

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет обліку та фінансів  
Кафедра комп'ютерних технологій  
і моделювання систем

Кваліфікаційна робота  
на правах рукопису

Резніченко Микола Юрійович

УДК 044.4:550.34

## КВАЛІФІКАЦІЙНА РОБОТА

«Інформаційна система моніторингу оцінювання сейсмічної обстановки  
регіону»

---

Кн-19-1, Комп'ютерні науки

---

Подається на здобуття освітнього ступеня бакалавр

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

Резніченко М.Ю.

Керівник роботи  
Веретюк С.М.

Старший викладач,  
кандидат технічних наук

Житомир – 2023

## Висновок кафедри

за результатами попереднього захисту: \_\_\_\_\_

Протокол засідання кафедри

№ \_\_\_ від «\_\_\_» \_\_\_\_\_ 20\_\_\_ р.

Завідувач кафедри

Доктор економічних наук,

професор

Николюк Ольга Миколаївна

«\_\_\_» \_\_\_\_\_ 2023р.

## Результати захисту кваліфікаційної роботи

Здобувач вищої освіти Резніченко Микола Юрійович захистив кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою \_\_\_\_\_

за шкалою ECTS \_\_\_\_\_

за національною шкалою \_\_\_\_\_

Секретар ЕК

\_\_\_\_\_ (науковий ступінь, вчене звання)

\_\_\_\_\_ (підпис) (прізвище, ім'я, по батькові)

## АНОТАЦІЯ

Резніченко М.Ю. Інформаційна система моніторингу оцінювання сейсмічної обстановки регіону. – Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття освітнього ступеня бакалавр за спеціальністю 122 – комп'ютерні науки. – Поліський національний університет, Житомир, 2021.

*У світі немає жодної людини, яка б не чула про землетруси – природні явища, які непередбачувано втручаються у життя людей по всьому світу. Землетруси виникають внаслідок раптового вивільнення енергії у земній літосфері та генерації сейсмічних хвиль. Інтенсивність землетрусів може варіюватись, від майже непомітних поштовхів до потужних та руйнівних, які можуть спричинити значні пошкодження та зруйнувати міста. Сейсмічна активність визначається частотою, типами та силою землетрусів, які відбуваються протягом певного періоду часу. Для захисту від цих небезпечних явищ необхідно розробляти методи їх прогнозування. Точність такого прогнозування залежить від наявності системи моніторингу та оцінювання сейсмічної обстановки у регіоні. Об'єктом дослідження є підсистема моніторингу стану сейсмічної обстановки регіону, а предметом дослідження – теоретичні, методичні та практичні аспекти проектування інформаційної підсистеми моніторингу сейсмічної активності в Україні. Для проведення дослідження використовувалися теоретичні та методичні роботи вітчизняних та зарубіжних науковців, а також такі методи дослідження, як аналіз і синтез, порівняльний аналіз, абстрактно-логічний та інші.*

Ключові слова: землетруси, тремтіння, сейсмічна активність, моніторинг, прогнозування, сейсмічна обстановка, підсистема, інформаційна підсистема, проектування.

## SUMMARY

*Nature often introduces unexpected changes into the lives of people all over the world, and there is no person who has never heard of the very important word - earthquake. Earthquakes (also known as tremors or quakes) are vibrations of the Earth's surface caused by the sudden release of energy in the Earth's lithosphere and the generation of seismic waves. Earthquakes can vary in intensity, ranging from imperceptible tremors to powerful events that can lift objects and people into the air, damage critical infrastructure, or destroy entire cities. Seismic activity refers to the frequency, type, and strength of earthquakes occurring within a certain period of time. Therefore, it is necessary to have the ability to protect against earthquakes through methods of prediction. Such prediction becomes more accurate when there is a monitoring system for assessing the seismic situation in a region. The object of research is the subsystem for monitoring the state of seismic conditions in a region, while the subject of research includes theoretical, methodological, and practical aspects of designing an information subsystem for monitoring seismic activity in Ukraine. The research is based on the works of domestic and foreign scientists, and the methods used include analysis and synthesis, comparative analysis, abstract-logical analysis, and others.*

*Keywords: earthquakes, tremors, seismic activity, monitoring, prediction, seismic situation, subsystem, information subsystem, design.*

## ЗМІСТ

<b>ВСТУП</b> .....	7
<b>Розділ 1 ТЕОРЕТИЧНИЙ АНАЛІЗ ОСОБЛИВОСТЕЙ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ</b> .....	8
<b>1.1 Аналіз інформаційних потреб і визначення предметної області дослідження.</b> .....	8
<b>1.2 Моделювання бізнес-процесів предметної області.</b> .....	9
<b>Висновки до першого розділу</b> .....	10
<b>Розділ 2. Використання рекурентних нейронних мереж для прогнозування сейсмічної ситуації в регіоні.</b> .....	Ошибка! Закладка не определена.
<b>Розділ 2. Проектування інформаційної системи моніторингу регіональної сейсмічної активності</b> .....	10
<b>2.1 Моделювання інформаційної системи моніторингу регіональної сейсмічної активності.</b> .....	14
<b>2.2 Проектування структури бази даних інформаційної системи</b> .....	16
<b>Висновок до другого розділу</b> .....	18
<b>Розділ 3. Реалізація інформаційної системи моніторингу регіональної сейсмічної активності</b> .....	
<b>3.1 Процес роботи алгоритму визначення землетрусу</b> .....	22
<b>3.2 Обґрунтування вибору бібліотеки машинного навчання ML .NET</b>	17
<b>3.3 Реалізація модулів системи</b> .....	23
<b>3.4 Деталізація розроблених алгоритмів</b> .....	25
<b>3.5 Проведення експериментів з метою перевірки ефективності розробленої системи</b> .....	27
<b>Висновок до третього розділу</b> .....	28
<b>Висновки та пропозиції</b> .....	29
<b>Перелік літературних джерел</b> .....	32

### **Скорочення та умовні позначки**

МБ – Мегабайт.

ІК – Інтерфейс користувача.

ОС – Операційна система.

ПК – Персональний комп'ютер.

СМ – Система моніторингу

ССА – Стан сейсмічної активності.

## ВСТУП

Землетрус в Туреччині в 2023р в котрий раз актуалізував питання ефективного моніторингу за сейсмічною обстановкою. Ефективний контроль за сейсмічною ситуацією допоможе попередити можливі наслідки землетрусу, зменшити його вплив та забезпечити швидку та ефективну допомогу потерпілим. Для цього можуть використовуватися різноманітні інструменти та технології, зокрема сейсмічні датчики, рекурентні нейронні мережі для прогнозування сейсмічної активності, системи збору та обробки даних тощо.

Належне управління ризиками та контроль за сейсмічною ситуацією є надзвичайно важливим для забезпечення безпеки населення та сталого розвитку країни.

Як ми бачимо природа часто вносить свої неочікувані корективи у життя різних людей всюди по світу, у світі немає жодної людини, яка ніколи не чула такого дуже важливого слова, як землетрус.

Землетруси (також відомі як тремтіння, трясіння або землетруси) - це поштовхи земної поверхні, спричинені раптовим вивільненням енергії в земній літосфері та генерацією сейсмічних хвиль. Землетруси мають різну інтенсивність і можуть бути настільки слабкими, що їх не відчувається, або настільки сильними, що можуть підняти в повітря предмети і людей, пошкодити критично важливу інфраструктуру або зруйнувати цілі міста. Сейсмічна активність - це частота, тип і сила землетрусів, що відбуваються протягом певного періоду часу.

Саме тому необхідно мати можливість захисту від тих самих землетрусів методами їх прогнозування. Таке прогнозування більш точне коли ми маємо систему моніторингу оцінювання сейсмічної обстановки регіону.

**Об'єкт дослідження** – підсистема моніторингу стану сейсмічної обстановки регіону.

**Предмет дослідження** – теоретичні, методичні, практичні питання проектування інформаційної підсистеми моніторингу сейсмічної активності України.

**Методи дослідження.** Теоретичною і методичною основою дослідження стали праці вітчизняних і зарубіжних науковців. У роботі використано такі методи дослідження: аналіз і синтез, порівняльний аналіз, абстрактно-логічний та інший.

**Структура дипломної роботи.** Дипломна робота складається із чотирьох розділів, висновків, список використаної літератури.

## **Розділ 1 ТЕОРЕТИЧНИЙ АНАЛІЗ ОСОБЛИВОСТЕЙ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ**

### **1.1 Аналіз інформаційних потреб і визначення предметної області дослідження.**

Спершу ніж розроблювати підсистему моніторингу потрібно детальніше дослідити дану предметну область та її функції для ліпшого розуміння того, що за продукт у нас має вийти, варто діяти крок за кроком для економії часу та кращого усвідомлення.

#### **Відомості про термін “сейсмічної активності”.**

Сейсмічна активність у певній точці Землі - це середнє вивільнення сейсмічної енергії на одиницю об'єму. Сейсмічні коливання, не пов'язані з землетрусами, також відомі як тремор.

На поверхні землетруси проявляються у вигляді трясіння та зміщення/розриву ґрунту. Якщо епіцентр сильного землетрусу знаходиться близько до узбережжя, морське дно може значно зміститися, викликавши цунамі. Землетруси також можуть викликати зсуви.

