

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ**  
**ТЕХНОЛОГІЙ**

*Кваліфікаційна наукова праця  
на правах рукопису*

**ГЕРЦЮК МИКОЛА МОДЕСТОВИЧ**

УДК 004:[628.191:628.394]:355:271

**ДИСЕРТАЦІЯ**

**ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПРОГНОЗУВАННЯ ЗАБРУДНЕННЯ**  
**РІЧОК В УМОВАХ НАДЗВИЧАЙНИХ СИТУАЦІЙ НА ОСНОВІ МЕТОДІВ**  
**РЕГРЕСІЇ ТА ЕВРИСТИЧНИХ АЛГОРИТМІВ**

123 «Комп'ютерна інженерія»

12 «Інформаційні технології»

Подається на здобуття ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей,  
результатів і текстів інших авторів мають посилання на відповідне джерело

М.М.Герцюк

(підпис, ініціали та прізвище здобувача)

Науковий керівник

**Жебка Вікторія Вікторівна**

доктор технічних наук, професор

Київ 2024

## АНОТАЦІЯ

Герцюк М.М. Інформаційна технологія прогнозування забруднення річок в умовах надзвичайних ситуацій на основі методів регресії та класифікації. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії з галузі знань 12 «Інформаційні технології» за спеціальністю 123 «Комп'ютерна інженерія» - Державний університет інформаційно-комунікаційних технологій Міністерства освіти і науки України, Київ, 2024.

В дисертаційній роботі вирішено наукове завдання розробки інформаційної технології, що прогнозує рівень забруднення річок небезпечними речовинами в умовах надзвичайних ситуацій.

Робота висвітлює проведені дослідження, що дозволяють здійснити прогнозування забруднення русла річок при обмеженій кількості вхідних характеристик.

У роботі проаналізовано такі існуючі рішення, як Qual2K, MONERIS, Aquatox, WASP8, EXAMS, Visual Plumes, RUSLE2, MMSOILS, RESRAD, MEPAS, Hydrologic Micro Services, SWToolbox, MOIRA DSS та деякі емпіричні моделі. Аналіз дозволяє провести прогнозування наслідків забруднення річок та вивести такі недоліки цих систем, як:

- незручний користувацький інтерфейс, що включає в себе складність аналізу отриманих результатів;
- необхідність залучення широкого спектру характеристик, необхідних для проведення прогнозування;
- недостатню адаптованість до умов конкретної надзвичайної ситуації;
- необхідність мати широку експертизу в моделях, щоб застосовувати їх правильно;
- відсутність користувацького інтерфейсу.

В роботі проаналізовані закони Фіка, як закони моделювання процесу дифузії. Особлива увага приділяється рівнянню Фіка виведеним Тейлором, що здійснює обчислення в потоці. Важливість такого аналізу полягає в розумінні

процесів дифузії та логіки роботи методів типу Фіка, тобто методів прогнозування наслідків забруднення, що моделюють тільки поширення дифузії, не включаючи інші фактори навколишнього середовища, які описують моделі не Фікового типу.

Для усунення таких недоліків здійснено ряд досліджень, що дадуть змогу усунути їх. Для цього, вибрано та удосконалено емпіричний метод проведення наслідків забруднення. Результатом методу є концентрація речовини з деякому місці, в деякий час. Такі характеристики усувають недолік складності аналізу отриманих результатів. Враховуючи такий недолік, як необхідність залучення широкого спектру характеристик, необхідних для проведення прогнозування прийнято рішення використовувати серію емпіричних гідрологічних рівнянь Харві Джобсона, як модель, що не вимагає деталізованих характеристик та може бути отримана шляхом ГІС аналізу та деяких статистичних даних.

Вибраний метод здійснює прогнозування між точкою забруднення та точкою вимірювання. Це є недоліком, що унеможлиблює використання методу в необхідних умовах. Тому, розроблено такі базові удосконалення, як метод багатоточковості та метод мультимаршрутизації. Метою методу багатоточковості є адаптація вибраної моделі до використання між декількома точками. Метод мультимаршрутизації передбачає використання моделі у випадку таких явищ, як розгалуження та з'єднання русла річки.

Частину методу прогнозування наслідків забруднень можна використати для прогнозування рівня спадання забруднення в деякій локації. В роботі проведені та описані дослідження, що показують логіку роботи цієї частини.

Метод прогнозування забруднення, що вимагає базової кількості характеристик може мати меншу точність та недостатню адаптованість до умов надзвичайної ситуації, тому на основі нейронної мережі розроблено метод коригування наслідків забруднення, що навчається на основі існуючої інформації про попередні забруднення. Навчання відбувається шляхом регресії обчислених даних мережею до існуючих статистичних. В основі такої нейронної мережі лежить задача регресії. Вхідною характеристикою до обчислення є концентрація речовини в деякій точці. Даний метод допомагає усунути такий недолік, як недостатня

адаптованість до умов конкретної надзвичайної ситуації та адаптувати результати до деяких можливих закономірностей локації з ефективністю, залежно від самої локації та інформації, що зібрана для даної локації.

Метод прогнозування наслідків забруднення є точковим, тобто використовує інформацію з точок моніторингу, що розташовані по руслу річки. Відповідно, результати обчислення доступні лише в цих локаціях. Тому, розроблено метод, що дає змогу отримувати результати між точками моніторингу. Базою для цього методу є методи інтерполяції та рекурсії. Такий підхід дозволяє проводити обчислення з масштабуванням відстані, таким чином формуючи градієнт результатів. Вхідними параметром до методу є концентрації в двох сусідніх точках моніторингу та дистанція між точками. Отже, така інформація доступна, з результату та вхідної характеристики до основного методу.

Інформація про можливі підприємства, що могли здійснити забруднення може бути корисною для ліквідації наслідків надзвичайної ситуації. Тому, на основі алгоритмів фільтрації та сортування розроблено метод, що дозволяє класифікувати такі підприємства для надзвичайної ситуації. У якості вхідного параметру такого методу є точка моніторингу на річці, біля якої стався викид, а отже ця інформація доступна, як вхідна характеристика до методу прогнозування забруднення.

Важливою інформацією є розуміння рівня небезпеки для деякого рівня концентрації. Тому, в роботі описується логіка оцінки шляхом порівняння значень концентрації з граничними значеннями бази даних потенційно-небезпечних об'єктів. Таку базу формують відповідні нормативно-правові документи, а саме санітарні правила і норми охорони поверхневих вод від забруднення - СанПіН 4630-88. Через громіздкість такого документу, база даних наповнюється тільки можливими для деякої річки речовинами.

Всі описані методи потребують бази даних, застосунку для здійснення обчислень, застосунку для наповнення бази даних та застосунку для проведення навчання. База даних необхідна для збереження інформації для роботи всіх існуючих методів. Застосунок для здійснення обчислень має використовуватись сторонами, що потребують в проведенні досліджень надзвичайної ситуації. Такими



можуть бути співробітники розрахунково-аналітичних груп Державної служби з надзвичайних ситуацій України. Наявність такого застосунку усуває такий недолік, як відсутність користувацького інтерфейсу та незручність користувацького інтерфейсу, що включає в себе складність аналізу отриманих результатів. Застосунок для наповнення бази даних необхідний для актуалізації інформації в базі даних. Такий процес може проводитись співробітниками Метеорологічної служби України, та/або Екологічної інспекції України. Застосунок для проведення навчання є технічним застосунком, що дозволяє актуалізувати метод коригування наслідків забруднення та проводитись, як автоматизована технічна процедура. Такий ряд вимог сформував необхідність розробити інформаційну систему для поєднання усіх методик та автоматизованих вимог. Єдиною додатковою вимогою є наявність серверу, як служби, що забезпечує всі складові системи цілісною інформацією.

Інформація про поточні витрати води в точках моніторингу є необхідною складовою, що повинна бути врахована при проведенні обчислення. З цією метою розроблено теоретичний метод інтеграції датчиків обчислення поточних витрат води, що знаходяться у точках моніторингу, з інформаційною системою. Його принцип полягає у взаємодії точки моніторингу з сервером через протокол HTTP, таким чином включаючи і цю частину взаємодії з пристроями до інформаційної системи.

Разом, усі описані складові, тобто методики та інформаційна система формують інформаційну технологію.

Валідність розроблених методів підтверджується шляхом проведення симуляцій та регресії очікуваних даних з обчисленими.

Правильність роботи інформаційної системи підтверджується тестуванням її компонентів. З цією метою, для кожного компонента розроблено ряд тестових ситуацій, що при правильному результаті підтверджують правильність роботи функції. Такі тести можуть бути проведеними вручну або автоматизованими.

Описані розробки усувають недоліки, що виявлені в ході аналізу існуючого програмного забезпечення та формують такі переваги, як наявність бази даних, та

інтерактивний користувацький інтерфейс. Інші такі переваги, як широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів, широкий спектр моделей для проведення прогнозування, можливість здійснювати моделювання не тільки якості води, а й вплив на водні екосистеми не є важливими для проведення обчислень в умовах надзвичайної ситуації.

**Ключові слова:** оцінка якості водойм, нейронна мережа, штучний інтелект, русло річки, регресія, прогнозування наслідків забруднення водойм, математичне моделювання, оцінка якості водних об'єктів, русло, прогноз наслідків забруднення водойм, математичне моделювання, інтерполяція, рекурсія, потенційно небезпечний об'єкт, алгоритм фільтрації, інформаційна система, інформаційна технологія, RESit, автоматизовані методи прогнозування забруднення, надзвичайна ситуація.

## SUMMARY

Gertsyuk M.M. Information technology for river pollution forecasting in emergencies based on regression and classification methods. Qualifying scientific work on manuscript rights.

The dissertation for obtaining Philosophy Doctor scientific degree in the field of knowledge 12 “Information technologies”, specialty 123 “Computer engineering”. State University of Information and Communication Technologies of the Ministry of Education and Science of Ukraine, Kyiv, 2024.

An information technology development that predicts rivers pollution level with hazardous substances in emergencies scientific task is solved in the dissertation.

The work highlights conducted research that allows riverbed pollution forecasting with a limited input characteristics number.

The work analyzed such existing solutions as Qual2K, MONERIS, Aquatox, WASP8, EXAMS, Visual Plumes, RUSLE2, MMSOILS, RESRAD, MEPAS, Hydrologic Micro Services, SWToolbox, MOIRA DSS and some empirical models. An

analysis makes it possible to forecast river pollution consequences and to deduce such disadvantages of mentioned systems as:

- inconvenient user interface, that includes the obtained results analyzing complexity;
- a need to involve a characteristics wide range necessary for forecasting;
- insufficient adaptability to the specific emergency conditions,
- a necessity to have extensive expertise in models to apply them correctly;
- user interface absence.

Fick's laws are analyzed in work as modeling the diffusion process laws. Special attention is allocated to the Fick equation derived by Taylor, which performs calculations in the flow. Such analysis is important cause of diffusion processes understanding and Fickian-type methods logic, means pollution effect predicting methods, that model only diffusion spread, without including other environmental factors that describe non-Fickian type models.

To eliminate such disadvantages, a number of studies were carried out, which will make it possible to eliminate them. For this purpose, a pollution effect assessing empirical method are selected and improved. A method result is substance concentration from a certain location, at certain time. Such characteristics eliminate obtained results analysis complexity disadvantage. Considering a need to involve a wide range of characteristics necessary for forecasting, a decision to use Harvey Jobson empirical hydrological equations series as a model that does not require detailed characteristics and can be obtained through GIS analysis and some statistical data was made.

Chosen method performs predictions between pollution and measurement points. This is a disadvantage that makes it impossible to use method in necessary conditions. Therefore, such basic improvements as the multipoint method and the multirouting method were developed. The purpose of the multipoint method is to adapt selected model to use between several points. A multirouting method involves model usage in the case of such phenomena as branching and merging of river channels.

A part of pollution effects forecasting method can be used to predict pollution level decline in some location. Researches that show work logic of this part are conducted and described in the work.

A pollution forecasting method, that requires a basic number of characteristics, may has less accuracy and insufficient adaptability to the emergency conditions. Therefore, a method for adjusting pollution consequences, which is learned based on existing information about previous pollution, was developed based on a neural network. Training is performed based on regressing of concentration data calculated by the network to the existing statistical ones. Such neural network basis is based on regression problem. An input characteristic for the calculation is the substance concentration at some point. This method helps to eliminate such a disadvantage as insufficient adaptation to specific emergency conditions and to adapt the results to some possible regularities of the location, with efficiency depending on the location itself and the information collected for the given location.

A pollution consequences forecasting method is point-based, means it uses information from monitoring points located along the riverbed. Accordingly, calculation results are available only in these locations. Therefore, a method that allows obtaining results between monitoring points was developed. A basis for this method is interpolation and recursion methods. This approach allows calculations with distance scaling, thus forming a results gradient. Input parameters to the method are concentrations in two neighboring monitoring points and distance between the points. So, such information is available, from result and input characteristic from main method data.

Information about possible enterprises that could have caused pollution can be useful for emergency consequences eliminating. Therefore, based on filtering and sorting algorithms, a method that allows classifying such enterprises for an emergency was developed. As an input parameter of such a method, there is a monitoring point on the river near which the release occurred, and therefore this information is available as an input characteristic to the pollution forecasting method.

Understanding the hazard level for a given concentration value is important information. Therefore, the work describes assessment logic by comparing concentration

values with potentially dangerous objects database limit values. Such a database is formed by relevant legal documents, namely sanitary rules and regulations for the surface water protection from pollution - SanPiN 4630-88. Due to the bulkiness of such a document, the database is filled only with substances possible for a certain riverbed.

All described methods require a database, an application for performing calculations, an application for filling the database, and an application for training. A database is necessary to store information for all existing methods operations. The calculation application should be used by parties requiring emergency investigations. Such parties can be State Emergency Service of Ukraine calculation and analytical group employees. Such an application presence eliminates such a disadvantage as the lack of a user interface and user interface inconvenience, that includes obtained results analyzing complexity. Application for database filling are necessary for updating database information. Such a process can be carried out by employees of the Meteorological Service of Ukraine and/or the Environmental Inspection of Ukraine. The training application is a technical application that allows you to update the method of pollution effect correcting and conduct it as an automated technical procedure. Such a series of requirements created a necessity to develop information system to combine all methods and automated requirements. The only additional requirement is the server as a service presence that provides all system components with complete information.

Information about current water consumption at monitoring points is a necessary component that must be taken into account during calculation process. For this purpose, a sensors integration theoretical method for calculating current water flows located at monitoring points with the information system was developed. Its principle lies in interaction between monitoring point and server through HTTP protocol, thus including this interaction part with the devices in the information system.

Together, all described components, i.e. methods and information system, form information technology.

A developed methods validation is confirmed by conducting simulations and expected with calculated data regression.

An information system operation correctness is confirmed by testing its components. For this purpose, a few test situations were developed for each component, which, if the result is correct, confirm operation correctness of the function. Such tests can be conducted manually or automated.

Described developments eliminate disadvantages revealed during existing software analysis and form such advantages as presence of a database and an interactive user interface. Other advantages such as a wide range of characteristics that can be considered for forecasting, which improves the forecasted results accuracy, models wide range for forecasting and the ability to model not only water quality but also the impact on aquatic ecosystems are not important for calculations in emergencies.

**Key words:** water body quality assessment, neural network, artificial intelligence, riverbed, regression, water body pollution consequences prediction, mathematical modeling, water bodies quality assessment, river bed, water body pollution effects prediction, mathematical modeling, interpolation, recursion, potentially dangerous object, filtering algorithm, information system, information technology, RESit, automated pollution forecasting methods, emergency situation.

## Список опублікованих праць за темою дисертації

### Матеріали й тези наукових конференцій

1. **Gertsyuk Mod.M., Skobley M.P., Gertsyuk Myk.M., Lysychenko G.V.** Chemically dangerous objects Zakarpattia region Ukraine. *Third international conference: «Chemical and radiation safety: problems and solutions». Works and messages* (Kyiv, 19-22 May 2015), 2015. P. 3.

2. **Герцюк Мод.М., Мельниченко Т.І., Герцюк Мик.М., Лисиченко Г.В.** Дослідження та моделювання стану забруднення токсичними речовинами річки Тиса на території України при надзвичайних ситуаціях. *Четверта міжнародна конференція: «Хімічна і радіаційна безпека: проблеми і рішення». Праці та повідомлення* (Київ, 17-20 травня 2016 р.), Київ, 2016. P. 8.

3. **Gertsyuk Myk.M., Horvath C., Gertsyuk Mod.M., Lysychenko G.V.** Software for modeling and predicting pollution dispersion on the Tisza river and its tributaries in case of emergencies. *Fourth International Conference: «Chemical and radiation safety: problems and solutions». Work and message* (Kyiv, 17-20 May 2016), Kyiv, 2016. P. 11.

4. **Gertsyuk Mod., Melnychenko T., Gertsyuk Myk., Lysychenko G.** Research and modeling of the state Tisza river pollution by toxic substances on the territory of Ukraine in emergencies. *Book of abstracts 18th International symposium on advances in extraction technologies & 22nd international symposium on separation sciences*. Torun, Poland, July 3-6, 2016. P. 83.

5. **Герцюк Мод.М., Лисиченко Г.В., Мельниченко Т.І., Герцюк Мик.М., Товмаченко А.В., Дмитрієва Т.Ф., Пилипенко В.О., Кулібаба Т.О.** Дослідження складу забруднюючих речовин у річці Тиса на території України. *П'ята міжнародна конференція: «Хімічна і радіаційна безпека: проблеми і рішення». Праці та повідомлення* (Київ, 24-26 травня 2017 р.), Київ, 2017. С. 7.

6. **Герцюк Мод.М., Кулібаба Т.О., Мельниченко Т.І., Герцюк Мик.М., Дмитрієва Т.Ф., Товмаченко А.В., Кузенко С.В.** Моніторинг вмісту важких металів в річці Тиса Закарпаття. *П'ята міжнародна конференція: «Хімічна і радіаційна безпека: проблеми і рішення». Праці та повідомлення*. Київ, 24-26 травня 2017 р. С. 8.

7. **Герцюк Мод.М., Пилипенко В.О., Мельниченко Т.І., Герцюк Мик.М., Дмитрієва Т.Ф., Товмаченко А.В.** Моніторинг вмісту поліциклічних ароматичних вуглеводнів у річці Тиса. *П'ята міжнародна конференція: «Хімічна і радіаційна безпека: проблеми і рішення».* Праці та повідомлення. Київ, 24-26 травня 2017 р. С. 9.

8. **Gertsyuk M, Horvath C.** RESit: program for predicting of spread rivers pollution. *Fifth International Conference: «Chemical and radiation safety: problems and solutions».* Work and message. Kyiv, 24-26 May 2017. P. 10.

9. **Gertsyuk Myk., Horvath C., Gertsyuk Mod.** Software for predicting of spread rivers pollution in emergencies. *23rd International Symposium on Separation Sciences. Abstracts.* Vienna, Austria, September 19-22, 2017, (PO-46). P. 194.

10. **Gertsyuk Mod., Lysychenko G., Melnychenko T., Gertsyuk Myk.** Determination and Monitoring of the content of pollutants in the river Tisza on Ukrainian territory. *23rd International Symposium on Separation Sciences. Abstracts,* (KL-8). P. 80

11. **Герцюк М.М.** Метод та можливості для прогнозування забруднення річок web-додатком «RESit». *Проблеми інформатизації. тринадцятої міжнародної науково-технічної конференції. Тези доповідей.* Київ, 11–12 квітня 2019 року. С. 113-114.

12. **Герцюк М.М.** Застосування нейронних мереж для підвищення точності прогнозування наслідків забруднення річок. *Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в інфокомунікаційних технологіях»,* Київ, Україна, 12 лютого 2021 року. С. 78-79.

13. **Герцюк М.М.** Доцільність створення інформаційної системи RESit на базі програмного забезпечення RESit. Міжнародна науково-практична конференція «Сучасні аспекти діджиталізації та інформатизації в програмній та комп'ютерній інженерії», Київ, Україна, 1-3 червня 2023 року. С. 98-99.



### Статті в наукових фахових виданнях

1. **Герцюк М.М., Жебка В.В.** Аналіз програмного забезпечення для прогнозування забруднення річок небезпечними речовинами. *Зв'язок*, N 2, 2019. С. 45-48.
2. **Gertsyuk Myk., Horvath C., Gertsyuk Mod.** Computer application for predicting pollution of the Tisza River in emergency situations. *Aparatura badawcza i dydaktyczna*, 2. 2019, 110-114.
3. **Герцюк М.М.** Модель математичного моделювання наслідків забруднення водойм річок з використанням нейронної мережі, що базується на основі задач регресії. Системи управління, навігації та зв'язку. Збірник наукових праць. Полтава: ПНТУ, 2022, Т. 2 (68). С. 95-98. DOI: <https://doi.org/10.26906/SUNZ.2022.2.095>.
4. **M. Gertsyuk** Prediction of Pollution Level Between Measurement Points by Mathematical Modeling Using Interpolation and Recursion. *Science and Innovation*, 19(3), pp. 65–70. DOI: <https://doi.org/10.15407/scine19.03.065>.
5. **Герцюк М. М.** Методика визначення потенційно небезпечних об'єктів, причасних до викидів забруднюючих речовин в русло річки. Телекомунікації та інформаційні технології. Київ, 2022, №3(76), С. 21-27. DOI: <https://doi.org/10.31673/2412-4338.2022.032127>.
6. **Герцюк М.М.** Створення інформаційної системи RESit для прогнозування забруднення річок в умовах надзвичайних ситуацій. Телекомунікації та інформаційні технології. Київ, 2023, №2(79). С. 13-21. DOI: <https://doi.org/10.31673/2412-4338.2022.021322>.

## ЗМІСТ

ВСТУП.....	19
РОЗДІЛ 1 АНАЛІЗ СУЧАСНИХ МАТЕМАТИЧНИХ МЕТОДІВ ТА АВТОМАТИЗОВАНИХ МЕТОДІВ, ЩО ВИРІШУЮТЬ ПРОБЛЕМУ ПРОГНОЗУВАННЯ НАСЛІДКІВ ЗАБРУДНЕННЯ РІЧОК .....	25
1.1. Моделювання та прогнозування оцінки якості води.....	25
1.2. Огляд програмної моделі QUAL2K.....	28
1.3. Аналіз системи автоматизованих моделей MONERIS.....	29
1.4. Аналіз автоматизованої моделі AQUATOX.....	31
1.5. Аналіз програми моделювання Water Quality Analysis Simulation Program ....	32
1.6. Аналіз платформи Hydrologic Micro Services.....	34
1.7. Аналіз додатку SWToolbox .....	36
1.8. Аналіз додатку для моделювання EXAMS.....	38
1.9. Аналіз додатку Visual Plumes.....	40
1.10. Аналіз програмного забезпечення RUSLE2 .....	42
1.11. Аналіз програмного забезпечення MMSOILS .....	44
1.12. Аналіз програмного забезпечення RESRAD .....	45
1.13. Аналіз програмного забезпечення MEPAS .....	46
1.14. Аналіз програмного забезпечення MOIRA DSS .....	48
1.15. Огляд необхідних для дослідження приладів .....	50
1.16. Постановка завдання дисертаційного дослідження .....	51
1.17. Вимоги до методології.....	53
Висновки до розділу 1 .....	57
РОЗДІЛ 2 ВИЗНАЧЕННЯ МАТЕМАТИЧНОЇ МОДЕЛІ СИСТЕМИ ТА АРХІТЕКТУРИ.....	58
2.1. Визначення математичної моделі системи.....	58
2.2. Закон Фіка .....	58

	15
2.3. Основний метод проведення оцінки забруднення.....	59
2.4. Оцінка динаміки забруднення в конкретній точці .....	62
2.5. Алгоритм багатоточковості.....	63
2.6. Алгоритм мультимаршрутизації.....	65
2.7. Коригування результатів прогнозування, шляхом використання задачі регресії .....	66
2.8. Використання інтерполяції для згладження прогнозованих результатів.....	69
2.9. Визначення джерела забруднення.....	72
2.10. Автоматизація описаних методів та алгоритмів .....	75
2.11. Інформаційна система RESit .....	75
2.12. Технології та підходи RESit .....	75
2.13. Структура RESit .....	76
2.14. База даних .....	79
2.15. Структура .....	80
2.16. Сервер.....	89
2.17. Функціонал серверу .....	89
2.18. Загальна архітектура .....	90
2.19. Побудова та функціонал шару з'єднання з базою даних.....	91
2.20. Архітектура та функціонал REST API .....	93
2.21. Адміністративний додаток .....	94
2.22. Графічний інтерфейс.....	96
2.23. Взаємодія вимірювача з сервером .....	98
2.24. Архітектура сервісу адаптеру .....	98
2.25. Утиліта навчання .....	100
2.26. Настільний додаток.....	106

	16
2.27. Функціонал та призначення .....	106
2.28. Графічний інтерфейс та архітектура головного вікна.....	107
2.29. Майстер проведення обчислень.....	111
2.30. Майстер проведення ручних обчислень .....	113
2.31. Майстер проведення автоматичних обчислень.....	117
2.32. Дашборд: графічний інтерфейс та архітектура .....	122
2.33. Вікно налаштування серверів .....	129
2.34. Математичне ядро застосунку .....	133
Висновки до розділу 2 .....	135
<b>РОЗДІЛ 3 ВАЛІДАЦІЯ МЕТОДІВ ТА АЛГОРИТМІВ</b>	
.....	137
3.1. Валідація методів та алгоритмів.....	137
3.2. Критерії ефективності системи та обмеження .....	137
3.3. Обмеження системи .....	139
3.4. Валідація алгоритмів багатоточковості та мультимаршрутизації .....	141
3.5. Валідація методу коригування результатів прогнозування, що використовує задачу регресії.....	143
3.6. Валідація методу інтерполяції для згладження прогнозованих результатів ...	145
3.7. Валідація методу визначення джерела забруднення, що використовує базу даних фактів.....	149
3.8. Тестування серверу .....	151
3.9. Тестування настільного додатку.....	153
3.10. Головне вікно.....	153
3.11. Майстер проведення ручних обчислень .....	154
3.12. Майстер проведення автоматичних обчислень.....	155
3.13. Дашборд .....	156

	17
3.14. Налаштування серверів.....	156
3.15. Математичне ядро застосунку .....	157
Висновки до розділу 3 .....	158
РОЗДІЛ 4 ДОКУМЕНТАЦІЯ                      ДО                      СИСТЕМИ                      RESIT	
.....	160
4.1. Інформація про систему.....	160
4.2. Робота с сервером.....	160
4.3. Робота з адміністративним додатком.....	161
4.4. Робота з утилітою навчання .....	162
4.5. Робота з програмою .....	162
4.6. Майстер ручного обчислення .....	165
4.7. Початкове вікно моделювання.....	165
4.8. Форма вводу даних.....	165
4.9. Форма «про точку» (вікно відображення повних даних про точку).....	167
4.10. Генерація результуючого файлу.....	168
4.11. Майстер автоматичного обчислення.....	169
4.12. Виклик обчислення та вибір серверу .....	169
4.13. Вибір точок для вимірювання.....	170
4.14. Введення необхідних даних про точку .....	171
4.15. Результат .....	172
4.16. Налаштування програми для роботи з сервером .....	173
4.17. Дашборд: режим повного опису результату .....	174
Висновки до розділу 4 .....	175
ВИСНОВКИ.....	176
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	179
Додаток А. Набір кінцевих точок для REST API.....	194

Додаток Б. Позитивні тестові ситуації для REST API сервісу .....	197
Додаток В. Тестові ситуації головного вікна .....	199
Додаток Г. Тестові ситуації дашборду.....	201
Додаток Д. Акти впровадження.....	202

## ВСТУП

Екологічна безпека є невід'ємною частиною політики кожної держави щодо моніторингу довкілля. За оцінками міжнародних експертів, внаслідок антропогенної діяльності та глобальних змін клімату ресурси прісної води у світовому масштабі стрімко скорочуються і можуть стати головною причиною потенційних екологічних загроз.

Тому, передбачення наслідків забруднення довкілля і використання прогнозів посідає важливе місце в моніторингу навколишнього природного середовища загалом і водойм зокрема [1].

У зв'язку з можливістю забруднення водойм небезпечними та токсичними речовинами, що спричиняють надзвичайну ситуацію, актуальним є опрацювання комплексу ліквідаційних заходів, направлених на мінімізацію наслідків забруднення.

Базуючись на існуючих стратегіях для прогнозування небезпеки виникнення надзвичайної ситуації при забрудненні річок існують три основні підходи:

- профілактичні заходи для запобігання викидів – перевірки щодо запровадження та дотриманням потенційно небезпечними підприємствами стандартів моніторингу стану водойми;
- заходи з упередження забруднення – консультації та навчання персоналу для запобігання викидів, що можуть спричинити надзвичайну ситуацію;
- запровадження комплексу протоколюваних заходів щодо мінімізації наслідків забруднення [2].

Для розв'язування функціональних задач такі дії проводяться органами Державного управління з оцінки стану водних ресурсів, службами з надзвичайних ситуацій, екологічними інспекціями, розрахунково-аналітичними групами, іншими компетентними сторонами, які займаються проблемами оцінки впливу забруднюючих факторів на стан довкілля з метою підвищення рівню контролю, регулювання та моніторингу, обґрунтування найбільш оптимальних рішень з усунення або зниження рівнів ризику наслідків надзвичайних ситуацій [3].

Проблема оцінки наслідків забруднення річок в умовах надзвичайних ситуацій охоплює наступні напрямки:

- екологія – аналіз негативних наслідків на здоров'я людей та стан навколишнього середовища;
- гідрологія – вибір та удосконалення існуючих моделей оцінки якості водойм;
- збір даних – використання інформації, що характеризує русло водойми та визначення потенційних об'єктів, які можуть спричинити забруднення;
- інформаційні системи – автоматизація прогнозування наслідків забруднення річок з метою створення інтеграційних програмних моделей, здатних використовувати інформацію з датчиків, що стоять по ходу русла водойми;
- нейронні мережі – коригування та поліпшення результатів обчислень.

Програмне моделювання якості води є важливим інструментом, що використовується при управлінні водними ресурсами. Зазвичай, моделі розробляються для конкретних регіонів з певним кліматом і фізичними характеристиками. Застосування таких моделей в інших регіонах можуть призвести до неточностей ситуаційної оцінки.

Вирішенню вищезазначених завдань призначене відповідне програмне забезпечення, таке як Qual2K [4], MONERIS [5], WASP8 [6], EXAMS [7], Visual Plumes [8], RUSLE2 [9], MMSOILS [10], RESRAD [11], MEPAS [11].

Мінімізація наслідків забруднення потребує оцінки стану русла, в основі якої лежать система моніторингу та методи прогнозування наслідків розповсюдження забруднення. В цілому, такого типу методи засновані на наступних принципах:

- аналіз існуючих даних та співставлення їх з раніше зафіксованими випадками;
- емпіричні методи виведені для різних ситуацій, що прогнозують статичний або динамічний стан русла;
- законах Фіка (переважно, виведених Тейлором формулах з другого закону Фіка для дифузії в потоці), що використовуються для розрахунків наслідків дифузії.



Для оцінки ступеню загрози необхідно прогнозування рівню забруднення, як в місцях з відомими гідрологічними даними річки, так і тих місцях, де гідрологічні дані про річку відсутні. Через це, доцільно вибрати модель, що не вимагає введення значного перемінного масиву гідрологічних даних.

З метою досягнення високої точності обчислень раціонально використовувати методи, що можуть калібрувати дані про забруднення води у визначених точках, базуючись на існуючих статистичних даних та мають додаток для автоматизації обчислень. Для цього доречно використати підходи регресії.

Інформація про можливі джерела забруднення важлива для оцінки рівню небезпечності ситуації.

Проблема введення значних масивів перемінних даних за необхідності швидкого опрацювання та залучення додаткових інформаційних ресурсів спричиняє потребу у використанні значних інформаційних ресурсів, що призводить до матеріальних та часових затрат і є важливим фактором для оперативної оцінки наслідків надзвичайних ситуацій.

Важливим чинником є збір деяких характеристик у режимі реального часу. Тому, важливою складовою є процес інтеграції з датчиками заміру різних характеристик русла, що передбачає отримання, збереження та відображення актуальних значень. Зважаючи на відстань до деяких віддалених місць на руслі, така взаємодія має бути автоматизованою. В такому випадку серверна система отримує дані з датчиків, а клієнтська системи відображає їх у користувацькому інтерфейсі. Інформаційною основою взаємодії між серверною системою та датчиками можуть бути протоколи TCP або HTTP, як протоколи клієнт-серверних запитів.

Вирішення описаних проблем вибраними або розробленими методами формує методику, що потребує автоматизації. Результатом має бути інформаційна система, що складається з інформаційних елементів. Такі складові включають частини обчислень, інтеграцій з датчиками, взаємодію з базою даних та інтерфейс для відображення результатів прогнозування.

Описані проблеми та їх можливі рішення дозволяють сформулювати актуальне науково-прикладне завдання дослідження, яке полягає у розробці інформаційної технології прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій.

**Об'єктом дослідження** є прогнозування забруднення річок небезпечними та токсичними речовинами в умовах надзвичайних ситуацій.

**Предметом дослідження** є методи прогнозування наслідків забруднення річок.

**Метою роботи** є підвищення ефективності процесу прогнозування наслідків надзвичайних ситуацій на водних об'єктах, таких як скиди токсичних та шкідливих речовин у русло річки за допомогою розробленої інформаційної технології.

**Основними завданнями роботи** є:

- проаналізувати існуючі автоматизовані методики, їх переваги та недоліки;
- проаналізувати існуючі підходи до створення методів прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій;
- розробити метод коригування результатів прогнозування концентрації забруднювача;
- розробити метод прогнозування рівнів забруднення між точками моніторингу;
- розробити метод визначення джерела забруднення;
- розробити інформаційну технологію прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій.

**Методами дослідження**, що дозволять отримати нові наукові результати в частині математично-аналітичних обчислень є методи інтерполяції, рекурсії, регресії, фільтрації та сортування. Підходи до розробки інформаційних систем та взаємодії різних частин системи між собою стали основою розробки клієнт-серверної системи.

**Наукова новизна** отриманих результатів полягає в наступному:

- розроблено метод визначення джерела забруднення на основі методів фільтрації та сортування, що дозволяє отримати список можливих потенційно-небезпечних об'єктів, які могли стати забруднювачами русла річки;

- розроблено метод коригування результатів прогнозування концентрації забруднювача на основі нейронних мереж, що при використанні задач регресії дозволяють підвищити точність прогнозування наслідків забруднення за рахунок існуючих статистичних даних без потреби введення детальних вхідних характеристик;

- розроблено метод прогнозування рівнів забруднення між точками моніторингу на основі методу інтерполяції з використанням рекурсії, що дозволяє отримати більш широкий масив результатів та підвищити точність аналізу наслідків забруднення;

- розроблено інформаційну технологію прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій на основі інформаційної системи, що складаються з різних частин та мають визначену зону відповідальності. Актуалізація даних та використання зазначеної технології дозволяє отримати широкий спектр результатів та збільшити ефективність реагування на наслідки надзвичайних ситуацій за рахунок автоматизації вибраних та розроблених методів, базах існуючих даних та взаємодії з датчиками виміру поточних витрат води в конкретних точках моніторингу.

Практичне значення одержаних результатів полягає в підвищенні ефективності формування бази вхідних характеристик за рахунок використання загальних даних, що можуть бути одержані шляхом ГІС-аналізу, статистичних характеристик та можливій взаємодії системи з датчиками виміру витрат води. Як наслідок, основним перевагами є підвищення ефективності процесу реагування на надзвичайні ситуації при забрудненні води річки токсичними речовинами.

**Зв'язок роботи з науковими темами, грантами.** Частина досліджень дисертаційної роботи була виконана в рамках:

- науково-дослідної роботи «Розробка концептуальних засад моніторингу, прогнозу, моделювання і оцінки стану забруднення водних об'єктів токсичними

речовинами при надзвичайних ситуаціях (на прикладі ріки Тиса)» (Державний реєстраційний номер 0116U003026), яку виконала лабораторія комплексних геохімічних досліджень Державної установи «Інститут геохімії навколишнього середовища НАН України»;

- гранту SfP 984440 програми НАТО «Наука заради миру».

**Структура та обсяг дисертації.** Вміст дисертаційної роботи включає вступ, 4 розділи, висновки, список використаних літературних джерел із 151 найменування, додатків і містить 90 рисунків, 16 таблиць, 9 сторінок додатків

Перший розділ описує існуючі методи та автоматизовані методи, що використовуються для вирішення подібних проблем.

Другий розділ описує принцип розроблених математичних методів та архітектуру інформаційної системи.

Третій розділ описує методи валідації методів та тестування системи.

Четвертий розділ описує документацію до використання описаної системи.

Загальний обсяг дисертації складає 212 сторінок друкованого тексту, що включає у себе 160 сторінок основного тексту.

# РОЗДІЛ 1

## АНАЛІЗ СУЧАСНИХ МАТЕМАТИЧНИХ МЕТОДІВ ТА АВТОМАТИЗОВАНИХ МЕТОДІВ, ЩО ВИРІШУЮТЬ ПРОБЛЕМУ ПРОГНОЗУВАННЯ НАСЛІДКІВ ЗАБРУДНЕННЯ РІЧОК

### 1.1. Моделювання та прогнозування оцінки якості води

Існують математичні методи, що моделюють та прогнозують ситуацію забруднення річки і вибираються в залежності від складності та наявних даних. Поверхнєве прогнозування наслідків якості води вирішується за допомогою простих моделей. Для отримання більш глибокої інформації використовуються, або більш складні моделі, або більш специфічні. Велика кількість моделей, що вимагають більш складних обсягів даних може не використовуватись через те, що такі дані можуть бути недоступними або важкодоступними.

Модель має бути достатньо гнучкою та такою, що має можливість удосконалення. В основі принципу моделей якості води є баланс маси. Зазвичай, водну систему річки ділять на різні сегменти, де описують необхідні гідрологічні, хімічні та інші характеристики води. Більшість моделей імітують якість протягом деякого періоду часу  $\Delta t$ , сек. Час ділиться на інтервали, при цьому потоки вважаються сталими протягом цього інтервалу часу. За математичною моделлю, що описана в роботі Д.П. Лукса та Е. Ван Біка[12], складові балансу маси для сегмента включають:

- транспортні зміни в конкретному сегменті та поза ним,  $Tr$ ;
- зміни у фізичних або хімічних процесах, що відбуваються в сегменті,  $P$ ;
- зміни джерела скиду з сегменту,  $S$ .

Масовий баланс має наступні компоненти:

- маса в обчислювальній комірці  $i$  на початку часового кроку  $t$  (сек), мг:  $M_i^t$ ;
- маса в обчислювальній комірці  $i$  в кінці часового кроку  $t$ , мг:  $M_i^{t+\Delta t}$ ;
- зміни обчислювальної комірки  $i$  під час транспортування (переміщення речовини через простір), мг:  $\left(\frac{\Delta M_i}{\Delta t}\right)_{Tr}$ ;

- зміни обчислювальної комірки за фізичним, хімічним або біологічним процесами, мг:  $\left(\frac{\Delta M_i}{\Delta t}\right)_P$ ;

- зміни обчислювальної комірки за джерелами (наприклад, навантаження відходами, скиди річок), мг:  $\left(\frac{\Delta M_i}{\Delta t}\right)_S$ .

Формула масового балансу має наступний вигляд:

$$M_i^{t+\Delta t} = M_i^t + \Delta t \left(\frac{\Delta M_i}{\Delta t}\right)_{Tr} + \Delta t \left(\frac{\Delta M_i}{\Delta t}\right)_P + \Delta t \left(\frac{\Delta M_i}{\Delta t}\right)_S, \quad (1.1)$$

Транспортування є адвективним та дисперсійним. Адвективний транспорт – це транспорт потоку води та має наступну формулу для обчислення:

$$T_{x_0}^A = v_{x_0} * A * C_{x_0}, \quad (1.2)$$

де  $v_{x_0}$  – середня швидкість води, м<sup>3</sup>/сек,  $A$  – площа поперечного перерізу, м<sup>2</sup>,  $C_{x_0}$  – середня концентрація, мг/м<sup>3</sup>,  $x$  – сегмент ( $x_0$  – початковий сегмент).

Дисперсний транспорт є результатом різниці концентрацій та визначається за формулою:

$$T_{x_0}^D = -D_{x_0} * A * \frac{\partial C}{\partial x} \Big|_{x=x_0}, \quad (1.3)$$

де  $D_{x_0}$  – коефіцієнт дифузії.

Основний принцип градієнту дисперсії визначається дифузійним законом Фіка та визначається для визначеного сегменту за наступною формулою:

$$\frac{\partial C}{\partial x} \Big|_x = \lim(\Delta x \rightarrow 0) \left( \frac{C_{x+0.5\Delta x} - C_{x-0.5\Delta x}}{\Delta x} \right), \quad (1.4)$$

Для сталого прогнозування моделі використовується рівняння наступного виду:

$$C(x) = \begin{cases} \left(\frac{W_C}{Q_m}\right) * \exp\left[\left(\frac{v}{2D}\right)(1+m)x\right], & x \leq 0 \\ \left(\frac{W_C}{Q_m}\right) * \exp\left[\left(\frac{v}{2D}\right)(1-m)x\right], & x \geq 0 \end{cases}, \quad (1.5)$$

де  $W_C$  - вкидання, г/день,  $Q_m$  – стандартний рівень потоку, м<sup>3</sup>/с (використовується для оцінки відхилення від норми або для визначення небажаних становищ у річковій системі),  $m$  – кількість дефіцитів від стандартного рівня потоку, од.,  $D$  – коефіцієнт дисперсії,  $v$  - швидкість руху води у річковому потоці, м<sup>3</sup>/сек.

$$Q_m = \frac{\frac{A_v(v^2 + 4kD)}{2}}{v}, \quad (1.6)$$

де  $k$  – коефіцієнт речовини

$$m = \left(1 + \left(\frac{4kD}{v^2}\right)\right)^{1/2} \quad (1.7)$$

Максимальна концентрація в місці при розливі обчислюється наступною формулою:

$$C(0) = \frac{W_c}{Q_m} \quad (1.8)$$

Для розповсюдження концентрації при невеликих швидкостях використовується наступна формула:

$$C(x) = \begin{cases} \left(\frac{W_c}{2A(KD)^{\frac{1}{2}}}\right) * \exp\left[+x\left(\frac{k}{D}\right)^{\frac{1}{2}}\right], & x \leq 0 \\ \left(\frac{W_c}{2A(KD)^{\frac{1}{2}}}\right) * \exp\left[-x\left(\frac{k}{D}\right)^{\frac{1}{2}}\right], & x \geq 0 \end{cases}, \quad (1.9)$$

Коефіцієнт дисперсії та коефіцієнт речовини – характеристики, що залежно від типу річки та речовини приймають ті чи інші функції та значення. Існує велика кількість функцій, що дозволяють знайти коефіцієнт дисперсії. Зазвичай, для річок використовується формула Ельдера [13]:

$$D = 5.93Hu^* \quad (1.10)$$

де  $H$  – середня глибина потоку в вимірюваній секції, м,  $u^*$  – швидкість зсуву(швидкість, з якою водяний потік рухається поблизу дна), м/сек;

формула Кашефіпура та Фальконера [14]:

$$D = 10.612 \left(\frac{U}{u_*}\right) HU, \quad (1.11)$$

де  $U$  – поперечна швидкість потоку, м/сек;

та інші формули, що описані в роботі Hossien Riahi-Madvar та Seyed Ali Ayyoubzadeh «Developing an Expert System for Predicting Pollutant Dispersion in Natural Streams» [15].

## 1.2. Огляд програмної моделі QUAL2K

Модель Qual2K є одновимірною модельованою структурою для імітування якості води у річках і водотоках, основним рівнянням якої є одновимірна дисперсія, що враховує для контролю якості води в водоймах такі компоненти, як ступінь розчинність та розведення речовин у воді, їх надходження із зовнішніх джерел [4].

Має наступні налаштування:

- одновимірне, повне вертикальне та горизонтальне перемішування води;
- потік води біля річки постійний;
- зовнішні змінні, такі як температура, вологість, потужність вітру та вологість, виражаються в днях, внутрішні метеорологічні рівняння розраховуються на добовій осі часу;
- регулярний точковий вхід і вихід.

Модель вимагає даних щодо просторових особливостей та гідрологічних характеристик басейну, структури річки, швидкості потоку, форми перетину каналами, дані про просочуванні води, замулення тощо.

Ця модель налаштована на одновимірне, налаштування по вертикальному та горизонтальному перемішуванні води, при умові, що потік, точковий вхід і вихід води в річці постійний. Зовнішні змінні, такі як температура, вологість, потужність вітру відображаються в днях, внутрішні метеорологічні рівняння розраховуються на добу.

Модель QUAL2K робить прогнози за такими показниками, як біологічне споживання кисню, співвідношення між водоростями, поживними речовинами та світлом, загальним неорганічним вуглецем, нітрифікацією, показниками рН.

QUAL2K дозволяє задавати інші кінетичні параметри залежно від конкретної мети.

Qual2K розбиває річку на сегменти, що є базовими одиницями розрахунку та мають сталі гідравлічні характеристики (нахил дна, ширина водойми тощо), проте, за необхідності, розмір сегментів може змінюватися.

Річка може бути без притоків або з притоками, що також поділяються на сегменти (рис. 1.1).



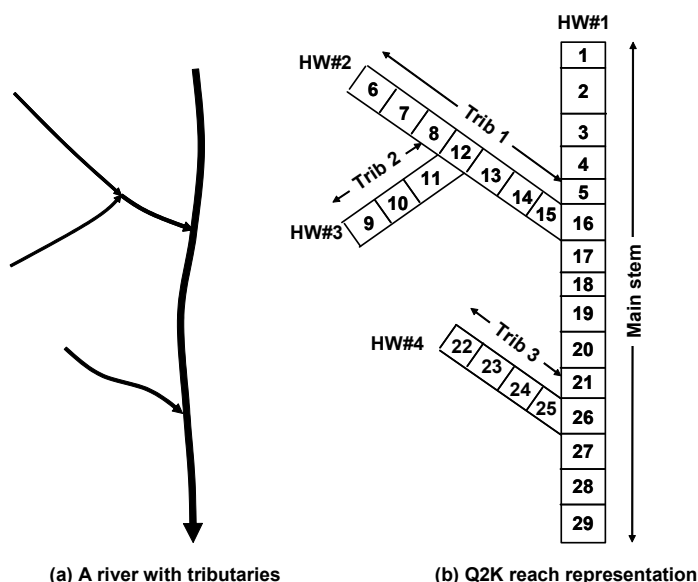


Рис. 1.1. Схема сегментації річки без приток (а) та з притоками (б) у програмі «Qual2K» [4]

Ураховується наявність водоспадів, водозливів, перепадів рельєфу [4][16][17][18].

Перевагами QUAL2K є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- наявність бази даних: у вигляді XLS файлу;

Зазначені характеристики дають можливість використовувати цю модель в умовах надзвичайних ситуацій. Однак, ряд недоліків накладає ряд обмежень для проведення таких прогнозувань. А саме:

- незручний користувацький інтерфейс, що включає в себе складність аналізу отриманих результатів;
- необхідність залучення широкого спектру характеристик, необхідних для проведення прогнозування.

### 1.3. Аналіз системи автоматизованих моделей MONERIS

MONERIS – це система автоматизованих моделей для оцінки якості навколишнього середовища, що має можливість кількісно оцінювати викиди органічних речовин [19].

Метою для даної системи є визначення джерел викидів речовин у річкові басейни, що потрапили у воду в межах регіону, аналіз міграції речовин та затримки органічних речовин у річкових потоках.

Основними складовими автоматизованої системи є дані стоку, дослідження якості води для басейнів, що досліджуються та ГІС-інтегровані мапи. Користувацький інтерфейс складається з ГІС мап, що відображаються розділені по регіонах викиди органічних речовин в потоки річки.

Моделі для поверхневих вод написані при використанні мови VBA. Основний принцип моделі лежить в дослідженні даних водозабору та водопостачання.

Цю модель було розроблено з метою:

- ідентифікації джерела викидів поживних речовин на регіональній основі;
- аналізу міграції та утримання поживних речовин у річкових системах;
- забезпечення структури для вивчення альтернатив управління.

Існує можливість використання в цілях сільськогосподарського землекористування.

Основою розрахунку є масообмін азоту і фосфору у воді, що є головними маркерами органічних сполук. Модель ураховує природний фон азоту і фосфору, витоки з комунальних очисних споруд, промислові скиди, застосування добрив, атмосферне осадження азоту на орних землях та інших територіях, надходження речовин із підземних вод, ерозію через дренажні труби, викиди з міських територій.

Шляхи і підходи стосовно джерел є необхідною умовою для вжиття заходів щодо зменшення неточкових викидів та викидів у точці-джерелі, що обчислюються окремо для різних видів землекористування.

В рамках MONERIS, основною та найменшою одиницею проведення обчислень є аналітична одиниця (AU). Від так, басейн річки (RB) – ділянка землі, де поверхневий стік проходить через послідовність потоків річок, суббасейн (SB) – деяка ділянка, де весь стік проходить через ряд потоків річок, в яких досліджується вода, водозбір (CAT) – сума аналітичних одиниць, підрозділ (SU) - частка одиниці конкретного суббасейну.

Даний застосунок може бути застосований для розуміння плану дій щодо ліквідації наслідків надзвичайних ситуацій за рахунок пошуку джерела забруднення. Однак, не може бути застосований для прогнозування наслідків забруднення водою.

#### **1.4. Аналіз автоматизованої моделі AQUATOX**

AQUATOX – механістична екосистемна модель, що оцінює якість води. Базується на результатах оцінки якості води за такими характеристиками, як надлишок органічних речовин та хімікатів, прогнозує розподіл різних типів забруднювачів води не тільки у водоймах, а й наслідки їх впливу на елементи водної екосистеми, зокрема риби, безхребетних, водних рослин [20].

Переважно, може бути використана для глибшого розуміння процесів взаємозв'язку між хімічним, фізичним та біологічним середовищем. Від так, модель може бути використана для вирішення широкого кола питань, пов'язаних з екологією.

Можливими варіантами використання можуть бути:

- аналіз можливих подразників спостережуваного біологічного порушення;
- прогноз впливу токсичних речовин на воду;
- прогнозування рівня зміни клімату відносно потенційних реакцій;
- оцінки впливу змін екосистеми на водне середовище;
- оцінка часу відновлення забруднення до безпечного рівня, як наслідок забруднення.

