Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій

**Магістерська кваліфікаційна робота на тему:**

# «ІНФОРМАЦІЙНА СИСТЕМА АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ У РІЧЦІ ПІВДЕННИЙ БУГ»

Виконав: студент 2 курсу, групи 2ІСТ-22м
спеціальності 126 – «Інформаційні системи та технології»
_____ Денис МАРЕЦЬКИЙ

Керівник: к.т.н., доц.каф. САІТ
_____Євгеній КРИЖАНОВСЬКИЙ
«01» 12 2023 р.

Рецензент: к.т.н., доц. каф. КН
_____Володимир ОЗЕРАНСЬКИЙ
«05» 12 2023 р.

**Допущено до захисту**

Завідувач кафедри САІТ
_____ д.т.н., проф. Віталій МОКІН
«01» 12 2023 р.

Вінниця ВНТУ – 2023 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра системного аналізу та інформаційних технологій
Рівень вищої освіти – ІІ-й (магістерський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 126 Інформаційні системи та технології
Освітньо-професійна програма – Інформаційні технології аналізу даних та зображень

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ
д.т.н., проф. Віталій МОКІН
«01» 09 2023 р.

**ЗАВДАННЯ**
**НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**
Марецькому Денису Євгенійовичу

1. Тема роботи: "Інформаційна система аналізу та прогнозування якості води у річці Південний Буг",
керівник роботи: Євгеній КРИЖАНОВСЬКИЙ, к.т.н., доц.каф. САІТ,
затверджені наказом закладу вищої освіти від «18» 09 2023 року №247
2. Строк подання студентом роботи «30» 11 2023 року
3. Вихідні дані до роботи:
Датасет Kaggle «River Water Quality EDA and Forecasting» з даними про вміст речовин в поверхневих водах річки Південний Буг на різних ділянках;
Дані державного моніторингу поверхневих вод: https://data.gov.ua/dataset/surface-water-monitoring
4. Зміст текстової частини:
   1) Характеристика проблеми аналізу та прогнозування якості води в річці Південний Буг;
   2) Вибір оптимальних інформаційних технологій;
   3) Розроблення інформаційної системи аналізу та прогнозування якості води у річці Південний Буг;
   4) Економічна частина.
5. Перелік ілюстративного матеріалу:
   1) алгоритм роботи програми;
   2) Схема бази даних;
   3) Ілюстрація мап та даних геоприв'язки;
   4) Графіки результатів розвідувального аналізу даних;
   5) Результати розробленої інформаційної системи.

6. Консультанти розділів МКР

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| 4 | Наталія БУРЄННІКОВА, д.е.н., проф. каф. ЕПВМ | 01.11.23 | 15.11.23 |

7. Дата видачі завдання «04» 09 2023 року

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва та зміст етапу | Термін виконання | | Примітка |
|---|---|---|---|---|
| | | початок | закінчення | |
| 1 | Аналіз предметної області | 04.09 | 20.09 | вик |
| 2 | Вибір оптимальних інформаційних технологій та проведення розвідувального аналізу | 20.09 | 10.10 | вик |
| 3 | Розроблення інформаційної системи | 10.10 | 20.10 | вик |
| 4 | Тестування розробленої інформаційної системи | 20.10 | 01.11 | вик |
| 5 | Економічна частина | 01.11 | 15.11 | вик |
| 6 | Оформлення матеріалів до захисту МКР | 15.11 | 30.11 | вик |

Студент _____ Денис МАРЕЦЬКИЙ

Керівник роботи _____ Євгеній КРИЖАНОВСЬКИ

# АНОТАЦІЯ

УДК 004.9+ 574.55

Марецький Д.Є. Інформаційна система аналізу та прогнозування якості води у річці Південний Буг. Магістерська кваліфікаційна робота зі спеціальності 126 – інформаційні системи та технології, освітньо-професійна програма – інформаційні технології аналізу даних та зображень. Вінниця: ВНТУ, 2023. 105 с.

На укр. мові. Бібліогр.: 32 назв; рис.: 71; табл.: 13.

В магістерській кваліфікаційній роботі розроблено концепцію інформаційної системи аналізу та прогнозування якості води у річці Південний Буг. Вибрано оптимальні технології та за їх допомогою було розроблено систему у вигляді веб-додатку, роботу системи було випробувано на авторському датасеті, що було створено з наявних даних про вміст забруднюючих речовин в річці Південний Буг.

Ілюстративна частина складається з 10 плакатів із результатами тестування розробленої системи.

У розділі економічної частини розглянуто питання про доцільність розробки та впровадження інформаційної системи аналізу та прогнозування якості води у річці Південний Буг.

Ключові слова: Python, PHP, MySQL, OpenStreetMap інформаційна система, проблеми навколишнього середовища, ГІС

# ABSTRACT

Maretskyi D.E. Information system for analysis and forecasting of water quality in the Southern Bug River. Master's thesis in specialty 126 - Information systems and technologies, educational and professional program - Information technologies of data and image analysis. Vinnytsia: VNTU, 2023. 105 p.

In Ukrainian. Bibliography: 32 titles; Figs. 71; Tables: 13.

The master's thesis developed the concept of an information system for analyzing and forecasting water quality in the Southern Bug River. The optimal technologies were selected and used to develop the system in the form of a web application, the system was tested on the author's dataset, which was created from the available data on the content of pollutants in the Southern Bug River.

The illustrative part consists of 10 posters with the results of testing the developed system.

The economic section considers the feasibility of developing and implementing an information system for analyzing and forecasting water quality in the Southern Bug River.



Keywords: Python, PHP, MySQL, OpenStreetMap information system, environmental problems, GIS.

# ЗМІСТ

ВСТУП ..... 4
1 ХАРАКТЕРИСТИКА ПРОБЛЕМИ МОНІТОРИНГУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ ..... 6
1.1 Аналіз предметної області ..... 6
1.2 Огляд забруднюючих речовин та їх нормування ..... 8
1.3 Технологія збору даних та моніторингу забруднення поверхневих вод ..... 12
1.4 Сучасний стан моніторингу та аналізу якості води у річці Південний Буг ..... 13
1.5 Особливості реалізації системи аналізу та прогнозування якості води у річці Південний Буг ..... 16
1.6 Висновки ..... 16
2 ОСНОВНІ ЕТАПИ ВИКОНАННЯ РОБОТИ ТА ОГЛЯД НАБОРУ ВХІДНИХ ДАНИХ ..... 17
2.1 Вибір оптимальних інформаційних технологій. ..... 17
2.1.1 Огляд та вибір технології зберігання та відображення даних ..... 17
2.1.2 Огляд та вибір ГІС технологій ..... 21
2.1.3 Огляд та вибір технологій для аналізу та прогнозування даних ..... 24
2.1.4 Огляд та вибір технологій для реалізації системи у вигляді Web-додатку ..... 28
2.1.5 Середовище реалізації технології аналізу та прогнозування ..... 31
2.1.6 Огляд та вибір середовища для реалізації системи у вигляді Web-додатку ..... 34
2.2 Огляд вхідних даних ..... 35
2.3 Першочергова обробка даних. Розвідувальний аналіз ..... 38
2.4 Вибір моделей ..... 47
2.6 Висновки ..... 50
3 РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ СИСТЕМИ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ В РІЧЦІ ПІВДЕННИЙ БУГ ..... 51
3.1 Розроблення інформаційної технології аналізу та прогнозування для інформаційної системи ..... 51
3.2 Розробка бази даних для інформаційної системи ..... 59
3.3 Реалізація системи у веб–додатку ..... 60
3.4 Реалізація ГІС технології в системі ..... 63

3.5 Застосування інформаційної системи ..... 66
3.6 Висновки ..... 73
4 ЕКОНОМІЧНА ЧАСТИНА ..... 75
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки ..... 76
4.2 Визначення рівня конкурентоспроможності розробки ..... 79
4.3 Розрахунок витрат на проведення науково-дослідної роботи ..... 82
4.3.1 Витрати на оплату праці ..... 82
4.3.2 Відрахування на соціальні заходи ..... 85
4.3.3 Сировина та матеріали ..... 85
4.3.4 Розрахунок витрат на комплектуючі ..... 87
4.3.5 Спецустаткування для наукових (експериментальних) робіт ..... 87
4.3.6 Програмне забезпечення для наукових (експериментальних) робіт ..... 88
4.3.7 Амортизація обладнання, програмних засобів та приміщень ..... 89
4.3.8 Паливо та енергія для науково-виробничих цілей ..... 91
4.3.9 Службові відрядження ..... 92
4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації ..... 93
4.3.11 Інші витрати ..... 93
4.3.12 Накладні (загальновиробничі) витрати ..... 93
4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором ..... 95
4.5 Висновки ..... 99
ВИСНОВКИ ..... 101
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ..... 103
Додаток А (обов’язковий). Технічне завдання ..... 106
Додаток Б (обов’язковий). Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень ..... 109
Додаток В (довідковий). Лістинг програми ..... 110
Додаток Г (обов’язковий). Ілюстративна частина ..... 113

# ВСТУП

**Актуальність теми.** На сьогоднішній день проблематика забруднення прісних запасів води є вкрай важливим питанням – за роки активної індустріалізації навколишнє середовище нашої планети пережила великі зміни, здебільшого вкрай негативні, що також негативно вплинуло на життя людей, забруднення прісних запасів води спричинило до збільшення захворюваності, дефіцит риби в річках, озерах та інших водних басейнах.

Аналіз якості поверхневих вод є комплексною процедурою під час якої завдяки об'єднанню великої кількості даних, складається узагальнена картина поточної ситуації з якістю води на різних ділянках та в самому басейні річки Південний Буг загалом.

Прогнозування даних допоможе побудувати прогнозований масив показників, що показують загальну характеристику якості води, тим самим з'ясувавши тренд якості води в майбутньому – чи погіршиться якість води в майбутньому, чи покращиться.

Об'єднання технологій аналізу та прогнозування дозволить розробити унікальну систему, що по своїм завданням не буде мати відповідних аналогів як окремого рішення в сфері інформаційних технологій, саме тому актуальність розробки даної системи є вкрай високою.

**Мета і завдання роботи.** Метою роботи є розроблення інформаційної системи аналізу та прогнозування якості води у річці Південний Буг.

Для досягнення поставленої мети потрібно виконати такі завдання:

– розроблення веб системи з можливістю імпорту бібліотек для аналізу даних, розвідувального аналізу даних, та побудови моделей;

– розвідувальний аналіз даних;

– побудова моделей;

– застосування моделей для прогнозування якості вод.

**Об'єктом дослідження** магістерської кваліфікаційної роботи є процес аналізу та прогнозування якості води в річці Південний Буг.

**Предметом дослідження** магістерської кваліфікаційної роботи є методи аналізу, прогнозування даних та проектування інформаційних систем.

**Методи дослідження.** У дослідженнях було використано методи розвідувального аналізу даних та прогнозування даних. Також було використано методи побудови web-системи, що включали в себе також проектування бази даних та використання методів геоінформаційного аналізу даних.

**Новизна одержаних результатів.** Новизна полягає в тому, що дістала подальший розвиток інформаційна аналітична система моніторингу поверхневих вод Південного Бугу за рахунок імплементації нових функцій таких як прогнозування, а також актуалізації даних.

**Апробація результатів магістерської кваліфікаційної роботи.** Результати кваліфікаційної роботи подані на LIII Науково-технічну конференцію факультету інтелектуальних інформаційних технологій та автоматизації (м. Вінниця, 2023-2024 рр.).

**Публікації результатів магістерської кваліфікаційної роботи.** За результатами даної роботи опубліковано тези на конференції ВНТУ, а саме LIII Науково-технічній конференції факультету інтелектуальних інформаційних технологій та автоматизації (м. Вінниця, 2023-2024 рр.), на тему «Аналіз та прогнозування якості води в річці Південний Буг на основі даних моніторингу за 2000 – 2023 роки» [1].

# 1 ХАРАКТЕРИСТИКА ПРОБЛЕМИ МОНІТОРИНГУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ

## 1.1 Аналіз предметної області

Проблема забруднення навколишнього середовища на даний час є вкрай важливим та глобальним питанням, адже проблема забруднення прісних запасів води напряму впливає на якість життя людей.

За останніми даними лише 2.5% всієї води на планеті Земля є запасами прісної води, решта становлять океани та моря, притому лише 30 відсотків з запасів прісної води знаходяться в ґрунтових водах, озерах, річках та вологих ґрунтах, решта це льодовики та водяна пара в атмосфері. Оскільки вода є вкрай необхідною для всіх живих істот на планеті, якість питної води впливає на забезпечення різних галузей таких як водопостачання, рибальство, туристичну галузь, екосистему водних басейнів та інше [2].

В період з XX по XXI сторіччя, проблемою забруднення води почали вкрай гостро турбуватись суспільство, тим самим було створено безліч асоціацій з питань охорони навколишнього середовище, а також багато людей одноосібно, або організовувавшись в групи, привертають увагу суспільства до цього питання за допомогою мітингів та засобів масової інформації.

В більшості цивілізованих країн питання моніторингу та контролювання якості води водних об’єктів, що знаходяться під управлінням країн, вирішується на державному рівні – за рахунок державних коштів створюються установи, зазвичай на регіональному рівні, основною метою яких є аналіз вмісту шкідливих речовин та координація дій по забезпеченню задовільного стану води в регіоні. Так наприклад, в Україні працюють дрібні установи такі як асоціація «Питна Вода» основним завданням якої є співпраця з державними установами задля покращення стану води, головним державним інструментом контролю якості води в Україні є Державне агентство водних ресурсів

України. Дане агентство напряму підпорядковується Міністру захисту довкілля та природних ресурсів [3].

Також існують глобальні установи основна задача яких є покращення стану навколишнього середовища, в тому числі підвищення якості води у водних басейнах планети. Серед організацій, що підтримуються суспільством та фінансуються за рахунок благодійності найпопулярнішою є організація Грінпіс.

Грінпіс (англ. GreenPeace) — це міжнародна неприбуткова екологічна організація, заснована у 1971 році, що здійснює свою діяльність у понад 55 країнах світу. Метою організації є захист природи, боротьба із забрудненням навколишнього середовища та сприяння сталому використанню природних ресурсів. Грінпіс, активно займається проблемами збереження якості води та водних ресурсів. Організація фокусується на захисті океанів від забруднення, включаючи протидію викидам пластику та хімічних речовин. Вона також здійснює кампанії з боротьби із забрудненням вод від промислових викидів та інших джерел, сприяючи усвідомленню глобальних проблем екосистем та ресурсів води. Діяльність Грінпіс також орієнтована на захист річок та озер від забруднення та відновлення екологічно стійкого використання водних ресурсів [4].

Також існують органи управління, що фінансуються безпосередньо органами влади, або об’єднаннями держав. Так наприклад, на платформі Організації Об'єднаних Націй (ООН) функціонує спеціалізована ініціатива — Глобальна ініціатива ООН з водних ресурсів (UN-Water). Ця ініціатива створена для координації дій та співпраці між різними агентствами та програмами ООН, а також між міжнародними та регіональними організаціями з питань води.

Основні завдання та функції Глобальної ініціативи ООН з водних ресурсів включають:

- Координація: UN-Water допомагає в узгодженні робіт різних агентств ООН та інших стейкхолдерів у галузі водних ресурсів.

- Сприяння Співпраці: Ініціатива створює умови для підтримки співпраці між різними галузями, включаючи здоров'я, освіту, сільське господарство та інші, для вирішення проблем водних ресурсів.
- Поширення Інформації: UN-Water забезпечує обмін інформацією та дослідженнями щодо водних питань, сприяючи світовому розумінню проблем в цій галузі [5].

Враховуючи актуальність проблеми забруднення вод, а також намагання державних, міжнародних організацій, а також суспільства в покращенні якості води в світі, можна дійти до висновку, що розроблення інформаційної системи аналізу та прогнозування даних є доцільним.

### 1.2 Огляд забруднюючих речовин та їх нормування

До основних сполук, що досліджуються під час моніторингу водних об’єктів є хімічні сполуки, що так чи інакше негативно впливають на якість самої води. Такі сполуки зазвичай потрапляють у водойми різними шляхами, такими як стікання земельних вод, атмосферний осад, промислові викиди та інші джерела.

Одним з основних шляхів занесення забруднюючих речовин у водойму є стікання з поверхні ґрунту. Наприклад, використання агрохімікатів, таких як пестициди та добрива, може призводити до вивільнення хімічних сполук у ґрунт, які потім переносяться водяним потоком у річки та озера. Це може призвести до забруднення води різноманітними хімічними речовинами, такими як азотні та фосфорні сполуки.

Забруднюючі речовини також можуть потрапляти в атмосферу та осідатися на поверхні води. Наприклад, промислові викиди, включаючи токсичні гази та частки, можуть осідатися в поверхневих водах, призводячи до забруднення. Це може впливати на якість води та екосистему водойм.

Механізми транспорту забруднюючих речовин у водні середовища також включають протікання стічних вод від промислових об'єктів та міських

каналізаційних систем. У цих стічних водах можуть міститися різні хімічні сполуки, бактерії та інші забруднюючі речовини, що можуть негативно впливати на водні екосистеми.

Загальні наслідки забруднення води включають втрату біорізноманіття, зниження якості питної води, загрозу здоров'ю людини та інші негативні екологічні ефекти [6].

Хоча під час дослідження в зразках находять велику кількість хімічних сполук, серед них виділяють кілька ключових показників що в сукупності дозволяють описати загальну якість води. Нижче наведено список основних показників, що будуть досліджуватись під час розроблення інформаційної системи:

Амоній (хім. форм. $NH_4$) - іон, походження якого пов'язане з різноманітними джерелами, зокрема сільським господарством, розкладом органічних речовин та іншими джерелами водних систем. Його високі концентрації можуть спричинити дисбаланс в розчиненні кисню в воді, що негативно впливає на акватичні екосистеми та рибний склад. Крім того, відзначається його потенційною загрозою для здоров'я людини при використанні води для пиття.

Біохімічне споживання кисню (БСК-5) є показником, що вказує на кількість кисню, яку мікроорганізми витрачають при біохімічному окисленні органічних речовин у водному середовищі протягом п'яти діб. Цей показник використовується для оцінки біологічної активності води та ступеня забруднення органічними речовинами. Процедура визначення БСК-5 включає вимірювання зміни концентрації розчинених кисневмісних сполук у воді протягом п'яти діб. Водний зразок поміщається у спеціальний контейнер, де мікроорганізми мають можливість витрачати кисень при біохімічних процесах. Після п'ятиденного періоду вимірюється зміна концентрації кисню, що дає інформацію про розмір біохімічного споживання кисню. Високий БСК-5 може свідчити про значну кількість органічних речовин у воді, що підвищує активність мікроорганізмів. Зменшення цього показника може свідчити про

поліпшення якості води. Оцінка БСК-5 є важливою для контролю та підтримки екологічної стабільності водних систем.

Суспензія - це показник, який характеризує кількість та розмір твердих часток, які утримуються у воді чи іншому розчиннику протягом певного періоду часу. Суспензія може містити різноманітні частки, такі як глина, пісок чи органічні залишки. Оцінка суспензії важлива для визначення якості води та може служити індикатором забруднення або ерозії водних джерел.

Вміст кисню в воді – показник, що вказує на кількість розчиненого кисню в воді. Цей показник є ключовим для життя водних організмів, оскільки вони залежать від кисню для дихання та метаболічних процесів. Зниження вмісту кисню може бути наслідком забруднення води або екологічних змін у водоймах. Високий вміст кисню свідчить про стабільні та здорові водні умови, тоді як його низькі рівні призводять до стресу для водних організмів та порушення екологічного балансу

Нітрати (хім. форм. $NO_3$) – іони азоту та кисню, що потрапляють в воду внаслідок застосування азотних добрив та розкладу органічних речовин. Підвищені рівні нітратів можуть вказувати на агропромислове забруднення. Високі концентрації нітратів можуть мати негативний вплив на водні екосистеми та представляти загрозу для здоров'я при використанні води для пиття.

Нітрити (хім. форм. $NO_2$) - іони, що містять азот та кисень, їх вміст у воді може виникнути внаслідок процесів розкладу азотних сполук. Високі рівні нітритів можуть вказувати на незадовільну обробку стічних вод або інші джерела забруднення. Ці іони можуть стати токсичними для риб і інших водних організмів, а також можуть впливати на якість води для пиття.

Сульфати (хім. форм. $SO_4$) - солі, що містять аніон сульфату. Вони потрапляють в воду внаслідок природних процесів, таких як розщеплення мінералів або вивільнення в атмосфері. Високі рівні сульфатів можуть виникнути внаслідок промислової діяльності або водної ерозії.

Хлориди ( хім. форм. Cl) - солі, що містять аніон хлориду. Їх можна знайти в воді як результат природних процесів, таких як розщеплення мінералів чи вивільнення в атмосфері. З іншого боку, підвищені рівні хлоридів можуть виникнути внаслідок антропогенного впливу, наприклад, від використання солі на дорогах або від викидів промислових джерел. Хлориди, як правило, не є токсичними для багатьох видів водних організмів, але їх великі концентрації можуть впливати на смак води та створювати проблеми для використання води в господарстві [7].

Для контролю вмісту вище описаних показників в поверхневих водах, були введені міжнародні граничні значення кожного показника, суттєве перевищення яких вказує на критичну ситуацію вмісту того чи іншого показника в водному об'єкті, дані показники отримали назву гранично допустимих концентрацій (ГДК) та поділяються на дві категорії:

– Гранично допустимі величини для водойм рибогосподарського призначення ($ГДК_{РГ}$);

– Гранично допустимі величини для водойм господарсько–побутового використання($ГДК_{ГП}$).

На таблиці 1.1 вказано гранично допустимі концентрації.

Таблиця 1.1 – Нормативи показників якості води

| Гідрохімічний показник | ГДК | ГДК |
|---|---|---|
| | Для водойм рибогосподарського призначення ($ГДК_{РГ}$) | Для водойм господарсько-побутового використання ($ГДК_{ГП}$) |
| Розчинений кисень, мгО/дм$^3$ | >6,0 | >4,0 |
| БСК$_5$, мг$O_2$/дм$^3$ | -------- | 3,0 |
| Хлориди, мг/дм$^3$ | 300 | 350 |
| Сульфати, мг/дм$^3$ | 100 | 500 |
| Азот амонійний, мг/дм$^3$ | 0,39 | 1,5 |
| Азот нітратний, мг/дм$^3$ | 9,0 | 10,0 |
| Азот нітритний, мг/дм$^3$ | 0,02 | 1,0 |
| Фосфати, мг/дм$^3$ | -------- | 3,5 |

Дані показники будуть використовуватись для оцінки забруднення води під час розроблення інформаційної системи [8].

### 1.3 Технологія збору даних та моніторингу забруднення поверхневих вод

Оскільки основою аналізу та прогнозування якості води є саме вхідні дані, аналіз застосованих технологій для збору даних є вкрай необхідними для постановки та виконання поставлених задач.

У процесі розробки інформаційної системи аналізу та прогнозування річки Південний Буг використовуються вхідні дані, які отримані внаслідок збору інформації у різних точках басейну. Збір даних щодо забруднення поверхневих вод проводиться за допомогою різноманітних методів, кожен із яких відрізняється своєю точністю та тривалістю проведення.

Один із ключових методів збору даних полягає у проведенні лабораторних досліджень зразків води, зібраних з різних ділянок басейну. Цей метод вважається найточнішим і дозволяє вивчати багато параметрів води. Згідно з програмою досліджень, обов'язково визначаються показники, такі як пил, нітрати, сульфати, оксид вуглецю та свинець. Крім того, деякі станції розширюють перелік досліджуваних речовин у воді.

Інший метод, менш точний, надає загальну характеристику зразка, і зазвичай вказує на високий рівень концентрації певних речовин. Цей метод базується на загальних звітах та скаргах щодо стану води. Наприклад, якщо надходить скарга на неприємний запах води або дивний відтінок, то ця характеристика може свідчити про високий рівень певних показників.

Всі ці методі мають свої недоліки та переваги, але головною метою застосування їх в комплексі є забезпечення повноцінного та комплексного моніторингу якості води в річці Південний Буг. Враховуючи різноманітність методів збору даних, важливо забезпечити їхню взаємодію та інтеграцію для отримання найбільш об'єктивних результатів. Такий комплексний підхід

дозволяє ефективно вирішувати завдання аналізу та прогнозування якості води, щоб забезпечити сталий розвиток та збереження екологічної рівноваги в регіоні басейну річки Південний Буг [9].

### 1.4 Сучасний стан моніторингу та аналізу якості води у річці Південний Буг

Серед основних проблем аналізу якості води у річці Південний Буг можна виділити наступні фактори впливу:

- Відсутність уніфікованої інформаційної системи з усіх регіонів;
- Брак графічної інформації, що була б зрозумілою для користувачів;
- Не актуальні дані у вже існуючих системах.

Серед наявних рішень моніторингу поверхневих вод, основною масою проектів є проекти що надають мало візуальної інформації, натомість надаючи користувачам текстові дані для подальшого опрацювання. Можна виділити один з сервісів Державного агентство водних ресурсів, а саме Басейнове Управління водних ресурсів річки Південний Буг.

Басейнове управління водних ресурсів річки Південний Буг – це бюджетна організація, основними завданнями якої є координація діяльності організацій, що належать до сфери управління Державного агентства водних ресурсів України, а також забезпечення реалізації державної політики у сфері використання та управління поверхневий водних ресурсів в району басейну річки Південний Буг.

На офіційному сайті Басейнового управління водних ресурсів річки Південний Буг можна знайти мапу, де описані 4 точки досліджень якості води, а також актуальні дані в описаних точках вимірювання. На рисунках 1.1 та 1.2 зображено головну сторінка Басейнового управління водних ресурсів річки Південний Буг, а також мапу з точками досліджень.


Рисунок 1.1 – Головна сторінка Басейнового управління водних ресурсів річки Південний Буг

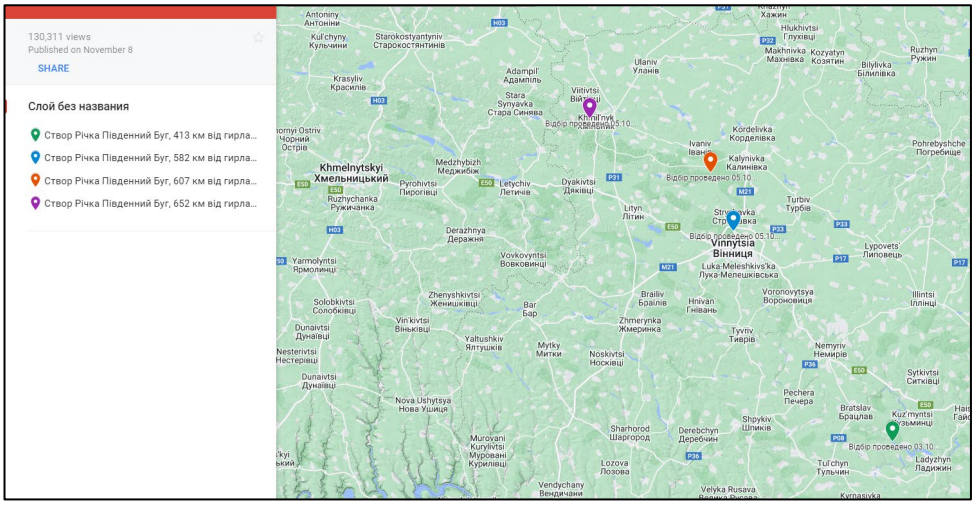



Рисунок 1.2 – Мапа з геоданими

Основними недоліками даної системи є брак графічного матеріалу, такого як діаграми, а також мала кількість досліджуваних станцій та наявність лише останніх актуальних даних на мапі .

В питанні саме аналізу та прогнозуванні даних, найкращими інформаційними рішеннями на даний момент є нотбуки учасників платформи Kaggle, де зібрано різні рішення проблеми аналізу та прогнозування даних якості води в річці Південний Буг. Сторінку датасету та наявних рішень аналізу та прогнозування зображено на рисунках 1.3 та 1.4.

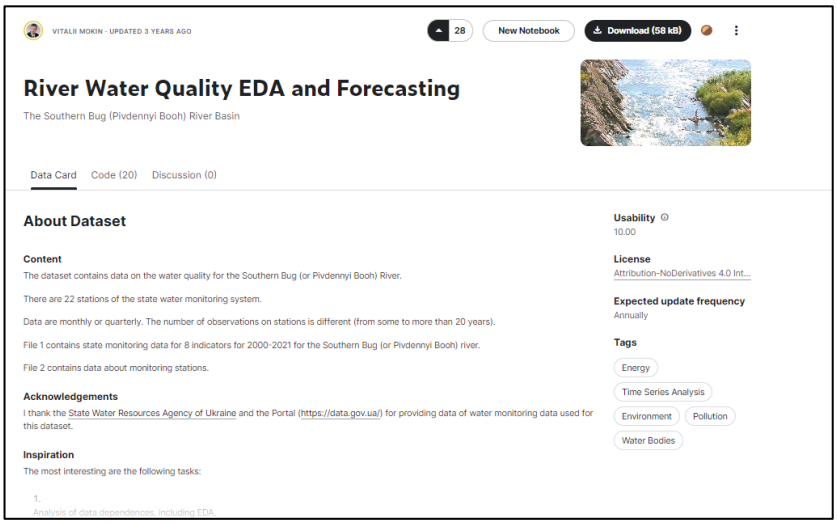



Рисунок 1.3 – Сторінка датасету аналізу та прогнозування якості води в річці Південний Буг на платформі Kaggle

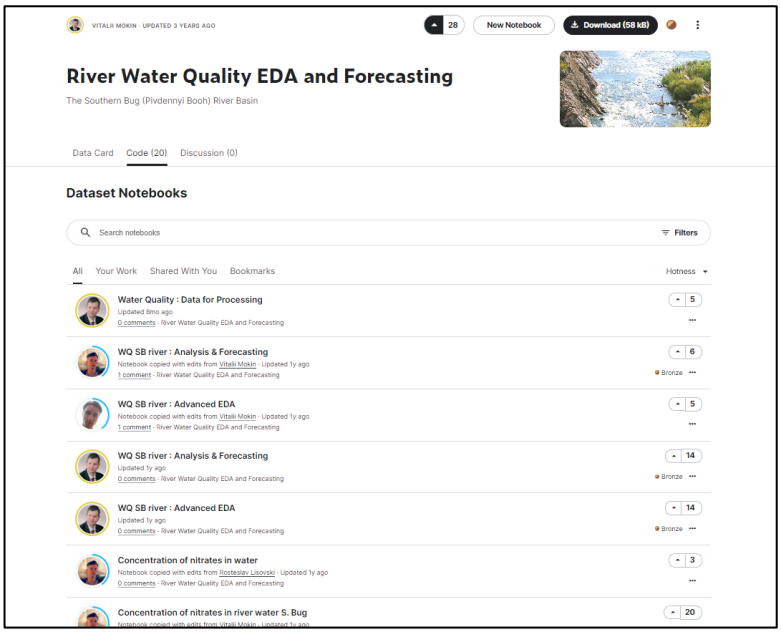



Рисунок 1.4 – Перелік рішень проблеми аналізу та прогнозування якості води у річці Південний Буг на платформі Kaggle

Дані рішення в тій чи іншій мірі вирішують завдання аналізу та прогнозування якості води, але серед основних недоліків даних рішень вирізняються саме відсутність актуальних даних, адже у всіх цих рішеннях використовуються лише дані з датасету «River Water Quality EDA and Forecasting», що обмежуються лише даними з 2000 по 2021 роки. Також

основним недоліком всіх цих рішень є відсутність самостійності як окремої системи, всі вони знаходяться на платформі Kaggle тому користувачам для огляду наявних рішень необхідно зареєструватись на сайті, а також мати базові знання мови програмування Python та її бібліотек.