Термін "землетрус" використовується в найзагальнішому сенсі для позначення будь-якого сейсмічного явища, що спричиняє сейсмічні хвилі, як природного, так і техногенного походження. Землетруси в основному спричиняються розривами геологічних розломів, але також можуть бути викликані іншими подіями, такими як вулканічна активність, зсуви, вибухи в шахтах і ядерні випробування. Місце, де починаються поштовхи землетрусу, називається епіцентром або фокусною точкою. Епіцентр - це точка на земній поверхні безпосередньо над центром вибуху.

#### **Основні показники сейсмічної активності**

Залежно від характеру та форми запису здійснюється визначення:

- хвиль;
- параметрів сейсмічних сигналів
- епіцентральної відстані
- визначення азимуту на епіцентр
- часу землетрусу в джерелі
- глибини гіпоцентру сейсмічного джерела
- магнітуди сейсмічного джерела

#### **Суть системи моніторингу**

Моніторинг – це система постійного стеження за певними процесами, які проходять у певному середовищі спостереження, результати яких використовують для прийняття певних рішень. При автоматичному контролі відбувається обробка інформації про стан певного об'єкта для визначення управлінських дій.



## 1.2 Моделювання бізнес-процесів предметної області.

Моделювання сейсмічних процесів – це формалізований за певними правилами опису потенційних сейсмічних процесів у формі логічних блок-схем, що демонструють певну можливу подію. Моделі сейсмічної активності мають чітку послідовність дій, які об'єднуються у певні процедури і сценарії. Моделювання активності, як правило, відображають у графічному вигляді, наприклад моніторинг стану сейсмічної активності України, дані графічних процесів можна вдосконалювати.

Моделювання використовують дослідники та сейсмологи для спрощення процесів. У регіонах без використання показників сейсмічної активності дуже складно забезпечити нормальний рівень контролю землетрусів. Система відображає показники, наявність яких необхідна для прогнозів землетрусів.

Найновішими та найрозповсюднішими мовами графічного моделювання є такі мови як: UML, ARIS, IDEF, BPMN.

Діаграма IDEF0 підсистеми моніторингу представлено на рис. 1.

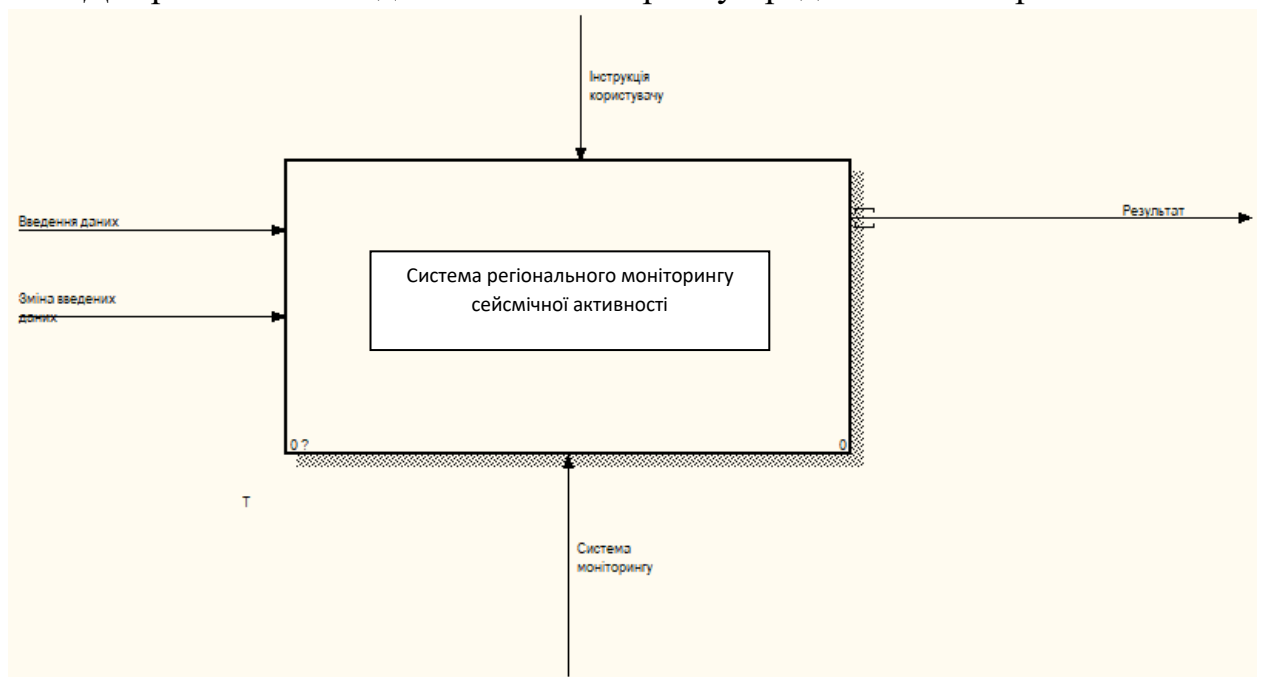


Рисунок 1.1 Контекстна діаграма IDEF0 підсистеми моніторингу сейсмічної активності в регіонах

Після створення контекстної діаграми, при необхідності, проводиться подальша декомпозиція – рисю 1.2

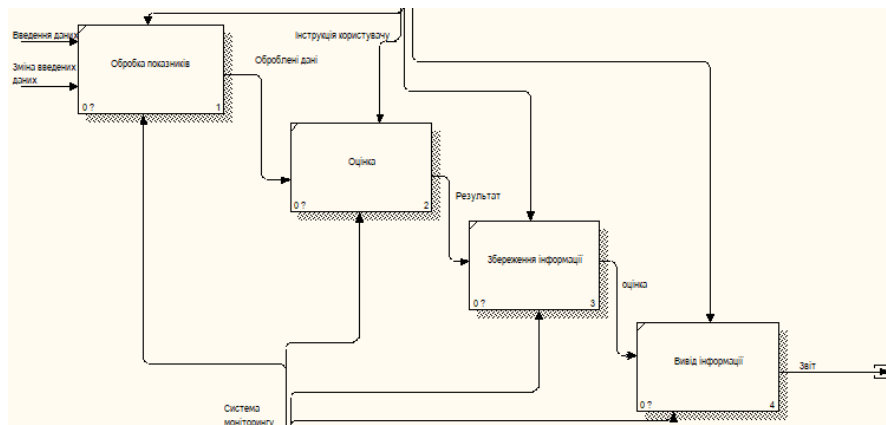


Рисунок 1.2 Діаграма декомпозиції IDEF0 підсистеми моніторингу регіональної сейсмічної активності

### Висновки до першого розділу

В розділі 1.1 та 1.2 було проведено аналіз предметної області дослідження, проведено обґрунтування обраних рішень. Була побудована модель контекстної діаграми, та діаграми декомпозиції IDEF0 підсистеми регіонального моніторингу сейсмічної активності та було пояснені основні деталі функціонування обраної предметної області дослідження.

## РОЗДІЛ 2. ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ МОНІТОРИНГУ РЕГІОНАЛЬНОЇ СЕЙСМІЧНОЇ АКТИВНОСТІ

### 2.1 Моделювання інформаційної системи моніторингу регіональної сейсмічної активності.

IDEF3 – методологія моделювання процесів, яка представляє собою механізм створення документів та збір інформації. Показує зв'язки між процесами у спрощеній формі, зрозумілих для звичайних людей, використовуючи структурний метод показу знань як функціонує система процесів.

Підсистема **моніторингу регіональної сейсмічної активності** (рисунок 2.1) має таку модель IDEF3. На ній показані всі можливі дії оператора від старту програми у меню до її всіх можливих варіантів завершення. Запустивши програму, оператор має два варіанти роботи із підсистемою моніторингу. Перший це ввести шість основних показників сейсмічної активності для отримання результату виконання програми, а саме ввести показники: - хвиль;

- параметрів сейсмічних сигналів

- епіцентральної відстані
- визначення азимуту на епіцентр
- часу землетрусу в джерелі
- глибини гіпоцентру сейсмічного джерела
- магнітуди сейсмічного джерела

(рисунок 2.1), які оброблюються підсистемою моніторингу та зводяться до єдиного показника. Якщо введені дані були не коректними, то програма автоматично завершить свою роботу, та не виведе результат, якщо дані були коректними, то виконається порівняння із іншими країнами й цей результат виведеться у вигляді звіту. Другий варіант, нічого не вводити у підсистему і завершити виконання програми.

