримка високорівневих абстракцій для багатовимірних сіток графіків;
4. візуалізація одновимірного і біваріативного розподілу – при побудові графіків можна задавати два атрибути (рисунок 3.11). Двовимірна гістограма зберігає

результати біваріації у прямокутниках, і при відображенні можна бачити кількість потраплянь у область прямокутника з відповідним кольором [23].

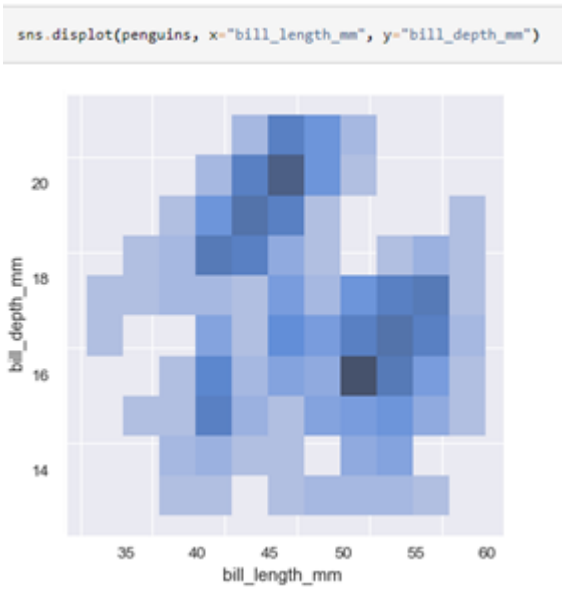

Рисунок 3.11 – Приклад біваріативного графіку – heatmap

Менш нав'язливим способом показу маргінальних розподілів з виокремленням кожного спостереження можна вважати діаграми розсіювання та KDE-діаграми (рисунок 3.12).

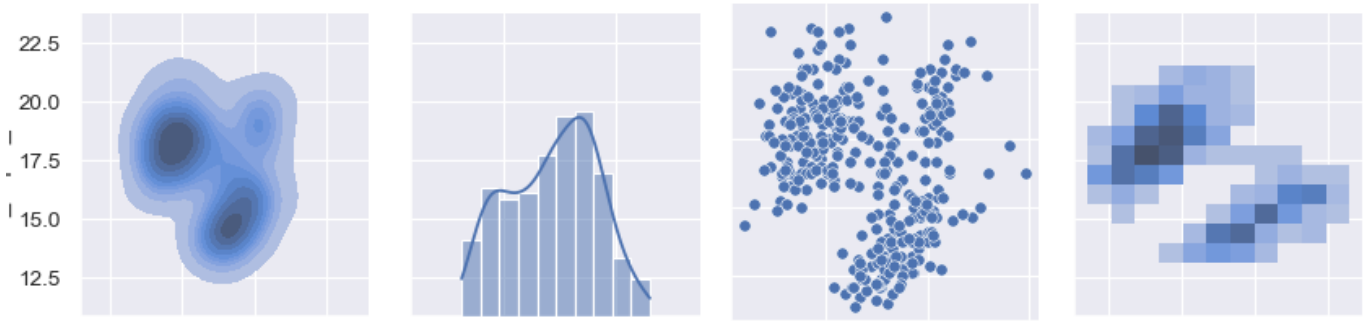

Рисунок 3.12 – Приклади біваріативних графіків з різними межами

Отже, мова програмування python3 та її бібліотеки NumPy, Pandas, MatplotLib та Seaborn є гарним вибором для створення гнучкого специфічного ПЗ для статистичного аналізу у системі КЕЕЕМ з гарною візуалізацією результатів.

3.2 Формат вхідних даних

Для того, щоб підготувати дані для роботи спочатку було визначено на яких саме даних можна будувати аналіз, а потім перетворено до обраного формату. Comma separated value (рисунок 3.13) – це, як видно з назви, дані, розділені комами.

	1	2	3	4	5
1	x,y,value				
2	Злоякісні новоутворення-всього,Київська,11588				
3	Губи,Київська,27				
4	Ротової порожнини,Київська,170				
5	Глотки,Київська,231				
6	Органів травлення,Київська,2678				
7	Стравоходу,Київська,160				
8	Шлунку,Київська,652				
9	Тонкого кишечника,Київська,31				
10	Ободової кишки,Київська,789				

Рисунок 3.13 – Перегляд csv файлу на практиці

Вони організовані таким чином, щоб дані можна було швидко імпортувати у таблиці у різних інструментах [24]. Причиною вибору використання цього формату є його здатність зберігати складні дані простим і читабельним способом. Крім того, файли CSV пропонують більше безпеки порівняно з форматами файлів схожих на JSON. А у обраному раніше Python легко читати ці типи файлів та оперувати ними за допомогою вище згаданої бібліотеки Pandas.

3.3 Система керування базами даних

Можливості СКБД MySQL забезпечують підтримку деяких типів даних, серед них цілочисельні типи(як зі знаком так і без), числа з плаваючою крапкою, строкові дані змінної та фіксованої довжини та інші [25].

MySQL дозволяє оперувати SQL-запитами SELECT, INSERT, UPDATE, REPLACE, DELETE, підтримує оператори і функції в SELECT & WHERE частинах запитів, працює з ORDER BY і GROUP BY, також забезпечує підтримку функцій

AVG(), SUM(), COUNT(), STD(), MAX(), MIN(), допускається використання JOIN в запитах[1]. Також MySQL дозволяє працювати з зовнішними ключами, підримує реплікації, транзакції, а також має багато інших функціонльних можливостей. Гнучкість використання MySQL полягає у широкому виборі таблиць, доступних для адміністратора – від типу MyISAM, що підтримують текстовий пошук до типу таблиць InnoDB, що підтримують транзакціїї на позаписному рівні, інші таблиці, що є посередніми модифікаціями базових таблиць були розроблені спільнотою для ефективного керування у вузьких галузях, тому ми їх не розглядаємо [25].

Для роботи з SQL-даними було використано СКБД MySQL Workbench (рисунок 3.14), що є уніфікованим візуальним інструментом для архітекторів БД, розробників та адміністраторів БД [25].

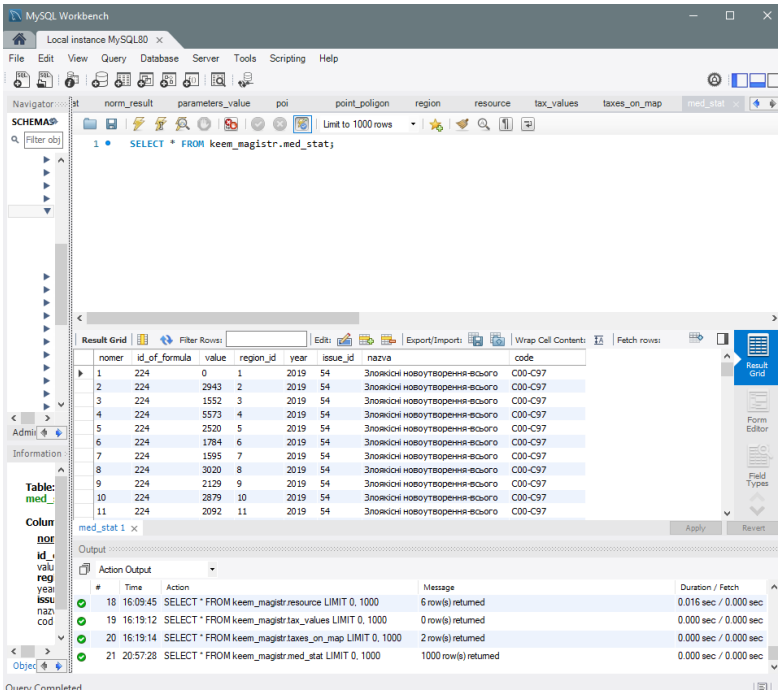

Рисунок 3.14 – Інтерфейс СКБД MySQL Workbench

Дана СКБД пропонує моделювання даних, розробку SQL-скриптів та інструменти для адміністрування готовою БД, у тому числі засоби для конфігурації сервера, користувацького адміністрування, створення бекапів та іншого [25].

MySQL Workbench провадить візуальні інструменти для створення, виконання та оптимізації SQL-запитів. Редактор допомагає у розробці, виділяючи синтаксис

рядків, автодоповнення, підказки із зразками коду, що був написаний користувачем раніше та історію виконаних запитів. Панель підключень бази даних дозволяє розробникам легко встановлювати стандартні підключення, у тому числі до MySQL Fabric [26]. Вбудований об’єктний браузер дозволяє отримувати миттєвий доступ до схеми БД та її об’єктів (рисунок 3.15).

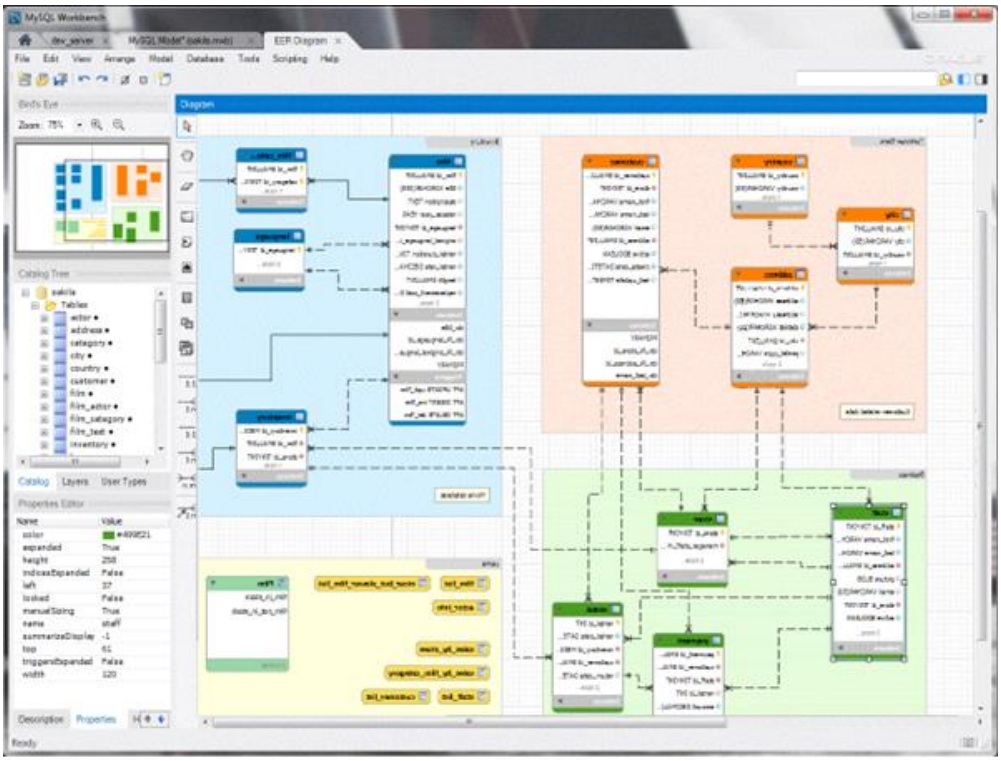

Рисунок 3.15 – Візуальне створення бази даних.

Також СКБД передбачає можливість міграцій баз даних та моніторинг швидкодії системи у цілому (рисунок 3.16) та окремих запитів. Інструменти оптимізації, в першу чергу корисні при адмініструванні системи, а репорти швидкодії провадять легку ідентифікацію та швидкий доступ до Input/Output даних, масивних SQL виразів та інше. Visual Explain Plan, що є одним із інструментів в один клік оптимізує вже написані запити та проводить тестування, щоб вдосконалити систему [26].