## 1.5 Особливості реалізації системи аналізу та прогнозування якості води у річці Південний Буг

Основною проблемою реалізації інформаційної системи є актуалізація наявних даних – оскільки дані якості води на різних ділянках є непостійними, а дослідження в сфері моніторингу якості поверхневих вод здійснюються на регулярній основі, для зберігання актуальності самої системи необхідно регулярно оновлювати дані, базуючись на регулярних звітах. Також від отриманих звітів потрібно відфільтровувати інформацію, яка не має цінни в рамках даного дослідження.

Наступною проблемою розроблення системи є вибір технологій, а також середовища, що дозволять виконати всі поставлення завдання. Обрані технології та середовище повинні мати змогу обробляти велику кількість даних з достатньою швидкодією та підтримувати життєздатність за високої активності користувачів.

## 1.6 Висновки

У даному розділі визначено суть наявної проблеми, було розглянуто основні дані на основі яких буде будуватись система аналізу та прогнозування якості води у річці Південний Буг, описано гранично допустимі значення та описані технології збору даних. Також було розглянуто аналоги, що в тій чи іншій мірі частково виконують завдання кваліфікаційної роботи та описано особливості реалізації системи аналізу та прогнозування даних.

# 2 ОСНОВНІ ЕТАПИ ВИКОНАННЯ РОБОТИ ТА ОГЛЯД НАБОРУ ВХІДНИХ ДАНИХ

## 2.1 Вибір оптимальних інформаційних технологій.

У сучасному науково-дослідному контексті, вибір оптимальних інформаційних технологій є стратегічним завданням, яке визначається необхідністю ефективного збору, обробки та аналізу інформації. Цей вибір є ключовим етапом, оскільки від нього залежить ефективність збору, обробки та аналізу великої кількості даних, які використовуються для здійснення наукових досліджень та прийняття обґрунтованих рішень в галузі управління водними ресурсами.

Основними технологіями для розгляду та вибору найбільш оптимальних для виконання поставленої задачі є:

- Технології для зберігання вхідних даних та їх відображення в спроектованій системі;
- ГІС технології для відображення геоданих;
- Технології реалізації блоку аналізу та прогнозування даних;
- Технології об’єднання вище описаних технологій в Web-систему;
- Середовище виконання блоку аналізу та прогнозування;
- Середовище розроблення інформаційної системи аналізу та прогнозування.

Обравши всі необхідні технології для розроблення інформаційної системи, ми зможемо перейти до безпосереднього розроблення системи.

### 2.1.1 Огляд та вибір технології зберігання та відображення даних

Основою будь-якої інформаційної системи є дані, тому вибір технології даних є першочерговим завданням, що дозволить безпечно зберігати великі

об’єми даних та швидко звертатись до даних. Найкращим рішенням в даному питанні є реляційні бази даних.

Реляційна база даних (РБД) - це колекція даних, організованих у відповідності до принципів реляційної моделі даних. Реляційна модель даних була запропонована Едгаром Коддом у 1970 році та є широко використовуваною для створення та управління базами даних.

У реляційній базі даних дані представлені у вигляді таблиць, які складаються з рядків (кортежів) та колонок (атрибутів). Кожна таблиця визначає конкретний тип даних, а кожен рядок у таблиці представляє конкретний запис або кортеж. Кожна колонка представляє конкретний атрибут, а значення в кожній клітинці таблиці є даними, пов'язаними з відповідним рядком та колонкою.

Реляційні бази даних використовують мову структурованих запитів (SQL) для взаємодії з даними. Вони забезпечують ефективний та структурований спосіб організації, зберігання та отримання інформації, що робить їх популярними для різноманітних застосувань, включаючи корпоративні системи управління базами даних, веб-розробку та аналітику даних [10].

Переваги реляційних баз даних:

- Структурованість та чіткість: Реляційні бази даних дозволяють створювати чіткі та структуровані відносини між даними за допомогою таблиць.
- Запити SQL: Використання мови SQL дозволяє легко створювати складні запити для вибору, вставки, оновлення та видалення даних з бази.
- Цілісність даних: Реляційні бази даних підтримують інтегрітет даних, визначаючи унікальність, первинні та зовнішні ключі, що гарантує консистентність та точність даних.
- Підтримка транзакцій: РБД надають механізми для виконання транзакцій, що дозволяє забезпечувати атомарність, консистентність, ізольованість та довершеність операцій з базою даних.

Недоліки реляційних баз даних:

– Спрощеність моделі: Реляційна модель даних може виявитися занадто спрощеною для представлення складних структур даних, таких як зображення, графи та інші.

– Велика кількість зв'язків: Для представлення взаємодії між різними сутностями може знадобитися велика кількість зв'язків, що може ускладнити модель.

– Погана продуктивність при Великому обсязі даних: При великому обсязі даних реляційні бази можуть виявити погану продуктивність, особливо при складних операціях з'єднання (join) та запитах [11].

В питанні управління баз даних, як реляційних так і не реляційних баз виступають системи управління базами даних(СУБД).

СУБД або система управління базами даних (СУБД) — це програмне забезпечення, яке дозволяє створювати, управляти та взаємодіяти з базами даних. СУБД використовується для забезпечення ефективного та структурованого зберігання даних, а також для виконання різноманітних операцій з цими даними. Існує безліч СУБД, основна відмінність яких полягає в наявних інструментах, функціях, а також способом відображення даних.

Список реляційних СУБД:

– MySQL;
– PostgreSQL;
– Microsoft SQL Server;
– Oracle Database;
– SQLite;
– MariaDB;
– IBM Db2 [12].

Для розроблення інформаційної системи аналізу та прогнозування поверхневих вод було обрано СУБД MySQL.

MySQL - це система управління базами даних (СУБД), яка надає можливість створення, управління та взаємодії з реляційними базами даних.

Вона є відкритою та безкоштовною, і широко використовується веб-розробниками та програмістами для зберігання та отримання даних

MySQL відомий своєю надійністю, швидкодією та простотою використання. Ця система управління базами даних підтримує стандартні мови запитів SQL (Structured Query Language) і дозволяє виконувати різноманітні операції з даними, такі як вибірка, вставка, оновлення та видалення.

MySQL широко використовується в інтернет-проектах та додатках, що вимагають надійного та ефективного зберігання даних. Вона підтримує різноманітні операційні системи, включаючи Windows, Linux та macOS, що робить її вибором для багатьох розробників у всьому світі [13].

Плюси MySQL:

– Відкритий доступ: MySQL є відкритою СУБД, що означає, що її можна використовувати безкоштовно, і відкритий вихідний код дозволяє розробникам вносити зміни за їхніми потребами;

– Швидкодія: MySQL славиться високою продуктивністю та швидкодією виконання операцій з базою даних.;

– Підтримка широкого діапазону операційних систем: MySQL може працювати на різних операційних системах, таких як Windows, Linux та macOS;

– Легкість використання: MySQL має простий та легко зрозумілий SQL-діалект, що робить її доступною для широкого кола користувачів.

Мінуси MySQL:

– Обмежена підтримка деяких функцій: Порівняно з деякими комерційними аналогами, MySQL може не мати певних функцій або їхній рівень підтримки може бути обмеженим;

– Відсутність орієнтованих на підприємства функцій: Деякі функції, які можуть бути важливі для підприємств, можуть бути обмеженими в MySQL у порівнянні з деякими комерційними СУБД;

– Менша продуктивність за деякими сценаріями: за великим обсягом даних або складними операціями, продуктивність MySQL може бути менш ефективною порівняно з деякими іншими реляційними СУБД;

– Обмежена спеціалізація: Хоча MySQL підходить для багатьох випадків використання, вона може бути менш оптимальною для певних спеціалізованих завдань порівняно з деякими іншими рішеннями [14].

Дивлячись на особливість використання MySQL, її переваги та недоліки, можна дійти до висновку, що дана СУБД є відмінним рішенням для виконання поставлених задач розроблення системи аналізу та прогнозування якості води у річці Південний Буг.

### 2.1.2 Огляд та вибір ГІС технологій

Враховуючи що досліджуванні дані напряму пов’язані з геоданими, для подальшої візуалізації системи необхідно обрати відповідну технологію, що дозволить нам візуалізувати одну зі складових даних, а саме точок моніторингу даних, що дозволить користувачам простіше орієнтуватись в системі та обирати та вибирати саме ті дані, що їх цікавлять. В даному питанні найкращим рішенням будуть ГІС технології.

Геоінформаційна система (ГІС) є комплексною інформаційною технологією, спрямованою на збір, обробку, аналіз, візуалізацію та управління геопросторовою інформацією. Геопросторова інформація описується як дані, які пов'язані з конкретним місцем на Землі, включаючи географічні координати, висоти, кліматичні умови, природні ресурси та інші атрибути.

На даний момент існує безліч Геоінформаційних систем, нижче наведено список найпопулярніших систем даної галузі:

– ArcGIS (Esri);
– QGIS (Quantum GIS);
– GRASS GIS;
– MapInfo Professional;

– Google Earth Engine;
– PostGIS;
– OpenStreetMap (OSM);
– ERDAS IMAGINE;
– Leaflet;
– Carto [15].

Всі описані геоінфірмаційні системи мають свої переваги та недоліки, але для виконання поставлених завдань було обрано саме OpenStreetMap.

OpenStreetMap (OSM) - це проєкт, запущений у 2004 році, який націлений на створення та поширення вільних географічних даних та карт. OSM відкритий для участі будь-якої особи та визначеної команди розробників. Проєкт став одним з найважливіших джерел географічних даних у світі, доступних для всіх.

Основні характеристики OSM включають відкритий характер проекту. Всі дані, що збираються та публікуються у межах OSM, доступні під вільною ліцензією Open Database License (ODbL), яка дозволяє використовувати та переробляти їх для різних цілей, включаючи комерційні. Це робить OSM відкритим та гнучким джерелом географічних даних. На рисунку 2.1 показано приклад мапи OpenStreetMap.

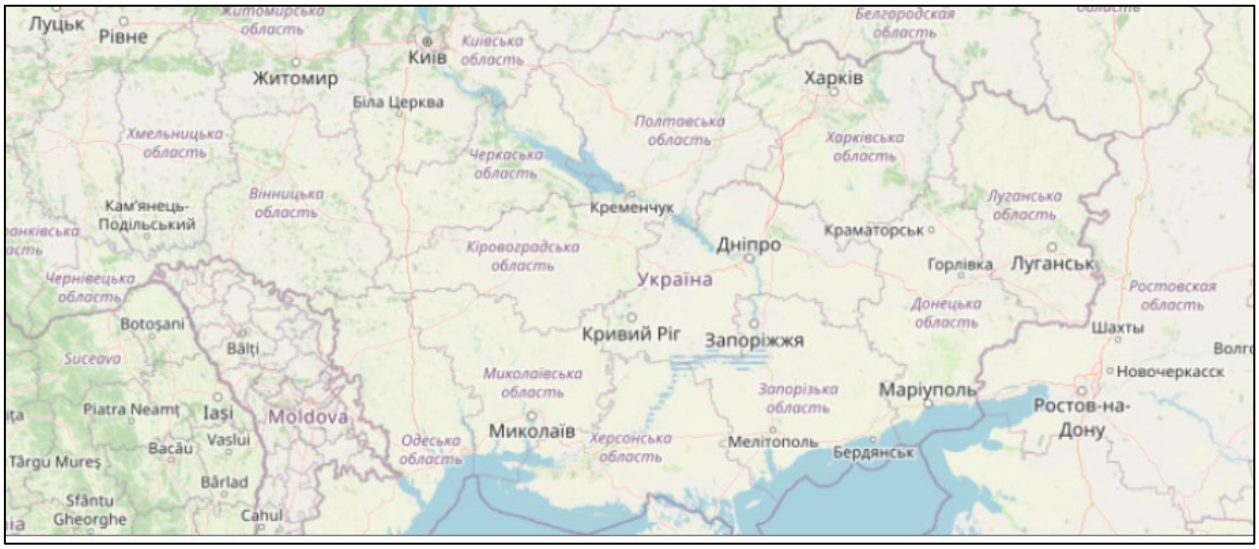



Рисунок 2.1 – Приклад мапи в ГІС OpenStreetMap

Однією з ключових переваг OSM є активна та глобальна спільнота користувачів та розробників. Ця спільнота постійно працює над поповненням та підтримкою даних OSM. Велика кількість користувачів, що редагують та вносять свій внесок у OSM, забезпечує постійне оновлення географічних даних.

OSM надає доступ до географічних даних, включаючи дані про дороги, річки, будівлі, ліси, гірські хребти та багато іншого. Ці дані можуть бути використані для створення різних типів карт та додатків. Вони широко використовуються у навігаційних програмах, мобільних додатках, геоінформаційних системах та для аналізу даних.

Окрім того, OSM пропонує різноманітне апаратне та програмне забезпечення для збору та редагування географічних даних. Від GPS-пристроїв та фотокамер до спеціалізованих програм для редагування даних, які спрощують процес створення та актуалізації інформації на мапі [16].

Завдяки своїй гнучкості та відкритості, OSM став важливим джерелом географічних даних, яке використовується в різних проектах та застосуваннях, забезпечуючи доступність та гнучкість для користувачів з усього світу.

Плюси OpenStreetMap (OSM):

– Відкритий вихідний код: OSM є проектом з відкритим вихідним кодом, що дозволяє користувачам вносити власні внески та зміни до карти.

– Безкоштовна інформація: Всі дані OSM є безкоштовними та доступними для використання без обмежень.

– Спільнота користувачів: Широка та активна спільнота користувачів та розробників сприяє розвитку та підтримці проекту.

– Широкий функціонал: OSM не обмежується лише картографією, включаючи в себе інші можливості, такі як реєстрація маршрутів, геокодування та інше.

– Гнучкість та підтримка проектів: Можливість використання карт OSM у різноманітних проектах, включаючи веб-сайти, додатки та дослідження.

Мінуси OpenStreetMap (OSM):

– Недостатній контроль якості: Оскільки дані надходять від користувачів, які різняться за досвідом та уважністю, може виникнути проблема недостатньої якості та актуальності деяких даних.

– Обмежений обсяг даних: За певними межами населених пунктів або популярних областей може бути менше інформації або вона може бути менш деталізованою.

– Залежність від користувачів: Проект може стикатися з проблемою недостатнього числа активних користувачів та внесків у деяких областях.

– Відсутність комплексних сервісів: У порівнянні з комерційними аналогами, OSM може не мати такого широкого спектру додаткових послуг та інтеграцій.

– Проблеми з конфіденційністю: Оскільки дані OSM є відкритими, може виникати питання щодо конфіденційності деякої інформації [17].

З урахуванням загальної характеристики описаної ГІС та дивлячись на її переваги та недоліки, можна дійти висновку, що OpenStreetMap є оптимальним рішенням в питані питання візуалізації та прив’язки геоданих.

### 2.1.3 Огляд та вибір технологій для аналізу та прогнозування даних

Наступним етапом в розроблені аналітичної системи є вибір мови програмування для аналізу даних та побудови моделей для подальшого передбачення.

На даній арені основними конкурентами є дві мови програмування, а саме Python та R.

Python - це високорівнева мова програмування, яка вирізняється простим та легко читабельним синтаксисом. Вона була розроблена Гвідо ван Россумом

та вперше випущена в 1991 році. Python відомий своєю легкістю вивчення, що робить його привабливим для новачків у програмуванні. Мова має чистий та зрозумілий синтаксис, що нагадує англійську мову.

Ще однією важливою особливістю Python є його інтерпретованість. Це означає, що програми на Python виконуються "на льоту" без необхідності компіляції, що робить процес розробки більш швидким та інтерактивним. Python є крос-платформеним і може працювати на різних операційних системах, що робить його універсальним і доступним для розробників на будь-якій платформі.

Однією з сильних сторін Python є велика кількість бібліотек та модулів, які розширюють його можливості. Для аналізу даних та машинного навчання існують такі бібліотеки, як NumPy, Pandas, Matplotlib, Scikit-Learn, TensorFlow та багато інших. Це робить Python популярним інструментом для розробки різних програм, включаючи веб-додатки, наукові дослідження, аналіз даних, робототехніку та багато інших галузей.

Python також активно підтримується активною спільнотою розробників та має велику кількість документації та онлайн-ресурсів для навчання та розвитку навичок програмування [18].

R є інтерпретованою мовою програмування, розробленою для виконання статистичних обчислень та аналізу даних. Вона була створена в 1990 році й до цього часу залишається потужним інструментом для статистичного моделювання та візуалізації даних.

R має високий рівень експресивності та дозволяє легко виконувати складні статистичні аналізи, включаючи лінійну та нелінійну регресію, аналіз дисперсії, кластерний аналіз і багато інших методів. Мова також підтримує векторизацію операцій, що робить її ефективною для роботи з великими наборами даних.

R відомий своєю багатою екосистемою пакетів і бібліотек, які дозволяють розширювати його можливості. Наприклад, пакет ggplot2 використовується для створення високоякісних графіків та візуалізації даних.

Багато пакетів розроблені спеціально для різних галузей, включаючи біоінформатику, економетрію, геоінформатику та інше.

Однією з переваг R є активна спільнота користувачів і розробників, яка підтримує мову та надає велику кількість документації та онлайн-ресурсів для навчання [19].

На основі переваг та недоліків кожної з мов програмувань то їх віхи, було обрано саме Python, оскільки завдяки його гнучкості та наявній кількості бібліотек, що ідеально підходить для виконання поставлених завдань.

Наступним кроком у питанні виконання аналізу та прогнозування буде вибір необхідних бібліотек для аналізу даних та побудови моделей.

NumPy – NumPy є потужною бібліотекою для наукового обчислення в мові програмування Python. Основною структурою даних є масиви, які можуть бути багатовимірними, що полегшує ефективну обробку та аналіз числових даних. У бібліотеці реалізовано велику кількість математичних функцій, що спрощує виконання операцій над великими обсягами даних. Однією з ключових особливостей є універсальні функції, які можна застосовувати елемент за елементом, що сприяє векторизації коду та полегшує обробку даних. Масиви NumPy також підтримують потужні операції індексації та зрізів, дозволяючи вибирати конкретні елементи чи підмасиви. Основною перевагою NumPy є його швидкодія та ефективність, завдяки використанню мови програмування C внутрішньо. Бібліотека широко використовується у наукових дослідженнях, аналізі даних та розробці алгоритмів завдяки своїм потужним функціоналом та високому рівню оптимізації [20].

Pandas є високопродуктивною бібліотекою для мови програмування Python, спеціалізованою на обробці та аналізі структурованих даних. Заснована на двох основних структурах - DataFrame та Series, Pandas надає зручний та потужний інтерфейс для операцій з даними. DataFrame володіє гнучкістю таблиці з можливістю обробки різноманітних типів даних, включаючи числові, текстові та категоріальні. Завдяки розширеним можливостям індексації, фільтрації та агрегації, Pandas стає потужним

інструментом для виконання складних операцій з даними. Бібліотека також підтримує зчитування та запис даних у різноманітних форматах, що полегшує взаємодію з різними джерелами інформації [21].

Matplotlib представляє собою впливову бібліотеку для створення графіків та візуалізації даних у Python. Вона використовується для зображення різноманітних типів графіків, спрощуючи сприйняття та аналіз інформації. Matplotlib забезпечує можливість створювати як базові візуалізації, так і більш складні графічні представлення даних. Ця бібліотека є необхідним інструментом для наукових досліджень, аналітики та представлення даних у середовищі Python [22].

Scikit-Learn, з важливою роллю у наукових дослідженнях та прикладних доменів машинного навчання, є комплексною бібліотекою для реалізації алгоритмів навчання з учителем та без учителя. Вона включає широкий спектр алгоритмів, від класичних методів до сучасних технік глибокого навчання, що дозволяє вченим і дослідникам ефективно моделювати та розв'язувати складні завдання. Scikit-Learn визначається своєю прозорою та зрозумілою реалізацією, яка дозволяє високий рівень контролю над параметрами моделей та гарантує надійність результатів. Завдяки широкій підтримці уніфікованих інтерфейсів, бібліотека сприяє стандартизації методів оцінки моделей та обробки даних, що робить її невід'ємною складовою для високоякісних досліджень та впроваджень в області машинного навчання [23].

XGBoost (eXtreme Gradient Boosting) є високоефективною бібліотекою для машинного навчання, базованою на алгоритмі градієнтного бустингу. Вона вирізняється високою ефективністю та гнучкістю у вирішенні завдань класифікації, регресії та ранжування. XGBoost використовує оптимізовані алгоритми для швидкого та точного навчання моделей, що робить її популярною серед дослідників та практиків в області машинного навчання. Завдяки вбудованим можливостям регуляризації та обробки відсутніх даних, XGBoost володіє високою стійкістю та здатністю до обробки великих обсягів

інформації. Бібліотека знаходить широке застосування в різноманітних галузях, включаючи фінанси, біотехнології та аналітику даних [24].

Glob це бібліотека в мові програмування Python, яка забезпечує зручний інтерфейс для вибору файлів та директорій в системі операцій за допомогою шаблонів імен та символів підстановки. Вона входить до складу стандартної бібліотеки Python і використовується для виконання операцій з файловою системою, враховуючи не лише конкретні імена файлів, але й гнучкі шаблони, що полегшують пошук та обробку файлових ресурсів [25].

Warnings – бібліотека що використовується для обробки та керування попередженнями, що можуть виникнути під час виконання програм. Вона дозволяє розробникам встановлювати фільтри для управління виведенням попереджень, включаючи їх ігнорування, конвертацію в винятки або виведення на екран. Ця бібліотека корисна для покращення управління повідомленнями про потенційні проблеми в коді під час вивчення та оптимізації програмних рішень [26].

### 2.1.4 Огляд та вибір технологій для реалізації системи у вигляді Web-додатку

Завершальним етапом розроблення системи є об’єднання всіх використаних технологій в один інтерфейс у вигляді Web-додатку, для даного завдання необхідно, обрати мову програмування для розробки веб-додатку. До найбільш популярних мов програмування, що включають в себе рішення розробки веб-додатків відносяться:

- JavaScript;
- Python;
- Ruby;
- Java;
- PHP;
- C#;

– Go (Golang);
– TypeScript [27].

Для розробки інформаційної системи аналізу та прогнозування якості Південного Бугу було обрано мову програмування PHP.

PHP – це високорівнева мова програмування, створена для веб-розробки. Ім'я PHP походить від "Personal Home Page," але тепер офіційно розшифровується як "PHP: Hypertext Preprocessor." Вона вбудована в HTML-код веб-сторінок і виконується на веб-сервері перед відправкою результату на клієнтський браузер. Мова відома своїм простим синтаксисом, що полегшує вставку PHP-коду в HTML, роблячи веб-сторінки динамічними. PHP взаємодіє з різними системами управління базами даних, що дозволяє створювати веб-додатки зі зберіганням та обробкою даних. Велика спільнота розробників та різні фреймворки та розширення роблять PHP популярним інструментом для веб-розробки. Основною його перевагою є наявність гнучких фреймворків на типу Laravel та Yii, систем керування вмістом (CMS – Content Management System) таких як WordPress, Moodle, Prestashop та інші. На рисунку 2.1 зображено системи керування вмістом та фреймворки, що базуються на мові програмування PHP.

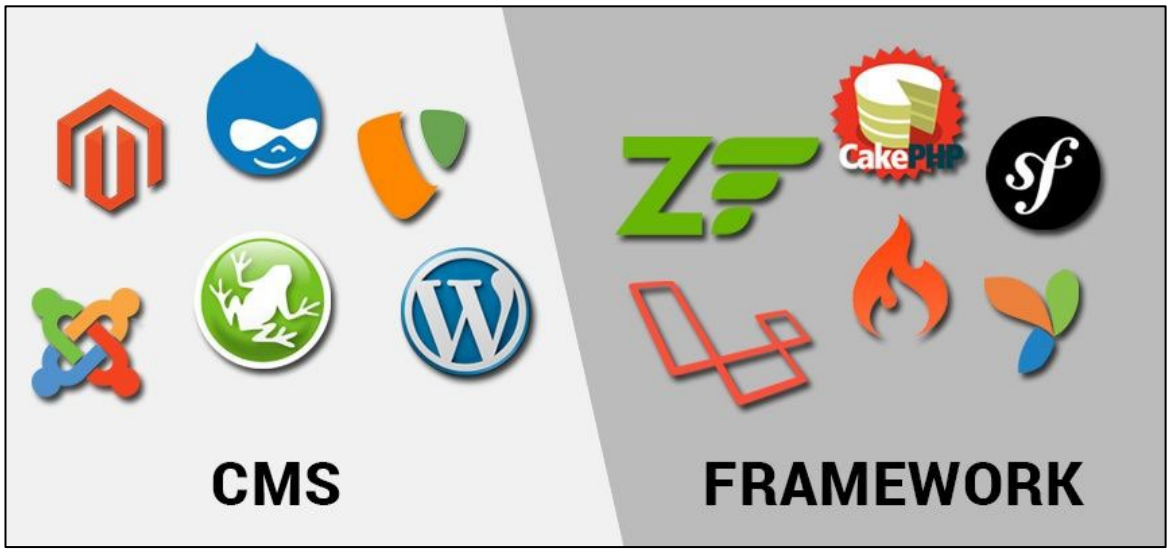



Рисунок 2.2 – Фреймворки та системи керування вмістом базовані на PHP

Плюси PHP:

– Простота вивчення: PHP вважається однією з найпростіших мов для вивчення. Відомий простим та зрозумілим синтаксисом, особливо для тих, хто вже знайомий з HTML.

– Велика спільнота: PHP має велику та активну спільноту розробників, що робить його підтримку та вирішення проблем доступними.

– Кількість ресурсів: Існує велика кількість онлайн-ресурсів, підручників та документації для PHP, що полегшує навчання та розробку.

– Вбудована підтримка баз даних: PHP легко взаємодіє з різними системами управління базами даних, що дозволяє зберігати та оброблювати дані.

Мінуси PHP:

– Специфіка веб-розробки: PHP був створений для веб-розробки, і його використання обмежено в цій галузі. Він не так сильний для загальних програмних завдань.

– Безпека: Необережне програмування на PHP може призвести до вразливостей безпеки, таких як SQL-ін'єкції та вразливості зв'язані з обробкою файлів.

– Інтерпретованість: В порівнянні з деякими іншими мовами, PHP може бути менш продуктивним через свою інтерпретованість [28].

Загалом PHP є багатофункціональною мовою програмування, загальний функціонал якої дозволить виконати всі поставленні задачі під час розроблення інформаційної системи.

### 2.1.5 Середовище реалізації технології аналізу та прогнозування

Для виконання поставлених завдань аналізу та прогнозування було обрано мову програмування Python, але для належного застосування даної мови програмування, необхідно вибрати середовище, що буде відповідати всім вимогам щодо розроблення інформаційної системи.

Найбільш доцільним середовищем для виконанням поставлених, завдань аналізу та прогнозування є платформа Kaggle.

Kaggle - це платформа для проведення конкурсів та спільного співпраці в галузі машинного навчання та аналізу даних. Заснована в 2010 році, Kaggle створена для залучення та об'єднання спеціалістів з усього світу з метою вирішення реальних завдань у сферах науки про дані та штучного інтелекту.

Основні риси та можливості Kaggle включають:

– Конкурси та виклики: Kaggle проводить регулярні конкурси, де учасники можуть змагатися у розв'язанні важких задач машинного навчання та аналізу даних. Це включає в себе завдання від класифікації до прогнозування та візуалізації даних.

– Датасети та ноутбуки: Kaggle надає доступ до широкого спектру даних для аналізу. Користувачі можуть завантажувати, ділитися та використовувати датасети, а також створювати та виконувати Python-базовані ноутбуки для аналізу даних та розробки моделей.

– Спільнота та форуми: Kaggle об'єднує глобальну спільноту професіоналів, дослідників та студентів. Користувачі можуть обговорювати проекти, ділитися досвідом та взаємодіяти на форумах.

– Навчання та ресурси: Платформа надає навчальні матеріали, ресурси та відеокурси для розвитку навичок у галузі машинного навчання та аналізу даних.

– Робота з партнерами: Kaggle співпрацює зі світовими компаніями та організаціями для проведення конкурсів та розв'язання реальних завдань.

Завдяки своїй активній спільноті та різноманіттю завдань, Kaggle став важливим центром для розвитку та співпраці в області аналізу даних та машинного навчання (рис. 2.3)

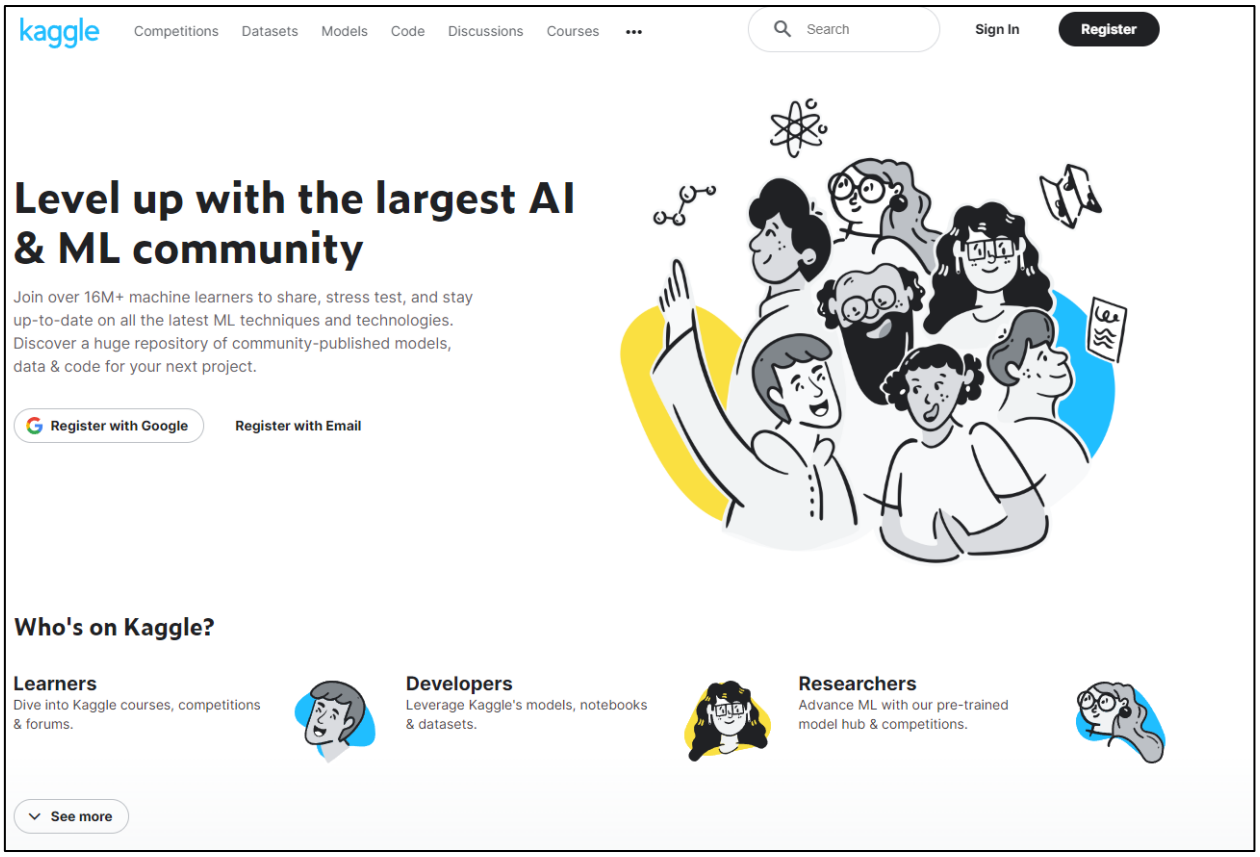



Рисунок 2.3 – Головна сторінка Kaggle

Оскільки дана платформа націлена на дослідження машинного навчання, аналізу даних та задач прогнозування, всі необхідні бібліотеки та потужності для виконання поставлених завдань під час виконання магістерської кваліфікаційної роботи вже реалізовані на даній платформі, тобто її функціоналу та потужності буде цілком достатньо для виконання завдань аналізу та прогнозування даних.