Ззовні, користувацький інтерфейс нагадує офісний додаток. Для проведення різного роду обчислень, використовується Simulation Wizard – функціонал, метою якого є задати необхідні дані для проведення обчислення. В результаті роботи з даним функціоналом, тобто заповнення всіх необхідних, детальних даних для проведення обчислення можна отримати вікно, що містить групи та показує результати обчислення.

Додаток має можливість збереження та завантаження обчислення. Ще однією особливістю додатку є можливість використання глобальних баз даних для проведення обчислень.

Підсумовуючи вищеописану модель, можна констатувати, що застосунок може бути використаний, як помічник для проведення різних навчальних та наукових досліджень екологічних змін водойм, оскільки містить вимогу до розуміння та роботи з рядом детальних характеристик[20][21].

Метод може бути застосований для оцінки якості води в умовах надзвичайних ситуацій. Перевагами такого використання є:

- можливість здійснювати моделювання не тільки якості води, а й вплив на водні екосистеми;
- інтерактивний користувацький інтерфейс.

Недоліками є:

- необхідність залучення спектру характеристик самого русла та екосистеми, необхідних для проведення прогнозування;

В цілому, метод є корисним для моделювання якості води та екологічних процесів у водних системах, в нормальних умовах. Але, в умовах надзвичайних ситуацій, принципи проведення прогнозувань вимагають уважного врахування деталей та можливих попередніх досліджень.

### **1.5. Аналіз програми моделювання Water Quality Analysis Simulation Program**

Water Quality Analysis Simulation Program (WASP) – це програма динамічного моделювання водних систем, що інтерпретує та прогнозує вплив якості води на природні явища та антропогенне забруднення при прийнятті різноманітних рішень та управління забрудненнями. WASP має функціонал дослідження 1,2 та 3-мірних систем при забрудненні речовинами різноманітних типів [22].

WASP містить наступний вибір модулів що дозволяють моделювати:

- змінні якості води;
- транспорт осаду;

- нагрівання температури води;
- та інші [6].

Кінетичні моделі WASP базуються на наборі рівнянь транспорту та трансформації. Транспорт водойм здійснюється потоком води через обчислювальну мережу. Приклад мережі показано на рис. 1.2.

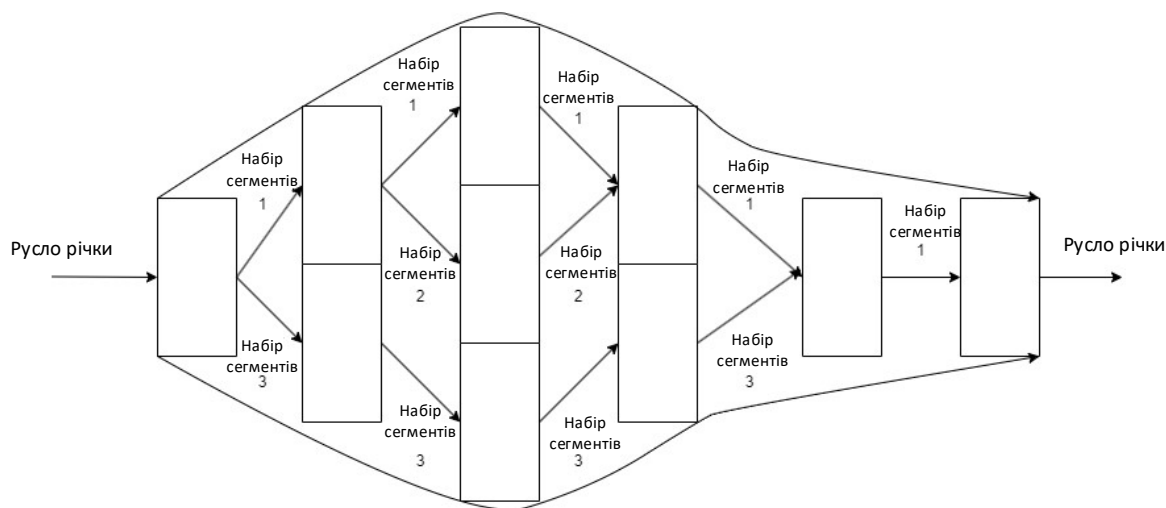


Рис. 1.2. Приклад обчислювальної мережі потоку води WASP

Транспорт у WASP розділений на шість різних типів або «полів»:

- перше транспортне поле - це потік у товщі води. Потік несе складові якості води «нижче за течією» і забезпечує розведення їх в потоці;
- друге транспортне поле визначає рух пористої води в осадовому шарі;
- розчинені компоненти якості води 2 проходять через русло за допомогою пористої води;
- третє, четверте та п'яте транспортні поля визначають транспорт твердих частинок забруднювачами шляхом відстоювання та захоронення твердих речовин. WASP дозволяє користувачеві визначати швидкість осідання безпосередньо використовуючи параметри сегмента. WASP також реалізує механістичний підхід для транспортування осаду. Складові якості води, сорбовані на тверді частинки, транспортуються між товщею води та осадовим шаром;
- шосте транспортне поле визначає випаровування або випадання опадів із сегменту поверхневої води або до нього [23].

WASP має можливість зберігати введені дані у файл. Метод може застосовуватись в умовах надзвичайних ситуацій. Перевагами WASP є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;

- наявність бази даних;

- зручний користувацький інтерфейс.

Зазначені характеристики дають можливість використовувати цю модель в умовах надзвичайних ситуацій. Однак, ряд недоліків накладає ряд обмежень для проведення таких прогнозувань. А саме:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;

- недостатня адаптованість до умов конкретної надзвичайної ситуації.

Загалом, WASP є корисним інструментом для моделювання забруднення, але в умовах надзвичайних ситуацій можливе залучення додаткових джерел інформації для проведення прогнозування.

## **1.6. Аналіз платформи Hydrologic Micro Services**

Hydrologic Micro Services (HMS) – платформа ГІС, що дозволяє вирішити питання взаємодії, прозорості та ефективності в екологічному моделюванні. Основною метою платформи є вирішення проблем гідрологів при моделюванні якості води.

HMS – це набір веб-сервісів та компонентів моделювання даних, що стосуються гідрології та якості води. Веб-служби з надання даних постачають необроблені дані через запити сценаріїв щодо таких гідрологічних параметрів, як опади, температура повітря, сонячна радіація, вологість ґрунту, випаровування, поверхневий та підземний потік і стік води. HMS дозволяє користувачам швидко характеризувати гідрологію водозбору, цим самим скорочуючи час на збір та попередню обробку даних та параметризувати робочі процеси моделі. Як результат, моделювальники можуть витратити більше часу на аналіз результатів, аніж на збір та попередню обробку вхідних даних.

Веб-сервіс HMS (HMS-PCAT) – це інструмент порівняння та аналізу атмосферних опадів, обчислення статистичних даних для попередньої обробки та

порівняння для екологічного моделювання. Сервіс забезпечує інтерфейс для дослідження джерела даних атмосферних опадів та завантаження даних, незалежно від обчислювальних платформ або мов програмування. HMS-PCAT було розроблено для гармонізації даних про опади з кількох джерел з різними протоколами пошуку даних, форматами, просторовою та часовою роздільною здатністю та посиланнями на час [24].

З точки зору архітектури програмного забезпечення, HMS-PCAT - сукупність веб-служб, що забезпечують послідовний спосіб доступу до даних про атмосферні опади в різних джерелах з різними форматами, роздільною здатністю та протоколами доступу. HMS-PCAT доступний браузеру, що інтегрує внутрішні та зовнішні веб-служби для забезпечення обробки даних, геопросторових та статистичних обчислень, які створюють доступну базу даних. Веб-сервіси були розроблені за принципом REST API через протокол HTTP. Такий підхід дозволяє різним сервісам комунікувати між собою, не зважаючи на те, якою мовою програмування був розроблений черговий сервіс. Інтерфейс користувача в браузері побудований при використанні стандартного фронт-енд стеку – HTML/JavaScript/CSS.

Платформа вимагає таких типів вхідних даних, як місцезнаходження, часовий ступінь, рівень агрегування та джерело даних із сіткою для порівняння.

Отримавши ці дані, сервер HMS перевіряє їх на точність і створює завдання. Далі сервер HMS розподіляє інформацію до відповідних внутрішніх служб для отримання даних із зовнішніх сервісів. Внутрішні служби виконують ряд операцій з базами даних, тимчасово зберігаючи їх до повного отримання. Остаточний аналіз та результати порівняння виконуються на сервері і повертаються до веб-браузера для відображення. Результат може бути представлено в JSON, CSV формах або переглянутим через користувальницький інтерфейс як інтерактивний лінійний графік.

В якості результату сервіс показує:

- інтерактивний графік часових рядів, записані у певний день;

- таблицю зведених статистичних даних, що показує стандартне відхилення, середнє, медіанне та процентильне значення;
- матрицю кореляції Пірсона для візуалізації та порівняння даних, що демонструє лінійне порівняння наборів даних про атмосферні опади;
- метадані, що показують місце, де дані були отримані.

На основі цих результатів можна оцінити варіації між наборами даних і прийняти рішення щодо відповідності кожного набору даних для проекту.

Аналіз HMS показує, що застосунок можна використати для надзвичайних ситуацій. Перевагами платформи є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- широкий спектр моделей для проведення прогнозування;
- наявність бази даних;
- зручний користувацький інтерфейс;
- наявність розподіленого середовища.

Недоліками HMS є:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;
- необхідність мати широку експертизу в моделях, щоб застосовувати їх правильно.

Загалом, платформа є достатньо потужною для проведення прогнозувань, в умовах надзвичайних ситуацій. Особливістю методу є наявність розподіленого середовища, що дає можливість розподілити обов'язки при підтримуванні бази даних системи та адаптовувати під різні моделі взаємодії з персоналом. Проте, необхідність залучення спектру детальних характеристик та необхідність широкої експертизи в моделях ускладнює використання моделі.

### **1.7. Аналіз додатку SWToolbox**

SWToolbox – набір інструментів для поверхневий вод має вигляд настільного додатку, що надає набір таких процедур для статистичного аналізу, таких як:



- статистична частота за  $n$  днів 7Q10, що визначає мінімальний потік за десять років протягом семи днів;
- біологічно обґрунтовані дані щодо потоків;
- статистику тривалості потоку;
- різноманітні річні, місячні та сезонні статистичні дані.

В якості вхідних даних додаток використовує набори даних з Національної системи інформації про водні ресурси, Геологічної служби США. SWToolbox було створено при використанні платформи BASINS. Ці функції включають в себе наступні статистичні можливості для підтримки моделювання кількості і якості води:

- список даних часових рядів;
  - графічні дані часових рядів, криві тривалості потоку, криві частоти потоку, поточні суми, а також залишки, кумулятивні відмінності і діаграми розкиду двох часових рядів;
  - обчислення  $n$ -денного максимуму або мінімуму річного часового ряду з денного часового ряду;
  - виконання частотного аналізу будь-якого річного часового ряду з використанням логарифмічного розподілу Пірсона типу III (LP3);
  - обчислення біологічно обґрунтованих розрахункових потоків з використанням даних річних часових рядів;
  - перевірку на монотонність тенденції в річних часових рядах;
  - виконання аналізу тривалості потоку;
  - розрахунок таблиць і кривих гідрографа тривалості;
  - обчислення різних річних, сезонних і місячних статистичних даних.
- Наприклад середнє, мінімальне і максимальне значення для обраного часового ряду.

Для збору даних використовується метод статистичної частоти 7Q10, суть якого полягає в зборі середнього значення даних мінімального потоку за  $n$ -денний період в рік, сезон, чи інших період часу та середнього значення даних максимального потоку.

Додаток включає мапу, на яку наносяться шари, що відображають результати та можливість використання вищеописаних методів, як окремого функціоналу. Виходячи з цього можна констатувати, що графічні дані результатів відображаються у вигляді мап та графіків залежності[25].

Загалом, інструменти SWToolbox можна використовувати для надзвичайних ситуацій. Перевагами набору інструментів є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- широкий спектр моделей для проведення прогнозування;
- зручний користувацький інтерфейс.

Недоліки наступні:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;
- необхідність мати широку експертизу в моделях, щоб застосовувати їх правильно.

Отже, платформа є достатньо гнучкою та широкою для проведення прогнозувань, в умовах надзвичайних ситуацій. Проте, необхідність залучення спектру детальних характеристик, необхідність широкої експертизи в моделях та необхідність адаптації моделі до середовища ускладнює використання моделі.

### **1.8. Аналіз додатку для моделювання EXAMS**

EXAMS (система моделювання аналізу експозицій) – настільний додаток для моделювання водних екосистем та швидкої оцінки концентрації та міграції синтетичних органічних хімічних речовин, таких як пестициди, промислові забруднювачі та фільтрати з місць захоронення. Програмне забезпечення містить систему управління базами даних, розроблену для зберігання та управління базами даних, що необхідні додатку. Основною метою додатку є оцінка небезпеки, заподіяної хімічно синтезованими токсичними сполуками.

В EXAMS закладені методи для:

- стаціонарного аналізу хімічних процесів;

- підходів для вивчення короточасних викидів хімічних речовин;
- повне кінетичне моделювання, що дозволяє змінювати місячні середні кліматологічні параметри та змінювати хімічні навантаження на добу.

EXAMS забезпечує аналіз впливу:

- очікуваних концентрацій синтетичних хімічних речовин та продуктів їх перетворення в навколишньому середовищі;
- розподіл хімічних речовин у водній екосистемі та аналіз відносної чутливості кожного процесу трансформації та міграції;
- стійкості, тобто часу, необхідного для природного очищення екосистеми після закінчення викидів хімічної речовини.

Додаток EXAMS було розроблено використовуючи мову FORTRAN 95. Складову інтерфейсу складає інтерпретатор командного рядка. Такі параметри, як складність екологічного опису, кількості хімічних речовин повністю контролюється користувачем. Існує також можливість зберігати файли прямого доступу (UDB) п'яти взаємодіючих хімічних сполук та 100 сегментів навколишнього середовища[7].

Загалом, функціонал додатку дає можливість використання для прогнозування надзвичайних ситуацій. Перевагами застосунку є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- широкий спектр характеристик, що прогножуються;
- наявність бази даних.

Недоліками є:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;
- відсутність користувацького інтерфейсу, управління здійснюється через командну строку.

Отже, платформа може бути застосовна. Однак, відсутність інтерактивного користувацького інтерфейсу та необхідність залучення спектру детальних характеристик ускладнює її застосування в умовах надзвичайних ситуацій.

### 1.9. Аналіз додатку Visual Plumes

Visual Plumes – це комп’ютерна програма, що симулює поширення речовин, в поверхневих водах довільного потоку води в поверхневих водах. Може імітувати, як одноразовий шлейф (тобто, коли речовина потрапляє в потік в один момент), так і злитий шлейф (коли речовина потрапляє в потік протягом тривалого часу або в декількох точках). Додаток вирішує проблему узгодженості моделі способом, що включає інші моделі до набору закладених моделей. Основна ідея моделювання полягає в тому, що існують рекомендації по використанню конкретних моделей у вибраних категоріях потоків. Програмне забезпечення включає наступні основні моделі:

- модель поверхневого розрядного PDS (surface discharge model) - тривимірна модель шлейфу, що застосовується до скидів у водні об’єкти з припливних каналів. В основі PDS лежить модель Ейлера для інтегрального потоку поверхневого скидання плавучої води в рухоме навколишнє водоймище, що включає ефекти поверхневого теплообміну. Передбачається, що шлейф залишається на поверхні з плавучістю, змушуючи його підніматися і поширюватися в усіх напрямках. Початковий стан шлейфу спричиняється до проникання в навколишнє середовище. PDS обчислює траєкторію шлейфу, середнє та розведення середньої лінії, ширину та глибину шлейфу, надлишкову температуру центральної лінії. Він також обчислює площі в межах вибраних ізотерм;

- модель DKNW (Davis, Kannberg, Hirst model) - тривимірна модель шлейфу, що також застосовується до одно- та багатопортових затоплених скидів;

- тривимірна модель UM3(three-dimensional Updated Merge) на основі UM(Updated Merge) - лагранжева модель, що містить гіпотезу прогнозованої зони захоплення. Передбачається, що шлейф знаходиться в стійкому стані. У формулюванні Лагранжа це означає, що послідовні елементи йдуть по одній і тій же траєкторії. Конверт шлейфу залишається незмінним, тоді як елементи, що рухаються крізь нього, змінюють свою форму та положення. Однак умови навколишнього середовища можуть змінюватися в часі. Ці зміни можуть сильно

відрізнятися за часом, за яким розряджений елемент досягає кінця початкової фази розведення, як правило, з максимальним спадом. В результаті однопортові шлейфи моделюються як тривимірні об'єкти;

- модель NRFIELD - це емпірична модель заснована на експериментальних дослідженнях багатопортових дифузоров у багат шарових струмах. Модель базується на експериментах із використанням Т-розділів, кожен з яких має два порти, тому для його застосування необхідно вказати принаймні чотири порти. Важливим є припущення, що дифузор може бути представлений лінійним джерелом. Це припущення може мати важливе значення для малих зон змішування, в яких шлейфи можуть не злитися.

Описані моделі можна запускати послідовно та порівнювати графічно.

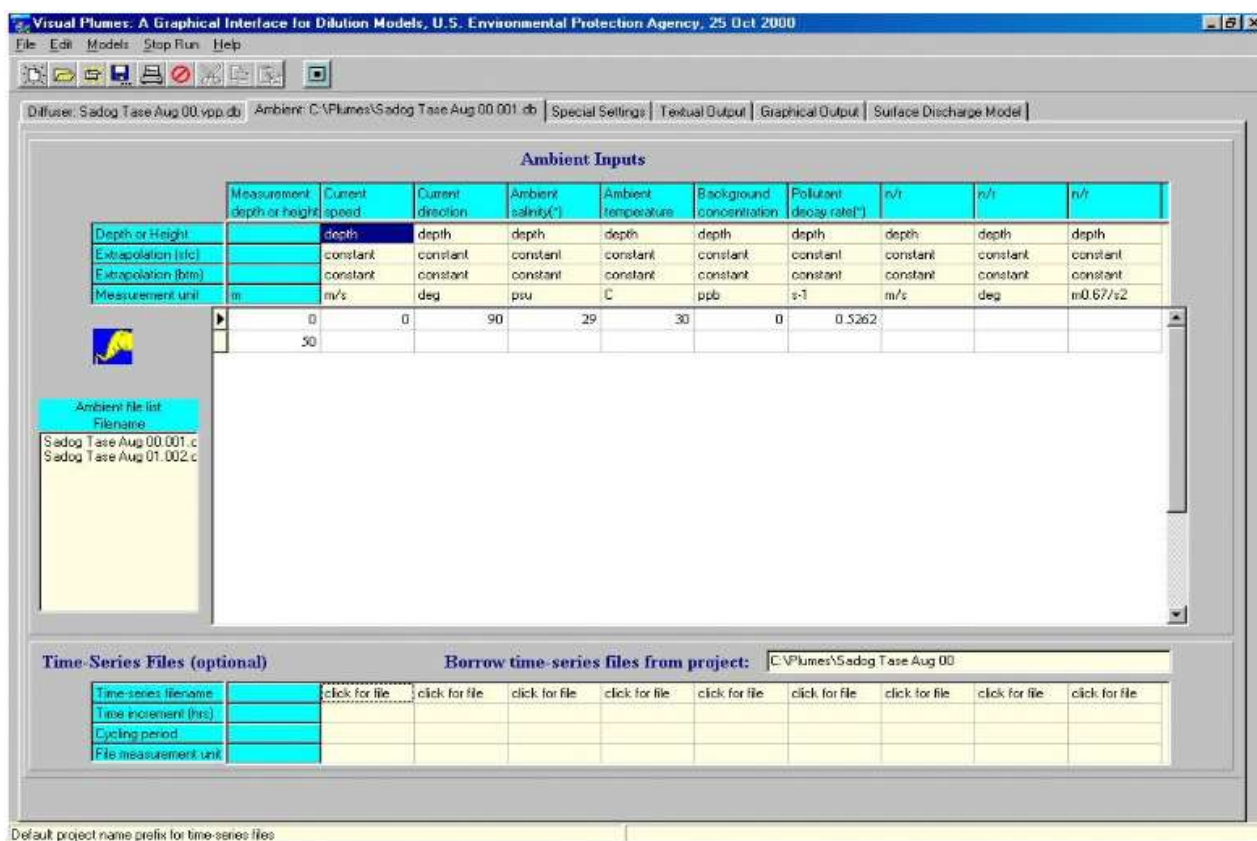


Рис. 1.3. Інтерфейс інформації про потік Visual Plumes

Графічний інтерфейс вводу інформації містить дві функціональні форми:

- інформація про дифузор - форму, що має бути заповнена деякими характеристиками дифузії, за якими будуть моделювати процеси розповсюдження самого шлейфа;

- інформація про потік (навколишнє середовище). Інтерфейс форми, що має бути заповнена деякими характеристиками потоку, за якими будуть моделюватися процеси протікання самого шлейфа показано на рис. 1.3.

В цілому додаток дозволяє оцінити стан водних потоків та надає розуміння процесів розповсюдження та міграції в цілому. Враховуючи той факт, що побудовані алгоритми моделювання не враховують великої кількості факторів, проводити передбачення зміни потоків, або протікання шлейфа неможливо [8].

Зазвичай, Visual Plumes використовується для моделювання та візуалізації поширення забруднення в атмосфері або водних тілах. Перевагами програмного забезпечення є:

- широкий спектр характеристик, що прогнозуються;
- зручний та інтерактивний користувацький інтерфейс;

Однак, для прогнозування надзвичайних ситуацій, додаток може бути менш ефективним та мати деякі обмеження, оскільки початкова мета застосунку створена для прогнозування навколишнього середовища в цілому.

### **1.10. Аналіз програмного забезпечення RUSLE2**

RUSLE2 - це комп'ютерна програма, що оцінює швидкість процесів ерозії, використовуючи систему математичних рівнянь. Метою програмного забезпечення є керівництво щодо планування збереження та контролю ерозії. Оператор описує русло, використовуючи інформацію з бази даних для обчислення оцінок ерозії. Додаток обчислює відкладення в каналах і невеликих водоймах, що закінчуються сухопутними шляхами потоку. RUSLE2 має три основні компоненти:

- науковий компонент, що включає математичні рівняння. Вхідні дані для рівнянь вибираються користувачем для представлення чотирьох основних факторів, що впливають на ерозію на певній ділянці. Цими факторами є клімат конкретної локації, властивості ґрунтів, топографію та землекористування;

- база даних. Використовується для описання умов, характерних для русла. В базі даних міститься інформація про клімат, ґрунти, рослинність, залишки, пористі смуги та бар'єри, перехоплювачі потоку, включаючи канали відведення, тераси та

невеликі відтоки, системи поверхневого дренажу, зрошення, шляхи сухопутного потоку;

- комп'ютерна програма. Включає обчислювальний механізм, що вирішує математичні рівняння, засоби обслуговування баз даних та інтерфейс (екран комп'ютера), який приймає введені користувачем дані та відображає обчислені значення. Служба сільськогосподарських досліджень USDA (United States Department of Agriculture) несла загальну відповідальність за розробку RUSLE2 та математичні рівняння, що використовуються в програмі. Університет Теннессі мав головну відповідальність за розробку комп'ютерної програми RUSLE2, включаючи її інтерфейс та обчислювальну систему. Служба охорони природних ресурсів USDA несла головну відповідальність за розробку бази даних RUSLE2 для посівів. Інші організації розробили інформацію про базу даних, посібники користувача та навчальні матеріали для RUSLE2.

Інтерфейс додатку показано на рис. 1.4.

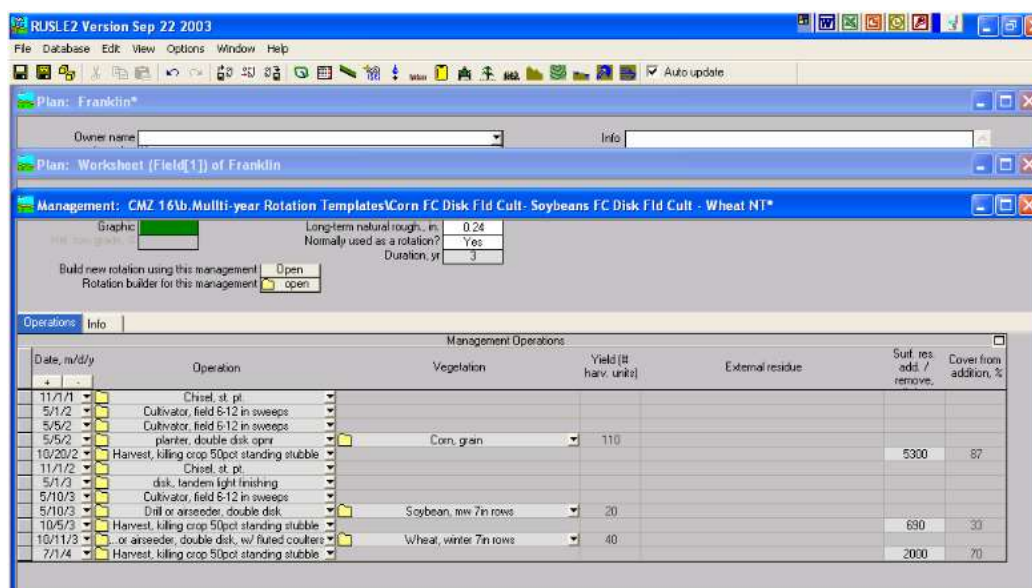


Рис. 1.4. Інтерфейс додатку RUSLE2

Принцип інтерфейсу побудовано за логікою, схожою до додатку Microsoft Excel 2003. Одиницею проведення обчислень є робочий лист, а результати обчислень показано у табличному вигляді та у вигляді графіку.

Додаток використовується для аналізу ерозії ґрунтів, а отже не може бути використаний для прогнозування наслідків забруднення річок.

### 1.11. Аналіз програмного забезпечення MMSOILS

MMSOILS - це програмне забезпечення для оцінки ризиків впливу на здоров'я людини викидів забруднень та небезпечних відходів [10]. В основі математичної моделі лежить комбінація емпіричних рівнянь та аналітичних рішень. Емпіричні рівняння включають алгоритми для оцінки швидкості викиду забруднюючих речовин, потоків проміжних речовин, поглинання забруднювачів рослинами та тваринами, вплив на здоров'я людини та ризики для довкілля. Модель використовується для проведення аналізу, виявлення потенційних загроз, прогнозування оцінки опромінення та ризиків у конкретних руслах.

Модель працює з такими видами речовин, як органічні речовини, метали та радіонукліди.

Оцінює джерела забруднення: забруднений ґрунт, сміттєзвалище, поверхневий водозбір, закачувальна свердловина, підземний резервуар для зберігання та купа відходів.

Проводить оцінку за наступними характеристиками: баланс маси, розпад джерела.

Симулює наступні механізми скиду джерела: ерозія, випаровування, стік та вимивання.

Включає наступні фактори динаміки русла:

- для повітря - змішування та такі характеристики, як швидкість вітру, напрямок та ін.;
- для гідрології поверхні – опади, витрати та просочування;
- для поверхневих водойм – сталий стан русла;
- для ґрунтових вод – зона аерації та сталий стан.

Оцінює шляхи впливу на їстівні культури (рослини, рибу, молоку, воду і т.д.), продукти вдихання (гази) та шкіру. Також оцінює ризик щодо нанесення шкоди здоров'ю, зокрема рак та інші наслідки [11].

Перевагою цього програмного забезпечення є його метод оцінки, а саме комбінація емпіричних рівнянь та аналітичних рішень. Програмне забезпечення можна використовувати для прогнозування наслідків забруднення річок. Однак



модель розроблена для оцінки ризику для здоров'я людей від забруднювачів ґрунту. Тому вона не може враховувати всі фактори, які можуть впливати на забруднення річок. Перевагами застосування є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- широкий спектр характеристик, що прогножуються;
- наявність бази даних;
- наявність зручного користувацького інтерфейсу.

Недоліками є:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;
- може бути складною для використання в умовах надзвичайних ситуацій.

В цілому, MMSOILS можна застосовувати в умовах надзвичайних ситуацій, але для вирішення обмеженого набору цілей, зв'язаних з оцінкою впливу на людину та навколишнє середовище, що можуть бути вирішені.

### **1.12. Аналіз програмного забезпечення RESRAD**

RESRAD – це програмне забезпечення для оцінки впливу забруднень на людину та навколишнє середовище. Як і MMSOILS, в основі математичної моделі RESRAD лежить комбінація емпіричних рівнянь та аналітичних рішень. Модель використовується для проведення аналізу виявлення потенційних загроз, прогнозування оцінки опромінення та ризиків на конкретних руслах. Модель є вузько наведеною, але має можливість прогнозування наслідків радіаційних забруднень.

Модель працює з такими видами речовин, як органічні речовини, метали та радіонукліди.

Оцінює джерела забруднення: забруднений ґрунт, сміттєзвалище, купу відходів, траншеї.

Проводить оцінку за наступними характеристиками: баланс маси, розпад та ріст джерел забруднень.

Дозволяє симулювати декілька забруднень за моделюванням.

Симулює наступні механізми скиду джерела: ерозія, випаровування, стік, вимивання та прямий скид в аераційну зону, ґрунтові води, поверхневі води та повітря.

Включає наступні фактори динаміки русла:

- для повітря - змішування та такі характеристики, як швидкість вітру, напрямок та ін.;

- для ґрунтових вод – зона аерації та сталий стан.

Оцінює шляхи впливу на їстівні культури (рослини, рибу, молоко, воду і т.д.), продукти вдихання (гази), радіаційний фон та шкіру. Також оцінює ризик на нанесення шкоди здоров'ю, зокрема радіаційні опіки, рак та інші наслідки [11].

Перевагою цього програмного забезпечення є його метод оцінки, а саме комбінація емпіричних рівнянь та аналітичних рішень. Такий підхід має легку адаптацію до ситуації та не вимагає необхідності залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування. Основною його метою є оцінки наслідків радіаційного впливу на людину та навколишнє середовище. Однак, застосунок не має можливості прогнозувати наслідки забруднення річок.

### **1.13. Аналіз програмного забезпечення MEPAS**

MEPAS – як і MMSOILS та RESRAD, це програмне забезпечення для оцінки впливу на людину та ризику для здоров'я, що пов'язане із викидами забруднень із місць небезпечних відходів. В основі математичної моделі лежить комбінація емпіричних рівнянь та аналітичних рішень. Модель є достатньо широкою та об'єднує майже всі можливості MMSOILS та RESRAD.

Модель працює з такими видами речовин, як органічні речовини, метали та радіонукліди. Оцінює джерела забруднення: забруднений ґрунт, сміттєзвалище, поверхневий водозбір, закачувальна свердловина, підземний резервуар для зберігання, купу відходів та траншеї.

Проводить оцінку за наступними характеристиками: баланс маси, розпад та ріст джерела.

Дозволяє симулювати декілька забруднень за моделювання з декількох джерел.

Симулює наступні механізми скиду джерела: ерозія, випаровування, стік, вимивання та прямий скид в аераційну зону, ґрунтові та поверхневі води, повітря та суходіл.

Оцінює шляхи впливу на їстівні культури (рослини, рибу, молоко, воду і т.д.), продукти вдихання (гази), радіаційний фон та шкіру. Також оцінює ризик на нанесення шкоди здоров'ю, зокрема радіаційні опіки, рак та інші наслідки [11].

Перевагою цього програмного забезпечення є його метод оцінки, а саме комбінація емпіричних рівнянь та аналітичних рішень. Програмне забезпечення можна використовувати для прогнозування наслідків забруднення річок. MERAS може бути корисною у розробленні стратегій зменшення забруднення річок та прийнятті рішень щодо охорони та відновлення довкілля. Перевагами застосування є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- широкий спектр моделей, що можуть бути використані;
- наявність бази даних та можливість інтеграції з глобальними базами даних;
- наявність зручного користувацького інтерфейсу.

Недоліками є:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;
- може бути складною для використання в умовах надзвичайних ситуацій.

В цілому, MERAS можна застосовувати в умовах надзвичайних ситуацій, але за рахунок недоліків використання може бути ускладнене.

### 1.14. Аналіз програмного забезпечення MOIRA DSS

MOIRA – це програмне забезпечення для користувачів з різним рівнем відповідальності при прийнятті рішень, різним досвідом у роботі з комп'ютерами, екологічним моделюванням та радіоекологією. З графічним інтерфейсом користувача оператор працює з мапами, таблицями, графіками та звітами. Користувач може проводити всі етапи прийняття рішення або починати з прямого запиту актуальних даних. MOIRA DSS містить десять прогностичних екологічних моделей, дозову модель, економічну модель та модуль багатоатрибутного аналізу (МАО)[26].

Моделі, що використовує MOIRA в якості вхідних даних використовують екологічні параметри з європейських баз даних та ГІС-систем. Моделі базуються на «колективних параметрах». Система включає наступні моделі для озер та складних систем прісної води, що імітують:

- поведінку часу гідрологічних, морфологічних, екологічних величин та параметрів міграції забруднювачів через водні екосистеми;
- міграцію забруднюючих речовин із водозбору у водну систему;
- міграцію забруднюючих речовин через абіотичні компоненти водної системи;
- міграцію радіонуклідів від абіотичних компонентів до риби.

Складні системи прісної води, такі як великі річки, складаються з ланцюга взаємопов'язаних «елементарних сегментів (ЕС)», що складаються з:

- товщі води;
- верхнього шару осаду, що безпосередньо взаємодіє з водою («інтерфейсний шар»);
- проміжного шару осаду нижче «шару розділу» («донний осад»);
- осадового шару під «донним осадом»;
- правого і лівого підвідбору кожного сегменту[27].

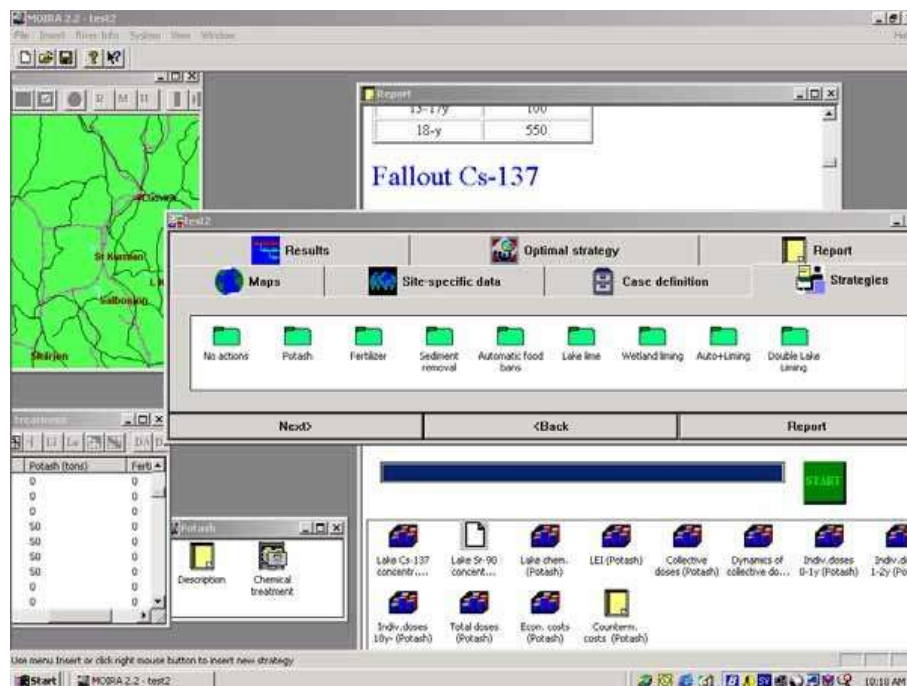


Рис. 1.5. Інтерфейс користувача додатку MOIRA DSS[28]

Інтерфейс користувача включає декілька таких функціональних компонентів, як мапи, панель керування, табличні дані та інші ГІС-подібні функціональні компоненти. Вид інтерфейсу показано на рис. 1.5.

Загалом, програмне забезпечення є платформою, що містить набір різних моделей для проведення прогнозувань в водному середовищі. При наявності відповідних моделей, платформу можна використовувати для прогнозування наслідків забруднення річок. Перевагами програмного забезпечення є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;

- широкий спектр моделей для проведення прогнозування;

- зручний користувацький інтерфейс;

Недоліки наступні:

- необхідність залучення спектру детальних характеристик самого русла, необхідних для проведення прогнозування;

- необхідність мати широку експертизу в моделях, щоб застосовувати їх правильно.

### 1.15. Огляд необхідних для дослідження приладів

З метою пришвидшення процесу збору даних, що є важливим фактором в умовах надзвичайних ситуацій необхідно здійснити інтеграцію з датчиками заміру різних характеристик русла. Однією з таких характеристик, що необхідні в процесі дослідження є швидкість течії в режимі реального часу.

Таким тип вимірювання може бути здійснений різними вимірювачами швидкості течії – приладами, що вимірюють швидкість води в річках каналах[29]. Існує різний спектр вимірювання води: від маленьких струмків до великих океанів. Вимірювачі існують різних типів:

- механічний вимірювач – заснований на вимірюванні швидкості обертання спірального гвинту у воді;
- вимірювач електромагнітного типу – працює на основі вимірювання напруги електродним датчиком, що рухається провідником, енергія якого генерується за рахунок електромагнітного поля;
- вимірювач акустичного типу – використовує ефект Доплера, тобто заснований на вимірюванні величини зсуву відбитого сигналу щодо частоти випромінюваного акустичного сигналу.

Існують різні виробники пристроїв, такі як Aanderaa, SonTek, Global Water, Valeport, OTT Hydromet GmbH та ін., що випускають такі моделі, як Model 801, MIDAS ECM, Model ROV, Model 001, Model 106 (компанія Valeport), OTT Qliner, OTT Qliner 2 (компанія OTT Hydromet GmbH)[30]. Кожен з вимірювачів для має свій ряд характеристик та може бути застосований в конкретному місці течії.

Вищеперераховані та інші моделі взаємодіють через комунікатор – пристрій, що працює на базі Windows Mobile або іншій мобільній операційній системі, що зв'язується з комунікатором кабелем або через технологію Bluetooth. Комунікатор оснащений програмним забезпеченням, що реалізовує взаємодію через власні драйвери.

### **1.16. Постановка завдання дисертаційного дослідження**

Метою роботи є підвищення ефективності процесу прогнозування наслідків надзвичайних ситуацій на водних об'єктах, таких як скиди токсичних та шкідливих речовин у русло річки за допомогою розробленої інформаційної технології.

Об'єктом дослідження є прогнозування забруднення річок небезпечними та токсичними речовинами в умовах надзвичайних ситуацій.

Предметом дослідження є методи прогнозування наслідків забруднення річок та інформаційна для прогнозування забруднення річок на основі вибраного методу.

Результатами аналізу моделей, методів, програмних застосунків оцінки якості води, що можуть бути використані для прогнозування наслідків забруднення русла річок в умовах надзвичайних ситуацій є ряд переваг та недоліків, при розробці методики.

Основними перевагами методів є:

- широкий спектр характеристик, що можуть бути враховані для прогнозування, що покращує точність прогнозованих результатів;
- широкий спектр моделей для проведення прогнозування;
- наявність бази даних;
- інтерактивний користувацький інтерфейс;
- можливість здійснювати моделювання не тільки якості води, а й вплив на водні екосистеми;
- наявність розподіленого середовища.

Основними недоліками є:

- незручний користувацький інтерфейс, що включає в себе складність аналізу отриманих результатів;
- необхідність залучення широкого спектру характеристик, необхідних для проведення прогнозування;
- недостатня адаптованість до умов конкретної надзвичайної ситуації;
- необхідність мати широку експертизу в моделях, щоб застосовувати їх правильно;

- відсутність користувацького інтерфейсу та здійснення управління через командну строку.

- проблема обмеженої точності, що властива всім проаналізованим рішенням.

Загалом, розроблена методика має відрізнятись від інших наступними перевагами:

- наявність розподіленого середовища. Проблему потрібно вирішувати шляхом розробки інформаційної системи, що дозволяє легке масштабування;

- наявність бази даних. Проблема вирішується шляхом введення бази даних, як складової інформаційної системи;

- інтерактивний користувацький інтерфейс. Проблема вирішується шляхом розробки настільного застосунку, що дозволяє проводити прогнозування та введення «градієнту» результатів. Такий градієнт може бути досягнутий обчислення проміжних значень методом інтерполяції, з інтерактивним відображенням результатів;

- відсутність вимоги введення характеристик русла. Проблема вирішується таким комплексом заходів, як вибір методики, що вимагає найменшої кількості таких характеристик для введення, впровадження частини таких характеристик в базу даних та використання датчиків в режимі реального часу;

- наявність легкого або автоматизованого механізму адаптації під умови конкретної локації. Проблема вирішується шляхом запровадження нейронної мережі, що використовує задачу регресії для калібрування результатів;

- відсутність великої кількості моделей для налаштування, або вхідних даних до введення з метою відсутності необхідності широкої експертизи моделей;

- надзвичайна ситуація також потребує інформації про те, який потенційно небезпечний об'єкт міг здійснити скид. Для використання такої можливості використовується аналітично-класифікаційний метод, що здійснює аналіз бази даних відповідно до умов надзвичайної ситуації.

Таким чином, формуються наступні завдання дослідження, а саме:

- розробити метод коригування результатів прогнозування концентрації забруднювача;



- розробити метод прогнозування рівнів забруднення між точками моніторингу;
- розробити метод визначення джерела забруднення;
- розробити інформаційну технологію прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій.

### **1.17. Вимоги до методології**

Приведені завдання дисертаційного дослідження дають можливість виведення списку вимог до кожного заданого методу.

#### **Метод коригування результатів прогнозування забруднення**

Метою методу є коригування результату прогнозування рівня забруднення водойми, базуючись на виявлених закономірностях локації. Характеристикою прогнозування рівня забруднення є концентрація речовини-забруднювача. Русло може мати різні закономірності течії в залежності від висоти, погодних умов та певних події у навколишньому середовищі. Відповідно, значення коригування можуть бути різними. Тому фактор пори року потрібен для розділення значень коригування в певній локації.

Кожна локація може мати велику кількість факторів, що впливають на забруднення, тому обчислення значення коригування та виявлення закономірних похибок може відбуватись лише регресивним шляхом, базуючись на статистиці. Таке значення може варіюватись і не мати закономірного коефіцієнту, а тому необхідно визначити таке значення, що буде найбільш прийнятним для деякої локації в певну пору року. Таким чином, механізм навчання з використанням нейронних мереж стає основою даного методу.

Отже, першою частиною методу є збереження даних для навчання, другою – навчання на існуючих даних, третьою – безпосереднє коригування результатів забруднення.

Наступний список підсумовує вищеописану аналітику:

- в якості основи має бути використана нейронна мережа, що працює на основі задачі регресії;

- метод має виявляти закономірності, тобто навчатись на існуючому наборі даних;
- дані мають бути розділені за локацією та порою року;
- обчислений коефіцієнт має використовуватись для коригування результатів відповідної локації у відповідну пору року;
- результатом, що має коригуватись є концентрація рівня забруднення речовиною забруднювачем;
- метод має складатись з трьох частин: обчислення коефіцієнту, зберігання даних та коригування результатів.

### **Метод прогнозування рівня забруднення між точками моніторингу.**

Метою методу є прогнозування рівня забруднення між існуючими локаціями, для яких розрахунок проведено, в тих місцях, де інформація про характеристик русла є недоступною. Такі значення допомагають створити «градієнт» результатів забруднення, розширивши картину ситуації. У якості вхідних даних для цього методу є наявні результати в локаціях, що знаходяться поблизу. Тому метод інтерполяції є доречним рішенням для його використання у якості бази. Основною характеристикою для роботи є концентрації рівня забруднення.

Таким чином, список вимог до методу наступний:

- метод має бути побудований на основі інтерполяції;
- основною характеристикою для роботи методу є концентрація рівня забруднення в деякій локації у деякий час;
- метод не має вимагати додаткової інформації про характеристики русла, а базуватися лише на виведених результатах.

### **Метод визначення джерела забруднення.**

Метою такого методу є визначення можливого джерела забруднення, не вимагаючи додаткової інформації, аніж місце виливу річки в русло. Така інформація дає більше уявлення про надзвичайну ситуацію, що сталась, або потенційно може статись. Джерелом забруднення, зазвичай, може бути підприємство, що внесене в список потенційно небезпечних об'єктів. Тому,

ресурсом даних для роботи даного методу є база даних потенційно небезпечних об'єктів.

Таким чином, список вимог має наступний вигляд:

- метод повинен використовувати базу даних потенційно небезпечних об'єктів;
- вхідною інформацією є точки виливу речовини в русло;
- результатом має бути список наймовірніших потенційно небезпечних об'єктів, що можуть бути причетними до забруднення.

**Інформаційна система прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій.**

З метою автоматизації обраних, удосконалених та розроблених методів та можливості використання їх в умовах надзвичайних ситуацій необхідно розробити інформаційну систему.

Така система має включати наступні складові:

- база даних – для зберігання даних;
- сервер – для забезпечення цілісності бази даних з можливих підключених датчиків;
- адміністративна панель – для забезпечення можливості редагування та оновлення даних;
- утиліта навчання – для проведення обчислень, пов'язаних з навчанням нейронної мережі;
- настільний програмний застосунок – для проведення прогнозувань та відображення комплексного результату.

Метою бази даних є збереження даних, що необхідні системі. Враховуючи складність інформації, база даних повинна мати певну структуру. Зазвичай, складні структури будуються реляційними базами даних. Таким чином, вимогами до бази даних є:

- необхідність використання реляційної бази даних;
- можливість зберігання даних та взаємодії з інформацією.

Метою серверу є забезпечення цілісності бази даних в умовах інтегрованих датчиків та взаємодія з адміністративною панеллю та настільним програмним застосунком. Важливою є можливість взаємодії з усіма складовими системи, створюючи центральну точку, до якої ці складові можуть бути підключеними. Таким чином вимоги до серверу наступні:

- взаємодія з усіма складовими системи;
- інтеграції з датчиками;
- забезпечення цілісності даних.

Метою утиліти навчання є навчання нейронної мережі методу коригування результатів прогнозування забруднення. Таким чином, вимогами до утиліти є:

- автоматизація частини нейронної мережі, що відповідає за навчання;
- взаємодія з базою даних, тобто внесення виведених коефіцієнтів для всіх необхідних даних;
- у якості ресурсів для навчання використовується інформація з бази даних.

Метою настільного програмного застосунку є проведення прогнозувань та відображення комплексного результату. Застосунок має наступний ряд вимог:

- у якості ресурсу виступає інформація отримана з серверу;
- має бути присутня можливість взаємодії з декількома серверами на випадок необхідності роботи з декількома річками;
- дані повинні зберігатися локально на випадок необхідності автономної роботи (без з'єднання з сервером);
- застосунок повинен мати можливість здійснювати прогнозування;
- вхідними даними, що мають бути введені для обчислення є вибір точки вливу, вибір точки прогнозування, введення додаткових характеристик ситуації та речовини, що була викинута;
- основним результатом має бути концентрація в деякій локації у деякий час;

- результати за методами коригування результатів прогнозування забруднення, прогнозування рівня забруднення між точками моніторингу та визначення джерела забруднення повинні обчислюватись та відображатись;
- результати можливих удосконалень основного методу повинні відображатись;
- результати мають відображатись у вигляді карти, графіку та таблиці;
- застосунок має бути настільним з метою забезпечення необхідності автономної роботи.

Всі описані вимоги до розробки методів та створення інформаційної системи формують інформаційну технологію прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій.

### **Висновки до розділу 1**

Грунтуючись на вищезазначених дослідженнях, в даному розділі:

1. Проаналізовано принцип розробки методів моделювання оцінювання якості води та прогнозування наслідків забруднення. Отримана інформація дозволить удосконалити методику для використання в умовах надзвичайних ситуацій.

2. Проаналізовано існуюче програмне забезпечення, метою якого є оцінка якості води та прогнозування наслідків забруднення річок та водойм в цілому. Аналіз показав переваги існуючих систем та методів, на основі яких працюють зазначені системи. Базуючись на основі зазначеної інформації, можна сформулювати додаткові критерії ефективності при розробці інформаційної технології.

3. Проаналізовано принципи роботи вимірювача швидкості течії води та взаємодії пристрою з пристроями комунікаторами. Базуючись на отриманій інформації, можна сформулювати список вимог та основні принципи до розробки інтеграції вимірювачів швидкості течії води з інформаційною системою.

## РОЗДІЛ 2

### ВИЗНАЧЕННЯ МАТЕМАТИЧНОЇ МОДЕЛІ СИСТЕМИ ТА АРХІТЕКТУРИ

#### 2.1. Визначення математичної моделі системи

Головними критеріями ефективності, водночас, результатами функціонування системи є наступні дві характеристики:

- пікова концентрація забруднювача в певній точці (мг/л);
- час, через який концентрація забруднювача набуде свого піку.

В розділі описано методи для вирішення різних проблем системи, що складають методики для оцінювання наслідків забруднення. Безумовно, такі методи визначають свої змінні та взаємозалежності.

#### 2.2. Закон Фіка

Процеси дифузії описано використовуючи залежності, вказані в законах Фіка, що прогнозуються як динаміка дифузії, так і деякі числові характеристики дифузії в трьох вимірах. Існують два виведених закони Фіка.

Перший закон визначає дифузійний потік  $J$  в напрямку  $X$  з градієнтом концентрації речовини  $dC/dx$ . Формула, що відображає цей закон є наступною:

$$J = -D \frac{dC}{dx}, \quad (2.1)$$

де  $D$  – деякий коефіцієнт дифузії, що виведено різними шляхами для конкретного об'єкту дифузії [31]. Знак «-» в вищенаведеній формулі показує напрямок потоку з більшого до меншого.

Другий закон визначає швидкість зміни концентрації речовини в деякій точці, яка зумовлена дифузією в напрямку  $X$  з градієнтом концентрацій речовини  $\frac{dC}{dx}$  в конкретний час  $t$ . Формула, що відображає цей закон має наступний вигляд:

$$\frac{dC}{dt} = D \frac{d^2C}{dx^2}. \quad (2.2)$$

Для водного об'єкту, типу річки, особливістю якого є течія рівняння повинне мати дещо інший вигляд. Зокрема, має бути врахована швидкість течії. Тому

дифузійне рівняння типу Фіка для одновимірного середовища, що було виведене Тейлором використовується, як базовий принцип для створення конкретних моделей. Формула рівняння для одновимірного середовища має наступний вигляд:

$$\frac{dC}{dt} + u \frac{dC}{dx} = D \frac{d^2C}{dx^2}, \quad (2.3)$$

де  $C$  – середня концентрація;  $u$  – постійна швидкість в напрямку  $x$ ;  $t$  - час;  $x$  – поздовжній напрямок потоку;  $D$  – коефіцієнт повздовжньої дисперсії.