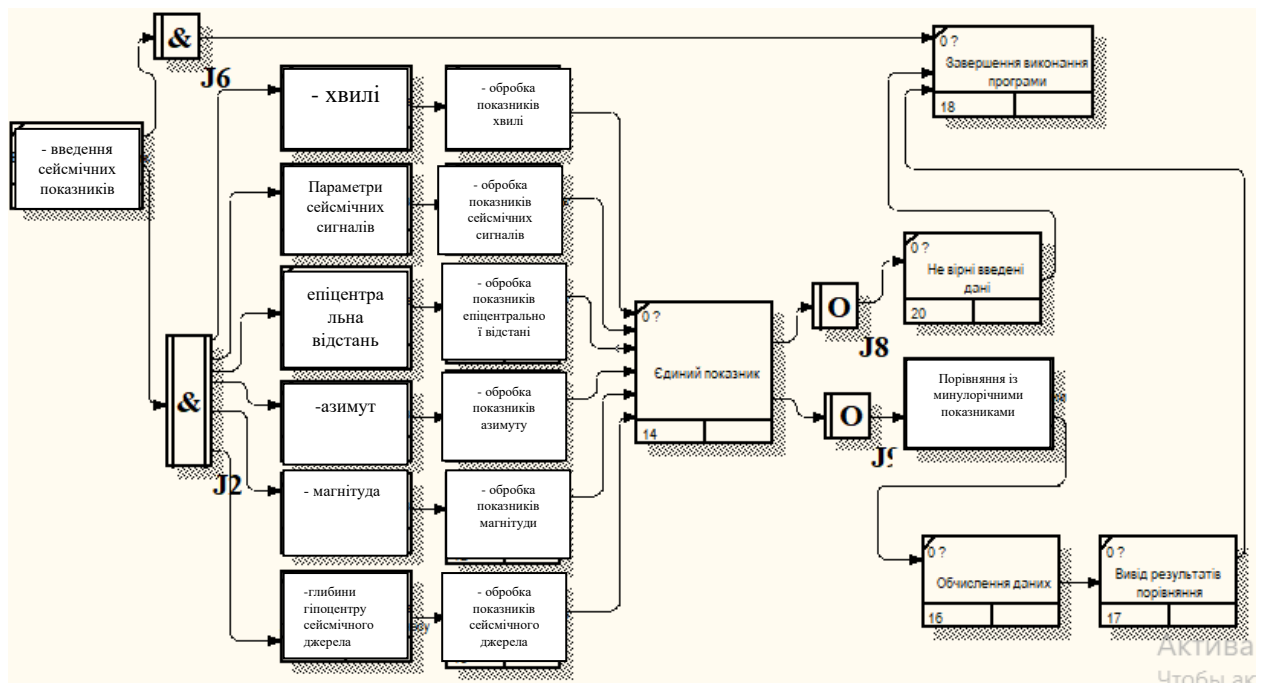


Рисунок 2.1 IDEF3 підсистеми моніторингу сейсмічної активності

Діаграма послідовності (рисунок 2.4) – одна із видів UML діаграм, на якій для певного набору об’єктів і прецедентів показаний весь цикл життєдіяльності об’єктів і взаємодію діючих обличч в інформаційній системі. Основним елементом діаграми послідовності є об’єкти, які позначаються у вигляді прямокутника з назвою прецеденту всередині його та вертикальної пунктирної лінії, яку називають “лінією життя”, вона відображає плив часу всередині системи, прямокутник відображає діяльність об’єкта, чи прецеденту, які виконують певну функцію. Стрілки відображають обмін повідомленнями між елементами системи.

На UML діаграмі підсистеми моніторингу регіональної сейсмічної активності(рисунок 2.2) продемонстровано покрокові функції у вигляді

діаграми послідовності, де оператор виконує процес “Авторизація”, вводячи логін та пароль, який написаний у інструкції користувача, після чого дані авторизації звіряються з тими, що зберігаються у блоці “База даних”, якщо введений пароль такий, який записаний у базі даних, то користувачу підсистема надає доступ, якщо авторизаційні дані не коректні, то програма видає помилку. Після отримання доступу користувач має ввести дані для розрахунку через “Інтерфейс” (рисунок 2.2), які потрапляють до модуля обчислення, який їх оброблює та зберігає у “База даних” і виводить користувачу у вигляді звіту.

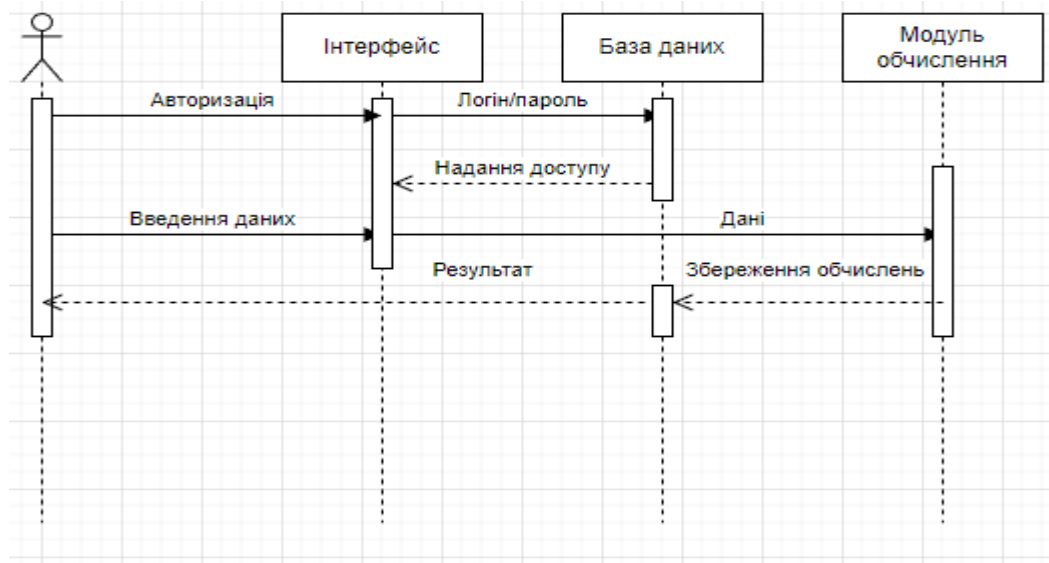


Рисунок 2.2 UML діаграма послідовності підсистеми моніторингу регіональної сейсмічної активності

На схемі UML діаграми станів (рисунок 2.3) показана зміна стану програми під час роботи за нею оператора, при спробі авторизації програма набуває один із двох станів, а саме при коректних логіну та паролі користувач попадає до головного меню, а при хибних не змінює свій стан і чекає на коректні дані авторизації. У головному меню маємо три вкладки: моніторинг, редагування даних та оцінка моніторингу сейсмічної активності, є можливість нічого не обчислювати і завершити роботу. Для отримання певного результату потрібно працювати із вкладками: моніторинг, редагування даних та оцінка На схемі діаграми прецедентів (рис.2.4) відображено всі можливі відносини між користувачем та підсистемою моніторингу, а саме: авторизація, обчислення даних, корегування та моніторинг.

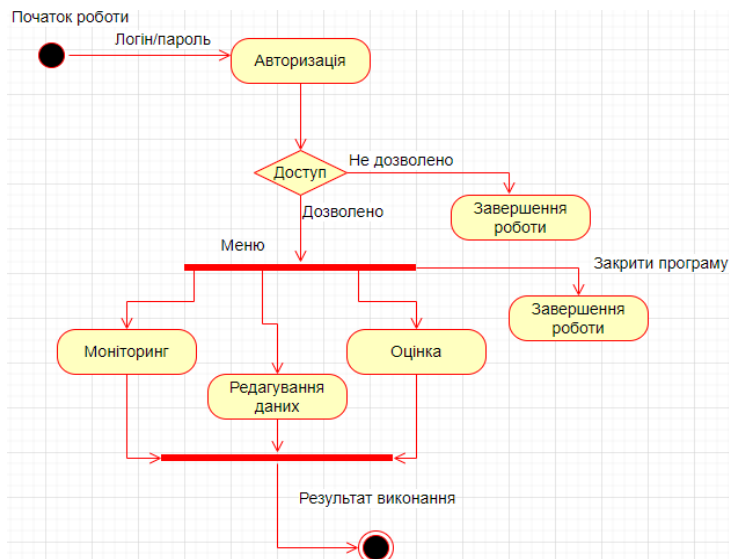


Рисунок 2.3 UML діаграма станів підсистеми моніторингу регіональної сейсмічної активності



Рисунок 2.4 UML діаграма прецедентів підсистеми моніторингу оцінювання сейсмічної обстановки регіону

## 2.2 Проектування структури бази даних інформаційної системи

UML діаграма “База даних підсистеми моніторингу оцінювання сейсмічної обстановки регіону” (рисунок 2.5) описує взаємодію функціонування програми із базою даних, у якій зберігаються всі дані та обчислення, які оператор проводить під час роботи. Під час авторизації оператор вводить логін та пароль у вікно авторизації, які звіряються із таблицею бази даних, у якій записані логін та пароль. Після авторизації користувач може виконувати три функції: моніторинг, редагування та оцінка. Результати додавання і редагування даних записуються у таблицю 2.1 “Модель таблиці показників підсистеми оцінювання сейсмічної обстановки регіону”, яку програма використовує для функції моніторингу та надання оцінки:

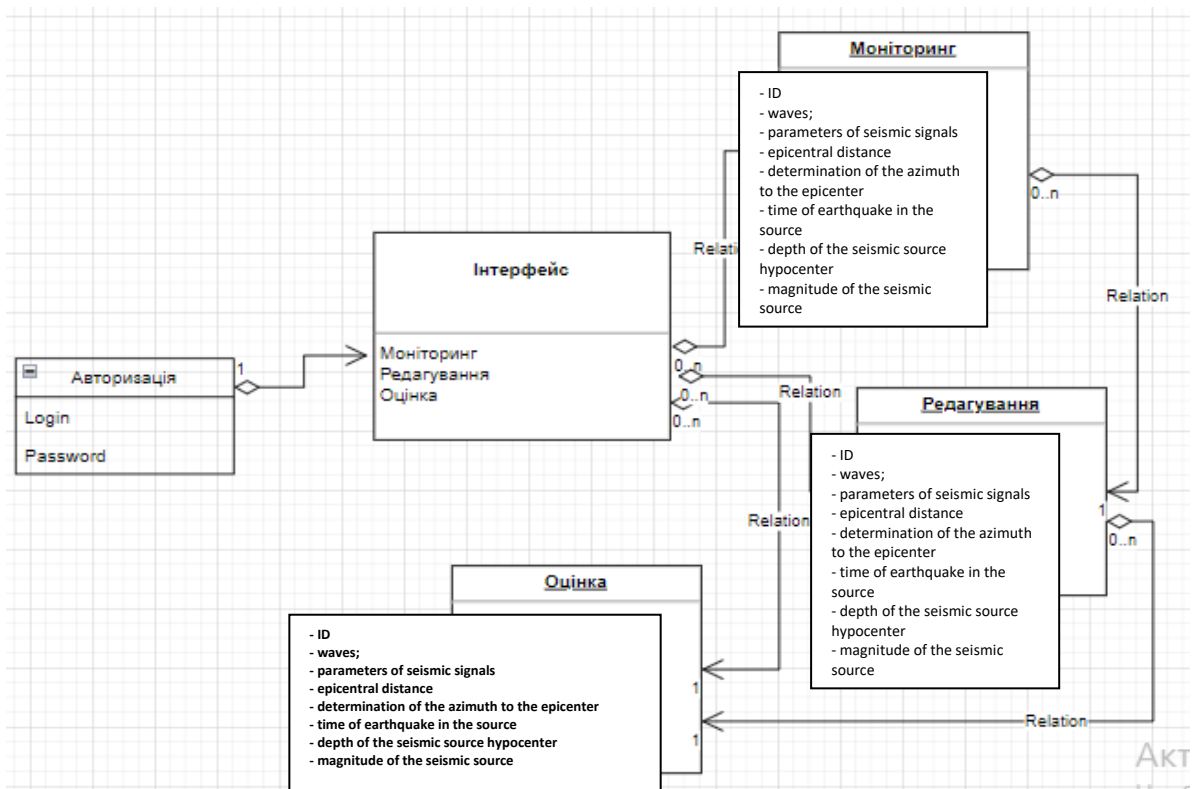


Рисунок 2.5 Структура бази даних підсистеми моніторингу регіональної сейсмічної активності

Таблиця 2.1, структуровано зберігає всі введені та обчислені у підсистему моніторингу регіональної сейсмічної активності. “Ім’я” це назва показників, які оператор вводить у програму. “Тип” це характеристика введених даних у систему (`int` числовий, `text` письменний). Довжина даних – це характеристика займаному місцю на носіях інформації для кожного типу даних.

Таблиця 2.1 Показники підсистеми моніторингу

Ім'я	Тип	Довжина,байт
ID	Byte	4
- waves;	Text	20
- parameters of seismic signals	String	4
- epicentral distance	float	4
- determination of the azimuth to the epicenter	float	4
- time of earthquake in the source	String	4
- depth of the seismic source hypocenter	float	4

### **Висновок до третього розділу**

У цьому розділі було спроектовано інформаційну систему моніторингу регіональної сейсмічної активності. Створено схематично принципи роботи підсистеми у вигляді діаграми IDEF3, діаграми послідовності, діаграми прецедентів, діаграми станів та діаграми бази даних і її таблиці. В деталях описано суть та функції кожної деталі та прецеденту.

## **РОЗДІЛ 3. РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ МОНІТОРИНГУ РЕГІОНАЛЬНОЇ СЕЙСМІЧНОЇ АКТИВНОСТІ**

### **3.1 Процес роботи алгоритму визначення землетрусу**

Розробка алгоритмів для інформаційної системи моніторингу оцінювання сейсмічної обстановки регіону включає в себе ряд ключових етапів:

1. Збір та обробка даних: Першим кроком в будь-якому процесі машинного навчання є збір і обробка даних. У нашому випадку, ми збираємо історичні дані про землетруси, які включають такі параметри, як назва регіону, магнітуда, глибина, частота, тривалість землетрусу, глибина гіпоцентру та магнітуда сейсмічного джерела. Дані зазвичай потребують попередньої обробки, що може включати видалення неповних або невірних записів, нормалізацію числових значень, кодування категорійних значень та інше.

2. Вибір моделі машинного навчання: Наступним кроком є вибір відповідної моделі машинного навчання. Модель вибирається в залежності від задачі (в даному випадку це бінарна класифікація - відбувся землетрус або ні) та виду даних.

3. Навчання моделі: Модель навчається на основі наданих навчальних даних. Під час навчання модель "вчиться" розпізнавати шаблони в даних, які пов'язують вхідні параметри (назва регіону, магнітуда, глибина тощо) з цільовим параметром (чи відбувся землетрус).

4. Валідація моделі: Після навчання моделі проводиться валідація. Це включає в себе використання моделі на тестових даних (які не використовувалися під час навчання) та оцінку її ефективності. Метрики оцінки можуть включати точність, повноту, F1-метрику та інше.

5. Використання моделі для прогнозування землетрусу у регіоні: Якщо модель проходить етап валідації, вона може бути використана для прогнозування майбутніх землетрусів. Вхідні дані (назва регіону, магнітуда тощо) подаються на вхід моделі, яка дає прогноз, чи відбудеться землетрус.

6. Оптимізація моделі: На основі результатів валідації можна оптимізувати модель, змінюючи її параметри, використовуючи різні алгоритми оптимізації або примінюючи ансамблеві методи.

7. Обробка неоднозначних ситуацій: В процесі машинного навчання іноді виникають неоднозначні ситуації, коли модель виявляється не здатною точно визначити, чи відбудеться землетрус. В цих випадках потрібно розробити стратегію обробки таких ситуацій, яка може включати в себе використання додаткових даних, альтернативних моделей або експертних оцінок.



Блок-схема на рис. 4.1 відображає процес роботи алгоритму, який використовує модель машинного навчання для прогнозування землетрусу у регіоні.

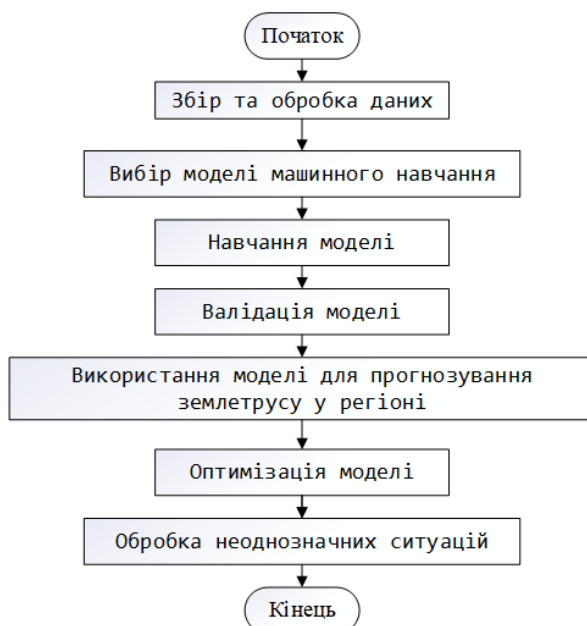


Рисунок 4.1 – Блок-схема процесу роботи алгоритму прогнозування землетрусу у регіоні

### 3.2 Обґрунтування вибору бібліотеки машинного навчання ML.NET

Вибір відповідної бібліотеки машинного навчання відіграє суттєву роль у процесі розробки інформаційної системи моніторингу оцінювання сейсмічної обстановки регіону. З метою створення такої системи, було обрано бібліотеку ML.NET, і цей вибір має свої науково-обґрунтовані підстави.

Таблиця 2.2 Обґрунтування вибору бібліотеки

Бібліотека	Мова програмування	Підтримка задач	Виділені особливості
Scikit-learn	Python	Загальнопризначена	Простота використання, широкий спектр алгоритмів
TensorFlow	Python	Загальнопризначена	Моделі глибокого навчання, розподілені обчислення
Keras	Python	Глибоке навчання	Простота використання, високорівневий інтерфейс
PyTorch	Python	Глибоке навчання	Гнучкість, динамічний обчислювальний граф
ML.NET	C#, F#, VB.NET	Загальнопризначена	Інтеграція з .NET-екосистемою, підтримка мови C#

### 3.3 Реалізація модулів системи

Розроблено 2 класи: `EarthquakeData` і `EarthquakePrediction`, які відображають структуру вхідних та вихідних даних для моделі машинного навчання, яка використовується для інформаційної системи моніторингу оцінювання сейсмічної обстановки регіону.

Клас `"EarthquakeData"` використовується в інформаційній системі моніторингу оцінювання сейсмічної обстановки регіону. Даний клас містить наступні поля:

- `"RegionName"` (тип: `string`): це поле містить назву регіону, пов'язану з сейсмічною активністю;
- `"Magnitude"` (тип: `float`): воно відображає магнітуду зареєстрованого землетрусу. Магнітуда є числовим показником сили землетрусу і використовується для оцінки його потенційної небезпеки;
- `"Depth"` (тип: `float`): це поле вказує глибину, на якій стався землетрус. Глибина є важливою характеристикою, оскільки вона може впливати на поширення та наслідки землетрусу;
- `"Frequency"` (тип: `float`): воно представляє собою частоту сейсмічних коливань, пов'язаних з землетрусом. Це важлива характеристика, оскільки частота може впливати на поведінку будівель та інфраструктури під час землетрусу;
- `"EarthquakeDuration"` (тип: `float`): це поле вказує тривалість землетрусу, тобто час, протягом якого спостерігаються сейсмічні коливання. Ця характеристика допомагає оцінити тривалість події та її потенційні наслідки;
- `"HypocenterDepth"` (тип: `float`): воно вказує на глибину гіпоцентру, тобто точки, де виникають сейсмічні коливання. Інформація про глибину гіпоцентру допомагає визначити можливі ризики та наслідки землетрусу;
- `"SeismicSourceMagnitude"` (тип: `float`): це поле відображає магнітуду сейсмічного джерела, що спричинило землетрус. Ця характеристика допомагає розуміти силу та інтенсивність сейсмічного події;
- `"EarthquakeOccurrence"` (тип: `bool`): це логічне поле, яке показує, чи сталася землетрусна подія в даному регіоні. Значення `"true"` вказує на те, що землетрус стався, а значення `"false"` вказує на відсутність землетрусу.