Після виконання завдань аналізу та прогнозування, вихідні дані коду в Kaggle будуть імпортовані в систему для подальшої візуалізації.

Також для завдань графічної візуалізації інформації, а саме побудови графіків в веб-додатку, буде застосовуватись портативне середовище для Python під назвою Py-Script.

PyScript – це фреймворк, який дозволяє користувачам створювати потужні Python-застосунки у веб-браузері, використовуючи інтерфейс HTML на основі Pyodide, MicroPython та WASM, а також сучасні веб-технології (рис. 2.4).

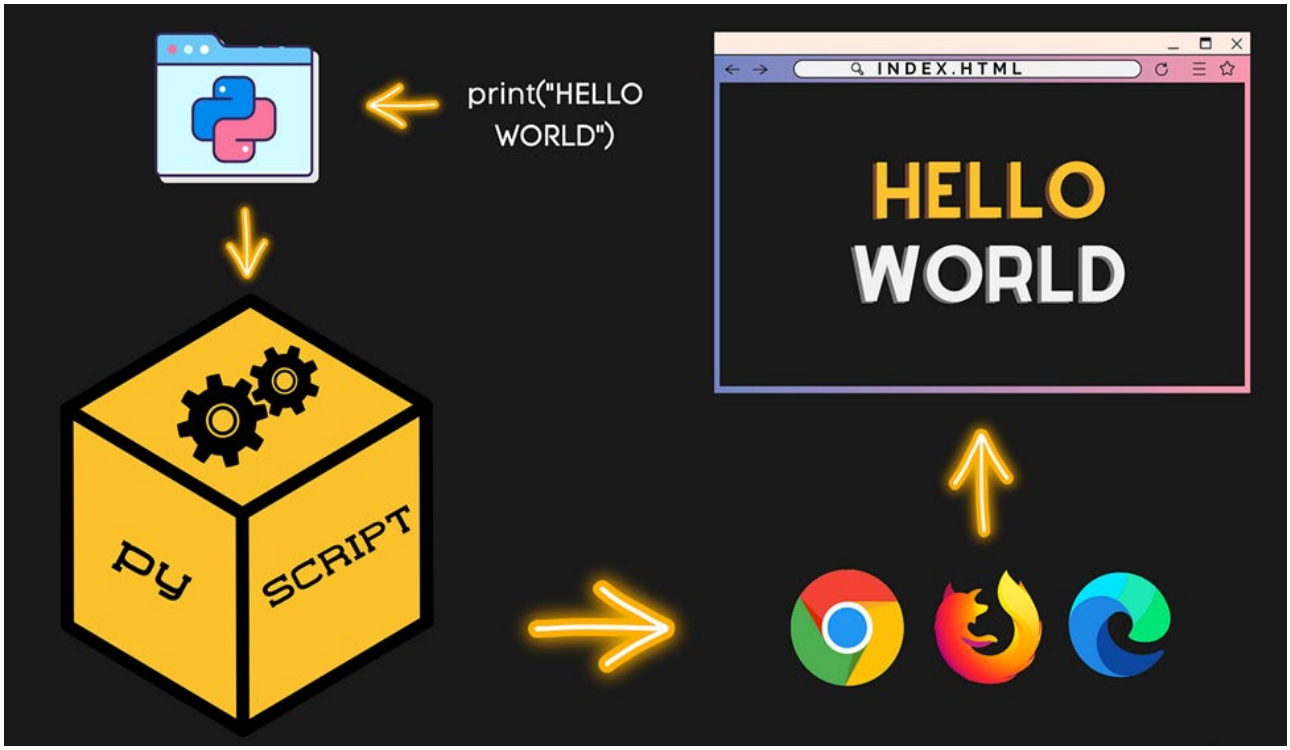



Рисунок 2.4 – Принцип роботи PyScript

Даний фреймворк підтримує велику кількість бібліотек Python, в тому числі numpy, matplotlib та pandas, що допоможе в реалізації детальних графічних матеріалів у веб інтерфейсі [29].

### 2.1.6 Огляд та вибір середовища для реалізації системи у вигляді Web-додатку

Беручи до уваги обрані технології для розроблення інформаційної системи аналізу та прогнозування даних, вибір середовища є одним з найважливіших пунктів, саме від вибору середовище залежить, чи всі поставленні задачі будуть виконані, а обрані технології реалізовані в повну міру.

З урахуванням поставлених завдань та обраних технологій, найкращим середовищем для розроблення інформаційної системи аналізу та прогнозування є сервер, що базується на панелі керування cPanel.

cPanel представляє собою інноваційний веб-орієнтований інтерфейс управління хостингом, спроектований для надання користувачам ефективного та інтуїтивно зрозумілого інструментарію для адміністрування та керування різними аспектами їхніх хостингових облікових записів. Розроблений з урахуванням вимог сучасного веб-середовища, cPanel забезпечує високий рівень функціональності та легкість використання, що робить його невід'ємною складовою сучасної веб-інфраструктури.

Цей інтерфейс пропонує розширені можливості керування файлами, що дозволяє користувачам ефективно взаємодіяти з контентом своїх веб-сайтів. Крім того, cPanel уніфікує процеси управління доменами та піддоменами, дозволяючи адміністраторам з легкістю налаштовувати DNS записи та здійснювати різноманітні налаштування.

Особливий акцент в cPanel робиться на безпеці, що виявляється в доступі до SSL-сертифікатів для ефективного шифрування даних. Крім того, система забезпечує можливість керування версіями PHP, що є важливим для забезпечення сумісності веб-додатків [30].

Одним з внутрішніх інструментів cPanel є phpMyAdmin – інструмент керування базами даних написаний на мові програмування PHP, що базується

на СУБД MySQL та MariaDB, яка по суті є аналогом MySQL. На рисунку 2.5 зображено головну панель інструменту phpMyAdmin.

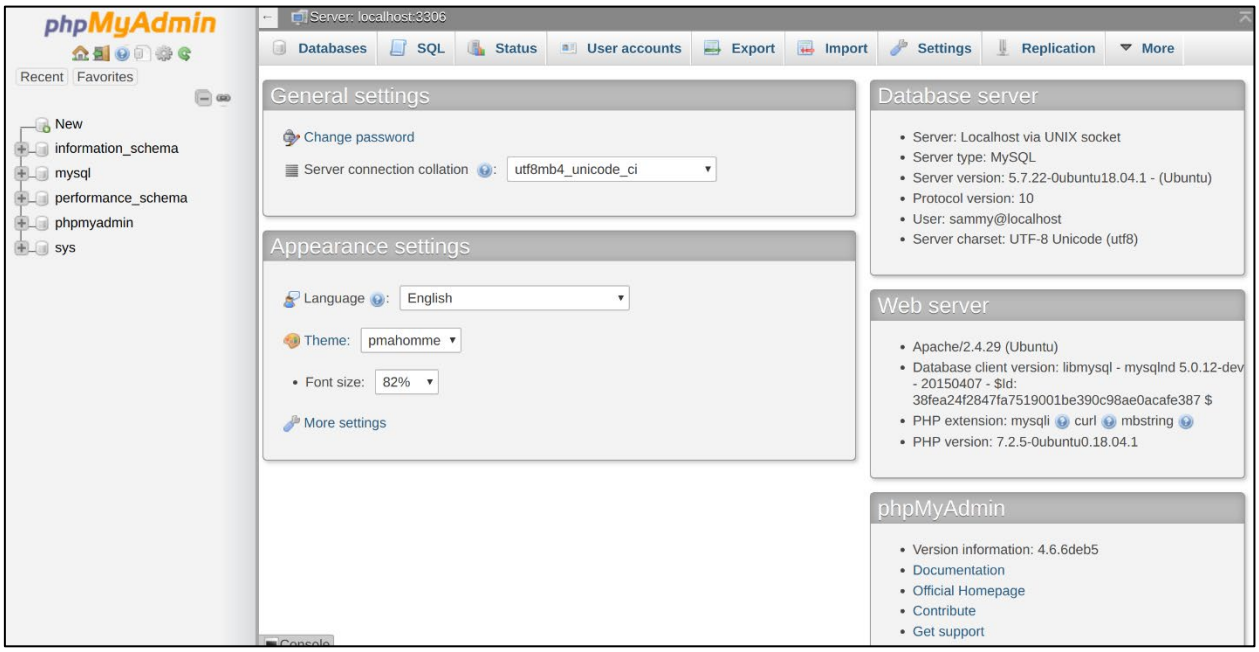



Рисунок 2.5 – Панель управління phpMyAdmin

Завдяки інтеграції із сучасними технологіями, cPanel є ідеальним інструментарієм для ефективного адміністрування веб-серверів та ресурсів, забезпечуючи комплексні можливості управління, що відповідають високим стандартам веб-розробки та адміністрування, а завдяки підтримці MySQL та PHP, дана панель грає вирішальну роль в ході розробки інформаційної системи.

## 2.2 Огляд вхідних даних

Для роботи використовується датасет «River Water Quality EDA and Forecasting». Він складається з двох файлів кожен з яких має низку ознак:

В файлі «PB_All_2000_2021» присутні наступні ознаки:

– «Id» – даний показник являє собі індекс для відношення точок вимірювання показників до іншого файлу «PB_stations» де описані станції вимірювань забруднення поверхневих вод в річці Південний Буг;

– «NH4» – вміст амонію в поверхневих водах;

– «BSK5» – біохімічне споживання кисню;

– «Suspended» – суспензія;

– «O2» – концентрація розчиненого кисню в поверхневих водах;

– «NO3» – вміст нітратів в поверхневих водах;

– «NO2» – вміст нітритів в поверхневих водах;

– «SO4» – вміст сульфатів;

– «PO4» – вміст фосфатів;

– «CL» – вміст хлоридів;

– Наступний файл датасету має назву «PB_stations» та включає в себе наступні ознаки:

– «id» – індекс станції який напряму зв’язаний з «PB_All_2000_2021» та відображає номер станції де було проведено вимірювання;

– «length» – відстань від однієї станції вимірювань до іншої;

– «name» – короткий опис кожної станції, її розташування, відстань до наступної станції.

Частину даних про забруднення річки Південний Буг з відображенням ознак показані на рисунку 2.6.

| id | date | NH4 | BSK5 | Suspended | O2 | NO3 | NO2 | SO4 | PO4 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 17.02.2000 | 0.33 | 2.77 | 12 | 12.3 | 9.5 | 0.057 | 154 | 0.454 |
| 1 | 11.05.2000 | 0.044 | 3 | 51.6 | 14.61 | 17.75 | 0.034 | 352 | 0.09 |
| 1 | 11.09.2000 | 0.032 | 2.1 | 24.5 | 9.87 | 13.8 | 0.173 | 416 | 0.2 |
| 1 | 13.12.2000 | 0.17 | 2.23 | 35.6 | 12.4 | 17.13 | 0.099 | 275.2 | 0.377 |
| 1 | 02.03.2001 | 0 | 3.03 | 48.8 | 14.69 | 10 | 0.065 | 281.6 | 0.134 |
| 1 | 07.06.2001 | 0.02 | 4.02 | 34 | 10.61 | 11.8 | 0.016 | 287 | 0.208 |
| 1 | 10.09.2001 | 0.863 | 3.91 | 147 | 10.96 | 20.5 | 0.284 | 595.2 | 0.674 |

Рисунок 2.6 – Ознаки датасету «River Water Quality EDA and Forecasting»

Іншим джерелом даних для подальших досліджень, є дані Держводагентства, ці дані містять дані вимірів всіх досліджуваних водних об’єктів включаючи Південний Буг, на рисунку 2.7 зображено загальний вигляд даних моніторингу за жовтень 2023 року.

| | | | | | |
|---|---|---|---|---|---|
| 38437 | водосховище Саф'яни, 0 км, с. Саф'яни | UA_M5.3. | Дунай | Лаборато | 45.38672228.890194 |
| 38463 | р. Південний Буг, 393 км, скид м. Ладижин ТОВ "Вінницька птахофабрика" | UA_M5.4_ | Південни | Лаборато | 48.66183329.305000 |
| 38465 | р. Південний Буг, 0 км, с. Мигія, нижче м. Первомайськ , (Олександрівське водосховище), Націо | UA_M5.4_ | Південни | Лаборато | 48.03219430.952167 |
| 38473 | р. Прут, 864 км, с. Шепарівці, питний водозабір м.Коломия, КП "Коломияводоканал" | UA_M5.3. | Дунай | Лаборато | 48.55368324.951406 |
| 27099 | канал Дніпро-Кривий Ріг, 43 км, Південне в-ще, питний в/з | UA_M5.1. | Дніпро | Лаборато | 47.78161133.556506 |
| 27118 | р. Інгулець, 393 км, смт Петрово, Іскрівське вдсх., питний в/з селища | UA_M5.1. | Дніпро | Лаборато | 48.33482533.262072 |
| 27120 | р. Інгулець, 335 км, м. Кривий Ріг, Карачунівське в-ще, питний в/з міста | UA_M5.1. | Дніпро | Лаборато | 47.90764233.283538 |
| 27122 | р. Інгулець, 265 км, с. Андріївка | UA_M5.1. | Дніпро | Лаборато | 47.60314433.212303 |
| 27135 | р. Південний Буг, 652 км, м. Хмільник, питний в/з, вище міста | UA_M5.2_ | Південни | Лаборато | 49.55638927.913056 |
| 27136 | р. Південний Буг, 607 км, с. Гущинці, нижче села , питний водозабір м.Калинівка | UA_M5.4_ | Південни | Лаборато | 49.42111128.382500 |
| 27137 | р. Південний Буг, 582 км, м. Вінниця, Сабарівське вдсх, питний в/з міста, вище міста | UA_M5.4_ | Південни | Лаборато | 49.27305628.471389 |
| 27138 | р. Південний Буг, 570 км, смт Сабарів, (Сутиське водосховище), 500 м нижче скиду ВОКВП ВКП | UA_M5.4_ | Південни | Лаборато | 49.19069428.443417 |
| 27140 | р. Південний Буг, 413 км, с. Маньківка, вище села, питний в/з м.Ладижин | UA_M5.4_ | Південни | Лаборато | 48.73888929.089167 |
| 27146 | р. Південний Буг, 153 км, с. Олексіївка, питний в/з м. Південно-Українськ | UA_M5.4_ | Південни | Лаборато | 47.86878931.119731 |
| 27147 | р. Південний Буг, 136 км, у межах с. Олександрівка , Олександрівське вдсх. | UA_M5.4_ | Південни | Лаборато | 47.70270031.254800 |

Рисунок 2.7 – Вигляд даних моніторингу об’єктів

В кожному файлі звіту моніторингу присутні такі дані:

– «Post_ID» – індекс точки моніторингу;
– «Post_Name» – назва точки моніторингу;
– «Post_Code» – код масиву поверхневих вод;
– «Riverbas_Name» – назва водного басейну;
– «WaterLab_Name» – назва лабораторії дослідження;
– «Latitude» – Широта;
– «Longitude» – Довгота;
– «Controle_Date» – дата взяття проб;
– «Azot» – концентрація азоту;
– «BSK5» – біохімічне споживання кисню;
– «Zavisli» – суспензія;
– «Kisen» – концентрація кисню;
– «Sulfat» – концентрація сульфатів;
– «Hlorid» – концентрація хлориду;
– «Amoniy» – концентрація амонію;

– «Nitrat» – концентрація нітратів;
– «Nitrit» – концентрація нітритів;
– «Fosfat» – концентрація фосфатів;
– «SPAR» – концентрація синтетичних поверхнево-активних речовин;
– «Permanganat» – концентрація перманганату;
– «HSK» – хімічне споживання кисню;
– «Fitoplan» – концентрація фітопланктону;
– «Atrazin» – концентрація атразину;
– «Simazin» – концентрація симазину.

Оскільки кількість даних в наборі даних відрізняються, для подальших досліджень буде проведено початкова обробка даних та відбір суміжних показників.

## 2.3 Першочергова обробка даних. Розвідувальний аналіз

Оскільки для розроблення системи вхідні дані надходять з різних джерел, першочерговим етапом розроблення система буде упорядкування даних та їх об’єднання в один набір даних для зручності використання, для цього скористаємось бібліотеками pandas та glob (рис. 2.8).

```
import pandas as pd
import glob
```

Рисунок 2.8 – Імпортування бібліотек

Дані бібліотеки потрібні для імпорту набору даних, що представляють собою декілька окремих файлів, так замість імпортування кожного файлу окремо, достатньо використати символ підстановки «*» (рис.2.9)

```
# Path to CSV files
csv_files = glob.glob('/kaggle/input/pivdennyi-buh/*.csv')
```

Рисунок 2.9 – Код вибірки файлів даних

Після цього, оголошену змінну можна використати в циклі в поєднанні з бібліотекою pandas (рис.2.10).

```
# Initialize an empty list to store DataFrames
dfs = []

# Specify the columns you want to keep
selected_columns = ['Post_ID', 'Post_Name', 'Controle_Date', 'BSK5', 'Zavisli', 'Kisen', 'Sulfat', 'Hlorid', 'Amoniy', 'Nitrat', '

for csv_file in csv_files:
    # Assuming the CSV files have a header row
    current_data = pd.read_csv(csv_file, sep=';', header=0, usecols=selected_columns)

    # Filter rows where 'Riverbas_Name' is equal to 'Південний Буг'
    current_data = current_data[current_data['Post_Name'].str.startswith('р. Південний Буг')]

    dfs.append(current_data)
```

Рисунок 2.10– цикл для імпорту масивів даних в один датафрейм

Також на даному елементі коду видно вибірку даних, а саме колонок які будуть застосовуватись у подальшому аналізі даних та прогнозуванні, а також виконано фільтрування даних – було обрано лише ті дані, в яких значення поля «Post_Name» починалось з «р. Південний Буг». Далі всі дані було занесено в порожній датафрейм, що був оголошений перед циклом. Розглянемо отриманий датафрейм на рисунку 2.10.

| | Post_ID | Post_Name | Controle_Date | BSK5 | Zavisli | Kisen | Sulfat | Hlorid | Amoniy | Nitrat | Nitrit | Fosfat |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 27135 | р. Південний Буг, 652 км, м. Хмільник, питний в/з, вище міста | 2022-07-07 | 12.30 | 11.0 | 3.30 | 30.09 | 43.60 | 0.890 | 0.94 | 0.110 | 0.350 |
| 1 | 27136 | р. Південний Буг, 607 км, с. Гущинці, нижче села , питний водозабір м.Калинівка | 2022-07-07 | 11.20 | 9.0 | 8.00 | 26.78 | 38.20 | 0.330 | 0.22 | 0.003 | 0.094 |
| 2 | 27137 | р. Південний Буг, 582 км, м. Вінниця, Сабарівське вдсх, питний в/з міста, вище міста | 2022-07-07 | 10.30 | 12.0 | 3.30 | 27.85 | 36.50 | 0.340 | 0.12 | 0.005 | 0.088 |
| 3 | 27138 | р. Південний Буг, 570 км, смт Сабарів, (Сутиське водосховище), 500 м нижче скиду ВОКВП ВКП "Вінницяводоканал" | 2022-07-05 | NaN | NaN | 2.80 | NaN | NaN | NaN | NaN | NaN | NaN |
| 4 | 27140 | р. Південний Буг, 413 км, с. Маньківка, вище села, питний в/з м.Ладижин | 2022-07-05 | 10.00 | 8.0 | 14.40 | 41.53 | 28.70 | 0.130 | 0.11 | 0.002 | 0.043 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 127 | 27138 | р. Південний Буг, 570 км, смт Сабарів, (Сутиське водосховище), 500 м нижче скиду ВОКВП ВКП "Вінницяводоканал" | 2023-08-08 | NaN | NaN | 8.90 | NaN | NaN | NaN | NaN | NaN | NaN |
| 128 | 27140 | р. Південний Буг, 413 км, с. Маньківка, вище села, питний в/з м.Ладижин | 2023-08-09 | 5.00 | 8.0 | 8.00 | 31.90 | 33.30 | 0.410 | 0.76 | 0.039 | 0.510 |
| 129 | 27146 | р. Південний Буг, 153 км, с. Олексіївка, питний в/з м. Південно-Українськ | 2023-08-07 | 1.36 | 9.0 | 7.09 | 71.68 | 60.97 | 0.006 | 0.76 | 0.009 | 0.594 |
| 130 | 27147 | р. Південний Буг, 136 км, у межах с. Олександрівка , Олександрівське вдсх. | 2023-08-07 | NaN | NaN | 5.51 | NaN | NaN | NaN | NaN | NaN | NaN |
| 131 | 27148 | р. Південний Буг, 97 км, м. Вознесенськ, пит.в/з м. Вознесенськ, 2 км до в'їзду у м. Вознесенськ по трасі з м. Миколаїв | 2023-08-07 | 2.47 | 12.4 | 8.06 | 275.20 | 113.44 | 0.036 | 0.00 | 0.001 | 0.462 |

Рисунок 2.11 – Отриманий датафрейм

Як можна бачити було сформовано саме дані тих об’єктів, що відносяться до теми нашої роботи, але структура даних поки що є незадовільною – поля датафрейму даних за 2021 – 2023 роки, що ми сформували сильно відрізняється від даних представлених за 2000 – 2021 роки, тому нам потрібно уподібнити всі поля з метою подальшого об’єднання створеного датафрейму з датасетом «River Water Quality EDA and Forecasting». Для цього імпортуємо сам датасет в наш нотбук а також перейменуємо поля першого датафрейму (рис. 2.12).

```
other_data = pd.read_csv('../input/wq-southern-bug-river-01052021/PB_All_2000_2021.csv', sep=';', header=0)

# Rename columns in combined_data
combined_data.rename(columns={
    'Post_ID': 'id',
    'Controle_Date': 'date',
    'Amoniy': 'NH4',
    'BSK5': 'BSK5',
    'Zavisli': 'Suspended',
    'Kisen': 'O2',
    'Nitrat': 'NO3',
    'Fosfat': 'PO4',
    'Hlorid': 'CL',
    'Nitrit': 'NO2',
    'Sulfat': 'SO4'
}, inplace=True)
```

Рисунок 2.12 – Імпорт даних датасету «River Water Quality EDA and Forecasting» та перейменування полів створеного датафрейму

Поля було перейменовано на користь датасету з даними попередніх років, адже велика перевага в кількості наявних даних грає ключеву роль в даному питані і тому буде простіше відкорегувати менший датафрейм.

Наступним кроком буде зміна порядку полів в датафреймі з даними 2021 – 2023 років, а також видалення поля «Post_Name» з датафрейму, адже основною його задачею було відкидання непотрібних даних, що було виконано. Також для подальшого об’єднання датафреймів потрібно змінити формат дати в новому датафреймі, адже як видно з фрагментів даних зображених на рисунках 2.7 та 2.11, формат дати в обох датафреймах відрізняється, що стане перешкодою в подальшому об’єднанні даних, аналізі

та прогнозуванні. На рисунках 2.13 та 2.14 можна побачити код реалізації описаних дій а також отриманий датафрейм.

```
# Drop 'Riverbas_Name' column
combined_data.drop(columns=['Post_Name'], inplace=True)

# Reorder columns in combined_data
combined_data = combined_data[['id', 'date', 'NH4', 'BSK5', 'Suspended', 'O2', 'NO3', 'NO2', 'SO4', 'PO4', 'CL']]
combined_data['date'] = pd.to_datetime(combined_data['date'], errors='coerce').dt.strftime('%d.%m.%Y')

# Concatenate combined_data and other_data
final_dataframe = pd.concat([combined_data, other_data], ignore_index=True)
```

Рисунок 2.13 – Корегування даних датафрейму та об’єднання даних

| | id | date | NH4 | BSK5 | Suspended | O2 | NO3 | NO2 | SO4 | PO4 | CL |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 27135 | 07.07.2022 | 0.890 | 12.30 | 11.0 | 3.30 | 0.94 | 0.110 | 30.09 | 0.350 | 43.60 |
| **1** | 27136 | 07.07.2022 | 0.330 | 11.20 | 9.0 | 8.00 | 0.22 | 0.003 | 26.78 | 0.094 | 38.20 |
| **2** | 27137 | 07.07.2022 | 0.340 | 10.30 | 12.0 | 3.30 | 0.12 | 0.005 | 27.85 | 0.088 | 36.50 |
| **3** | 27138 | 05.07.2022 | NaN | NaN | NaN | 2.80 | NaN | NaN | NaN | NaN | NaN |
| **4** | 27140 | 05.07.2022 | 0.130 | 10.00 | 8.0 | 14.40 | 0.11 | 0.002 | 41.53 | 0.043 | 28.70 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **2988** | 22 | 06.10.2020 | 0.046 | 2.69 | 3.6 | 8.28 | 3.80 | 0.038 | 160.00 | 0.726 | 77.85 |
| **2989** | 22 | 27.10.2020 | 0.000 | 1.52 | 0.5 | 11.26 | 0.56 | 0.031 | 147.20 | 0.634 | 71.95 |
| **2990** | 22 | 03.12.2020 | 0.034 | 0.29 | 0.8 | 11.09 | 2.58 | 0.042 | 209.92 | 0.484 | 61.17 |
| **2991** | 22 | 12.01.2021 | 0.000 | 2.10 | 0.0 | 14.31 | 3.94 | 0.034 | 121.60 | 0.424 | 63.49 |
| **2992** | 22 | 10.02.2021 | 0.000 | 1.78 | 0.0 | 14.30 | 6.30 | 0.033 | 134.40 | 0.582 | 66.31 |

Рисунок 2.14 – Отриманий датафрейм

Як бачимо з отриманого датафрейму, в ньому наявні аномальні значення «id», що собою позначають точку моніторингу, пов'язано це з тим, що в наборі даних Держводагентства, індекси об’єктів спостережень річки Південний Буг відрізняються, для корегування даних індексів було взято масив точок спостережень, з одного датафрейму, та було змінено на значення датасету «River Water Quality EDA and Forecasting», а саме значення «id» з файлу

«PB_stations». Процес заміни та результати зображено на рисунках 2.15 та 2.16.

```
id_mapping = {
    27610: 5,
    27135: 16,
    27136: 15,
    27137: 14,
    27140: 10,
    27146: 3,
    27148: 22,
    38463: 9,
    27147: 2,
    27138: 13
}

final_dataframe['id'] = final_dataframe['id'].replace(id_mapping)
```

Рисунок 2.15 – код коригування індексів

| | id | date | NH4 | BSK5 | Suspended | O2 | NO3 | NO2 | SO4 | PO4 | CL |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **1560** | 14 | 2000-01-10 | 1.30 | 1.9 | 9.0 | 15.8 | 6.300 | 0.30 | 40.1 | 1.61 | 31.3 |
| **1263** | 12 | 2000-01-10 | 1.50 | 1.5 | 9.0 | 15.2 | 7.400 | 0.32 | 37.6 | 0.06 | 36.7 |
| **1819** | 15 | 2000-01-10 | 1.10 | 2.1 | 22.0 | 16.1 | 7.000 | 0.28 | 31.9 | 0.01 | 28.6 |
| **2077** | 16 | 2000-01-10 | 1.20 | 2.6 | 8.0 | 16.5 | 6.700 | 0.30 | 34.3 | 0.02 | 29.5 |
| **952** | 10 | 2000-01-11 | 2.50 | 2.6 | 7.0 | 8.7 | 7.700 | 0.15 | 40.9 | 0.04 | 34.0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **125** | 15 | 2023-08-10 | 2.44 | 5.2 | 12.0 | 3.4 | 1.060 | 0.18 | 34.3 | 1.41 | 40.1 |
| **26** | 16 | 2023-09-05 | 0.50 | 7.0 | 10.0 | 5.1 | 0.110 | 0.10 | 30.8 | 0.21 | 33.3 |
| **27** | 15 | 2023-09-05 | 0.75 | 6.0 | 13.0 | 5.1 | 0.094 | 0.06 | 32.9 | 0.16 | 40.1 |
| **28** | 14 | 2023-09-05 | 0.45 | 8.0 | 4.0 | 8.6 | 0.066 | 0.02 | 31.6 | 0.11 | 38.3 |
| **29** | 10 | 2023-09-06 | 0.47 | 11.3 | 6.0 | 2.5 | 0.048 | 0.09 | 33.5 | 0.20 | 40.1 |

Рисунок 2.16 – Відображення датафрейму

Після отримання повноцінного датафрейму, проведемо розвідувальний аналіз даних для чого нам потрібно розглянути кількість даних на кожній з станцій (рис. 2.17).

```
final_dataframe['id'].value_counts().sort_values().plot(kind='barh')
```

```
<Axes: ylabel='id'>
```

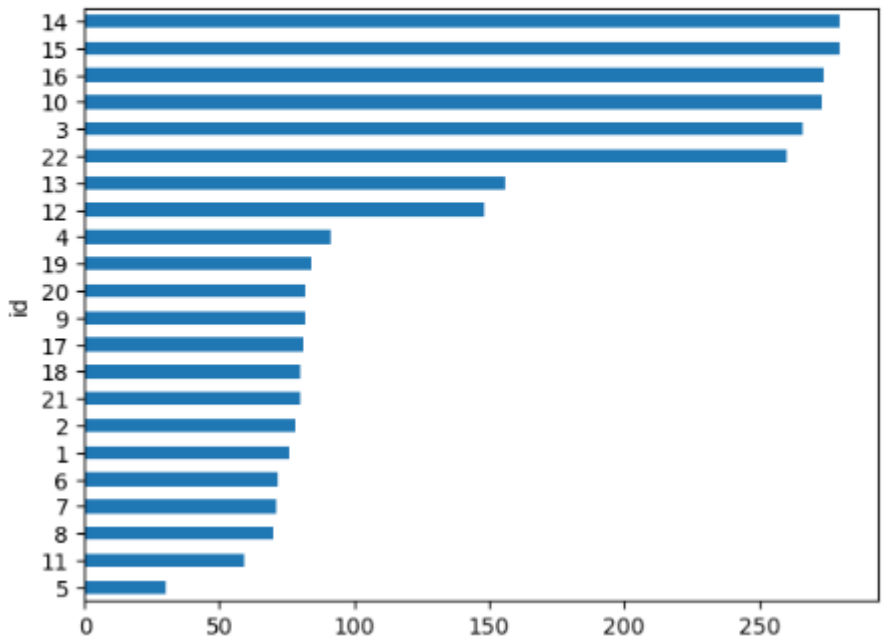



Рисунок 2.17 – Кількість даних на кожній з точок спостережень

Як бачимо найбільше даних наявно саме для станцій 14, 15 та 16, тому потрібно розглянути дані станції (рис. 2.18).

```
stations_good = [14, 15, 16]
data_about[data_about['id'].isin(stations_good)]
```

| | id | length | name_station |
|---|---|---|---|
| 13 | 14 | 582.0 | р. Південний Буг, 582 км, м. Вінниця, Сабарівське вдсх, питний в/з міста, вище міста |
| 14 | 15 | 607.0 | р. Південний Буг, 607 км, с. Гущинці, нижче села , питний водозабір м.Калинівка |
| 15 | 16 | 652.0 | р. Південний Буг, 652 км, м. Хмільник, питний в/з, вище міста |