Також, існує варіант такого рівняння, що описує тривимірну дифузію за наступною формулою:

$$\frac{dC}{dt} + u \frac{dC}{dx} + v \frac{dC}{dy} + w \frac{dC}{dz} = D \left( \frac{d^2C}{dx^2} + \frac{d^2C}{dy^2} + \frac{d^2C}{dz^2} \right), \quad (2.4)$$

де  $v$  – постійна швидкість в напрямку  $y$ ,  $w$  – постійна швидкість в напрямку  $z$ ,  $y$  та  $z$  – напрямки потоку [31].

Існуючі моделі типу Фіка для прогнозування динаміки забруднення спираються на принципи, що описують ці закони. Зокрема, моделі що створюються для оцінки якості води в річці виводяться з принципу, описаного в рівнянні типу Фіка для одновимірного середовища, що було-виведене Тейлором. Існують також моделі не Фікового типу, що включають додаткові закономірності навколишнього середовища для уточнення прогнозу.

### 2.3. Основний метод проведення оцінки забруднення

Для проведення необхідних розрахунків, потрібно урахувати, яка концентрація забруднювача води буде в конкретній точці в певний час та коли вона досягне піку. Крім того, проведення таких обчислень має бути виконане, спираючись лише на базову характеристики русла, що можуть бути отримані шляхом проведення ГІС-аналізу. Таке обмеження існує через проблему того, що не всі місця в руслі річки досліджені.

Тому, доцільно вибрати модель, описану в роботі Харві Джобсона [32].

Модель заснована на серії емпіричних рівнянь, що базується на рівнянні регресії для обчислення різних гідрологічних характеристик оцінки якості води.

Основним рівняннями, що обчислюють пікову концентрацію є:

$$C_p = \frac{C_{up} \cdot M}{Q}, \quad (2.5)$$

де  $M$  – маса забруднення, наведена в мг,  $Q$  – витрати води на ділянці під час вимірювання, наведений в м<sup>3</sup>/сек,  $C_{up}$  – одинична-пікова концентрація, що є однією з найважливіших характеристик моделі і має особливість, що ґрунтується лише на часі руху та співвідношенні річкового стоку до середньорічного розряду[18]. Рівняння регресії для даної характеристики має наступний вигляд:

$$C_{up} = 857 \cdot T_p^{-0.760 \cdot \frac{Q}{R} - 0.079}, \quad (2.6)$$

де  $R$  – середній річний потік.

Основним рівнянням, що обчислює час, через який концентрація набуде свого піку є наступне:

$$T_p = \frac{D}{V_p \cdot 3600}, \quad (2.7)$$

де  $D$  – дистанція (в метрах),  $V_p$  - швидкість пікової концентрації. Оптимальним рівнянням для обчислення є:

$$V_p = 0.152 + 8.1 \cdot A_{rel}^{0.595} \cdot \frac{Q}{A}, \quad (2.8)$$

де  $A$  – середня площа водозбору,  $A_{rel}$  – безрозмірна площа водозбору.

Безрозмірна площа водозбору обчислюється наступним чином:

$$A_{rel} = \frac{A \cdot \sqrt{g}}{R}, \quad (2.9)$$

де  $g$  – прискорення сили тяжіння ( $g \sim 9.81$ ),  $R$  – середні витрати води. Середня площа водозбору є середнім арифметичним між площею водозбору в місці розливу та площею водозбору в місці обчислення. Формула середньої площі водозбору обчислюється за наступною формулою:

$$A = \frac{A_{pol} + A_{prof}}{2}. \quad (2.10)$$

де  $A_{pol}$  – площа водозбору в місці розливу,  $A_{prof}$  – площа водозбору в місці обчислення. Також можна вирахувати рівняння можливого максимуму швидкості за наступною формулою:

$$V_m = 0.2 + 40.0 \cdot A_{rel}^{0.595} \cdot \frac{Q}{A}. \quad (2.11)$$

Рівняння безрозмірних відносних витрати визначається, як:



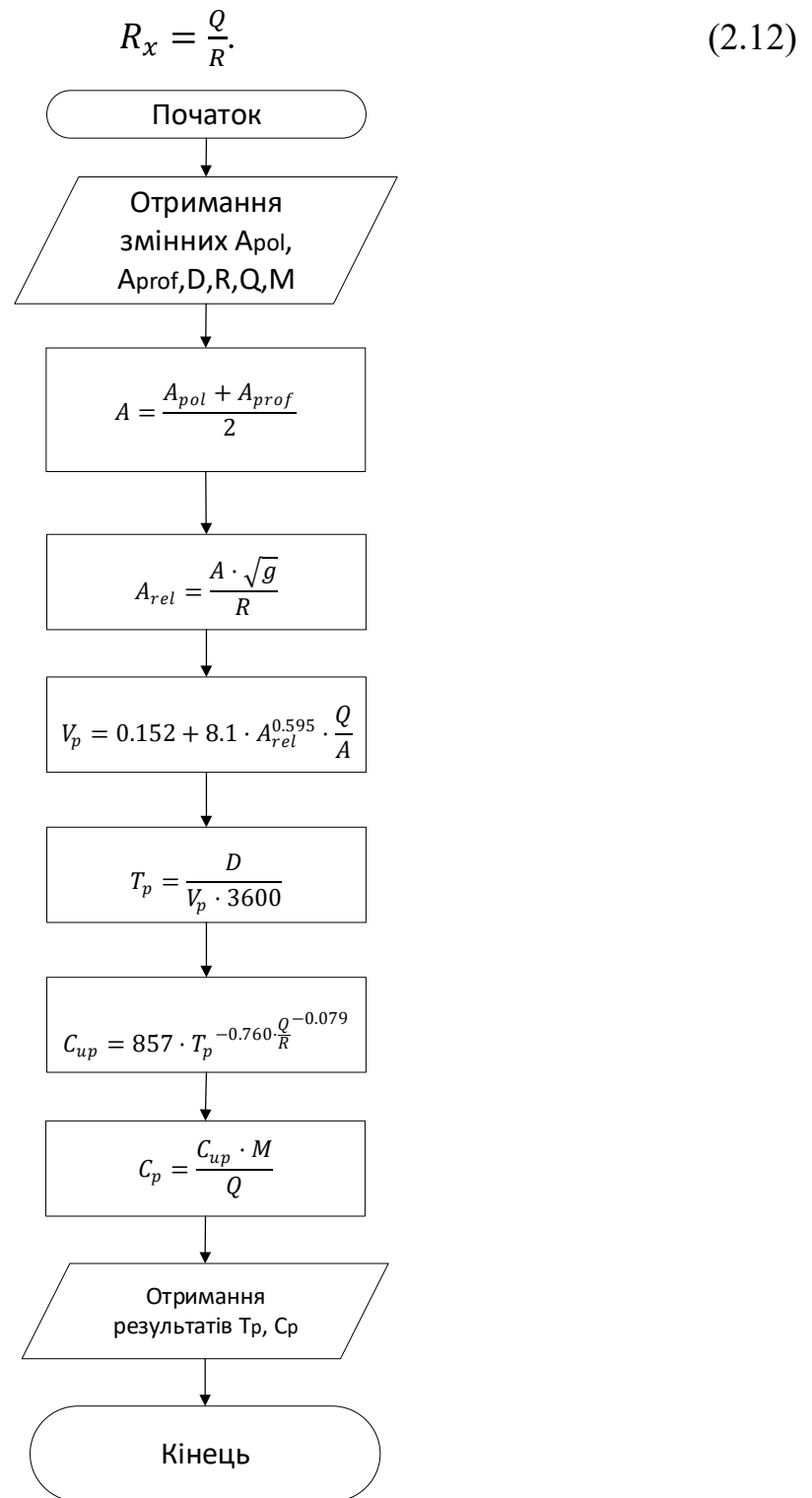


Рис. 2.1. Блок-схема методики виконання обчислень за вибраною моделлю

Отже, серед вхідних характеристик, що мають бути введені для проведення обчислення є:

- площа водозбору в точці розливу, м<sup>2</sup>;
- площа водозбору в місці обчислення, м<sup>2</sup>;
- дистанція до точки забруднення, м;

- середньорічний річковий стік, м<sup>3</sup>/сек;
- витрати в секції в час вимірювання, м<sup>3</sup>/сек;
- маса речовини, що була викинута в річку, мг.

Алгоритм обчислення показано на блок-схемі (рис. 2.1).

#### 2.4. Оцінка динаміки забруднення в конкретній точці

Характеристика піку концентрації речовини в конкретний час в певній точці може бути недостатньою для визначення безпечності рівню її величини.

Деякі рівняння, відображені в роботі Харві Джобсона можна використати для моделювання піку забруднення в точці у визначений час після настання пікового моменту [32][33]. Беручи до уваги рівняння пікової концентрації

$$C_p = \frac{C_{up} \cdot M}{Q}, \quad (2.13)$$

та рівняння одиничної пікової концентрації

$$C_{up} = 857 \cdot T_p^{-0.760 \cdot \frac{Q}{R} - 0.079}, \quad (2.14)$$

можна визначити пікову концентрацію в статичному середовищі в конкретний час.

Таким чином, серед вхідних відомих статичних даних урахується:

- середньорічний річковий стік, м<sup>3</sup>/сек;
- витрати води в секції в час вимірювання, м<sup>3</sup>/сек;
- маса, що було викинута в річку, мг.

Серед змінних даних потрібно ввести лише час, через який концентрація буде мати певне значення.

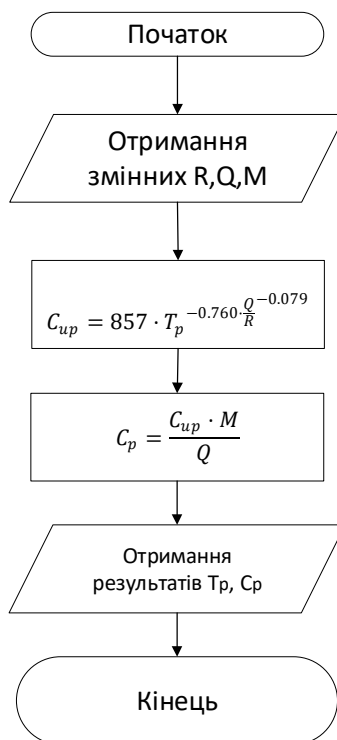


Рис. 2.2. Блок-схема методики виконання обчислень за вибраними рівняннями  
Блок-схему, що відображає алгоритм обчислення показана на рис. 2.2.

## 2.5. Алгоритм багатоточковості

Вибраний метод проводить обчислення між двома точками. Цей метод може бути точковим, тобто таким, що працює між деякими вибраними точками.

Такими точками є точки моніторингу – деякі точки спостереження за станом русла річки, обладнані датчиками виміру швидкості течії, температури, рівня води та іншими гідрологічними та метеорологічними характеристиками. В базовій реалізації основний метод може працювати між двома точками, де одна є точкою моніторингу, друга – точкою прогнозування. Від так, існує доцільність розробки алгоритму, що дозволить проводити моделювання між декількома точками одночасно.

Для цього потрібно побудувати таку послідовність точок, де кожна точка посиляється на попередню, формуючи русло. Якщо посилення немає, дана точка вважається початком русла річки. Схематично така послідовність показана на рис. 2.3.



Рис. 2.3. Русло ріки, що утворюється з послідовності точок моніторингу за послідовними посиленнями

Така послідовність є маршрутизацією. Під час обчислення кожна наступна точка може бути точкою вимірювання для основного методу. Таким чином перша точка – це місце розливу, а кожна наступна точка – це окреме вимірювання. Умовно, можна позначити точку розливу, як « $i$ », а кожную наступну точку, як  $i+n$ , де  $n$  – порядковий номер точки після місця розливу. В цьому випадку обчислення проводиться між точкою  $i$  та точкою  $i+n$ .

Гідрологічні характеристики при цьому мають бути адаптовані наступним чином:

- площа водозбору в точці розливу ( $A_{pol}$ ) – константа, що включає всю область вище за течією;
- площа водозбору в точці обчислення ( $A_{prof}$ ) – константа, що включає всю область вище за течією;
- середні витрати ( $R$ ) – константа, взята для конкретної точки моніторингу;
- витрати води в точці під час вимірювання ( $Q$ ) – константа, що взята для конкретної точки моніторингу;
- дистанція ( $D$ ) – змінна, що включає значення з попередніх точок за наступною формулою:

$$D = \sum_{i=0}^n D_i, \quad (2.15)$$

де  $i$  – номер точки моніторингу,  $n$  – кількість точок, що обчислюються,  $D_i$  – дистанція від попередньої до наступної точки;

- маса ( $M$ ) – константа, що взята для конкретної ситуації.

З перелічених, наступні характеристики можуть зберігатися в базі даних:

- площа водозбору ( $A$ ),  $m^2$ ;
- середні витрати води ( $R$ ),  $m^3/сек$ ;
- дистанція ( $D$ ),  $m$ ;

Характеристиками, що мають бути введені для коректного розливу є:

- витрати води в точці під час вимірювання ( $Q$ ) – вводиться для конкретної точки моніторингу, м<sup>3</sup>/сек;

- маса ( $M$ ) – вводиться для ситуації, мг. Для спрощення введення, модель повинна приймати значення в міліграмах, але введення варто виконувати в тонах. Для конвертації значення потрібно застосувати наступну формулу:

$$M = M_t * 1\,000\,000\,000, \quad (2.16)$$

де  $M_t$  – введена маса в тонах.

Таким чином, використовуючи цей алгоритм, моделювання може бути проведене в умовах декількох точок.

## 2.6. Алгоритм мультимаршрутизації

При використанні алгоритму багатоточковості, усі точки мають посилання на попередні, утворюючи при цьому русло річки, що не допускає явища розгалудження одного русла річки та з'єднання декількох гілок русла річки в одне. Відтак, існує необхідність в деякому алгоритмі, що включає можливість роботи з такого роду явищами.

Як і в алгоритмі багатоточковості, кожна точка має мати посилання на попередню. Розширенням є те, що кожна точка має посилання на декілька попередніх точок. У випадку з'єднання кожна перша точка після точки поєднання має декілька записів точок з одним індексом. Ці записи є рівними кількістю гілок, що мають відповідний опис для певної гілки та служать для адаптування характеристик до конкретної точки. У випадку розгалудження, кожна перша точка після точки розгалудження має посилання на попередню точку. Схематично, даний принцип показано на рис. 2.4.

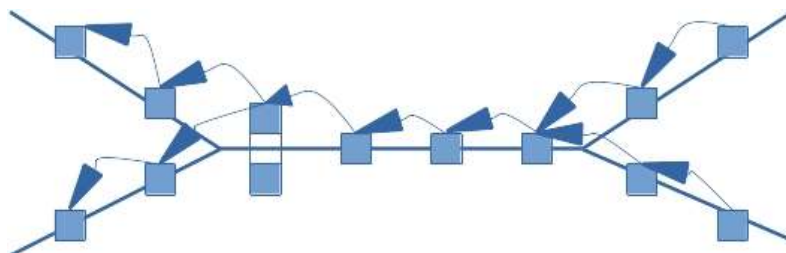


Рис. 2.4. З'єднання та розгалудження русла ріки, що утворюється з послідовності точок моніторингу за посиланнями

Враховуючи сенс характеристик, необхідних для конкретної гілки потрібно адаптувати лише таку характеристику, як дистанція від попередньої точки. Всі інші характеристики вже включають всю необхідну інформацію, що не потребує адаптації.

## **2.7. Коригування результатів прогнозування, шляхом використання задачі регресії**

Результати, отримані шляхом прогнозування рівня забруднення можуть мати похибку, що зв'язана з різними факторами характеристик самої точки моніторингу (наприклад, поворот русла, глибина русла, підводні течії і т.п.). Тому доцільно ввести коефіцієнт коригування результатів забруднення, тобто відкалібрувати результати. Очевидно, що для кожної точки такий коефіцієнт має своє значення, так як кожна має свої особливості місцевості, або враховує такі особливості вище за течією. Такий коефіцієнт має вираховуватись шляхом використання задачі регресії [34], що виводить коефіцієнт між реальними значеннями та прогнозованими. Також, цей коефіцієнт може змінюватись.

Таким чином, виникає необхідність в обчисленні та коригуванні значення коефіцієнту. Тому, розроблено нейронну мережу для коригування концентрації, метою якої є шляхом навчання виводити коефіцієнт калібрування результатів.

Така задача регресії має формулу:

$$Cp_{Cor} = Cp * k, \quad (2.17)$$

де  $Cp_{Cor}$  – реальна концентрація,  $Cp$  – прогнозована концентрація, що є одною з вихідних характеристик емпіричних рівнянь,  $k$  – коефіцієнт коригування прогнозованої концентрації, що виводиться нейронною мережею.

Серед факторів, що можуть впливати на результати забруднення є дані, які описують особливості самої локації та деякий динамічний фактор, що може впливати на забруднення з перемінним ефектом. Таким фактором може бути пора року.

Тому у якості вхідних даних можна ввести наступні:

- $x_1$  – фактор локації;

-  $x_2$  – фактор пори року.

Значення для вхідних змінних можуть мати значення 0 або 1, в залежності від того, чи даний коефіцієнт має бути враховано. Базуючись на вищеописаних факторах можна побудувати перцептрон, що працює на основі задачі регресії (прогнозовані дані до реальних).

Перший шар для кожної вхідної змінної повинен мати свій нейрон, що приймає рішення через функцію активації reLu [36] та виводить результат, використовуючи формулу:

$$W = \max(0, w * x), \quad (2.18)$$

де  $w$  – вага даного нейрону,  $x$  – чи даний фактор включається. Через те, що існують два фактори  $x_1$  та  $x_2$ , мають існувати два відповідні коефіцієнти  $w_1$  та  $w_2$ . Тому формула 2.18 справедлива для двох нейронів першого рівня.

Для виведення коефіцієнту регресії потрібно розробити нейрон другого рівня. Він приймає рішення на основі вхідних шарів попереднього рівня шляхом використання функції суматора:

$$W_n = \sum W. \quad (2.19)$$

В даному випадку, коефіцієнт, що буде виведено, базується на двох нейронах попереднього рівня та має формулу наступного виду:

$$W_n = W_1 + W_2, \quad (2.20)$$

де  $W_1$  – де коефіцієнт фактору локації,  $W_2$ - коефіцієнт фактору пори року. Результатом, а отже виведеним коефіцієнтом буде результат:

$$k = \sum W_n. \quad (2.21)$$

Схематично, дану нейронну мережу показано на рис. 2.5.

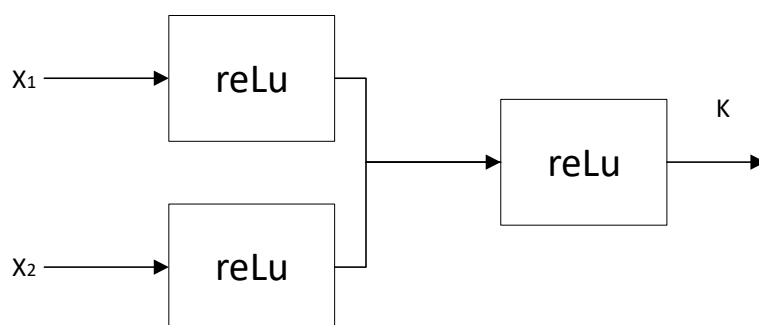


Рис. 2.5. Нейронна мережа для виведення коефіцієнту похибки

Коефіцієнти  $k = \sum W_n$  можуть бути проставлені експертами в межах від 0 до 1. Такий підхід є достатньо оптимізованим до швидкості передбачення, але не є гнучким та самокоригуючим. Через це, існує необхідність введення функції, що буде коригувати коефіцієнти відповідно до виведених реальних результатів. Тобто, потрібно використати метод оберненого розповсюдження помилки. В такому випадку, коефіцієнти можуть бути проставлені випадковим чином. Обернувши формулу 1.17 можна отримати реальний результат коефіцієнту для даної ситуації. Формула виведення реального коефіцієнту наступна:

$$k_{cor} = \frac{Cp}{Cp_{cor}}, \quad (2.22)$$

де  $k_{cor}$  – правильно виведений коефіцієнт. Якщо значення  $k_{cor}=1$ , або близьке до цього, то коригування помилки не є необхідним. На практиці, таке явище є достатньо рідким.

Отримавши результат можна коригувати всі коефіцієнти методом оберненого розповсюдження. При навчанні нейронної мережі потрібно включати всі фактори локації, а отже  $x_1$  та  $x_2$  повинні мати значення 1. Для отримання ваги для коректування коефіцієнтів при використанні функції активації reLu [36] використовується наступна формула:

$$\Delta w = \frac{k_{cor} * k}{2}. \quad (2.23)$$

Помилка для ваги нейрону вираховується за наступною формулою:

$$w_{error} = w * \Delta w, \quad (2.24)$$

де  $w$  – коефіцієнт конкретного нейрону. При використанні даної ваги можна отримати новий, більш правильний коефіцієнт за наступною формулою:

$$w_{new} = w - (\Delta w * lr), \quad (2.25)$$

де  $lr$  – коефіцієнт швидкості навчання, що виставляється емпіричним шляхом.

Схему даного перцептрону показано на рис. 2.6.



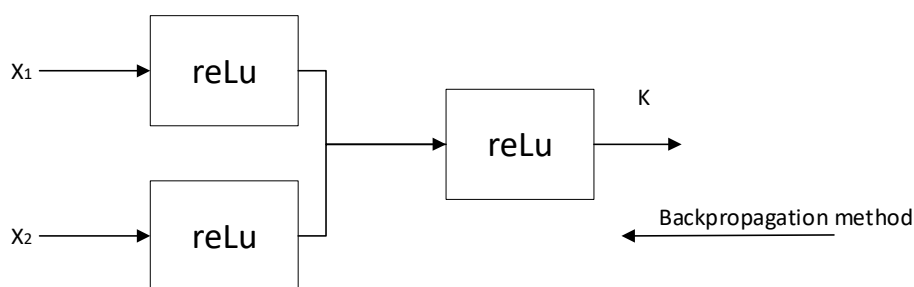


Рис. 2.6. Загальна архітектура перцептрону для виведення коефіцієнту похибки

Нейронна мережа має виводити різні значення для кожної точки і пори року, тому дані за фактором локації мають бути використані для кожної точки моніторингу і всіх пір року, а дані за фактором пори року мають бути виведені для всіх точок моніторингу і для кожної пори року.

Кількість даних, необхідних для конкретної точки моніторингу має бути розраховано експертним шляхом, спираючись на баланс гнучкості нейронної мережі та коректності коректування помилок. Тому в деякий момент, навчання нейронної мережі має бути виключене. Для витримки балансу при навчанні нейронної мережі, рекомендоване значення для ітерацій навчання – 5000 епох. Проте, таке значення повинне мати можливість коректування, так як може залежати від суб'єктивної логіки експерта та фактора спостереження за поведінкою нейронної мережі.

Отже, запропонований метод допускає обчислення коефіцієнтів коригування, що дозволяє коригувати результати прогнозування забруднення.

## 2.8. Використання інтерполяції для згладження прогнозованих результатів

Метод прогнозування забруднення є точковим, тобто дає результати для конкретних точок моніторингу. Такий підхід може відобразити ключову інформацію, але цього може бути недостатньо для відображення повної картини забруднення. Тому розроблено метод, що дасть можливість відображення даних не тільки в самих точках моніторингу, але й в інших точках русла в межах прогнозування. Це дозволить згладити графік та наповнити його більшою кількістю результатів. Необхідні точки повинні стояти між точками моніторингу.

Найбільш доречним методом для використання є метод інтерполяції результатів [37]. Загальна формула інтерполяції результатів має наступний вигляд:

$$y = \frac{1}{n} \sum_{i=1}^n y_i, \quad (2.26)$$

де  $y$  - шукане значення інтерполяції,  $n$  – кількість чисел, між якими проводиться обчислення,  $y_i$  – конкретне число, що включається для інтерполяції.

Задача інтерполяції має вирішуватись між двома точками, а від так, формулою для знаходження значення, що стоїть посередині між цими точками є:

$$Cp_{mid} = \frac{Cp_1 + Cp_2}{2}, \quad (2.27)$$

де  $Cp_{mid}$  – шукане значення концентрації посередині;  $Cp_1$  – значення концентрації попередньої точки моніторингу;  $Cp_2$  - значення концентрації наступної точки моніторингу. Враховуючи фактор того, що дані обчислено в статичному середовищі, аналогічно буде виглядати формула і для часу, через який дана точка набуде свого піку концентрації:

$$Tp_{mid} = \frac{Tp_1 + Tp_2}{2}, \quad (2.28)$$

де  $Tp_{mid}$  – шукане значення часу посередині;  $Tp_1$  – значення часу, через який точка набуде піку концентрації в попередній точці моніторингу;  $Tp_2$ - значення часу наступної точки моніторингу.

При використанні рекурсії для вищеописаних функцій можна отримати достатньо великий масив необхідних значень. Загальна формули для виведення значень в умовах рекурсії мають наступний вигляд:

$$y_{mid-} = \frac{y_1 + y_{mid}}{2}, \quad (2.29)$$

де  $y_{mid-}$  - шукане значення посередині попередньої та середньою точкою;  $y_1$  – деяке значення попередньої точки моніторингу,  $y_{mid}$  – деяке значення знайденої точки моніторингу посередині.

Аналогічно можна визначити дані між другою та середньою точками:

$$y_{mid+} = \frac{y_{mid} + y_2}{2}, \quad (2.30)$$

де  $y_{mid+}$  - шукане значення посередині від середньої точки до наступної;  $y_2$  – деяке значення наступної точки моніторингу.

Схематично, таку логіку можна показати у вигляді відрізка, де вихідними даними є точки на початку та в кінці відрізка, а першим шуканим значенням є середня точка. Після чого середня точка стає знайденою точкою між середньою і останньою, і першою і середньою, а середня точка між ними стає шуканою (рис. 2.7, рис. 2.8, рис. 2.9).

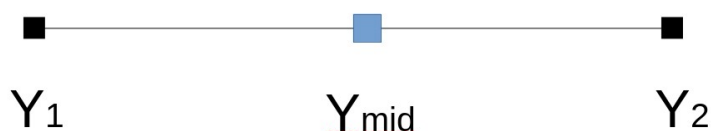


Рис. 2.7. Шукане значення точки між попередньою та наступною



Рис. 2.8. Шукане значення точки між середньою та наступною

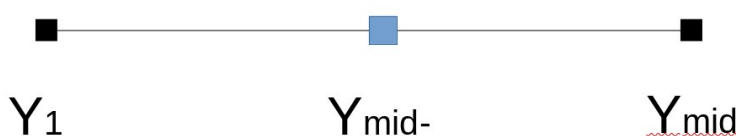


Рис. 2.9. Шукане значення точки між першою та середньою

Дистанція може бути границею кінця обчислень. Тому для дистанції інтерполяцію можна проводити наступним способом:

$$D = D_2 - D_1, \quad (2.31)$$

де  $D$  – різниця дистанцій;  $D_2$  – дистанція наступної точки від початку забруднення;  $D_1$  – дистанція попередньої точки від початку забруднення.

$$D = \frac{D}{2}. \quad (2.32)$$

Значення  $D = 200$  м може бути ознакою того, що даний рівень рекурсії є останнім. Таким чином, можна отримати масив результатів за такими наявними характеристиками, як:

- пік концентрації;
- час, за який концентрація набуде свого піку;

- дистанція.

Отже, запропонований метод дозволяє отримання результатів не тільки в самих точках моніторингу, але й в інших точках русла, або максимально до них наближених в межах прогнозування.

## **2.9. Визначення джерела забруднення**

Для конкретизації забруднення, дані про джерело забруднення можуть бути корисні через можливі особливості або характер забруднення.

Найбільш ймовірним забруднювачем може бути підприємство, внесене до переліку потенційно небезпечних об'єктів. Тому, виникає необхідність в розроблено метод, що дозволяє висувати гіпотези щодо причетності потенційно небезпечного об'єкту до деякого забруднення.

Найбільш ймовірним об'єктом-забруднювачем може бути той, що знаходиться найближче до точки виливу та працює з речовинами, що було вилито в русло ріки. Крім того, фактор наскільки часто це підприємство забруднює річку може бути другорядним. Оскільки основна модель є точковою, точкою виливу є точка моніторингу, від якої обчислюється забруднення.

Конкретні фактори, що можуть впливати на визначення джерела забруднення можуть бути різними та варіюватись між локаціями. Тому, для вирішення поставленої проблеми може бути створена база даних фактів, що асоціюють сутності між собою, що обробляються певним алгоритмом фільтрації.

### **База даних**

Враховуючи вищезазначену характеристику такого підприємства, можна виділити наступні сутності:

- потенційно небезпечний об'єкт;
- точка моніторингу;
- речовина;
- надзвичайна ситуація.

Фактором того, що потенційно небезпечний об'єкт відноситься до деякої точки моніторингу є зв'язка об'єкту до речовини. Фактором того, що точка моніторингу працює з визначеною речовиною є зв'язка об'єкту до речовини. Фактором того, що деяке підприємство стало причиною деякої надзвичайної ситуації є зв'язка підприємства до надзвичайної ситуації.

Вищезазначена база даних фактів має визначатися експертним шляхом. Таким чином, формуються наступні зв'язки між сутностями:

- зв'язка об'єкту до точки моніторингу;
- зв'язка об'єкту до речовини;
- зв'язка об'єкту до надзвичайної ситуації.

Схему зв'язків між потенційно небезпечним об'єктом та сутностей показано на Рис. 2.10.

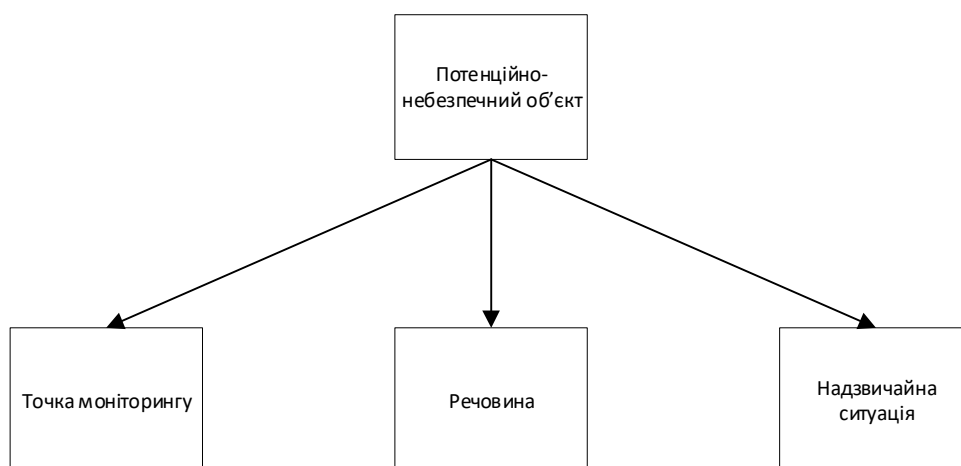


Рис. 2.10 Схema зв'язків між сутностями.

### Алгоритм фільтрації та сортування

В результаті формування бази даних фактів, необхідно використати алгоритм фільтрації, що складається з наступних етапів:

1. Фільтрація списку об'єктів, зв'язаних з точкою моніторингу.
2. Фільтрація сформованого списку потенційно небезпечних об'єктів по відношенню до речовини, що стала причиною надзвичайної ситуації.
3. Сортування сформованого списку потенційно небезпечних об'єктів до кількості надзвичайних ситуацій, що були спричинені ним.

Блок схему даного алгоритму показано на Рис. 2.11.



Рис. 2.11 Блок схема алгоритму фільтрації об'єктів

Отже, вихідним значенням є список об'єктів, що можуть бути забруднювачами, де перший запис у списку має найбільшу вірогідність, а останній - найменшу.

Таким чином, запропонований метод дозволяє висувати гіпотези щодо причетності потенційно небезпечного об'єкту до деякого забруднення, тобто джерела забруднення.

### **2.10. Автоматизація описаних методів та алгоритмів**

Вищеописані методи можуть бути автоматизовані та використані для розробки інформаційної системи прогнозування наслідків забруднень. При цьому, має підтримувати, як режим роботи з використанням інтернет з'єднання, так і можливість роботи на комп'ютерах, що не підтримують зв'язку з мережею.

### **2.11. Інформаційна система RESit**

В результаті розроблено інформаційну систему, що має клієнт-серверну архітектуру, де сервер – REST API сервіс, клієнт – настільний додаток, що має зв'язок з REST API.

REST - це архітектурний стиль для розподілених гіпермедійних систем [38]. API – прикладний програмний інтерфейс. Тобто REST API – прикладний програмний інтерфейс, що має реалізацію архітектурного стилю для розподілених гіпермедійних систем, в основі якого лежить протокол HTTP.

Дане програмне забезпечення має назву RESit, що означає River Emergency Situation. Отже, RESit – клієнт-серверна інформаційна система, що складається з комплексу застосунків та мають на меті вирішення проблеми прогнозування наслідків забруднень в умовах надзвичайних ситуацій.

### **2.12. Технології та підходи RESit**

При розробці RESit використовувались такі мови, як C# [39], Python [40], SQL [41]. Наступні відповідні платформи та технології були використані для покращення якості коду, використання існуючих рішень та покращення архітектури програмного забезпечення:

- SQL:

- MySql Server [41];

- Python [40]:
  - Django Framework [42];
  - Django Framework admin application [43];
  - NumPy [44];
  - PyMySQL [45];
- C#:
  - .NET Framework [39];
  - Windows Forms [46];
  - WPF [47];
  - GMaps [48];
  - MySQL.Data [41];
  - LiveCharts [49];
  - EPPlus [50].

Враховуючи складність логіки для автоматизації є необхідним використання переваг об'єктно орієнтованого та функціонального програмування, принципів SOLID [51], та таких шаблонів, як MVVM [52], MVP [53] та інших.

### 2.13. Структура RESit

Структурно, RESit складається з наступних сервісів та додатків:

- настільний додаток – програма для проведення симуляцій, що має можливість проведення обчислень за існуючими даними, отриманими з серверу. Також існує можливість роботи з методикою без використання будь-яких завантажених даних, ввівши при цьому перемінний масив вхідної інформації, однак деякий функціонал при цьому є недоступним;

- сервер – сервіс, що має на меті постачання настільного додатку оновленими даними та оновлення даних з адміністраторського та настільного додатку. Основною метою сервісу є забезпечення цілісності необхідних даних;

- адміністративний додаток – веб-додаток, що дозволяє проводити операцій додавання, зміни та видалення, тобто маніпулювання даними;



- сервіс навчання – консольний додаток, що дозволяє проводити навчання нейронної мережі за існуючою інформацією бази даних;

- база даних – працює на основі MySQL сервісу та відповідає за зберігання необхідних даних для системи.

Систему спроектовано для реалізації можливості використання схеми «багато до багатьох» при підключенні настільного додатку до серверу, тобто багато додатків мають можливість підключатись та працювати з даними одного серверу, а один сервер постачає дані для багатьох додатків.

Описані схеми показана на рис. 2.12 та рис. 2.13.

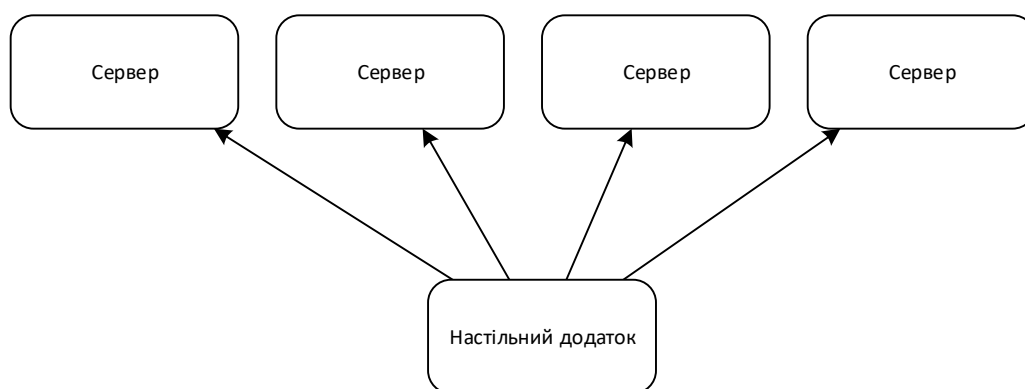


Рис. 2.12. Підключення настільного додатку до серверів

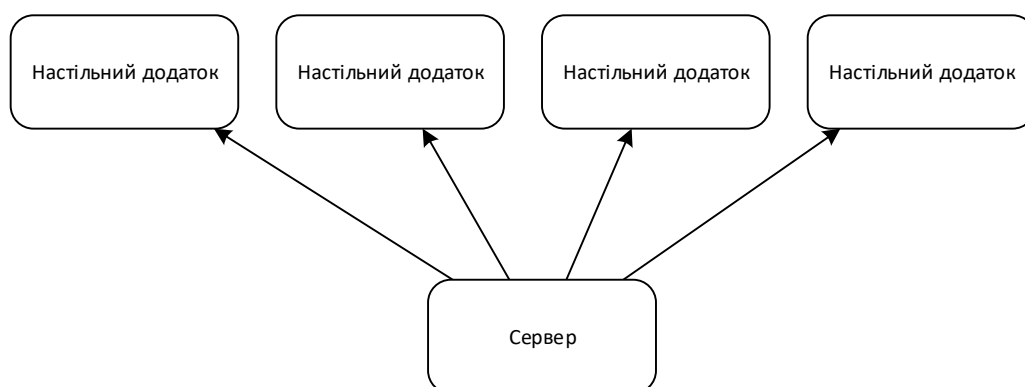


Рис. 2.13. Підключення серверу до настільних додатків

Оскільки сервер відповідає за забезпечення цілісності даних, проект схеми підключення передбачає взаємодію схеми «один до одного», тобто один сервер може підключитись до однієї бази даних та навпаки.

Схему показано на рис. 2.14.

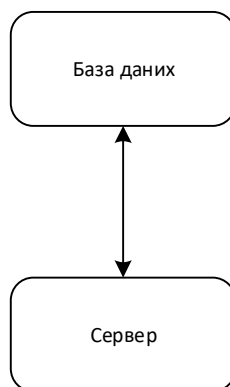


Рис. 2.14. Взаємодія серверу та бази даних

Утиліта навчання взаємодіє з сервером бази даних та вимагає, щоб сервер був зупинений. Такий підхід захищає цілісність даних при навчанні та оновленні відповідних коефіцієнтів.

Схему взаємодії утиліти навчання, серверу та бази даних показано на рис. 2.15.

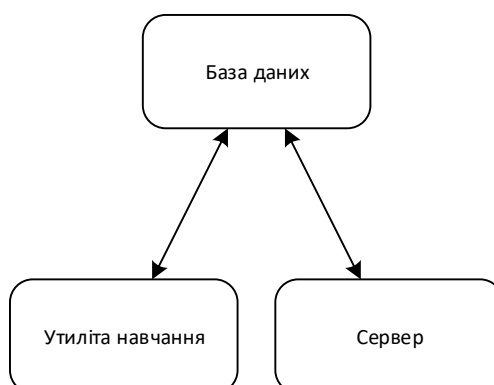


Рис. 2.15. Взаємодія серверу, утиліти навчання та бази даних

Загальну схему показано на рис. 2.16.

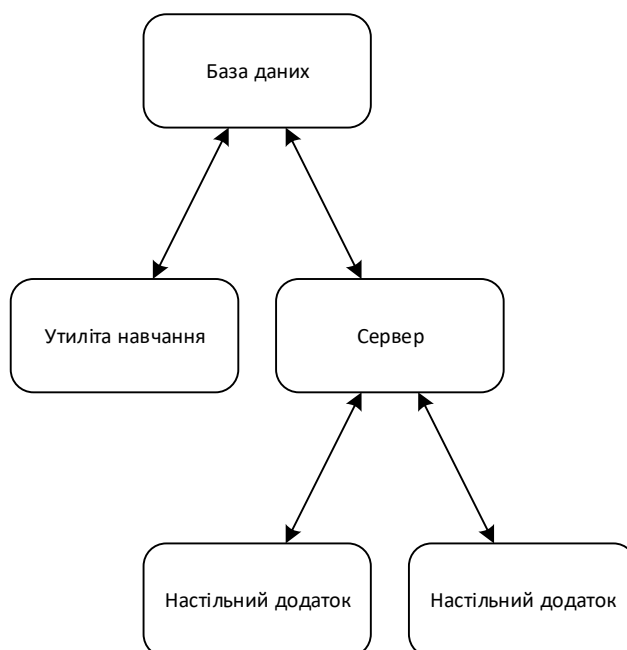


Рис. 2.16. Загальна схема роботи системи RESit

Локації серверу, утиліти навчання та бази даних на одному сервері дає ряд переваг, але при правильному налаштуванні вона не є обов'язковою.

## 2.14. База даних

Шар бази даних проєктовано з метою зберігання даних у структурованому вигляді. Система не передбачає роботи з великими даними, відтак технології, що побудовані на базі NoSQL не відповідають цим вимогам. Враховуючи складність даних, що мають бути збережені, платформи, в основі яких закладено декларативну парадигму та мову SQL можуть бути використані для вирішення проблеми. Проєктування реляційної бази даних, що має визначений рівень нормалізації є необхідною складовою для коректної побудови системи в цілому. У якості платформи для розгортання бази даних обрано MySQL, як один з найпопулярніших безкоштовних варіантів, що мають можливість масштабування та розширення системи. База даних також повинна мати міграційну систему, тобто систему внесення змін в структуру за міграціями. Первинні ключі таблиці є ідентифікаторами записів або сутностей. Вторинні ключі є посиланнями на записи

або сутності за їх ідентифікаторами. Для відображення проектованої схеми, найкраще використовувати EER діаграми [54].

## 2.15. Структура

Загальну схему проектованої бази даних відображено на рис. 2.17.

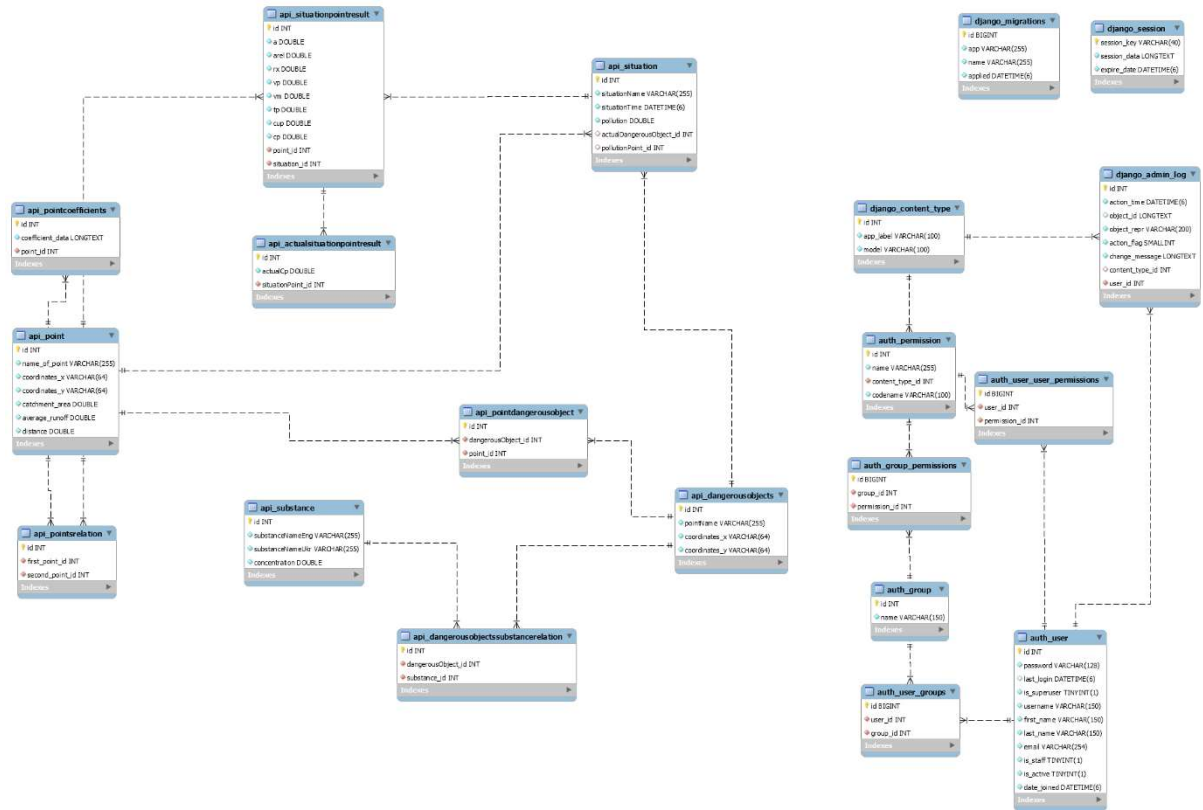


Рис. 2.17. Загальна схема проектованої бази даних

Як видно з рис. 2.17, база даних розділена на 3 основні складові:

- зберігання даних для роботи основної логіки системи;
- зберігання даних для коректної роботи сервісу адміністративної панелі;
- технічна таблиця для відслідковування системи міграцій.

Складова зберігання даних для роботи основної логіки системи має 10 таблиць (Рис. 2.18).

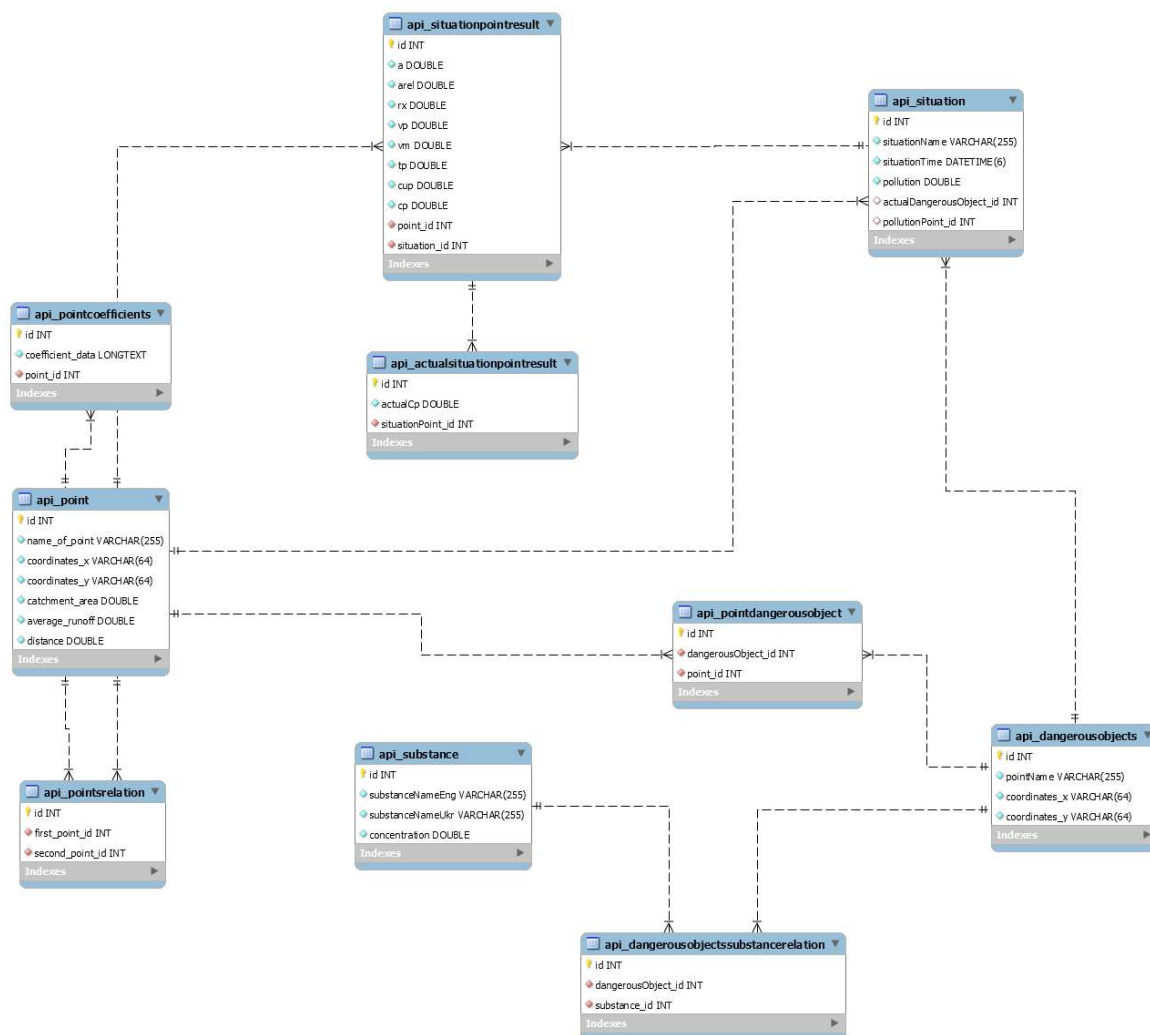


Рис. 2.18. Схема даних для роботи основної логіки системи

Функціональна частина має 6 частин:

- інформація про саму точку, маршрут;
- коефіцієнти похибки для визначеної точки;
- інформація про існуючі забруднення;
- перелік можливих речовин, що можуть бути викинуті у аналізованій локації;
- перелік можливих потенційно небезпечних об'єктів, що знаходяться на аналізованій локації.

Частина, що зберігає інформацію про саму точку та маршрут відображена в таблицях `api_point` та `api_pointrelation`.

Схему даної частини відображено на рис. 2.19.

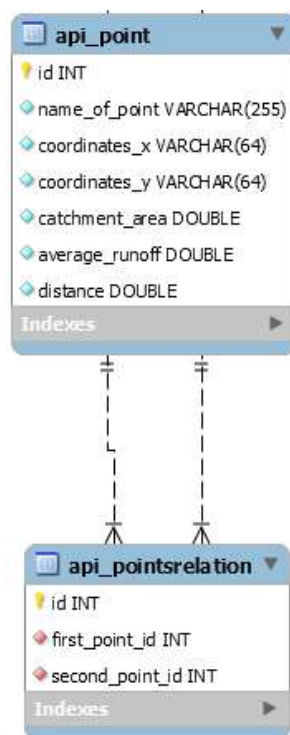


Рис. 2.19. Схема функціональної частини, що зберігає інформацію про точку та маршрут

Таблиця `api_point` призначена для зберігання географічних/гідрологічних характеристик точки моніторингу має наступні колонки:

- `id` – порядковий номер/ідентифікатор;
- `name_of_point` – ім'я;
- `coordinate_x` – географічна координата X;
- `coordinate_y` – географічна координата Y;
- `catchment_area` – гідрологічна характеристика, площа водозбору;
- `average_runoff` – гідрологічна характеристика, середні витрати;
- `distance` – дистанція від попередньої точки.