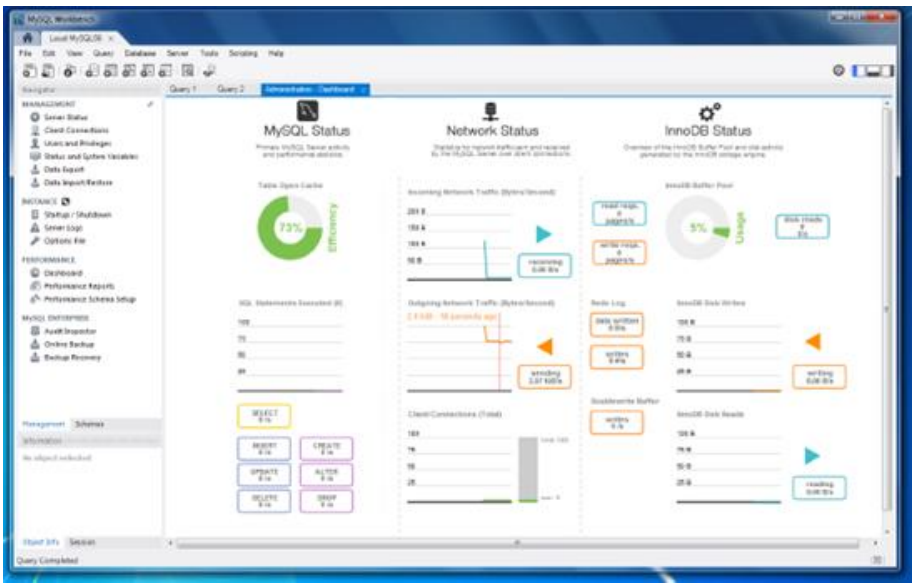

Рисунок 3.16 – Загальний вигляд трьох компонентів активної БД

Загалом, MySQL Workbench є найкращим рішенням для адміністрування баз даних, що написані на MySQL і дозволяють робити швидкі маніпуляції над запитами. MySQL була обрана за замовчуванням у вже готовій системі, хоча вона і поступається у багатьох факторах PostgreSQL [25], проте досі залишається актуальною для веб-розробників.

3.4 Інструменти для інтеграції ПЗ

Node.js — середовище для JavaScript з відкритим кодом, є крос-платформеним, та back-end для JavaScript, що працює на JavaScript Engine (т.зв V8 Engine) і виконує JavaScript код поза веб-браузером [27].

Він був розроблений для побудови масштабованих мережевих застосунків. Node.js дозволяє розробникам використовувати JavaScript для написання інструментів командного рядка і для серверних скриптів — запуск серверної сторони скриптів для створення динамічного вмісту веб-сторінки перед відправкою сторінки в веб-браузер користувача. Отже, Node.js підтверджує собою парадигму «JavaScript всюди», що об'єднує розробку веб-застосунків навколо єдиної мови програмування, а не різних мов для серверних і клієнтських скриптів [27].

Щоб запустити скрипт, написаний мовою python3 у середовищі node.js треба було спочатку вивантажити скрипт на сервер, а потім викликати його через Child Process методом spawn(). Node.js тоді збереже результат виконання скрипту, який можна буде використати для відображення даних. Також, щоб викликати скрипт діями зі сторони клієнта, необхідно додати POST [28] метод (рисунок 3.17).

```
app.post("/readPython", (request, response) => {
      // Reading Python files
      var dataToSend;
      // spawn new child process to call the python script
      const python = spawn('python3', ['public/script.py']);

     // collect data from script
     python.stdout.on('data', function (data) {
      dataToSend = data.toString();
     });

     python.stderr.on('data', data => {
      console.error(`stderr: ${data}`);
     });

     // in close event we are sure that stream from child process is closed
     python.on('exit', (code) => {
     console.log(`child process exited with code ${code}, ${dataToSend}`);
     response.sendFile(`${__dirname}/public/result.html`);
    });
});
```

Рисунок 3.17 – Приклад коду з використанням POST-методу

Даний метод інтеграції Python скрипту є доволі розповсюдженим рішенням, що дозволить легко знайти готовий приклад.

3.5 Редактори

Для того, щоб підготувати дані для подальшої обробки був обраний csv формат, а за його проміжне відображення (у процесі розробки, щоб можна було прослідкувати за помилками та недоліками) став відповідальним редактор Sublime Text (рисунок 3.18), також він використовувався для швидкого редагування .js/.json файлів.

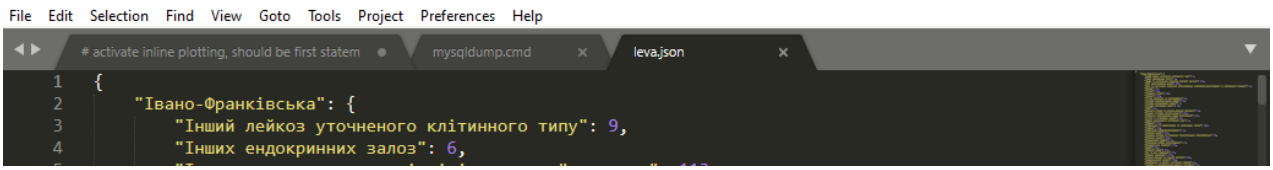

Рисунок 3.18 – Приклад використання Sublime Text

Даний редактор має крос-платформенну підтримку, мультивкладки, простий та гарний інтерфейс, виправлення помилок у мовах для веб-програмування,

багатовимірну синтаксичну підтримку та розвантажує центральний процесор, тому що його рендер відбувається на відеокарті [29]. Через ці та інші якості вважається найкращим редактором для швидких правок чи перегляду коду.

Для написання програмного забезпечення мовою python3 використовувався редактор, що був розроблений JetBrains спеціально під потреби даної мови програмування – PyCharm Community .

PyCharm вже є не простим редактором, а IDE для python3, він впроваджує аналіз коду, застосунок відлову багів, інтегровані тести, контроль версій, підсвітку синтаксису (рисунок 3.19) та підтримує веб розробку у Flask та web2py [30].

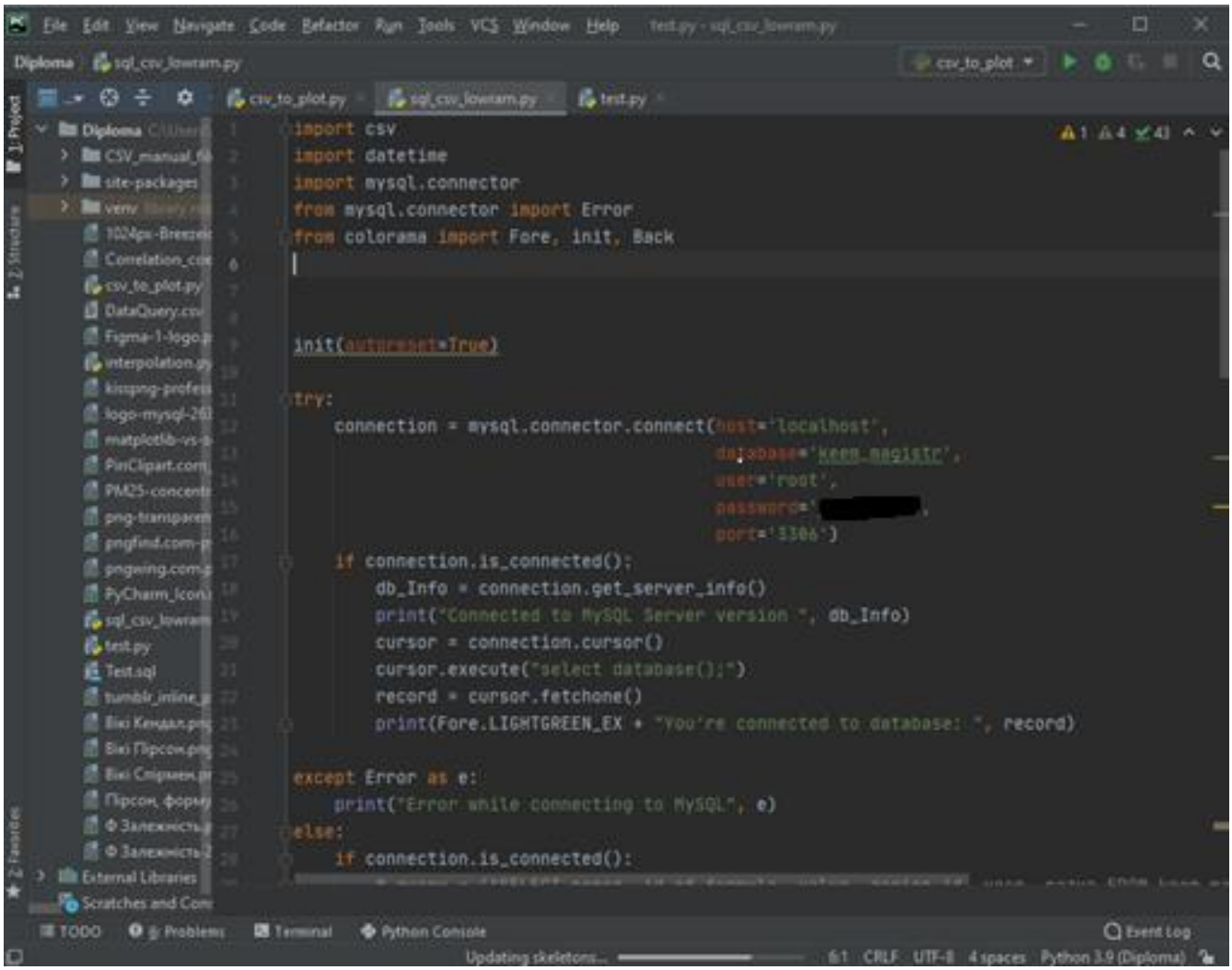

Рисунок 3.19 – Інтерфейс розробника у PyCharm Community

Головними причинами обрати дане середовище став попередній досвід роботи у ньому, зрозумілий інтерфейс, вбудований PowerShell та підсвітка синтаксису.

Для ініціалізації серверної частини та роботи з .js файлами був обраний гнучкий редактор Visual Code, що базується на фреймворці Electron.

VS Code є редактором із відкритим вихідним кодом, тому у магазині Microsoft можна побачити багато безкоштовних розширень, що створили «розробники для розробників», доволі часто це пришвидшує роботу у редакторі. Однією з цікавих функцій, що допомагає у розробці комплексних рішень є відсутність системи проекту – користувач може одночасно відкривати декілька директорій (рисунок 3.20) на своєму ПК, щоб дивитися як процес розробки однієї програмної частини впливає на іншу та на програмний продукт в цілому [31].

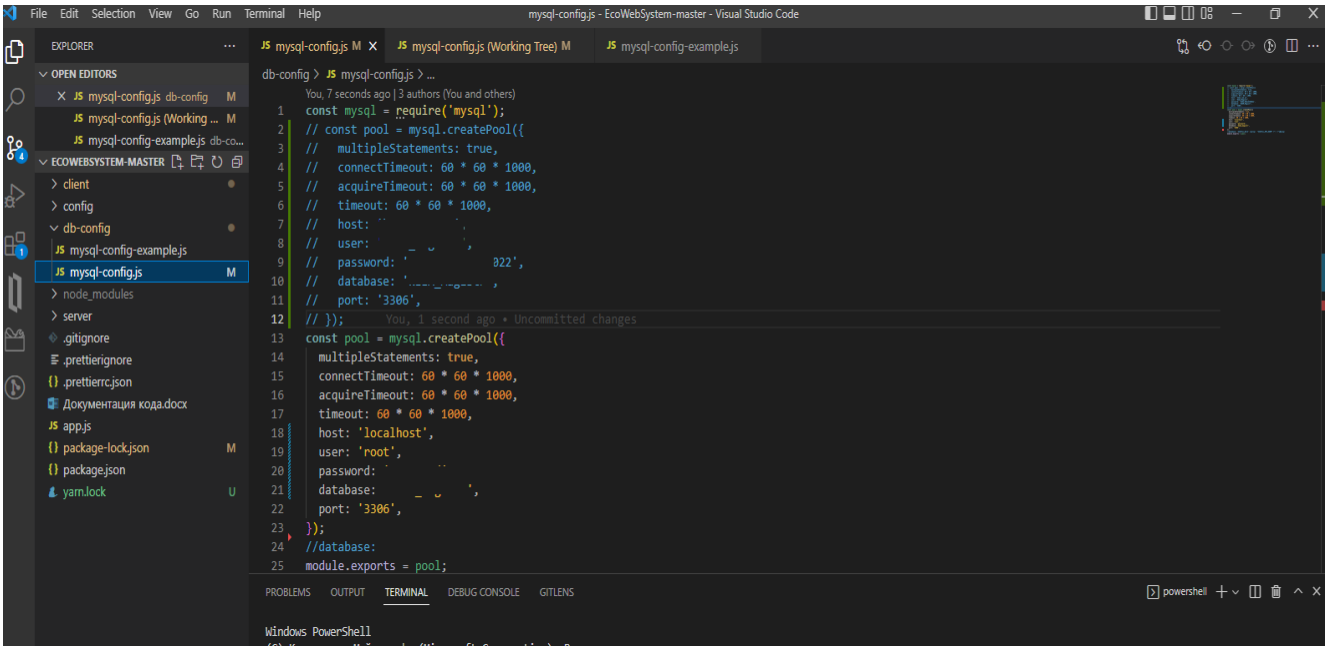

Рисунок 3.20 – Модуль запуску віддаленого чи локального серверу мовою js у VSCode

З недоліків даного редактора можна виокремити збирання даних користувача, та передача їх до корпорації Microsoft. Тобто, телеметричний код може бути доступний широкому загалу людей, а проект використаним у власних цілях [31].