Рисунок 2.18 – Станції з індексом 14, 15 та 16

Також розглянемо роки дослідження даних, а саме початок та кінець досліджень, це дозволить нам, дізнатись чи актуальні дані в обраних станцій. Початок та кінець дослідження зображено на рисунках 2.18 та 2.19.

```
final_dataframe[['id', 'year']].groupby(by=['id']).min().sort_values(by=['year'], ascending=False)
```

| id | year |
|---|---|
| 5 | 2019 |
| 13 | 2006 |
| 1 | 2000 |
| 21 | 2000 |
| 20 | 2000 |
| 19 | 2000 |
| 18 | 2000 |
| 17 | 2000 |
| 16 | 2000 |
| 15 | 2000 |
| 14 | 2000 |
| 12 | 2000 |
| 2 | 2000 |
| 11 | 2000 |
| 10 | 2000 |
| 9 | 2000 |
| 8 | 2000 |
| 7 | 2000 |
| 6 | 2000 |
| 4 | 2000 |
| 3 | 2000 |
| 22 | 2000 |

Рисунок 2.19 – Датування перших даних моніторингу на різних станціях

```
final_dataframe[['id', 'year']].groupby(by=['id']).max().sort_values(by=['year'], ascending=False)
```

| id | year |
|---|---|
| 22 | 2023 |
| 3 | 2023 |
| 5 | 2023 |
| 6 | 2023 |
| 9 | 2023 |
| 10 | 2023 |
| 2 | 2023 |
| 13 | 2023 |
| 14 | 2023 |
| 15 | 2023 |
| 16 | 2023 |
| 17 | 2018 |
| 21 | 2018 |
| 20 | 2018 |
| 19 | 2018 |
| 18 | 2018 |
| 1 | 2018 |
| 11 | 2018 |
| 8 | 2018 |
| 7 | 2018 |
| 4 | 2018 |
| 12 | 2018 |

Рисунок 2.20 – Датування останніх даних моніторингу на різних станціях

Як бачимо, для обраних станцій 14, 15 та 16 присутні дані 2023 року, тобто дані є актуальними. Наступним кроком є обрання, ключових показників для утворення датафрейму та його розбиття на тестові та тренувальні дані, було обрано біохімічне споживання кисню, вміст амонію та розчиненого кисню (рис. 2.21).

```
# Set target indicator
target_data_name = 'O2'
#feature_target_all = ['NH4', 'BSK5', 'NO3', 'NO2', 'SO4', 'PO4', 'CL']
feature_target_all = ['BSK5', 'NH4']
feature_data_all = feature_target_all + [target_data_name]
feature_data_all
```

Рисунок 2.21 – вибір цільових показників для складання тренувальних та тестових даних

Далі потрібно сформувати датафрейм що буде складатись лише з даних вмісту кисню, амонію та БКС5 на обраних станціях (рис.2.22).

```
# Data sampling only for good stations
df_indicator = final_dataframe[['id', 'ds'] + feature_data_all]
df_indicator = df_indicator[df_indicator['id'].isin(stations_good)].dropna().reset_index(drop=True)
df_indicator
```

| | id | ds | BSK5 | NH4 | O2 |
|---|---|---|---|---|---|
| 0 | 14 | 2000-01-10 | 1.9 | 1.30 | 15.8 |
| 1 | 15 | 2000-01-10 | 2.1 | 1.10 | 16.1 |
| 2 | 16 | 2000-01-10 | 2.6 | 1.20 | 16.5 |
| 3 | 15 | 2000-02-01 | 2.7 | 2.20 | 11.9 |
| 4 | 14 | 2000-02-01 | 2.5 | 2.20 | 10.6 |
| ... | ... | ... | ... | ... | ... |
| 826 | 14 | 2023-08-10 | 6.3 | 0.24 | 2.5 |
| 827 | 15 | 2023-08-10 | 5.2 | 2.44 | 3.4 |
| 828 | 16 | 2023-09-05 | 7.0 | 0.50 | 5.1 |
| 829 | 15 | 2023-09-05 | 6.0 | 0.75 | 5.1 |
| 830 | 14 | 2023-09-05 | 8.0 | 0.45 | 8.6 |

831 rows × 5 columns

Рисунок 2.22 – Складання датафрейму

Далі потрібно провести сформувати датафрейму для подальшого розбиття на тестові та тренувальні дані (рис.2.23).

```
df = pd.pivot_table(df_indicator, index=["ds"], columns=["id"], values=feature_data_all).dropna()
df.columns = cols
df
```

| ds | 14_BSK5 | 14_NH4 | 14_O2 | 15_BSK5 | 15_NH4 | 15_O2 | 16_BSK5 | 16_NH4 | 16_O2 |
|---|---|---|---|---|---|---|---|---|---|
| 2000-01-10 | 1.9 | 2.1 | 2.6 | 1.300 | 1.10 | 1.20 | 15.8 | 16.1 | 16.5 |
| 2000-02-01 | 2.5 | 2.7 | 3.1 | 2.200 | 2.20 | 2.40 | 10.6 | 11.9 | 10.3 |
| 2000-03-01 | 4.1 | 4.4 | 6.0 | 0.680 | 0.87 | 0.54 | 10.3 | 10.3 | 11.9 |
| 2000-04-04 | 3.4 | 3.1 | 3.4 | 0.810 | 1.22 | 0.51 | 8.7 | 8.7 | 8.1 |
| 2000-05-16 | 3.0 | 3.9 | 4.3 | 0.270 | 0.34 | 0.37 | 6.5 | 6.5 | 6.8 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 2022-07-07 | 10.3 | 11.2 | 12.3 | 0.340 | 0.33 | 0.89 | 3.3 | 8.0 | 3.3 |
| 2022-12-07 | 5.6 | 4.6 | 5.2 | 0.160 | 0.19 | 0.15 | 6.2 | 4.9 | 6.2 |
| 2023-07-06 | 7.4 | 6.8 | 7.6 | 0.595 | 0.40 | 0.30 | 4.0 | 4.6 | 4.6 |
| 2023-08-10 | 6.3 | 5.2 | 7.3 | 0.240 | 2.44 | 0.70 | 2.5 | 3.4 | 6.8 |
| 2023-09-05 | 8.0 | 6.0 | 7.0 | 0.450 | 0.75 | 0.50 | 8.6 | 5.1 | 5.1 |

Рисунок 2.23 – Сформований датафрейм

Візуалізуємо дані сформованого датафрейму (рис.2.24).

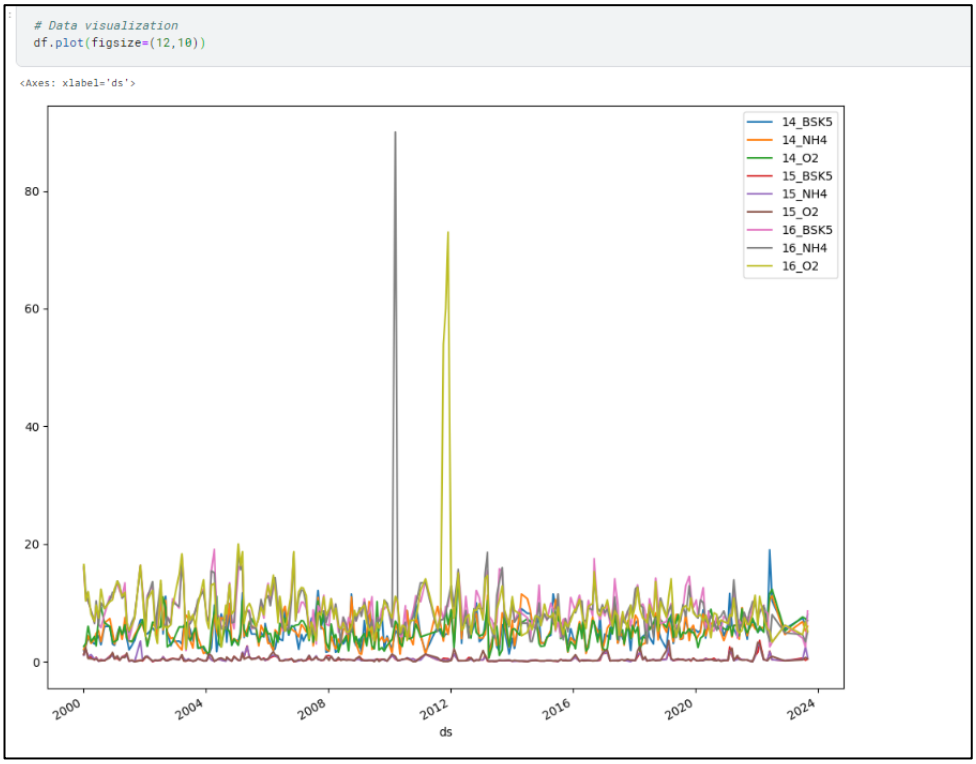



Рисунок 2.24 – Візуалізація даних

Після подальшого розвідувального аналізу розбиваємо дані на тестові та тренувальні (рис.2.25).

```
# Dividing data into training and test
train, test, target, target_test = train_test_split(df, target_data, test_siz
e=0.4, random_state=0)
print(train.shape, test.shape)

(97, 8) (66, 8)
```

Рисунок 2.25 – Розбиття даних на тестові та тренувальні

Отримані датафрейми будуть використовуватись для тестування моделей та подальшої оцінки точності кожної моделі.

## 2.4 Вибір моделей

Для початку імпортуємо необхідні для роботи бібліотеки, ключовою з яких є «sklearn.metrics» яка буде використовуватись для розрахунку точності кожної моделі, а більш точніше його функція «r2_score» (рис. 2.26).

```
# Modeling
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
import xgboost as xgb
from xgboost.sklearn import XGBRegressor
from sklearn.neighbors import KNeighborsRegressor
import lightgbm as lgb

# Metrics
from sklearn.metrics import r2_score
```

Рисунок 2.26 – Імпорт бібліотек

Наступним кроком буде створення датафрейму з результатами точності тої чи іншої моделі на тренувальних та валідаційних даних (рис. 2.27).

```
2]:
# Creation the dataframe with the resulting score of all models
result = pd.DataFrame({'model' : ['Linear Regression', 'Random Forest Regress
or', 'XGBoost Regressor', 'KNeighbors Regressor', 'LightGBM Regressor'],
                       'train_score': 0, 'valid_score': 0})
result
```

2]:

| | model | train_score | valid_score |
|---|---|---|---|
| 0 | Linear Regression | 0 | 0 |
| 1 | Random Forest Regressor | 0 | 0 |
| 2 | XGBoost Regressor | 0 | 0 |
| 3 | KNeighbors Regressor | 0 | 0 |
| 4 | LightGBM Regressor | 0 | 0 |

Рисунок 2.27 – Створення датафрейму

Після створення датафрейму потрібно реалізувати моделі для їх тестування на тестових та тренувальних даних, а також порівняння їх точності за допомогою «r2_score».

KNeighborsRegressor є алгоритмом навчання з учителем в галузі машинного навчання, відомим як метод найближчих сусідів (k-NN) для вирішення задач регресії. Основна ідея полягає в тому, щоб призначити значення новому екземпляру даних на основі середнього (або іншого міркування) значень його k найближчих сусідів у просторі ознак (рис.2.28).

```
]:
knn = KNeighborsRegressor()
param_grid_knn = {'n_neighbors': [3, 5, 7, 9], 'weights': ['uniform', 'distan
ce'], 'algorithm': ['auto', 'ball_tree', 'kd_tree', 'brute']}
```

Рисунок 2.28 – Побудова моделі KNeighbourRegressor

LightGBM (Light Gradient Boosting Machine) представляє собою потужну модель градієнтного бустингу, розроблену для ефективного розв'язання завдань класифікації та регресії в області машинного навчання. Вона визначається своєю високою швидкодією та можливістю працювати з великими обсягами даних. Однією з ключових особливостей LightGBM є використання гістограм для зберігання та обчислення градієнтів під час тренування моделі. Це дозволяє оптимізувати обчислення та прискорює процес навчання. Дану модель побудовано на рисунку 2.29.

```
:
# LightGBM Regressor
lgbm = lgb.LGBMRegressor()
param_grid_lgbm = {'n_estimators': [50, 100], 'learning_rate': [0.05, 0.1],
'max_depth': [4, 5], 'reg_lambda': [0.3]}
```

Рисунок 2.29 – Побудова моделі LightGBM

Лінійна регресія є однією з фундаментальних моделей в статистиці та машинному навчанні. Ця модель використовує лінійну функцію для моделювання залежностей між однією або кількома незалежними змінними (пояснювальними ознаками) та залежною змінною (цільовою ознакою) (рис.2.30).

```
[5]:
# Linear Regression
lr = LinearRegression()
lr.fit(train, target_train)
```

Рисунок 2.30 – Побудова моделі LinearRegression

RandomForestRegressor є моделлю машинного навчання, яка використовує метод випадкового лісу для регресійного аналізу. Цей алгоритм є частиною сімейства ансамблевих методів та використовує декілька дерев рішень для досягнення більш точних та стійких прогнозів (рис.2.31).

```
]:
%%time
# Random Forest Regressor
rf = RandomForestRegressor()
param_grid = {'n_estimators': [50, 100], 'min_samples_leaf': [i for i in rang
e(3,7)],
              'max_features': ['auto'], 'max_depth': [i for i in range(3,6)],
              'criterion': ['mse'], 'bootstrap': [False]}
```

Рисунок 2.31 – Побудова моделі RandomForestRegressor

XGBoost (eXtreme Gradient Boosting) - це потужний алгоритм машинного навчання, який відноситься до градієнтного бустінгу. Його особливість полягає в тому, що він ефективно комбінує низку слабких моделей (зазвичай дерев рішень) для створення сильного ансамблю (рис. 2.32)

```
%%time
# XGBoost Regressor
xgbr = xgb.XGBRegressor()
# parameters = {'n_estimators': [60, 70, 80, 90, 95, 100, 105, 110, 120, 130, 1
40],
#               'learning_rate': [0.005, 0.01, 0.05, 0.075, 0.1],
#               'max_depth': [3, 5, 7, 9],
#               'reg_lambda': [0.1, 0.3, 0.5]}

parameters = {'n_estimators': [50, 100],
              'learning_rate': [0.05, 0.01],
              'max_depth': [4, 5],
              'reg_lambda': [0.3]}
```

Рисунок 2.32 – Побудова моделі XGBoost

На основі побудованих моделей буде проводитись прогнозування та визначення найбільш точної моделі серед побудованих.

## 2.6 Висновки

У даному розділі було розглянути основні технології розроблення інформаційної системи а також обрано найбільш доцільні для виконання поставлених завдань, було обрано наступні технології для розроблення інформаційної системи:

– Технологія зберігання даних: MySQL;

– ГІС технологія: OpenStreetMap;

– Технологія аналізу та прогнозування даних: Python та відповідні бібліотеки, а також середовище розробки;

– Технологія реалізації системи у вигляді веб-додатку: PHP.

Також було проведено огляд, формування та розвідувальний аналіз вхідних даних. В кінці розділу було описано основні моделі що будуть застосуватись під час розроблення системи.

# 3 РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ В РІЧЦІ ПІВДЕННИЙ БУГ

## 3.1 Розроблення інформаційної технології аналізу та прогнозування для інформаційної системи

Для початку побудуємо алгоритм роботи аналізу та прогнозування якості води. (рис. 3.1).

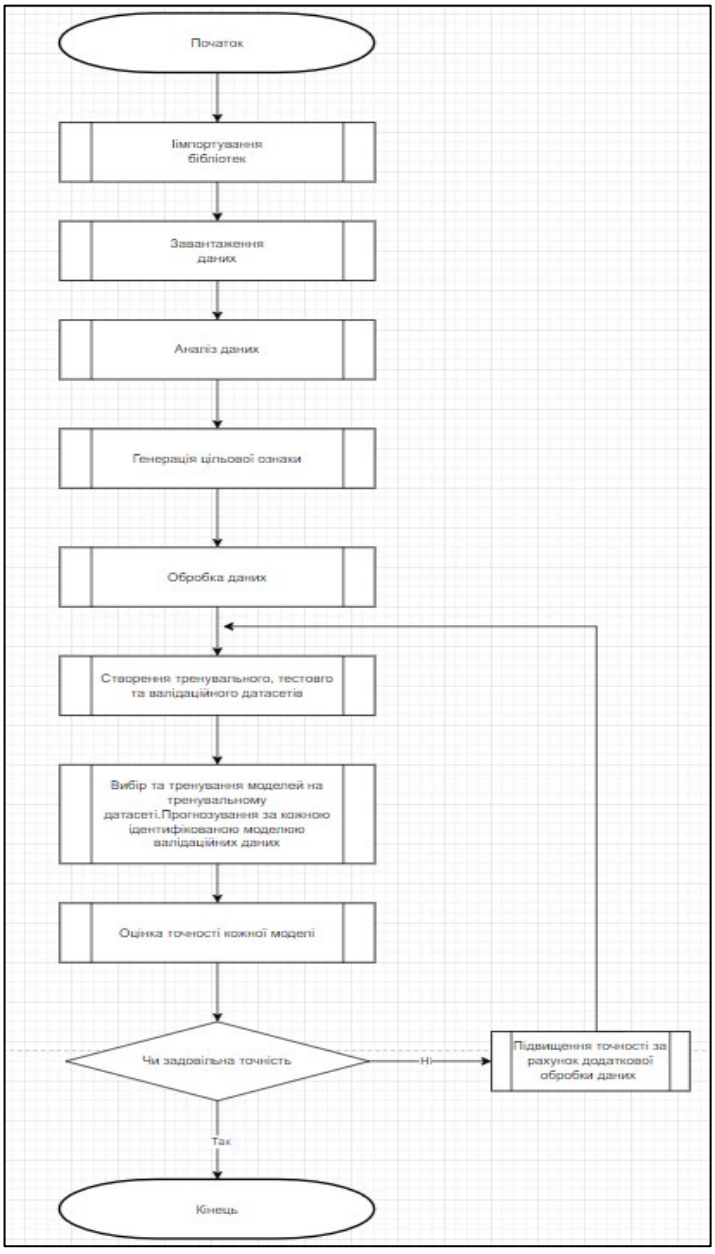



Рисунок 3.1 – Алгоритм роботи програми

Згідно даного алгоритму необхідно виконати наступні завдання:

1. Здійснити аналіз даних.
2. Виконати синтез ознак.
3. Генерування цільової ознаки.
4. Попереднє оброблення даних з використанням методу «sklearn.preprocessing.StandardScaler»
5. Для різних типів моделей створити тренувальний, валідаційний і тестовий датесет.
6. Здійснення вибору і тренування моделей на «train» датасеті і за кожною ідентифікованою моделлю «valid» даних виконання прогнозу.
7. Вибір найоптимальнішої моделі за різними метриками.
8. За розширеним датасетом, до якого входить валідаційний і тренувальний, здійснити тренування оптимальної моделі. Здійснити прогнозування даних.
9. На основі прогнозу тестових даних, потрібно здійснити оцінку точності прогнозування.

Після побудови алгоритму було розпочато його виконання. Оскільки імпорт необхідних бібліотек було здійснено в 2 пункті, наступним кроком буде тренування моделей та визначення найбільш точної моделі, так на рисунку 3.2 зображено код тренування моделі KNeighbourRegressor.

```
# Training model
knn_CV = GridSearchCV(knn, param_grid=param_grid_knn, cv=cv_train, verbose=False)
knn_CV.fit(train, target_train)
print(knn_CV.best_params_)
```

```
{'algorithm': 'auto', 'n_neighbors': 7, 'weights': 'distance'}
```

Рисунок 3.2 – Тренування моделі KNeighbourRegressor

Далі було визначено точність моделі з використанням «r2_score». Точність зображено на рисунку 3.3.


Рисунок 3.3 – Визначення точності моделі KNeighbourRegressor

Точність моделі на тренувальних даних та валідаційних даних склала 1.0 та 0.59 відповідно.

Наступною моделлю для тренування є модель LightGBM, для тренування було обрано наступні параметри: learning_rate 0.1; max_depth 4; n_estimators 50; reg_lambda 0.3. Тренування зображено на рисунку 3.4.

```
# Training model
lgbm_CV = GridSearchCV(lgbm, param_grid=param_grid_lgbm, cv=cv_train, verbose=False)
lgbm_CV.fit(train, target_train)
print(lgbm_CV.best_params_)
```

Рисунок 3.4 – Тренування моделі LightGBM

Після тренування моделі, застосуємо її до тренувальних даних та порівняємо з актуальними даними, тим самим визначивши r2_score (рис. 3.5).

```
# Prediction for training data
y_train_lgbm = lgbm_CV.predict(train)
```

```
# Accuracy of model
r2_score_acc = round(r2_score(target_train, y_train_lgbm), 2)
print(f'Accuracy of LightGBM Regressor model training is {r2_score_acc}')
```

```
Accuracy of LightGBM Regressor model training is 0.66
```

```
# Save to result dataframe
result.loc[result['model'] == 'LightGBM Regressor', 'train_score'] = r2_score_acc
```

```
# rounded r2_score_acc to 2 decimal values after the text
y_val_lgbm = lgbm_CV.predict(valid)
r2_score_acc_valid = round(r2_score(target_valid, y_val_lgbm), 2)
result.loc[result['model'] == 'LightGBM Regressor', 'valid_score'] = r2_score_acc_valid
print(f'Accuracy of LightGBM Regressor model prediction for valid dataset is {r2_score_acc_valid}')
```

```
Accuracy of LightGBM Regressor model prediction for valid dataset is 0.58
```

Рисунок 3.5 – Визначення точності моделі LightGBM

Точність моделі на тренувальних даних та валідаційних даних склала 0.66 та 0.58 відповідно.

Наступною моделлю є модель лінійної регресії, тренування моделі на тренувальних даних а також розрахування точності зображено на рисунку 3.6.

```
# Linear Regression
lr = LinearRegression()
lr.fit(train, target_train)

# Prediction for training data
y_train_lr = lr.predict(train)

# Accuracy of model
r2_score_acc = round(r2_score(target_train, y_train_lr), 2)
print(f'Accuracy of Linear Regression model training is {r2_score_acc}')

# Save to result dataframe
result.loc[result['model'] == 'Linear Regression', 'train_score'] = r2_score_acc
```

```
Accuracy of Linear Regression model training is 0.65
```

```
# Print rounded r2_lr = lr.predict(valid)
y_val_lr = lr.predict(valid)
r2_score_acc_valid = round(r2_score(target_valid, y_val_lr),2)
result.loc[result['model'] == 'Linear Regression', 'valid_score'] = r2_score_acc_valid
print(f'Accuracy of Linear Regression model prediction for valid dataset is {r2_score_acc_valid}')
```

```
Accuracy of Linear Regression model prediction for valid dataset is 0.76
```

Рисунок 3.6 – Тренування моделі Linear Regression, визначення точності

Точність моделі на тренувальних та валідаційних даних склала 0.65 та 0.76 відповідно.

Наступна модель – модель RandomForestRegressor, результати тренування моделі та визначенні точності на тестових і валідаційних даних зображено на рисунку 3.7.

```
%%time
# Random Forest Regressor
rf = RandomForestRegressor()
param_grid = {'n_estimators': [50, 100], 'min_samples_leaf': [i for i in range(3,7)],
              'max_features': ['auto'], 'max_depth': [i for i in range(3,6)],
              'criterion': ['mse'], 'bootstrap': [False]}

# Training model
rf_CV = GridSearchCV(rf, param_grid=param_grid, cv=cv_train, verbose=False)
rf_CV.fit(train, target_train)
print(rf_CV.best_params_)

# Prediction for training data
y_train_rf = rf_CV.predict(train)

# Accuracy of model
r2_score_acc = round(r2_score(target_train, y_train_rf),2)
print(f'Accuracy of RandomForestRegressor model training is {r2_score_acc}')

# Save to result dataframe
result.loc[result['model'] == 'Random Forest Regressor', 'train_score'] = r2_score_acc
```

```
{'bootstrap': False, 'criterion': 'mse', 'max_depth': 3, 'max_features': 'auto', 'min_samples_leaf': 4, 'n_estimators': 50}
Accuracy of RandomForestRegressor model training is 0.7
CPU times: user 13.2 s, sys: 69.1 ms, total: 13.2 s
Wall time: 13.2 s
```

+ Code + Markdown

```
# Print rounded r2_score_acc to 2 decimal values after the text
y_val_rf = rf_CV.predict(valid)
r2_score_acc_valid = round(r2_score(target_valid, y_val_rf),2)
result.loc[result['model'] == 'Random Forest Regressor', 'valid_score'] = r2_score_acc_valid
print(f'Accuracy of RandomForestRegressor model prediction for valid dataset is {r2_score_acc_valid}')
```

```
Accuracy of RandomForestRegressor model prediction for valid dataset is 0.63
```

Рисунок 3.7 – Тренування моделі RandomForestRegressor, визначення точності

R2_score для моделі RandomForest склала 0.7 для тестових даних та 0.63 для валідаційних даних.

Останньою моделлю для розгляду буде модель XGBoostRegressor, тренування даних, а також визначення r2_score для тестових даних зображена на рисунку 3.8.

```
# Training model
xgb_CV = GridSearchCV(estimator=xgbr, param_grid=parameters, cv=cv_train, n_jobs=-1)
xgb_CV.fit(train, target_train)
print("Best score: %0.3f" % xgb_CV.best_score_)
print("Best parameters set:", xgb_CV.best_params_)

# Prediction for training data
y_train_xgb = xgb_CV.predict(train)

# Accuracy of model
r2_score_acc = round(r2_score(target_train, y_train_xgb),2)
print(f'Accuracy of XGBoost Regressor model training is {r2_score_acc}')

# Save to result dataframe
result.loc[result['model'] == 'XGBoost Regressor', 'train_score'] = r2_score_acc
```

```
Best score: 0.376
Best parameters set: {'learning_rate': 0.05, 'max_depth': 4, 'n_estimators': 100, 'reg_lambda': 0.3}
Accuracy of XGBoost Regressor model training is 0.99
CPU times: user 524 ms, sys: 23.9 ms, total: 548 ms
Wall time: 4min 2s
```

```
# Print rounded r2_score_acc to 2 decimal values after the text
y_val_xgb = xgb_CV.predict(valid)
r2_score_acc_valid = round(r2_score(target_valid, y_val_xgb),2)
result.loc[result['model'] == 'XGBoost Regressor', 'valid_score'] = r2_score_acc_valid
print(f'Accuracy of XGBoost Regressor model prediction for valid dataset is {r2_score_acc_valid}')

result
```

```
Accuracy of XGBoost Regressor model prediction for valid dataset is 0.7
```

Рисунок 3.8 – Тренування моделі XGBoost, визначення точності

R2_score для моделі XGBoostRegressor склала 0.99 для тренувальних даних та 0.7 для валідаційних даних.

Наступним кроком буде порівняння всі моделей та вибір моделі з найбільшою точністю. На рисунку 3.9 зображено значення r2_score для кожної з моделей.

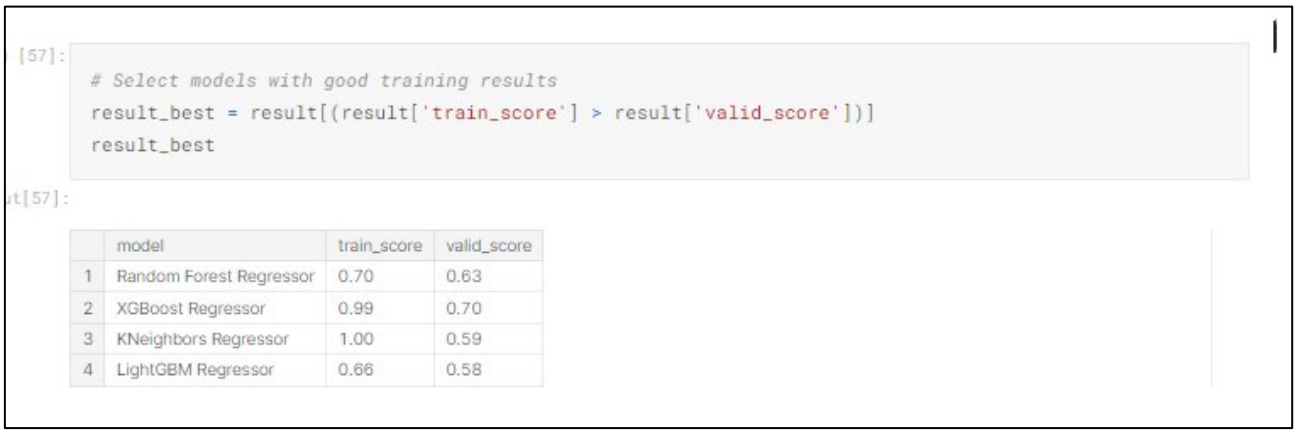
In [56]:
```
# Display results of modeling
result.sort_values(by=['valid_score', 'train_score'], ascending=False)
```

Out[56]:

| | model | train_score | valid_score |
|---|---|---|---|
| 0 | Linear Regression | 0.65 | 0.76 |
| 2 | XGBoost Regressor | 0.99 | 0.70 |
| 1 | Random Forest Regressor | 0.70 | 0.63 |
| 3 | KNeighbors Regressor | 1.00 | 0.59 |
| 4 | LightGBM Regressor | 0.66 | 0.58 |

Рисунок 3.9 – Відображення точності кожної моделі

Першим етапом вибору найбільш точної моделі є вибір моделей, що показали високу оцінку на тренувальних даних (рис.3.10).



```
[57]:
# Select models with good training results
result_best = result[(result['train_score'] > result['valid_score'])]
result_best
```

t[57]:

| | model | train_score | valid_score |
|---|---|---|---|
| 1 | Random Forest Regressor | 0.70 | 0.63 |
| 2 | XGBoost Regressor | 0.99 | 0.70 |
| 3 | KNeighbors Regressor | 1.00 | 0.59 |
| 4 | LightGBM Regressor | 0.66 | 0.58 |

Рисунок 3.10 – Відкидання моделей з не прийнятною точністю для тренувальних даних

Таким чином, модель лінійної регресії є незадовільною моделлю через низьку точність на тренувальних даних.