Технічна таблиця `api_pointrelation` призначена для зберігання даних, що використовуються при реалізації алгоритмів багатоточковості та мультимаршрутизації. Має такі колонки:

- `id` – порядковий номер/ідентифікатор запису;
- `first_point_id` – ідентифікатор попередньої точки;
- `second_point_id` – ідентифікатор наступної точки.

Частина, що зберігає коефіцієнти похибки для визначеної точки проєктована в таблиці `api_pointcoefficient`, як основа частини та `api_point` та базова складова.

Схему відображено на рис. 2.20.

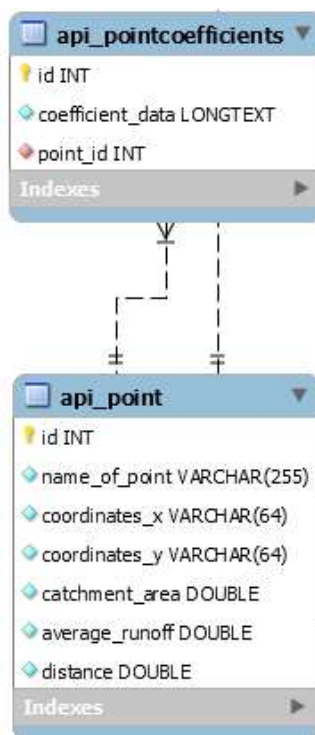


Рис. 2.20. Схема частини, що зберігає коефіцієнти похибки для визначеної точки

Технічна таблиця `api_pointcoefficient` має наступні колонки:

- `id` – порядковий номер/ідентифікатор;
- `coefficient_data` – значення коефіцієнту похибки;
- `point_id` – посилання на ідентифікатор точки таблиці `api_point`.

Частина, що зберігає інформацію про існуючі забруднення містить характеристики забруднення та детальну інформацію по кожній симульованій точці. Здебільшого, дана інформація необхідна для аналізу даних та навчання нейронної мережі. Для реалізації логіки даної частини використовуються 3 таблиці, названі як `api_situation`, `api_situationpointresult` та `api_actualsituationpointresult`.

Структурну схему можна побачити на рис. 2.21.

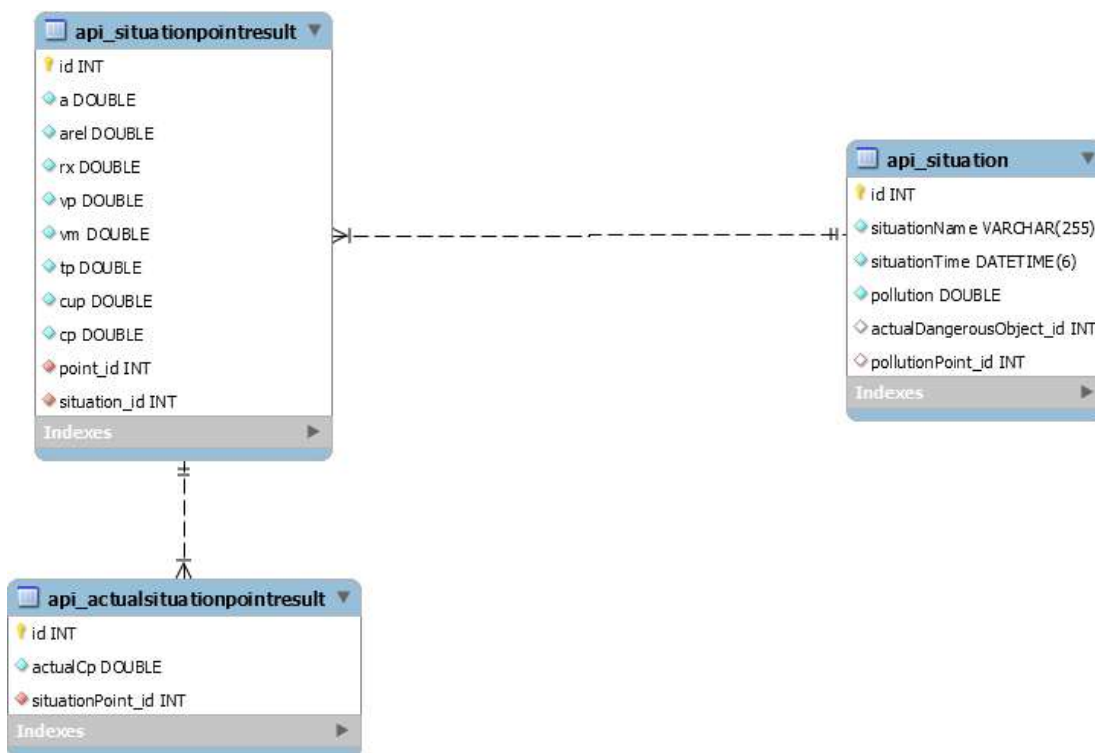


Рис. 2.21. Схема частини, що зберігає інформацію про існуючі забруднення

Таблиця `api_situation` призначена для зберігання характеристик самого забруднення. Колонки наступні:

- `id` – порядковий номер/ідентифікатор запису;
- `situationName` – умовне ім'я ситуації;
- `situationTime` – дата та час настання викиду;
- `pollution` – кількість речовини, що була вилита;
- `actualDangerousObject_id` – підтверджений потенційно небезпечний об'єкт забруднювач;
- `pollutionPoint_id` – точка, біля якої сталось початкове забруднення.

Метою таблиці `api_situationpointresult` є зберігання інформації про результати самого забруднення в конкретній точці. Має наступні колонки:

- `id` – порядковий номер/ідентифікатор запису;
- `a` – результуюча гідрологічна характеристика;
- `arel` – результуюча гідрологічна характеристика;
- `rx` – результуюча гідрологічна характеристика;
- `vp` – результуюча гідрологічна характеристика;
- `vm` – результуюча гідрологічна характеристика;



- tr - результуюча гідрологічна характеристика;
- sup - результуюча гідрологічна характеристика;
- sp - результуюча гідрологічна характеристика;
- point\_id – посилання на точку, для якої даний результат було обчислено;
- situation\_id - посилання на ситуацію, в рамках якої дана точка була забруднена.

Таблиця `api_actualsituationpointresult` використовується для зберігання інформації про реально підтверджені результати забруднення. Використовується для аналізу правдоподібності симуляції та навчання нейронної мережі, що виводить коефіцієнт похибки. Оскільки основною результуючою характеристикою є концентрація в аналізованій точці, підтверджені дані необхідні для цього результату. Задля можливості масштабування характеристик структура `api_actualsituationpointresult` була відокремлена, як окрема сутність, тобто таблиця. Колонки:

- id – порядковий номер/ідентифікатор запису;
- actualSp – підтверджене значення концентрації;
- situationPoint\_id – посилання на ідентифікатор запису результату самого забруднення в конкретній точці.

Перелік можливих речовин, що можуть бути викинуті у аналізованій локації реалізовано однією таблицею `api_substance`.

Схему показано на рис. 2.22.

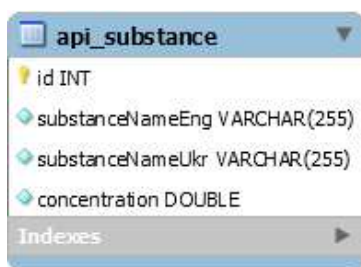


Рис. 2.22. Схема переліку можливих речовин, що можуть бути викинуті у аналізованій локації

Дана таблиця має такі колонки:

- id – порядковий номер/ідентифікатор запису;
- substanceNameEng – назва речовини англійською;

- substanceNameUkr – назва речовини українською;
- concentration – граничне допустиме значення концентрації.

Частина, що відповідає за функціонування зберігання переліку можливих потенційно небезпечних об'єктів, що знаходяться на аналізованій локації реалізована такими трьома таблицями, як api\_pointdangerousobject, api\_dangerousobjects, api\_dangerousobjectssubstancerelation.

Схему відображено на рис. 2.23.

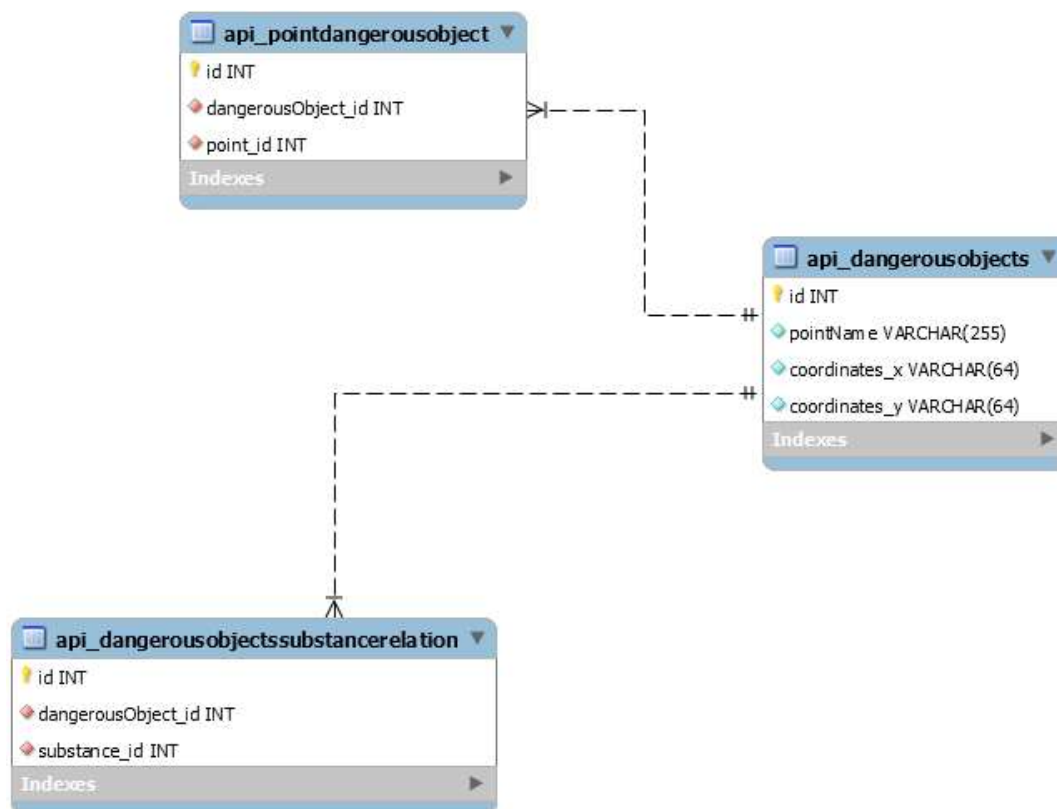


Рис. 2.23. Схема переліку можливих потенційно небезпечних об'єктів

Таблиця api\_dangerousobjects складається з переліку потенційно небезпечних об'єктів. Структура колонок наступна:

- id – порядковий номер/ідентифікатор запису;
- pointName – назва об'єкту;
- coordinates\_x – географічна координата x;
- coordinates\_y – географічна координата y.

Технічна таблиця api\_dangerousobjectssubstancerelation відображає факти можливості виливу потенційно небезпечним об'єктом визначеної речовини. Даний

взаємозв'язок є одним з двох алгоритмів, на основі яких працює алгоритм визначення забруднювача. Колонки наступні:

- id – порядковий номер/ідентифікатор запису;
- dangerousObject\_id – посилання на ідентифікатор потенційно небезпечного об'єкту;
- substance\_id - посилання на ідентифікатор забруднювача.

Технічна таблиця `api_pointdangerousobject` відображає факти взаємозв'язку між точкою забруднення та потенційно небезпечним об'єктом, що є одним з двох алгоритмів, на основі яких працює алгоритм визначення забруднювача. Колонки наступні:

- id – порядковий номер/ідентифікатор запису;
- dangerousObject\_id – посилання на ідентифікатор потенційно небезпечного об'єкту;
- point\_id - посилання на ідентифікатор точки моніторингу.

Складова зберігання даних, для коректної роботи сервісу адміністративної панелі згенерована автоматичними генераторами структури, що необхідна для роботи відповідних зовнішніх рішень. Ця складова включає дані для роботи механізму аутентифікації в адміністративну панель та панелі в цілому. Частина аутентифікації має приставку `auth_`. Приставка `_django` використовується для основної логіки та роботи сервісу в цілому.

Схема складової зберігання даних показана на рис. 2.24.

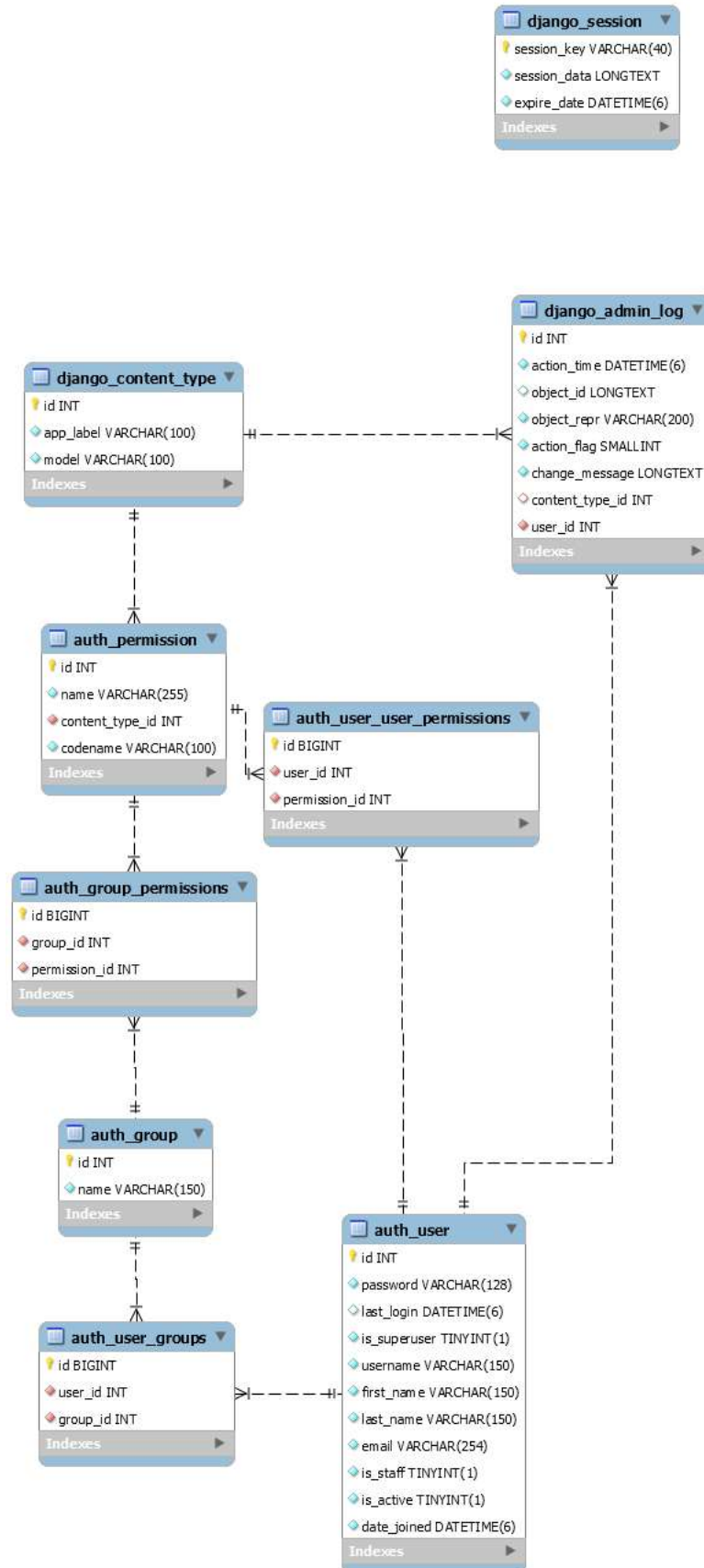


Рис. 2.24. Схема складової зберігання даних для коректної роботи сервісу адміністративної панелі

Технічна таблиця для відслідковування системи міграцій згенерована у вигляді однією таблиці. Її метою є відслідковування міграцій, що були включені.



Рис. 2.25. Схема технічної таблиці для відслідковування системи міграцій  
Схема показана на рис. 2.25.

## 2.16. Сервер

Сервером є сервіс, що постачає настільному додатку оновлену інформацію та оновлення цієї інформації з адміністраторського та настільного додатку. Основною метою сервісу є забезпечення цілісності необхідних даних. Цей сервіс було розроблено, використовуючи мову Python. Для спрощення та структурування використано бібліотеку-рішення Django Framework, як найбільше рішення, що може масштабуватись та підтримувати складну, нагромаджену логіку.

## 2.17. Функціонал серверу

Сервер має наступний функціонал:

- вибірка необхідних для настільного додатку даних з бази даних;
- конвертація форми та запис необхідних даних з настільного додатку в базу даних;
- можливість відслідковування даних з датчиків у режимі реального часу;
- взаємодія з адміністративною панеллю.

Сервер було проєктовано за принципом мінімального значимого продукту, у який було закладено мінімальний функціонал. Тому, існує можливість масштабування, як рівня складності так і кількості функціоналу.

## 2.18. Загальна архітектура

Серверну частину системи розроблено при використанні мови Python. У якості основної платформи, на якій побудовано сервіс використовується платформа Django.

Логіка такого рішення передбачає створення мікрододатків, що функціонують в рамках одного сервісу. Кожен додаток має свою зону відповідальності та свій функціонал. В сервіс вкладено два основних додатки:

- додаток для взаємодії з даними, що реалізує основну логіку роботи функціоналу з точки зору серверної частини. Основу роботи додатку представляє технологія REST Api, що дозволяє взаємодіяти з іншими частинами системи за допомогою HTTP запитів;

- адміністративний додаток, що дозволяє маніпулювати даними через веб додаток.

Сервер, як і інше програмне забезпечення такого виду реалізовує трьохшарову топологію, що передбачає інтерфейс взаємодії, логіку програмного забезпечення та логіку взаємодії з даними.

Інтерфейс взаємодії представляє REST Api, тобто взаємодіє через HTTP запити та веб-інтерфейс користувача. Додатки реалізують два шари з трьох.

Object Relation Mapping (ORM) – це принцип взаємодії з реляційними базами даних, використовуючи об'єктно-орієнтований підхід, тобто клас моделі представляє таблицю, поле – колонку таблиці. Зміни в структуру бази даних вносяться через міграції – окремі логічні частини, що при послідовну виконанні вносять зміни в базу даних. Міграції носять послідовний характер, а отже, можуть додаватися в ході підтримки програмного забезпечення. Самі міграції мають ім'я та час генерації. Від так вони можуть застосовуватись послідовно. Додаткова технічна таблиця в базі даних `django_migration`, зберігає функціональний журнал за міграціями, що застосовувались. Таким чином, сервер відслідковує поточний стан бази даних. Шар логіки взаємодії з даними реалізовано вбудованою в платформу Django ORM.

Сервіс має деякі конфігураційні файли, що дозволяють активувати та деактивувати додатки та налаштовувати саме програмне забезпечення.

Основу представляє додаток для взаємодії з даними. Ідеологічно, адміністративний додаток представляє окрему структуру. Тому, цей розділ описує логіку основи.

## **2.19. Побудова та функціонал шару з'єднання з базою даних**

Шар з'єднання працює на основі принципу ORM та представляє набір моделей міграцій та конфігурації підключення до бази даних. Всю основну логіку підключення виконує платформа `django.db`. Виходячи з розділу 2.15, у якому описується структура бази даних, в рамках цього шару реалізована частина зберігання даних для роботи основної логіки системи.

Від так, 10 таблиць представляють собою 10 відповідних класів моделей, а саме:

- `api_point`;
- `api_pointsrelation`;
- `api_pointscoefficient`;
- `api_situationpointresult`;
- `api_actualsituationpointresult`;
- `api_situation`;
- `api_substance`;
- `api_dangerousobjects`;
- `api_pointdangerousobject`;
- `api_dangerousobjectsubstancerelation`.

Міграції генеруються автоматичним способом, використовуючи команду `python manage.py makemigrations`. В новоствореній або існуючій папці `migrations` генерується додатковий клас, який в подальшому буде запущено платформою при запуску самого сервісу.

Основний файл конфігурації програмного забезпечення описує налаштування для підключення мікрододатків до бази даних. Для цього, в файлі `settings.py` описано об'єкт `DATABASES`.

Файл `__init__.py` використовується для виконання дій при запуску програмного забезпечення.

Він відповідає за запуск бібліотеки `mysql`, що є ядром для підключення будь-якої програми `python` до бази даних.

Схему, що описує архітектуру даного шару логіки взаємодії з даними показано на рис. 2.26.

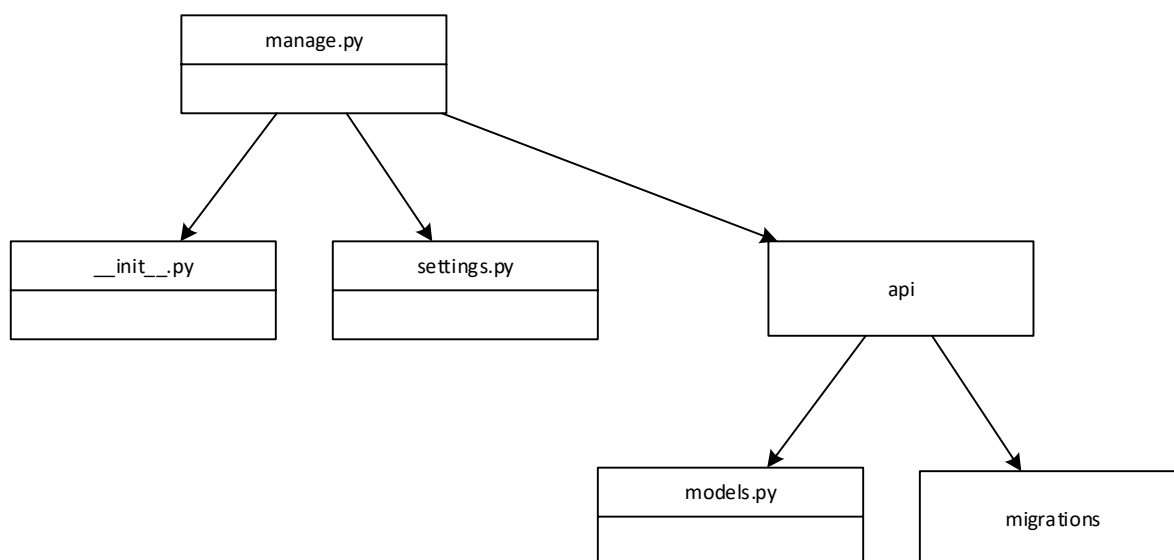


Рис. 2.26. Схема шару логіки взаємодії в даними

Отже, схема описує наступні файли:

- `manage.py` – файл, що є вхідною точкою для шару;
- `__init__.py` – файл, що запускається механізмом Django при запуску сервісу;
- `settings.py` – файл конфігурації;
- `api` – описаний додаток;
- `models.py` – описані класи моделі;
- `migrations` – міграції, що використовуються для підтримування стану бази даних.

Така архітектура спроектована з можливістю автономності та масштабованості.



## 2.20. Архітектура та функціонал REST API

Основою додатку є платформа Django, що за замовчанням реалізує роботу REST API за допомогою заготовленого сету файлів. Механізмом реалізації функціоналу є розширення двох механізмів:

- реєстрація кінцевих точок. Файлом, що використовується для реалізації логіки є `urls.py`. Основою даного механізму є прив'язка посилання до функції реалізації;
- реалізація кінцевих точок. Функції, що втілюють логіку роботи кінцевих точок реалізовані в файлі `views.py`.

Для спрощення формування моделей, що можуть бути прийняті та відправлені на сервер реалізовано додатковий набір таких представлень у файлі `apimodels.py`.

Загальну схему архітектури показано на рис. 2.27.

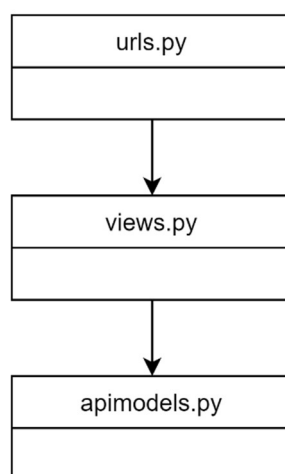


Рис. 2.27. Схема архітектури REST API

Функціонал реалізації кінцевих точок має підключеним шар взаємодії з базою даних, тобто в файлі `views.py` існує підключення файлу `models.py`. Для доступу функціоналу до бази даних існує Django рішення, що дозволяє виконати запит до бази даних. Запит на отримання всіх точок має наступний вигляд:

*Point.objects.all()*

де `Point` – вибір необхідної таблиці, `objects` – доступ до записів таблиці, `all()` – функція вибірки всіх записів.

Реалізований набір кінцевих точок показано в додатку А.

Використовуючи описаний набір кінцевих точок, настільний додаток отримує необхідні дані.

## 2.21. Адміністративний додаток

Адміністративним додатком є функціональна частина системи, що дозволяє проводити операцій додавання, зміни та видалення даних, що зберігаються у базі даних. За своїм типом програмного забезпечення застосунок є веб-додатком, тобто сайтом, що працює на базі та платформі серверу.

Метою функціональної частини є моніторинг та актуалізація даних системи.

Додаток реалізовано на базі налаштованого рішення Django Admin, що є частиною платформи Django.

Одиницями роботи з даними є таблиці бази даних. Основними функціями є:

- перегляд існуючих таблиць та записів;
- перегляд існуючих зв'язок між даними;
- додавання нових даних;
- додавання нових зв'язок між цими даними;
- редагування існуючих даних;
- редагування зв'язок між існуючими даними;
- «обережне» видалення застарілих даних, тобто запису або логічної зв'язки записів.

Оскільки додаток є існуючим рішенням, що перевикористовується, в різні частини сервісу серверу додані окремі частини коду, що налаштовують сервіс під роботу в рамках існуючого сервісу.

Як зазначалось, Django є масштабованою системою, що є сервісом, який складається з мікрододатків, а отже для початку потрібно підключити необхідні мікрододатки в загальну екосистему. Зробити це можна в файлі settings.py. До масиву `INSTALLED_APPS` потрібно додати наступний перелік основних додатків:

- `django.contrib.admin` – адміністративна панель;
- `django.contrib.auth` – функціонал авторизації;
- `django.contrib.sessions` – функціонал веб-сесій;

- `django.contrib.staticfiles` – можливість взаємодії зі статичними файлами, що зберігаються на сервері;
- інші можливі розширення.

Внаслідок включення мікрозастосунків, необхідна частина функціоналу буде підключена.

Реєстрація застосунку з екосистему REST API здійснюється в файлі `urls.py`.

Структура посилань та допоміжні методи імпортуються наступною частиною коду:

```
from django.contrib import admin
from django.urls import path
```

Реєстрація самої структури:

```
path('admin/', admin.site.urls)
```

В результаті застосунок буде доступний за абстрактним посиланням `http://%ipaddress%/admin/`.

Реєстрація моделей виконується у файлі `admin.py`.

Результатом вищеперерахованих місць є застосунок, що можна відкрити у браузері, виконати авторизацію та працювати з даними.

Інформативною складовою таблиць є їх текстове представлення. Таким представленням є метод `__str__()`, що реалізовано в файлах `models.py`.

Загальну схему показано на рис. 2.28.

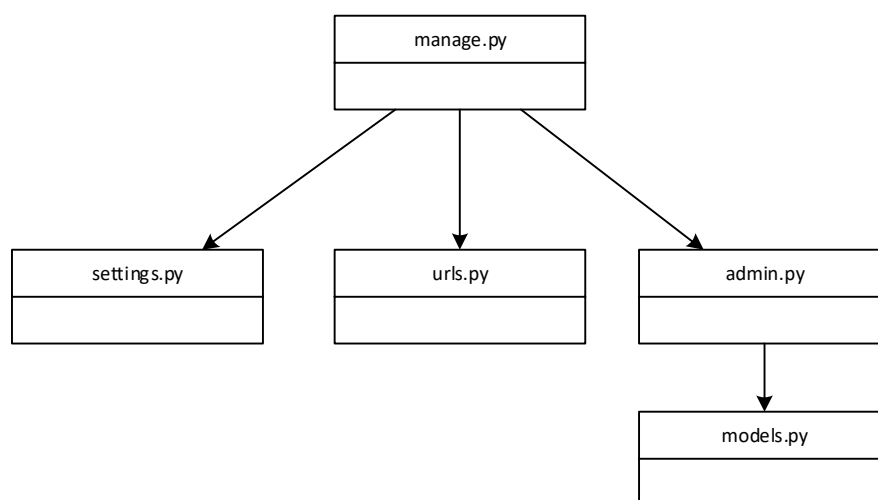


Рис. 2.28. Загальна схема частин додатку, що підключається

Підсумовуючи описану схему підключення та налаштування додатку можна зазначити, що використання зазначеного рішення є легким, гнучким та масштабованим способом для реалізації адміністраторської функції.

## 2.22. Графічний інтерфейс

Відповідно до виділеного функціоналу, розробниками був розроблений інтерфейс користувача, що дозволяє маніпулювати даними. Функціонал авторизації супроводжується інтерфейсом показаним на рис. 2.29.

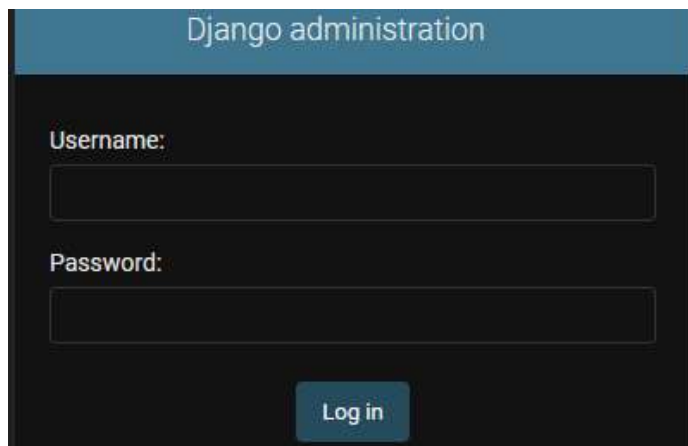
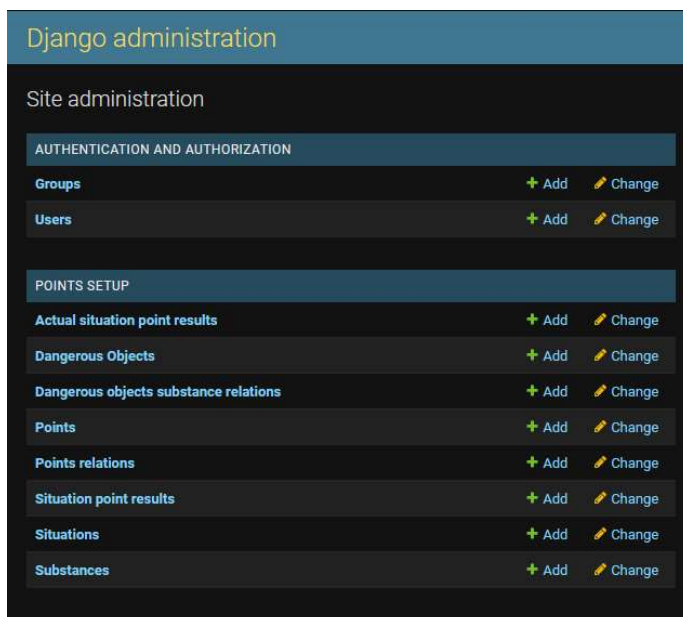


Рис. 2.29. Інтерфейс користувача адміністративної панелі

Це вікно потребує введення правильних ім'я користувача та паролю. Таким чином реалізована взаємодія з функцією авторизації.

Головне вікно супроводжується переліком сутностей/об'єктів, що зареєстровані в системі. Приклад даного графічного інтерфейс показано на рис. 2.30.



AUTHENTICATION AND AUTHORIZATION	
Groups	+ Add Change
Users	+ Add Change
POINTS SETUP	
Actual situation point results	+ Add Change
Dangerous Objects	+ Add Change
Dangerous objects substance relations	+ Add Change
Points	+ Add Change
Points relations	+ Add Change
Situation point results	+ Add Change
Situations	+ Add Change
Substances	+ Add Change

Рис. 2.30. Інтерфейс користувача головного вікна

Дані сутності розбиті на категорії відповідності до функціональної та частини системи та частини системи авторизації та аутентифікації користувачів.

Кожна сутність дозволяє виконувати такі маніпулятивні дії, як перегляд, додавання, редагування та видалення запису.

При додаванні нового запису має згенеруватись вікно, що дозволить введення полів, що генеруються динамічно. Приклад такого вікна показано на рис. 2.31.

Рис. 2.31. Приклад інтерфейсу користувача введення нового запису

Вікно дозволяє зберігати нові дані, після чого проводити додаткові дії. Формування зв'язок забезпечується за допомогою меню, що випадає. При цьому існує можливість додавання «дочірніх елементів», якщо їх не існує.

Варто відзначити, що відповідно до типу поля бази даних генерується свій елемент. Основними типами є числові, зв'язки, строкові та дата/час.

Редагування виконується в тому ж вікні, за виключенням того, що поля заповнені існуючими даними.

Також, інтерфейс підтримує функцію видалення запису. Функціонал перегляду записів реалізовано шляхом переліку заголовків записів з реалізацією можливості посторінкового виведення даних. Приклад такого переліку показано на рис. 2.32.

ACTUAL SITUATION POINT RESULT		
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	5Point: Tyachiv city - above city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	5Point: Sighetul Marmatiei
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	5Point: Rachiv city - bellow city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	5Point: Rachiv city - above city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	5Point: Jasinja settlement
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	4Point: Tyachiv city - above city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	4Point: Sighetul Marmatiei
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	4Point: Rachiv city - bellow city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	4Point: Rachiv city - above city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	4Point: Jasinja settlement
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	3Point: Tyachiv city - above city
2021-08-20 11:17:00+00:00	Jasinja settlement - Tyachiv city - above city	3Point: Sighetul Marmatiei

Рис. 2.32. Приклад інтерфейсу, що містить перелік записів

Заголовки записів формуються при виконанні методу `__str__` в відповідному класі в файлі `models.py`. Тобто, запис приводиться до строкового значення моделі.

Таким чином інтерфейс, що реалізовано на базі описуючого рішення покриває весь необхідний функціонал частини системи.

### **2.23. Взаємодія вимірювача з сервером**

Як описано у пункті 1.15, вимірювач взаємодіє з сервером, використовуючи комунікатор, що працює на базі Windows Mobile, або іншої мобільної операційної системи, або персональний комп'ютер, на якому встановлено визначену версію операційної системи Windows. Такий мобільний пристрій оснащений власницьким драйвером та програмним забезпеченням. Прикладами такого програмного забезпечення є QLiner PDA Software [30] та CDU Express [55]. Таке програмне забезпечення експортує дані у вигляді txt або xls файлів.

В цілому, можна виділити такі типи експорту даних:

- експорт у вигляді файлу з даними;
- експорт у вигляді HTTP запити;
- експорт у вигляді запити, що використовує будь-яку іншу технологію, що не сумісна з протоколом HTTP.

У випадку використання HTTP протоколу, вимірювач або комунікатор може здійснити POST запит на сервер за посиланням `api/setPointCurrentRunoff`.

Інші типи експорту вимагають розробки додаткового сервісу-адаптеру, що має спеціалізовану архітектуру.

### **2.24. Архітектура сервісу адаптеру**

Сервіс адаптер – деякий сервіс, що дозволяє перетворювати результат формату, що надають власницькі програмні продукти у дані, які необхідні серверу для постачання результату.

Архітектура програмного забезпечення складається з трьох шарів:

- отримання результату з сервісів – шар, що відповідає за зчитування/отримання даних з деякої сутності, з якою працюють власницькі

продукти (це може бути папка з файлами у спеціалізованому форматі, TCP або UDP клієнт, що постачає дані на сервер тощо);

- маніпулювання чергою результатів – основою даного шару є організація зберігання та доступу по черги задач, що готові до відправлення на сервер;

- надсилення даних на сервер – шар, що спостерігає за чергою та відправляє нову інформацію.

Загальну схему архітектури такого додатку показано на рис. 2.33.

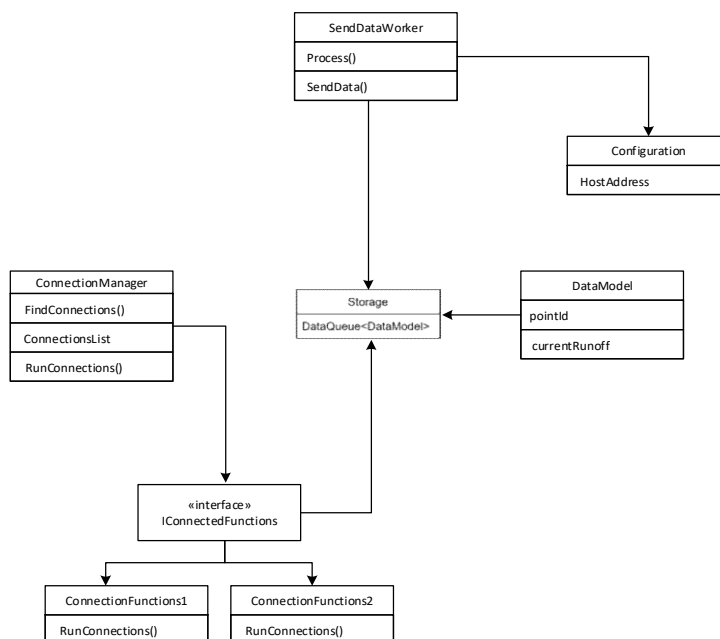


Рис. 2.33. Загальна архітектура програмного забезпечення

До шару отримання результатів з сервісів відносяться такі класи, як ConnectionManager та IConnectedFunctions. Метою класу ConnectionManager є робота зі з'єднаннями, тобто моніторинг та взаємодія між безпосередньою логікою інтеграції з власницьким програмний забезпеченням. Інтерфейс IConnectedFunctions є безпосереднім інтерфейсом, за яким ConnectionManager взаємодіє з класами, що описують логіку інтеграції. Інтерфейс має один метод – RunConnections(), який в середині, або відслідковує визначену папку, або є сервером для власницького, або іншого протоколу. Від інтерфейсу реалізуються конкретні реалізації, що описують конкретну логіку взаємодії. Кожен такий клас має бути запущений в окремому потоці. Отже, вхідними даними для цього шару є вхідні дані з шару взаємодії, вихідними даними є модель DataModel, що має бути додана в до шару маніпулювання чергою результатів.

Шар маніпулювання чергою результатів складається з двох класів Storage та DataModel. DataModel – модель що описує дані, які необхідно відправити на визначеною структурою. Даний клас має два поля – pointId та currentRunoff. Структура, що необхідна для відправки даних має аналогічний цій моделі вигляд. Клас Storage складається з черги, тобто структури, з якою необхідно взаємодіяти для формування черги даних для відправки.

Шар надсилання даних є окремим потоком, в якому має бути запущено метод Process в класі SendDataWorker. Логікою зазначеного методу є перевірка черги за таймером. Якщо в черзі є новий запис, то необхідно сформувати на відправити конфігурований HTTP запит на сервер та видалити його.

Інформація для конфігурації HTTP запиту зазначається в технічному класі Configuration, що є технічним для зчитування даних з деякої конфігурації сервісу за вибраною технологією.

Сервіс адаптер може бути розроблений використовуючи будь-яку технологію або мову програмування, але рекомендується використовувати мову C#, що працює на базі кросплатформової платформи .NET 5 або .NET 6. Такий підхід дозволить чітко структурувати описану архітектуру та масштабувати її.

## **2.25. Утиліта навчання**

Утилітою навчання є додаток, що взаємодіє з сервісом бази даних та виконує безпосереднє навчання нейронної мережі з метою виведення необхідних коефіцієнтів за існуючою інформацією, що вже зберігається у базі даних.

В системі, функціоналом, що виводить результат на основі статистичних даних є нейронна мережа, що побудована для коригування результатів прогнозування, шляхом використання задачі регресії. Тому первинно розроблена для навчання нейронної мережі коригування результатів прогнозування. Однак, її структура описана таким чином, що і інші методи можуть навчатися, тобто утиліта є доволі розширюваною. Результатом є коефіцієнти похибки результатів обчислення в конкретній точці, тобто коригування обчисленої характеристики пікової концентрації для конкретної точки.



Статистичними даними, тобто вхідним масивом даних для навчання є дані про ситуації по точках, що зберігаються. У якості характеристик для навчання використовується фільтрована інформація, що була відібрана з завантаженого перемінного масиву. Необхідні наступні характеристики:

- дата настання надзвичайної ситуації;
- реальна пікова концентрація в точці;
- прогнозована пікова концентрація в точці.

Ці дані необхідні для формування більш детальних масивів для проведення навчання.

В якості мови програмування, що використовувалась для розробки додатку використано мову програмування Python. Додатковими рішеннями, що використані для розробки програмного забезпечення є:

- NumPy – бібліотека для роботи з перемінним масивом даних та здійснення операцій над цим масивом;
- MySQL Connector Python – бібліотека для підключення утиліти до сервісу бази даних.

Модульно, додаток складається з двох компонентів та основного компоненту, що їх зв'язує:

- підключення до бази даних;
- ядро системи;
- запускатч.

Загальну схему показано на рис. 2.34.

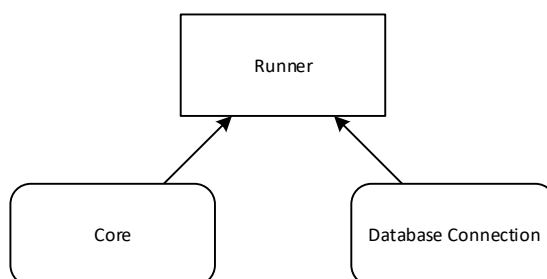


Рис. 2.34. Загальна схема компонентів додатку

Компонент підключення до бази даних складається з одного основного класу DatabaseInquirer, метою якого є виконання запитів до бази даних та моделі

ActualPredictPointModel, що описувала необхідний масив даних, необхідний для роботи додатку.

Діаграма класів для даного компоненту показана на Рис. 2.35.

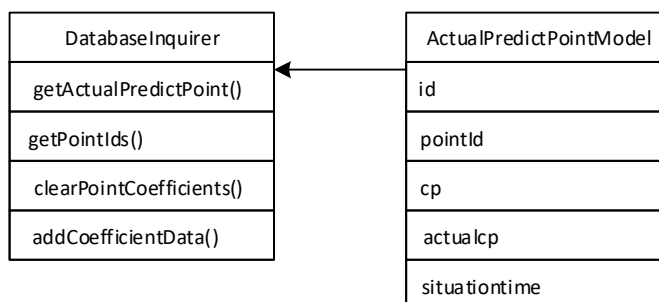


Рис. 2.35. Діаграма класів компоненту підключення до бази даних

Основна модель, як видно з рис. 2.35, складається з 5 характеристик:

- `id` – порядковий номер запису в базі даних;
- `pointId` – ідентифікатор номеру точки;
- `cp` - прогнозована пікова концентрація в точці;
- `actualCp` - реальна пікова концентрація в точці;
- `situationTime` - дата настання надзвичайної ситуації.

Клас **DatabaseInquirer** складається з необхідних запитів:

- `getActualPredictPoint()` – отримує повну модель даних в точці;
- `getPointIds()` – отримує список ідентифікаторів в точці;
- `clearPointCoefficients()` – очищує таблиці коефіцієнтів;
- `addCoefficientData()` – додає коефіцієнт навчання, тобто результат.

Компонент ядра складається з частини нейронної мережі та класу `xmlParser`, метою якого є формування xml виду результату, оскільки вихідними даними є комплекс коефіцієнтів в точці, а не лише одне число.

Схему ядра показано на рис. 2.36.

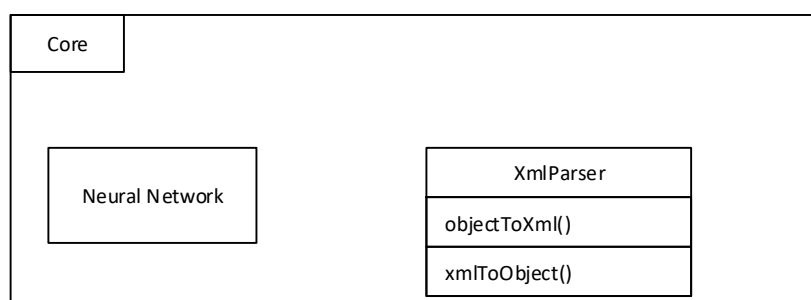


Рис. 2.36. Загальна схема ядра

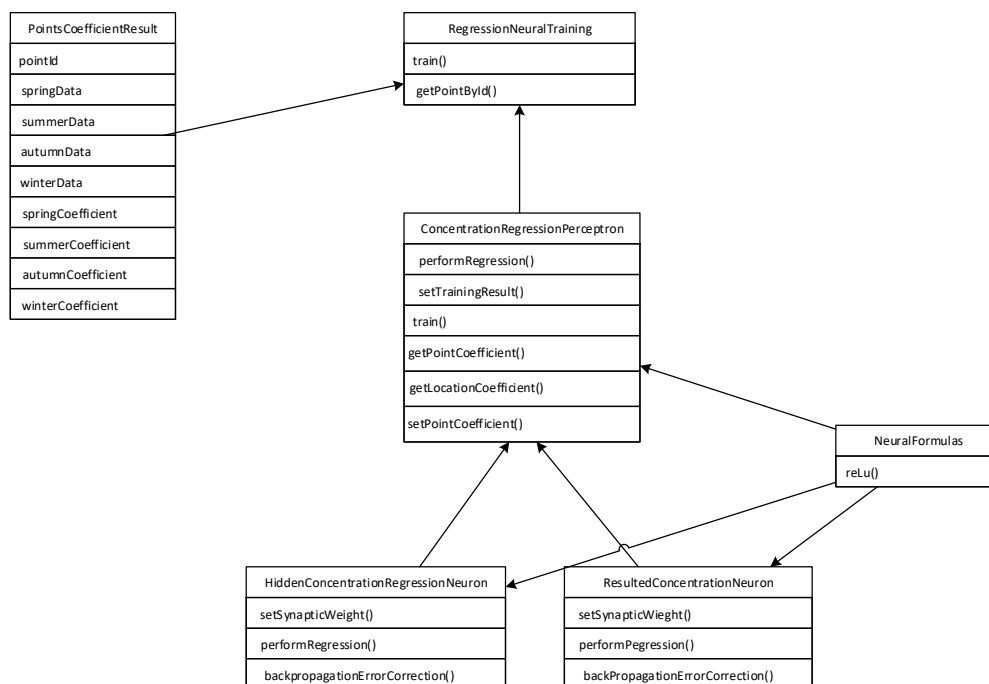


Рис. 2.37. Діаграма класів частини нейронної мережі

Частина нейронної мережі складається з класу, що описує нейрони прихованого шару нейронної мережі `HiddenConcentrationRegressionNeuron` та видимого шару нейронної мережі `ResultConcentrationRegressionNeuron`. Разом ці шари об'єднуються в перцептрон `ConcentrationRegressionPerceptron`. Також, існує допоміжний клас `NeuralFormulas` метою якого є опис загальних формул, що використовуються нейронною мережею. Для формування та зберігання даних, існує також модель `PointsCoefficientResult` метою якої є зберігання даних та результуючих коефіцієнтів за точкою.

Діаграма класів даної схеми показана на рис. 2.37.

Класом-обгорткою над нейронною мережею є клас `RegressionNeuralTraining`, що має 2 методи:

- `train()` – метод, що отримує список з вхідними даними та формує з них масив `PointsCoefficientResult`, попередньо групувавши дані за сезонами року у масиви `springData`, `summerData`, `autumnData` та `winterData`. Для кожного з цих масивів, ініціалізується свій екземпляр перцептрону та викликається метод `train()`, що тренує нейронну мережу та через метод `getPointCoefficient()` отримує результат та вносить його в поля `springCoefficient`, `summerCoefficient`, `autumnCoefficient` та `winterCoefficient` відповідно. Тренування може бути, як послідовним (тобто, за

масивом даних визначається коефіцієнт), так і ітеративно-вибірковим (тобто, для кожної точки вилучаються деякі коефіцієнти та застосовуються, як початкові коефіцієнти для інших масивів). При використанні ітеративно-вибіркового варіанту, коефіцієнти локації можуть бути вилучені та перенесені в ітерацію навчання за іншими сезонами. Для цього в перцептрон додані методи `getLocationCoefficient` та `setLocationCoefficient`. При цьому, задавши від 5 до 10 ітерацій навчання можна уникнути суттєвих похибок значень коефіцієнтів у випадку, якщо залежність, що має бути проаналізована нейронною мережею, не співпадає. Але у такому випадку, похибка може бути більшою, ніж при використанні послідовного навчання;

- `getPointById()` – повертає екземпляр класу `PointsCoefficientResult` за заданим ідентифікатором точки.

Клас `ConcentrationRegressionPerceptron` має наступні методи:

- `performRegression()` – запуск механізму виведення результатів для нейронів нейронної мережі за існуючими коефіцієнтами шляхом виклику `performRegression()` та задавання для них логіки суматора, що є основою нейронної мережі;

- `setTrainingResult` – внесення в перцептрон даних для навчання;

- `train()` – запуск механізму навчання перцептрону. При цьому відбувається запуск регресії, визначення та коректування похибки методом оберненого розповсюдження помилки;

- `getPointCoefficient()` – повертає загальний коефіцієнт нейронної мережі;

- `getLocationCoefficient()` – повертає коефіцієнт нейрону, що відноситься до навчання за характеристикою місцезнаходження точки;

- `setLocationCoefficient()` – задає коефіцієнт нейрону, що відноситься до навчання за характеристикою місцезнаходження точки.