Цей клас використовує атрибут `"LoadColumn"` для вказівки порядкового номера стовпця при завантаженні даних. Це допомагає системі коректно ідентифікувати та відповідати значенням відповідних полів класу під час обробки сейсмічної інформації.

Клас `"EarthquakePrediction"` містить наступні поля:

– "PredictedLabel" (тип: bool): це поле представляє прогнозовану мітку, що вказує на наявність або відсутність землетрусу в даному регіоні. Значення "true" вказує на прогнозовану наявність землетрусу, а значення "false" вказує на прогнозовану його відсутність;

– "Probability" (тип: float): це поле відображає ймовірність здійснення прогнозованої мітки. Значення цього поля вказує на впевненість системи щодо свого прогнозу і може варіюватися від 0 до 1, де 0 означає низьку ймовірність, а 1 - високу ймовірність;

– "Threat" (тип: bool): це поле відображає потенційну загрозу землетрусу. За допомогою атрибутів "LoadColumn" та "ColumnName", дане поле відповідає значенню "EarthquakeOccurrence" з вхідних даних. Значення "true" вказує на потенційну загрозу землетрусу, а значення "false" вказує на відсутність потенційної загрози.

Цей клас використовує атрибути "ColumnName" для вказівки імені стовпців при обробці даних. Це допомагає системі коректно ідентифікувати та відповідати значенням відповідних полів класу під час прогнозування сейсмічної активності та оцінювання потенційних загроз.

Наступним кроком було створення змінних: model, context та data.

Змінна "model" є типом "TransformerChain <BinaryPredictionTransformer <CalibratedModelParametersBase <FastTreeBinaryModelParameters, PlattCalibrator>>>" і використовується для зберігання моделі, яка використовується в системі. Ця модель включає ланцюжок трансформаторів, що дозволяє виконувати послідовні перетворення над даними. Конкретно, цей ланцюжок містить бінарний прогностичний трансформатор, який забезпечує функціонал для бінарного прогнозування, та калібровані параметри моделі, які базуються на параметрах FastTreeBinaryModelParameters та PlattCalibrator.

Змінна "context" є типом "MLContext" і використовується для створення та налаштування контексту машинного навчання (ML). Цей контекст забезпечує доступ до різних функцій та інструментів, які використовуються для розробки моделей машинного навчання, включаючи завантаження даних, побудову моделей та оцінку їх ефективності.

Змінна "data" є типом "IDataView" і представляє набір даних, який використовується для навчання та тестування моделі. Цей об'єкт даних містить вхідні дані, на основі яких модель буде виконувати прогнози. IDataView є одним з основних типів даних, які використовуються в ML.NET для обробки та передачі даних у моделі машинного навчання.

Після цього була додана подія "OpenBtn\_Click", яка має на меті відкриття файлу, що містить тренувальні дані, та проведення навчання нейронної моделі. Для забезпечення відкриття файлу використовується

компонент OpenFileDialog. Після успішного відкриття файлу тренувальні дані зберігаються у вигляді списку класів SensorInput, який називається "datas". Процес зчитування даних був реалізований у відповідності до пункту 4.1. у лістингу.

#### Лістинг 4.1 – Процес зчитування тренувальних даних із файлу

```
using (StreamReader reader = new StreamReader(_Path, Encoding.GetEncoding(1251))) {  
    // Пропускаємо перший рядок з іменами стовпців  
    reader.ReadLine();  
    while (!reader.EndOfStream) {  
        var line = reader.ReadLine();  
        var values = line.Split(';');  
        EarthquakeData item = new EarthquakeData() {  
            RegionName = values[0],  
            Magnitude = float.Parse(values[1]),  
            Depth = float.Parse(values[2]),  
            Frequency = float.Parse(values[3]),  
            EarthquakeDuration = float.Parse(values[4]),  
            HypocenterDepth = float.Parse(values[5]),  
            SeismicSourceMagnitude = float.Parse(values[6]),  
            EarthquakeOccurrence = bool.Parse(values[7])  
        };  
        datas.Add(item);  
    }  
}
```

Наведений код виконує завдання зчитування даних з файлу для інформаційної системи моніторингу оцінювання сейсмічної обстановки регіону. Для цього використовується об'єкт StreamReader для читання файлу із вказаного шляху (\_Path) з використанням кодування 1251 (Windows-1251).

Усередині блоку using, після відкриття файлу, перший рядок, який містить імена стовпців, пропускається за допомогою методу ReadLine(). Потім відбувається процес зчитування решти рядків файлу.

Кожен рядок зчитується за допомогою методу ReadLine() і розбивається на окремі значення за допомогою розділювача ";" за допомогою методу Split(). Отримані значення зберігаються у змінній values у вигляді масиву рядкових значень.

Наступною дією є створення нового об'єкта класу EarthquakeData. Значення з масиву values перетворюються на відповідні типи даних та призначаються відповідним властивостям об'єкта EarthquakeData.

Створений об'єкт item додається до списку datas, який, очевидно, містить дані про землетрусну активність регіону.

Цей процес повторюється для кожного рядка в файлі до тих пір, поки не буде досягнутий кінець файлу (EndOfStream).

Після завершення циклу зчитування, об'єкт `StreamReader` автоматично закривається завдяки використанню блоку `using`.

Для використання бібліотеки `ML.NET`, набір даних потребує підготовки та розділення відповідно до контексту (ліс. 4.2).

#### Лістинг 4.2 – Підготовка та розділення даних

```
context = new MLContext(seed: 0);  
// Завантаження даних  
data = context.Data.LoadFromEnumerable<EarthquakeData>(datas);  
var trainTestData = context.Data.TrainTestSplit(data, testFraction: 0.2, seed: 0);  
var trainData = trainTestData.TrainSet;  
var testData = trainTestData.TestSet;
```

Наведений код відповідає за роботу з даними в інформаційній системі моніторингу оцінювання сейсмічної обстановки регіону.

Спочатку створюється об'єкт контексту `MLContext` з вказаною початковою точкою (`seed`) рівною 0. `MLContext` - це основний компонент бібліотеки `ML.NET`, який використовується для взаємодії з різними функціями та алгоритмами машинного навчання.

Потім виконується завантаження даних за допомогою методу `LoadFromEnumerable<EarthquakeData>(datas)`, де `datas` є списком об'єктів класу `EarthquakeData`. Контекст `MLContext.Data.LoadFromEnumerable<T>()` використовується для завантаження даних з джерела, переданого як `Enumerable`, тобто у вигляді колекції або списку. Завантажені дані зберігаються у змінній `data` типу `IDataView`.

Далі, за допомогою методу `TrainTestSplit()`, дані розділяються на навчальний та тестовий набори даних. Параметр `testFraction: 0.2` визначає відсоток даних, які будуть використані для тестування (у цьому випадку 20%), а `seed: 0` встановлює початкову точку для генератора випадкових чисел, що дозволяє повторювати результати при кожному запуску. Результатом розділення є два набори даних - `trainData` для навчання моделі та `testData` для тестування моделі.

Після попередніх кроків, необхідно створити конвеєр для обробки даних перед тренуванням моделі машинного навчання в бібліотеці `ML.NET`, згідно лістингу 4.3.

#### Лістинг 4.3 – Створення конвеєра для обробки даних

```
// Визначення конвеєра обробки даних  
var pipeline = context.Transforms.Concatenate("Features", nameof(EarthquakeData.Magnitude),  
nameof(EarthquakeData.Depth),  
nameof(EarthquakeData.Frequency), nameof(EarthquakeData.EarthquakeDuration),  
nameof(EarthquakeData.HypocenterDepth), nameof(EarthquakeData.SeismicSourceMagnitude))  
.Append(context.Transforms.NormalizeMinMax("Features"))  
.Append(context.Transforms.CopyColumns(inputColumnName: "EarthquakeOccurrence",  
outputColumnName: "Label"))
```

```
.Append(context.BinaryClassification.Trainers.FastTree());
```

Даний код відповідає за створення конвеєра обробки даних перед тренуванням моделі машинного навчання в інформаційній системі моніторингу оцінювання сейсмічної обстановки регіону.