Наступним кроком буде вибір моделей, що показали низьке відхилення r2_score для тестових та валідаційних даних (рис. 3.11).

```
# Select models with minimal overfitting
result_best = result_best[(result_best['train_score'] - result_best['valid_score']).abs() < 0.15]
result_best.sort_values(by=['valid_score', 'train_score'], ascending=False)
```

]:

| | model | train_score | valid_score |
|---|---|---|---|
| 1 | Random Forest Regressor | 0.70 | 0.63 |
| 4 | LightGBM Regressor | 0.66 | 0.58 |

Рисунок 3.11 – Вибір моделей з найменшою різницею між «train_score» та «valid_score»

Оскільки моделі KNeighboursRegressor та XGBoost Regressor мають велике відхилення r2_score для валідаційних та тестових даних, дані моделі було відкинути.

Заключним кроком в виборі моделі постає вибір однієї з двох моделей що залишились, для цього було обрано модель що показала найбільшу точність на валідаційних даних (рис. 3.12).

```
# Select the best model
result_best.nlargest(1, 'valid_score')
```

| | model | train_score | valid_score |
|---|---|---|---|
| 1 | Random Forest Regressor | 0.7 | 0.63 |

```
# Find a name of the best model (with maximal valid score)
best_model_name = result_best.loc[result_best['valid_score'].idxmax(result_best['valid_score'].max
()), 'model']
```

```
print(f'The best model is "{best_model_name}"')
```

```
The best model is "Random Forest Regressor"
```

Рисунок 3.12 – Вибір найбільш точної моделі

Згідно отриманих даних, найбільш точною моделлю є «RandomForest Regressor» з показниками «train_score» = 0.7 та «valid_score» = 0.63.

На основі даної моделі буде проводитись прогнозування даних, а також виведення цих даних з метою подальшого імпорту в реалізовану систему.

### 3.2 Розробка бази даних для інформаційної системи

Першим етапом в побудові базі даних є проектування самої бази даних. Для побудови схеми бази даних було використано сервіс DBDesigner.

Спроектована база даних зображена на рисунку 3.13.

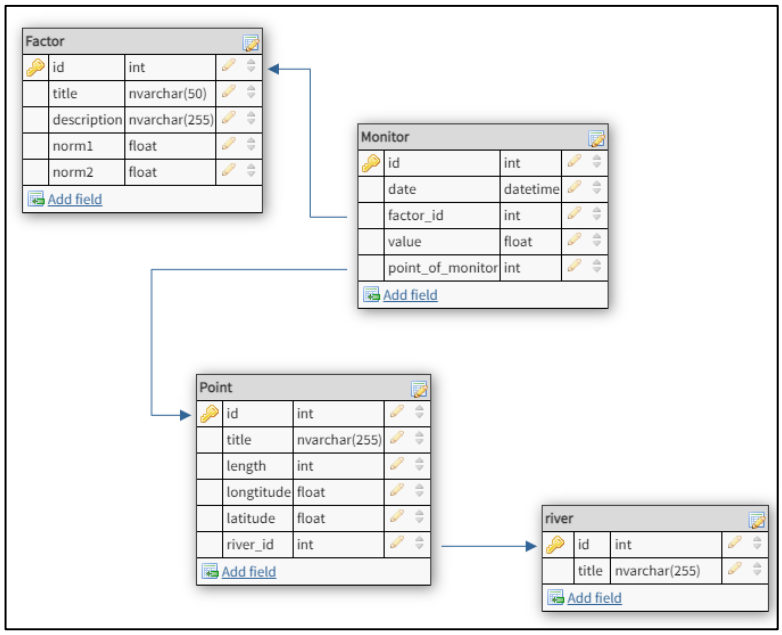



Рисунок 3.13 – Схема бази даних інформаційної системи аналізу та прогнозування

Створення бази даних та реалізація зв'язків відбувалась в інструменті phpMyAdmin, що працює на СУБД MySQL. Під час реалізації системи були створенні дані сутності та їх поля:

- Показник забруднення Factor (id);
- Точка вимірювання Point (id);
- Річка (id);
- Моніторинг (id);

Приклад створення таблиці зображено на рисунку 3.14.

```
CREATE TABLE `point` (`id` INT NOT NULL, `title` VARCHAR NOT NULL,
                      `length` INT NOT NULL, PRIMARY KEY (`id`))
ENGINE = INNODB;
```

Рисунок 3.14 – Приклад створення таблиці точок

Після створення таблиці, її структуру з полями та їх атрибутами можна переглянути в розділі «Структура» в phpMyAdmin, що зображена на рисунку 3.15.

| # | Name | Type | Comparison | Attributes | Null | Default | Comments | Additionally | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(11) | | | No | *No* | | | Change Delete More |
| 2 | title | varchar(255) | utf8mb4_general_ci | | No | *No* | | | Change Delete More |
| 3 | length | int(11) | | | No | *No* | | | Change Delete More |
| 4 | longtitude | float | | | No | *No* | | | Change Delete More |
| 5 | latitude | float | | | No | *No* | | | Change Delete More |

Рисунок 3.15 – Структура створеної таблиці

Після створення всіх таблиць, наступним кроком буде під’єднання база даних до розробленої системи, після чого буде проводитись подальша реалізація інших технологій та об’єднання їх в одну систему.

### 3.3 Реалізація системи у веб–додатку

Оскільки обраною мовою програмування стала саме PHP, для реалізації веб системи було обрано cPanel, за що завдяки підтримці Apache та MySQL, дозволяє розробляти веб додатки на PHP. Панель керування cPanel зображено на рисунку 3.16

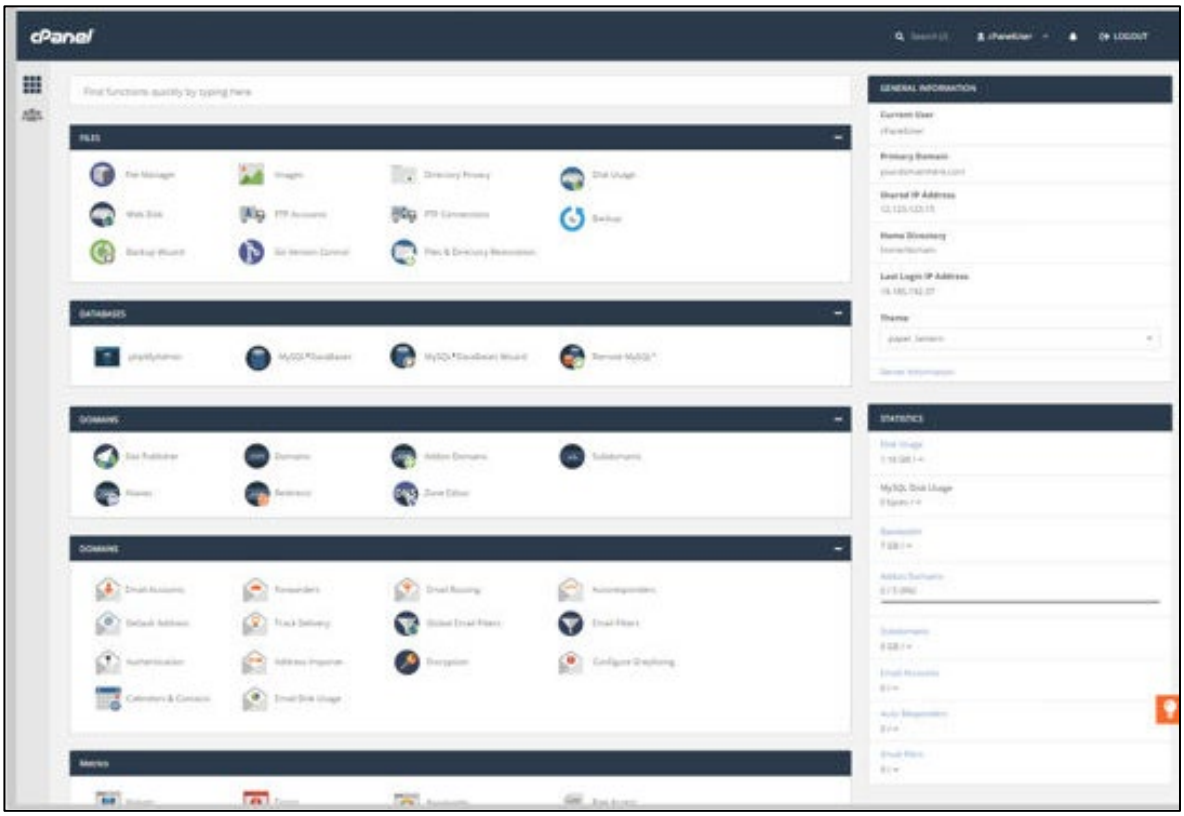



Рисунок 3.16 – Панель керування cPanel

Для розроблення системи, необхідно під’єднати базу даних за допомогою блоку PHP та підтримуваних модулів PHP таких як `mysqli` що за допомогою PHP коду налаштовує під’єднанням до бази даних (рис. 3.17).

```php
<?php
// Your PHP code goes here

// Example: Establishing a database connection (replace with your database credentials)
$servername = "localhost";
$username = "suth_bug";
$password = "your_password";
$dbname = "south_bug";

$conn = new mysqli($servername, $username, $password, $dbname);

// Check the connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Example: Fetching data from the database
$sql = "SELECT id, name, email FROM users";
$result = $conn->query($sql);
```

Рисунок 3.17 – Під’єднання до бази даних

Даний код демонструє приклад під'єднання до бази даних а також ініціалізацію SQL запиту, шляхом застосування модуля mysqli для PHP в самому коді.

Для реалізації імпорту даних отриманих в ході розроблення інформаційної системи аналізу та прогнозування, було застосовано фреймворк pyscript, що дозволяє використовувати мову програмування Python та її бібліотеки безпосередньо в HTML, а оскільки HTML повністю сумісний з PHP, дане рішення є цілком задовільним. Приклад застосування даного фреймворку зображено на рисунку 3.18.

```
<html>
    <head>
        <title></title>

        <link rel="stylesheet" href="https://pyscript.net/alpha/pyscript.css" />
        <script defer src="https://pyscript.net/alpha/pyscript.js"></script>

        <py-env>
            - matplotlib
        </py-env>
    </head>
    <body>
        <h1> Plotting a graph </h1>
        <div id="lineplot"> </div>

        <py-script>
            import matplotlib.pyplot as plt
            fig, ax = plt.subplots()

            year_2 = [2016, 2017, 2018, 2019, 2020, 2021]
            population_2 = [43, 43, 44, 44, 45, 45]

            plt.plot(year_2, population_2)


            pyscript.write('lineplot',fig)

        </py-script>
    </body>
</html>
```

Рисунок 3.18 – Приклад реалізації Python в HTML файлі

Саме даний фреймворк допоможе імпортувати дані отримані в ході аналізу та прогнозування даних, а також візуалізувати отримані дані, за допомогою таких бібліотек як Matplotlib.

### 3.4 Реалізація ГІС технології в системі

Після створення проекту необхідно додати мапу зі створеними точками досліджень в OpenStreetMaps (рис. 3.19).

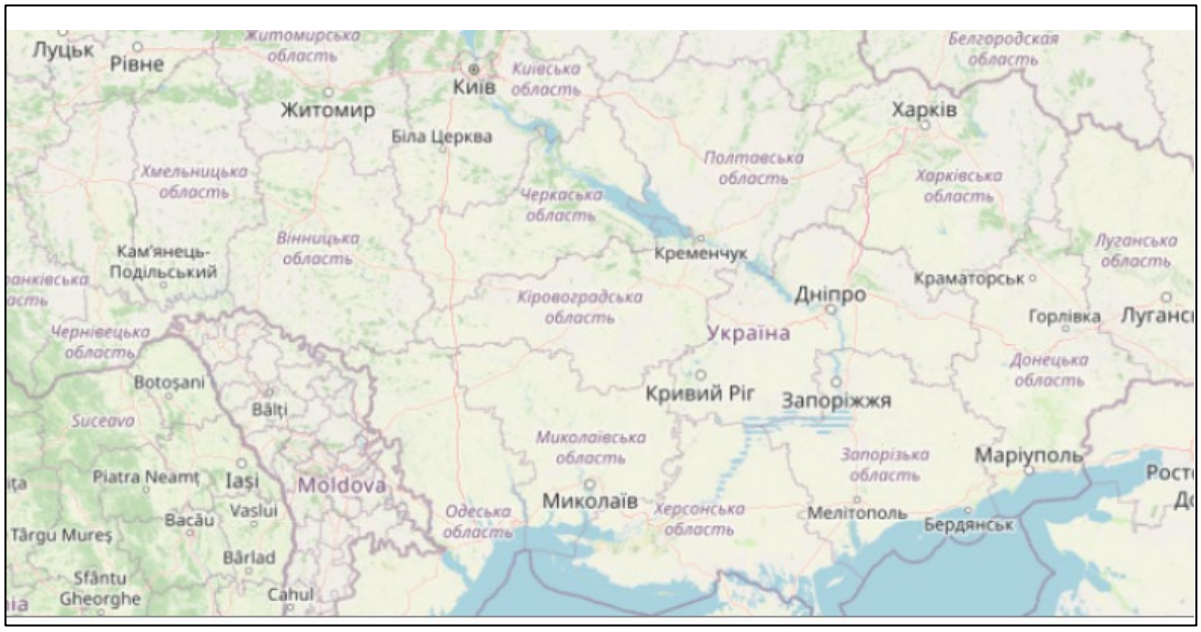



Рисунок 3.19 – приклад мапи в OpenStreetMap

OpenStreetMap має зручне API а також безліч рішень на PHP, що дозволяють просто та зручно додавати свої мапи, а також прив'язувати до них геодані.

Завдяки даним інструментам було побудовано мапу з прив'язкою точки дослідження до сторінки де зображено статистичні дані зібрані на станції моніторингу.

На рисунку 3.20 зображено інструмент редагування мапи, а саме інструмент прив'язки точок на мапі до сторінки, редагування точок, а також створення єдиної мапи використовуючи вже створені сторінки як геодані.

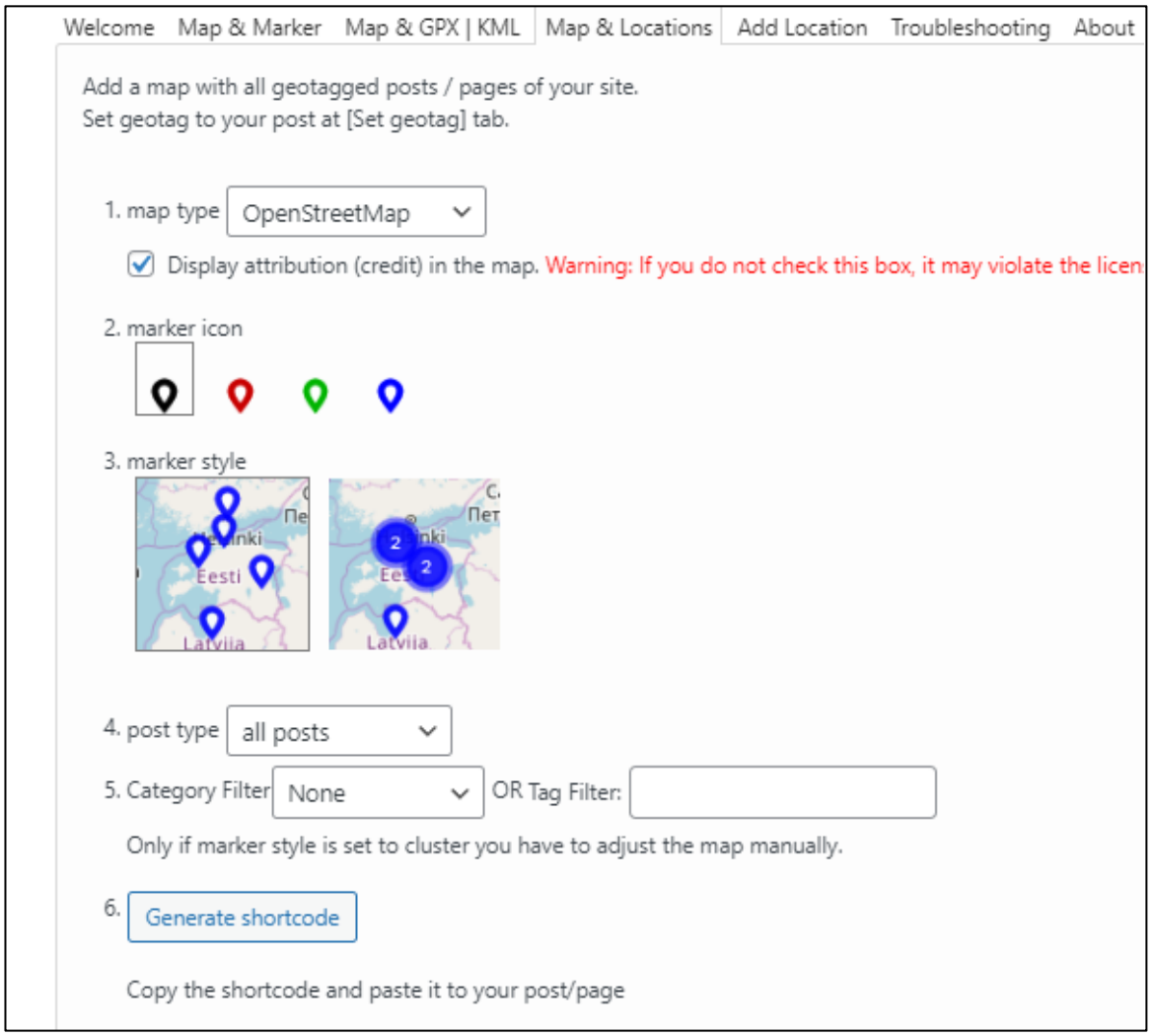



Рисунок 3.20 – Функція вбудови мапи в веб-сайт окремим блоком

Після копіювання коду, потрібно вставити блок на будь-яку сторінку, блок є окремим елементом OSM. Код додавання мапи та результат зображено на рисунку 3.21 та рисунку 3.22.

```
[osm_map_v3 map_center="43.7586,6.9242" zoom="11.0" width="100%" height="450"
tagged_type="page" marker_name="mic_black_pinother_02.png" map_border="thin solid grey"
tagged_param="cluster" tagged_color="black"]
```

Рисунок 3.21 – Код додавання мапи в проект

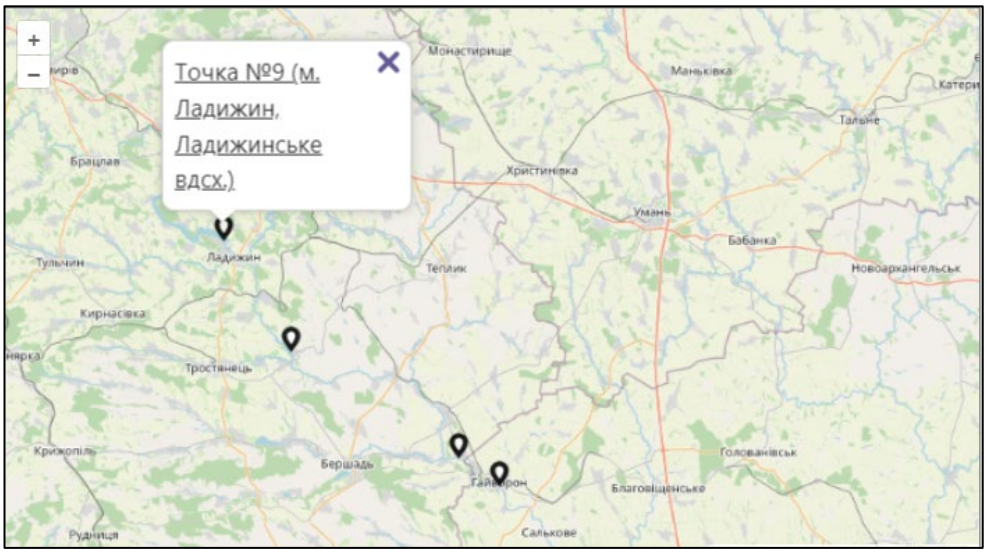



Рисунок 3.22 – Додана в проект мапа

На створеній мапі відображені точки, натискаючи на яку, користувач переспрямовувався на сторінку, де більш детально описано точку та зображено дані графіки що відносяться до цієї точки.

Після об’єднання всіх описаних технологій, в результаті ми отримаємо наступну схему інформаційної системи (рис. 3.23).

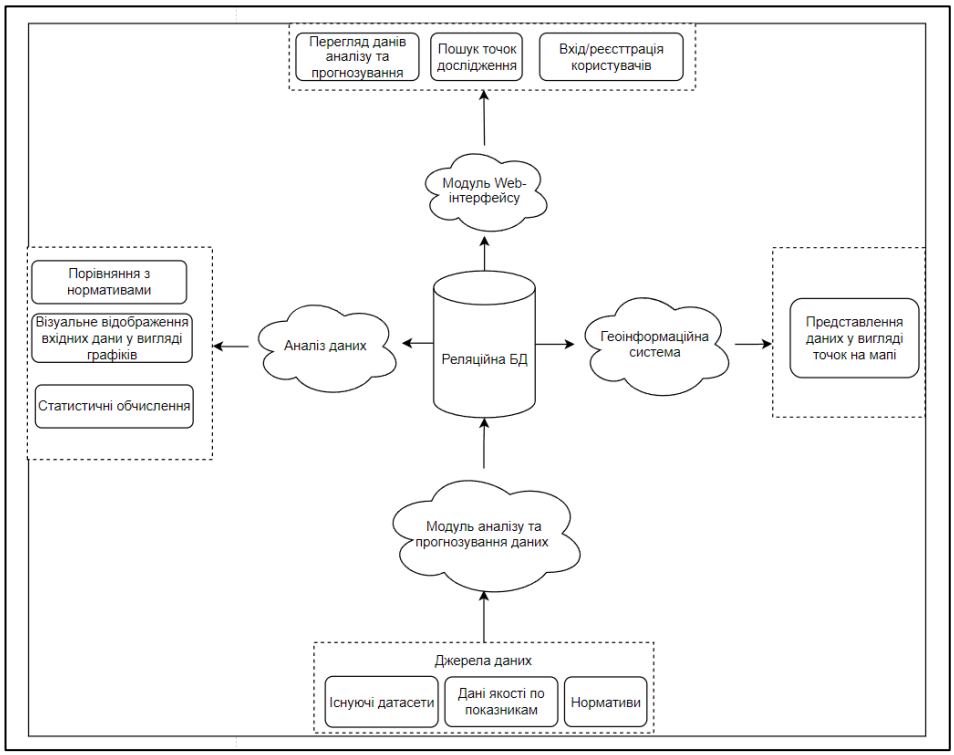



Рисунок 3.23 – Загальна структура системи

Для загального огляду функціоналу та ключових модулів системи було створено UML-діаграму варіантів використання (use case). Зображення результату цієї діаграми представлене на рисунку 3.24.

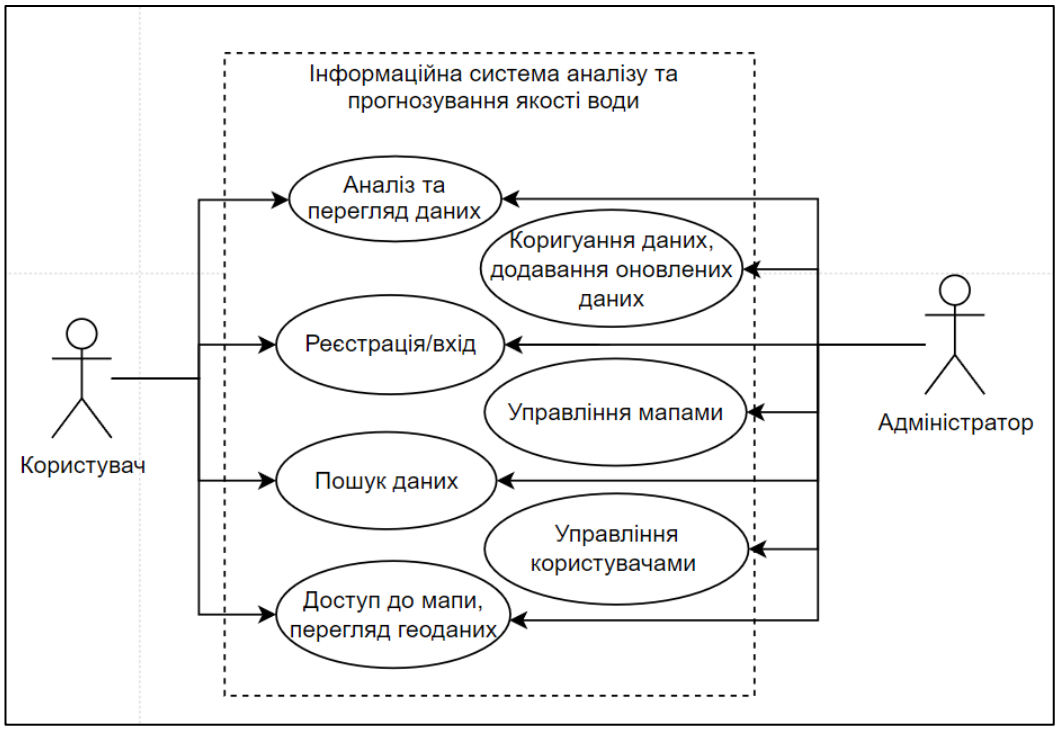



Рисунок 3.24 – Діаграма варіантів використання інформаційної системи

По завершенню проектування інформаційної системи, заключними етапами є заповнення бази даних вхідними даними інформаційної системи, для відображення текстової інформації, відображення графічного та текстового матеріалу.

### 3.5 Застосування інформаційної системи

З метою перевірки дієздатності розробленої системи, розглянемо головні її функції, а саме перегляд даних, що були отримані в результаті обробки даних в платформі Kaggle, а також розглянемо застосування моделей в самій системі, та перевіримо чи відповідають вони тим що були розраховані в платформі Kaggle.

Так на рисунку 3.25 зображено графік насичення киснем на станції дослідження в Ладижині, а також були додані ГДК для візуальної оцінки насичення киснем на даній ділянці.

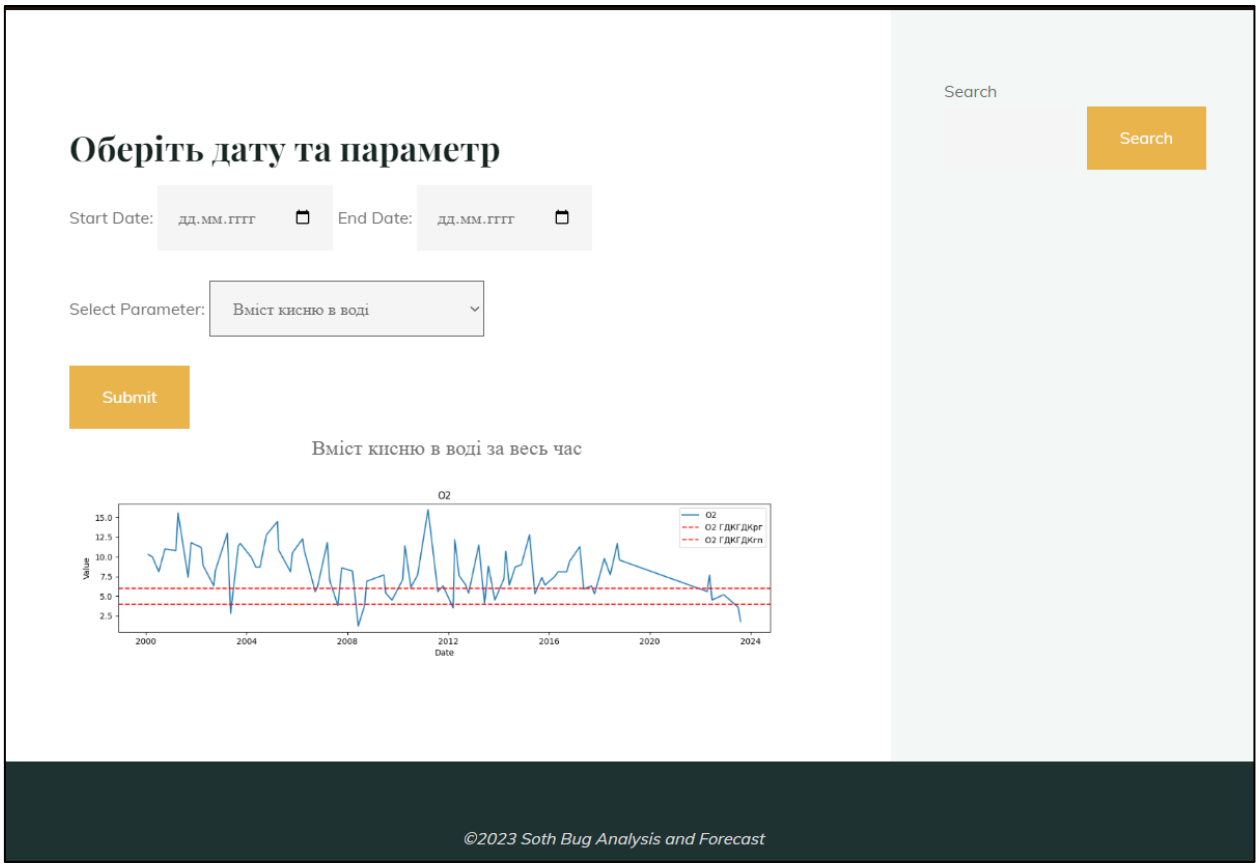



Рисунок 3.25 – Графік відображення насичення киснем води в точці дослідження м. Ладижин

Як видно з отриманого графіку, на початку двохтисячних даний показник знаходився на відносно високому рівні, але в останні роки пішов на спад, що може негативно вплинути на екосистему описаного об'єкту, адже для більшості показників, ГДК показує саме максимальну граничну концентрацію, але в випадку з насиченням кисню, даний показник є нижньою граничною допустимою концентрацією, тобто падіння даного показника може спричинити негативні наслідки. Даний показник було обрано з причини, що під час розвідувального аналізу було з'ясовано, що саме даних насичення киснем води є найбільше в даних за останні роки. Також розглянемо, функцію

вибору параметрів таких як діапазон спостережень, для цього оберемо дати за які ми хотіли б побачити дані, а також сам параметр (рис.3.26).