Рис. 2.38. Алгоритм дій компоненту Runner

Класи `HiddenConcentrationRegressionNeuron` та `ResultedConcentrationNeuron` мають наступні методи:

- `setSynapticWeight()` – задає новий коефіцієнт для нейрону;
- `performRegression()` – запуск механізму виведення результатів для нейрону за існуючим коефіцієнтом. При цьому використовується метод `relu()` в класі `NeuralFormulas`;
- `backPropagationErrorCorrection()` – обчислює та повертає значення похибки за описаною формулами для прихованого та результуючого нейрону відповідно.

Клас `NeuralFormulas` має лише один метод `relu`, що обчислює дані за методом `ReLU`, але може бути масштабовано у випадку розширення функціоналу або складності додатку.

Компонент Runner має алгоритм дій, вказаний на рис. 2.38.

Описаний алгоритм дій використовує всі складові системи, шляхом зв'язування їх у логічну послідовність і таким чином виконує операції, що відповідають меті.

## **2.26. Настільний додаток**

Настільний додаток є програмним забезпеченням, що взаємодіє з сервером. Також, існує можливість автономної роботи. Метою даного програмного забезпечення є безпосередня взаємодія з системою. Зазначену утиліту побудовано для роботи користувача з даними та проведення обчислень в цілому.

## **2.27. Функціонал та призначення**

Основним призначенням настільного додатку є:

- проведення обчислень, використовуючи існуючу базу даних;
- проведення обчислень на основі деяких метаданих, введених вручну;
- проведення аналізу результату.

Виходячи з цього, основний та допоміжний функціонал наступний:

- проведення обчислень, використовуючи існуючу базу даних;
- проведення обчислень на основі деяких метаданих, введених вручну;
- синхронізація даних з сервером та проведення відповідних налаштувань;
- виведення результату в різних формах та видах;
- експорт результату в різних форматах.

Додаток розроблено на базі .NET Framework, використовуючи мову C#. Ряд існуючих рішень було використано для побудову логіки та роботи програмного забезпечення в цілому:

- EPPlus – рішення, що дозволяє експортувати дані у вигляді xls файлу;
- GMaps.NET – рішення, що дозволяє виводити необхідні дані у вигляді мапи, на яку накладено точки;
- LiveCharts – рішення, що дозволяє будувати діаграми та графіки;

- Newtonsoft.JSON – рішення, що дозволяє десеріалізовувати інформацію, що була отримана з бази даних;
- System.ValueTuple – деякі розширення мови C#.

Для розробки додатку використовувались такі технології, як Windows Forms та WPF.

## **2.28. Графічний інтерфейс та архітектура головного вікна**

Головне вікно є основним та стартовим вікном для роботи з додатком. Даний компонент побудовано на базі технології Windows Forms. При розробці, було використано шаблон проектування MVP, що передбачає розділення компоненту на три шари, що взаємодіють між собою за принципом «модель відображається у вікні через бізнес логіку». Вікно має наступний функціонал:

- створення обчислення, використовуючи базу даних;
- створення обчислення, ввівши дані вручну;
- експортувати результат;
- виведення результату в скороченому вигляді;
- введення даних вручну та формування результату;
- виклик вікна налаштувань серверів;
- виклик вікна виведення результату у повному вигляді.

У якості шару вікна виступає інтерфейс `IMainWindowView`, що реалізується класом `MainWindowView`, який і є самою розміткою вікна. Також, цей клас призначено для опису логіки роботи вікна. Класом логіки виступає `MainWindowsPresenter`, що має посилання на `IMainWindowView` та через інтерфейс взаємодіє з вікном. Модель, у такому випадку, не є обов'язковою складовою, а тому, вона замінюється на класи логіки нижчого рівня, що, в свою чергу мають власні моделі.

Діаграму класів компоненту показано на рис. 2.39.

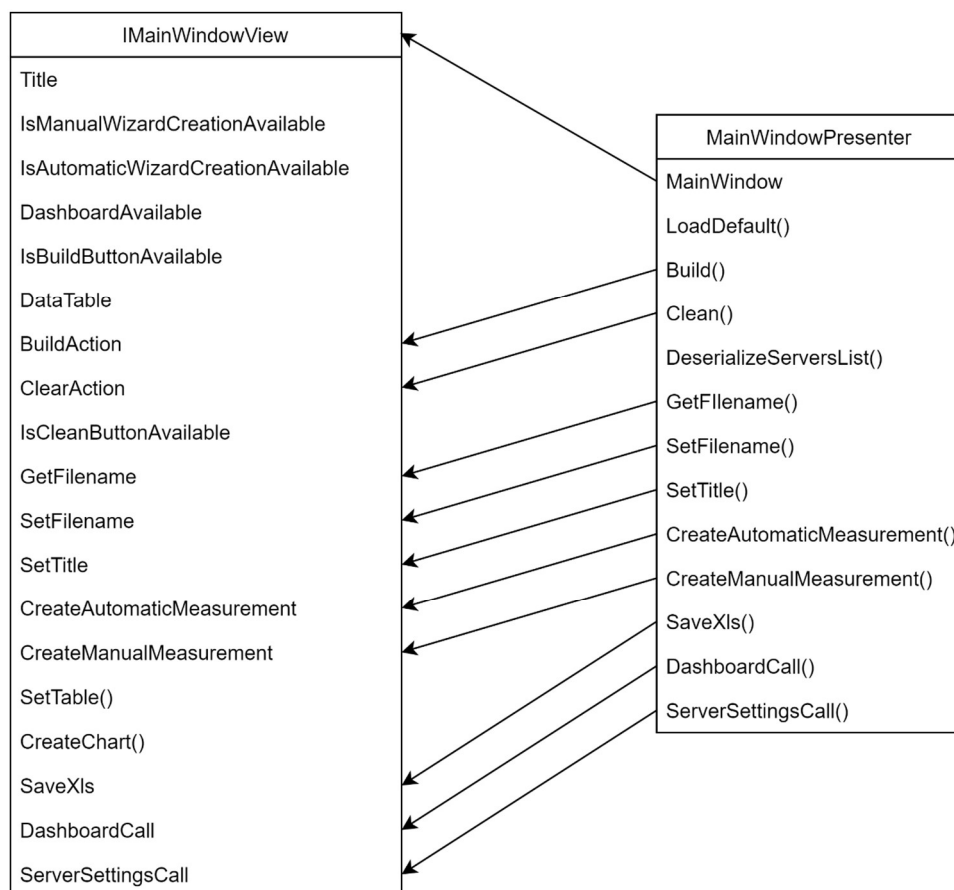


Рис. 2.39. Діаграма класів компонента головного вікна

Інтерфейс IMainWindowView має наступні властивості та методи:

- Title – властивість, що повертає назву вікна;
- IsManualWizardCreationAvailable – чи дозволено створювати ручне обчислення (тобто ввівши дані вручну);
- IsAutomaticWizardCreationAvailable – чи дозволено створювати автоматичне обчислення (тобто, обчислення даних з деякої бази даних, що була вивантажена з серверу);
- DashboardAvailable – чи дозволено відкривати дашборд (вікно виведення результату в повному вигляді);
- IsBuildButtonAvailable – маніпулює можливістю використання функції формування результату;
- DataTable – доступ до значень результату;
- BuildAction – подія, що викликається вікном для виконання функції формування результату;



- ClearAction - подія, що викликається вікном для виконання функції очистки результату;
- IsCleanButtonAvailable – маніпулює можливість використання функції очистки результату;
- GetFilename – подія, що викликається вікном для отримання імені файлу, за яким виконується експорт;
- SetFilename - подія, що викликається вікном для встановлення імені файлу, за яким виконується експорт;
- SetTitle – подія, що викликається вікном для встановлення нового заголовку вікна;
- CreateAutomaticMeasurement - подія, що викликається вікном для виклику проведення автоматичних обчислень;
- CreateManualMeasurement - подія, що викликається вікном для виклику проведення ручних обчислень;
- SetTable() – метод встановлення значень результату;
- CreateChart() – метод формування графіку на основі встановлених даних Table;
- SaveXls – подія, що викликається вікном для збереження даних у вигляді xls.
- DashboardCall - подія, що викликається вікном для виклику дашборда;
- ServerSettingsCall – подія, що викликається вікном для виклику налаштувань серверів.

Клас MainWindowPresenter має наступні властивості та методи:

- MainWindow – посилання на вікно;
- LoadDefault – метод завантаження первинного стану вікна;
- Build() – метод, що виконує логіку формування результатів;
- Clean() – метод, що виконує логіку очистки результатів;
- DeserializeServersList() – метод, що з локальних файлів збереження завантажує метадані та внутрішні дані, що були взяті з серверу;
- GetFilename() - отримання імені файлу, за яким виконується експорт;

- SetFilename() - встановлення імені файлу, за яким виконується експорт;
- SetTitle() - встановлення нового заголовку вікна;
- CreateAutomaticMeasurement() - виклик проведення автоматичних обчислень;

- CreateManualMeasurement() - виклик проведення ручних обчислень;
- SaveXls() – збереження даних у вигляді xls;
- DashboardCall() - виклик дашборда;
- ServerSettingsCall() – виклик налаштувань серверів.

Хоча класи і мають нагромаджену логіку, але побудовані за принципом «Single responsibility principle», тобто призначені тільки для однієї мети, що відповідає меті самого вікна – диспетчер для функціоналу програмного забезпечення.

Верстка головного вікна показана на рис. 2.40.

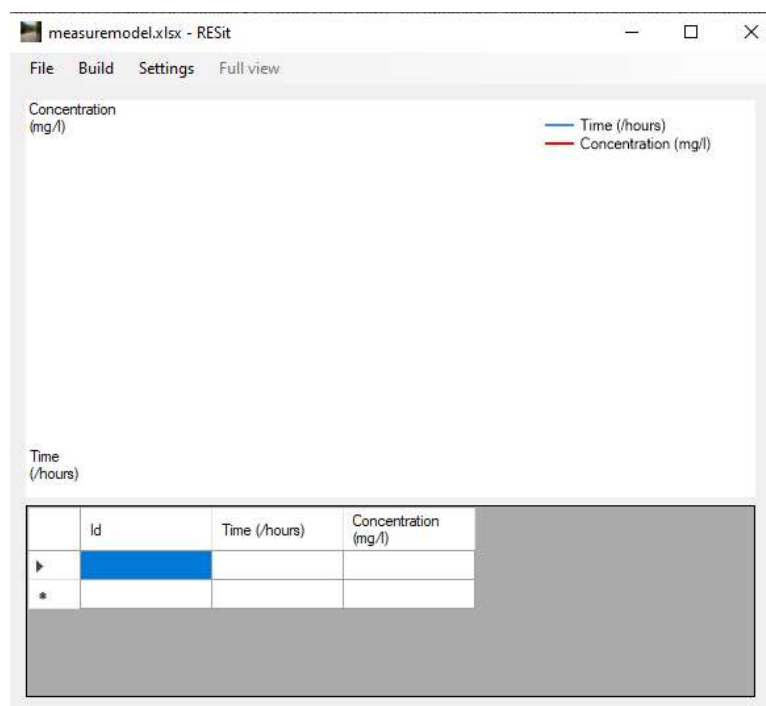


Рис. 2.40. Верстка головного вікна

Саме вікно ділиться на три компоненти:

- меню;
- графік зміни концентрації в динаміці, у кожній наступній точці;
- таблиця, що висвітлює числові значення графіків.

Меню має наступні кнопки:

- Файл

- Нове обчислення

- Ручний режим – при цьому викликається функціонал проведення ручних обчислень;

- Автоматичний режим – викликається функціонал проведення автоматичних обчислень;

- Експорт у

- XLS – експорт даних таблиці;

- JPS – експорт даних графіку;

- Побудувати

- Побудувати - формування результатів на основі введених в таблицю даних;

- Очистити – очистка будь-яких результатів;

- Налаштування - Налаштування серверу – виклик функціоналу налаштування серверу;

- Повний вигляд – викликається дашборд.

Виходячи з функцій вікна та визначених позицій меню можна описати два основні стани вікна:

- вікно виведення результатів ручних обчислень – досягається проведенням ручних обчислень або через функціонал формування результату;

- вікно виведення результатів автоматичних обчислень – досягається проведенням автоматичних обчислень;

- початковий стан вікна – стан вікна за замовчанням.

Програмне забезпечення побудовано таким чином, що об'єкт головного вікна є одним на весь життєвий цикл роботи додатку, при цьому воно лише має два стани – активний та неактивний.

## 2.29. Майстер проведення обчислень

Майстер проведення обчислень є деякою абстрактною сутністю, метою якої є формування платформи для створення майстрів проведення обчислень.

Основним принципом реалізації даної одиниці логіки є формування покрокових інструкцій у вигляді окремих методів, що сприймаються, як делегати. При ініціалізації абстрактного майстра формується список з делегатів, що мають виконатись. Запуском майстра є ітерація за зазначеним списком та виконання делегатів по черзі з перевіркою результатів. У випадку негативного результату або виключення – робота майстра зупиняється. Сутність має визначену абстрактну модель, що визначає необхідний масив даних, необхідний для повернення результату. Діаграму класів показано на рис. 2.41.

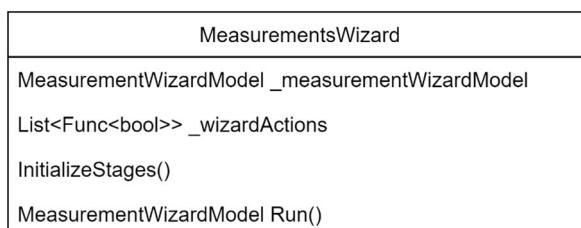


Рис. 2.41. Діаграма класів сутності майстра проведення обчислень

Зазначена логічна частина описана класом MeasurementsWizard. Набір методів та полів має наступний вигляд:

- `_measurementWizardModel` – внутрішня модель, що характеризує кінцеву модель результату;
- `_wizardActions` – список функцій, що повертають булеве значення. Характеризує список методів, що мають виконатись;
- `InitializeStages()` – метод, що характеризує кроки, необхідні для формування кінцевого результату;
- `Run()` – запуск сформованого списку делегатів та заповнення моделі.

Для перевикористання цього механізму, необхідно реалізувати дочірний клас від абстрактного класу MeasurementWizard та MeasurementWizardModel. Такий метод, як InitializeStages() є абстрактним, а тому вимагає конкретної реалізації. Клас моделі так само вимагає наслідування та присвоєння в реалізації.

Основним принципом роботи майстра є виклик форм та логічних методів, що вимагають покрокових інструкцій. Як результат, користувач взаємодіє з майстром та отримує результати обчислень.

Отже, даний механізм може бути розцінений, як деякий шаблон проектування, так як має визначений механізм, мету, реалізацію та шлях перевикористання.

Загальну діаграму класів шаблону проектування показано на рис. 2.42.

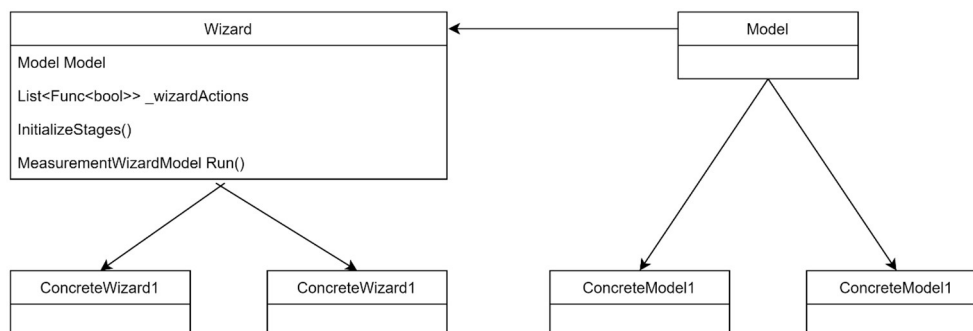


Рис. 2.42. Діаграма класів описаного шаблону проектування

Шаблон може дістати назву Wizard-Model-Result, що означає «модель виступає результатом після роботи майстра». Аббревіатура WMR є аббревіатурою до назви шаблону.

### 2.30. Майстер проведення ручних обчислень

Майстер проведення ручних обчислень розроблено с метою емуляції локацій для обчислення надзвичайних ситуацій, сервери для яких не були створені. У цьому випадку необхідно вказати умовний початковий та кінцевий ідентифікатори точок, а далі за інкрементним принципом ввести необхідну гідрологічну інформацію для кожної з запитаних точок.

Для взаємодії головного вікна з логікою майстра, використовується клас ManualWizardPresenter, що є ієрархічним класом логіки для класу логіки головного вікна. Моделлю для головного вікна, в цьому випадку, виступає ManualMeasurementWizardModel. Для опису логіки роботи майстра створено клас ManualMeasurementWizard, що працює на базі шаблону WMR. У якості кроків, що описують роботу майстра реалізовано класи MainForm, InputForm та HidrologicalPointsMeasurements.

Діаграму класів функціоналу показано на рис. 2.43.

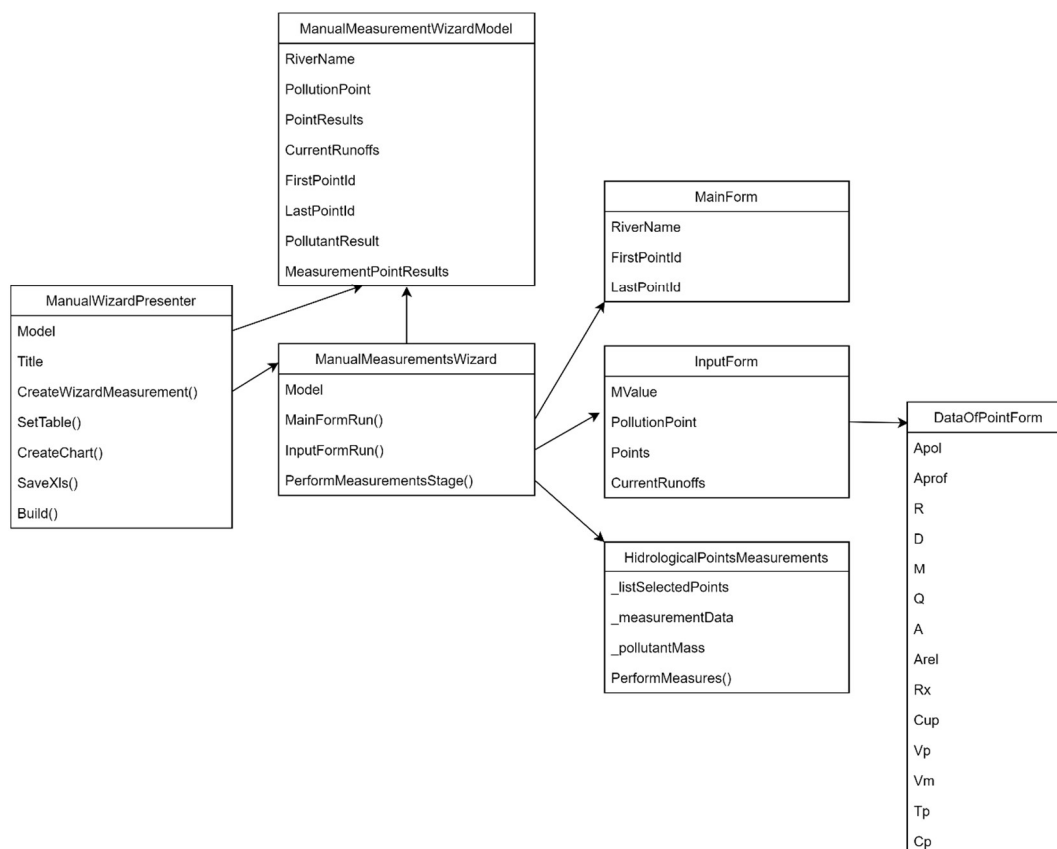


Рис. 2.43. Діаграма класів майстра проведення ручних обчислень

Клас ManualWizardPresenter має наступні поля та методи:

- Model – основна модель даної логіки, відображає необхідні характеристики;
- Title – відображає значення заголовку головного вікна;
- CreateWizardMeasurements() – метод, що викликається головним вікном для запуску майстра проведення обчислень;
- SetTable() – метод, що викликається головним вікном для заповнення даними таблиці, що знаходяться в моделі;
- CreateChart() - метод, що викликається головним вікном для заповнення графіку, що знаходяться в моделі;
- SaveXls() – метод, що дозволяє виконувати експорт результатів у вигляді xls файлу;
- Build() – метод, що формує модель на основі введених в таблицю результатів.

Метод ManualMeasurementWizard описує кроки, необхідні для роботи майстра. Має наступні методи:

- MainFormRun() – метод запуску початкового вікна введення інформації про точки та збору результатів у модель;
- InputFormRun() – метод запуску вікна введення гідрологічних даних за точками та збору результатів у модель;
- PerformMeasurementsStage() – метод виконання обчислень та збору кінцевого результату в модель.

Клас MainForm є формою введення даних про точки. Має наступні функціональні поля:

- RiverName – ім'я локації;
- FirstPointId – повертає порядковий номер початкової точки;
- LastPointId – повертає порядковий номер кінцевої точки.

Верстку початкового вікна показано на рис. 2.44.

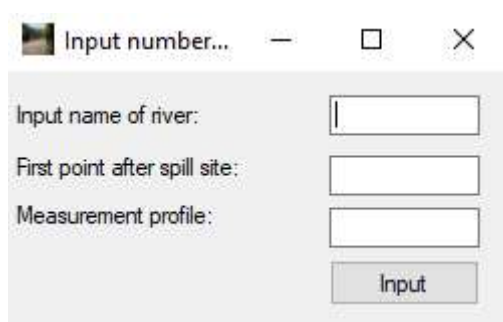


Рис. 2.44. Верстка початкового вікна

Логіка вікна побудована на трьох текстових полях та кнопці підтвердження введення даних.

Клас InputForm є формою введення гідрологічних даних за точками, що працює ітеративно. Має наступні функціональні поля:

- MValue – повертає значення маси забруднювача, що було викинуто;
- PollutionPoint – повертає інформацію про точку забруднення. У якості точки забруднення виступає перша точка;
- Points – список точок, для яких потрібно провести обчислення;
- CurrentRunoffs – список значень поточних витрат, що прив'язані до точок.

Введення даних для першої точки вимагає заповнення даних для точки забруднення. Тому введення даних про першу точку вимагає введення додаткових даних. Логіка вікна побудована на текстових полях та кнопці підтвердження

введення даних. Також існує можливість переглянути введені дані, прорахувати результат у режимі реального часу та експортувати дані у вигляді XLS файлу. Для цього на формі існує додаткова кнопка.

Форма введення гідрологічних даних показана на рис. 2.45.

Input your data of 2 point

Catchment area at spill site:  km2

Catchment area at measurement:  km2

Average runoff:  m3/s

Distance:  m

Pollutant mass:  t

Measured discharge at spill site:  m3/s

Count full data for this point

Input data

Рис. 2.45. Верстка форми введення гідрологічних даних

Клас DataOfPointForm має мету реалізації функціоналу обчислення результату в режимі реального часу та експортування даних у вигляді XLS файлу. Цей клас частково виконує необхідні обчислення, використовуючи шар математичного ядра. Всі функціональні поля даного класу є вхідними та вихідними характеристиками. Логіку форми відображення даних за точкою показано на рис. 2.46.

Data of point

Catchment area at spill site: 100000000000

Catchment area at measurement: 300000000000

Average runoff: 13

Distance: 2000

Pollutant mass: 10000000000

Measured discharge at spill site: 13

Arithmetic mean catchment area: 200000000000

Dimensionless area: 32207512210918,6

Dimensionless discharge: 1

Unit-peak concentrations: 408,199145794619

Velocity of the peak concentration: 0,209363137562194

Probable maximum velocity: 0,483274753393549

Probable traveltime of the peak: 2,65355000896718

Peak concentration: 31,3999342918938

Save in \*.xls

Рис. 2.46. Верстка форми відображення даних за точкою

Інформаційний сенс цього вікна полягає у виведенні всіх характеристик та збереженні їх у форматі xls файлу.



Клас `HidrologicalPointsMeasurement` є функціоналом, що виконує обчислення, використовуючи математичне ядро. Включає метод `PerformMeasures()`, метою якого є запуск проведення обчислень та повернення результату.

### 2.31. Майстер проведення автоматичних обчислень

Майстер проведення автоматичних обчислень створено з метою проведення обчислень, використовуючи існуючу базу даних, що була завантажена з вибраного серверу.

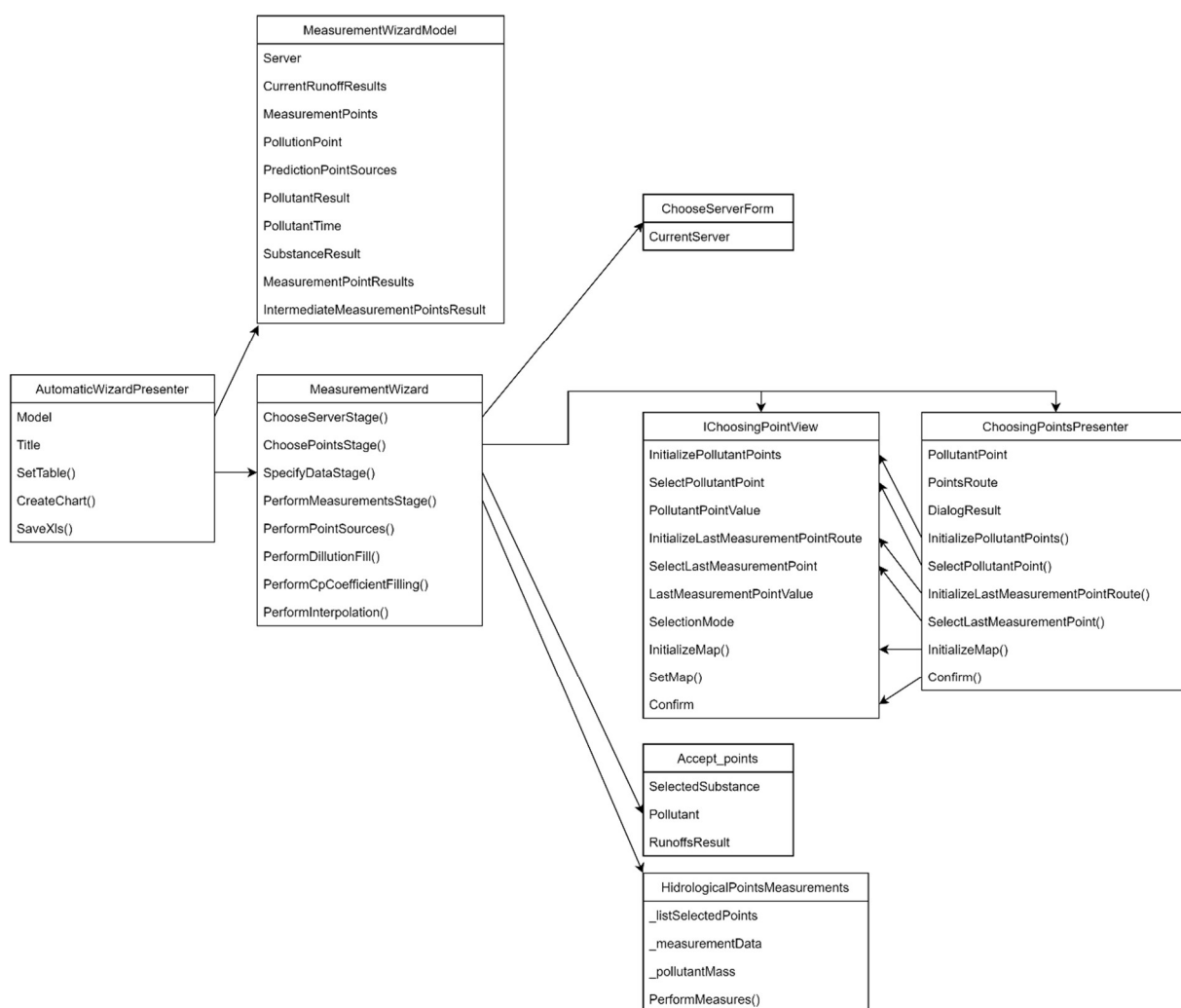


Рис. 2.47. Діаграма класів майстру проведення автоматичних обчислень

Для взаємодії головного вікна з логікою майстра, використовується клас `AutomaticWizardPresenter`, що є ієрархічним класом логіки для головного вікна. Моделлю для головного вікна, в цьому випадку, виступає `MeasurementWizardModel`. Для опису логіки роботи майстра істворено клас `MeasurementWizard`, що працює на базі шаблону WMR. У якості кроків, що

описують роботу майстра реалізуються класами ChooseServerForm, IChoosingPointView, IChoosingPointPresenter, Accept\_points та HidrologicalPointsMeasurements.

Діаграму класів функціоналу показано на рис. 2.47.

Клас AutomaticWizardPresenter має наступні поля та методи:

- Model – основна модель даної логіки, відображає необхідні характеристики;
- Title – відображає значення заголовку головного вікна;
- SetTable() – метод, що викликається головних вікном для заповнення елемента таблиці, що знаходяться в моделі;
- CreateChart() - метод, що викликається головних вікном для заповнення графіку, що знаходяться в моделі;
- SaveXls() – метод, що дозволяє виконувати експорт результатів у вигляді xls файлу.

Метод MeasurementWizard описує кроки, необхідні для роботи майстра. Має наступні методи:

- ChooseServerStage() – метод запуску початкового вікна вибору серверу бази даних;
- ChoosePointsStage() – метод запуску вікна, метою якого є вибір початкової та кінцевої точок забруднення;
- SpecifyDataStage() – метод запуску вікна введення необхідних характеристик надзвичайної ситуації;
- PerformMeasurementsStage() – метод виконання обчислень та збору кінцевого результату в модель;
- PerformPointSources() – метод визначення потенційно небезпечних об'єктів, що могли б забруднювати вибрану точку;
- PerformDillutionFill() – метод моделювання динаміки спадання забруднення в конкретній точці;
- PerformCpCoefficientFilling() – метод коригування результатів забруднювача коефіцієнтами, що були виведені задачею регресії;

- `PerformInterpolation()` – метою запуску алгоритму інтерполяції для існуючих результатів.

Клас `ChooseServerForm` має одне функціональне поле – `CurrentServer`, що повертає інформацію про вибраний сервер. Логіка вікна побудована на списку серверів, один з яких необхідно вибрати на кнопці підтвердження.

Логіку вікна показано на рис. 2.48.

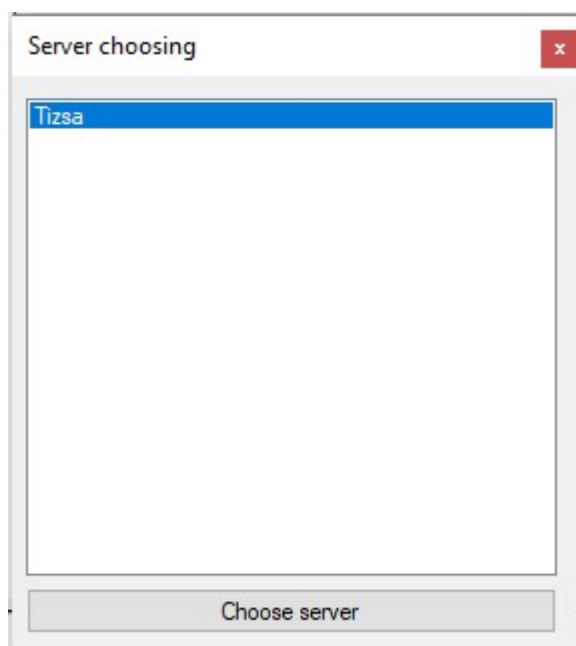


Рис. 2.48. Верстка форми вибору серверу

Класи `IChoosingPointView` та `ChoosingPointPresenter` використовуються для реалізації логіки форми. В основі реалізації функціоналу лежить шаблон MVP.

Інтерфейс `IChoosingPointView` описує метод та події, що стосуються самої форми:

- `InitializePollutionPoints` – подія, що реалізовує логіку ініціалізації списку точок;
- `SelectPollutantPoint` – подія, що реалізовує вибір точки забруднення;
- `PollutantPointValue` – посилення назви на вибраної точки забруднення;
- `InitializeLastMeasurementPointRoute` – подія, що ініціалізує маршрут точок на основі точки забруднення;
- `SelectLastMeasurementPoint` – подія, що вибирає останню точку вимірювання;

- `SelectionMode` – відображає режим вікна (початковий, вибрана перша точка, вибрана остання точка);

- `InitializeMap` – подія, що ініціалізує мапу;

- `SetMap` – метод, що ініціалізує точки на мапі;

- `Confirm` – подія підтвердження вибору.

Клас `ChoosingPointPresenter` складається з методів та подій, що реалізують логіку функціоналу:

- `PollutantPoint` - посилання назву на вибрану точку забруднення (результат);

- `PointsRoute` – маршрут від другої до останньої точки вимірювання (результат);

- `DialogResult` – інформація про те, чи був вибір успішним та підтвердженим;

- `InitializePollutantPoint` - метод, що реалізовує логіку ініціалізації списку точок;

- `SelectPollutantPoint` – метод, що реалізовує вибір точки забруднення;

- `InitializeLastMeasurementPointRoute` – метод, що ініціалізує маршрут точок на основі точки забруднення;

- `SelectLastMeasurementPoint` – метод, що вибирає останню точку вимірювання;

- `InitializeMap` – метод, що ініціалізує мапу;

- `Confirm` - метод підтвердження вибору.

Вікно побудовано з таких функціональних елементів, як ярлики вибраних точок (побудовано на основі текстового поля, що відображає назву поля та кнопки ініціалізації вибору нової точки), мапи для відображення та вибору даних графічним способом, кнопки підтвердження результату. Пусте та заповнене вікна відображаються на рис. 2.49 та рис. 2.50.

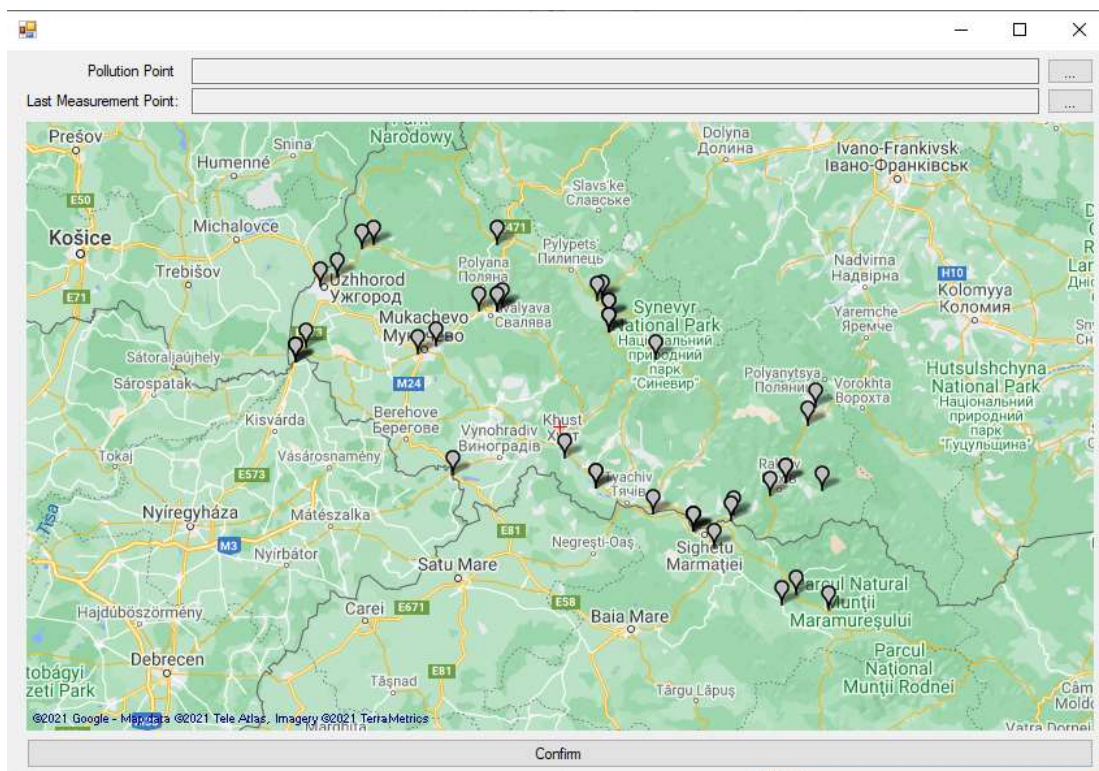


Рис. 2.49. Верстка форми вибору вибір початкової та кінцевої точки забруднення

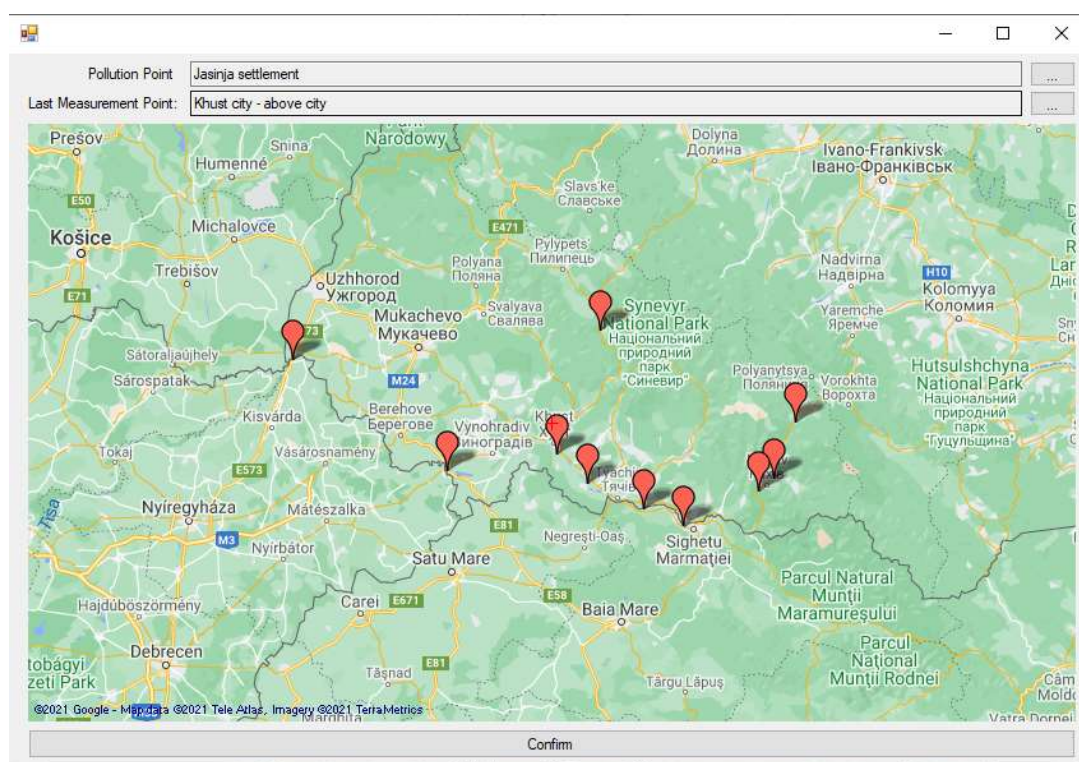


Рис. 2.50. Верстка форми вибору вибір початкової та кінцевої точки забруднення

Метою класу `Assert_points` є опис вікна та функціоналу введення необхідних характеристик надзвичайної ситуації. Клас має наступні функціональні поля:

- `SelectedSubstance` – повертає вибраного забруднювача;
- `Pollutant` – повертає введену масу забруднення;

- RunoffsResult – повертає список результатів поточних витрат води.

Вікно складається з текстового поля введення маси забруднювача, меню вибору речовини-забруднювача та таблиці результатів поточних витрат води, де запис відображає інформацію по конкретній точці. Інформація про поточні витрати формується на основі даних з серверу про поточні витрати води.

Верстку форми показано на рис. 2.51.

№	Name of point	Measurement discharge
4	Jasinja settlement	8.82
5	Rachiv city - abo...	26.9
6	Rachiv city - bell...	30
16	Sighetul Marmatiei	77.2
17	Tyachiv city - ab...	109
20	Tyachiv city - bel...	157
21	Khust city - abov...	164

Рис. 2.51. Верстка форми введення необхідних характеристик надзвичайної ситуації

Клас `HidrologicalPointsMeasurement` є функціоналом, що виконує обчислення, використовуючи математичне ядро. Містить метод `PerformMeasures()`, метою якого є запуск проведення обчислень та повернення результату. Інші такі методи, як `PerformMeasurementsStage()`, `PerformPointSources`, `PerformDillutionFill()`, `PerformCpCoefficientFilling`, `PerformInterpolation`, що описують кроки використовують класи з математичного ядра.

### 2.32. Дашборд: графічний інтерфейс та архітектура

Дашборд – це вікно, призначене для виведення результатів обчислення. Через комплексність результатів, має достатньо складну логіку. Результат обчислення та

додаткові дані, створені внаслідок інтелектуальної обробки результатів формують комплексний масив характеристик, що потрібно показати на інтерфейсі користувача, діляться на чотири категорії:

- результат забруднення за всіма точками;
- результат забруднення за всіма точками у вигляді графіку;
- інформація по забрудненню, що стосується конкретної точки;
- прогнозування можливих об'єктів, що забруднюють вибрану точку викиду.

Можливість відображення результату забруднення за всіма точками та між точками досягається використанням алгоритмом багатоточковості, алгоритмом мультимаршрутизації та методом інтерполяції для згладження результатів. Додатковою можливістю є коригування результатів забруднення для збільшення точності, що досягається відповідним методом.

Можливість відображення по забрудненню досягається шляхом алгоритму оцінки динаміки забруднення в конкретній точці.

Можливість прогнозування списку можливих об'єктів, що забруднюють вибрану точку викиду досягається алгоритмом визначення джерела забруднення.

При розробці даного вікна, використовуються одночасно технології WinForms та WPF. Таке рішення прийняте в результаті необхідності використання елементів користувацького інтерфейсу, що були присутні та оптимізовані в обидвох технологіях. Як результат, основними шаблонами для використання є HMMVM та MVC. HMMVM заснований на підході MVVM за виключенням ієрархічної структури вкладеності. Вікно та ViewModel при цьому складаються з менших логічних елементів. Дочірні елементи засновані на шаблоні MVP. У якості батьківських класів логіки виступають IMainDashboardWindow та MainDashboardViewModel. Інші класи Control та Presenter є дочірніми класами логіки.

Діаграма класів архітектури дашборду відображена на рис. 2.52.



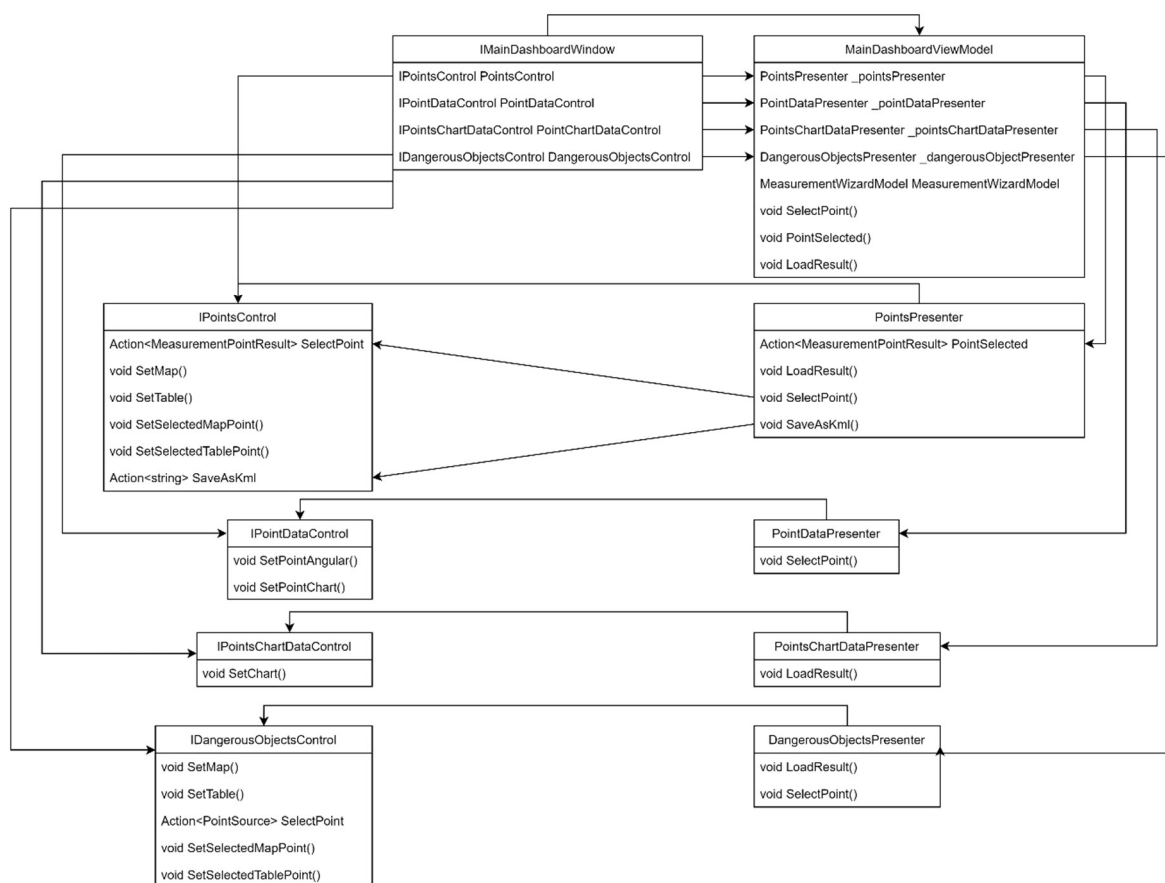


Рис. 2.52. Діаграма класів архітектури дашборду

Інтерфейс IMainDashboardWindow має наступні поля:

- PointsControl – посилання на тип компонента IPointsControl;
- PointDataControl – посилання на тип компонента IPointDataControl;
- PointsChartDataControl – посилання на тип компонента IPointsChartDataControl;
- DangerousObjectsControl – посилання на тип компонента IDangerousObjectsControl.

Клас MainDashboardWindow має наступні поля та методи:

- \_pointsPresenter - посилання на тип компонента PointsPresented;
- \_pointDataPresenter – посилання на тип компонента PointDataPresenter;
- \_pointsChartDataPresenter - посилання на тип компонента PointsChartDataPresented;
- \_dangerousObjectPresenter - посилання на тип компонента DangerousObjectPresenter;
- MeasurementWizardModel – модель результату;



- `SelectPoint()` – метод, що ініціює вибір точки на користувацькому інтерфейсі;
- `PointSelected()` – метод-оброблювач події, що виконує деякі дії після вибору точки;
- `LoadResult()` - викликається при завантаженні нової моделі.

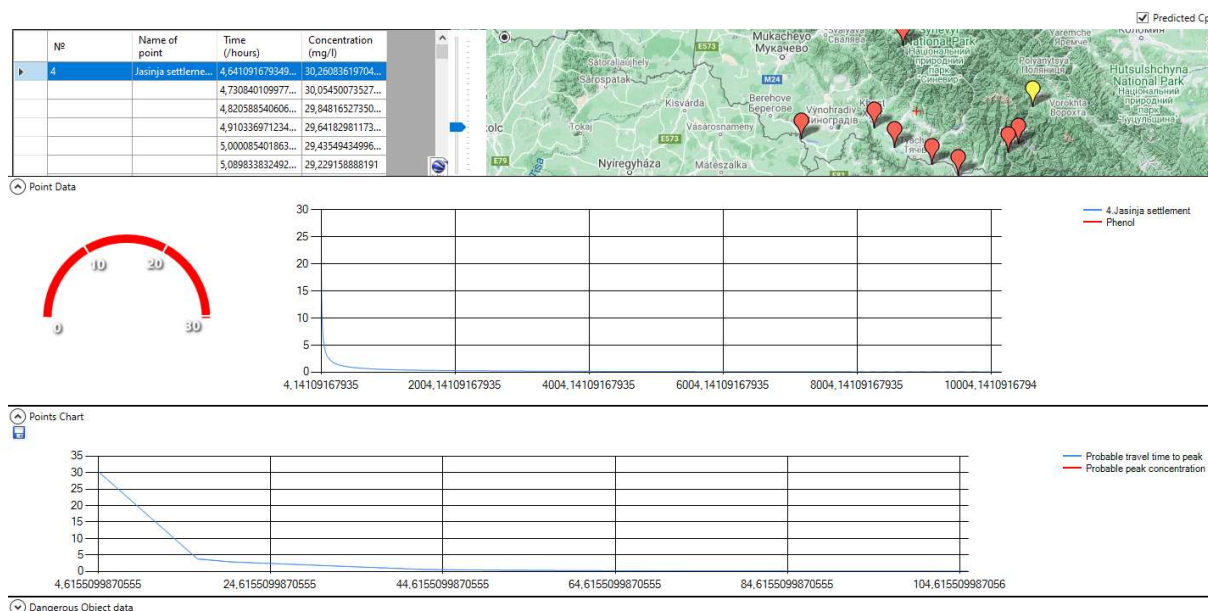



Рис. 2.53. Логіка проектування дашборду

Вікно спроектовано при використанні елементів `Expander` (елемент контейнер, що дозволяє приховувати вміст), що мають дочірні елементи. Логіку вікон показано на рис. 2.53.

Класи `PointsControl` та `PointsPresenter` реалізують логіку дочірнього елементу, що зберігає таблицю з численними значеннями результатів в точках моніторингу та між ними. Мапа показує точки моніторингу, задіяні для проведення моделювання. Має деякий функціонал, що дозволяє маніпулювати видом:

- масштабування мапи;
- фокусування на місцевості, натиснувши кнопку .

Логіку побудови інтерфейсу взаємодії з мапою показано на рис. 2.54.



Рис. 2.54. Інтерфейс взаємодії з картою

Клас PointsControl має наступні функціональні методи та поля:

- SelectPoint – подія ініціалізації вибору точки;
- SetMap() – метод завантаження точок на карту;
- SetTable() – метод завантаження даних в таблицю;
- SetSelectedMapPoint() – метод реакції мапи на зміну вибраної точки;
- SetSelectedTablePoint() – метод реакції таблиці на зміну вибраної точки;
- SaveAsKml – подія, що ініціалізується вікном для збереження даних у вигляді kml файлу.

Клас PointsPresenter має наступні функціональні методи та поля:

- PointSelected – подія, що ініціалізує вибір точки для всієї мапи;
- LoadResult() – метод, що завантажує результат;
- SelectPoint() – метод ініціалізації вибору точки;
- SaveAsKml – метод збереження даних у вигляді kml файлу.

Логіку даного логічного елемента показано на рис. 2.55.