Дана частина програмного забезпечення відноситься до системи КЕЕЕМ та використовувалася для того, щоб зрозуміти як виглядає створений раніше проект та з чим доведеться працювати.

Висновки до розділу

У даному розділі був описаний процес аналізу найкращих чи знайомих засобів для розробки серед доступних для вибору. Для збору, аналізу, розробки та візуалізації даних були обрані такі компоненти:

1. Серед мов та бібліотек – python3, JavaScript, MySQL, NumPy, Pandas, MatplotLib, Seaborn, SQLConnector – даний набір інструментів розробника передбачає гнучкість та легкість інтеграції у готову систему КЕЕЕМ.

2. Як СКБД – MySQL Workbench 8 – вона була обрана, оскільки база даних КЕЕЕМ знаходиться на MySQLServer.

3. Серед редакторів та IDE – PyCharm, Sublime Text, VS Code – за звичністю та функціоналом.

4. Як проміжний формат для подальшої обробки даних – csv – він гарантує легкість використання та швидкість обробки даних.

4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ТА РОБОТИ СИСТЕМИ

Модуль статистичного аналізу у системі КЕЕЕМ складається з двох частин (рисунок 4.1):

— частини, що перетворює вхідні дані SQL формату у дані формату csv;

— частини, що оброблює дані csv формату для аналізу у форматі pandas.dataframe та візуалізує дані.

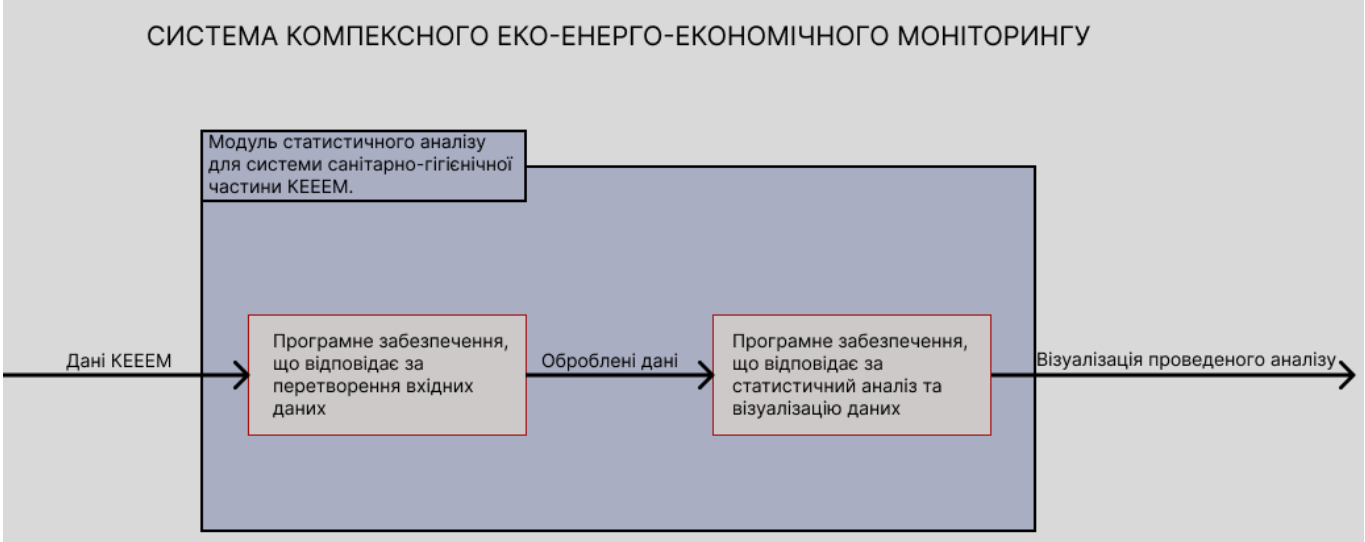

Рисунок 4.1 – Загальна схема роботи модулю статистичного аналізу у системі КЕЕЕМ

Для кращого розуміння роботи модулю необхідно розглянути кожну з частин програмного забезпечення.

4.1 Частина ПЗ, що забезпечує отримання та перетворення даних

Для початку роботи з даними, що належать КЕЕЕМ, необхідно їх імпортувати з бази даних. Для цього необхідно підключитися до бази даних і надати SQL-запит – він може бути специфікований, в залежності від тих даних, що необхідно отримати.

Для цього необхідно встановити бібліотеку python mysql, було використано функції підключення до БД та читання з неї. Для підключення використовуємо вбудовану функцію mysql.connector connection, при цьому вказуємо необхідні дані, щоб підключитися. Для зручності при успішному з'єднанні у консоль виводиться повідомлення, як і при помилці – це було зроблено, щоб наступним розробникам було легше працювати з цією частиною програми. Підключення (рисунок 4.2) відбувається у два етапи – спочатку до серверу MySQL, потім до самої бази даних. При цьому спроба підключення до бази даних відбувається тільки при успішному з'єднанні з MySQL сервером.

```
init(autoreset=True)

try:
    connection = mysql.connector.connect(host='localhost',
                                         database='keem_magistr',
                                         user='root',
                                         password='password',
                                         port='3306')
    if connection.is_connected():
        db_Info = connection.get_server_info()
        print("Connected to MySQL Server version ", db_Info)
        cursor = connection.cursor()
        cursor.execute("select database();")
        record = cursor.fetchone()
        print(Fore.LIGHTGREEN_EX + "You're connected to database: ", record)
```

Рисунок 4.2 – Підключення до локального серверу і бази даних

При успішному підключені виконується запис даних до csv файлу. Для цього спочатку виконується запит, що називається query командою cursor.execute()– у ній будемо писати запит на витягування даних з БД. Наступною конструкцією з циклом for програма записує дані у циклі по рядкам (рисунок 4.3).

```
cursor.execute("use keem_magistr;")
query = ('''SELECT
                med_stat.nazva AS x,
                 region.name AS y,
                 sum(med_stat.value) AS value
                FROM
                med_stat
                INNER JOIN region ON med_stat.region_id=region.id
                WHERE
                year >= 2019 AND med_stat.region_id = 10
                GROUP BY region.name, med_stat.nazva; ''')


cursor.execute(query)
with open('DataQuery.csv', 'w') as f1:
    f1.write("x,y,value\n")
    writer = csv.writer(f1, delimiter='\t', lineterminator='\n', )
    # writer.writerow("value;region_name;nazva")
    for (nazva, region_name, value) in cursor:
        # Для демонстрації даних
        # print(f'Amount of sickness: {value} for {region_name} with {nazva} diagnosis.')
        # Write to csv
        row = [f"{''.join(str(nazva).split(','))},{region_name},{value}"]
        writer.writerow(row)
```

Рисунок 4.3 – Запис даних з БД до csv-файлу

Також для правильного запису даних у файл формату csv коми з назв хвороб прибираються методом split(), щоб не спотворювати формат файлу. У результаті можна побачити файл із отриманими значеннями, розділеними комою та першим рядком як заголовком (рисунок 4.4).

	1	2	3	4	5	6
1	x,y,value					
2	Злоякісні новоутворення-всього,Київська,11588					
3	Губи,Київська,27					
4	Ротової порожнини,Київська,170					
5	Глотки,Київська,231					
6	Органів травлення,Київська,2678					
7	Стравоходу,Київська,160					
8	Шлунку,Київська,652					
9	Тонкого кишечника,Київська,31					
10	Ободової кишки,Київська,789					
11	Прямої кишки ректосигмоїдного з'єднання анусу,Київська,551					
12	Печінки та внутріпечінков.жовчних протоків,Київська,85					
13	Жовчного міхура та позапеч.жовчних протоків,Київська,68					
14	Підшлункової залози,Київська,341					
15	Органів дихання та грудної клітини,Київська,1195					
16	Порожн.носасеред.вухапридатк.пазух,Київська,20					
17	Гортані,Київська,131					
18	Трахеїбронхівлегенів,Київська,1027					
19	Плеври,Київська,5					
20	Кісток та суглобових хрящів,Київська,38					
21	Сполучної та інших тканин,Київська,59					
22	Меланоми шкіри,Київська,259					
23	Інші новоутворення шкіри,Київська,1170					
24	Молочної залози,Київська,1396					
25	Жіночих статевих органів-всього,Київська,1516					
26	Вульви,Київська,45					
27	Шийки матки,Київська,435					
28	Тіла матки,Київська,662					
29	Яєчників,Київська,338					
30	Інших придатків матки,Київська,11					
31	Плаценти,Київська,1					
32	Чоловічих статевих органів,Київська,540					
33	Статевого органу,Київська,18					

Рисунок 4.4 – Структура csv-файлу

Без обробки вхідних даних, неможливо працювати над створенням ПЗ, що повинно оброблювати дані методом статистичного аналізу, тому важливим фактором

було створення даної частини ПЗ і її тестування, щоб виявити недоліки попередньої версії.

4.2 Частина ПЗ, що забезпечує статистичний аналіз та візуалізацію оброблених даних

Спираючись на дані з попередніх розділів, можна чітко визначити схему роботи з ПЗ для медика-експерта системи КЕЕЕМ. Користувач викликає візуалізацію вхідних даних або візуалізацію відносин на знаходження кореляції чи залежності різнорідних даних через користувацький інтерфейс. Запит надсилається до модулю статистичного аналізу, що у свій час за вхідними параметрами складає прості або складені запити до бази даних. База даних, у свою чергу, повертає дані для візуалізації існуючої статистики чи складений набір даних, що використовується для подальшої обробки та аналізу у модулі. Після обробки даних вони візуалізуються, файл повертається на користувацький інтерфейс (рисунок 4.5).