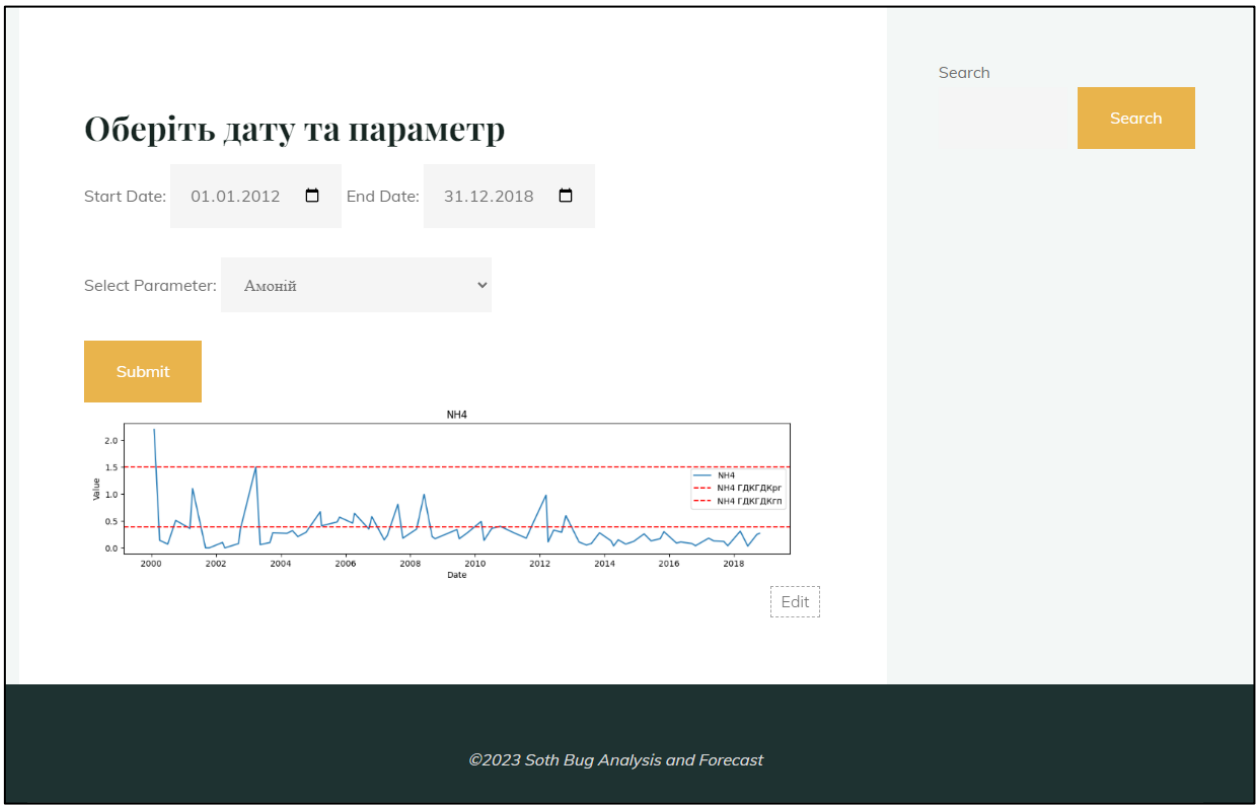



Рисунок 3.26 – Графік відображення вмісту амонію на станції Ладижин періодом з 2012 по 2018 рік.

Як видно з вище зображених рисунків, модуль відображення графічних даних, працює без ніяких проблем, тому візьмемо на розгляд функцію прогнозування даних в системі, а саме функцію вибору точки дослідження, ключової ознаки, факторів для моделювання, а також самих моделей. Для простоти порівняння даних системи, було обрано саме ті показники, що застосовувались в пункті 3.1 під час розроблення інформаційної технології аналізу та прогнозування даних, а саме:

- Точки досліджень: 14, 15 та 16 станція;
- Ключова ознака: O2;
- Фактори для моделювання: NH4, BSK5;

– Моделі: Linear Regression, Random Forest Regressor, XGBoost Regressor, KNeighbors Regressor, LightGBM Regressor (рис. 3.27).

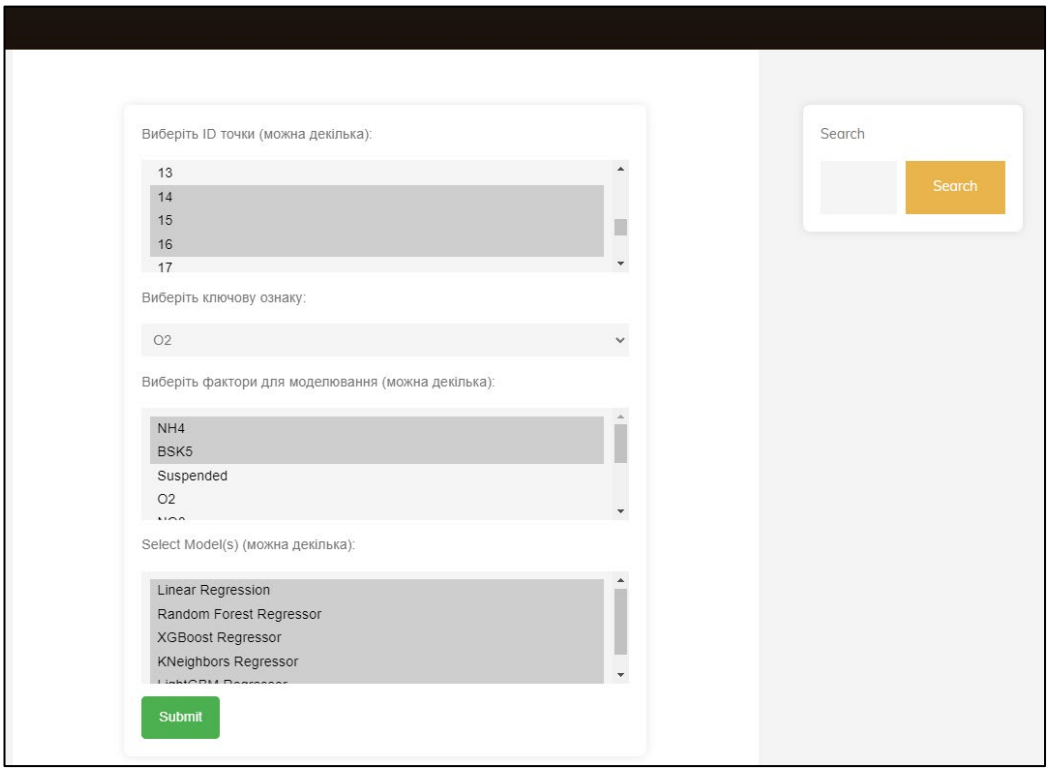



Рисунок 3.27 – Вибір параметрів для відображення навчання моделей.

Після вибору всіх показників, ми отримали таблицю точності кожної моделі, яку можна порівняти з рисунком 3.9, та запевнитись, що вони є ідентичними (рис. 3.27).

**Точність моделей для обраних моделей**

| model | Модель | train_score | valid_score |
|---|---|---|---|
| 1 | Random Forest Regressor | 0.7 | 0.63 |
| 0 | Linear Regression | 0.66 | 0.58 |
| 2 | XGBoost Regressor | 0.99 | 0.70 |
| 4 | LightGBM Regressor | 0.66 | 0.58 |
| 3 | KNeighbors Regressor | 1.00 | 0.59 |

Рисунок 3.28 – Точність кожної обраної моделі

Також на сторінці після обрання даних зображуються графіки передбачень обраних моделей на тренувальних, валідаційних та тестових даних – дані графіки показують настільки точно кожній моделі вдалось спрогнозувати значення ключової ознаки відносно дійсного значення (рис. 3.29 – 3.31).

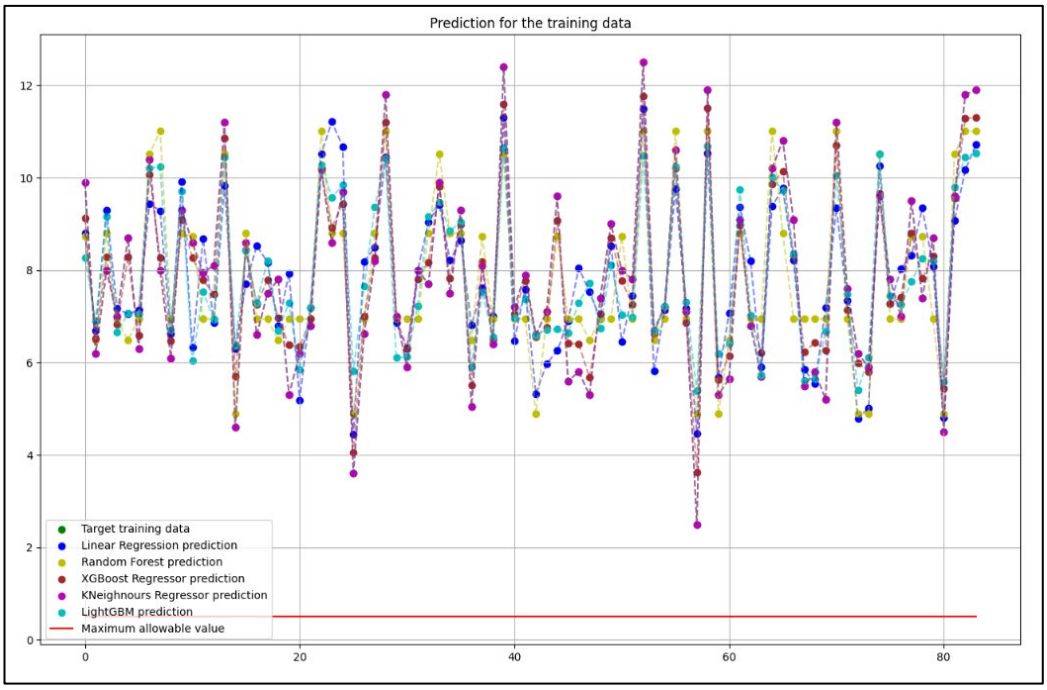



Рисунок 3.29 – Графік прогнозування для тренувальних даних

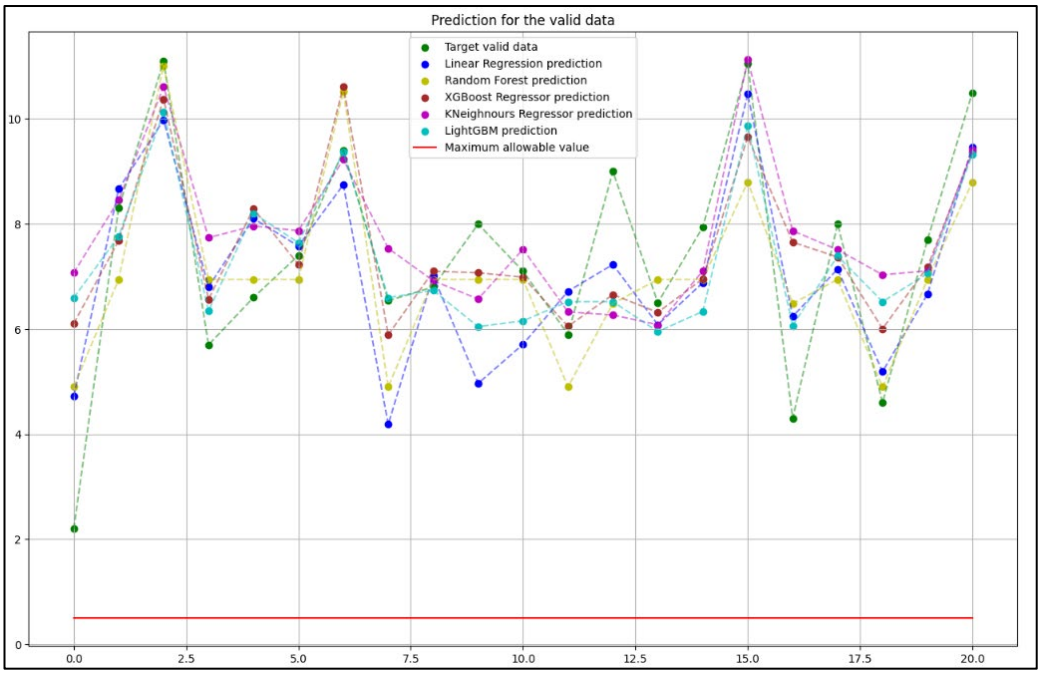



Рисунок 3.30 – Графік прогнозування для валідаційних даних

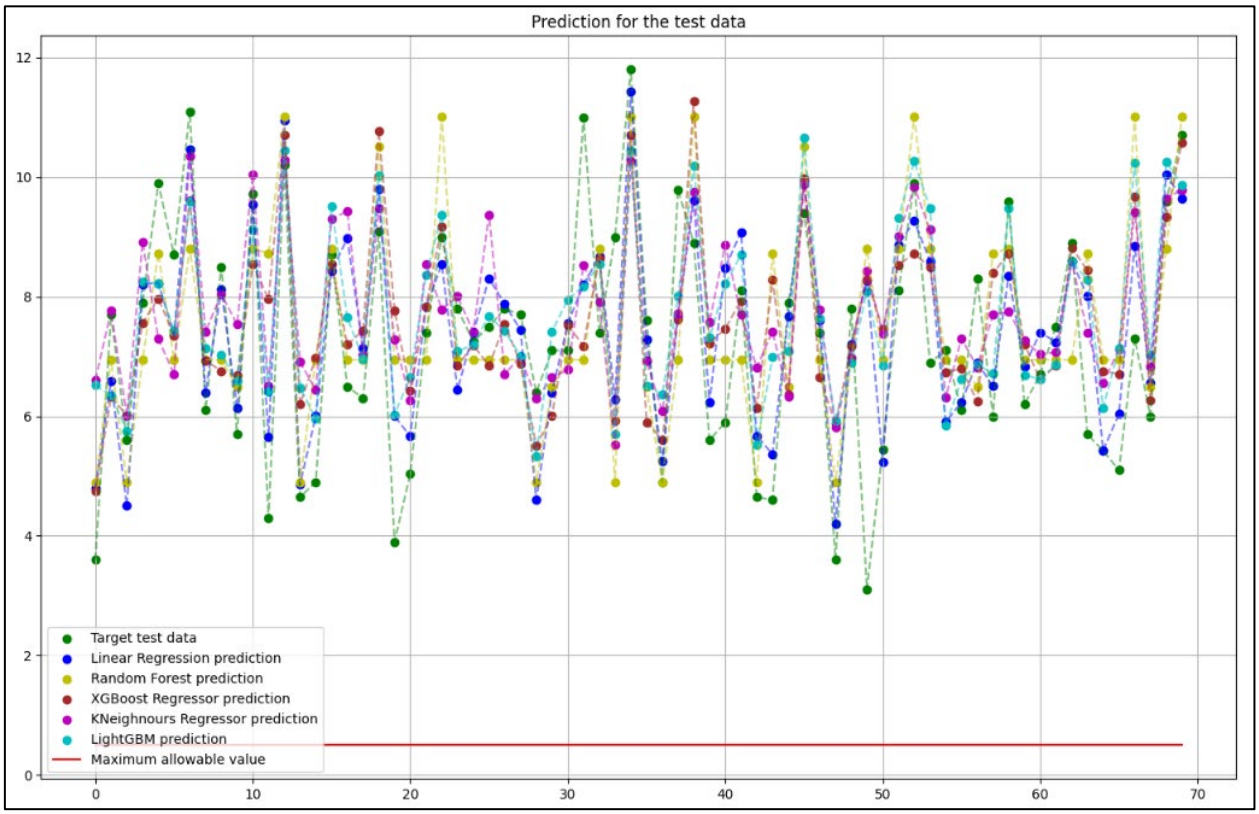



Рисунок 3.31 – Графік прогнозування для тестових даних

Також в звіті присутній відбір кожної моделі, на рисунку 3.30 зображено результат відбору моделі за такими показниками:

– Обираються моделі з гарними результатами тренування;

– Обираються моделі з мінімальним відхиленням між «train_score» та «valid_score»;

– Обирається модель з найбільшим значенням «valid_score» (рис.3.32)

**Найкраща модель**

| Модель | train_score | valid_score |
|---|---|---|
| Random Forest Regressor | 0.7 | 0.63 |

Рисунок 3.32 – Вибір найкращої моделі для обраних даних

За результатами обраної моделі будується графік, що графічно відображує відхилення найкращої моделі від тестових, тренувальних та валідаційних даних. Дані графіки показано на рисунках 3.33, 3.34 та 3.35.

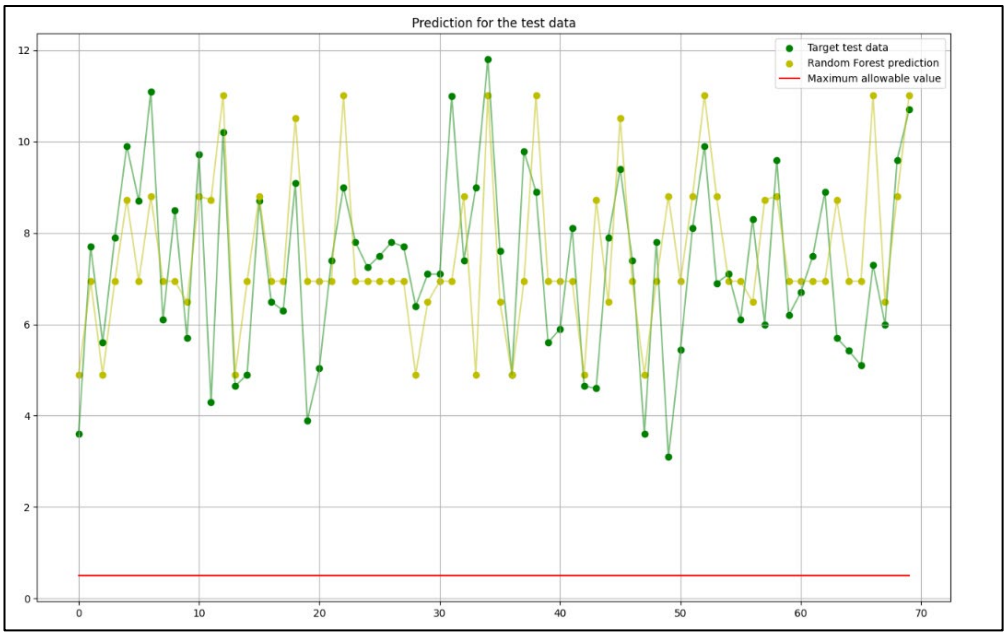



Рисунок 3.33 – Графік прогнозування моделі RandomForestRegressor для тестових даних

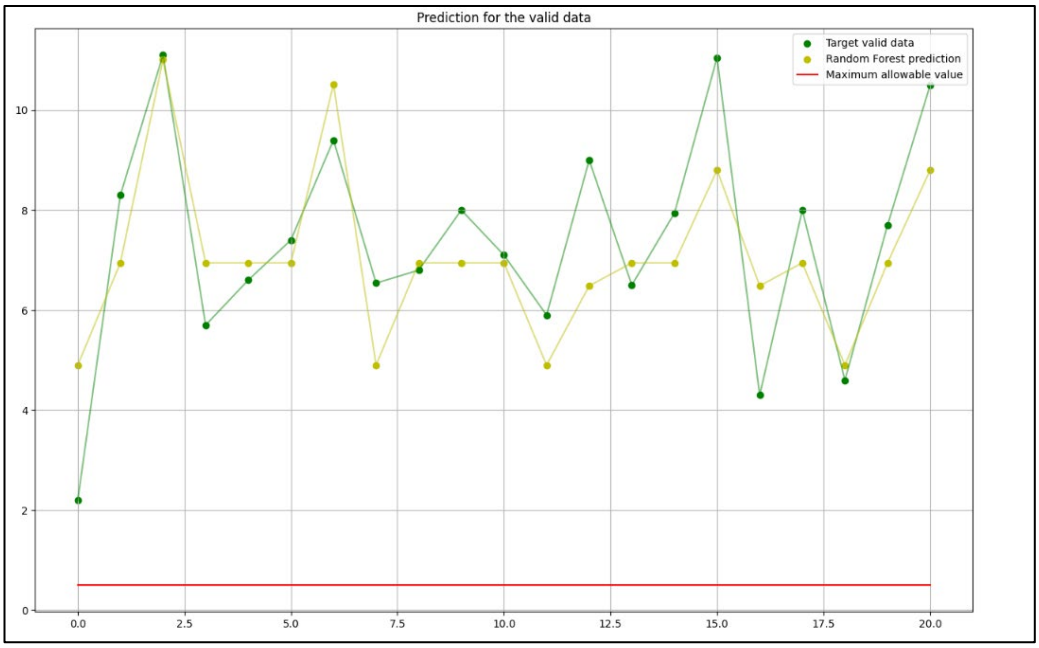



Рисунок 3.34 – Графік прогнозування моделі RandomForestRegressor для валідаційних даних

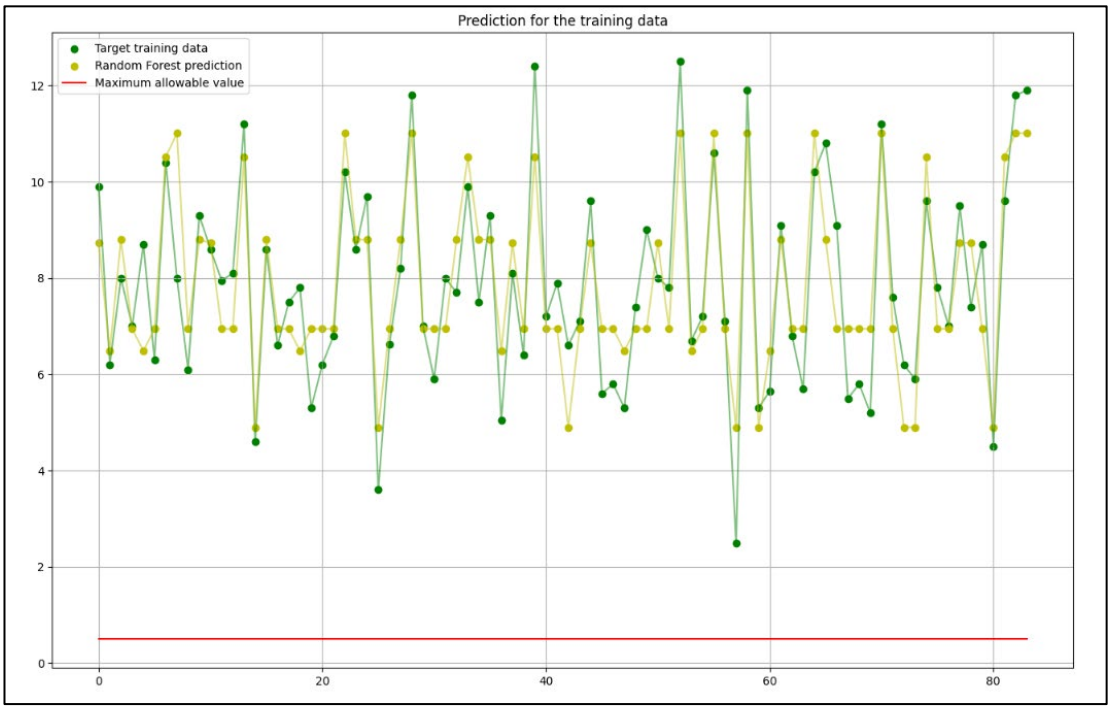



Рисунок 3.35 – Графік прогнозування моделі RandomForestRegressor для тренувальних даних

Отже за даними отримані під час тестування, можна дійти висновку, що дані побудовані на платформі Kaggle були успішно імпортовані.

### 3.6 Висновки

В даному розділі було реалізовано інформаційні технології, що складають інформаційну систему аналізу та прогнозування, складено алгоритм роботи однієї з технологій, схему бази даних, а також веб-системи, зображено UML діаграму використання, проведено тренування моделей та прогнозування даних на основі тренованих моделей, обрано найкращу з моделей. Також було розроблену базу даних, веб-інтерфейс та ГІС для інформаційної системи, імпортовано результати аналізу та прогнозування даних, проведено тестування системи, де за обраними параметрами найточнішою моделлю виявилась RandomForestRegressor з оцінкою «r2_score» в 0.7 для тренувальних даних та 0.63 для валідаційних.

За результатами тестування можна дійти висновку, що всі етапи розроблення системи були виконанні успішно, а розроблена система представляє собою самостійне рішення, що може застосовуватись для вирішення завдання аналізу та прогнозування даних якості води в річці Південний Буг.

# 4 ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота з розробки та дослідження «Інформаційної системи аналізу та прогнозування якості води у річці Південний Буг» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

Для наведеного випадку нами мають бути виконані такі етапи робіт:

1) проведено комерційний аудит науково-технічної розробки, тобто встановлення її науково-технічного рівня та комерційного потенціалу;

2) розраховано витрати на здійснення науково-технічної розробки;

3) розрахована економічна ефективність науково-технічної розробки у випадку її впровадження і комерціалізації потенційним інвестором і проведено обґрунтування економічної доцільності комерціалізації потенційним інвестором.

## 4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в таблиці 4.1 [31].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

| Бали (за 5-ти бальною шкалою) | | | | | |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 |
| Технічна здійсненність концепції | | | | | |
| 1 | Достовірність концепції не підтверджена | Концепція підтверджена експертними висновками | Концепція підтверджена розрахунками | Концепція перевірена на практиці | Перевірено працездатність продукту в реальних умовах |
| Ринкові переваги (недоліки) | | | | | |
| 2 | Багато аналогів на малому ринку | Мало аналогів на малому ринку | Кілька аналогів на великому ринку | Один аналог на великому ринку | Продукт не має аналогів на великому ринку |
| 3 | Ціна продукту значно вища за ціни аналогів | Ціна продукту дещо вища за ціни аналогів | Ціна продукту приблизно дорівнює цінам аналогів | Ціна продукту дещо нижче за ціни аналогів | Ціна продукту значно нижче за ціни аналогів |
| 4 | Технічні та споживчі властивості продукту значно гірші, ніж в | Технічні та споживчі влас-тивості продукту трохи гірші, ніж в аналогів | Технічні та споживчі властивості продукту на рівні аналогів | Технічні та споживчі властивості продукту трохи кращі, ніж в | Технічні та споживчі властивості продукту значно кращі, ніж в |
| 5 | Експлуатаційні витрати значно вищі, ніж в аналогів | Експлуатаційні витрати дещо вищі, ніж в аналогів | Експлуатаційні витрати на рівні експлуатаційних витрат аналогів | Експлуатаційні витрати трохи нижчі, ніж в аналогів | Експлуатаційні витрати значно нижчі, ніж в аналогів |
| Ринкові перспективи | | | | | |
| 6 | Ринок малий і не має позитивної динаміки | Ринок малий, але має позитивну динаміку | Середній ринок з позитивною динамікою | Великий стабільний ринок | Великий ринок з позитивною динамікою |

Продовження таблиці 4.1

| Бали (за 5-ти бальною шкалою) | | | | | |
|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 |
| Ринкові перспективи | | | | | |
| 7 | Активна конкуренція великих компаній на | Активна конкуренція | Помірна конкуренція | Незначна конкуренція | Конкурентів немає |
| Практична здійсненність | | | | | |
| 8 | Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї | Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців | Необхідне незначне навчання фахівців та збільшення їх штату | Необхідне незначне навчання фахівців | Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї |
| 9 | Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні | Потрібні незначні фінансові ресурси. Джерела фінансування відсутні | Потрібні значні фінансові ресурси. Джерела фінансування є | Потрібні незначні фінансові ресурси. Джерела фінансування є | Не потребує додаткового фінансування |
| 10 | Необхідна розробка нових матеріалів | Потрібні матеріали, що використовуються у військово промисловому комплексі | Потрібні дорогі матеріали | Потрібні досяжні та дешеві матеріали | Всі матеріали для реалізації ідеї відомі та давно використовуютьс я у виробництві |
| 11 | Термін реалізації ідеї більший за 10 років | Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років | Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років | Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років | Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років |
| 12 | Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту | Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу | Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу | Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту | Відсутні будь- які регламентні обмеження на виробництво та реалізацію продукту |

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки наведено в таблиці 4.2.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

| Критерії | Експерт (ПІБ, посада) | | |
|---|---|---|---|
| | 1 | 2 | 3 |
| | Бали: | | |
| 1. Технічна здійсненність концепції | 4 | 5 | 4 |
| 2. Ринкові переваги (наявність аналогів) | 3 | 2 | 3 |
| 3. Ринкові переваги (ціна продукту) | 4 | 4 | 3 |
| 4. Ринкові переваги (технічні властивості) | 3 | 3 | 3 |
| 5. Ринкові переваги (експлуатаційні витрати) | 2 | 2 | 2 |
| 6. Ринкові перспективи (розмір ринку) | 3 | 3 | 3 |
| 7. Ринкові перспективи (конкуренція) | 2 | 2 | 2 |
| 8. Практична здійсненність (наявність фахівців) | 5 | 5 | 5 |
| 9. Практична здійсненність (наявність фінансів) | 2 | 3 | 2 |
| 10. Практична здійсненність (необхідність нових матеріалів) | 4 | 5 | 5 |
| 11. Практична здійсненність (термін реалізації) | 3 | 4 | 5 |
| 12. Практична здійсненність (розробка документів) | 4 | 5 | 4 |
| Сума балів | 39 | 43 | 41 |
| Середньоарифметична сума балів $СБ_с$ | 41,0 | | |

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в таблиці 4.3 [31].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

| Середньоарифметична сума балів СБ ,розрахована на основі висновків експертів | Науково-технічний рівень та комерційний потенціал розробки |
|---|---|
| 41…48 | Високий |
| 31…40 | Вище середнього |
| 21…30 | Середній |
| 11…20 | Нижче середнього |
| 0…10 | Низький |

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» становить 41,0 бала, що, відповідно до таблиці 4.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

## 4.2 Визначення рівня конкурентоспроможності розробки

В процесі визначення економічної ефективності науково-технічної розробки також доцільно провести прогноз рівня її конкурентоспроможності за сукупністю параметрів, що підлягають оцінюванню.

Одиничний параметричний індекс розраховуємо за формулою [31].

$$q_i = \frac{P_i}{P_{баз\,i}}. \qquad (4.1)$$

де $q_i$ – одиничний параметричний індекс, розрахований за $i$-м параметром;

$P_i$ – значення $i$-го параметра виробу;

$P_{базі}$ – аналогічний параметр базового виробу-аналога, з яким проводиться порівняння.

Загальні технічні та економічні характеристики розробки представлено в таблиці 4.4.

Таблиця 4.4 – Основні техніко-економічні показники аналога та розробки, що проектується

| Показники (параметри) | Одиниця вимірю-вання | Аналог | Проектований пристрій | Відношення параметрів нової розробки до аналога | Питома вага показника |
|---|---|---|---|---|---|
| Кількість прогнозованих показників інформаційної системи | одиниць | 5 | 10 | 2 | 0,3 |

Продовження таблиці 4.4

| Показники (параметри) | Одиниця вимірю-вання | Аналог | Проектований пристрій | Відношення параметрів нової розробки до аналога | Питома вага показника |
|---|---|---|---|---|---|
| Об’єм бази даних інформації | МВ | 500 | 1000 | 2 | 0,2 |
| Точність прогнозу | % | 64 | 70 | 1,09 | 0,15 |
| Кількість аналізованих показників інформаційної системи | одиниць | 6 | 10 | 1,88 | 0,25 |
| Кількість станцій дослідження | одиниць | 7 | 24 | 3,43 | 0,1 |
| Експлуатаційні витрати | грн | 2850 | 2100 | 0,74 | 0,5 |
| Вартість інформаційної системи | грн | 180000 | 110000 | 0,61 | 0,5 |

Нормативні параметри оцінюємо показником, який отримує одне з двох значень: 1 – пристрій відповідає нормам і стандартам; 0 – не відповідає.

Груповий показник конкурентоспроможності за нормативними параметрами розраховуємо як добуток частинних показників за кожним параметром за формулою [31]:

$$I_{НП} = \prod_{i=1}^{n} q_i \ , \qquad (4.2)$$

де $I_{нп}$ – загальний показник конкурентоспроможності за нормативними параметрами;

$q_i$ – одиничний (частинний) показник за $i$-м нормативним параметром;

$n$ – кількість нормативних параметрів, які підлягають оцінюванню.