№	Name of point	Time (/hours)	Concentration (mg/l)
4	Jasinja settleme...	4,641091679349...	30,26083619704...
		4,730840109977...	30,05450073527...
		4,820588540606...	29,84816527350...
		4,910336971234...	29,64182981173...
		5,00085401863...	29,43549434996...
		5,089833832492...	29,229158888191

Рис. 2.55. Логіка інтерфейсу користувача певного логічного елемента

Класи IPointDataControl та PointDataPresenter відтворюють графічні дані за вибраною точкою. Форма будується на графіку, що відображає концентрацію для

точки та наскільки таке значення перевищує безпечне на спадані концентрації в динаміці.

Верстка даного компоненту відображена на рис. 2.56.

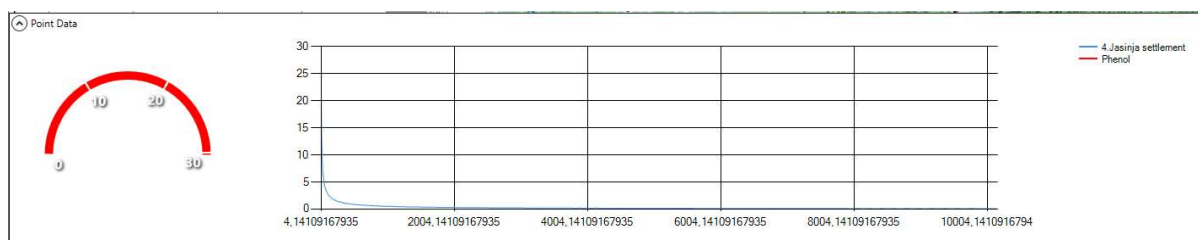


Рис. 2.56. Верстка компоненту відображення графічних даних за вибраною точкою

Інтерфейс IPointDataControl відповідає за логіку роботи самого компоненту та має наступні функціональні методи:

- SetPointAngular – присвоює круглому графіку значення;
- SetPointChart – присвоює графіку значення динаміки в конкретній точці.

Клас PointDataPresenter має один метод SelectPoint, метою якого є опис логіки присвоювання вибраної точки.

Компонент, що описують класи IPointChartDataControl та PointChartDataPresenter призначені за виведення графіку з численними значеннями результатів в точках моніторингу та між ними. Елемент складається з графіку та кнопки збереження його у вигляді jpg файлу.

Приклад форми показано на рис. 2.57.

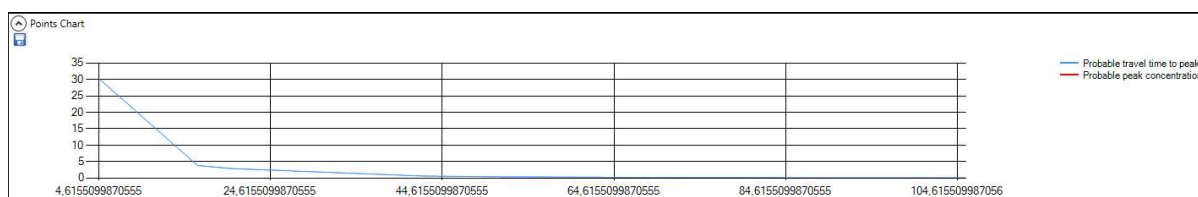



Рис. 2.57. Верстка компоненту виведення графіку з численними значеннями результатів в точках моніторингу

Інтерфейс опису користувацького інтерфейсу IPointChartDataControl метод SetChart, метою якого є формування графіку на основі переданих даних.

Клас PointChartDataPresenter має метод LoadResult, що має на меті опис логіки завантаження даних для виведення.

Компонент, що виводить інформацію про прогнозування можливих об'єктів, що забруднюють вибрану точку викиду описується класами IDangerousObjectsControl та DangerousObjectsPresenter. Логіка форми побудована на таблиці з описом небезпечних об'єктів та мапі, що візуалізує кожен об'єкт. Мапа має функціонал, що дозволяє маніпулювати видом:

- масштабування мапи;
- фокусування на місцевості, натиснувши кнопку .

Приклад верстки форми показаний на рис. 2.58.



Рис. 2.58. Верстка компоненту, що виводить інформацію про прогнозування можливих об'єктів

Інтерфейс IDangerousObjectsControl описує функціональну логіку форми та має наступні поля та методи:

- SelectPoint – подія ініціалізації вибору точки;
- SetMap() – метод завантаження точок на мапу;
- SetTable() – метод завантаження даних в таблицю;
- SetSelectedMapPoint() – метод реакції мапу на зміну вибраної точки;
- SetSelectedTablePoint() – метод реакції таблиці на зміну вибраної точки.

Клас DangerousObjectsPresenter має наступні функціональні методи та поля:

- LoadResult() – метод, що завантажує результат;
- SelectPoint() – метод ініціалізації вибору точки.

Описані компоненти логічно собою зв'язані функціоналом. Таким функціоналом може бути вибір конкретної точки або приховування та показ окремих компонентів. При цьому інші компоненти підлаштовуються за розміром до вільного місця.

### 2.33. Вікно налаштування серверів

Клієнтський функціонал може працювати з даними, завантаженими з серверу. Відповідний функціонал було реалізовано, як частина додатку. Даний функціонал має два шари:

- механізм, що описує логіки завантаження даних з серверу;
- вікно налаштувань серверу.

Зазначене вікно можна викликати з головного вікна. Логіка форми складається зі списку серверів, з якими взаємодіє додаток та можливості проводити такі маніпулятивні дії, як додавання серверу, редагування даних серверу, оновлення інформації та видалення серверу.

Логіку форми налаштування серверів показано на рис. 2.59.

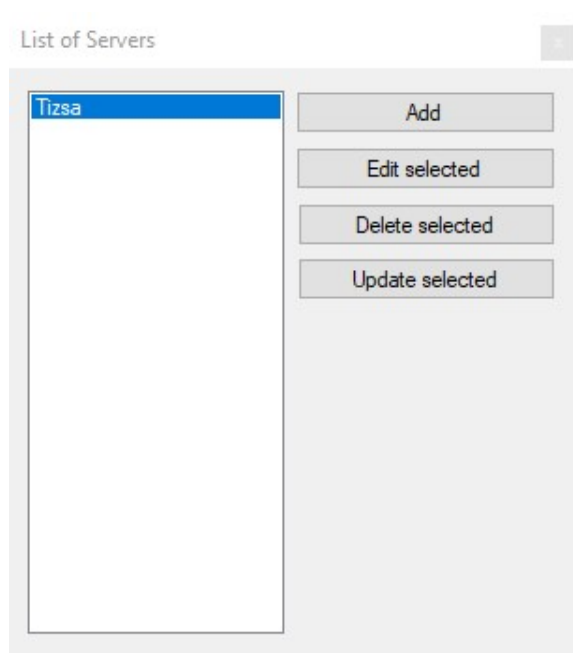


Рис. 2.59. Верстка форми налаштування серверів

Кнопки додавання та редагування серверів викликають форму, що описує характеристики серверу та дає можливість їх додавати або редагувати. Особливість зазначеної форми полягає в тому, що вона використовується в двох режимах – додавання та редагування. Серед характеристик, що необхідні серверу для роботи є IP адреса серверу. Ім'я серверу завантажується з самого API серверу.

Логіка зазначеної форми показана на рис. 2.60.



Рис. 2.60. Верстка форми додавання/редагування серверу

Форму налаштування серверів описує клас FromDBToUI, побудовано на шаблоні Code Behind – стандартному шаблоні, закладеному в технології WinForms.

Форму додавання/редагування серверу спроектовано з використанням технології WPF та шаблону MVVM. Класи AddEditServerWindow та AddEditServerViewModel описують зазначену логіку.

Діаграму класів показано на рис. 2.61.

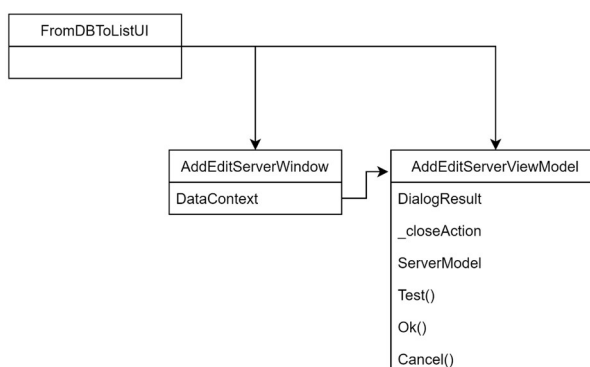


Рис. 2.61. Діаграма класів шару налаштування серверів

Клас AddEditServerWindow має DataContext поле у якості функціонального, що зв'язує логіку форми та ViewModel.

Клас AddEditServerViewModel має наступні поля та методи:

- DialogResult – результат діалогу;
- \_closeAction – подія закриття форми;
- ServerModel – модель серверу, що використовується для взаємодії;
- Test() – метод функціоналу тестування з'єднання;
- Ok() – метод збереження результату та закриття форми;

- Cancel() – метод закриття форми без збереження результату.

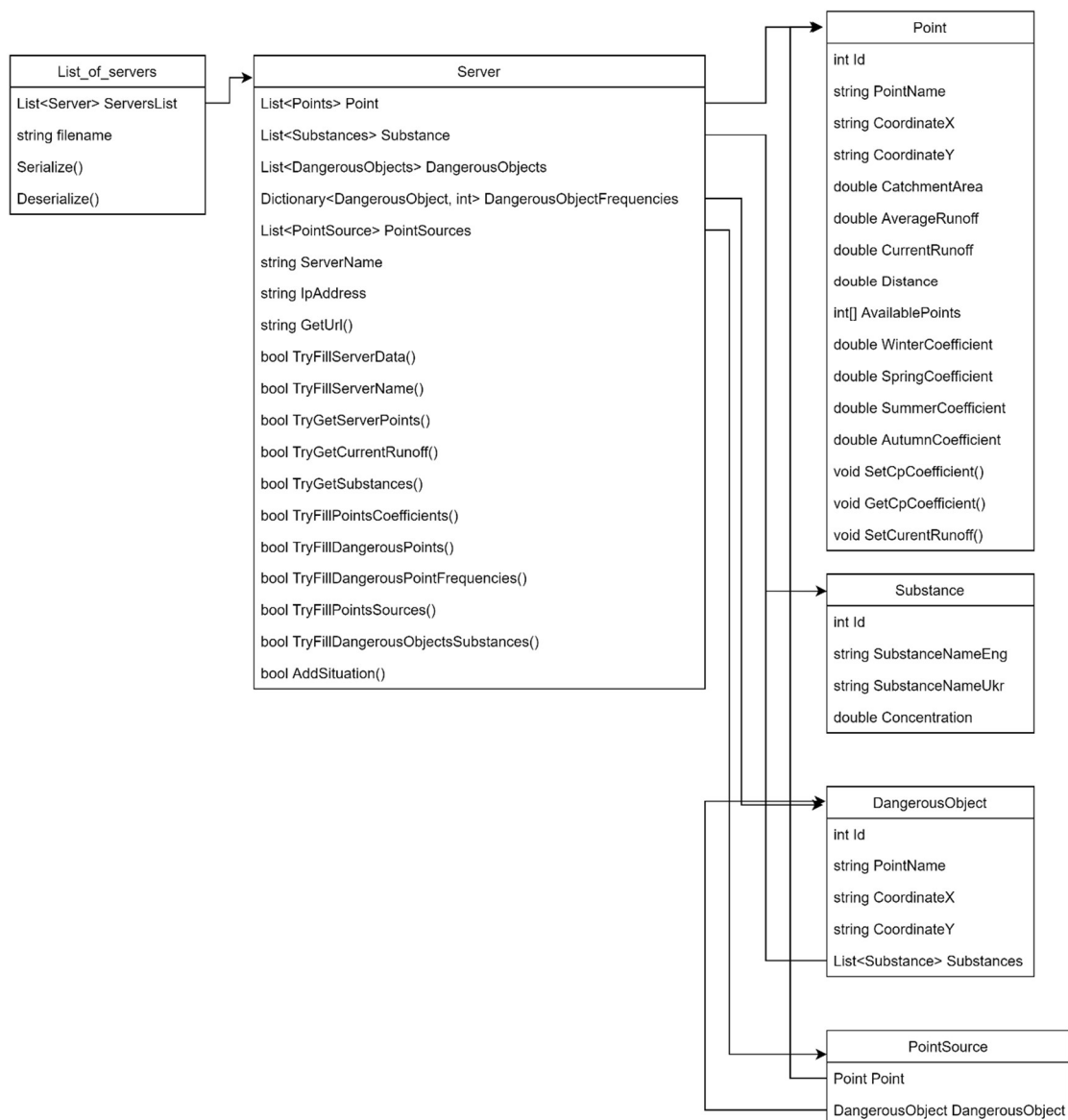


Рис. 2.62. Діаграма класів механізму, що описує логіки завантаження даних з серверу

При зберіганні даних відбувається їх оновлення у списку серверів та серіалізація цього списку у файл. При запуску програми відпрацьовує функціонал десеріалізації, що дає можливість працювати з даними автоматично. Таким чином, функціонал налаштування серверу взаємодіє з шаром, що описує логіки завантаження даних з серверу. Діаграму класів даного шару показано на рис. 2.62.

Клас List\_of\_servers описує логіку зберігання та доступ до списку серверів, функціонал серіалізації та десеріалізації. Має наступні поля та методи:

- ServersList – посилання на список серверів;

- Serialize() – серіалізація списку серверів у файл;
- Deserialize() – десеріалізація списку серверів з файлу.

Клас Server описує поля та методи, необхідні для опису даних серверу:

- Point – список точок;
  - Substance – список речовин;
  - DangerousObjects – список небезпечних об'єктів;
  - DangerousObjectFrequencies – список частот, що забруднюють навколишнє середовище (кількість разів, коли даний об'єкт забруднював середовище);
  - PointSources – список фактів, що зв'язують точку, речовину та небезпечний об'єкт;
  - ServerName – ім'я серверу;
  - IPAddress – IP адреса серверу;
  - GetUrl() – метод отримання URL серверу;
  - TryFillServerData, TryFillServerName, TryFillServerPoints, TryGetCurrentRunoff, TryGetSubstances, TryFillPointsCoefficients, TryFillDangerousObjects, TryFillDangerousObjectFrequencies, TryFillPointSources, TryFillDangurousObjects Substances – методи, що перезаповнюють відповідні характеристики серверу;
  - AddSituation() – метод, що додає нову ситуацію.
- Клас Point описує характеристики точки. Серед них:
- Id, PointName, CoordinateX, CoordinateY, CatchmentArea, AverageRunoff, CurrentRunoff, Distance – характеристики з бази даних;
  - AvailablePoints – опис маршруту точок;
  - WinterCoefficient, SpringCoefficient, SummerCoefficient, AutumnCoefficient – виведені коефіцієнти похибки для точки;
  - SetCpCoefficient() – оновлення значення для вибраного коефіцієнту похибки;
  - GetCpCoefficient() – отримання значення вибраного коефіцієнту похибки;
  - SetCurrentRunoff() – оновлення значення характеристики поточних витрат води.



### 2.34. Математичне ядро застосунку

Математичне ядро призначене для зберігання алгоритмів математичних моделей, необхідних для роботи застосунку. Основним підходом до функціонування ядра є робота з примітивними типами та їх значеннями. Такий підхід забезпечить максимальну гнучкість роботи модулю. Ядро складається з чотирьох алгоритмів:

- алгоритм виконання обчислень. Створений для обчислювання результатів за основною методикою моделювання наслідків забруднення. Описаний у класі `HydrologicalMethod`, складається з таких методів, як `A()`, `Arel()`, `Rx()`, `Vp()`, `Vm()`, `Tr()`, `Cup()`, `Cr()`, що описують математичні формули та є основою методики. На вхід приймаються відповідні необхідні змінні, вихідними значеннями є результати;

- алгоритм інтерполяції. Розроблено для збільшення кількості результатів за рахунок методу інтерполяції для згладження прогнозованих результатів. У якості вхідних даних використовуються дві сусідні точки. Клас `InterpolationMethod` має на меті реалізацію зазначеного алгоритму. Метод `GetIntermediatePoints()` описує математичний алгоритм з використанням рекурсії та збереження результату у список `_measurementPointResults`. Метод `GetResultedList()` – повертає результат обчислення;

- алгоритм моделювання динаміки забруднення. Розроблено для розуміння динаміки спадання концентрації. Досягається відповідним алгоритмом. Клас `PointDilutionMethod` виконує обчислення за даним алгоритмом. Конструктор приймає такі необхідні параметри, як точки, потрібні для проведення прогнозування, список середніх витрат за точками, викинуту масу забруднювача та час настання ситуації. Метод `FillPointDilution()` виконує всі необхідні обчислення для характеристики концентрації та коригованої концентрації, взявши при цьому коефіцієнт, що відповідає сезону року настання обчислюваної події. Метод `FillPointsDilution()` реалізовує логіку інтерполяції для вхідних точок. Результатом є список значень, що зберігається для кожної точки;

- алгоритм коригування характеристики концентрації. Створений для коригування концентрації визначеним коефіцієнтом. Реалізує метод коригування

результатів прогнозування. Описується класом `CpRegressionMethod`, основою якого є метод `PerformanceRegressionMethodMeasurement()`. Цей метод проводить операцію ітерації за зазначеними даними та за визначеним коефіцієнтом коригує необхідну характеристику. Отже входними даними є список точок та дата настання надзвичайної ситуації. Вихідними даними є оновлення значень у відповідній характеристиці;

- алгоритм визначення потенційних забруднювачів вибраної точки забруднення. Реалізується класом `PointSourceFilter`. В основі лежить метод `Filter()`. Приймає список всіх можливих фактів, що були завантажені з серверу, ідентифікатор точки та речовини. За зазначеними даними проводить фільтрацію та сортує за частотою об'єктом у порядку спадання.

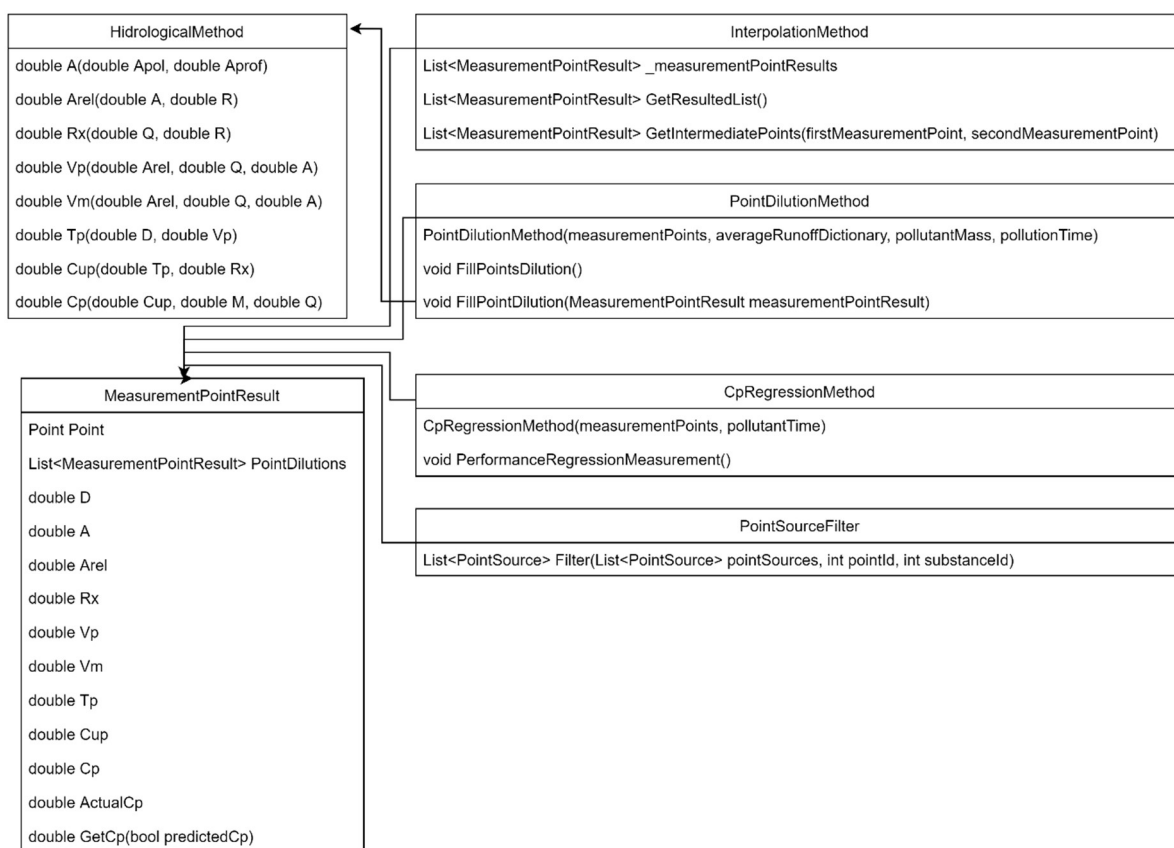


Рис. 2.63. Діаграма класів математичного ядра

В основі зберігання результатів та необхідних характеристик лежить модель `MeasurementPointResult`, що має наступні поля:

- `Point` – посилання на батьківську точку;

- PointDilutions – список результатів, отриманих при використанні алгоритму моделювання динаміки забруднення;

- D, A, Arel, Rx, Vp, Vm, Tr, Cup, Cp – проміжні та кінцеві результати обчислень;

- ActualCp – кориговане значення поля Cp.

Метод GetCp повертає значення характеристики піку концентрації за необхідністю, значення якої вказане у вхідних змінних.

Діаграму класів описаних частин функціоналу показано на рис. 2.63. Як видно з діаграми, компонент має достатньо автономні модулі. Такий підхід забезпечує такі нефункціональні вимоги до архітектури, як масштабованість, можливість повторного використання та автономність.

## **Висновки до розділу 2**

В розділі описані основні розробки та удосконалення, що розглядаються в роботі. Зокрема:

1. Описано закон Фіка, як теоретичну основу вибраної методики прогнозування наслідків забруднення річок.

2. Описано серію емпіричних гідрологічних рівнянь, як методику прогнозування наслідків забруднення річок. Методика є основою для розробленої інформаційної системи

3. Описані такі базові удосконалення, як методи багатоточковості та мультимаршрутизації, метод оцінки динаміки забруднення в конкретній точці. Такі розширення дозволяють масштабувати методику та дають можливість розробки більш широкого набору функціоналу інформаційної системи.

4. Описано розроблений метод коригування результатів забруднення, шляхом використання задачі регресії. Метод підвищує точність результатів за рахунок існуючих статистичних даних, що використовуються в нейронній мережі для навчання. Є однією з основних складових до інтелектуальної обробки результатів.

5. Описано розроблений метод інтерполяції для зглаження прогнозованих результатів. Алгоритм дозволяє отримати результати в місцях, що знаходяться між точками моніторингу та дає більш широкий масив результатів. Це дає можливість кращого розуміння рівнів наслідків забруднення русла річки. Є однією з основних складових до інтелектуальної обробки результатів.

6. Описано розроблений метод визначення джерела забруднення, використовуючи базу даних фактів та їх класифікацію. Метод дозволяє визначити можливий список забруднювачів. Є однією з основних складових інтелектуальної обробки результатів.

7. Описано інформаційну систему RESit, як основну інформаційну технологію, що вирішує проблему автоматизації обраної методики. Описано призначення та архітектуру основних складових частини зазначеної системи.

Даний розділ описує основні розробки та удосконалення, розроблені в рамках дисертаційної роботи.

## **РОЗДІЛ 3**

### **ВАЛІДАЦІЯ МЕТОДІВ ТА АЛГОРИТМІВ**

#### **3.1. Валідація методів та алгоритмів**

Розроблені математичні методи та алгоритми вимагають валідації для підтвердження правильності роботи та відповідності результатів вирішуваним проблемам. Цей підрозділ містить опис критеріїв ефективності та обмежень системи, а також опис валідації таких алгоритмів та методів, як:

- алгоритми багатоточковості та мультимаршрутизації;
- метод коригування результатів прогнозування, що використовує задачу регресії;
- метод інтерполяції для згладження прогнозованих результатів;
- метод визначення джерела забруднення, що використовує базу даних фактів.

#### **3.2. Критерії ефективності системи та обмеження**

При вирішенні питання оцінки якості води в умовах надзвичайних ситуацій найважливішим критерієм ефективності функціонування системи є швидкість проведення оцінювання. Включає 3 складові:

- час, затрачений на збір даних, що описують характеристики русла;
- час, затрачений на збір даних, що описують характеристики самої ситуації (такі, як температура води, рівень води, температура навколишнього середовища та ін.);
- час, затрачений на проведення обчислень за визначеною методикою.

При створенні спеціалізованої бази даних, що описує характеристики русла, зазначену складову можна нівелювати. Крім того, при виборі методики, що оцінює якість води, виходячи лише з загальних та статистичних характеристик локації, другу складову можна мінімізувати від умовних 5 характеристик, що мають бути введені (температура води, висота русла, нахил русла, фізичні властивості русла) до одної. Таким чином, час, затрачений на збір даних, що описують характеристики

самої ситуації можна зменшити на 80%. За рахунок автоматизації обчислень, що використовують дані зі вже створеної бази даних, час можна нівелювати.

Порівнюючи вищеописані критерії з автоматизованою системою, що має існуючу базу даних та вимагає введення даних, що описують характеристики самої ситуації, критерій похибки можна покращити умовно на 80%. Точність обчислень може бути зменшена за рахунок деяких ситуативних змін та ігнорування деяких факторів.

Від так, існує необхідність в розробці нейронної мережі, що коректує результати обчислень та підвищувала очікувану ефективність методу в залежності від конкретної локації та ситуації шляхом введення таких поліпшених критеріїв:

- збільшення швидкості проведення оцінювання (зменшення кількості годин та ресурсів);
- зменшення вірогідності набутої похибки за рахунок виведених коефіцієнтів.

З огляду на те, що результати є точковими, кількість значень буде рівна кількості точок (10 точок – 10 значень). Використовуючи методи інтерполяції можна збільшити кількість результатів, чим розширити розрахунок можливого забруднення. При умові існування 10 точок при відстані в 10 км та точності в умовах 200 м можна отримати 50 значень. Поліпшення має таке формулювання, як збільшення кількості виведених результатів в 5 разів (з 10 до 50 значень).

Наступним критерієм, що може бути проаналізований – визначення джерела забруднення відносно визначеної ситуації на річці. При існуванні точки виливу можна визначити безпосередньо суб'єкт, який здійснив вилив речовини за рахунок методу визначення джерела, базуючись на базі даних фактів.

Можна вивести такі поліпшені критерії, як створення можливості визначення суб'єкту забруднення.

Таким чином, виведено чотири критерія ефективності, що поліпшують отримані результати роботи системи в умовах надзвичайної ситуації.

### 3.3. Обмеження системи

Вибрані та розроблені методи та інформаційна система мають ряд обмежень, що пов'язані з їх можливостями. Цей підрозділ присвячений опису цих обмежень для частин досліджуваної інформаційної технології.

Обмеження для основного методу проведення оцінки забруднення з удосконаленнями наступні:

- у якості вхідних даних метод має прийняти точку скиду та останню точку вимірювання;
- метод приймає тільки загальні характеристики русла та не має адаптованості до умов локації, або надзвичайної ситуації, через що точність обчислення може бути обмеженою;
- метод є одновимірним, тобто прогнозує тільки сталий стан;
- метод працює тільки з однією точкою викиду (тобто, джерело викиду може бути тільки одне).

Обмеженнями для методу коригування результатів прогнозування:

- мережа не захищена від неправильного обчислення у випадку аномалій, то ж вхідні дані до навчання нейронної мережі не мають містити аномальних значень та бути інтерпретованими;
- для більш правдоподібних обчислених значень кількість вхідних даних має містити від 10 записів на точку моніторингу, що відображають закономірність похибки локації (значення для навчання підібране емпіричним шляхом);
- для виведення нового коефіцієнту, мережа має перенавчитись, алгоритми донавчання закладені не були;
- мережа повинна навчатись на зліпку бази даних без одночасного внесення змін до системи, щоб уникнути конфлікту даних, таких як додавання нової точки моніторингу, видалення точки моніторингу і т.д.

Список обмежень для методу інтерполяції для згладження прогнозованих результатів наступний:

- метод здійснює прогнозування між двома точками моніторингу;

- метод не враховує експоненційність результатів, через що точність є обмеженою;

- результати показують лише рівномірне спадання і не враховують особливостей локації.

Обмеженнями для методу визначення джерела забруднення є:

- джерелом може бути лише об'єкт, що вказано в базі даних, тобто підприємство, що є потенційно небезпечним;

- при прив'язці нового підприємства до бази даних потрібно проаналізувати причетність його до деякої точки моніторингу експертним шляхом;

- час роботи алгоритму збільшується з кількістю підприємств, внесених до бази даних, тому рекомендується наповнювати базу даних, впевнившись в причетності експертним шляхом.

Інформаційна система має наступні обмеження:

- питання безпеки системи не розглядаються;
- деталізоване під'єднання датчиків вимірювання витрат води не розглядаються, система описує лише принцип взаємодії та точку доступу;

- обсяг інформації, що зберігається в базі даних не має відповідати обсягу великих даних;

- механізми бекапів, оновлення версії системи та інші, пов'язані з підтримкою системи не розглядаються;

- логіка побудови інфраструктури для роботи інформаційної системи не розглядається;

- адаптація системи під різні бізнес-процеси не розглядається.

Інформаційна технологія має наступні обмеження:

- метод визначення рівня небезпеки забруднення для навколишнього середовища не розглядається;

- алгоритми дій для ліквідації наслідків забруднення не розглядаються;

- методи підрахування збитків внаслідок надзвичайної ситуації не розглядаються;



- методи визначення джерела забруднення, базуючись на визначенні рівня забруднення в деякій локації русла не розглядаються.

Отже, описані обмеження мають враховуватись при валідації існуючих методів та алгоритмів та використанні інформаційної технології.

### **3.4. Валідація алгоритмів багатоточковості та мультимаршрутизації**

Алгоритми багатоточковості та мультимаршрутизації розроблено для реалізації можливості проводити обчислення між декількома точками та маршрутами. Ці алгоритми потребують аргументації та валідації їх роботоспроможності.

Розглянемо наступні приклади. В прикладі № 1, точка виливу має наступні характеристики:

- площа водозбору – 350000000 м<sup>2</sup>.

Точка вимірювання має такі характеристики:

- площа водозбору – 430000000 м<sup>2</sup>;
- середні витрати води – 4,5 м<sup>3</sup>/сек;
- дистанція від точки виливу до точки вимірювання – 25000 м;
- поточні витрати води – 15 м<sup>3</sup>/сек.

Викинуто – 6000000000 мг забруднюючої речовини.

Результати:

- час, через який концентрація речовини в точці вимірювання досягне свого піку – 9 годин;
- пікова концентрація – 75 мг/л.

Задача № 2: необхідно провести обчислення в умовах трьох точок, де перша – точка виливу, а інші дві – вимірювання.

Точка викиду має наступні характеристики:

- площа водозбору – 189000000 м<sup>2</sup>;
- викинуто - 1000000000 мг забруднюючої речовини.

Перша точка вимірювання має такі характеристики:

- площа водозбору – 371600000 м<sup>2</sup>;

- середні витрати води –  $8,82 \text{ м}^3/\text{сек}$ ;
- дистанція від точки виливу –  $6950,318196 \text{ м}$ ;
- поточні витрати води –  $10 \text{ м}^3/\text{сек}$ .

Друга точка вимірювання має наступні характеристики:

- площа водозбору –  $568500000 \text{ м}^2$ ;
- середні витрати води –  $26,9 \text{ м}^3/\text{сек}$ ;
- дистанція від точки виливу –  $226580,501 \text{ м}$ ;
- поточні витрати води –  $25 \text{ м}^3/\text{сек}$ .

Результати для першої точки:

- час, через який концентрація забруднюючої речовини досягне свого піку –  $4,277890553$  годин;
- пікова концентрація –  $28,70658119 \text{ мг/л}$ .

Використовуючи, характеристики другої точки можна провести операції з вхідними характеристиками, таким чином, симулювати для методики другу точку, як точку вимірювання. Така точка буде мати наступні вхідні характеристики:

- площа водозбору (оскільки охоплює все русло вище) –  $568500000 \text{ м}^2$ ;
- середні витрати води –  $26,9 \text{ м}^3/\text{сек}$ ;
- дистанція від точки виливу (сума першої точки вимірювання та другої) –  $6950,318196 + 219630,182804 = 226580,501 \text{ м}$ ;
- поточні витрати води –  $25 \text{ м}^3/\text{сек}$ .

Результати для другої точки:

- час, через який концентрація досягне свого піку –  $132,9164238$  годин;
- пікова концентрація –  $0,979345209 \text{ мг/л}$ .

Інші точки за маршрутом також можна симулювати описаним чином. В результаті формується масив результатів, що дозволяють побудувати графік спадання концентрації в часі та просторі.

Основним принципом роботи алгоритму мультимаршрутизації є прокладання маршруту з точок моніторингу. У випадку розгалуження або з'єднання гілок русла, єдиною характеристикою, що потребує обчислення є сума

дистанцій від попередньої точки. Тобто, перша точка після розгалуження має зберігати значення дистанції від останньої точки до розгалуження.

У випадку з ситуацією з'єднання гілок перша точка після з'єднання має мати дистанцію від останньої точки до з'єднання відповідної точки з відповідної гілки. Вибрана методика не описує жодних обмежень на дистанцію між точками.

Таким чином, будь-яка точка після точки забруднення при використанні обчислень виступає, як точка вимірювання. З точки зору роботи основної методики жодних змін не внесено, а отже правильність роботи зазначених алгоритмів залежить від ступеню коректності вхідних характеристик, що вводяться оператором та зберігаються в базі даних.

### **3.5. Валідація методу коригування результатів прогнозування, що використовує задачу регресії**

Метод коригування результатів розроблено для коректування отриманих результатів на основі статистичних даних. Метою такого методу є виявлення закономірності похибки та виведення коефіцієнту.

Вхідною характеристикою є відношення реальних величин до прогнозованих. В результаті виводиться деякий масив коефіцієнтів для кожної точки, що і є значенням для навчання.

Таблиця 3.1.

Вхідні значення для виведення коефіцієнту для першої точки

<b>Очікуване значення</b>	<b>Реальне значення</b>
28,71	22,968
31,98	25,584
360	288,024
14,35	11,48
0,58	0,464

Валідацією даних є підтвердження правильності роботи методу. Тому, доцільним є створення деякого масиву даних, що має еталонне значення

коефіцієнту з метою навчання. Коректність роботи методу має обернено пропорційну залежність до похибки між еталонним та обчисленим значенням. Таким чином, за еталонне значення можна взяти коефіцієнт 0,8 та створити масиви вхідних даних, що мають значення, вказані на таблицях 3.1, 3.2, 3.3, 3.4, 3.5.

Таблиця 3.2.

Вхідні значення для виведення коефіцієнту для другої точки

Очікуване значення	Реальне значення
0,8	0,64
0,98	0,784
9,66	7,728
0,41	0,328
0,016	0,0128

Таблиця 3.3.

Вхідні значення для виведення коефіцієнту для третьої точки

Очікуване значення	Реальне значення
0,71	0,568
0,91	0,728
8,57	6,856
0,38	0,304
0,015	0,012

Таблиця 3.4.

Вхідні значення для виведення коефіцієнту для четвертої точки

Очікуване значення	Реальне значення
0,255	0,204
0,33	0,264
3,08	2,464
0,13	0,104
0,05	0,04

Таблиця 3.5.

Вхідні значення для виведення коефіцієнту для п'ятої точки

Очікуване значення	Реальне значення
0,174	0,1392
0,21	0,168
2,09	1,672
0,09	0,072
0,03	0,024

Результатом є масив коефіцієнтів, що показано на таблиці 3.6.

Таблиця 3.6.

Вихідні значення

№ точки	Реальне значення
1	0,799974132805534
2	0,7951551855680876
3	0,7965094022097843
4	0,7933086155135414
5	0,7627572392513363

Усі інші точки мають коефіцієнт 1, оскільки не мають масиву даних, на основі яких потрібно проводити навчання.

В результаті аналізу результатів, можна зазначити, що вихідні значення є близькими до еталонного, а отже метод працює правильно.

У якості даних для навчання, варто використовувати значення, що відображають закономірність. В іншому випадку, точність результатів та ефективність методу може бути зменшена.

Ефективність методу може бути оцінена та покращена тільки при використанні реальних даних.

### 3.6. Валідація методу інтерполяції для згладження прогнозованих результатів

Метод згладження прогнозованих результатів розроблено для отримання даних про пікову концентрацію між точками. Оскільки метод лише прогнозує середнє значення між двома сусідніми точками, доцільно підтвердити його правильність графічним методом.

Дано дві точки, характеристики яких описані в таблиці 3.7.

Таблиця 3.7

Вхідні значення

№ точки	Дистанція від початкової точки	Час, через який концентрація набуде піку	Пікова концентрація
1	6950,318	4,28	22,97
2	226580,5	126,07	0,64

Графік для даних точок показано на рис. 3.1, рис. 3.2.

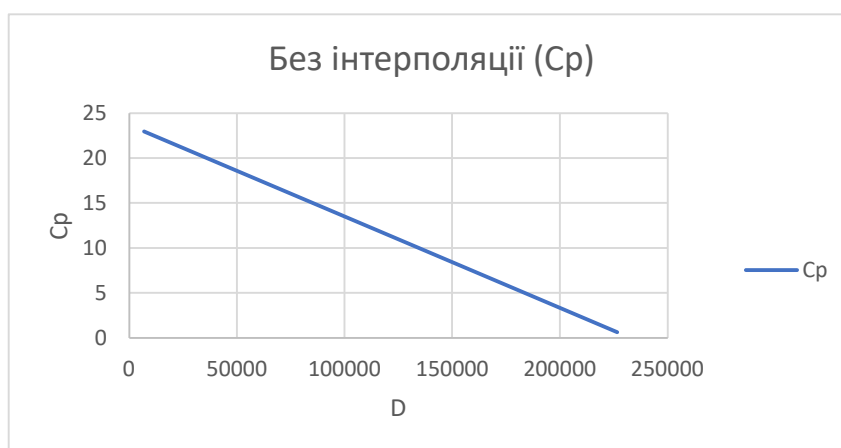


Рис. 3.1. Графік залежності між піковою концентрацією та дистанцією

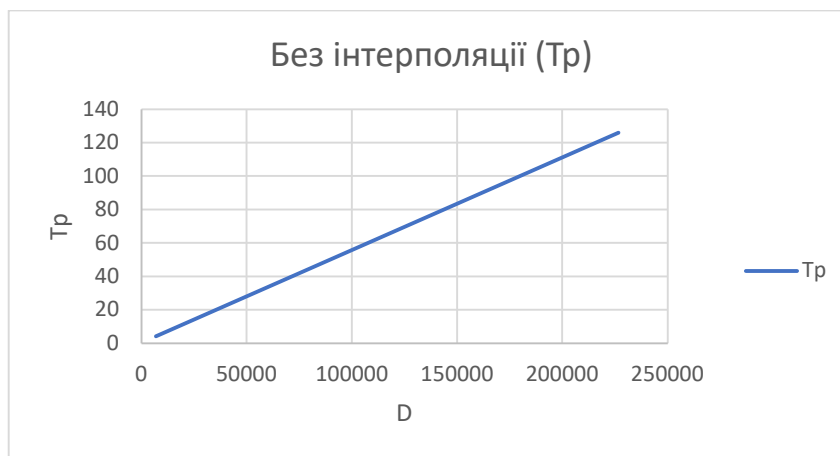


Рис. 3.2. Графік залежності між часом, через який концентрація набуде піку та дистанцією

При використанні методу інтерполяції можна отримати масив результатів, показаний в таблиці 3.8.

Таблиця 3.8.

Вихідні значення

№ точки	Дистанція від початкової точки	Час, через який концентрація набуде піку	Пікова концентрація
1	6950,318196	4,28	22,97
	61857,8639	34,7275	17,3875
	116765,4096	65,175	11,805
	171672,9553	95,6225	6,2225
2	226580,501	126,07	0,64

Графіки, що побудовані на основі таблиці 3.8 показано на рис. 3.3 та рис. 3.4.

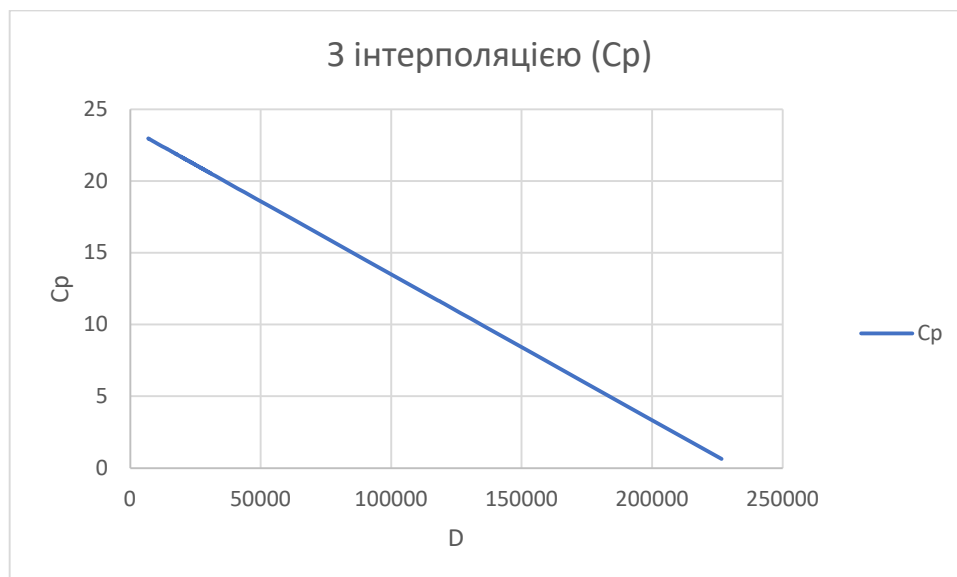


Рис. 3.3. Графік залежності між піковою концентрацією та дистанцією

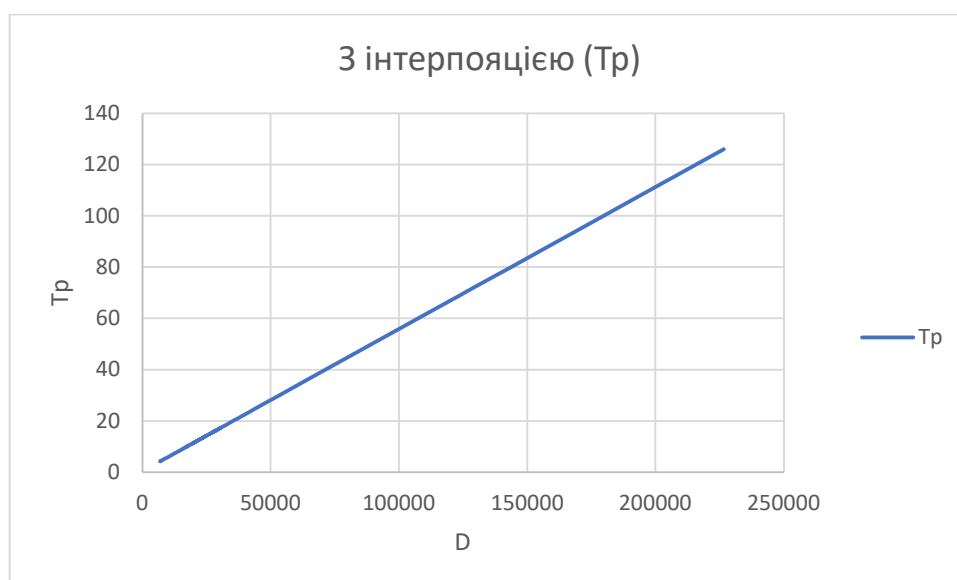


Рис. 3.4. Графік залежності між часом, через який концентрація набуде піку та дистанцією

Вихідні дані відображають використання методу інтерполяції з першим рівнем глибини рекурсії. Даний рівень має можливість збільшення до необхідного рівня деталізації.

Проаналізувавши два графіки, можна зазначити, що вони є рівними, а отже метод інтерполяції лише збільшує масив результатів не створюючи при цьому похибок.

Таким чином, правильність методу можна вважати доведеною.



### 3.7. Валідація методу визначення джерела забруднення, що використовує базу даних фактів

Метою даного методу є розширення картини прогнозування, тобто аналізу можливих забруднювачів точки забруднення. Розробка є алгоритмом, що фільтрує дані, спираючись на тестову базу даних фактів.

Таблиця 3.9.

Тестові ситуації до даного алгоритму

№	Вхідні дані	Опис	Очікуваний результат
1	Точка моніторингу.	Знайти можливі забруднювачі до точки моніторингу без додавання речовини та небезпечного об'єкту	Жодного результату не повернуто
2	Точка моніторингу та речовина, що мають зв'язку.	Знайти можливі забруднювачі до точки моніторингу без додавання небезпечного об'єкту	Жодного результату не повернуто
3	Точка моніторингу, речовина та небезпечний об'єкт що мають зв'язку.	Знайти можливі забруднювачі до точки моніторингу	Повернуто об'єкт
4	Точка моніторингу, речовина та небезпечний об'єкт, що не мають зв'язків.	Знайти можливі забруднювачі до точки моніторингу без зв'язки на рівні бази даних зв'язок	Жодного результату не повернуто
5	Точка моніторингу, речовина та небезпечний об'єкт що мають зв'язки. Інший ідентифікатор речовини.	Знайти можливі забруднювачі до точки моніторингу та іншою речовиною	Жодного результату не повернуто

Валідація методу є тестуванням алгоритму та проведенням симуляцій на тестовій базі даних. Якщо визначений алгоритмом список можливих об'єктів співпадає з очікуваним, то метод можна вважати валідованим.

Тестові ситуації до даного алгоритму показано на табл. 3.9.

Описаний список ситуацій покриває більшість значень, що можуть бути прийняті методом.

Для проведення симуляцій необхідно створити деяку тестову базу даних умовних об'єктів, точок моніторингу та речовин. Такі списки мають наступний вигляд:

- Точки моніторингу: Точка А, Точка В, Точка С, Точка D.
- Потенційно небезпечні об'єкти: Об'єкт 1, Об'єкт 2, Об'єкт 3, Об'єкт 4, Об'єкт 5, Об'єкт 6, Об'єкт 7, Об'єкт 8.
- Речовини: Речовина 1, Речовина 2, Речовина 3, Речовина 4, Речовина 5, Речовина 6, Речовина 7, Речовина 8.

Таблиця 3.10.

Тестова база даних

<b>Точка моніторингу</b>	<b>Потенційно небезпечний об'єкт</b>	<b>Речовина</b>
Точка А	Об'єкт 1	Речовина 1
		Речовина 2
	Об'єкт 2	Речовина 2
Точка В	Об'єкт 3	Речовина 3
		Речовина 4
	Об'єкт 4	Речовина 4
		Речовина 5
	Об'єкт 5	Речовина 5
Точка С	Об'єкт 6	Речовина 6
	Об'єкт 7	Речовина 7
Точка D	Об'єкт 8	Речовина 8

Зв'язка списків має вигляд, показаний у Таблиця 3.10

Симуляційні ситуації мають вигляд, показаний у табл. 3.11.

Таблиця 3.11.

Симуляційні ситуації

Дано		Очікуваний результат
Вхідна точка	Вхідна речовина	Можливі об'єкти
Точка А	-	-
Точка В	Речовина 2	-
Точка В	Речовина 3	Об'єкт 3
Точка С	Речовина 1	-
Точка D	Речовина 7	-
Точка А	Речовина 2	Об'єкт 1, Об'єкт 2

Результати, що виникли в ході використання алгоритму, блок схема якого показана на Рис. 2.11 потрібно порівняти з очікуваним результатом. У випадку співпадіння метод можна вважати валідованим.

Алгоритм сортування не потребує валідації, т.я. використовує уже існуючі валідовані алгоритми сортування.

### 3.8. Тестування серверу

Метою серверу є забезпечення системи даними, тому є однією з найважливіших частин для тестування. Від так, правильність роботи є одним з ключових завдань. Оскільки, сервіс має складну структуру, тестування вимагає декомпозиції на такі частини:

- частина забезпечення даними (тобто REST API сервіс);
- адміністративний додаток.

REST API сервіс може бути протестований шляхом взаємодії з кінцевими точками, забезпечуючи базу даних необхідними вхідними значеннями. Список методів та позитивні тестові ситуації показано в Додатку Б.

Описаний список ситуацій покриває лише позитивні тестові ситуації. Від так, тест критичного функціоналу може бути проведений автоматичним чином. Також, до списку тестів необхідну додати негативні тести. Таким чином, сервіс перевіриться на стійкість до непередбачуваних ситуацій. Також, комплексні тести, що перевіряють різні складні ситуації можуть виявити вразливі та приховані місця.

Частина адміністративного додатку є повним зовнішнім рішенням, а отже вимагає поверхневий тест функціоналу та не потребує поглиблених перевірок. Тим не менш, доцільною буде перевірка критичного функціоналу, використовуючи тести інтерфейсу користувача. Список ситуацій показано в табл. 3.12.

Таблиця 3.12.

## Тестові ситуації адміністративної панелі

№	Опис	Вхідні дані	Результат
1	Перевірка сервісу на доступність	-	Сервіс доступний
2	Логін користувача	1. Пароль у базі даних	Користувач успішно авторизований
3	Перевірка списку сутностей у виведеній таблиці	1. Сформована база даних	Список сутностей в інтерфейсі рівний сутностям у базі даних
4	Перевірка можливості додавання даних у таблицю	1. Дані для додавання	Дані успішно додані
5	Перевірка можливості видалення даних з таблиці	1. Дані в базі даних	Дані видалено
6	Перевірка можливості редагування даних з таблиці	1. Дані в базі даних	Редаговані дані збережено в базі даних

Оскільки, основою частини є робота з базою даних, описані в Додаток Б. Позитивні тестові ситуації для REST API сервісу

Таблиця тестові ситуації частково покривають перевірку описаного сервісу.

Описаний список ситуацій у Таблиця 3.12 може бути проведений, як вручну, так і описаний при використанні рішення Selenium.