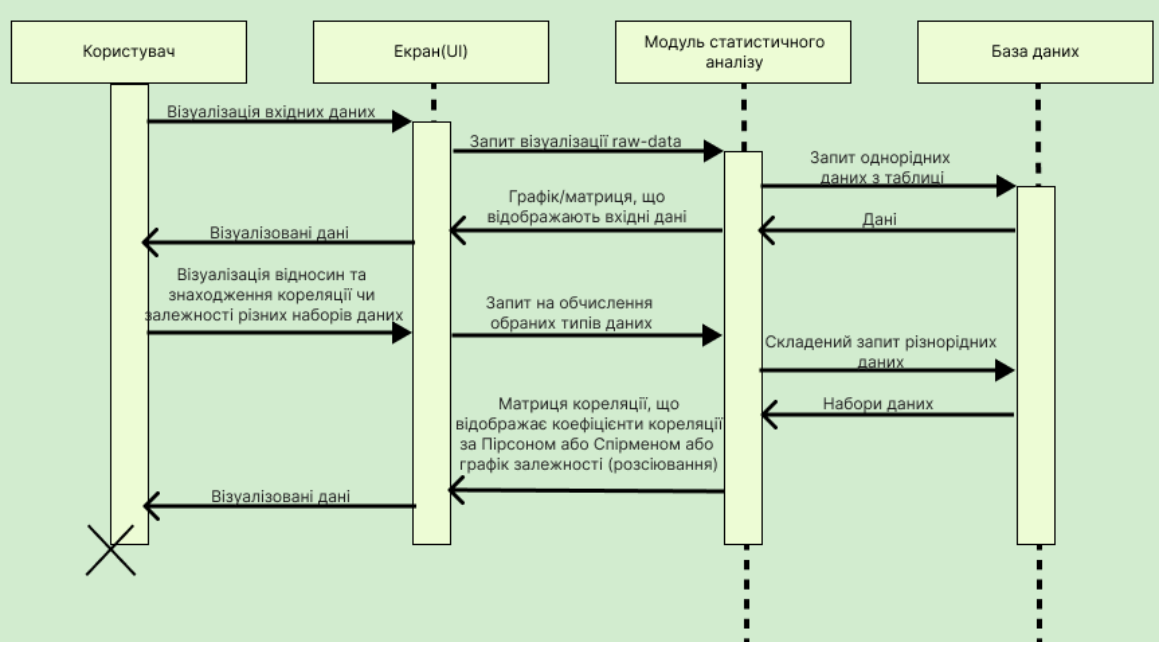

Рисунок 4.5 – Діаграма послідовності

Оброблені дані потрапляють в іншу частину ПЗ, що відповідає за встановлення причино-наслідкових зв’язків між досліджуваними ознаками, що формуються за парною кореляційною моделлю з прямолінійним зв’язком за принципом кількісних даних.

Імпорт даних відбувається методом read_csv бібліотеки pandas. Також вхідні дані до даної частини розбиваються на списки (рисунок 4.6), значення для роботи раніше були специфіковані користувачем за допомогою функцій інтерфейсу (рисунок 4.5).

```
import csv
from colorama import Fore, init, Back
import matplotlib.pyplot as plt
from pandas import *
import numpy as np
data = read_csv("DataQuery.csv", sep=',', encoding='cp1251')

illness = data['x'].tolist()
region_id = data['y'].tolist()
value = data['value'].tolist()
```

Рисунок 4.6 – Частина коду, що відповідає за імпорт оброблених даних

Далі, використовуючи формулу Спірмена, знаходимо коефіцієнти кореляції(рисунок 4.7).

```
def correlation(dataset, cor):
    df = dataset.copy()
    col_corr = set()  # set() дозволяє зберігати унікальні значення списків
    corr_matrix = dataset.corr()
    for i in range(len(corr_matrix.columns)):
        for j in range(i):
            if abs(corr_matrix.iloc[i, j]) > cor:  # абсолютні значення -1 до +1
                colname = corr_matrix.columns[i]
                col_corr.add(colname)
    df.drop(col_corr, axis=1, inplace=True)
    return df
```

Рисунок 4.7 – Функція створення кореляційної матриці

Оскільки формула вбудована у бібліотеку Pandas як стандартна формула для кореляції, викликаємо її через функцію corr(), доступ до атрибутів також видобуваємо

по рядкам та стовпчикам вбудованими методами бібліотеки pandas у циклі. Отримані дані записуємо у DataFrame.

Оскільки, дані є кількісними, а залежність кореляції лінійною, варто врахувати похибку, що може виникати при значеннях, наближених до цілочисельних. Для цього вводимо зміну Score з типом даних DataFrame для навчання, імпортуємо методи машинного навчання та оцінюємо похибку попередніх обчислень, функція повертає зміну Score у вигляді DataFrame. Різниця початкового результату та похибки є максимально точним результатом обчислень коефіцієнту кореляції.

```
def acc_score(df, label):
    Score = pd.DataFrame({"Classifier": classifiers})
    j = 0
    acc = []
    X_train, X_test, Y_train, Y_test = split(df, label)
    for i in models:...
    Score["Accuracy"] = acc
    Score.sort_values(by="Accuracy", ascending=False, inplace=True)
    Score.reset_index(drop=True, inplace=True)
    return Score
```

Рисунок 4.8 – Виправлення похибки статистичного аналізу

Для відображення матриці кореляції використовувалися бібліотеки matplotlib та seaborn, остання має можливість приймати та перетворювати типи DataFrame у таблиці візуалізації, однією з таких таблиць є матриця кореляції. Для налаштування кольорової гами [34] та виокремлення значень максимальної позитивної або негативної кореляції була створена функція highlight_max, що допомагає користувачеві зрозуміти тенденції зв’язку захворювань та чинників у регіоні.

Одним с прикладів парної кореляції є схожість впливу певного чиннику на хвороби у регіоні (рисунок 4.9).

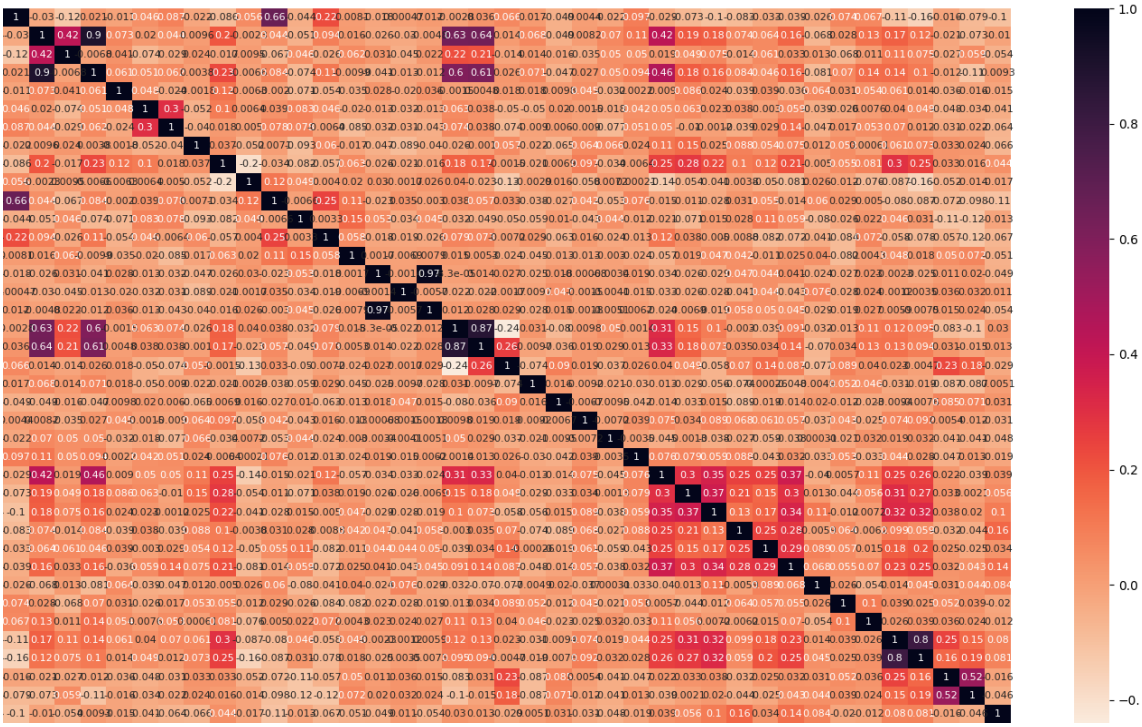

Рисунок 4.9 – Приклад парної кореляції хвороб у регіоні в залежності від викиду магнію та його сполук у атмосферу

Головна проблема матриць кореляції при великій кількості входжень даних з унікальними заголовками користувачу важко зрозуміти ситуацію по конкретним даним, а відображення заголовків зникає або стає незрозумілим, тому входження потрібно скорочувати (рисунок 4.10).

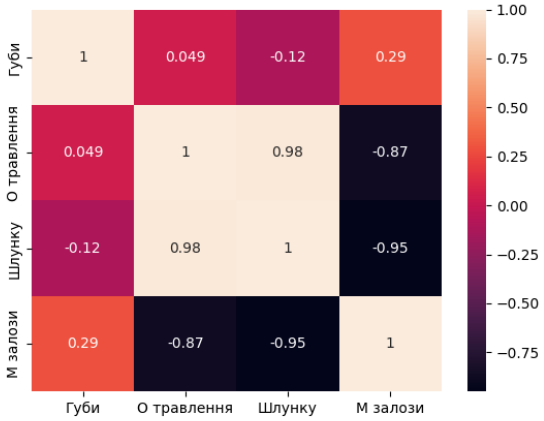

Рисунок 4.10 – Приклад невеликої матриці кореляції, вхідні дані, якої було обмежено користувачем

Реалізація ПЗ була створена одними з найгнучкіших інструментів для інтеграції у існуючу систему КЕЕЕМ і пропонує базовий функціонал статистичного аналізу. Функціонал приближеного перегляду матриць та графіків, що мають велику кількість

унікальних вхідних даних, зазвичай, передбачено у універсальних потужних застосунках, які були розглянуті у попередніх розділах.

4.3 Робота користувача з системою

Веб-додаток КЕЕЕМ складається з різних сторінок, що використовуються при різних видах роботи різними експертами. Для роботи з модулем статистичного аналізу необхідно авторизуватися як експерт-медик.

Для авторизації на сайті КЕЕЕМ необхідно натиснути кнопку у верхньому правому кутку головної сторінки (рисунок 4.11).

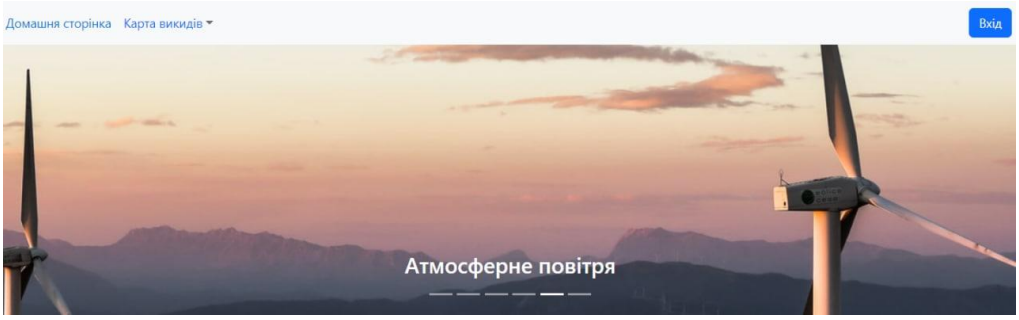

Рисунок 4.11 – Головна сторінка веб-додатку КЕЕЕМ

Щоб авторизуватися вводимо логін та пароль експерта у діалоговому вікні (рисунок 4.12).