Спочатку створюється об'єкт конвеєра "pipeline" за допомогою класу `MLContext.Transforms`. Конвеєр включає наступні кроки обробки даних:

- метод `Concatenate()` використовується для об'єднання декількох вхідних стовпців даних, що представлені як рядки, в один вихідний стовпець "Features". У нашому випадку, об'єднані стовпці включають `Magnitude`, `Depth`, `Frequency`, `EarthquakeDuration`, `HypocenterDepth` та `SeismicSourceMagnitude`;
- метод `NormalizeMinMax()` застосовує нормалізацію даних за допомогою мінімального та максимального значень. Це допомагає привести дані до одного діапазону для забезпечення стабільності та ефективності моделі;
- метод `CopyColumns()` копіює значення стовпця "EarthquakeOccurrence" у вихідний стовпець "Label". Це необхідно для задання цільової змінної моделі бінарної класифікації;
- метод `FastTree()` визначає алгоритм навчання моделі бінарної класифікації, в даному випадку - швидке дерево.

Отже, створений конвеєр об'єднує дані з декількох стовпців в один стовпець "Features", застосовує нормалізацію, копіює значення цільової змінної до вихідного стовпця "Label" та використовує алгоритм швидкого дерева для навчання моделі бінарної класифікації. Цей конвеєр буде використовуватися для обробки та тренування моделі машинного навчання.

Після створення конвеєра, наступним кроком є навчання нейронної мережі та збереження моделі, згідно лістингу 4.4.

#### Лістинг 4.4 – Навчання НМ

```
model = pipeline.Fit(trainData); //Навчання моделі
```

Навчена модель зберігається у змінній `model` для подальшого використання. Тепер навчену модель обов'язково потрібно перевірити (ліст 4.5).

#### Лістинг 4.5 – Перевірка навченої моделі

```
// Перевірка моделі  
var predictions = model.Transform(testData);  
var metrics = context.BinaryClassification.Evaluate(predictions);  
RaportTBox.Text += ("Accuracy: {metrics.Accuracy:P2}") + "\r\n";  
RaportTBox.Text += ("F1Score: {metrics.F1Score:P2}") + "\r\n";
```

Наведений код виконує перевірку моделі та оцінює її ефективність в інформаційній системі моніторингу оцінювання сейсмічної обстановки регіону.

Спочатку застосовується метод Transform() до моделі з використанням тестового набору даних testData. Цей метод виконує прогнозування на основі тестових даних та повертає результати прогнозу.

Потім використовується метод Evaluate() з параметром predictions для оцінки ефективності моделі. Цей метод розраховує різні метри для задачі бінарної класифікації, такі як точність (Accuracy) та F1-показник (F1Score), які вказують на якість прогнозів моделі.

Отримані метри ефективності моделі записуються у відповідні поля (Accuracy та F1Score) текстового поля ReportTextBox для подальшого відображення в інформаційній системі.

Таким чином, наведений код виконує оцінку та звіт про ефективність моделі, що застосовується для оцінювання сейсмічної обстановки регіону.

При натисканні кнопки, спочатку перевіряється коректність введених даних за допомогою методу IsDataEnteringCorrect(). Цей метод виконує перевірку належного формату та правильності введених даних.

Якщо дані введені коректно, виконуються наступні дії:

- Створюється рядок "pathName" для збереження моделі у вигляді архіву з розширенням ".zip". Рядок містить шлях до теки "teach" та згенероване унікальне ім'я файлу за допомогою методу GenerateFileName();

- Отримується локальний шлях до виконуваного файлу за допомогою класу System.IO.Path та System.Reflection.Assembly.GetExecutingAssembly().Location;

- Викликається метод InsertNeural() об'єкта "\_NeuralProvider" для вставки нейромережі до бази даних з вказаними параметрами, такими як назва нейромережі (отримана з текстового поля NeuralNamesTextBox) та ідентифікатор сценарію (отриманий з вибраного значення в ComboBox ScenariosCBox);

- Викликається метод Save() об'єкта контексту MLContext.Model для збереження моделі за вказаним шляхом. Модель, контекст даних (data.Schema) та локальний шлях (localProj + pathName) передаються як параметри;

- Очищуються всі введені дані за допомогою методу ClearAllData();

- Викликається метод InsertLogs() об'єкта "\_LogsProvider" для вставки запису про навчання нейромережі до бази даних. Цей запис містить ідентифікатор користувача, текстове повідомлення ("Було навчено нейронну мережу " + NeuralNamesTextBox.Text) та поточний час;

– Показується повідомлення користувачеві про успішне збереження даних за допомогою `MessageBox.Show("Дані успішно збережено!")`.

Було розроблено форму для прогнозування можливого землетрусу у реальному часі, яка дозволяє користувачу завантажити навчену модель даних. Для забезпечення завантаження моделі була реалізована подія `"ScenariosCBox_SelectedValueChanged"`, яка автоматично викликається після вибору потрібної моделі з випадваючого списку.

При зміні вибраного значення в `ComboBox "ScenariosCBox"`, викликається дана подія і виконуються наступні дії:

– перевіряється, чи завантажені теми (themes) даних, шляхом перевірки значення змінної `"_IsThemesLoad"`;

– якщо теми даних були завантажені, виконується наступний блок коду;

– змінна `"_SelectedNeural"` отримує значення навченої нейромережі, яка відповідає обраному ідентифікатору сценарію (`ScenariosCBox.SelectedValue`). Це виконується за допомогою методу `"SelectedNeuralByScenariosId"` об'єкта `"_NeuralProvider"`, який повертає об'єкт, що містить інформацію про вибрану нейромережу;

– викликається метод `"LoadData"` з аргументом `"_SelectedNeural.NeuralFileModel"`, який відповідає за завантаження даних моделі. Цей метод виконує необхідні дії для завантаження та підготовки даних моделі.

Отже, наведений код відповідає за обробку події зміни вибраного значення в `ComboBox "ScenariosCBox"` і виконує необхідні дії, щоб вибрати відповідну навчену нейромережу та завантажити відповідні дані для подальшого використання в системі моніторингу сейсмічної обстановки регіону.

Для генерації випадкових даних про землетрус для інформаційної системи моніторингу оцінювання сейсмічної обстановки регіону. було розроблено метод `«GenerateRandomEarthquakeData»`, код якого представлено у ліст. 4.6.

Лістинг 4.6 – Код методу `«GenerateRandomEarthquakeData»`

```
private EarthquakeData GenerateRandomEarthquakeData() {
    var rand = new Random();

    var earthquakeData = new EarthquakeData {
        RegionName = _CitiesList[rand.Next(_CitiesList.Count)].CitiesName,
        Magnitude = (float)(rand.NextDouble() * (10.0 - 1.0) + 1.0), // generate a random float
        between 1.0 and 10.0
    };
}
```



```

        Depth = (float)(rand.NextDouble() * (100.0 - 1.0) + 1.0), // generate a random float
        between 1.0 and 100.0
        Frequency = (float)(rand.NextDouble() * (10.0 - 0.1) + 0.1), // generate a random float
        between 0.1 and 10.0
        EarthquakeDuration = (float)(rand.NextDouble() * (60.0 - 1.0) + 1.0), // generate a
        random float between 1.0 and 60.0
        HypocenterDepth = (float)(rand.NextDouble() * (300.0 - 1.0) + 1.0), // generate a
        random float between 1.0 and 300.0
        SeismicSourceMagnitude = (float)(rand.NextDouble() * (10.0 - 1.0) + 1.0), // generate
        a random float between 1.0 and 10.0
    };

```

Функція "GenerateRandomEarthquakeData()" виконує наступні дії:

1. Створюється об'єкт типу Random для генерації випадкових чисел.
2. Створюється об'єкт "earthquakeData" типу EarthquakeData, який містить інформацію про землетрус.
3. За допомогою випадкових чисел із заданого діапазону, встановлюються значення для різних параметрів землетрусу, таких як:
  - RegionName: вибирається випадкове місто зі списку "\_CitiesList".
  - Magnitude: генерується випадкове дійсне число від 1.0 до 10.0.
  - Depth: генерується випадкове дійсне число від 1.0 до 100.0.
  - Frequency: генерується випадкове дійсне число від 0.1 до 10.0.
  - EarthquakeDuration: генерується випадкове дійсне число від 1.0 до 60.0.
  - HypocenterDepth: генерується випадкове дійсне число від 1.0 до 300.0.
  - SeismicSourceMagnitude: генерується випадкове дійсне число від 1.0 до 10.0.
4. Повертається об'єкт "earthquakeData", який містить згенеровані випадкові дані про землетрус.

Для виведення згенерованих даних та визначення можливого землетрусу у реальному часі та перевірки навченої моделі на адекватність, розроблено подію таймера «timer1\_Tick»

### **3.4 Деталізація розроблених алгоритмів**

У даному розділі викладено розробку основних алгоритмів, які необхідні для ефективної роботи інформаційної системи моніторингу та оцінювання сейсмічної обстановки в регіоні. На рис. 1.2 наведена блок-схема, яка ілюструє процес введення інформації про навчену модель тестових даних в систему.