### **3.9. Тестування настільного додатку**

Настільний додаток створений для взаємодії кінцевого користувача з системою, тому правильне тестування цього застосунку має ключову роль. Шаблони MVVP та MVP створені з максимальним ухилом в автономність та самостійність кожного шару. Тому, такі два основних шляхи, як тестування інтерфейсу та модульне тестування ViewModel/Presenter є єдиними можливими шляхами для створення автоматичних тестів. Оскільки додаток є достатньо складним та має велику кількість функціоналу, доцільним буде розділити весь додаток для тестування на наступні частини:

- головне вікно;
- майстер проведення ручних обчислень;
- майстер проведення автоматичних обчислень;
- дашборд;
- налаштування серверів;
- математичне ядро застосунку.

Даний підрозділ описує методику тестування перерахованих складових додатку, що включає в себе наведення методів проведення тестів та план позитивних ситуацій.

### **3.10. Головне вікно**

Цей компонент розроблено на основі шаблону MVP. Від так, доцільно використовувати метод тестування інтерфейсу користувача та модульного тестування частин класу Presenter'у. Позитивні тестові ситуації описано в Додатку В.

Описані ситуації, що не потребують виклику суміжного функціоналу можуть бути протестовані методом тестування інтерфейсу користувача. Інші функціональні методи можуть бути описані шляхом модульного тестування.

### 3.11. Майстер проведення ручних обчислень

Зазначений майстер має достатньо складну структуру. Тому, тестування критичного функціоналу проводиться шляхом тестування інтерфейсу користувача. Табл. 3.13 описує список позитивних тестових ситуацій.

Таблиця 3.13.

Тестові ситуації майстру проведення ручних обчислень

№	Опис	Вхідні дані	Результат
1	Запуск майстру	-	Початкове вікно майстру відобразилось
2	Перевірка переходу на наступний крок	Введення необхідних даних	Відобразилось вікно введення вхідних характеристик
3	Перевірка вікна проміжного прорахунку даних	Перший крок виконано. Дані введені у друге вікно	Вікно часткових обчислень для точки відображається
4	Збереження проміжних результатів у XLS файл	Перший крок виконано. Дані введені у друге вікно. Викликане вікно прорахунку даних	Файл у форматі XLS збережено
5	Введення даних за всіма точками	Перший крок виконано. Дані введені у друге вікно	Обчислення створено
6	Алгоритм рахує правильно	Масив вхідних даних. Масив вихідних еталонних результатів	Вихідний результат співпадає з еталонним

Описані тестові ситуації можна провести шляхом тестування інтерфейсу користувача, за виключенням шостої ситуації. Ситуацію потрібно проводити шляхом модульного тестування.

### 3.12. Майстер проведення автоматичних обчислень

Таблиця 3.14.

Тестові ситуації майстру проведення автоматичних обчислень

№	Опис	Вхідні дані	Результат
1	Перевірка запуску майстра	-	Вікно вибору серверу відображається
2	Перевірка запуску вікна вибору точок	Вибір серверу	Вікно з завантаженою картою, на якій стоять точки відображається
3	Вибір початкової точки забруднення	Вибір серверу. Запуск вікна вибору точок.	Початкова точка відображається вибраною
4	Перевірка, чи маршрут можливих забруднень відображається	Вибір серверу. Запуск вікна вибору точок. Вибір першої точки.	На карті показані тільки ті точки, що можуть бути вибрані для обчислення забруднення
5	Вибір кінцевої точки	Вибір серверу. Запуск вікна вибору точок. Вибір першої точки.	Кінцева точка відображається вибраною
6	Запуск введення вхідних даних	Вибір серверу. Вибір точок.	Вікно введення вхідних даних відображається
7	Перевірка врахування всіх даних	Проведення обчислення. Масив вхідних даних для обчислення. Масив еталонних результатів.	Обчислені дані співпадають з еталонними

Зазначений майстер має достатньо складну структуру. Тому, тестування критичного функціоналу проводиться шляхом тестування інтерфейсу користувача. Таблиця 3.14 описує список позитивних тестових ситуацій.

Дані ситуації є позитивними для тестування критичного функціоналу. Всі описані ситуації мають проводитись шляхом тестування користувацького інтерфейсу. Тестування алгоритму має проводитись в рамках тестування математичного ядра застосунку.

### **3.13. Дашборд**

Цей функціонал є елементом, що націлений на взаємодію з користувачем. Відтак, тестування основного функціоналу варто проводити методом тестування інтерфейсу користувача. Додаток Г описує список позитивних тестових ситуацій.

Наведений список ситуацій відображає ситуації, що перевіряють критичний функціонал.

### **3.14. Налаштування серверів**

Тестування функціональної частини, що стосується роботи додатку з серверами та налаштування серверів описана в цьому підрозділі. Описані тестові ситуації мають охоплювати тестування механізму налаштування серверів та локального зберігання вивантаженої і налаштованої інформації.

Список тестових ситуацій описано у табл. 3.15.

Деякі описані тести можна декомпонувати та проводити більш детально при використанні методу модульного тестування. В такій ситуації логіка додатку може потребувати створення деяких вхідних точок та зміненого алгоритму для мети тестування.



Таблиця 3.15.

## Тестові ситуації функціоналу налаштування серверів

№	Опис	Вхідні дані	Результат
1	Завантаження функціоналу налаштування серверів	-	Зазначене вікно завантажуються
2	Додавання нового серверу	Існуючий сервер	Сервер успішно додано
3	Редагування існуючого серверу	Існуючий сервер, посилання на якого змінилось	Посилання на сервер успішно змінено
4	Видалення існуючого серверу	Існуючий сервер	Сервер успішно видалено
5	Оновлення даних серверу	Існуючий сервер, застаріла інформація у клієнті	Дані серверу успішно оновлено
6	Тестування з'єднання серверу	Існуючий сервер, запис про сервер у клієнті	З'єднання успішно протестоване
7	Збереження/завантаження даних локально	Додавання серверу, закриття додатку, відкриття додатку	Дані успішно зберігаються /завантажуються

**3.15. Математичне ядро застосунку**

Ця частина додатку є ключовою і не має прямого впливу на роботу додатку. Але її роль є достатньо важливою для коректної роботи програми. Цю частину потрібно тестувати виключно методом модульного тестування. Список позитивних тестових ситуацій описано у табл. 3.16.

Успішне проходження зазначених тестів означає, що математичне ядро застосунку працює без похибок, підтверджуючи правильну роботу основи клієнтського додатку.

Таблиця 3.16.

## Тестові ситуації математичного ядра

№	Опис	Вхідні дані	Результат
1	Перевірка формул основного методу обчислення	Список вхідних даних та результатів для окремих формул	Очікувані результати співпадають з обчисленими
2	Перевірка обчислень основного методу	Список вхідних даних та результатів	Очікувані результати співпадають з обчисленими
3	Перевірка накладення коефіцієнтів	Список результатів обчислення основного методу та коефіцієнтів до точок. Список результатів	Очікувані результати співпадають з обчисленими
4	Перевірка результатів інтерполяції	Список результатів обчислення основного методу. Список інтерпольованих значень	Очікувані результати співпадають з обчисленими
5	Перевірка результатів динаміки забруднення	Список результатів обчислення основного методу за точками	Очікувані результати співпадають з обчисленими
6	Перевірка можливих точок, що можуть бути забруднювачами	Список фактів, що зв'язуються на точці. Ідентифікатор точки та речовини	Список об'єктів класифікується згідно з очікуваним

## Висновки до розділу 3

Розділ описує наступні частини дослідження:

1. Описано критерії ефективності та обмеження до досліджуваної системи.
2. Зокрема, описані критерії та принцип валідації до коректності роботи методу:

- коригування результатів забруднення;
- інтерполяції для зглаження прогнозованих результатів;
- визначення джерела забруднення.

3. Описані методи формуються такі критерії ефективності, як зменшення вірогідності набутої похибки за рахунок виведених коефіцієнтів, збільшення кількості виведених результатів та створення можливості визначення суб'єкту забруднення.

4. Описані критерії правильності та тестування інформаційної системи. Зокрема, таких частин, як сервер, адміністративна панель та настільний додаток. Такі частини функціоналу, як математичне ядро клієнтського застосунку, REST API серверу та частини інтеграції серверу з датчиками вимірювання швидкості течії формують критерій збільшення швидкості проведення оцінювання.

В результаті описаних частин можна зробити висновок, що в дисертаційній роботі охоплено повний список критеріїв ефективності до представлених розроблених методів.

## РОЗДІЛ 4

### ДОКУМЕНТАЦІЯ ДО СИСТЕМИ RESIT

#### 4.1. Інформація про систему

Як зазначалось, RESit є клієнт-серверною інформаційною системою, яка складається з комплексу застосунків, що мають на меті вирішення проблеми прогнозування наслідків забруднень в умовах надзвичайних ситуацій.

Основним завданням роботи системи є створення інформаційної технології для створення симуляції та обчислення наслідків забруднення в умовах надзвичайних ситуацій.

Оскільки, система є клієнт-серверною, необхідним є визначення програмних вимог для клієнтської частини, серверної частини, бази даних та утиліти проведення навчання.

Для роботи серверу необхідно встановити інтерпретатор Python 3+. Тому, може бути використана будь-яка \*nix подібна система або версія Windows Workstation/Server, що підтримує роботу з інтерпретатором.

Частина бази даних працює на базі MySQL серверу. Необхідно встановити MariaDb Server або MySQL Server.

Клієнт вимагає встановлення .NET Framework 4.5, отже такі ОС, як Windows Vista SP2, Windows 7 SP1, Windows 8, Windows 8.1, Windows 10, Windows 11, Windows Server 2008 R2 SP1(x64), Windows Server 2008 R2 SP2, Windows Server 2012, Windows Server 2016, Windows Server 2019, що підтримують роботу додатку.

#### 4.2. Робота с сервером

Як зазначалось, метою серверу є постачання настільному додатку оновлених даних в бази даних та оновлення бази даних з адміністраторського та настільного додатку. Як зазначалось, для роботи серверу необхідно встановити Python. Встановленням є копіювання змісту додатку в зручне місце. Налаштування серверу відбувається в файлі settings.py. Важливою частиною налаштувань є настройки підключення до бази даних, що налаштовується масивом DATABASES.

Запуск сервісу відбувається шляхом запуску наступної команди з папки програми *python manage.py runserver*.

Після цього буде показано консольний інтерфейс з інформацією про те, що сервер запущений та працює.

### 4.3. Робота з адміністративним додатком

Адміністративна панель є частиною серверу. При відкритті посилання *http://<ip адреса>:8000/admin* можна потрапити на панель адміністратора.

**Створення користувача.** Перед першим запуском панелі потрібно додати користувача командою *python manage.py createsuperuser*. Далі потрібно слідувати інструкціям та ввести ім'я користувача, пароль, тощо.

**Аутентифікація.** При переході за посилання адміністративної панелі, відкриється форма аутентифікації.

Для входу в панель, необхідно ввести ім'я користувача та пароль.

**Головна сторінка та функціонал програми.** Далі, користувачеві відображається головне вікно зі списком сутностей, та можливістю змінити пароль.

Основним функціоналом адміністративної панелі є:

- додавання нових користувачів, редагування інформації про них та видалення користувачів (здебільшого, користувачів, що мають права на роботи з адміністративною панеллю);

- зміна паролю поточного користувача;

- додавання, редагування та видалення записів, що стосуються системи RESit.

В цілому, існує можливість взаємодії з такими сутностями, як:

- користувачі та групи (Users, Groups)– вмонтовані в функціонал функції;
- точки моніторингу (Points);
- маршрути точок моніторингу (Points relations);
- коефіцієнти похибки для точок (Point coefficients);
- речовини (Substances);
- небезпечні об'єкти (Dangerous objects);

- визначення факту роботи небезпечних об'єктів з речовинами (Dangerous objects substance relations);
- визначення відношення небезпечних об'єктів до точок моніторингу (Point dangerous objects);
- опис ситуації (Situations);
- опис ситуації за точками (Situation points results);
- опис реальної ситуації за точками (Actual situation point results).

#### 4.4. Робота з утилітою навчання

Утиліта навчання є технічною частиною системи, що виконує необхідні навчання визначеним методом. Необхідним для навчання методом є метод коригування результатів прогнозування. Вхідними даними для утиліти є масив даних з бази даних RESit. Результатом є оновлені коефіцієнти, що записуються в базу даних.

Для роботи додатку необхідно встановити Python 3+.

Утиліта не вимагає інсталяції, але вимагає налаштування з'єднання.

Для запуску утиліти необхідно ввести наступну команду в папці утиліти:

```
python main.py
```

Після цього утиліта завантажить необхідну інформацію про ситуації з бази даних, проведе навчання та оновить необхідні коефіцієнти.

#### 4.5. Робота з програмою

Для запуску програми, необхідно запустити файл Main\_984440.exe.

Початкове вікно є основним та стартовим для роботи з додатком (рис 4.15).

Початкове вікно має наступні функції:

- створення нового обчислення в ручному режимі;
- створення нового обчислення в автоматичному режимі;
- перегляд результатів в загальному режимі;
- виклик налаштувань роботи з серверами, що містять інформацію про ріки та токсичні речовини;

- створення графіків за таблицями.

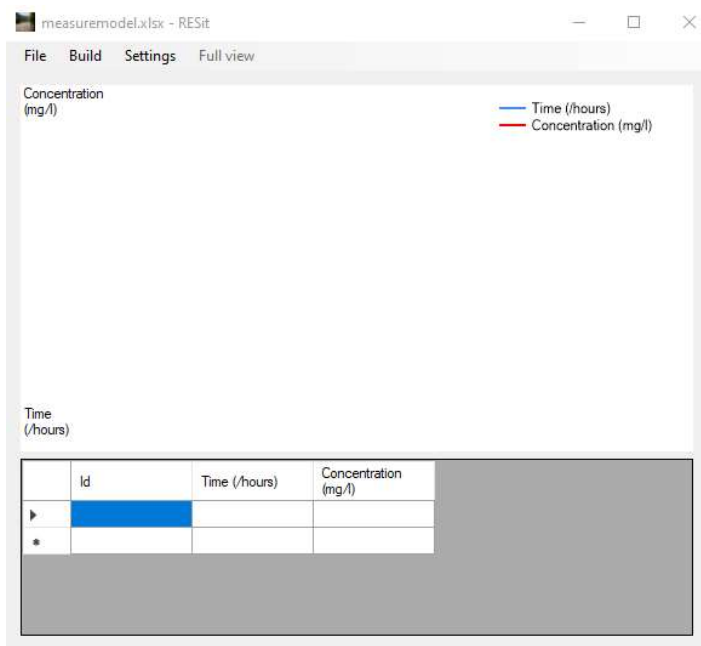


Рис. 4.1. Початкове вікно

**Панель управління програмою.** Панель управління дозволяє викликати функції, що реалізовані в програмі. Вигляд панелі показано на рис. 4.2.

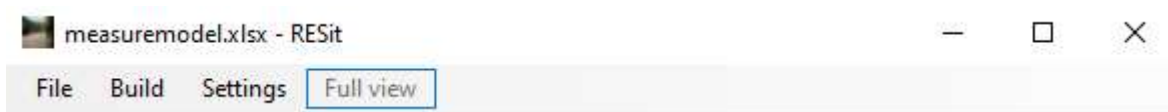


Рис. 4.2. Панель управління

Вкладка «Файл» містить такі опції:

- Новий:
  - «Ручний» режим – викликає майстра ручного обчислення;
  - «Автоматичний» режим – викликає майстра автоматичного обчислення.
- Експортувати в:
  - JPG - зберігає графіки у вигляді jpg малюнка;
  - XLS – зберігає таблиці у вигляді xls малюнка;
- Вийти - вихід з програми.

Вкладка «Побудувати» містить наступний набір функцій, що управляють графіками:

- Побудувати – побудувати графіки за даними в таблиці;

- Очистити – очистити вікно побудови графіків.

Вкладка «Налаштування» відповідає за налаштування роботи з серверами, підключеними в додатку:

- Налаштування сервера – містить в собі ряд функцій, що дозволяють налаштовувати програму для роботи з серверами.

Кнопка «Повний стан» відкриває вікно, що містить більш детальну інформацію про результати обчислення.

### Таблиця для роботи з точками

Таблиця дозволяє вносити результати власних обчислень (Рис. 4.3). На основі цих даних можна будувати графіки.

	Id	Time (/hours)	Concentration (mg/l)
▶			
*			

Рис. 4.3. Таблиця для роботи з точками

Таблиця відображає дані, за якими побудовані графіки. Містить три колонки:

- Id – порядковий номер точки. Важливо, щоб порядковий номер був виставлений у правильній послідовності (або інкрементом, або за принципом попереднього номеру, що є меншим за наступний), для коректного відображення графіків. У випадку невірного введення даних існує можливість автосортування за визначником натиснувши на «Nr»;

- Time (/hours) – містить дані за характеристикою «Тр» (час, через який концентрація набуде свого піку у визначеній точці);

- Concentration (mg/l) – містить дані за характеристикою «Ср» (пік концентрації).

### Графіки. Приклад побудованого графіку наведений на Рис. 4.4.

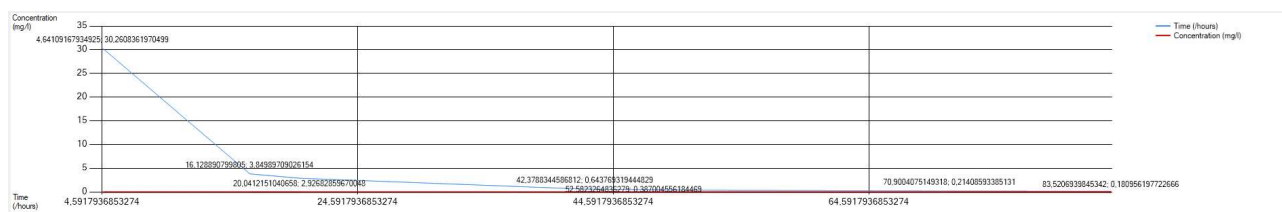


Рис. 4.4. Графік, що відображає результати прогнозування



Графіки відображають дані про значення характеристик з точками на основі табличних даних. За абсцисою відображається час, а за ординатою - концентрація.

#### 4.6. Майстер ручного обчислення

Призначенням майстру ручного обчислення є створення обчислення на основі повного набору самостійно введеного масиву характеристик. Майстер включає запит даних про ситуацію та запит характеристик про точку.

Даний підрозділ описує функціонал та логіку його роботи.

#### 4.7. Початкове вікно моделювання

Початкове вікно є першим етапом роботи майстру. Призначенням є запит даних про ситуацію. Викликається шляхом вибору опції головного вікна Файл – Новий – Ручний режим. Приклад форми показано на рис. 4.5.

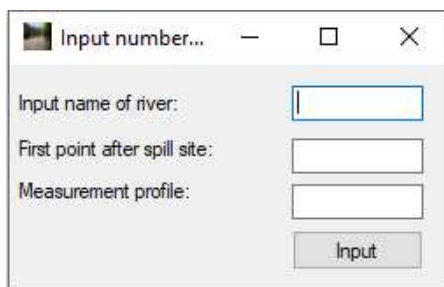


Рис. 4.5. Початкове вікно

Необхідно ввести ім'я ріки, порядковий номер початкової (що є 1-ою точкою моніторингу після виливу небезпечної речовини) та кінцевої (точка моніторингу) точки вимірювання та натиснути клавішу «Ввести». Після цього відкривається форма вводу даних про точку та починається цикл заповнення інформації про точки.

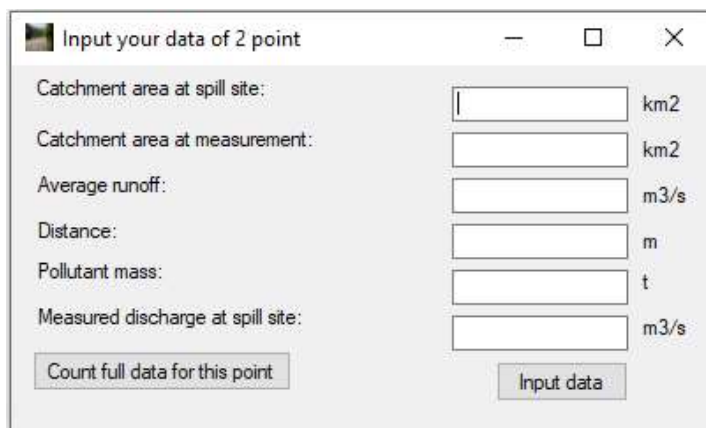
#### 4.8. Форма вводу даних

Цикл заповнення даних починається з форми, наведеної на рис. 4.6.

Після чого потрібно ввести масив даних (зразок введення даних показано на рис. 4.7):

- площу водозбору в місці розливу;

- площу водозбору при вимірюванні;
- середній стік;
- дистанцію;
- забруднюючу масу;
- витрати води в момент вимірювання.



Input your data of 2 point

Catchment area at spill site:  km2

Catchment area at measurement:  km2

Average runoff:  m3/s

Distance:  m

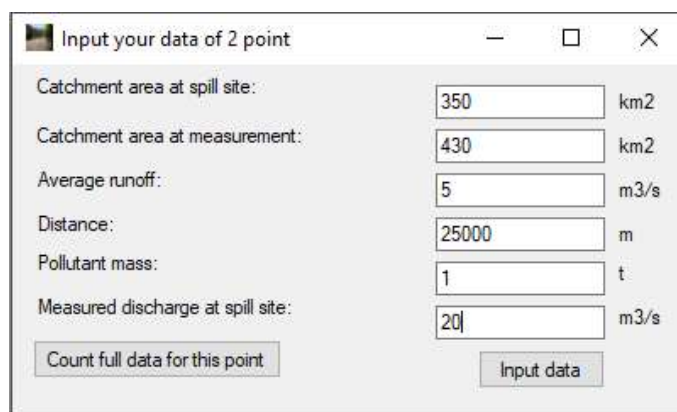
Pollutant mass:  t

Measured discharge at spill site:  m3/s

Count full data for this point

Input data

Рис. 4.6. Початок циклу заповнення інформації



Input your data of 2 point

Catchment area at spill site:  km2

Catchment area at measurement:  km2

Average runoff:  m3/s

Distance:  m

Pollutant mass:  t

Measured discharge at spill site:  m3/s

Count full data for this point

Input data

Рис. 4.7. Заповнення даних про точку

Після введення всіх необхідних даних про точку необхідно натиснути на клавішу «Підрахувати повні дані», або клавішу «Ввести дані». Кнопка «Підрахувати повні дані» відкриває форму «про точку», що відображає повний масив даних за конкретною точкою за всіма можливими результуючими характеристиками, задіяними в методиці. Кнопка «Ввести дані» приймає дані про першу точку та запитує дані про наступні точки.

Слід зауважити, що параметри «Площа водозбору в місці розливу» та «Забруднююча маса» залишаються незмінними для наступних точок, бо є сталими значеннями для поточного вимірювання.

Зразок заповнення даних вказано на рис. 4.8.

Рис. 4.8. Заповнення даних про наступні точки

#### 4.9. Форма «про точку» (вікно відображення повних даних про точку)

Рис. 4.9. Форма «про точку»

Форма відображає результуючі характеристики точки вимірювання. При цьому результуючими характеристиками є:

- середнє арифметичне площі водозбору;
- безрозмірна площа;
- безрозмірна витрата води;
- час підходу піка концентрації;
- швидкість піку концентрації забруднення;

- можлива максимальна швидкість;
- концентрація піка.

Форму показано на рис. 4.9.

Вікно містить клавішу «Зберегти в \*.xls», що дозволяє відобразити ці дані у вигляді в Excel. Приклад генерованого \*.xls файлу показано на рис. 4.10.

	A	B	C
1	<b>Input data:</b>		
2	Catchment area at spill site:	350000	km2
3	Catchment area at measurement:	430000	km2
4	Average runoff:	4,5	m3/s
5	Distance:	25000	m
6	Pollutant mass:	2E+09	t
7	Measured discharge at spill site:	15	m3/s
8			
9	<b>Output data:</b>		
10	Arithmetic mean catchment area:	390000	m2
11	Dimensionless area:	6780025	m2
12	Dimensionless discharge:	3,333333	m2
13	Unit-peak concentrations:	561,5664	seconds
14	Velocity of the peak concentration:	3,766883	m/s
15	Probable maximum velocity:	18,05128	m/s
16	Probable traveltime of the peak:	1,843552	hours
17	Peak concentration:	74,87552	mg/l

Рис. 4.10. Вигляд таблиці Excel

#### 4.10. Генерація результуючого файлу

По закінченню введення необхідних даних, програма вносить результати в таблицю та будує графік на основі таблиці.

Приклад головного вікна, що містить результати обчислення показано на рис. 4.11.

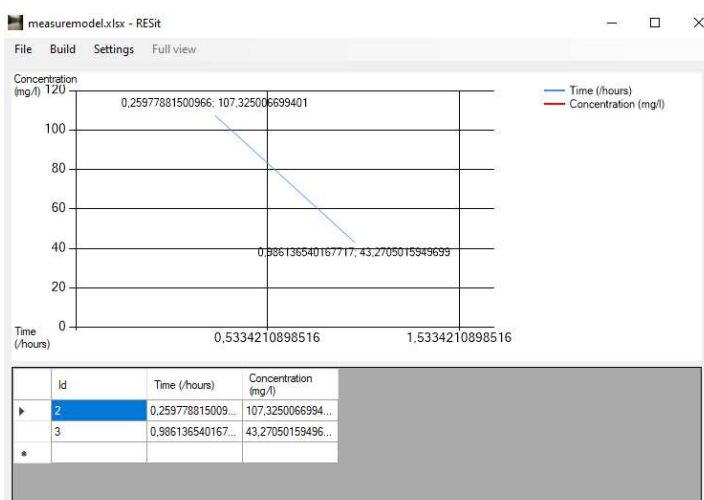


Рис. 4.11. Вікно результатів вимірювання

**Вигляд результуючого файлу.** Дані, експортовані у \*.xls файлі, можна відкрити за допомогою програми Excel, та інших аналогів, таких як Google Таблиці, OpenOffice Calc, LibreOffice Calc, WPF Office тощо. Файл містить таблицю з трьома стовбцями:

- «Nr» – порядковий номер точки;
- «Tr» – час, через який концентрація набуде свого піку;
- «Cr» – пік концентрації.

Вигляд таблиці можна побачити на рис. 4.12.

	A	B	C
1	Nr	Tr	Cr
2	2	0,259779	107,325
3	3	0,986137	43,2705

Рис. 4.12. Вигляд таблиці в генерованому файлі

Такі дані можуть бути використані для збереження у вигляді звіту або для подальшого аналітичного чи статичного аналізу результатів.

#### 4.11. Майстер автоматичного обчислення

Призначенням майстру автоматичного обчислення є створення обчислення на основі даних, що містяться у підключеній базі даних. Майстер включає наступні кроки:

- вибір серверу;
- вибір точки викиду та останньої точки вимірювання наслідків забруднення;
- введення таких ситуативних характеристик, як кількість викинутої речовини та поточні витрати води.

Підрозділ описує функціонал та його логіку роботи.

#### 4.12. Виклик обчислення та вибір серверу

Для початку проведення нового обчислення, на панелі головної форми потрібно натиснути «Файл – Новий – ‘Автоматичний’ режим», після цього з'явиться список збереженої в програмі інформації, про сервери, що містять необхідні дані (а саме, список збережених серверів).

Приклад форми показано на рис. 4.13.

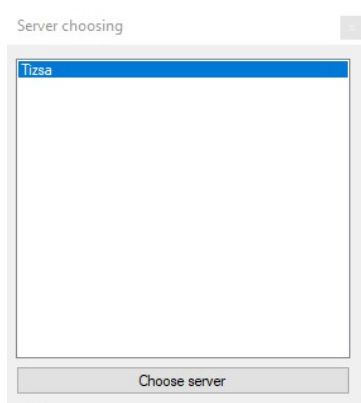


Рис. 4.13. Вікно, що містить список збережених серверів

Далі необхідно вибрати необхідний сервер та натиснути клавішу «Вибрати сервер». Після чого висвітиться вікно вибору точок для вимірювання.

#### 4.13. Вибір точок для вимірювання

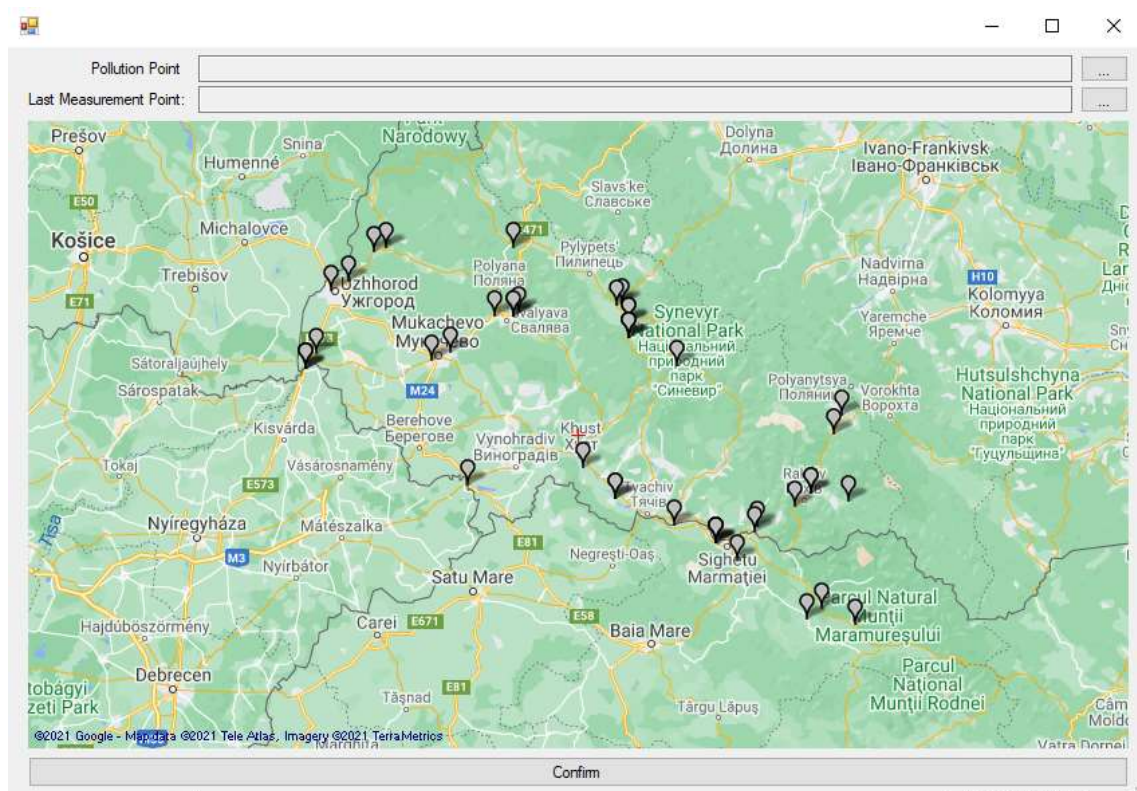


Рис. 4.14. Форма вибору точок

Наступним кроком за вибором серверу, з'явиться вікно, що показує список всіх точок, нанесених на мапі. Вибір точок здійснюється шляхом натискання на кнопку «...» навпроти типу точки, що має бути вибрана та безпосереднього вибору точки на мапі. Першою точкою необхідно вибрати точку викиду. Далі необхідно



натиснути на кнопку «...» навпроти строки, що потребує введення останньої точки вимірювання та вибрати кінцеву точку. В результаті вибору режиму введення останньої точки вимірювання буде показано список доступних точок для вимірювання, тобто точок нижче за течією. Таким чином можливість вибору неіснуючої точки виключена. Вигляд форми можна побачити на рис. 4.14.

Після вибору всіх точок, потрібно натиснути на клавішу «Підтвердити», після чого програма запитає про правильність введення даних. Далі викликається форма, на якій потрібно ввести необхідні дані про точки.

#### 4.14. Введення необхідних даних про точку

Наступним кроком, після вибору точок, є введення необхідних вхідних даних. При цьому відкривається форма, у яку необхідно ввести дані про ситуацію, тобто масу забруднювача скинутого в ріку, поточні витрати для точок та речовини. Маса забруднювача вводиться в текстове поле в тоннах. Поточні витрати води вводяться в текстове поле нижче. Значення поточних витрат заздалегідь заповнене значенням, що зберігаються на сервері. Оскільки характеристика є динамічною, з метою актуалізації існує можливість оновлення значень. Зробити таку дію можна шляхом натискання клавіші «Оновити». Для вибору точок слід користуватись клавішами «Наступна точка» або «Попередня точка», зміни будуть відображатись в таблиці.

№	Name of point	Measurement discharge
4	Jasinja settlement	8.82
5	Rachiv city - abo...	26.9
6	Rachiv city - bel...	30
16	Sighetul Marmatiei	77.2
17	Tyachiv city - ab...	109
20	Tyachiv city - bel...	157
21	Khust city - abov...	164
28	Khust city - below...	193
29	Vylk settlement	195

Measurement discharge at spill site for 4 point Jasinja settlement: 8.82 m3/sec

Buttons: Previous point, Next point, Do measures

Рис. 4.15. Форма введення необхідних даних про точку

Строки таблиці мають 2 кольори: зелений и червоний. Зелений означає, що все введено вірно, червоний – що якусь необхідну характеристику не введено, або введено невірно. Вигляд форми можна побачити на рис. 4.15.

Після введення всіх вхідних даних слід натиснути клавішу «Зробити обчислення». Програма перевірить правильність введення. Якщо все вірно, то результати обчислення будуть передані на головну форму для відображення результатів, як ні – потрібно буде виправити дані.

#### 4.15. Результат

Результат обчислення буде відображено у вигляді, показаному на рис. 4.30.

Зазначена форма відображає дані у стислому вигляді. При цьому зображується графік спадання концентрації по відношенню до рівня гранично допустимої концентрації. Тобто, на графіку можна побачити, в якому місці та через який час рівень забруднення повернеться до норми. Також на вікні існує можливість перегляду даних у табличному режимі. В таблиці висвітлені такі дані, як номер точки, ім'я точки та результати обчислень за точкою.

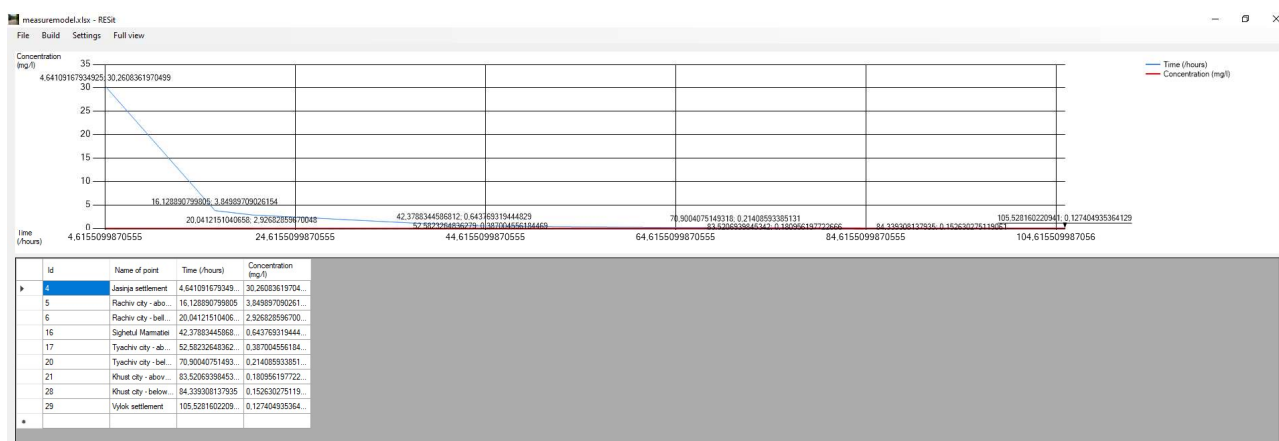


Рис. 4.16. Головне вікно з результатами обчислень

В цьому режимі доступна функція відображення даних в більш широкому спектрі характеристик та результатів, що можуть допомогти оцінити масштаби ситуації. При цьому, на екрані відображається більший спектр характеристик, що є аналітично або методологічно виведеними на основі різних методів інтелектуальної обробки вхідних або вихідних даних. Зазначений функціонал викликається при натисканні кнопки «Повний стан».



#### 4.16. Налаштування програми для роботи з сервером

Частина налаштування списку серверів призначення для проведення різних операцій, що стосується серверів та їх даних. Викликати їх можна з головного меню, шляхом натискання на панелі «Налаштування - Налаштування сервера», після чого буде викликане вікно, показане на рис. 4.17.

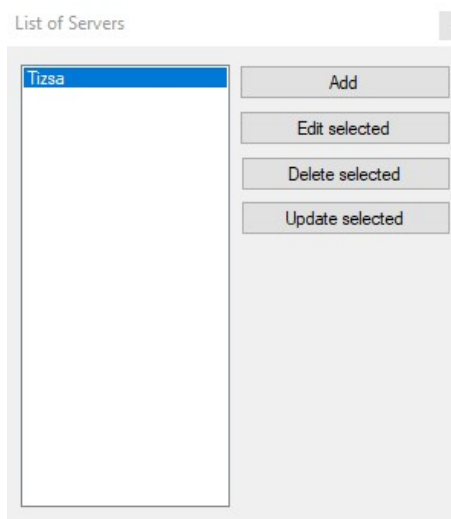


Рис. 4.17. Вікно налаштувань для роботи з серверами

Вікно «Налаштування сервера» містить список серверів, що завантажені в програму і ряд таких функцій, як:

- додати вибраний сервер – викликає форму для додавання даних про сервер, а згодом завантаження даних з цього серверу.
- редагувати вибраний сервер – викликає форму з даними про сервер для зміни цих даних і згодом їх оновлення.
- видалити вибраний сервер – видаляє вибраний зі списку сервер з пам'яті програми;
- оновити вибраний сервер – оновлює всю інформацію з вибраного зі списку серверу.

Збереження даних відбувається після будь-якої операції введення даних. Зазначений масив інформації зберігається у файл `servers.dat` у папці в додатком.

#### 4.17. Дашборд: режим повного опису результату

Дашборд є частиною клієнтського додатку RESit, що призначена для детального виведення результатів обчислення. Включає в себе наступний перелік результатів:

- інформацію в табличному вигляді про такі основні характеристики, як час, через який концентрація забруднювача набуде піку та пік концентрації в точках моніторингу;

- інформацію в табличному вигляді про такі основні характеристики, як час, через який концентрація набуде піку та пік концентрації між сусідніми точками моніторингу;

- графік спадання концентрації забруднювача у точці моніторингу в динаміці;

- графік перевищення концентрації забруднювача у порівнянні з гранично допустимою величиною;

- графік між точками спадання концентрації в часі та просторі;

- список можливих небезпечних об'єктів, що можуть забруднювати вибрану точку;

- список всіх описаних характеристик з урахуванням коефіцієнтів похибки.

Має наступний функціонал:

- збереження результату у вигляді KML файлу;

- завантаження результату на сервер у список надзвичайних ситуацій;

- збереження результату у вигляді JPG файлу.

Доступний, якщо головне вікно відображає результат створений шляхом «автоматичного» обчислення. Вікно викликається шляхом натискання клавіші «Повний стан» на панелі інструментів на панелі.

Вигляд описаного вікна показано на рис. 4.18.

Інтерфейс користувача поділено на наступні функціональні частини:

- результат забруднення за всіма точками;

- результат забруднення за всіма точками у вигляді графіку;

- інформація по забрудненню, що стосується конкретної точки;

- прогнозування можливих об'єктів, що забруднюють вибрану точку викиду.

Логіка інтерфейсу користувача побудована на розширювачах, що містять відповідну функціональну частину.

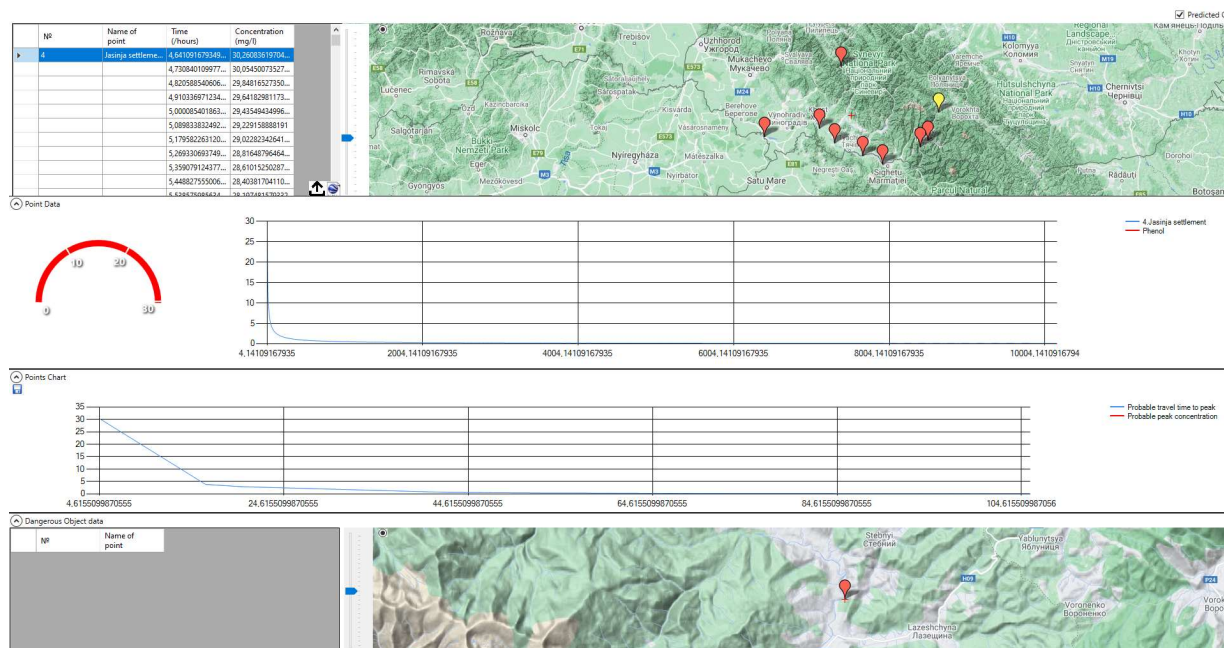


Рис. 4.18. Вікно, що відображає «Повний стан»

## Висновки до розділу 4

Розділ описує документацію користувача для роботи з системою. Висвітлюється питання:

- роботи інформаційної технології з точки зору користувача. Документація охоплює опис таких складових системи, як робота з сервером, адміністративним додатком, утилітою навчання та клієнтським програмним забезпеченням;

- встановлення та супроводження складових частини інформації системи.

В цілому, розділ описує сторону практичного застосування системи, що відображає практичну значимість при застосуванні зазначеного програмного забезпечення та досліджень дисертаційної роботи.

## ВИСНОВКИ

В результаті дисертаційних досліджень вирішені наступні актуальні питання:

- зменшення обсягу масиву даних для введення за рахунок створення інформаційної системи, що має базу даних з пропозицією щодо можливих інтеграційних рішень зазначеної системи з датчиками вимірювання швидкості потоку;
- впровадження системи інтелектуальної обробки результату, що включає коригування похибки та визначення можливого джерела забруднення.

Зокрема:

1. Розроблено метод коригування результатів прогнозування концентрації забруднювача, на основі використання методів регресії нейронної мережі, що дозволяє підвищити точність характеристики піку концентрації в конкретній точці моніторингу, тобто точність результатів. Конкретні результати ефективності розробленого методу можуть варіювати залежно від конкретного русла водойми та навчанні методу на конкретних даних.

2. Розроблено метод прогнозування рівня забруднення між конкретними точками за допомогою методу інтерполяції та алгоритму рекурсії, що дозволяє отримати згладження прогнозованих результатів, отримати результат характеристики піку концентрації між точками моніторингу та збільшити масив результатів для ширшого розуміння стану русла в умовах надзвичайних ситуацій.

3. Розроблено метод визначення джерела забруднення на основі алгоритму фільтрації та сортування даних, що працює на основі бази фактів. Конкретні результати ефективності методу можуть варіюватися залежності від впровадження його в конкретне русло та наповнення бази даних щодо конкретної території.

4. Удосконалено інформаційну технологію прогнозування забруднення річок в умовах надзвичайних ситуацій за рахунок:

- покращення архітектури існуючого програмного забезпечення RESit, що робить підтримку коду більш гнучкою та такою, що можна підтримувати;
- створення таких частин інформаційної системи RESit, як сервер та утиліта навчання, що дозволяють впровадити систему логічної перевірки цілісності даних,

інтеграцію з датчиками вимірювання швидкості потоку та можливість навчання нейронних мереж;

- впровадження методу коригування результатів прогнозування концентрації забруднювача, як функціоналу, що використовує нейронну мережу;

- впровадження методу інтерполяції, як частини результатів прогнозування наслідків забруднення;

- впровадження методу визначення джерела забруднення на основі бази даних фактів з використанням алгоритму фільтрації та сортування даних;

- розробки логіки взаємодії серверу з датчиками вимірювання швидкості потоку русла з опису можливих шляхів взаємодії з зазначеною точкою входу.

Удосконаленням системи RESit є:

- наявність інформаційної системи з можливістю децентралізації та наявності автономності при можливості роботи клієнтського додатку;

- наявність інтелектуальної обробки результату та функціоналу, що дозволяє зменшити необхідність введення перемінного масиву вхідних даних.

Система RESit також дозволяє виводити результат у вигляді графіків та мап, зберігати його у вигляді KML файлів, що дають можливість інтеграції системи з різними ГІС системами та з більшістю автоматизованих рішень.

В результаті дисертаційного дослідження, такі сформовані критерії, як:

- збільшення швидкості проведення оцінювання за рахунок зменшення кількості годин та ресурсів;

- зменшення вірогідності набутої похибки;

- збільшення кількості виведених результатів;

- створення можливості визначення потенційного суб'єкту забруднення, за рахунок удосконалення системи інтелектуального розрахунку, а саме:

- збільшення швидкості проведення оцінювання – створення інформаційної системи, що включає базу даних та інтеграцію з датчиком вимірювання швидкості потоку русла;

- зменшення вірогідності набутої похибки - розробка методу коригування результатів прогнозування концентрації забруднювача в умовах надзвичайних ситуацій;

- збільшення кількості виведених результатів – розробка методу визначення джерела забруднення та удосконалення інформаційної технології прогнозування забруднення річок в умовах надзвичайних ситуації, функціоналу відображення результату клієнтським додатком;

- методу прогнозування рівня забруднення між конкретними точками.

Перелічені розробки та удосконалення системи RESit у поєднанні з інформаційними технологіями визначають подальші можливості щодо прогнозування та масштабування розвитку наслідків надзвичайних ситуацій у водоймах.

Перспективи системи RESit до гнучкого впровадження та подальшої інтеграції в існуючі програмні продукти дозволить покращити розв'язування функціональних задач, що проводяться органами державного управління щодо стану водних ресурсів, службами з надзвичайних ситуацій, екологічними інспекціями, розрахунково-аналітичними групами, іншими компетентними сторонами, які займаються проблемами оцінки впливу забруднюючих факторів на стан довкілля, оптимізацію контролю, регулювання та моніторингу, обґрунтування найбільш оптимальних рішень з усунення або зниження рівнів ризику наслідків надзвичайних ситуацій.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Конвенція з охорони та використання транскордонних водотоків та міжнародних озер ООН; Конвенція, Міжнародний документ від 17.03.1992. Електронний ресурс. Режим доступу: [http://zakon1.rada.gov.ua/laws/show/994\\_273](http://zakon1.rada.gov.ua/laws/show/994_273).
2. Кафедра Безпеки життєдіяльності та охорони праці. «Курс лекцій № 2 з навчальної дисципліни Цивільний захист», ДУТ, Київ, 2014, с. 2.
3. Биченок М.М. Основи інформатизації управління регіональною безпекою. М.М. Биченок К.: Інститут проблем національної безпеки РНБО України, 2005. 196 с. Режим доступу: <https://docs.google.com/document/d/1GOLRpS77Ceggbf9B-Oqu4KJMglyzEEzN/edit>.
4. Chapra, S.C., Pelletier, G.J., Tao, H. QUAL2K: A Modeling Framework for Simulating River and Stream Water Quality. Version 2.11. Documentation and User's Manual, Civil and Environmental Engineering Department, Tufts University, Medford, USA, 2008.
5. Horst Behrendt, Markus Venohr, Ulrike Hirt, Jürgen Hofmann, Dieter Opitz, Andreas Gericke The model system. MONERIS. Version 2.0. User's Manual. 2007. pp. 99.
6. WASP8 Model documentation for: Water Transport, Sediment Transport, Eutrophication, Macro Algae and Periphyton, Sediment Diagenesis, Water Temperature, pH and Alkalinity, Light. [Online]. Available: <https://www.epa.gov/ceam/wasp-model-documentation>
7. Lawrence A. Burns Exposure Analysis Modeling System (EXAMS): User Manual and System Documentation, Georgia, 2020.
8. Frick W.E., Roberts P.J.W., Davis L.R., Keyes J., Baumgartner D.J., George K.P. Dilution Models for Effluent Discharges. 4-th Edition (Visual Plumes). Athens, Georgia, 2003, pp. 148. [Online]. Available: <https://www.epa.gov/sites/production/files/documents/VP-Manual.pdf>.
9. Science Documentation. Revised Universal Soil Loss Equation. Version 2 (RUSLE2) (for the model with release date of May 20, 2008), Washington, 2013, 335 p.

[Online]. Available: [https://www.ars.usda.gov/ARSEUserFiles/60600505/RUSLE/RUSLE2\\_Science\\_Doc.pdf](https://www.ars.usda.gov/ARSEUserFiles/60600505/RUSLE/RUSLE2_Science_Doc.pdf).

10. Environmental Modeling Community of Practice. MMSOILS US EPA. [Online]. Available: <https://www.epa.gov/ceam/mmsoils>.

11. Laniak G. F., Droppo J. G., Faillace E.R., Gnanapragasam E.K., Mills W.B., Streng D.L., Whelan G., Yu C. An Overview of a Multimedia Benchmarking Analysis for Three Risk Assessment Models: RESRAD, MMSOILS, and MEPAS. Risk Analysis. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/9202489/>.

12. Loucks D.P., van Beek E. Water Resource Systems Planning and Management, 2017. DOI:10.1007/978-3-319-44234-1.

13. Tayfour G., Singh V. Predicting longitudinal dispersion coefficient in natural streams by artificial neural network, journal of hydraulic engineering, т. Vol.131, 2005, pp. 991-1000. DOI: 10.1061/(ASCE)0733-9429(2005)131:11(991).

14. Kashefipour, S. A. Falconer Longitudinal dispersion coefficients in Natural channels, Proceedings of the fifth International hydro informatics Conference, 1-5 July 2002, pp. 95-102.