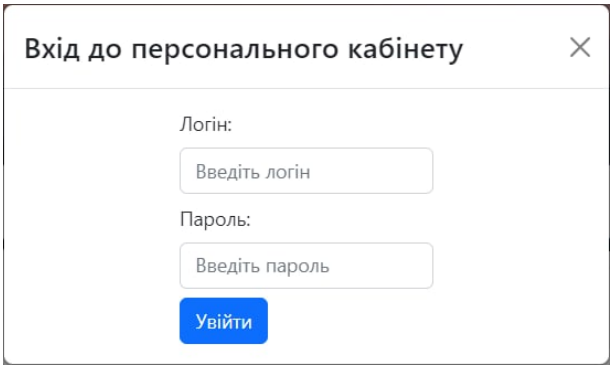

Рисунок 4.12 – Вікно входу до персонального кабінету

Після авторизації як експерт-медик, необхідно обрати карту викидів у випадаючому меню (рисунок 4.13)

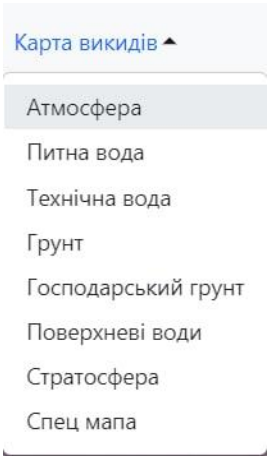

Рисунок 4.13 – Діалогове вікно карти викидів

Загальне відображення карти викидів виглядає наступним чином: (рисунок 4.14)

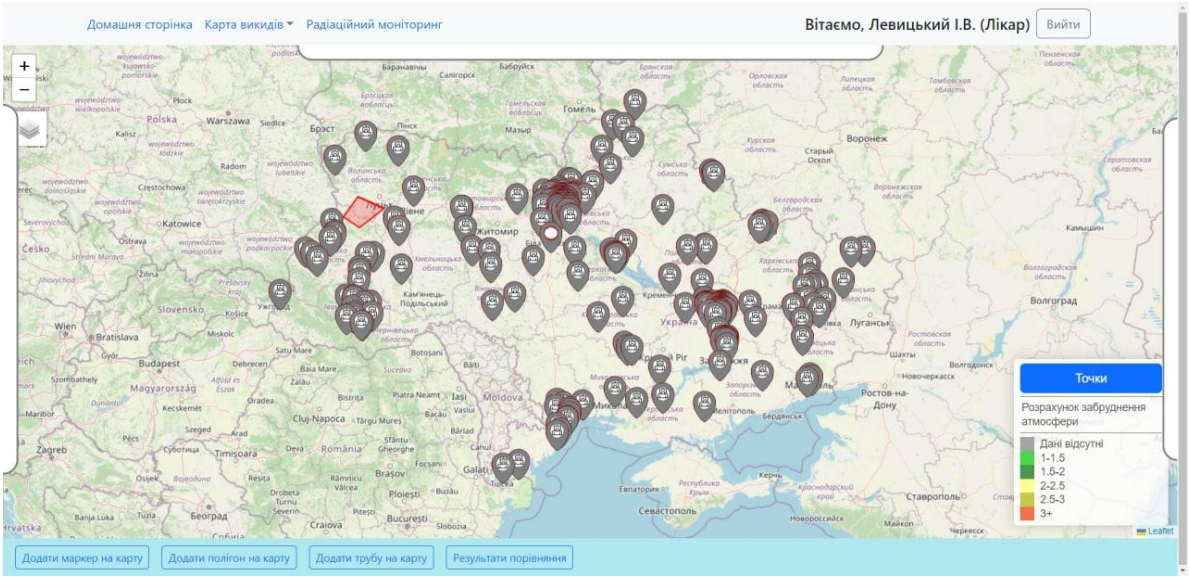

Рисунок 4.14 – Карта викидів

Для перегляду стану довкілля та здоров’я населення по областям, необхідно вибрати праве бокове меню у інтерфейсі карти викидів та у виборі полігонів (рисунок 4.10) встановити параметр «Області».

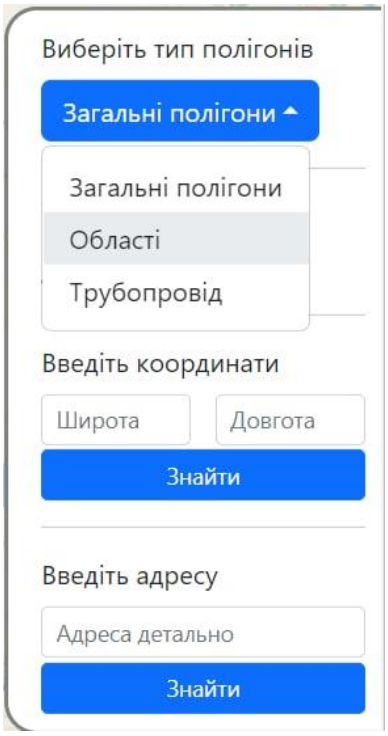

Рисунок 4.15 – Фільтрація полігонів

Далі медик-експерт натискає на область, з’являється pop-up меню, яке, в свою чергу, викликає діалогове вікно медичної статистики (рисунок 4.16).

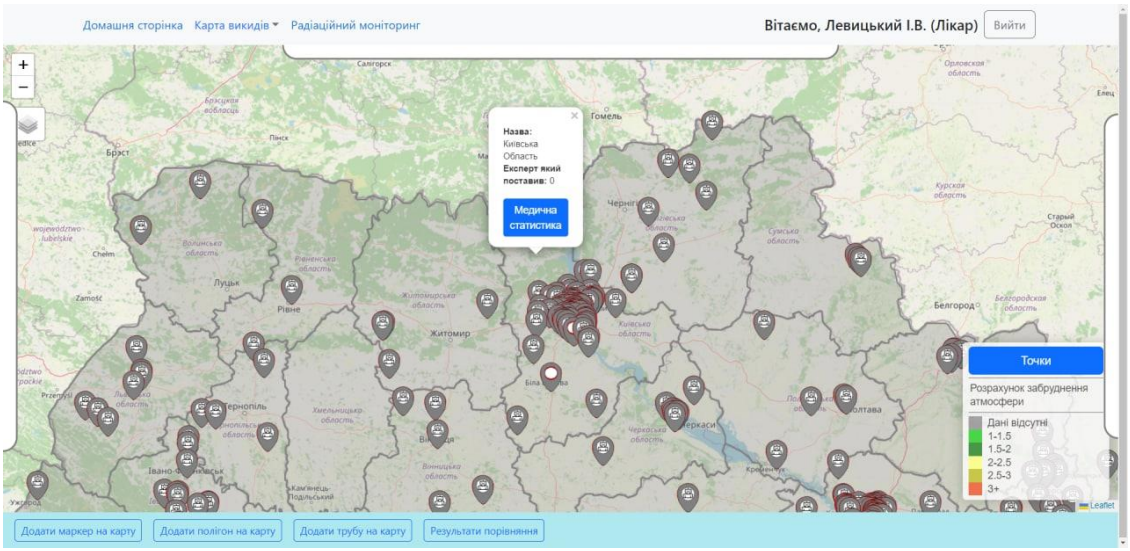

Рисунок 4.16 – Карта викидів по областям з елементом pop-up меню

Після того, як користувач натиснув на кнопку «Медична статистика», перед ним відкривається наступне діалогове вікно з можливістю вибору чинників впливу на здоров’я та захворюваннями, а також з можливістю обрати види графіків та що візуалізувати (рисунок 4.17).

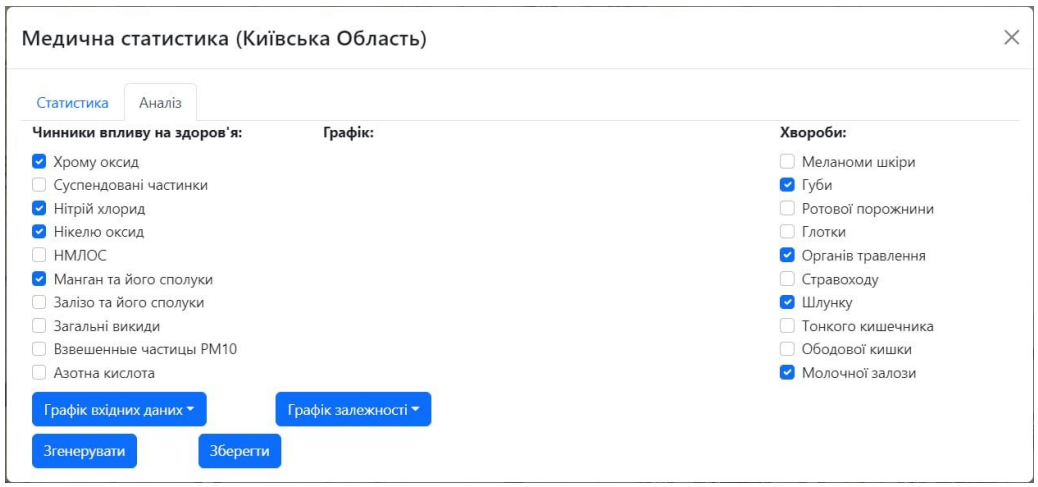

Рисунок 4.17 – Діалогове вікно медичної статистики для Київської області

Далі користувач обирає чинники впливу на здоров’я та хвороби у регіоні, щоб згенерувати графік залежності (рисунок 4.18).

Рисунок 4.18 – Результат виконання побудови матриці кореляції по чинникам.

Інтерфейс користувача є інтуїтивно зрозумілим та лаконічним для використання, проте, через лаконічність та простоту, користувач втрачає деякі можливості статистичного аналізу, що при необхідності можна додати програмно, оскільки вихідний код є відкритим.

Висновки до розділу.

У даному розділі було описано програмну реалізацію модуля статистичного аналізу для системи KEEEM, представлені схеми роботи ПЗ, пояснена робота користувача з системою, також були виокремлені недоліки та плани для майбутнього покращення ПЗ.

5 СТАРТАП-ПРОЄКТ

Для реалізації ПЗ важливо провести низку аналізів перед виходом на ринок, а також зробити порівняльну характеристику на тлі конкурентів.

5.1 Опис ідеї стартапу

У межах даного підпункту буде наведено аналіз та подано у вигляді таблиць наступне:

— зміст ідеї;

— можливі напрямки застосування;

— вигоди, які може отримати користувач товару;

— відмінні риси від наявних аналогів та напрямків.

Зміст ідеї, потенційні напрямки застосування та вигоди, що може отримати користувач ПЗ представлені у вигляді таблиці 5.1.

Таблиця 5.1. Опис ідеї стартапу.

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Програмний засіб аналізу медичних даних у готовій системі санітарно-гігієнічного моніторингу	Інтеграція модулю аналізу даних в існуючу систему моніторингу	Завдяки інтеграції програмного застосунку в існуючу веб-систему рішення є кросплатформенним
	Візуалізація вхідних даних, та даних, що були проаналізовані	Ілюстративне зображення даних та виявлення існуючих проблем у даних
	Аналіз залежностей споріднених чинників	Знаходження кореляції двох наборів даних між собою
		Знаходження аномальних чи нетипових даних у наборі значень
		Гнучкість використання системи – користувач обирає вибірку даних за різними параметрами

На ринку не існує окремих програмних продуктів, хоча є вже інтегровані модулі до моніторингу, пов'язані зі здоров'ям, також є велика кількість готових методів розв'язання на існуючому ПЗ IBM SPSS та SAS. Основною відмінністю даного програмного забезпечення, що воно не потребує для аналізу універсальних програмних продуктів як от SPSS та SAS, хоча розроблене ПЗ не є таким універсальним, проте є найкращим для існуючої системи моніторингу. У таблиці 5.2 наведено порівняльний аналіз визначення сильних, слабких та нейтральних характеристик ідеї ПЗ.

Таблиця 5.2. Визначення сильних, слабких та нейтральних характеристик проекту

№ п/п	Техніко-економічні характеристики ідеї	Порівняння проекту з популярними рішеннями			W	N	S
		Мій проект	Рішення на IBM SPSS	Рішення на базі ПЗ SAS			
1.	Автономність	Обов'язкове підключення до мережі інтернет	Після встановлення може працювати автономно	Після встановлення може працювати автономно	+		
2.	Призначення	аналіз даних в системі медико-санітарного моніторингу	є універсальною системою, модуль медико-санітарного аналізу треба закуповувати додатково	на базі програмного забезпечення можна створити модуль аналізу показників здоров'я		+	

Продовження таблиці 5.2.

3.	Ергономічність	Інтегрований модуль в готовий інтерфейс, результати можна зберегти у вигляді зображень	Необхідне попереднє навчання користувача для роботи з системою	Треба самому створювати інтерфейс та шукати набори даних			+
4.	Кросплатформність	Повністю кросплатформена веб-орієнтована система	ОС Microsoft Windows	ОС Microsoft Windows			+

5.2 Технологічний аудит ідеї проекту

У цьому підрозділі буде наведено аналіз аудиту технологій, за допомогою яких можна реалізувати ідеї проекту. Для визначення технологічної здійсненності ідеї проекту потрібно проаналізувати такі складові:

— за якою технологією буде виготовлено товар згідно з ідеєю проекту?

— чи існують такі технології, чи їх потрібно розробити/доробити?

— чи доступні такі технології авторам проекту?

У таблиці 5.3 наведено технологічну здійсненність ідеї проекту.

Таблиця 5.3. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1.	Перетворення даних у робочий формат	Pycharm, Sublime Text, CSV-reader	Наявна	Доступні для вільного користування

Продовження таблиці 5.3.