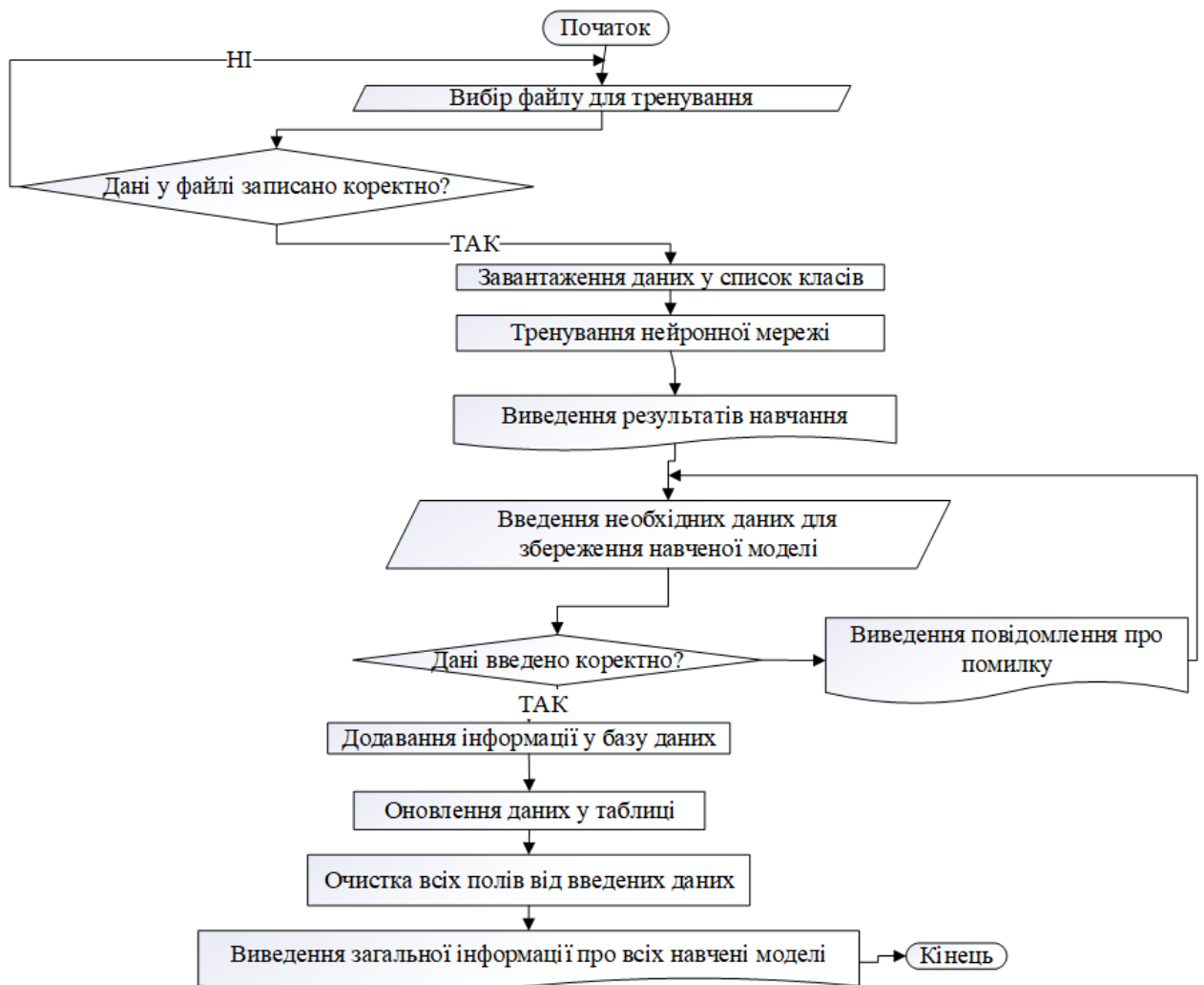


Рисунок 4.2 – Додавання інформації про навчену модель в базу даних

Блок-схема алгоритму тренування нейронної мережі може бути описана наступним чином:

1. Вибір файлу для тренування.
2. Перевірка коректності формату даних у файлі. Якщо дані записано коректно, переходимо до кроку 3, в іншому випадку повертаємося до кроку 1.
3. Завантаження даних з файлу та збереження їх у список класів.
4. Тренування нейронної мережі за допомогою навчального набору даних.
5. Виведення результатів навчання, таких як точність, помилка, графіки залежності тощо.
6. Введення необхідних даних для збереження навченої моделі, наприклад, назви моделі, параметрів, шляху для збереження.
7. Перевірка коректності введених даних. Якщо дані введено коректно, переходимо до кроку 8, в іншому випадку виводимо повідомлення про помилку та повертаємося до кроку 6.
8. Додавання інформації про навчену модель до бази даних.
9. Оновлення даних у таблиці, що відображає навчені моделі.

10. Очищення всіх полів від введених даних для наступного тренування.

11. Виведення загальної інформації про всі навчені моделі, наприклад, кількість моделей, середню точність, найкращу модель тощо.

Рис. 3.3 відображає блок-схему роботи навченої моделі у реальному часі.

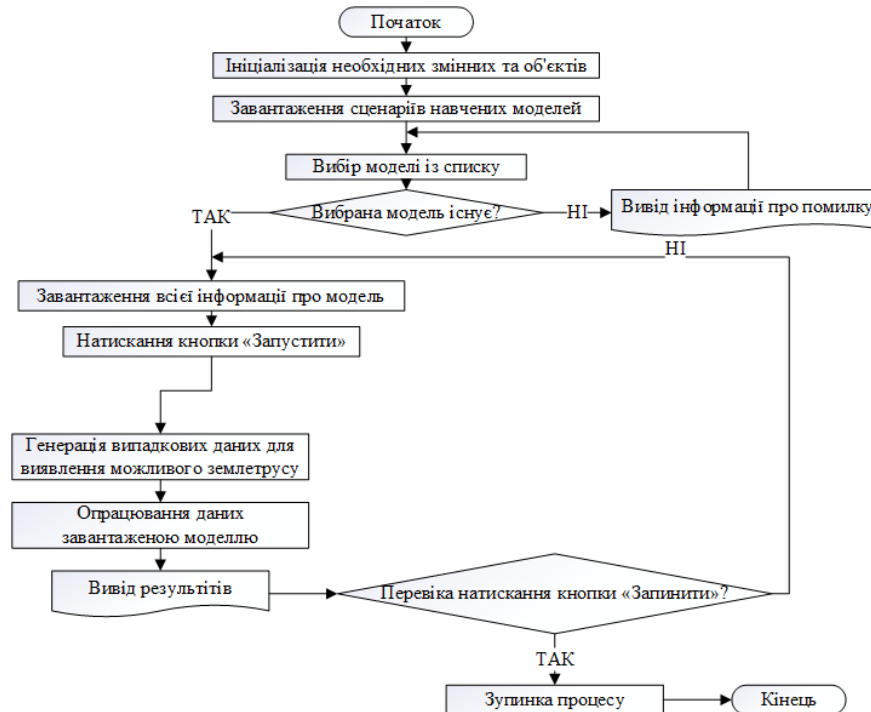


Рисунок 4.3 – Процес роботи навченої моделі

Блок-схема тренування нейронної мережі може бути описана наступним чином:

1. Ініціалізація необхідних змінних та об'єктів.
2. Завантаження сценаріїв навчених моделей.
3. Вибір моделі зі списку доступних моделей.
4. Перевірка, чи обрана модель існує. Якщо так, переходимо до кроку 5, в іншому випадку виводимо повідомлення про помилку та повертаємося до кроку 3.
5. Завантаження всієї необхідної інформації про обрану модель.
6. Натискання кнопки "Запустити" для початку процесу прогнозування.
7. Генерація випадкових даних для виявлення можливого землетрусу.
8. Обробка даних за допомогою завантаженої моделі.
9. Виведення результатів прогнозування.
10. Перевірка, чи була натиснута кнопка "Зупинити". Якщо так, переходимо до кроку 11, в іншому випадку повертаємося до кроку 5.
11. Зупинка процесу прогнозування.

### 3.5 Проведення експериментів з метою перевірки ефективності розробленої системи

Експериментальна перевірка або тестування є суттєвим етапом у процесі розробки програмного продукту з метою перевірки його функціональності та відповідності вимогам технічного завдання. Головною метою цього етапу є задоволення потреб користувачів та переконання у відповідності програми встановленим критеріям якості.

У контексті нашого проекту, експериментальна перевірка включає тестування навченої моделі на різних наборах тестових даних. Основною метою цього тестування є переконання у належному функціонуванні моделі та її здатності адекватно прогнозувати землетруси.

Для проведення тестування нейронної мережі використовуються створені дані, які містять різноманітні варіанти параметрів. Ці дані зберігаються у форматі CSV файлу, що спрощує їх обробку та подальше використання під час тестування. (рис. 3.4).