15. Riahi-Madvar H., Ayyoubzadeh S.A. Developing an Expert System for Predicting Pollutant Dispersion in Natural Streams Expert Systems, Shanghai, InTech, 2010, pp. 225-226.

16. Abdeveis S., Sedghi H., Hassonizadeh H., Babazadeh H. Application of Water Quality Index and Water Quality Model QUAL2K for Evaluation of Pollutants in Dez River, WATER RESOURCES, Volume: 47, Issue: 5, Iran, 2020. DOI:10.1134/S0097807820050188.

17. Chapra, S.C., Pelletier, G.J., Tao, H. Civil and Environmental Engineering Dept., Tufts University. Medford, MA, 2008.

18. Закорчевний М. Загальні принципи моделювання річкових систем з використанням програми «Qual2k» (на прикладі верхньої і середньої течії р. Дністер). Техногенна екологічна безпека та цивільний захист. – 2011. – випуск 3, С. 105-110.



19. Frank Wechsung, Stefan Kaden, Markus Venohr, Jürgen Hofmann, Jens Meisel, Zhenci XU Sustainable Water and Agricultural Land Use in the Guanting Basin under Limited Water Resources, 2017, pp. 397.

20. AQUATOX. Linking water quality and aquatic life. Technical Documentation. Electronic resource. Release 3.2. September 2018. [Online]. Available: <https://www.epa.gov/ceam/aquatox>.

21. Jaehoon Yeom, Injeong Kim, Minjeong Kim, Kyunghwa Cho, Sang Don Kim Coupling of the AQUATOX and EFDC Models for Ecological Impact Assessment of Chemical Spill Scenarios in the Jeonju River, Biology, Issue 9, 2020, Korea, 340 pp. DOI: 10.3390/biology9100340.

22. The Water Quality Analysis Simulation Program (WASP) model helps users interpret and predict water quality responses to natural phenomena and manmade pollution for various pollution management decisions. [Online]. Available: <https://www.epa.gov/ceam/water-quality-analysis-simulation-program-wasp>.

23. Robert B. Ambrose Jr., P.E, Tim A. Wool. WASP8 Stream Transport-Model Theory and User's Guide. Supplement to Water Quality Analysis Simulation Program (WASP) User Documentation, 2017, Washington.

24. Jan Sittersona, Sumathy Sinnathambyb, Rajbir Parmard, Joshua Koblich, Kurt Wolfed, Christopher D. Knightes Demonstration of an online web services tool incorporating automatic retrieval and comparison of precipitation data, Environmental Modelling and Software, Issue 123, 2020.

25. SWToolbox: A Surface-Water Toolbox for Statistical Analysis of Streamflow Time Series, Chapter 11 of Section A, Statistical Analysis, Book 4, Hydrologic Analysis and Interpretation, Virginia, USA, 2018.

26. Hofman D. MOIRA DSS – Architecture, Model Integration and User Interface Design. Studsvik Rad Waste AB. [Online]. Available: <https://www.researchgate.net/publication/240896053>.

27. Montea L., Brittain J. E., Gallego E., Håkansson L., Hofmane D., Jiménez A. MOIRA-PLUS: A decision support system for the management of complex fresh water ecosystems contaminated by radionuclides and heavy metals. Computers & Geosciences.

[Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0098300408001726>.

28. Monte L., Hakanson L., Steen J.V. D., Bergstroem U. MOIRA Brochure - A model-based computerised system for management support to identify optimal remedial strategies for restoring radionuclide contaminated aquatic ecosystems and drainage areas (MOIRA). Decision Support for emergency management and environmental restoration, Chapter: A Model-based computerised system for management support to Identify optimal remedial strategies for Restoring radionuclide contaminated Aquatic ecosystems and drainage areas: (MOIRA) Publisher: EURATOM, pp.26-27. January 2000, [Online]. Available: <https://www.researchgate.net/publication/278728229>.

29. Вимірювачі швидкості течії; Гідрографічне та гідрометричне обладнання; Каталог; Група «Деметра.5». URL: [http://www.demetra5.kiev.ua/ua/catalog/izmeritely\\_skorosty\\_techeniya](http://www.demetra5.kiev.ua/ua/catalog/izmeritely_skorosty_techeniya).

30. Operating instructions Mobile River Discharge Measurement System OTT Qliner 2. [Online]. Available: [https://www.hemmis.be/docs/ott/OTT\\_Qliner2\\_en.pdf](https://www.hemmis.be/docs/ott/OTT_Qliner2_en.pdf).

31. Fisher H., List J., Koh C., Imberger J., Brooks N. Mixing in Inland and Coastal Waters, Academic Press, 1979, pp. 104-138.

32. Jobson H. E. Prediction of Travel time and Longitudinal Dispersion in Rivers and Streams, 1996, pp. 69.

33. USGS Water-Resources Investigations Report 96-4013. [Online]. Available: <http://water.usgs.gov/osw/pubs/disp/dispersion.html>.

34. Вступ до машинного навчання (Частина 2). [Online]. Available: <http://specials.kunsht.com.ua/machinelearning2>.

35. Understand the Softmax Function in Minutes | by Uniqtech | Data Science Bootcamp | Medium. URL: <https://medium.com/data-science-bootcamp/understand-the-softmax-function-in-minutes-f3a59641e86d>.

36. A Gentle Introduction to the Rectified Linear Unit (ReLU). URL: <https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/>.

37. J. F. Steffensen Interpolation, Dover Publications, Mineola, New York, USA, 2006.
38. What is REST. [Online]. Available: <https://restfulapi.net/>.
39. C# docs - get started, tutorials, reference. | Microsoft Docs. [Online]. Available: <https://docs.microsoft.com/en-us/dotnet/csharp/>.
40. Our Documentation | Python.org. [Online]. Available: <https://www.python.org/doc/>.
41. MySQL :: MySQL Documentation. [Online]. Available: <https://dev.mysql.com/doc/>.
42. Django documentation | Django documentation | Django. [Online]. Available: <https://docs.djangoproject.com/en/4.0/>.
43. The Django admin site | Django documentation | Django. [Online]. Available: <https://docs.djangoproject.com/en/4.0/ref/contrib/admin/>.
44. Overview - NumPy v1.21. Manual. [Online]. Available: <https://numpy.org/doc/stable/>.
45. PyMySQL - PyPI. [Online]. Available: <https://pypi.org/project/PyMySQL/#documentation>.
46. Windows Forms for .NET 6 documentation. [Online]. Available: <https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/?view=netdesktop-6.0>.
47. What is WPF? - Visual Studio (Windows) | Microsoft Docs. [Online]. Available: <https://docs.microsoft.com/en-us/visualstudio/designers/getting-started-with-wpf?view=vs-2022>.
48. GitHub - judero01col/GMap.NET: GMap.NET Windows Forms & Presentation is an excellent open source, powerful, free and cross-platform .NET control. [Online]. Available: <https://github.com/judero01col/GMap.NET>.
49. Live Charts. [Online]. Available: <https://lvcharts.net/App/examples/Wpf/start>.
50. GitHub - JanKallman/EPPlus: Create advanced Excel spreadsheets using .NET. [Online]. Available: <https://github.com/JanKallman/EPPlus>.

51. The S.O.L.I.D Principles in Pictures | by Ugonna Thelma | Backticks & Tildes | Medium. [Online]. Available: <https://medium.com/backticks-tildes/the-s-o-l-i-d-principles-in-pictures-b34ce2f1e898>.

52. The Model-View-ViewModel Pattern – Xamarin. [Online]. Available: <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/enterprise-application-patterns/mvvm>.

53. MVP Design Pattern For Windows Forms. [Online]. Available: <https://www.c-sharpcorner.com/UploadFile/ankithakur/mvp-design-pattern-for-window-forms/>.

54. MySQL :: MySQL Workbench Manual :: 9.3.3.1 Adding an EER Diagram. [Online]. Available: <https://dev.mysql.com/doc/workbench/en/wb-creating-eer-diagram.html>.

55. Valeport. CDU Express - Software Manual, United Kingdom, July 2021.

56. В.Д. Шипулін Основні принципи геоінформаційних системи. Навчальний посібник. Харківська національна академія міського господарства, Харків, 2010, 312 с.

57. Bottacin Busolin A. Modeling the Effect of Hyporheic Mixing on Stream Solute Transport. Water Resource Research. 2019 DOI: <https://doi.org/10.1029/2019WR025697>, pp. 30.

58. Martin Peter Clarke Assessing the Water Quality Benefits of Riparian Buffer Zones in the Cherwell Catchment, Oxford. MSc, 2007, pp 84.

59. C. Gualtieri, P.A. Lopez-Jimenez Sustainability indicators for river water quality management in urban areas. 2012 International Congress on Environmental Modelling and Software Managing Resource of a Limited Planet, Leipzig, Germany.

60. Huimin Li, Yongchao Cao, Limin Su Multi-dimensional dynamic fuzzy monitoring model for the effect of water pollution treatment. Environmental Monitoring and Assessment. Article Number 191, DOI: <https://doi.org/10.1007/s10661-019-7502-4>.

61. Srimanta Pal, Ruma Pal A Mathematical Model for the Effect of Water Pollutants on the Growth Pattern of Polen. Biom J. 36 1994, 2, p 239-253.

62. Michael Waldon Lagrangian Water Quality Modeling and TMDL Development in Low-Gradient Streams. ASCE-CSCE Environmental Engineering Conference. Norfolk, VA, July 1999, pp. 563-569.

63. G. Boano, C. Comporeale, R. Revelli, L. Ridolfi Sinuosity-driver hyporheic exchange on meandering river. *Geophys. Res. Lett.*, 33(18), 2006, DOI: <https://doi.org/10.1029/2006GL027630>.

64. Botticini-Busolin A. Non-fickian dispersion in open-channel flow over a porous bed. *Water Resour. Res.*, 53(8), 2016 DOI: <https://doi.org/10.1002/2016WR020348>.

65. I. D. Chandler, I. Guymer, J.M. Pearson, R. van Egmond Vertical variation of mixing within porous sediment beds below turbulent flows. *Water Resour. Res.*, 52(5). 2015, DOI: <https://doi.org/10.1002/2015WR018274>.

66. L.C. Brown, Barnwell T.O. The enhanced stream water quality models QUAL2E and QUAL2E-UNCAS: documentation and user manual. Environmental Resource Laboratory, Athens, 1987.

67. Barnwell T.O., Brown L.C., Whittemore R.C. Importance of field data in stream water quality modeling using QUAL2E-UNCAS. *Journal of Environmental Engineering-Asce*, 130(6), 2004.

68. Ambrose R.B., Wool T.A. WASP7 stream transport model theory and user guide. U.S. Environmental Protection Agency, 2009.

69. Souchere, V., King, D., Daroussin, J., Papy, f., Capillon, A. Effects of tillage on runoff directions : consequences on runoff contributing area within agricultural catchments. *Journal of Hydrology*. 1998.

70. Schmitt, T.J., Dosskey, M.G., Hoagland, K.D. Filter strip performance and processes for different vegetation, widths and contaminants. *Journal of Environmental Quality*. 1999.

71. Tarboton, D.G. A new method for the determination of flow directions and upslope areas in grid digital elevation models. *Water Resources Research*, 33(2), 1997.

72. Tomer, M.D., James, D.E., Isenhardt, T.M. Optimising the placement of riparian practices in a watershed using terrain analysis. *Journal of Soil and Water Conservation*, 58(4), 2003.

73. Boyd, P.M., Baker, J.L., Mickelson, S.K., Ahmed, S.I. Pesticide transport with surface runoff and subsurface drainage through a vegetative filter strip. *Transactions of the American Society of Agricultural Engineers*, 46(3), 2003.
74. Bren, L.J.. The geometry of a constant buffer-loading design method for humid watersheds. *Forest Ecology and Management*, 1998.
75. Brothers M., Eisenhauer D.E., Helmers M.J., Dosskey, M.G., Franti, T.G. Modelling vegetative buffer performance considering topographic data accuracy. *ASAE Annual International Meeting*, 2001.
76. Fischer, H. B. The mechanics of dispersion in natural streams. *J. Hydr. Div., ASCE*, 93(6), 1967.
77. Elizabeth W. Boyer, Richard B. Alexander, William J. Parton, Changsheng Li, Klaus Butterbach-Bahl, Simon D. Donner, R. Wayne Skaggs, Stephen J. Del Grosso Modeling denitrification in terrestrial and aquatic ecosystems at regional scales, 2006. DOI: [https://doi.org/10.1890/1051-0761\(2006\)016\[2123:MDITAA\]2.0.CO;2](https://doi.org/10.1890/1051-0761(2006)016[2123:MDITAA]2.0.CO;2).
78. Richard B. Alexander, Elizabeth W. Boyer, Richard A. Smith, Gregory E. Schwarz, Richard B. Moore The Role of Headwater Streams in Downstream Water Quality, 2007. DOI: <https://doi.org/10.1111/j.1752-1688.2007.00005.x>.
79. Harvey E. Jobson Predicting Travel Time and Dispersion in Rivers and Streams. *Journal of Hydraulic Engineering*, Volume 123, Issue 11. DOI: [https://doi.org/10.1061/\(ASCE\)0733-9429\(1997\)123:11\(971\)](https://doi.org/10.1061/(ASCE)0733-9429(1997)123:11(971)).
80. Elisabeta-Cristina Ani, Steve Wallis, Andrzej Kraslawski, Paul Serban Agachi Development, calibration and evaluation of two mathematical models for pollutant transport in a small river. *Environmental Modelling & Software*, Volume 24, Issue 10, October 2009, Pages 1139-1152. DOI: <https://doi.org/10.1016/j.envsoft.2009.03.008>.
81. Tomas Morales, Inigo Fdez. de Valderrama, Jesus A. Uriarte, Inaki Antigüedad, Martin Olazar Predicting travel times and transport characterization in karst conduits by analyzing tracer-breakthrough curves. *Journal of Hydrology*, Volume 334, Issues 1–2, 20 February 2007, Pages 183-198.
82. Біланюк В.І. Практикум із загальної гідрології. Режим доступу: [https://geoknigi.com/book\\_view.php?id=265](https://geoknigi.com/book_view.php?id=265).

83. Fischer, H.B. Methods for predicting dispersion coefficients in natural streams, with application to lower reaches of the Green and Duwamish Rivers, 1968, p 27.
84. Longley, P - Geographic Information Systems and Science 2nd – ed – John Wiley & Sons - 2005.
85. Horváth Cs., Réti Kinga-Olga, Bilaşco Şt., Roşian Gh. Spatial Distribution of the Average Runoff in the Iza and Vişeu Watersheds - Air and Water Components of the Environment Conference Proceedings, Edit – 2015.
86. Кучерук І. М., Горбачук І. Т., Луцик П. П. Загальний курс фізики : навч. посібник у 3-х т, Т. 2 : Електрика і магнетизм, Київ, Техніка, 2006.
87. Philibert J. One and a Half Centuries of Diffusion: Fick, Einstein, before and beyond. Diffusion Fundamentals, 2005.
88. Zhou L, Nyberg K, Rowat AC Understanding diffusion theory and Fick's law through food and cooking. Advances in Physiology Education. DOI: <https://doi.org/10.1152/advan.00133.2014>.
89. Myk.M. Gertsyuk, C. Horvath, Mod.M. Gertsyuk, G.V. Lysychenko Software for modeling and predicting pollution dispersion on the Tisza river and its tributaries in case of emergencies. Fourth International Conference: «Chemical and radiation safety: problems and solutions». Work and message (Kyiv, 17-20 May 2016), Kyiv, 2016. p. 11.
90. Герцюк М.М., Жебка В.В. Аналіз програмного забезпечення для прогнозування забруднення річок небезпечними речовинами. Зв'язок, Випуск 2, 2019, С. 45-48.
91. Cussler, E. L. Diffusion: Mass Transfer in Fluid Systems (2nd ed.). New York: Cambridge University Press, 1997.
92. James R. Welty, Charles E. Wicks, Robert E. Wilson, Gregory Rorrer. Fundamentals of Momentum, Heat, and Mass Transfer, Wiley, 2001.
93. P. Grathwohl Diffusion in natural porous media: Contaminant transport, sorption / desorption and dissolution kinetics. Kluwer Academic, 1998.
94. Holley E. R. Diffusion and Dispersion. Environmental Hydraulics, 1996, 111-151.

95. Дідковська М. В., Тимошенко Ю. О. Тестування: Основні визначення, аксіоми та принципи. Текст лекцій. Частина I, 2010, Київ, КПІ, 62 С.
96. Санітарні правила і норми охорони поверхневих вод від забруднення. СанПіН 4630-88. Режим доступу: <https://zakon.rada.gov.ua/laws/show/v4630400-88>.
97. Itkonen, Juha; Mäntylä, Mika V. Are test cases needed? Replicated comparison between exploratory and test-case-based software testing. *Empirical Software Engineering*, 19 (2), 2013, p. 303–342.
98. Dustin, Rashka, Paul. Automated Software Testing -Introduction, Management, and Performance. Addison-Wesley, 1999, p. 43-44.
99. Cem Kaner, James Bach, Bret Pettichord Lessons Learned in Software Testing. Wiley Computer Publishing, 2002, p. 95.
100. T. Dohi, T. Nakagawa Stochastic Reliability and Maintenance Modeling: Essays in Honor of Professor Shunji Osaki on his 70th Birthday. Springer London, 2013, 360 p.
101. Proceedings from the 5th International Conference on Software Testing and Validation (ICST). Software Competence Center Hagenberg. Test Design: Lessons Learned and Practical Implications. DOI: <https://doi.org/10.1109/IEEESTD.2008.4578383>.
102. Adam Kolawa, Dorota Huizinga Automated Defect Prevention: Best Practices in Software Management. Wiley-IEEE Computer Society Press, 2007, p. 74.
103. Josh Berkus Wrecking Your Database, Computer, Aug. 2009, Режим доступу: <https://www.toolbox.com/tech/data-management/blogs/wrecking-your-database-080509/>.
104. M. Lorenz, J.P. Rudolph, G. Hesse, M. Uflacker, H. Plattner. Object–Relational Mapping Revisited - A Quantitative Study on the Impact of Database Technology on O/R Mapping Strategies. Hawaii International Conference on System Sciences (HICSS).
105. Steven Feuerstein, Bill Pribyl Oracle PL/SQL Programming, September, 1997.



106. Localization: Easily translate your DotNetNuke® module. Режим доступу: <http://www.adeftwebserver.com/DotNetNukeHELP/Localization/>.

107. Introduction to ASP.NET Identity - ASP.NET 4.x | Microsoft Docs. Режим доступу: <https://docs.microsoft.com/en-us/aspnet/identity/overview/getting-started/introduction-to-aspnet-identity>.

108. ApexChart.js. Режим доступу: <https://apexcharts.com>.

109. Charts - Google Developer. Режим доступу: <https://developers.google.com/chart/?hl=uk>.

110. Google Maps Platform - Google Developers. Режим доступу: <https://developers.google.com/maps/documentation>.

111. Entity Framework Overview | Microsoft Docs. Режим доступу: <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/ef/overview>.

112. Вступ в NUnit. Режим доступу: <https://itvdm.com/ua/blog/article/entry-into-nunit>.

113. Що таке Selenium? | Україномовний туторіал по Selenium Web Driver – Habr. Режим доступу: <https://www.quality-assurance-group.com/shho-take-selenium/>.

114. GUI Testing Tutorial: User Interface (UI) TestCases with Examples. Режим доступу: <https://www.guru99.com/gui-testing.html>.

115. Usability Testing - Tutorials Point. Режим доступу: [https://www.tutorialspoint.com/software\\_testing\\_dictionary/usability\\_testing](https://www.tutorialspoint.com/software_testing_dictionary/usability_testing).

116. Smoke Testing - Smoke Testing Fundamentals, Режим доступу: <http://softwaretestingfundamentals.com/smoke-testing/>.

117. Види тестування та відмінності між ними - шпаргалка з тестування (Частина 4) - Quality Assurance Group. Режим доступу: <https://www.quality-assurance-group.com/vydy-testuvannya-ta-vidminnosti-mizh-nymy-shpargalka-z-testuvannya-chastyna-4>.

118. Техніка тестування. Режим доступу: [http://www.qaclubkiev.com/2015/03/blog-post\\_17.html](http://www.qaclubkiev.com/2015/03/blog-post_17.html).

119. Биченок М.М. Проблеми інформатизації в сфері екологічної безпеки // Стохастические задачи теории оптимизации и надежности, Київ, Інститут кібернетики ім. В.М. Глушкова НАН України, 1995.

120. Биченок М.М. Про багаторівневе узгоджене управління ресурсами захисту у надзвичайних ситуаціях. Екологія і ресурси. — К.: РНБОУ, 2002., №4.

121. Биченок М.М., Довгий С.О., Трофимчук О.М. Системні принципи інформатизації управління природно-техногенною безпекою. Екологія і ресурси. К.: РНБОУ, 2003.

122. Биченок М.М., Трофимчук О.М. Проблеми природно-техногенної безпеки в Україні. К.: РНБОУ, 2002, 153 с.

123. Волошкіна О.С., Красовский Г.Я., Трофимчук О.М. Картографічна модель дифузних джерел забруднення водних об'єктів з використанням ДЗЗ і ГІС. Екологія і ресурси, К.: РНБОУ, 2002. №4, С. 94 — 97.

124. Дорогунцов С.І., Ральчук О.М. Управління техногенно-екологічною безпекою у парадигмі сталого розвитку, К.: Наук. думка, 2001, 174 с.

125. Іщук О.О., Коржнев М.М., Кошляков О.Е. Просторовий аналіз і моделювання в ГІС. ВПЦ “Київський університет”, 2003, 200с.

126. Desmet, P.J.J. and G. Govers. A GIS procedure for automatically calculating the USLE LS factor on topographically complex landscape units. Journal of Soil and Water Conservation Service 51:427-433. 1996.

127. Foster, G.R. Modeling the erosion process. Chapter 8. In: Hydrologic Modeling of Small Watersheds. C.T. Haan, H.P. Johnson, D.L. Brakensiek, eds. American Society of Agricultural Engineers. St. Joseph, MI. 1982. pp. 297-382.

128. D. K. McCool, G. R. Foster, A. H. Ingersoll, R. C. McClellan, R. W. Rickman. Cover-Management Enhancements for RUSLE2 in the Pacific Northwest USA. Vol. II, In: Proceedings of the 12th Conference of the International Soil Conservation Organization, Beijing, China. 2002, p. 513-517.

129. Spaeth, Jr., K.E. F.B. Pierson, M.A. Weltz, and W.H. Blackburn. Evaluation of USLE and RUSLE estimated soil loss on rangeland. Journal of Range Management, 2003.

130. Barlow, P.M., Cunningham, W.L., Zhai, Tong, Gray, Mark, U.S. Geological Survey Groundwater Toolbox, a graphical and mapping interface for analysis of hydrologic data (version 1.0)—User guide for estimation of base flow, runoff, and groundwater recharge from streamflow data: U.S. Geological Survey Techniques and Methods, book 3, chap. B10, 27 p., September 7, 2017. DOI: <https://dx.doi.org/10.3133/tm3B10>.

131. Gain, W.S. Time-of-travel of solutes in the Trinity River from Dallas to Trinidad, Texas, May and August 1987: U.S. Geological Survey Open-File Report 89-0614, 1 map page, 1990.

132. County, Iowa: U.S. Geological Survey Water-Resources Investigations Report 91-4145, 16 p. Laenen, Antonius, and Woo, W.H., 1994, Stream velocities and reaeration coefficients for the South Umpqua River between Tiller and Roseburg: U.S. Geological Survey Water-Resources Investigations Report 92-4191, 26 p.

133. K.E. White, and Kratzer, T.W., Traveltime in the Delaware River, Hancock, New York, to the Delaware Water Gap by use of a conservative dye tracer: U.S. Geological Survey Water-Resources Investigations Report 93-4203, 1993, 54 p.

134. Tentu Nageswara Rao Validation of Analytical Methods, Calibration and Validation of Analytical Methods - A Sampling of Current Approaches, 2018. DOI: <https://doi.org/10.5772/intechopen.72087>.

135. INRA Quality Policy and Quality Guidelines for the Research and Experimental Units. 2013.

136. Introduction to Analytical Method Validation. Режим доступа: <https://lab-training.com/introduction-to-analytical-method-validation/>.

137. Bottacin-Busolin, A. Non-Fickian dispersion in open-channel flow over a porous bed. Water Resources Research, 53, 7426–7456. DOI: <https://doi.org/10.1002/2016WR020348>. 2017.

138. Marion, A., & Zaramella, M. Diffusive behavior of bedform-induced hyporheic exchange in rivers. Journal of Environmental Engineering, 131(9), 1260–1266, 2005. DOI: [https://doi.org/10.1061/\(ASCE\)0733-9372\(2005\)131:9\(1260\)](https://doi.org/10.1061/(ASCE)0733-9372(2005)131:9(1260)).

139. Nordin, C. F., & Troutman, B. M. Longitudinal dispersion in rivers: The persistence of skewness in observed data. *Water Resources Research*, 16(1), 123–128, 1980. DOI: <https://doi.org/10.1029/WR016i001p00123>.
140. Ward, A. S., Payn, R. A., Gooseff, M. N., McGlynn, B. L., Bencala, K. E., Kelleher, C. A., Wondzell, S. M., & Wagener, T. Variations in surface water-ground water interactions along a headwater mountain stream: Comparisons between transient storage and water balance analyses. *Water Resources Research*, 49, 3359–3374. 2013. DOI: <https://doi.org/10.1002/wrcr.20148>.
141. Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron Softmax Units for Multinoulli Output Distributions. *Deep Learning*. MIT Press, 2016, pp. 180–184.
142. Liu, Danqing "A Practical Guide to ReLU". Medium. 2017. Режим доступу: <https://medium.com/@danqing/a-practical-guide-to-relu-b83ca804f1f7>.
143. TensorFlow Core | Machine Learning for Beginners and Experts. Режим доступу: <https://www.tensorflow.org/overview>.
144. Методи та моделі розроблення комп'ютерних систем і мереж : монографія / В. С.Пономаренко, С. В. Мінухін, С. В. Кавун, та ін. ; заг. ред. докт. кон. наук, професора Пономаренка В. С. – Х. : Вид. ХНЕУ, 2008. – 316 с.
145. Світличний О. О. Основи геоінформатики : навч. посібн./ О. О. Світличний, С. В. Злотницький ; за заг. ред. О. О. Світличного, Суми, ВТД "Університетська книга", 2006, 295 с.
146. Toy, T. J., Foster, G. R., and Renard, K. G. *Soil Erosion: Processes, Prediction, Measurement, and Control*. New York: John Wiley and Sons, 2002.
147. B.M. Ayyub, R.H. MCCUEN *Probability, statistics, and reliability for engineers and scientists*. BocaRaton, Chapman and Hill, CRC Press, 2002.
148. B.P. Carlin, T.A. Louis *Bayes and empirical Bayes methods for data analysis*, 2nd Edition. New York, Chapman and Hall, CRC, 2000.
149. C.A. Cohn, W.L. Lane, J.R. Stedinger Confidence intervals for EMA flood quantile estimates. *Water Resources Research*, Vol. 37, No. 6, pp. 1695–1706, 2001.

150. V.W. Griggs, J.R. Stedinger, T.A. Cohn, LP3 quantile estimators with regional skew information and low outlier adjustments. *Water Resources Research*, Vol. 40, forthcoming, 2004.

151. J.U. Chowdhury, J.R. STEDINGER, Confidence intervals for design floods with estimated skew coefficient. *Journal of Hydraulic Engineering*, Vol. 117, No. 7, 1991, pp. 811.

## Додаток А. Набір кінцевих точок для REST API

Таблиця

Посилання кінцевої точки	Призначення	Вхідні дані	Вихідні дані
api/rivername/	Повертає назву річки, тобто ім'я REST API.	-	<i>string</i>
api/riverpoints/	Повертає список точок за необхідними характеристиками.	-	[ { <i>id</i> , <i>nameOfPoint</i> , <i>coordinatesX</i> , <i>coordinatesY</i> , <i>catchmentArea</i> , <i>averageRunoff</i> , <i>distance</i> , <i>previousPoints</i> }, ... ]
api/pointsCurrentRunoff/	Повертає значення поточних витрат води	-	[ { <i>id</i> , <i>value</i> }, ... ]
api/setPointCurrentRunoff	Оновлює або додає інформацію про поточні витрати	{ <i>pointId</i> , <i>q</i> }	-
api/pointCoefficients/	Повертає значення коефіцієнтів за конкретними точками	-	[ { <i>pointId</i> , <i>winterCoefficient</i> , <i>springCoefficient</i> , <i>summerCoefficient</i> , <i>autumnCoefficient</i> }, ... ]

## Продовження таблиці

Посилання кінцевої точки	Призначення	Вхідні дані	Вихідні дані
api/substances/	Повертає список речовин та їхні гранично допустимі значення	-	[ { <i>id</i> , <i>substanceNameEng</i> , <i>substanceNameUkr</i> , <i>concentration</i> }, ... ]
api/dangerousObjects/	Повертає список небезпечних об'єктів	-	[ { <i>id</i> , <i>pointName</i> , <i>coordinates_X</i> , <i>coordinates_Y</i> } ] ]
api/addSituation/	Додає нову ситуацію	{ <i>situationName</i> , <i>situationTime</i> , <i>pollution</i> , <i>points</i> : [ { <i>pointId</i> , <i>A</i> , <i>Arel</i> , <i>Rx</i> , <i>Vp</i> , <i>Vm</i> , <i>Tr</i> , <i>Cup</i> , <i>Cp</i> }, ... ] }	{ <i>situationId</i> }

## Продовження таблиці

Посилання кінцевої точки	Призначення	Вхідні дані	Вихідні дані
api/pointSources/	Повертає зв'язку між точкою та потенційно небезпечним об'єктом	-	[ { pointId, dangerousObjectId }, ... ]
api/dangerousObjectsSubstances/	Повертає зв'язку між потенційно небезпечним об'єктом та речовиною	-	[ { dangerousObjectId, substanceId }, ... ]
api/dangerousObjectFrequencies/	Повертає інформацію про кількість забруднень за конкретним потенційно небезпечним об'єктом	-	[ { id, pollutionCount } ]



## Додаток Б. Позитивні тестові ситуації для REST API сервісу

Таблиця

№	Посилання	Опис	Вхідні дані	Результат
1	api/rivername/	Отримання назви серверу	1. Назва серверу в базі даних.	Додана назва рівна повернутій
2	api/riverpoints/	Отримання списку точок	1. Список точок в базі даних.	Доданий список точок рівний повернутому
3	api/pointsCurrentRunoff/	Отримання списку точок за характеристикою поточного забруднення	1. Список точок з базі даних.	Характеристика середньої витрати доданого списку точок рівна повернутій
4	api/setPointCurrentRunoff/	Присвоєння поточного забруднення для конкретної точки	1. Список точок з базі даних.	При запиті api/pointsCurrentRunoff/ характеристика визначеної точки рівна присвоєній
5	api/pointCoefficients/	Отримання списку коефіцієнтів за точками	1. Список точок з базі даних. 2. Список коефіцієнтів	Доданий коефіцієнт за точкою рівний повернутому
6	api/substances/	Отримання списку речовин	1. Список речовин в базі даних	Доданий список речовин рівний повернутому
7	api/dangerousObjects/	Отримання списку небезпечних об'єктів	1. Список небезпечних об'єктів в базі даних	Доданий список об'єктів рівний повернутому
8	api/addSituation/	Додавання ситуації	-	Ситуація, що описана в запиті має бути в базі даних

## Продовження таблиці

№	Посилання	Опис	Вхідні дані	Результат
9	api/pointSources/	Отримання списку зв'язки об'єктів до точок	1. Список небезпечних об'єктів в базі даних. 2. Список точок в базі даних. 3. Список зв'язок	Додані зв'язки рівні отриманим
10	api/dangerousObjectsSubstances/	Отримання списку зв'язки об'єктів до речовин	1. Список небезпечних об'єктів в базі даних. 2. Список речовин в базі даних. 3. Список зв'язок	Додані зв'язки рівні отриманим
11	api/dangerousObjectFrequencies/	Отримання списку сум забруднень	1. Декілька надзвичайних ситуацій в базі даних	Значення доданих сум співпадає і отриманим

## Додаток В. Тестові ситуації головного вікна

Таблиця

№	Опис	Вхідні дані	Результат
1	Запуск додатку	-	Відображається головне вікно
2	Виклик майстру проведення ручних обчислень	-	Початкове вікно зазначеного майстра відображається
3	Виклик майстру проведення автоматичних обчислень	-	Початкове вікно зазначеного майстра відображається
4	Експорт в JPG	1. Існуюче обчислення	Відображений графік зберігся у вигляді JPG файлу
5	Експорт у XLS	1. Існуюче обчислення	Відображений графік зберігся у вигляді XLS файлу
6	Перевірка ручного режиму вікна	1. Існуюче ручне обчислення	Головне вікно знаходиться у стані ручного обчислення, тобто кнопка виклику дашборду недоступна, графік має дві колонки
7	Перевірка автоматичного режиму вікна	1. Існуюче автоматичне обчислення	Головне вікно знаходиться у стані ручного обчислення, тобто кнопка виклику дашборду доступна, графік має п'ять колонок
8	Перевірка функціоналу формування даних	1. Введені дані в таблицю в режимі ручного обчислення	Вікно сформоване до ручного обчислення

## Продовження таблиці

№	Опис	Вхідні дані	Результат
9	Перевірка функціоналу очищення даних	Існуюче обчислення	Вікно скинуто до початкового стану
10	Проведення ручного обчислення	-	Вікно відображає результати ручного обчислення
11	Проведення автоматичного обчислення	-	Вікно відображає результати автоматичного обчислення
12	Виклик налаштувань серверу	-	Вікно налаштувань викликане
13	Вихід	-	Додаток завершив роботу

## Додаток Г. Тестові ситуації дашборду

Таблиця

№	Опис	Вхідні дані	Результат
1	Дашборд запускається коректно	Існуюче обчислення в автоматичному режимі	Вікно запускається без збоїв
2	Таблиця відображає числові результати	Існуюче обчислення в автоматичному режимі. Масив результатів	Еталонні числові значення співпадають з очікуваними
3	Мапа відображає точки правильно	Існуюче обчислення в автоматичному режимі	Еталонним список точок співпадає з очікуваним
4	Зміна вибраної точки	Існуюче обчислення в автоматичному режимі. Вибір точки	Елемент, що відображає дані за точкою показує значення вибраної точки
5	Графік співпадає з числовими значенням	Існуюче обчислення в автоматичному режимі. Еталонний графік	Еталонний графік співпадає з очікуваним
6	Список можливих забруднювачів відображається правильно	Існуюче обчислення в автоматичному режимі. Еталонний список результатів визначення забруднювачів	Еталонний список співпадає з очікуваним
7	Збереження даних у вигляді KML файлу	Існуюче обчислення в автоматичному режимі	KML файл з результатами створено
8	Збереження даних графіку у вигляді JPG файлу	Існуюче обчислення в автоматичному режимі	JPG файл з результатами створено
9	Перемикач, що відображає коректовані значення включає масив коректованих значень	Існуюче обчислення в автоматичному режимі. Еталонний масив коректованих результатів	Еталонний масив співпадає з очікуваним

## Додаток Д. Акти впровадження

### ЗАТВЕРДЖУЮ

Директор Інституту  
телекомунікацій і глобального  
інформаційного простору НАН

України,  
член-кор. НАНУ, д.тех.н., проф.



Трофимчук О.М.

« 15 » серпня 2023 р.

### А К Т

про впровадження результатів наукових досліджень

Герцюка М.М.

в практичну діяльність Інституту телекомунікацій і глобального  
інформаційного простору НАН України

Комісія у складі:

Голова комісії, головний науковий співробітник відділу досліджень  
навколишнього середовища д.тех.н., ст. наук.сп.,  
Яковлев Є.О.

члени комісії: к.тех.н., старший дослідник Охарев В.О.  
к.тех.п., Зозуля А.М.

Результати наукових досліджень на тему: «Інформаційна технологія прогнозування забруднення річок в умовах надзвичайних ситуацій на основі методів регресії та евристичних алгоритмів» за спеціальністю 123 - Комп'ютерна інженерія.

Комісія встановила, що результати наукових досліджень Герцюка Миколи Модестовича є актуальними для впровадження в науково-практичну діяльність Інституту телекомунікацій і глобального інформаційного простору НАН України.

В дисертаційному дослідженні розроблено інформаційну технологію прогнозування негативних наслідків забруднення річок потенційно небезпечними для довкілля і населення токсичними викидами. Мінімізація наслідків забруднення потребує оцінки стану русла річки, в основі якої лежать система моніторингу та методи прогнозування наслідків розповсюдження викидів. З цією метою розроблено інформаційну систему RESit, як інформаційну технологію прогнозування забруднення річок в умовах надзвичайних ситуацій. Дана інформаційна технологія, також, включає методи:

- коригування результатів прогнозування концентрації забруднювача, що дозволяє підвищити точність характеристики піку концентрації в конкретній точці моніторингу, тобто точність результатів;
- прогнозування рівня забруднення між конкретними точками за допомогою методу інтерполяції та алгоритму рекурсії, що дозволяє отримати згладження прогнозованих результатів, отримати результат характеристики піку концентрації між точками моніторингу та збільшити масив результатів для ширшого розуміння стану русла в умовах надзвичайних ситуацій;
- визначення джерела забруднення на основі алгоритму фільтрації та сортування даних, що працює на основі бази фактів.

Наукові та практичні результати дослідження можуть знайти подальше практичне застосування в процесі розробки прогнозування рівня забруднення при симуляції надзвичайної ситуації та інших інформаційних систем оцінки впливу забруднюючих факторів на стан довкілля з метою підвищення рівню контролю, регулювання, моніторингу та обґрунтування найбільш оптимальних рішень з усунення або зниження рівнів ризиків забруднення русла річок, що виникли як наслідок надзвичайної ситуації.

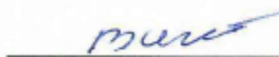
Даний акт не є підставою для фінансових взаєморозрахунків.

Голова комісії



Є.О.Яковлев

Члени комісії



В.О.Охарев



А.М.Зозуля

**ЗАТВЕРДЖУЮ**

Проректор з навчально-виховної та наукової роботи Державного університету інформаційно-комунікаційних технологій



д.т.н., проф.

Беркман Л.Н.

*Вересень*

2023 р.

**А К Т**

про впровадження результатів наукових досліджень

**Герцюка М.М.**

в науково-практичну та педагогічну діяльність Державного університету інформаційно-комунікаційних технологій

Комісія у складі:

Голова комісії д.т.н., проф. Бондарчук А.П.

члени комісії: д.т.н., проф. Вишнівський В.В.

к.т.н., доц. Золотухіна О.А.

результати наукових досліджень на тему: «Інформаційна технологія прогнозування забруднення річок в умовах надзвичайних ситуацій на основі методів регресії та евристичних алгоритмів» за спеціальністю 123 - Комп'ютерна інженерія.

Комісія встановила, що результати наукових досліджень Герцюка Миколи Модестовича є актуальними для впровадження в науково-практичну та педагогічну діяльність Державного університету інформаційно-комунікаційних технологій.

З метою оцінки наслідків забруднення річкових водойм при настанні надзвичайної ситуації у випадку викиду небезпечних речовин розроблено інформаційну технологію прогнозування негативних наслідків забруднення річок потенційно небезпечними речовинами, що включає:

- інформаційну систему RESit, що дозволяє провести необхідні обчислення ліквідації аварії, що веде до порушення структури і функціонування природних систем, призводить до негативних соціальних, економічних та інших наслідків;
- метод коригування результатів прогнозування концентрації забруднювача, що дозволяє підвищити точність результатів;



- метод прогнозування рівня забруднення між конкретними точками, що дозволяє отримати згладження прогнозованих результатів, тобто отримати результат характеристики піку концентрації між точками моніторингу та збільшити масив результатів для ширшого розуміння стану русла в умовах надзвичайних ситуацій;

- метод визначення джерела забруднення на основі алгоритму фільтрації та сортування даних, що працює на основі бази фактів.

Структура інформаційної системи RESit включає сервер – сервіс, метою якого є забезпечення цілісності даних при взаємодії з іншими складовими частинами системи, адміністративний веб-додаток, що дозволяє проводити операції маніпуляції даними, утиліту навчання, що проводить навчання нейронної мережі за існуючою інформацією бази даних та настільне програмне забезпечення для проведення обчислень користувачем. Систему проєктовано, виходячи з принципу «багато до багатьох», тобто кожен з ключових сервісів має можливість взаємодії з багатьма сервісами.

Наукові та практичні результати дослідження можуть знайти подальше практичне застосування в процесі розробки:

- комп'ютеризованої системи оцінки впливу забруднюючих факторів на стан довкілля з метою підвищення рівню контролю, регулювання, моніторингу та обґрунтування найбільш оптимальних рішень з усунення або зниження рівнів ризиків забруднення русла річок, як наслідки настання надзвичайної ситуації;

- наукове застосування в процесі використання методів у складі інших методів, або удосконалення цих методів;

- педагогічне застосування, як відображення принципів, за якими побудована інформаційна система в навчальному принципі.

Даний акт не є підставою для фінансових взаєморозрахунків.

Голова комісії

Члени комісії



А.П. Бондарчук

В.В. Вишнівський

О.А. Золотухіна

**ЗАТВЕРДЖУЮ**

Декан факультету інформаційних  
технологій та математики Київського  
університету імені Бориса Грінченка  
кандидат фізико-математичних наук,  
старший науковий співробітник



**Оксана ЛИТВИН**

2023 р.

**А К Т**

про впровадження результатів наукових досліджень

**Герцюка Миколи Модестовича**

в науково-практичну та педагогічну діяльність

кафедри інформаційної та кібернетичної безпеки імені професора Володимира  
Бурячка Факультету інформаційних технологій та математики  
Київського університету імені Бориса Грінченка

Комісія у складі:

Голова комісії: завідувач кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка, кандидат технічних наук, доцент **Павло СКЛАДАННИЙ**;

члени комісії: професор кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка, доктор технічних наук, професор **Геннадій ГУЛАК**;  
професор кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка, доктор технічних наук, професор **Наталія КОРШУН**.

Комісія встановила, що результати наукових досліджень Герцюка Миколи Модестовича на тему: «Інформаційна технологія прогнозування забруднення річок в умовах надзвичайних ситуацій на основі методів регресії та евристичних алгоритмів» за спеціальністю 123 - Комп'ютерна інженерія, є актуальними для впровадження педагогічну діяльність кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Факультету інформаційних технологій та математики Київського університету імені Бориса Грінченка.

З метою оцінки наслідків забруднення річкових водойм при настанні надзвичайної ситуації у випадку викиду небезпечних речовин розроблено інформаційну технологію прогнозування негативних наслідків забруднення річок потенційно небезпечними речовинами, що включає:

- інформаційну систему RESit, що дозволяє провести необхідні обчислення необхідні для ліквідації аварії;

- метод коригування результатів прогнозування концентрації забруднювача, що дозволяє підвищити точність результатів прогнозування наслідків забруднення;

- метод прогнозування рівня забруднення між конкретними точками, що дозволяє отримати результат прогнозування наслідків забруднення між точками моніторингу та збільшити масив результатів для ширшого розуміння стану русла в умовах надзвичайних ситуацій;

- метод визначення джерела забруднення на основі алгоритму фільтрації та сортування даних, що працює на основі бази фактів.

Наукові та практичні результати дослідження можуть знайти подальше педагогічне застосування, як відображення принципів, за якими побудована інформаційна система в навчальному процесі.

Даний акт не є підставою для фінансових взаєморозрахунків.

Голова комісії  
Члени комісії:



Павло СКЛАДАННИЙ  
Геннадій ГУЛАК  
Наталія КОРШУН



**ЗАТВЕРДЖУЮ**

Засновник ІТ компанії Cavabang

Малінов Владислав Андрійович

« \_\_\_\_ » \_\_\_\_\_ 2024 р.

**А К Т**

про впровадження результатів наукових досліджень

**Герцюка М.М.**

в науково-практичну діяльність ІТ компанії Cavabang.

Результати, отримані в науковому дослідженні Герцюка Миколи Модестовича на тему: «Інформаційна технологія прогнозування забруднення річок в умовах надзвичайних ситуацій на основі методів регресії та евристичних алгоритмів» за спеціальністю 123 – Комп'ютерна інженерія, є актуальними для впровадження в науково-практичну діяльність ІТ і маркетинг компанії Cavabang розроблених та удосконалених методів, що допомагають проводити прогнозування наслідків забруднення річок та архітектури програмного інформаційної системи RESit.

Наукова новизна дослідження полягає в удосконаленні існуючої інформаційної технології за рахунок:

- розробленого методу коригування результатів прогнозування концентрації забруднювача, що дозволяє підвищити точність результатів прогнозування наслідків забруднення;
- розробленого методу прогнозування рівня забруднення між конкретними точками, що дозволяє отримати результат прогнозування наслідків забруднення між точками моніторингу та збільшити масив результатів для ширшого розуміння стану русла в умовах надзвичайних ситуацій;
- розробленого методу визначення джерела забруднення на основі алгоритму фільтрації та сортування даних, що працює на основі бази фактів;
- розробленої інформаційної системи RESit на основі програмного застосунку RESit, що дозволяє проводити необхідні обчислення для ліквідації аварії.

Система RESit дозволяє виводити результат у вигляді графіків та мап, зберігати його у вигляді KML файлів, що дають можливість інтеграції системи з різними ГІС системами та з більшістю автоматизованих картографічних рішень.

Розробки та удосконалення інформаційної системи RESit у поєднанні з методами прогнозування визначають подальші можливості щодо прогнозування та масштабування розвитку наслідків надзвичайних ситуацій у водоймах.

Комісія встановила, що результати наукових досліджень дають можливість створити більш простий та інтерактивний інтерфейс користувача з використанням інформаційної системи, що полегшить процес прогнозування надзвичайної ситуації.

Наукові та практичні результати дослідження можуть знайти подальше застосування в науково-практичну діяльність ІТ і маркетинг компанії Cavabang.

Даний акт не є підставою для фінансових взаєморозрахунків.

Засновник/СЕО компанії


Малінов В.А.



ЗАТВЕРДЖУЮ

Заступник директора з питань  
персоналу

ТОВ "Хуавей Україна"

 **В.Б. Білавка**  
(підпис, ініціали, прізвище)

«\_\_» \_\_\_\_\_ 2023 р.

### АКТ

про реалізацію результатів дисертаційної роботи

Герцюка Миколи Модестовича

на тему: «Інформаційна технологія прогнозування забруднення річок в умовах надзвичайних ситуацій на основі методів регресії та евристичних алгоритмів»

Результати наукових досліджень Герцюка Миколи Модестовича, а саме метод коригування результатів прогнозування концентрації забруднювача, метод прогнозування рівня забруднення між конкретними точками за допомогою методу інтерполяції та алгоритму рекурсії, метод визначення джерела забруднення на основі алгоритму фільтрації та сортування даних, що працює на основі бази фактів, використано в науково-дослідних і дослідно-конструкторських роботах, що ведуться в ТОВ "Хуавей Україна".

Застосування розроблених Герцюком М.М. методів дає можливість збільшити ефективність прогнозування наслідків забруднення річок в умовах надзвичайних ситуацій. Практична ефективність методів може бути показана після проведення обчислень на основі реальної бази даних.

Заступник директора з питань  
персоналу ТОВ "Хуавей Україна"



**В.Б. Білавка**



ДЕРЖАВНА СЛУЖБА УКРАЇНИ З НАДЗВИЧАЙНИХ СИТУАЦІЙ

**ЗАКАРПАТСЬКИЙ ОБЛАСНИЙ ЦЕНТР З ГІДРОМЕТЕОРОЛОГІЇ  
(ЗАКАРПАТСЬКИЙ ЦГМ)**

88018, Ужгород,  
Слов'янська набережна, 5

Тел/Факс: (0312) 61-27-66,  
61-27-50, 61-71-23

e-mail:  
apgdzuzhgorod@meteo.gov.ua

55-01/1611 № 08.12.18  
На вх. № \_\_\_\_\_ від \_\_\_\_\_

Інститут ґеохімії навколишнього  
середовища НАН України

Цим листом підтверджуємо, що в рамках виконання проекту SPS 984440: «Модель для прогнозування та запобігання можливих катастрофічних наслідків токсичного забруднення у басейні р. Тиса» за програмою НАТО «Наука заради миру», розроблена співробітниками інституту Герцюком Миколою Модестовичем і Герцюком Модестом Миколайовичем комп'ютерна програма прогнозу, при виникненні надзвичайних ситуацій, стану забруднення річки Тиса та її приток, «RESit», була впроваджена для практичного використання у роботі Закарпатського обласного центру з гідрометеорології.

Ця програма дає можливість, у випадку надзвичайних ситуацій, пов'язаних з потраплянням токсичних речовин в русло Тиси та її приток, проводити розрахунки концентрацій токсичних речовин та визначати час добігання цих речовин до вибраних пунктів по течії річок, а також давати прогноз небезпечності води для населення.

Начальник

Манівчук В.М.





**ДЕРЖАВНА ЕКОЛОГІЧНА ІНСПЕКЦІЯ УКРАЇНИ  
ДЕРЖАВНА ЕКОЛОГІЧНА ІНСПЕКЦІЯ У ЗАКАРПАТСЬКІЙ ОБЛАСТІ**

вул. Швабська, 14, м. Ужгород, 88018, тел/факс 61-43-30, тел 61-35-65, E-mail: [zak@deji.gov.ua](mailto:zak@deji.gov.ua)

05.12.2014 № 2644/14

На № \_\_\_\_\_ від \_\_\_\_\_.

**Інститут геохімії навколишнього  
середовища НАН України**

Цим листом підтверджуємо, що розроблена в рамках виконання проекту SPS 984440: «Модель для прогнозування та запобігання можливих катастрофічних наслідків токсичного забруднення у басейні р. Тиса» за програмою НАТО «Наука заради миру» комп'ютерна програма прогнозу, при виникненні надзвичайних ситуацій «RESit», встановлена співробітниками Інституту Герцюком Модестом Миколайовичем та Герцюком Миколою Модестовичем, була впроваджена для практичного використання у роботі Державної екологічної інспекції у Закарпатській області.

Ця програма призначена для проведення розрахунків концентрацій токсичних речовин та визначення часу добігання цих речовин до вибраних пунктів на р. Тиса та її притоках, у випадку виникнення надзвичайних ситуацій, пов'язаних з потраплянням токсичних речовин в поверхневі води.

**Начальник**

**I. Мартин**

Скоблей  
613565