2.	База даних	СКБД MySQL	Наявна	Умовна безкоштовно
3.	Алгоритми розв’язання проблеми аналізу даних	PyCharm, pandas	Наявна	Доступні для вільного користування
4.	Засоби візуалізації результатів досліджень	MatplotLib, Seaborn	Наявна	Доступні для вільного користування
Висновок: проект реалізувати можливо				

Обрані технології є доступними для вільного користування та надають можливості для реалізації поставленої задачі.

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Передбачення і визначення ринкових можливостей, які можна використати під час ринкової імплементації проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дає змогу спланувати напрями розвитку проекту із урахуванням стану ринку, аналогічних пропозицій проектів конкурентів.

Спочатку проводиться аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 5.4).

Таблиця 5.4. — Характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	2
2	Загальний обсяг продажів, грн/ум.од	20000 грн
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	15

Середня норма рентабельності в галузі (або по ринку) порівнюється із банківським відсотком на вкладення. За умови, що останній є вищим, можливо, має сенс вкласти кошти в інший проект.

За результатами аналізу таблиці робиться висновок щодо того, чи є ринок привабливим для входження за попереднім оцінюванням.

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (таблиця 5.5).

Таблиця 5.5. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару

Продовження таблиці 5.5.

1	вирішення проблеми аналізу різнорідних масивів даних у системі медично-санітарного моніторингу	Експерти у медичній сфері, компанії моніторингу довкілля	Відмінностей між групами не має.	Стабільна робота продукту зі зрозумілим користувацьким інтерфейсом. Можливість працювати без інсталяції

Аналіз ринкового середовища складається з таблиць факторів, що перешкоджають ринковому впровадженню проекту, а також тих, що сприяють. Таблиці 5.6 та 5.7 представляють результати.

Таблиця 5.6. Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1.	Велика клієнтська база конкурентів	Конкуренти з досвідом продукту мають сильну базу клієнтів	Розвиток маркетингової кампанії та акційних пропозицій
2	Підходить для нових проектів	Потребує визначеної структури бази даних	Імпорт схеми бази даних

Продовження таблиці 5.6.

3	Зміна потреб користувача	Користувачам потрібні інші функції застосунку	Розширення застосунку новим функціоналом

Таблиця 5.7. Фактори можливостей

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1.	Незалежність від платформи	Система може працювати на будь-якій операційній системі за допомогою браузера та мережі Інтернет	Вихід на ринок мобільних додатків
2	Фокус продукту на можливостях, яких немає у конкурентів	Реалізація нових функціональних можливостей для користувачів	Планування задач та розподілення між розробниками
3	Відсутність повноцінних альтернатив	Наявні альтернативи не надають такий набір можливостей	Розширення набору можливостей

Надалі проводиться аналіз пропозиції: визначаються загальні риси конкуренції на ринку (таблиця 5.8).

Таблиця 5.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства
1. Вказати тип конкуренції: чиста	Існують фірми конкуренти на ринку	Презентація продукту на виставках, надати ширші можливості
2. За рівнем конкурентної боротьби: національний	Закордонні конкуренти	Публікація статей на міжнародних сайтах, можливість вибору мови ПЗ
3. За рівнем конкурентної боротьби: внутрішньогалузева	Спостерігається в пропозиціях на покупку ПЗ, якості функцій.	Розробка інтегрованого програмного забезпечення
4.Конкуренція за видами товарів: товарно-видова	Види товарів однакові – програмне забезпечення	Розроблене ПЗ повинно враховувати недоліки конкурентів
5.За характером конкурентних переваг: нецінова	Функціональні можливості	Надавати такі функціональні можливості, що не мають конкуренти
6.За інтенсивністю – не марочна	Конкуренція за надання послуг	Популяризація власного проекту

Результати аналізу умов конкуренції в галузі подано у таблиці 5.9

Таблиця 5.9. — Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Немає прямих конкурентів, тільки методи рішень	Наявність товарних знаків, доступ до ресурсів	Основним постачальником є інтернет ресурси	Система-інформації	немає
Висновки	Конкурента боротьба проходить з іншими розробниками гнучкого ПЗ	Є можливість виходу на ринок, готові ПЗ на базі наявної методології не є вузьконаправленими	Постачальники не диктують умов праці	Умови клієнтів змінюються у залежності від ситуації	-

Аналіз показує, що робота на ІТ арені України можлива, бо готових гнучких рішень не існує.

Обґрунтування факторів конкурентоспроможності наведено у таблиці 5.10.

Таблиця 5.10. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування
1	Функціональна перевага над конкурентами	Система є гнучкою і спрямована під специфічного користувача
2	Простота та інтуїтивність використання	Користувач викликає аналіз даних у своєму модулі користувача

Аналіз сильних та слабких сторін стартап-проекту наведений у таблиці 5.11.

Таблиця 5.11 Порівняльний аналіз сильних та слабких сторін програмного забезпечення.

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг програм-конкурентів у порівнянні з розробленим ПЗ						
			-3	-2	-1	0	+1	+2	+3
1	Споживчі потреби	18		+					
2	Основна результативність	15				+			
3	Технічне обслуговування	8						+	
4	Маркетинговий потенціал	4					+		

Аналіз проекту за схемою SWOT наведено у таблиці 5.12.

Таблиця 5.12. SWOT-аналіз

Сильні сторони: Якість продукту. Простий інтерфейс користувача. Функціональні можливості Доступність	Слабкі сторони: «Невідомість» програмного забезпечення Низька репутація
Можливості: Вихід на міжнародний ринок	Загрози: Блокування інтернет-ресурсу програмного забезпечення

На основі SWOT-аналізу були розроблені альтернативи ринкової поведінки, що представлені у таблиці 5.13.

Таблиця 5.13 Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива	Ймовірність отримання ресурсів	Строки реалізації
1	Розширення функціональності	25%	18 міс
2	Проведення конференції-демо для закордонних користувачів	50%	6 міс

Спочатку потрібно вивести на основний ринок розроблену систему, а вже потім шукати можливості розширення програмного функціоналу для користувачів.

5.4 Розробка ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: було проведено опис цільових груп потенційних споживачів (таблиця 5.14)

Таблиця 5.14 — Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Фізичні особи	Потребують недовгих переговорів	Низький	Висока	Складно
2	Підприємства	Готові	Високий	Висока	Середня
3	Державні установи	Готові	Високий	Висока	Середня
4	Стартапери	Готові	Середній	Високий	Просто
Висновок: цільова група – підприємства та державні установи					

Базова стратегія розвитку представлена у таблиці 5.15.

Таблиця 5.15. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку

Продовження таблиці 5.15.

1	Розширення виробничої лінії	Стратегія диференціації	Розширення функціоналу	Стратегія диференціації
2	Надання товару важливих з точки зору споживача відмітних властивостей	Стратегія диференціації	Кросплатформність забезпечує швидкість розробки, що необхідна для стартап-діячів	Стратегія диференціації

Вибір стратегії конкурентної поведінки представлено на таблиці 5.16.

Таблиця 5.16. Визначення базової стратегії конкурентної поведінки.

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати наявних у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Шукати нових споживачів	Так, треба збільшити підтримку мобільних пристроїв	Стратегія позиціонування

Визначення стратегії позиціонування наведено у таблиці 5.17

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкуренто спроможні позиції власного стартап проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
1	Ціна, якість	Стратегія спеціалізації (спирається на диференціацію)	Потребують швидкості розробки, яку надає підтримка багатьох платформ даним продуктом	Якість, надійність, кросплатформність

5.5 Розробка маркетингової компанії проекту

Початковим етапом є формування маркетингової концепції товару. У таблиці 5.18 наведені ключові концепції потенційного товару.

Таблиця 5.18. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами

Продовження таблиці 5.18.

1	Аналіз медичних даних	Отримання оброблених даних, з яких можна виконати візуалізацію та зробити висновки	Аналіз різних чинників, що впливають на здоров’я людини та виокремлення незвичностей
2	Візуалізація отриманих результатів	Полегшення роботи з отриманими даними для користувача-експерта у галузі	Користувачу легше робити висновки на основі отриманих результатів

Надалі розробляється трирівнева маркетингова модель товару: уточнюється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання. Опис трьох рівнів моделі товару наведено у таблиці 5.19.

Таблиця 5.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Програмний засіб для аналізу впливу навколишніх чинників на здоров’я людини		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. Зручність 2. Інтуїтивність 3. Швидкодія	1. М 2. М 3. М	4. Тл 5. Тл 6. Тх
	Якість: тестування та регресія		
	Пакування відсутнє		
	Марка: КЕЕЕМ Аналіз медико-санітарної частини		

Продовження таблиці 5.19.

III. Товар із підкріпленням	До продажу: Пробна 30-денна версія
	Після продажу: 24/7 підтримка
За рахунок чого потенційний товар буде захищено від копіювання: патент на продукт	

М/Нм – монотонні або немонотонні; Вр/Тх/Тл/Е/Ор – вартісні, технічні, технологічні, ергономічні або органолептичні (останній – для харчових продуктів).

Таблиця 5.20 представляє межі встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	2000-4000 грн.	35000 грн.	Всі	2000-35000

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення (таблиця 5.21)

Таблиця 5.21 Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту

Продовження таблиці 5.21.

1	Отримати більше за менші гроші	Клієнтська база	Тільки виробник	Вертикальна маркетингова система
2	Клієнт повинен надаватися в режимах “пробний” та “повний”.	Простота доступу	Розробник, веб-сайт, користувач	Збут силами посередника

Концепція маркетингових комунікацій подана у таблиці 5.22.

Таблиця 5.22. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Клієнти дізнаються про продукт з реклами, соціальних мереж	Інтернет, соціальні мережі	Довіра до бренду, ціна	Розповсюдження інформації про продукт	Висвітлення ключових особливих даних про продукт

Висновки до розділу.

Розроблене програмне забезпечення є конкурентним на ринку та має свої переваги над конкурентами. Поставлені маркетингові стратегії та шляхи збуту забезпечать подальший розвиток продукту. Підприємства та державні установи – основна цільова аудиторія, яка може використовувати системи для аналізу стану здоров’я людини в залежності від навколишніх чинників.

ВИСНОВКИ

Під час виконання магістерської дисертації було створено програмне забезпечення, що збирає та оброблює медичні і екологічні дані у системі комплексного Еко-Енерго-Економічного моніторингу, проводить статистичний аналіз зібраних даних і повертає користувачу у вигляді матриці кореляції.

Призначення програмного продукту полягає у покращені існуючої системи моніторингу модулем статистичного аналізу, щоб користувач, що є експертом-медиком у системі робив висновки на базі вхідних та оброблених даних.

Також під час дослідження готового програмного забезпечення статистичного аналізу у медико-санітарній сфері були проаналізовані SAS та IBM SPSS Statistics, що пропонували комплексне рішення задачі, проте обидві системи виявилися дороговартісними та недостатньо гнучким, а відсутність можливості інтеграції у готову систему КЕЕЕМ виключила обидва програмні забезпечення з можливих вирішень проблеми.

При аналізі кращого методу візуалізації залежності було розглянуто декілька видів графіків та обрано матрицю кореляції – під задачу інтерфейсу та для найкращої демонстрації відносин великої кількості даних.

Для роботи з даними і подальшої їх обробки була обрана мова програмування Python 3 та математично-наукові бібліотеки NumPy та Pandas – дані бібліотеки гарантують великий вибір та гнучкість у розробці програмного забезпечення. Бібліотеки Pandas та MatplotLib використовувалися як найкращі рішення серед можливих для візуалізації типів даних DataFrame бібліотеки Pandas.