За нормативними параметрами розроблюваний пристрій відповідає вимогам ДСТУ, тому $I_{нп} = 1$.

Значення групового параметричного індексу за технічними параметрами визначаємо з урахуванням вагомості (частки) кожного параметра [31]:

$$I_{ТП} = \sum_{i=1}^{n} q_i \cdot \alpha_i \ , \tag{4.3}$$

де $I_{тп}$ – груповий параметричний індекс за технічними показниками (порівняно з виробом-аналогом);

$q_i$ – одиничний параметричний показник $i$-го параметра;

$\alpha_i$ – вагомість $i$-го параметричного показника, $\sum_{i=1}^{n} \alpha_i = 1$;

$n$ – кількість технічних параметрів, за якими оцінюється конкурентоспроможність.

Проведемо аналіз параметрів згідно даних таблиці 4.4.

$$I_{mn} = 2 \cdot 0{,}3 + 2 \cdot 0{,}2 + 1{,}09 \cdot 0{,}15 + 1{,}88 \cdot 0{,}25 + 3{,}43 \cdot 0{,}1 = 1{,}98.$$

Груповий параметричний індекс за економічними параметрами розраховуємо за формулою [31]:

$$I_{ЕП} = \sum_{i=1}^{m} q_i \cdot \beta_i \ , \tag{4.4}$$

де $I_{ЕП}$ – груповий параметричний індекс за економічними показниками;

$q_i$ – економічний параметр $i$-го виду;

$\beta_i$ – частка $i$-го економічного параметра, $\sum_{i=1}^{m} \beta_i = 1$;

$m$ – кількість економічних параметрів, за якими здійснюється оцінювання. Проведемо аналіз параметрів згідно даних таблиці .

$$I_{ЕП} = 0{,}74 \cdot 0{,}5 + 0{,}61 \cdot 0{,}5 = 0{,}68.$$

На основі групових параметричних індексів за нормативними, технічними та економічними показниками розрахуємо інтегральний показник конкурентоспроможності за формулою [31]:

$$K_{ІНТ} = I_{НП} \cdot \frac{I_{ТП}}{I_{ЕП}}, \tag{4.5}$$

$$K_{ІНТ} = 1 \cdot 1{,}98 / 0{,}68 = 2{,}93.$$

Інтегральний показник конкурентоспроможності $К_{ІНТ} > 1$, отже розробка переважає відомі аналоги за своїми техніко-економічними показниками.

### 4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов’язані з проведенням науково-дослідної роботи на тему «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

#### 4.3.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників ($З_о$) розраховуємо у відповідності до посадових окладів працівників, за формулою [31]:

$$З_о = \sum_{i=1}^{k} \frac{М_{ni} \cdot t_i}{Т_р}, \quad (4.6)$$

де $k$ – кількість посад дослідників залучених до процесу досліджень;

$М_{ni}$ – місячний посадовий оклад конкретного дослідника, грн;

$t_i$ – число днів роботи конкретного дослідника, дн.;

$Т_р$ – середнє число робочих днів в місяці, $Т_р$=22 дні.

$$З_о = 18300{,}00 \cdot 28 / 22 = 23290{,}91 \ (грн).$$

Проведені розрахунки наведено в таблиці 4.5.

Таблиця 4.5 – Витрати на заробітну плату дослідників

| Найменування посади | Місячний посадовий оклад, грн | Оплата за робочий день, грн | Число днів роботи | Витрати на заробітну плату, грн |
|---|---|---|---|---|
| Керівник науково-дослідної роботи | 18300,00 | 831,82 | 28 | 23290,91 |
| Інженер-розробник інформаційних систем | 17500,00 | 795,45 | 28 | 22272,73 |
| Консультатнт (експерт-еколог) | 18000,00 | 818,18 | 11 | 9000,00 |
| Фахівець 1-ї категорії | 8500,00 | 386,36 | 22 | 8500,00 |
| Всього | | | | 63063,64 |

Основна заробітна плата робітників.

Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт НДР на тему «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» розраховуємо за формулою:

$$З_р = \sum_{i=1}^{n} С_i \cdot t_i, \quad (4.7)$$

де $С_i$ – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

$t_i$ – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду $С_i$ можна визначити за формулою:

$$С_i = \frac{М_М \cdot К_i \cdot К_с}{Т_р \cdot t_{зм}}, \quad (4.8)$$

де $М_М$ – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), приймемо *Мм*=6700,00 грн;

$К_і$ – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду [31];

$К_с$ – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

$Т_р$ – середнє число робочих днів в місяці, приблизно $Т_р$ = 22 днів;

$t_{зм}$ – тривалість зміни, год.

$С_1 = 6700{,}00 \cdot 1{,}10 \cdot 1{,}35 / (22 \cdot 8) = 56{,}53$ *(грн).*
$З_{р1} = 56{,}53 \cdot 5{,}40 = 305{,}27$ *(грн).*

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

| Найменування робіт | Тривалість роботи, год | Розряд роботи | Тарифний коефіцієнт | Погодинна тарифна ставка, грн | Величина оплати на робітника грн |
|---|---|---|---|---|---|
| Підготовка робочого місця розробника інформаційної системи | 5,40 | 2 | 1,10 | 56,53 | 305,27 |
| Інсталяція програмного забезпечення середовища розробки і моделювання | 7,30 | 3 | 1,35 | 69,38 | 506,47 |
| Компіляція програмних блоків моделювання інформаційної системи | 5,80 | 4 | 1,50 | 77,09 | 447,11 |
| Підготовка локального серверного обладнання для проведення досліджень | 6,00 | 5 | 1,70 | 87,37 | 524,20 |

Продовження таблиці 4.6

| Найменування робіт | Тривалість роботи, год | Розряд роботи | Тарифний коефіцієнт | Погодинна тарифна ставка, грн | Величина оплати на робітника грн |
|---|---|---|---|---|---|
| Налагодження програмних блоків збору інформації | 16,00 | 5 | 1,70 | 87,37 | 1397,86 |
| Тестування системи | 8,00 | 2 | 1,10 | 56,53 | 452,25 |
| Всього | | | | | 3633,16 |

Додаткова заробітна плата дослідників та робітників.

Додаткову заробітну плату розраховуємо як 10 … 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = (З_о + З_р) \cdot \frac{Н_{дод}}{100\%}, \quad (4.9)$$

де $Н_{дод}$ – норма нарахування додаткової заробітної плати. Приймемо 11%.

*$З_{дод}$ = (63063,64 + 3633,16) · 11 / 100% = 7336,65 (грн).*

### 4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_н = (З_о + З_р + З_{дод}) \cdot \frac{Н_{зп}}{100\%} \quad (4.10)$$

де $Н_{зп}$ – норма нарахування на заробітну плату. Приймаємо 22%.

*Зн = (63063,64 + 3633,16 + 7336,65) · 22 / 100% = 16287,36 (грн).*

### 4.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та

витрачені на проведення досліджень за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг».

Витрати на матеріали (*М*), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^{n} H_j \cdot Ц_j \cdot K_j - \sum_{j=1}^{n} B_j \cdot Ц_{вj}, \quad (4.11)$$

де $H_j$ – норма витрат матеріалу *j*-го найменування, кг;

*n* – кількість видів матеріалів;

$Ц_j$ – вартість матеріалу *j*-го найменування, грн/кг;

$K_j$ – коефіцієнт транспортних витрат, ($K_j = 1,1 \ldots 1,15$);

$B_j$ – маса відходів *j*-го найменування, кг;

$Ц_{вj}$ – вартість відходів *j*-го найменування, грн/кг.

$M_1 = 3,0 \cdot 206,00 \cdot 1,05 - 0,0 \cdot 0,0 = 648,90$ *(грн).*

Проведені розрахунки наведено в таблиці 4.7.

Таблиця 4.7 – Витрати на матеріали

| Найменування матеріалу, марка, тип, сорт | Ціна за 1 кг, грн | Норма витрат, кг | Величина відходів, кг | Ціна відходів, грн/кг | Вартість витраченого матеріалу, грн |
|---|---|---|---|---|---|
| Папір офісний А4 білий (80%) | 206,00 | 3,0 | 0,0 | 0,0 | 648,90 |
| Диск оптичний (CD-R) | 27,00 | 2,0 | 0,0 | 0,0 | 56,70 |
| Органайзер офісний BOX-16A | 198,00 | 4,0 | 0,0 | 0,0 | 831,60 |
| Канцелярське приладдя | 216,00 | 4,0 | 0,0 | 0,0 | 907,20 |
| Тонер HP-26 (для заправки картриджа) | 218,00 | 2,0 | 0,0 | 0,0 | 457,80 |
| USB-пам’ять | 219,00 | 2,0 | 0,0 | 0,0 | 459,90 |
| Всього | | | | | 3362,10 |

### 4.3.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі *($К_в$)*, які використовують при проведенні НДР на тему «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг», розраховуємо, згідно з їхньою номенклатурою, за формулою:

$$К_в = \sum_{j=1}^{n} Н_j \cdot Ц_j \cdot К_j \qquad (4.12)$$

де $Н_j$ – кількість комплектуючих $j$-го виду, шт.;

$Ц_j$ – покупна ціна комплектуючих $j$-го виду, грн;

$К_j$ – коефіцієнт транспортних витрат, ($К_j = 1,1 \ldots 1,15$).

$$К_в = 1 \cdot 3820,00 \cdot 1,05 = 4011,00 \text{ (грн)}.$$

Проведені розрахунки наведено в таблиці 4.8.

Таблиця 4.8 – Витрати на комплектуючі

| Найменування комплектуючих | Кількість, шт. | Ціна за штуку, грн | Сума, грн |
|---|---|---|---|
| RouterBOARD TP-Link230 | 1 | 3820,00 | 4011,00 |
| Диск жорсткий зовнішній Transcend StoreJet 1TB USB 3.1 Iron Gray Slim (TS1TSJ25M3S) | 1 | 2699,00 | 2833,95 |
| Всього | | | 6844,95 |

### 4.3.5 Спецустаткування для наукових (експериментальних) робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{спец} = \sum_{i=1}^{k} Ц_i \cdot C_{пр.i} \cdot K_i \ , \qquad (4.13)$$

де $Ц_i$ – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{пр.i}$ –кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$ – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($K_i$ = *1,10…1,12);*

$k$ – кількість найменувань устаткування.

$B_{спец} = 9999,00 \cdot 1 \cdot 1,05 = 10498,95$ *(грн).*

Отримані результати наведено в таблиці 4.9.

Таблиця 4.9 – Витрати на придбання спецустаткування по кожному виду

| Найменування устаткування | Кількість, шт | Ціна за одиницю, грн | Вартість, грн |
|---|---|---|---|
| Маршрутизатор VPN TP-Link ER7212PC | 1 | 9999,00 | 10498,95 |
| Серверне обладнання (локальне) на базі QUBE QB Ryzen 5 7600X RTX 4070 Ti 12GB W1 3210 (Ryzen57600XRTX4070Ti12GB W13210) | 1 | 67999,00 | 71398,95 |
| Імітатор станції досліджень моніторингу поверхневих вод | 1 | 21599,00 | 22678,95 |
| Всього | | | 104576,85 |

### 4.3.6 Програмне забезпечення для наукових (експериментальних) робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{прг} = \sum_{i=1}^{k} Ц_{iпрг} \cdot C_{прг.i} \cdot K_i \ , \qquad (4.14)$$

де $Ц_{iпрг}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{прг.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$ – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10…1,12$);

$k$ – кількість найменувань програмних засобів.

$B_{прг} = 7499,00 \cdot 1 \cdot 1,05 = 7873,95$ (грн).

Отримані результати наведено в таблиці 4.10.

Таблиця 4.10 – Витрати на придбання програмних засобів по кожному виду

| Найменування програмного засобу | Кількість, шт | Ціна за одиницю, грн | Вартість, грн |
|---|---|---|---|
| Мова програмування Python та її бібліотеки, | 1 | 7499,00 | 7873,95 |
| Панель керування cPanel з підтримкою PHP, | 1 | 6820,00 | 7161,00 |
| Система керування реляційними базами даних MySQL | 1 | 8459,00 | 8881,95 |
| Доступ до мережі Internet (високошвидкісний) грн/місяць | 2 | 299,00 | 627,90 |
| Всього | | | 24544,80 |

### 4.3.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_в} \cdot \frac{t_{вик}}{12} , \qquad (4.15)$$

де $Ц_б$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$A_{обл} = (38699,00 \cdot 2) / (2 \cdot 12) = 3224,92$ *(грн).*

Проведені розрахунки наведено в таблиці 4.11.

Таблиця 4.11 – Амортизаційні відрахування по кожному виду обладнання

| Найменування обладнання | Балансова вартість, грн | Строк корисного використання, років | Термін використання обладнання, місяців | Амортизаційні відрахування, грн |
|---|---|---|---|---|
| Електронний комплекс аналітичної системи обробки даних | 38699,00 | 2 | 2 | 3224,92 |
| Персональний комп’ютер розробника інформаційної системи | 21299,00 | 2 | 2 | 1774,92 |
| Спеціалізоване робоче місце розробника інформаційної системи | 7500,00 | 5 | 2 | 250,00 |
| Пристрій виводу текстової інформації | 6520,00 | 4 | 2 | 271,67 |
| Оргтехніка | 7399,00 | 5 | 2 | 246,63 |
| Приміщення лабораторії досліджень | 450000,00 | 25 | 2 | 3000,00 |
| ОС Windows 11 | 6510,00 | 2 | 2 | 542,50 |
| Прикладний пакет Microsoft Office 2019 | 6490,00 | 2 | 2 | 540,83 |
| Всього | | | | 9851,47 |

### 4.3.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію ($B_e$) розраховуємо за формулою:

$$B_e = \sum_{i=1}^{n} \frac{W_{yi} \cdot t_i \cdot Ц_e \cdot K_{вni}}{\eta_i}, \quad (4.16)$$

де $W_{yi}$ – встановлена потужність обладнання на визначеному етапі розробки, кВт;

$t_i$ – тривалість роботи обладнання на етапі дослідження, год;

$Ц_e$ – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), приймемо $Ц_e = 7,50$ грн;

$K_{вni}$ – коефіцієнт, що враховує використання потужності, $K_{вni} < 1$;

$\eta_i$ – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$B_e = 0,36 \cdot 200,0 \cdot 7,50 \cdot 0,95 / 0,97 = 540,00$ *(грн).*

Проведені розрахунки наведено в таблиці 4.12.

Таблиця 4.12 – Витрати на електроенергію

| Найменування обладнання | Встановлена потужність, кВт | Тривалість роботи, год | Сума, грн |
|---|---|---|---|
| Електронний комплекс аналітичної системи обробки даних | 0,36 | 200,0 | 540,00 |
| Персональний комп'ютер розробника інформаційної системи | 0,25 | 200,0 | 375,00 |
| Спеціалізоване робоче місце розробника інформаційної системи | 0,10 | 200,0 | 150,00 |
| Пристрій виводу текстової інформації | 0,20 | 3,7 | 5,55 |
| Оргтехніка | 0,45 | 1,5 | 5,06 |
| Маршрутизатор VPN TP-Link ER7212PC | 0,05 | 95,0 | 35,63 |

Продовження таблиці 4.12.

| Найменування обладнання | Встановлена потужність, кВт | Тривалість роботи, год | Сума, грн |
|---|---|---|---|
| Серверне обладнання (локальне) на базі QUBE QB Ryzen 5 7600X RTX 4070 Ti 12GB W1 3210 (Ryzen57600XRTX4070Ti 12GBW13210) | 0,20 | 95,0 | 142,50 |
| Імітатор станції досліджень моніторингу поверхневих вод | 0,10 | 95,0 | 71,25 |
| Всього | | | 1324,99 |

### 4.3.9 Службові відрядження

До статті «Службові відрядження» дослідної роботи на тему «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов’язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з’їзди, конференції, наради, пов’язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» розраховуємо як 20…25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{св} = (З_о + З_р) \cdot \frac{Н_{св}}{100\%}, \qquad (4.17)$$

де $Н_{св}$ – норма нарахування за статтею «Службові відрядження», приймемо $Н_{св}$ = 20%.

*$В_{св}$ = (63063,64 + 3633,16) · 20 / 100% = 13339,36 (грн).*

### 4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» розраховуємо як 30…45% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{сп} = (З_о + З_р) \cdot \frac{Н_{сп}}{100\%}, \quad (4.18)$$

де $Н_{сп}$ – норма нарахування за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації», приймемо $Н_{сп}$= 30%.

*$B_{сп}$ = (63063,64 + 3633,16) · 30 / 100% = 20009,04 (грн).*

### 4.3.11 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50…100% від суми основної заробітної плати дослідників та робітників за формулою:

$$І_в = (З_о + З_р) \cdot \frac{Н_{ів}}{100\%}, \quad (4.19)$$

де $Н_{ів}$ – норма нарахування за статтею «Інші витрати», приймемо $Н_{ів}$ = 50%.

*$І_в$ = (63063,64 + 3633,16) · 50 / 100% = 33348,40 (грн).*

### 4.3.12 Накладні (загальновиробничі) витрати

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов’язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів;

витрати, пов’язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов’язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100…150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = (З_о + З_р) \cdot \frac{Н_{нзв}}{100\%}, \quad (4.20)$$

де $Н_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», приймемо $Н_{нзв}$ = 100%.

*$B_{нзв}$ = (63063,64 + 3633,16) · 100 / 100% = 66696,80 (грн).*

Витрати на проведення науково-дослідної роботи на тему «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = З_о + З_р + З_{дод} + З_н + М + К_в + B_{спец} + B_{прг} + A_{обл} + B_е + B_{св} + B_{сп} + I_в + B_{нзв}. \quad (4.21)$$

*$B_{заг}$= 63063,64 +3633,16 +7336,65 +16287,36 +3362,10 +6844,95 +104576,85 + 24544,80 + 9851,47 +1324,99 +13339,36 +20009,04 +33348,40 +66696,80 = =374219,55 грн.*

Загальні витрати *ЗВ* на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (4.22)$$

де $\eta$ - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, приймемо $\eta$=0,9.

*ЗВ = 374219,55 / 0,9 = 415799,50 (грн).*

## 4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

Результати дослідження проведені за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» передбачають комерціалізацію протягом 4-х років реалізації на ринку. За комерційним напрямом дослідження відносяться до ситуації 1 – «*Розробка інформаційної системи (web-сайт, консолідований ресурс тощо) на основі нових алгоритмів, програмних або технічних засобів»* [31].

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

$\Delta N$ – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються:

– Перший рік – 1000 осіб;
– Другий рік – 2500 осіб;
– Третій рік – 1800 осіб;
– Четвертий рік – 900 осіб.

$N$ – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, приймемо 18500 осіб;

$Ц_б$ – вартість послуги у році до впровадження інформаційної системи, приймемо 650,00 грн;

$\pm\Delta Ц_о$ – зміна вартості послуги від впровадження результатів, приймемо 84,72 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta П_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [32]:

$$\Delta П_i = (\pm \Delta Ц_o \cdot N + Ц_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\vartheta}{100}), \quad (4.23)$$

де $\lambda$ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2023 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0,8333$;

$\rho$ – коефіцієнт, який враховує рентабельність інноваційного продукту). Приймемо $\rho = 40\%$;

$\vartheta$ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році $\vartheta = 18\%$;

Збільшення чистого прибутку 1-го року:

$\Delta П_1 = (84,72 \cdot 18500,00 + 734,73 \cdot 1000) \cdot 0,83 \cdot 0,4 \cdot (1\text{-}0,18/100\%) = 626733,91$ (грн).

Збільшення чистого прибутку 2-го року:

$\Delta П_2 = (84,72 \cdot 18500,00 + 734,73 \cdot 3500) \cdot 0,83 \cdot 0,4 \cdot (1\text{-}0,18/100\%) = 1126787,75$ (грн).

Збільшення чистого прибутку 3-го року:

$\Delta П_3 = (84,72 \cdot 18500,00 + 734,73 \cdot 5300) \cdot 0,83 \cdot 0,4 \cdot (1\text{-}0,18/100\%) = 1486826,51$ (грн).

Збільшення чистого прибутку 4-го року:

$\Delta П_4 = (84,72 \cdot 18500,00 + 734,73 \cdot 6200) \cdot 0,83 \cdot 0,4 \cdot (1\text{-}0,18/100\%) = 1666845,89$ (грн).

Приведена вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \quad (4.24)$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

$T$ – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

$\tau$ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau$=0,1;

$t$ – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$ПП$ =626733,91/(1+0,1)[1]+1126787,75/(1+0,1)[2]+1486826,51/(1+0,1)[3]+ +1666845,89/(1+0,1)[4]=569758,10+931229,54+1117074,76+1138478,17=3756540 ,58 (грн).

Величина початкових інвестицій $PV$, які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{інв} \cdot ЗВ, \tag{4.25}$$

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{інв}$=2;

$ЗВ$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 415799,50 грн.

$PV = k_{інв} \cdot ЗВ$ = 2 · 415799,50 = 831599,00 (грн).

Абсолютний економічний ефект $E_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV \tag{4.26}$$

де $ПП$ – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 3756540,58 грн;

$PV$ – теперішня вартість початкових інвестицій, 831599,00 грн.

$E_{абс} = ПП - PV$ = 3756540,58 - 831599,00 = 2924941,58 (грн).

Внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.27)$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, 2924941,58 грн;

$PV$ – теперішня вартість початкових інвестицій, 831599,00 грн;

$T_ж$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

$$E_в = \sqrt[Tж]{1 + \frac{E_{абс}}{PV}} - 1 = (1+2924941,58/831599,00)^{1/4} = 0,46.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{мін}$ :

$$\tau_{мін} = d + f, \quad (4.28)$$

де $d$ – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні $d$ =0,09;

$f$ – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,25.

$\tau_{мін}$ = 0,09+0,25 = 0,34 < 0,46 свідчить про те, що внутрішня економічна дохідність інвестицій $E_в$, які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_в}, \tag{4.29}$$

де $E_в$ – внутрішня економічна дохідність вкладених інвестицій.

$T_{ок}$ = 1 / 0,46 = 2,18 роки.

$T_{ок}$ < 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

### 4.5 Висновки

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг» становить 41,0 бала, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

При оцінюванні рівня конкурентоспроможності, згідно узагальненого коефіцієнту конкурентоспроможності розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,93 рази.

Також термін окупності становить 2,18 роки, що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже, можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг».

# ВИСНОВКИ

В ході магістерської кваліфікаційної роботи проведено аналіз предметної області, було оглянуто та обрано оптимальні технології для розроблення інформаційної системи аналізу та прогнозування якості води в річці Південний Буг, проведено обробку, аналіз та прогнозування даних, реалізовано базу даних, веб-інтерфейс, ГІС технологію, імпортовано результати аналізу та прогнозування систему, проведено тестування розробленої інформаційної системи.

Проблема аналізу та прогнозування якості води є вкрай важливим питанням для людини, адже саме вода є одним з необхідних компонентів для підтримки життя всіх живих організмів, в тому числі і людини, а запаси питної води є вкрай обмеженими, а через підвищення рівня забрудненості водних об’єктів у світі, дана проблема відчувається вкрай гостро. Саме тому розроблено інформаційну систему зі зрозумілим інтерфейсом та набором функцій для аналізу та прогнозування якості води у річці Південний Буг з метою подальшого аналізу водних ресурсів, та ,за необхідності, розроблені рішень для розв’язання проблеми забруднення поверхневих вод.

Здійснено аналіз предметної області, було визначено основні показники що характеризуються якість води, розглянуто нормування показників, технології збору даних про якість води. Було розглянуто існуючі аналоги а також описано потенційні проблеми під час розроблення інформаційної системи.

Проведено огляд та вибір оптимальних інформаційних технологій, оглянуто вхідні дані, проведено обробку даних, а саме об’єднання даних з різних джерел в один набір даних, проведено розвідувальний аналіз, обрано бібліотеки та моделі для реалізації аналізу та прогнозування.

Розроблено інформаційну систему аналізу та прогнозування, побудовано алгоритм роботи технології, спроектовано схему бази даних, та веб-системи, реалізовано ГІС в системі. Було імпортовано вихідні дані аналізу

та прогнозування в систему, тим самим об’єднавши всі розглянуті технології в одне інформаційне рішення. Було проведено тестування спроектованої системи, за результатами якого, найточнішою моделлю для обраних параметрів виявилась RandomForestRegressor з точністю 0.7 для тренувальних даних та 0.63 для валідаційних даних. Загалом інформаційна система показала себе як самостійне та конкурентоспроможне рішення в питанні аналізу та прогнозуванні якості води в річці Південний Буг.

Економічна частина даної роботи містить розрахунок рівня комерційного потенціалу розробку нового програмного продукту, що становить 41 бал. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат. Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розраховано період окупності витрат для інвестора та економічний ефект при використанні даної розробки. При оцінюванні рівня конкурентоспроможності, згідно узагальненого коефіцієнту конкурентоспроможності розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,93 рази. Термін окупності складе становить 2,18 років, що менше 3 років, що свідчить про комерційну привабливість науково-технічної розробки.

За результатами даної роботи опубліковано тези на конференції ВНТУ, а саме LIII Науково-технічній конференції факультету інтелектуальних інформаційних технологій та автоматизації (м. Вінниця, 2023-2024 рр.), на тему «Аналіз та прогнозування якості води в річці Південний Буг на основі даних моніторингу за 2000 – 2023 роки» [1].

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Крижановський Є. М., Марецький Д.Є. Аналіз та прогнозування якості води в річці Південний Буг на основі даних моніторингу за 2000 – 2023 роки. *LIII Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024)*. Вінниця, 2024. URL: https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2024/paper/view/19699/16297 (дата звернення 11.12.2023)

2. How Much Water is There on Earth? Water Science School, 2019. URL: https://www.usgs.gov/special-topics/water-science-school/science/how-much-water-there-earth (дата звернення 03.09.2023).

3. Місія та стратегія державного агентства водних ресурсів України, 2020. URL: https://davr.gov.ua/misiya-ta-strategiya (дата звернення 05.09.2023)

4. Greenpeace [UPSC Notes on Environment & Ecology], 2023. URL: https://byjus.com/free-ias-prep/greenpeace/ (дата звернення 10.09.2023)

5. UN-Water. History and Purposes, 2023. URL: https://www.unwater.org/about-un-water (дата звернення 12.09.2023)

6. Household Chemicals and the Water Environment, 2012. URL: http://www.fwr.org/environw/frr0010.htm (дата звернення 12.09.2023)

7. Water Pollution: Everything You Need to Know, 2018. URL: https://www.nrdc.org/stories/water-pollution-everything-you-need-know#whatis (дата звернення 13.09.2023)

8. Кофанов В.І., Огняник М.С. Нормативно–методичне забезпечення визначення якості води при оцінці впливу на навколишнє середовище. *Наукові праці Інституту геологічних наук НАН України*. Вінниця, 2008. URL: http://dspace.nbuv.gov.ua/bitstream/handle/123456789/5595/02-Kofanov.pdf?sequence=1 (дата звернення 15.09.2023)

9. Review on water quality monitoring technologies, 2020. URL: https://www.researchgate.net/publication/341795626_Review_on_water_quality_monitoring_technologies (дата звернення 15.09.2023)

10. Сільбершац А., Корт Ф.Г., Сударшан С., Концепції баз даних. Нью-Йорк: Макгроу Хіл, 2019. 1376 с.

11. Relational Database Benefits and Limitations (Advantages & Disadvantages), 2021. URL: https://databasetown.com/relational-database-benefits-and-limitations/ (дата звернення 21.09.2023)

12. Best Relational Databases, 2023. URL: https://www.g2.com/categories/relational-databases (дата звернення 21.09.2023)

13. Ботрос С., Тінлі Д. MySQL по максимум. Київ: Print2Print, 2023, 432 с.

14. MySQL Advantages and Disadvantages, 2023. URL: https://www.w3schools.blog/mysql-advantages-disadvantages (дата звернення 21.09.2023)

15. 30 Best GIS Software Applications [Rankings], 2023. URL: https://gisgeography.com/best-gis-software/ (дата звернення 22.09.2023)

16. What is OpenStreetMap, 2020. URL: https://subscription.packtpub.com/book/programming/9781847197504/1/ch01lvl1sec06/what-is-openstreetmap (дата звернення 22.09.2023)

17. OpenStreetMap and its use as open data, 2018. URL: https://www.e-education.psu.edu/geog585/node/738 (дата звернення 22.09.2023)

18. What is Python? 2020. URL: https://opensource.com/resources/python (дата звернення 23.09.2023)

19. What is R: Overview, its Applications and what is R used for? 2023. URL: https://www.simplilearn.com/what-is-r-article (дата звернення 23.09.2023)

20. NumPy: the absolute basics for beginners, 2023. URL: https://numpy.org/doc/stable/user/absolute_beginners.html (дата звернення 23.09.2023)

21. Pandas - Python Data Analysis Library, 2023. URL: https://pandas.pydata.org/ (дата звернення 23.09.2023)

22. Matplotlib: Visualization with Python, 2012. URL: https://matplotlib.org/ (дата звернення 23.09.2023)

23. Scikit Learn: Discover the Python library dedicated to Machine Learning, 2023. URL: https://datascientest.com/en/scikit-learn-discover-the-python-library-dedicated-to-machine-learning (дата звернення 23.09.2023)

24. Introduction to XGBoost Algorithm in Machine Learning, 2023. URL: https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/ (дата звернення 24.09.2023)

25. GLOB Module: File Searching In Python, 2022. URL: https://pythonology.eu/glob-module-file-searching-in-python/ (дата звернення 24.09.2023)

26. Working with warnings in Python, 2020. URL: https://lerner.co.il/2020/04/27/working-with-warnings-in-python/ (дата звернення 25.09.2023)

27. 12 Best Languages for Web Development in 2023, 2023. URL: https://www.browserstack.com/guide/best-language-for-web-development (дата звернення 25.09.2023)

28. Васильєв О.В. Програмування мовою PHP. Дніпро: Ліра-К, 2023. 368 с.

29. PyScript: Python in the Browser, 2023. URL: https://anaconda.cloud/pyscript-python-in-the-browser (дата звернення 27.09.2023)

30. An Introduction to cPanel (Control Panel) Web Hosting, 2022. URL: https://www.linkedin.com/pulse/introduction-cpanel-control-panel-web-hosting-vibhati-sharma-1f (дата звернення 29.09.2023)

31. Козловський В. О., Лесько О. Й., Кавецький В. В. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт. Вінниця : ВНТУ, 2021. 42 с.

32. Кавецький В.В. Економічне обґрунтування інноваційних рішень: практикум / за ред. В. В. Кавецький, В. О. Козловський, І. В. Причепа. Вінниця : ВНТУ, 2016. 113 с.

Додаток А

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ
Завідувач кафедри САІТ
_____ д.т.н., проф. Віталій МОКІН
«___» _____ 2023 р.

ТЕХНІЧНЕ ЗАВДАННЯ
на магістерську кваліфікаційну роботу
«ІНФОРМАЦІЙНА СИСТЕМА АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ В РІЧЦІ ПІВДЕНИИЙ БУГ»
08-34.МКР.012.00.000.ТЗ

Керівник: к.т.н., доц.
_____ Євгеній КРИЖАНОВСЬКИЙ
«___» _____ 2023 р.