Наразі, розроблене програмне забезпечення інтегроване у веб-додаток системи КЕЕЕМ у якості віджету та дозволяє користувачу обирати по області кількість захворювань (до 12) та чинників екологічної ситуації, що впливають на здоров’я населення (до 6), має 4 види візуалізації вхідних даних та 1 вид візуалізації оброблених даних. Для того, щоб мати гнучкий інструмент, можна збільшити кількість графіків для відображення або додати нову форму відображення – щоб користувач мав максимальну кількість інструментів статистичного аналізу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Verlan. A. Energy Sector and Sustainable Development Risks caused by the Climate Changes / Karaieva N, Verlan A. // Sustainable Development and Energy Security of the World and Ukrainian regions: Conflicts, Policy, Green Technologies: monograph / Edited by N. Karaieva. – K.: Tampodek XXI, 2012. – P.12-20
2. Verlan. A Application of fuzzy sets for determining the level of Sustainable Development of Ukraine / Karaieva N, Verlan A. // Sustainable Development and Energy Security of the World and Ukrainian regions: Conflicts, Policy, Green Technologies: monograph / Edited by N. Karaieva. – K.: Tampodek XXI, 2012. – c.62-72
3. Чайка В. Є. Урбоекологія / В. Є. Чайка. — Вінниця, 1999. — 368 с.
4. Analysis of observational health care using SAS URL: https://support.sas.com/content/dam/SAS/support/en/books/analysis-of-observational-health-care-data-using-sas/61876_excerpt.pdf (дата звернення: 16.11.2022).
5. Analysing data using SPSS URL: https://students.shu.ac.uk/lits/it/documents/pdf/analysing_data_using_spss.pdf (дата звернення 11.11.2022)
6. Основи статистики та аналізу даних - Український центр суспільних даних. Український центр суспільних даних. URL: https://socialdata.org.ua/manual/manual4/ (дата звернення: 11.11.2022).
7. Полягушко Л.Г, Левицький І.В., Костриця С.В. Еталонне тестування та аналіз реляційних баз даних моніторингових систем з великими обсягами даних. III Міжнародна студентська конференція “Розвиток суспільства та науки в умовах цифрової трансформації”, Луцьк, Україна, 16 грудня 2022.
8. Левицький І.В., Полягушко Л.Г. Единий реєстр наукових статей. V International Scientific and Practical Conference “Topical Issues of Modern Science, Society and

Education”, Харків, Україна, 28 листопада 2021.

9. Handbook of statistical analysis URL: http://fidy.andrianasy.free.fr/SAS%20Books/++!++%20A%20Handbook%20Of%20Statistical %20Analyses%20Using%20SAS.pdf (дата звернення: 11.11.2022)
10. Artur Araujo – Statistical Analysis of Series Using SAS URL: https://eprints.whiterose.ac.uk/137964/1/Artur_Araujo_Statistical_Analysis_of_ Series_of_N-of-1_Trials_Using_SAS_201809.pdf (дата звернення: 11.11.2022)
11. Офіційний сайт ПЗ IBM SPSS URL: https://www.ibm.com/products/spss-statistics
12. Посібник "Analysing data using SPSS" URL: https://students.shu.ac.uk/lits/it/documents/pdf/analysing_data_using_spss.pdf (дата звернення: 11.11.2022)
13. Посібник "SPSS Data Analysis in Medical Use" URL: https://students.shu.ac.uk/lits/it/documents/pdf/analysing_data_using_spss.pdf (дата звернення: 11.11.2022)
14. Social Big-Data Analysis of Particulate Matter, Health, and Society. PubMed Central (PMC). URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6801971/ (date of access: 11.11.2022).
15. NumPy Documentation. NumPy. URL: https://numpy.org/doc/ (date of access: 21.10.2022)..
16. pandas documentation – pandas 1.5.2 documentation. pandas - Python Data Analysis Library. URL: https://pandas.pydata.org/docs/ (date of access: 01.11.2022).
17. Merge, join, concatenate and compare – pandas 1.5.2 documentation. pandas - Python Data Analysis Library. URL: https://pandas.pydata.org/docs/user_guide/merging.html (date of access: 10.11.2022).
18. Article: Data Wrangling in Python. GeeksforGeeks. URL: https://www.geeksforgeeks.org/data-wrangling-in-python/ (date of access: 11.11.2022).
19. Article: Operations in Pandas. OReilly Media. URL:

https://www.oreilly.com/content/operations-in-pandas/ (date of access: 11.11.2022).
20. Correlation matrix : A quick start guide. STHDA - Accueil. URL: http://www.sthda.com/english/wiki/correlation-matrix-a-quick-start-guide-to-analyze-format-and-visualize-a-correlation-matrix-using-r-software (date of access: 11.11.2022).
21. Документація: Matplotlib. Matplotlib – Visualization with Python. URL: https://matplotlib.org (дата звернення: 11.11.2022).
22. Документація: seaborn 0.12.1. URL: https://seaborn.pydata.org/index.html (дата звернення: 11.11.2022).
23. Документація: seaborn 0.12.1. seaborn: statistical data visualization. URL: https://seaborn.pydata.org/tutorial/introduction.html (дата звернення: 11.11.2022).
24. Стаття: What are comma separated values URL: https://www.sage.com/en-gb/blog/glossary/what-are-comma-separated-values/ (дата звернення: 17.11.2022).
25. Левицький, І. В. База даних структурованих наукових матеріалів та пошукових запитів : дипломна робота бакалавра : 122 Комп'ютерні науки / Левицький Ілля Вячеславович. – Київ, 2021. – 10-26 с. URL: https://ela.kpi.ua/handle/123456789/43968 (дата звернення: 10.10.2022)
26. Документація: MySQL :: MySQL Workbench. MySQL. URL: https://www.mysql.com/products/workbench/ (дата звернення: 21.10.2022).
27. Документація: Node JS. URL: https://nodejs.org/en/docs/ (дата звернення: 21.10.2022).
28. Стаття: How to Run a Python Script using Node.js. plainenglish.io. URL: https://plainenglish.io/blog/how-to-run-python-script-using-node-js-6b351169e916 (дата звернення: 19.11.2022).
29. Офіційний сайт: Sublime Text. URL: https://www.sublimetext.com (дата звернення: 19.11.2022).
30. Стаття: Jet brains Pycharm. Hacker News. URL: https://news.ycombinator.com/item?id=1097772 (дата звернення: 11.11.2022).
31. Стаття: Is Visual Studio Code Really The Best Code Editor? URL:

https://www.tabnine.com/blog/visual-studio-code-really-the-best-code-editor/ (дата звернення: 11.11.2022).

32. Стаття: Використання кореляції на практиці URL: https://pidru4niki.com/14990528/ekonomika/korelyatsiyniy_analiz (дата звернення 5.12.2022)
33. Стаття: Знаходження кореляції Пірсона методами мови програмування Python URL: https://stackabuse.com/calculating-pearson-correlation-coefficient-in-python-with-numpy/ (дата звернення: 5.12.2022)
34. Документація: Choosing color palettes in seaborn. URL: https://seaborn.pydata.org/tutorial/color_palettes.html (дата звернення 5.12.2022)
35. Стаття: Матриця коваріативності. URL: https://mathworld.wolfram.com/ CovarianceMatrix.html (дата звернення: 5.12.2022)
36. Стаття: Адаптація вхідної матриці від користувача. URL: https://www.geeksforgeeks .org/take-matrix-input-from-user-in-python/ (дата звернення: 8.12.2022)
37. Стаття: Covariance Matrix. EMS Press. URL: https://www.encyclopediaofmath.org/ index.php?title=Covariance_matrix (дата звернення: 1.12.2022)
38. Стаття: Аналіз узгодженості двох рядів. URL: http://blacknick.info/index.php?subj=stat13 (дата звернення: 18.11.2022)
39. Офіційний сайт: RECASS NT. URL: http://www.rpatyphoon.ru/ products/software-hardware/recass.php (дата звернення: 18.11.2022)
40. Стаття: Модели прогнозирования: общая классификация. Чучуева И. URL: https://www.mbureau.ru/blog/modeli-prognozirovaniya-obshchaya-klassifikaciya.

ДОДАТОК А

Статистичний аналіз різнорідних масивів даних у системі санітарно-гігієнічного моніторингу

Апробації

УКР.НТУУ“КПІ ім. Ігоря Сікорського” ТР-1388мп_22М

Аркушів 8

2022

ЕТАЛОННЕ ТЕСТУВАННЯ ТА АНАЛІЗ РЕЛЯЦІЙНИХ БАЗ ДАНИХ МОНІТОРИНГОВИХ СИСТЕМ З ВЕЛИКИМИ ОБСЯГАМИ ДАНИХ

Костриця Сергій Володимирович
здобувач вищої освіти інституту атомної та теплової енергетики
Національний технічний університет «Київський політехнічний інститут ім. Ігоря Сікорського», Україна
Левицький Ілля В'ячеславович
здобувач вищої освіти інституту атомної та теплової енергетики
Національний технічний університет «Київський політехнічний інститут ім. Ігоря Сікорського», Україна
Науковий керівник: Полягушко Любов Григорівна
канд. тех. наук, доцент кафедри цифрових технологій в енергетиці ІАТЕ
Національний технічний університет «Київський політехнічний інститут ім. Ігоря Сікорського», Україна

Моніторингові системи завжди повинні працювати з актуальними даними, тому задля забезпечення системи новою інформацією використовується велика кількість датчиків і пристроїв, що отримують сирі (необроблені) дані з різних точок земного шару та неперервно виконують операції вставки до бази даних моніторингових систем. З часом база набирає великі обсяги даних і у випадку поганого проектування схеми та некоректних налаштувань – робота з базою даних сильно уповільнюється.

Прикладом погано побудованої системи з великими обсягами даних є система комплексного еколого-економіко-енергетичного моніторингу (КЕЕЕМ), тому для дослідження конкретних проблемних місць у її роботі було застосовано еталонне тестування до бази даних системи.

Еталонне тестування – це створення робочого навантаження, призначеного для того, щоб перевести систему у стресовий режим. Основною ціллю еталонного тестування є вивчення поведінки системи, однак не менш важливими причинами для

його виконання є відтворення бажаного стану системи та випробування нового обладнання на стійкість [1].

Існують дві основні стратегії еталонного тестування:

- *повнотекстове* – тестування всього додатку, включаючи веб-сервер, програмний код додатку, мережу та базу даних. При повнотекстовому тестуванні можна зрозуміти чи є СКБД вузьким місцем та як веде себе кеш кожної частини додатку;
- *компонентне* (модульне) – тестування лише СКБД, що допомагає порівнювати різні схеми та запити; протестувати конкретну проблему, яка виникла в додатку; уникнути довготривалого еталонного тестування, обмежуючись коротким тестом, який дозволить швидко внести зміни та виміряти їх.

Гарним інструментом компонентного тестування є *sysbench,* завдяки його багатопотоковій природі він дозволяє отримати представлення про швидкодію системи з точки зору таких важливих для роботи факторів як файловий ввід / вивід, планувальник операційної системи, розподіл пам’яті і швидкості передачі даних, потоків POSIX та самого сервера бази даних.

Для інтерактивних системи розрахованих на велику кількість користувачів краще всього підходить еталонний тест, який виміряє швидкодію оперативної обробки транзакцій (OLTP). Загальноприйнятою одиницею виміру є кількість транзакцій на секунду [2].

Було проведено таке тестування до бази даних КЕЕЕМ за допомогою інструменту *sysbench* для таблиці, що містить 1 мільйон записів показників з датчиків. Тест виконувався 60 секунд в режимі читання даних із використанням восьми конкурентних потоків (рис. 1).

```
kostr@MacBook-Pro ~ % sysbench /opt/homebrew/share/sysbench/tests/include/oltp_legacy/oltp.lua \
--oltp-table-size=1000000 --mysql-host=keem.com.ua --mysql-db=KEEM_Magistr \
--mysql-user=Keem_Magistr --mysql-password=KeEm_MaGiStR2022 \
> --time=60 --oltp-read-only=on --max-requests=0 --threads=8 run
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 8
Initializing random number generator from current time


Initializing worker threads...

Threads started!

SQL statistics:
    queries performed:
        read:                            65632
        write:                           0
        other:                           9376
        total:                           75008
    transactions:                        4688   (78.02 per sec.)
    queries:                             75008  (1248.24 per sec.)
    ignored errors:                      0      (0.00 per sec.)
    reconnects:                          0      (0.00 per sec.)

General statistics:
    total time:                          60.0903s
    total number of events:              4688

Latency (ms):
         min:                                   80.90
         avg:                                  102.46
         max:                                  232.77
         95th percentile:                        0.00
         sum:                                480320.22

Threads fairness:
    events (avg/stddev):           586.0000/3.57
    execution time (avg/stddev):   60.0400/0.03
```

Рис. 1. **Результат виконання першого тесту OLTP**

Отримані результати показали, що потоки завантажені нерівномірно і це вплинуло на швидкість виконання транзакцій. Тому в першу чергу було виконано огляд параметрів серверу бази даних, внаслідок чого було виявлено некоректні налаштування конкурентного доступу, які не були достатніми для системи з таким великим обсягом даних.

Після виправлень було виконано повторне тестування (рис. 2).

```
kostr@MacBook-Pro ~ % sysbench /opt/homebrew/share/sysbench/tests/include/oltp_legacy/oltp.lua \
--oltp-table-size=1000000 --mysql-db=KEEEM_Magistr_Optimized \
--mysql-user=root --mysql-password=password \
--time=60 --oltp-read-only=on --max-requests=0 --threads=8 run
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 8
Initializing random number generator from current time


Initializing worker threads...

Threads started!

SQL statistics:
    queries performed:
        read:                            3892182
        write:                           0
        other:                           556026
        total:                           4448208
    transactions:                        278013 (4633.27 per sec.)
    queries:                             4448208 (74132.29 per sec.)
    ignored errors:                      0      (0.00 per sec.)
    reconnects:                          0      (0.00 per sec.)

General statistics:
    total time:                          60.0033s
    total number of events:              278013

Latency (ms):
         min:                                    0.59
         avg:                                    1.73
         max:                                   29.62
         95th percentile:                        0.00
         sum:                               479837.36

Threads fairness:
    events (avg/stddev):           34751.6250/95.25
    execution time (avg/stddev):   59.9797/0.00
```

Рис. 2. **Результат виконання другого тесту OLTP**

Отримані результати повторного тестування бази даних KEEEM показали ефективність виконаних налаштувань, які пришвидшили роботу системи приблизно в 60 разів.

Еталонні тести OLTP дуже зручні для швидкого порівняння різних систем і диску є неоціненними для виправлення несправностей та ізолювання помилкових компонентів при виникненні проблем зі швидкодією системи.

У результаті проведення еталонних тестів OLTP до бази даних системи KEEEM було виявлено некоректні налаштування конкурентного доступу і виправлено згідно з вимогами до баз даних великих обсягів.

Список використаних джерел:

1. Schwarts B., Zaitsev P., Tkachenko V. Benchmarking MySQL. *High Performance MySQL*. 3rd ed. USA, 2012. P. 35–67.

2. Using sysbench for OLTP workload performance benchmark. *FlamingBytes*. URL: https://www.flamingbytes.com/posts/sysbench/ (дата звернення: 07.12.2022).

СЕРТИФІКАТ УЧАСНИКА

ЛЕВИЦЬКИЙ ІЛЛЯ ВЯЧЕСЛАВОВИЧ

ВЗЯВ(-ЛА) УЧАСТЬ У III МІЖНАРОДНІЙ СТУДЕНТСЬКІЙ НАУКОВІЙ КОНФЕРЕНЦІЇ

РОЗВИТОК СУСПІЛЬСТВА ТА НАУКИ В УМОВАХ ЦИФРОВОЇ ТРАНСФОРМАЦІЇ

16 ГРУДНЯ 2022 | М. ЛУЦЬК, УКРАЇНА

Конференцію схвалено УКРІНТЕІ (Посвідчення №456 від 05.10.2022)
Матеріали учасника конференції опубліковані та знаходяться у відкритому доступі на умовах ліцензії CC BY 4.0 за посиланням: https://ojs.ukrlogos.in.ua/index.php/liga/issue/view/16.12.2022

ДИРЕКТОР МОЛОДІЖНОЇ НАУКОВОЇ ЛІГИ
ГОЛОВА ОРГКОМІТЕТУ КОНФЕРЕНЦІЇ
ІГОР КОРЕНЮК

УДК 004.77:378.02

ЛЕВИЦЬКИЙ І.В., магістрант гр. ТР-13мп.
Керівник – ст.викл., к.т.н. Полягушко Л.Г.

ЄДИНИЙ РЕЄСТР НАУКОВИХ СТАТЕЙ

Вступ. Багато науково-педагогічних працівників та студентів при дослідженні предметної області, в якій воні зацікавлені, шукають інформацію в інтернеті в різних джерелах, що забирає певну кількість часу. На сьогоднішній день існує велика кількість електронних наукових бібліотек, що забезпечують доступ до потрібної інформації, проте необхідно витрачати багато часу, щоб знайти велику кількість інформації за обраною темою. Одним із популярних наукових ресурсів на сьогоднішній день є Wikipedia, де редагувати статті в ній може кожний пересічний користувач мережі «Інтернет», тому інформація частіше є суб'єктивною ніж правдивою у Wikipedia та схожих енциклопедіях. На відміну від Wikipedia, всі інші електронні наукові портали не містять у собі суб'єктивних правок, а лише статті, написані визнаними науковцями з усього світу – від історичних статей до пояснення принципу роботи винаходу в галузі електромеханіки.

Мета роботи. Головною метою дослідження є створення програмного продукту, який буде об'єднувати найбільш популярні наукові бібліотеки в одному місці. Дана система буде пропонувати тільки оригінальні статті зазначених авторів, серед яких користувач зможе обрати найбільш цікавішу для нього статтю.

Матеріали та методи. Було проаналізовано наступні схожі ресурси та рішення: систему Google Scholar, web-додаток ScienceResearch, наукову бібліотеку eLibrary, науковий сайт SCOPUS, сайт ліги націй AGRIS, веб-сайт CORE, збірник статей Cyberleninka та сайт наукових новин Індикатор. Після аналізу виокремлено наступні важливі функції, які будуть корисними при створенні власного застосунку:

1) авторизація користувача на сайті;
2) наявність доступу щонайменше до однієї бази наукових статей;
3) можливість пошуку статті у пошуковому полі;
4) відповідь на запит у вигляді, як правило, декількох статей;
5) наявність фільтрації запитів.

Вхідною інформацією є дані, що відправляються в БД за допомогою тонкого налаштування Django ORM у серверній частині проекту. Як правило, це дані пошукового запиту, що вже надходив раніше. До вихідних даних відносяться сутності, що формуються з наданої користувачем інформації та його пошукових запитів.

Для розроблення програмного продукту була обрана база даних PostgreSQL. Для того, щоб задовільнити логіку формату відповіді від зовнішнього API (рис.1), необхідно виокремити наступні сутності: користувач, пошуковий запит, стаття, закладки користувача, історія пошуку користувача, посилання на документи, історія усіх пошукових запитів. Для миттєвого реагування на зміні у базі даних потрібно прописати тригери та таймстампи. Для розгортання застосунку використовується хостинг-сервіс Heroku.

```
{
    "status": "200",            // HTTP status code
    "error": "string",          // Наявність помилки - може бути порожнім полем
    "total": 12,                // Усього співпадінь за запитом
    "data": [                   // Дані - масив
        {
            "id": "string",     // Ідентифікатор
            "title": "string",  // Заголовок
            "topics": [         // Теми - масив рядків
                "string",       //
                "string"        //
            ],
            "authors": [        // Автори - масив рядків
                "string",       //
                "string"        //
            ],
            "description": "string",    // Опис - рядок
            "language": "en",           // Мова, використовується стандарт ISO 639-1
            "publisher": "string",      // Видавництво
            "url": "string",            // Посилання
            "year": "string"}           // Date parsed to string

    ],
    "searchDate": "string",             // Date parsed to string
    "request": {                        // Дані про запит
        "searchQuery": "string",        // Запит повністю
        "filters": {                    // Фільтрація за:
            "title": "string",          // Назвою
            "topics": [                 // Темами
                "string",               //
                "string"                //
            ],
            "authors": "string, ",      // Авторами
            "publisher": "string",      // Видавництвом
            "language": "en",           // Мовою
            "year": "startYear - endYear"// Роками публікації
        }
    }
}
```

Рис.1.Формат відповіді зовнішнього API

Результати. Взаємодія користувача з додатком через інтерфейс (рис. 2) виконується через наступні елементи управління: а верхньому барі можемо обрати тему та мову. У текстовому полі вводимо запит, у ньому ж праворуч є кнопка фільтрації, а окремо праворуч кнопка пошуку. Під текстовим полем виводяться результати пошуку, якщо ввести запит та натиснути кнопку «Пошук».

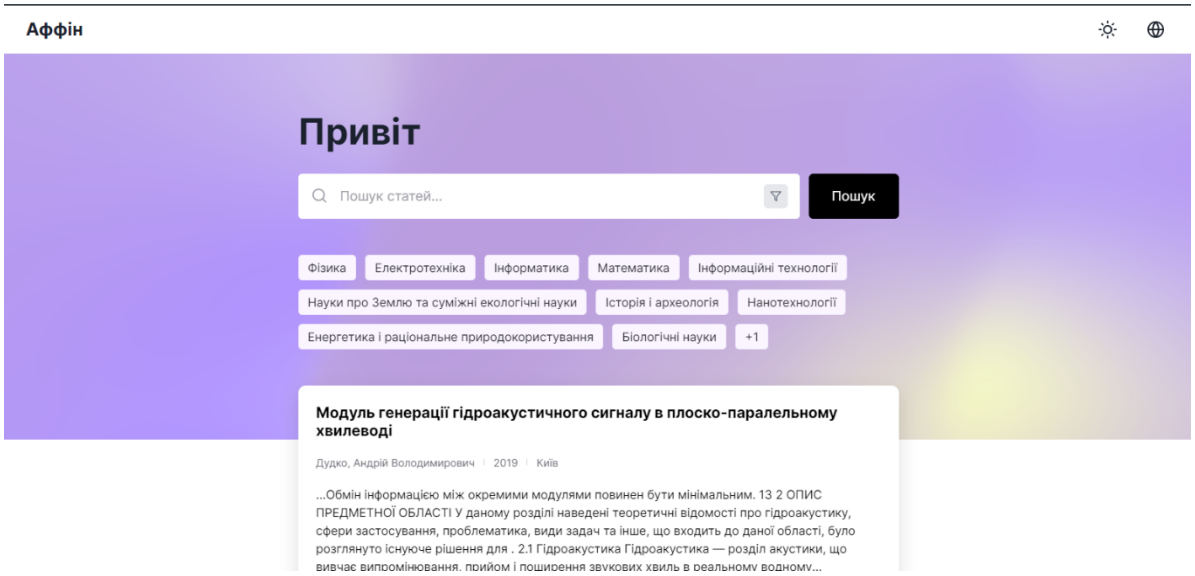

Рис.2. Інтерфейс веб-додатку

Висновки. В ході дослідження було створено застосунок, який є доволі потужним інструментом для пошуку необхідних статей, а також фундаментом для створення мережі пошуку наукових статей при додані більшої кількості зовнішніх API, поліпшенні фільтрації, покращенні хостингу.

Література:

1. Моргунов Е.П. PostgreSQL. Основи мови SQL: підручник/ Е. П. Моргунов; під ред. Е. В. Рогова, П. В. Лузанова. – СПб.: БХВ-Петербург, 2018. – 336 с.

2. Edition Neil Middleton Heroku: Up and Running: Effortless Application Deployment and Scaling 1st Edition. – Вид-во: O'Reilly Media (November 24, 2013) – 100 с.


CERTIFICATE
is awarded to
Levytskyi Illia
for being an active participant in
V International Scientific and Practical Conference
"TOPICAL ISSUES OF MODERN SCIENCE, SOCIETY AND EDUCATION"
24 Hours of Participation
(0,8 ECTS credits)
KHARKIV
28-30 November 2021
INTERNATIONAL SCIENTIFIC AND PRACTICAL CONFERENCE
TOPICAL ISSUES OF MODERN SCIENCE, SOCIETY AND EDUCATION
SCI-CONF.COM.UA
KHARKIV
28-30 NOVEMBER 2021
sci-conf.com.ua