Розробив: студент гр. 2ІСТ-22м
_____ Денис МАРЕЦЬКИЙ
«___» _____ 2023 р.

Вінниця 2023

1. Підстава для проведення робіт

Підставою для виконання роботи є наказ № __ по ВНТУ від «__» _____ 2023 р., та індивідуальне завдання на МКР, затверджене протоколом № __ засідання кафедри САІТ від «__» _____ 2023 р.

2. Джерела розробки:

- Mokin V. B. River Water Quality EDA and Forecasting – версія датасету – 2021 р.. URL: https://www.kaggle.com/datasets/vbmokin/wq-southern-bug-river-01052021.
- Дані державного моніторингу поверхневих вод. URL: https://data.gov.ua/dataset/surface-water-monitoring

3. Мета і призначення роботи:

Розроблення інформаційної системи аналізу та прогнозування якості води у річці Південний Буг.

4. Вихідні дані для проведення робіт:

Датасет Kaggle «River Water Quality EDA and Forecasting» з даними моніторингу води в річці Південний Буг за 2000 – 2021 роки в сукупності з даними моніторингу за 2021 – 2023 роки отримані в результаті обробки даних Державного агентства водних ресурсів.

5.Методи дослідження:

- розвідувальний аналіз;
- прогнозування даних;
- побудова баз даних;
- побудова веб-систем;
- геоінформаційні технології.

6. Етапи роботи і терміни їх виконання:

1. Аналіз предметної області _____ – _____
2. Вибір оптимальних технологій та проведення розвідувального аналізу _____ – _____
3. Розроблення інформаційної системи _____ – _____
4. Тестування системи _____ – _____
5. Економічна частина _____ – _____
6. Оформлення матеріалів до захисту МКР. _____ – _____

7. Очікувані результати та порядок реалізації:

Готова інформаційна система з реалізацією технологій аналізу та прогнозування даних, геоінформаційних технологій, а також технологій проектування бази даних, у вигляді веб-додатку.

8. Вимоги до розробленої документації

Пояснювальна записка оформлена у відповідності до вимог «Методичних вказівок до виконання магістерських кваліфікаційних робіт для студентів спеціальності 126 «Інформаційні системи та технології» (освітня програма «Інформаційні технології аналізу даних та зображень»).

9. Порядок приймання роботи

Публічний захист «____» _____ 2023 р.
Початок розробки «____» _____ 2023 р.
Граничні терміни виконання МКР «____» _____ 2023 р.

Розробив студент групи 2ІСТ-22м _____ Денис МАРЕЦЬКИЙ

Додаток Б

Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень

Назва роботи: «Інформаційна система аналізу та прогнозування якості води у річці Південний Буг»
Тип роботи: магістерська кваліфікаційна робота
Підрозділ: кафедра САІТ

**Показники звіту подібності Unicheck**

Оригінальність 94% Схожість 6%

Аналіз звіту подібності (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку  (підпис) Сергій ЖУКОВ

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи  (підпис) Денис МАРЕЦЬКИЙ

Керівник роботи  (підпис) Євгеній КРИЖАНОВСЬКИЙ

Додаток В

Лістинг програми

```
# Work with Data - the main Python libraries
import numpy as np
import pandas as pd
import pandas_profiling as pp
import glob
# Visualization
import matplotlib.pyplot as plt

# Preprocessing
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split, KFold, ShuffleSplit,
GridSearchCV

# Modeling
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
import xgboost as xgb
from xgboost.sklearn import XGBRegressor
from sklearn.neighbors import KNeighborsRegressor
import lightgbm as lgb

# Metrics
from sklearn.metrics import r2_score

import warnings
warnings.simplefilter('ignore')

# Path to CSV files
csv_files = glob.glob('/kaggle/input/pivdennyi-buh/*.csv')

data_about = pd.read_csv('../input/wq-southern-bug-river-
01052021/PB_stations.csv', sep=';', header=0, encoding='cp1251')

# Initialize an empty list to store DataFrames
dfs = []

# Specify the columns you want to keep
```

```
selected_columns = ['Post_ID', 'Post_Name', 'Controle_Date', 'BSK5', 'Zavisli',
'Kisen', 'Sulfat', 'Hlorid', 'Amoniy', 'Nitrat', 'Nitrit', 'Fosfat']

for csv_file in csv_files:
    # Assuming the CSV files have a header row
    current_data = pd.read_csv(csv_file, sep=';', header=0,
usecols=selected_columns)

    # Filter rows where 'Riverbas_Name' is equal to 'Південний Буг'
    current_data = current_data[current_data['Post_Name'].str.startswith('p.
Південний Буг')]

    dfs.append(current_data)

# Concatenate all DataFrames in the list
combined_data = pd.concat(dfs, ignore_index=True)

combined_data

other_data = pd.read_csv('../input/wq-southern-bug-river-
01052021/PB_All_2000_2021.csv', sep=';', header=0)

# Rename columns in combined_data
combined_data.rename(columns={
    'Post_ID': 'id',
    'Controle_Date': 'date',
    'Amoniy': 'NH4',
    'BSK5': 'BSK5',
    'Zavisli': 'Suspended',
    'Kisen': 'O2',
    'Nitrat': 'NO3',
    'Fosfat': 'PO4',
    'Hlorid': 'CL',
    'Nitrit': 'NO2',
    'Sulfat': 'SO4'
}, inplace=True)

# Drop 'Riverbas_Name' column
combined_data.drop(columns=['Post_Name'], inplace=True)

# Reorder columns in combined_data
```

```
combined_data = combined_data[['id', 'date', 'NH4', 'BSK5', 'Suspended', 'O2',
'NO3', 'NO2', 'SO4', 'PO4', 'CL']]
combined_data['date'] = pd.to_datetime(combined_data['date'],
errors='coerce').dt.strftime('%d.%m.%Y')

# Concatenate combined_data and other_data
final_dataframe = pd.concat([combined_data, other_data], ignore_index=True)

id_mapping = {
   27610: 5,
   27135: 16,
   27136: 15,
   27137: 14,
   27140: 10,
   27146: 3,
   27148: 22,
   38463: 9,
   27147: 2,
   27138: 13,
   38465: 6,
   38475: 15
}

final_dataframe['id'] = final_dataframe['id'].replace(id_mapping)
final_dataframe['date'] = pd.to_datetime(final_dataframe['date'],
format='%d.%m.%Y')

#Remove rows where values in all columns are null
final_dataframe = final_dataframe.dropna(subset=['NH4', 'BSK5', 'Suspended',
'O2', 'NO3', 'NO2', 'SO4', 'PO4', 'CL'], how='all')


final_dataframe = final_dataframe.sort_values(by='date')
# Display the final concatenated dataframe
final_dataframe

final_dataframe['ds'] = pd.to_datetime(final_dataframe['date'])
final_dataframe['year'] = final_dataframe['ds'].dt.year
final_dataframe.info()
```

Додаток Г

**ІЛЮСТРАТИВНА ЧАСТИНА**

ІНФОРМАЦІЙНА СИСТЕМА АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ЯКОСТІ ВОДИ У РІЧЦІ ПІВДЕННИЙ БУГ

Нормоконтроль: к.т.н., доцент
_____ Сергій ЖУКОВ
«___» _____ 2023 р.

Вінниця 2023

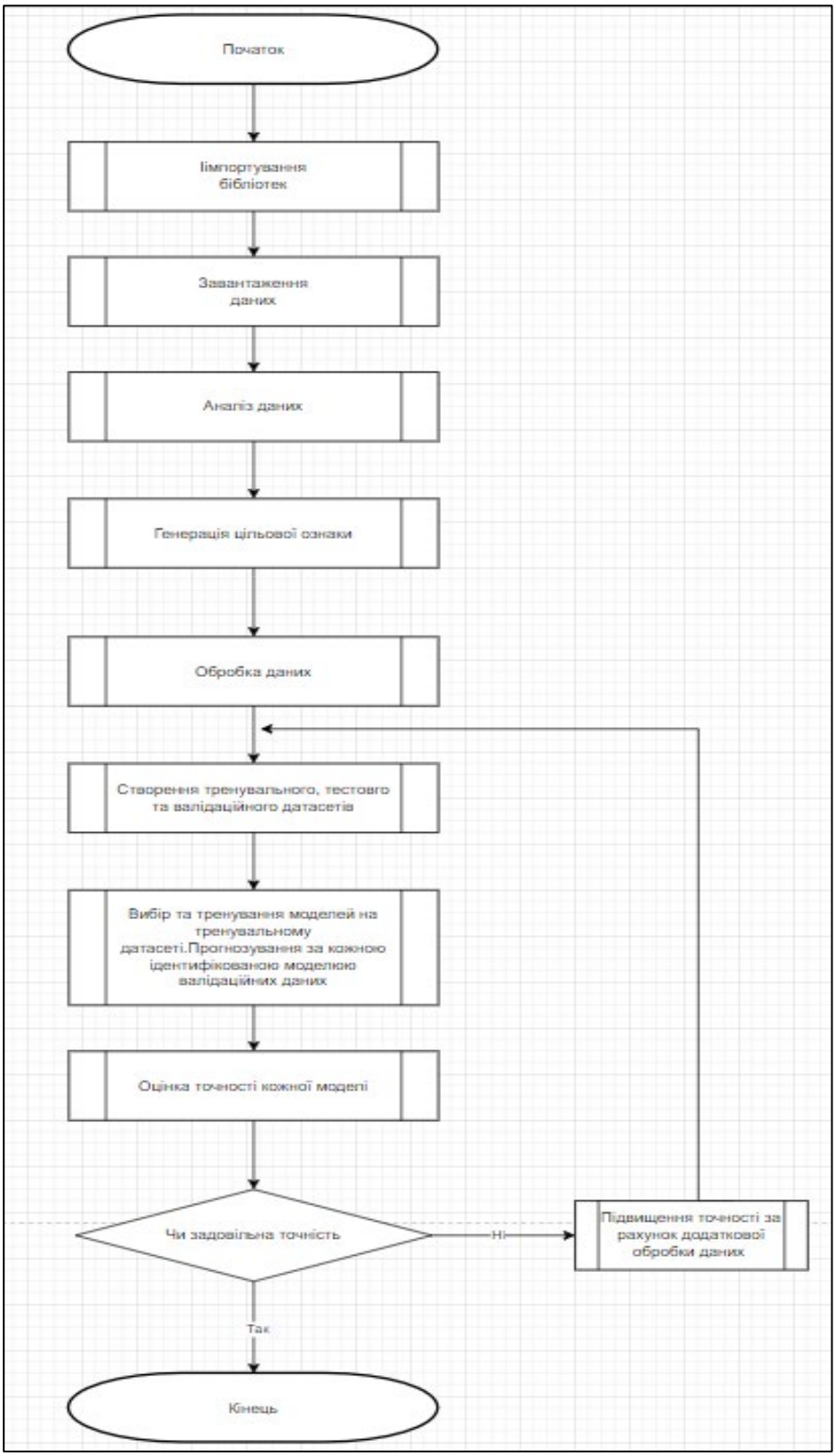



Рисунок Г.1 – Алгоритм блоку аналізу та прогнозування даних

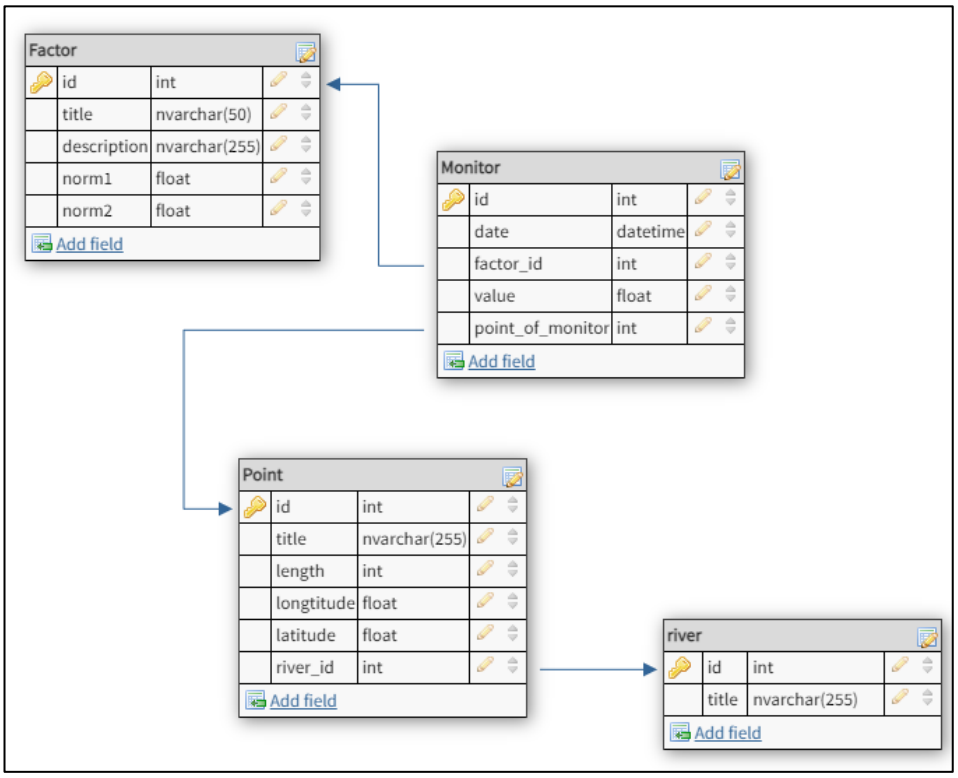



Рисунок Г.2 – Схема бази даних розробленої системи

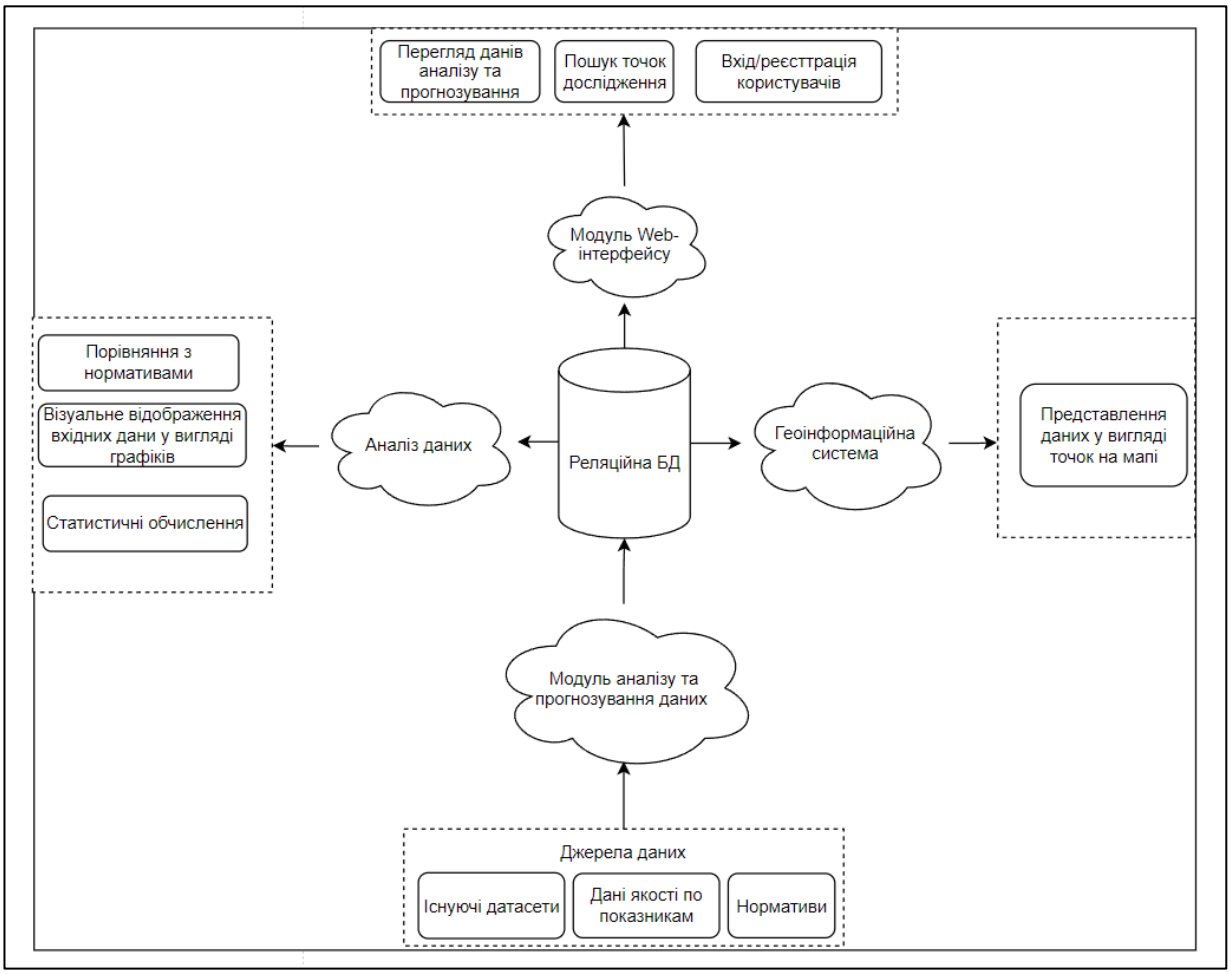



Рисунок Г.3 – Загальна структура системи

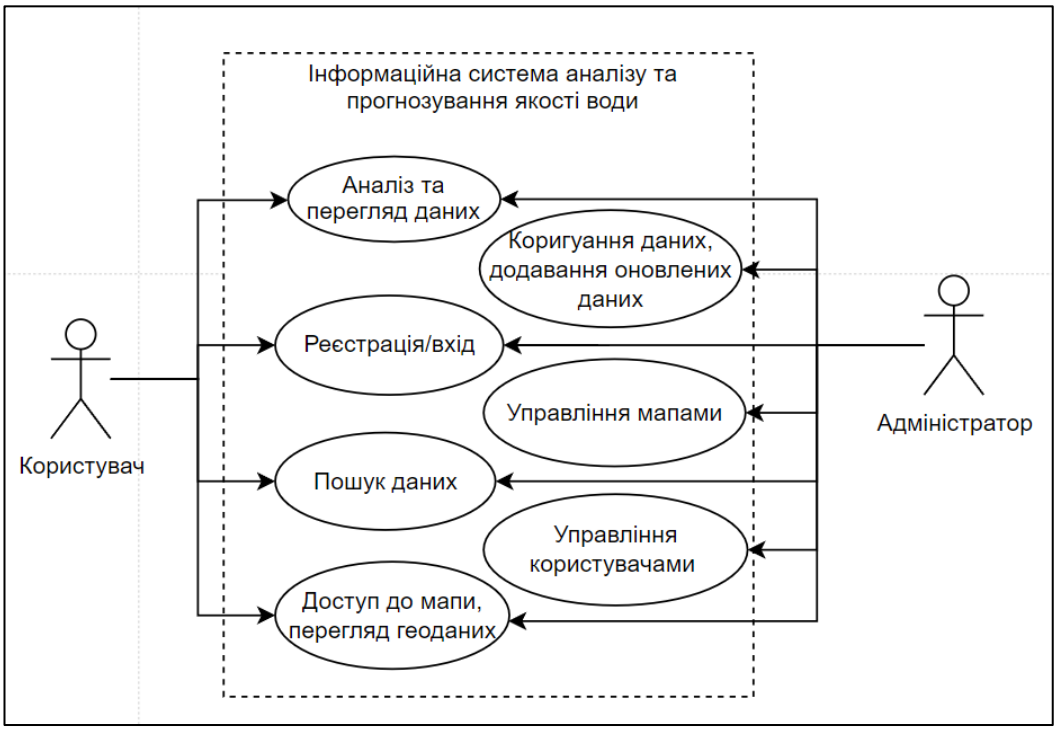



Рисунок Г.4 – Діаграма варіантів використання розробленої системи

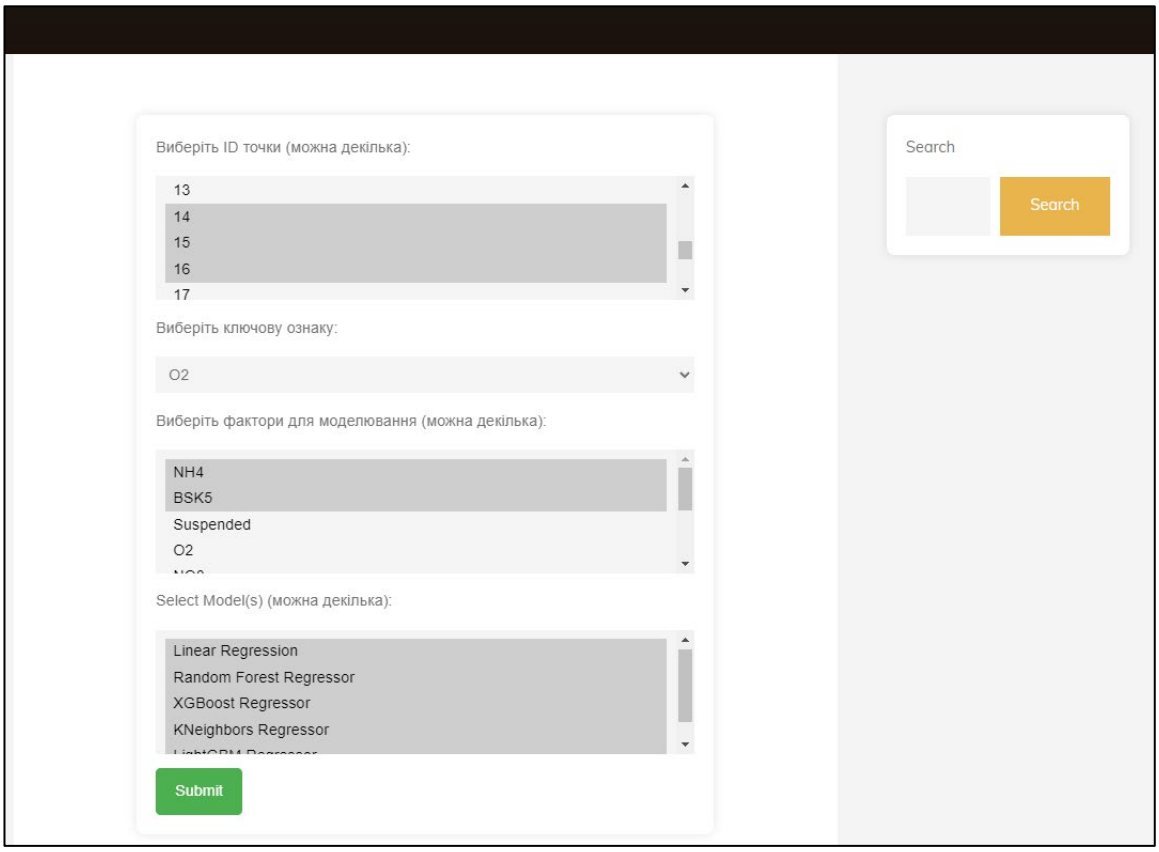



Рисунок Г.5 – Приклад застосування блоку прогнозування даних в системі

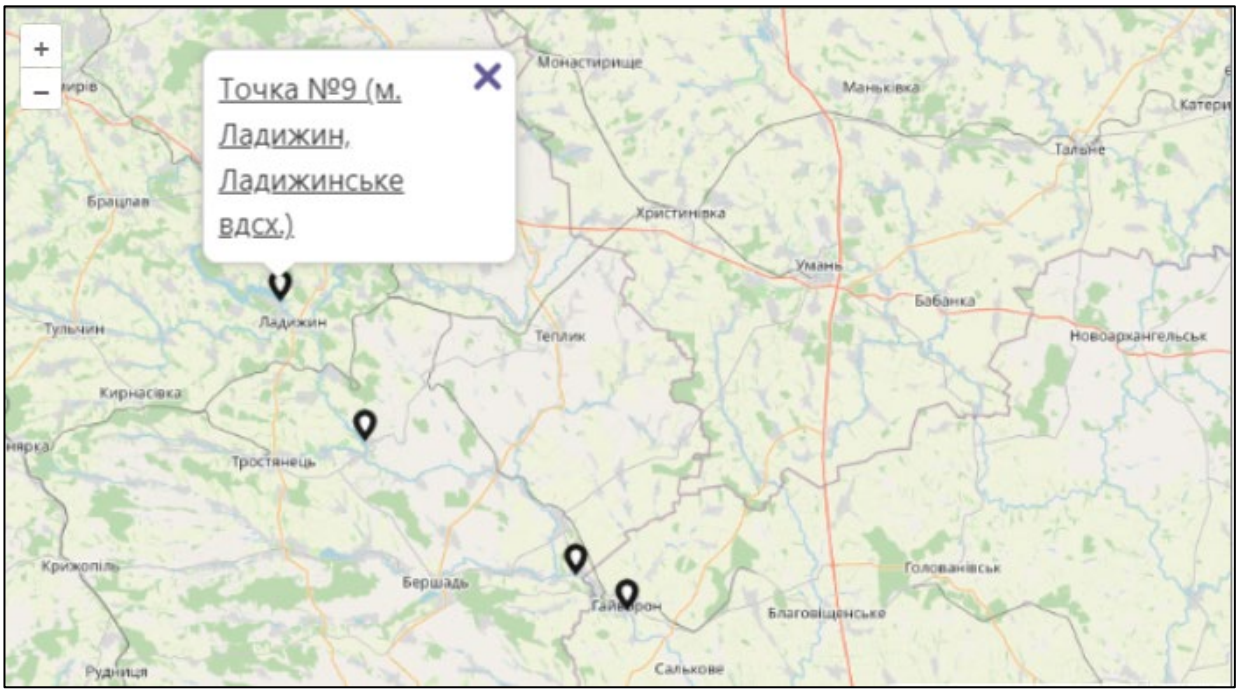



Рисунок Г.6 – Приклад мапи в розробленій системі

**Точність моделей для обраних моделей**

| model | Модель | train_score | valid_score |
|---|---|---|---|
| 1 | Random Forest Regressor | 0.7 | 0.63 |
| 0 | Linear Regression | 0.66 | 0.58 |
| 2 | XGBoost Regressor | 0.99 | 0.70 |
| 4 | LightGBM Regressor | 0.66 | 0.58 |
| 3 | KNeighbors Regressor | 1.00 | 0.59 |

Рисунок Г.7 – Результат тестування системи у вигляді розрахунку r2_score для обраних моделей

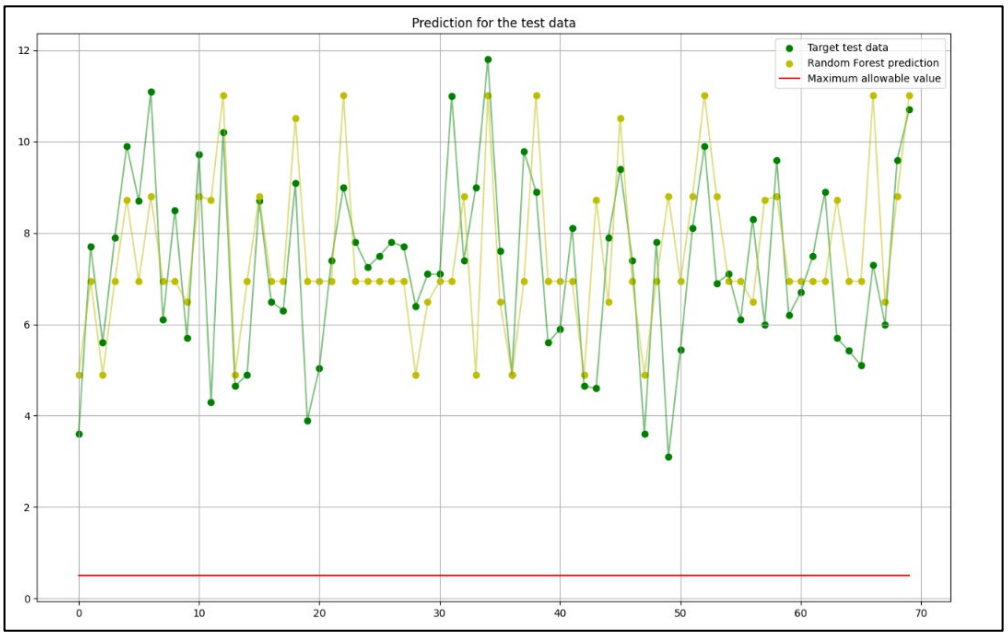



Рисунок Г.8 – Графік прогнозування моделі RandomForestRegressor для тестових даних


Рисунок Г.9 – Графік прогнозування моделі RandomForestRegressor для валідаційних даних

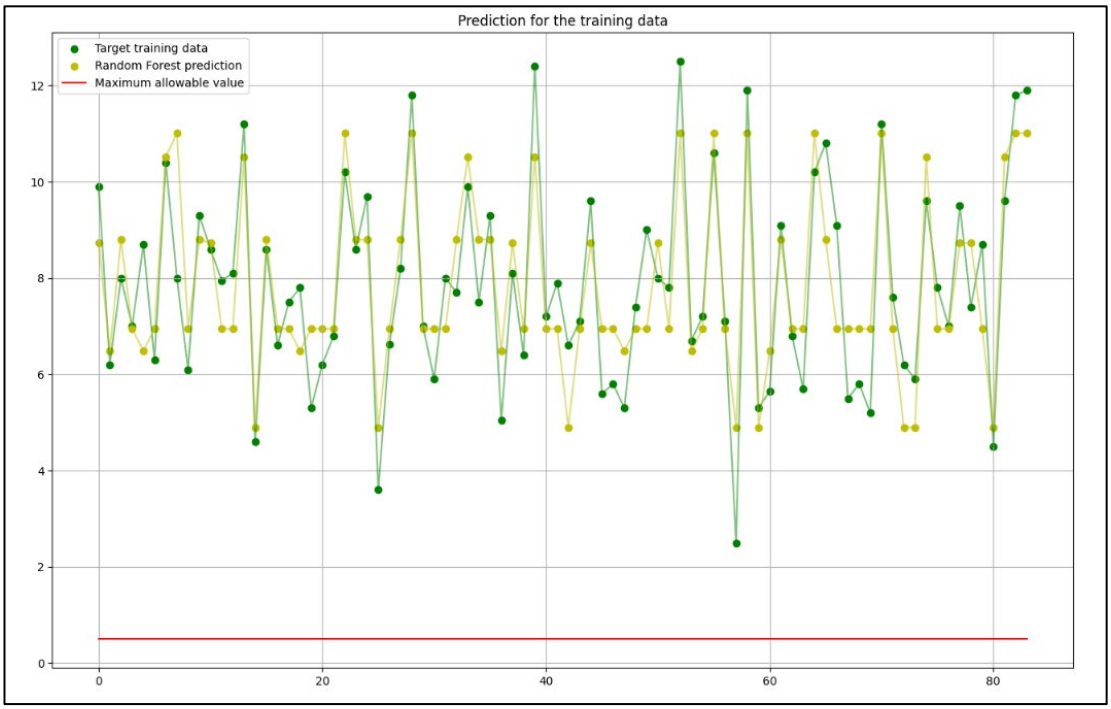



Рисунок Г.10 – Графік прогнозування моделі RandomForestRegressor для тренувальних даних