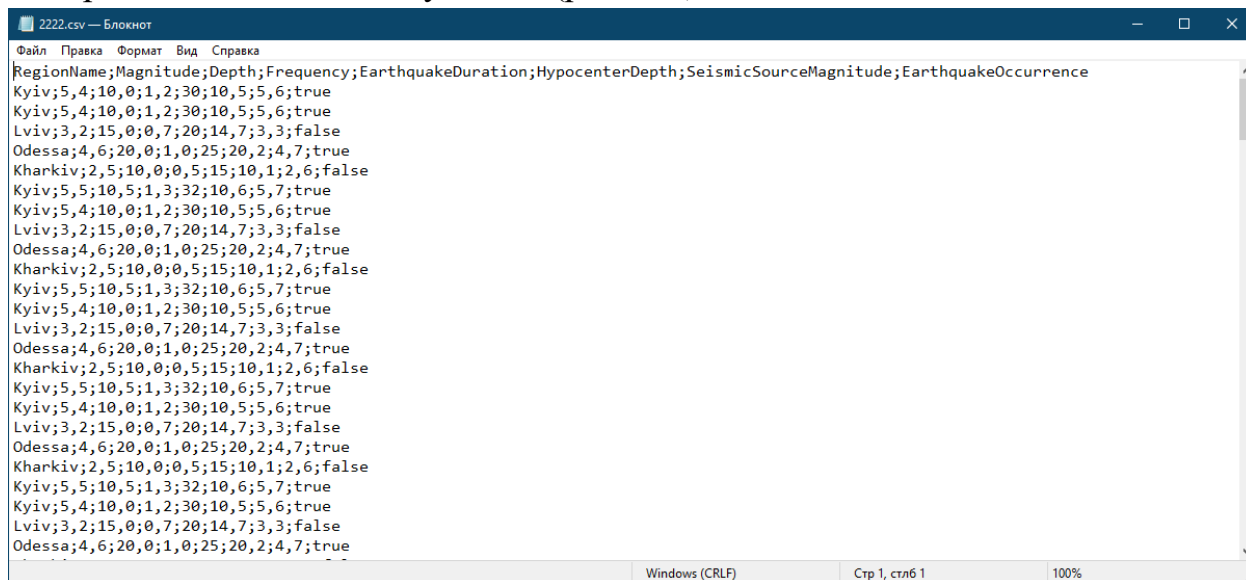


Рисунок 3.4— Фрагмент вікна із даними навчання

Після цього у програмі було вибрано цей файл та проведено навчання нейронної мережі (рис. 3.5).

Після відкриття файлу, навчання мережі було проведено за допомогою раніше згаданого методу. Процес навчання відображається в нижній лівій частині вікна. Згідно з рис. 3.5, були розраховані метрики для оцінки точності навчання:

- Точність (Accuracy).
- F1-оцінка (F1-Score).

Після завершення процесу навчання, було проведено експериментальні тести, використовуючи випадково згенеровані дані, з метою оцінки точності

навченої моделі нейронної мережі. Зображено результати цих тестів на рис. 3.6.

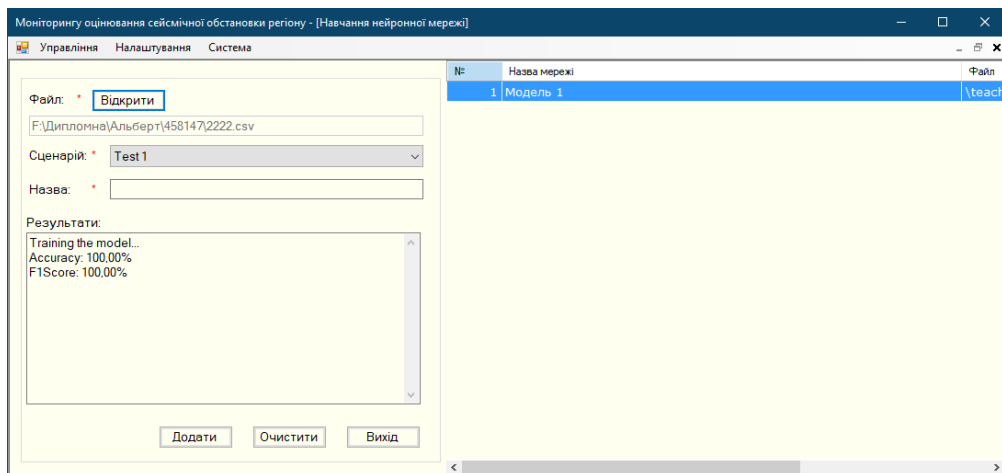


Рисунок 3.5– Форма із результатом навчання нейронної мережі

Назва регіону	Мігнітуда	Глибина	Частота	Тривалість землетрусу	Глибина гіпоцентру	Мігнітуда сейс. джерела	Можливість землетрусу
Викшиця	8,789409	17,83043	5,918552	37,43158	203,8944	6,339672	У даному регіоні є можливість землетрусу
Подільськ	7,249655	95,37128	4,086261	56,01873	63,36891	8,515309	У даному регіоні є можливість землетрусу
Підгородне	13,209133	4,112337	8,886885	47,96823	259,547	6,658596	У даному регіоні не може бути землетрусу
Ворожба	1,829378	82,2532	7,984294	7,555387	119,021	8,834312	У даному регіоні не може бути землетрусу
Волокиск	6,748857	90,59425	13,966818	58,50488	116,19962	6,97752	У даному регіоні є можливість землетрусу
П'ятикватка	15,399193	169,13511	12,824237	118,69204	174,6737	9,153236	У даному регіоні є можливість землетрусу
Лустинки	1,228581	77,47616	7,944951	18,84154	71,85224	7,296443	У даному регіоні не може бути землетрусу
Гола Пристань	8,788826	56,01782	6,84856	28,62869	126,3263	9,47216	У даному регіоні є можливість землетрусу
Гнізів	4,788384	64,35387	12,823884	28,57819	127,5849	7,615367	У даному регіоні є можливість землетрусу
Роздільна	3,26855	42,89893	8,1186929	39,16535	285,9789	9,791884	У даному регіоні не може бути землетрусу
Родинське	8,188018	51,23998	6,081217	31,11485	183,1575	7,934291	У даному регіоні є можливість землетрусу
Давняк	6,748274	29,78884	4,896826	49,762	42,63153	1,116889	У даному регіоні є можливість землетрусу
Двигач	4,667763	38,12189	8,97915	44,6515	238,8181	8,353216	У даному регіоні не може бути землетрусу
Світлодарськ	1,227998	16,66275	8,974959	1,238856	98,28415	1,428933	У даному регіоні не може бути землетрусу
Сватове	6,147476	25,8838	4,857483	52,18815	194,4627	8,57214	У даному регіоні є можливість землетрусу
Діброва	4,707722	3,544662	12,453892	11,77531	153,9368	1,747856	У даному регіоні є можливість землетрусу
Дубно	9,6272	11,88571	8,835616	3,724807	51,11536	8,891864	У даному регіоні є можливість землетрусу
Славутич	8,187446	89,42657	6,131225	22,31196	209,5894	2,866781	У даному регіоні є можливість землетрусу
Сколе	4,186924	97,76762	12,113749	14,26346	186,768	9,289988	У даному регіоні є можливість землетрусу
Залішки	2,66717	76,38849	8,2893579	32,84862	265,242	2,385784	У даному регіоні не може бути землетрусу
Залізнич порт	7,586648	84,64954	6,891882	24,79811	162,4286	9,528912	У даному регіоні є можливість землетрусу
Старокостянтинів	6,146884	63,1994	4,137491	43,38527	121,89465	2,784628	У даному регіоні є можливість землетрусу
Старий Самбір	2,066372	71,53144	8,1788149	35,33477	218,8732	9,847836	У даному регіоні не може бути землетрусу
Зоринськ	9,628617	58,0723	8,165624	53,92192	77,54727	3,823552	У даному регіоні є можливість землетрусу
Золочів	15,546896	58,41336	1,148148	45,87142	275,7259	1,166759	У даному регіоні є можливість землетрусу
Тернопіль	4,188342	36,89432	2,943757	5,455879	133,1899	3,365476	У даному регіоні є можливість землетрусу
Теребівля	9,82582	45,29527	8,126281	56,40888	18,37848	1,485683	У даному регіоні є можливість землетрусу
Калинівка	7,586866	23,83613	6,22189	15,99523	188,8525	3,6644	У даному регіоні є можливість землетрусу

Рисунок 3.6– Проведення експериментів із згенерованими даними

На основі проведеного експерименту можна зробити висновок, що навчена нейронна мережа успішно працює з наданими тренувальними даними.

### Висновок до третього розділу

У четвертому розділі було розроблено інформаційну систему. Було створено об'єктну структуру серверного застосунку, створено базу даних підсистеми моніторингу стану регіональної сейсмічної активності та розроблено інтерфейс користувача для взаємодії з нею, було продемонстровано функціональність системи за допомогою “Screenshot” із запущеного застосунку, створено інструкцію для оператора підсистеми моніторингу, якою необхідно користуватися при використуванні програми.

## **Висновки та пропозиції**

Розробка системи моніторингу регіональної сейсмічної активності виявилася вагомим внеском у сферу сейсмології та моніторингу землетрусів. Додаток успішно виконує покладені на нього завдання та функції, демонструючи свою ефективність у виявленні, аналізі та зберіганні сейсмічних даних. Функціональна модель системи моніторингу, представлена різними діаграмами, такими як IDEF0, IDEF3, послідовностей, прецедентів, станів, а також схемою і таблицею бази даних, дає комплексне розуміння роботи і структури системи.

Завдяки реалізації підсистеми моніторингу додаток покращує роботу операторів, які займаються моніторингом сейсмічної активності. Система розраховує заявки і зберігає результати в базі даних, що дозволяє ефективно управляти даними та аналізувати їх. Така автоматизація значно зменшує ручні зусилля, необхідні для обробки даних, і підвищує точність і надійність отриманої інформації. Здатність програми обробляти великі обсяги даних сприяє вдосконаленню процесів прийняття рішень і підтримує розробку комплексних стратегій управління сейсмічними ризиками.

Розроблена система моніторингу має великий потенціал для полегшення та вдосконалення операцій сейсмічного моніторингу. Її можливості виходять за рамки поточної сфери застосування, забезпечуючи міцну основу для майбутніх удосконалень і розробок. Інтеграція системи з передовими технологіями, такими як штучний інтелект і машинне навчання, може ще більше підвищити її точність і можливості прогнозування. Крім того, здатність програми зберігати та отримувати великий обсяг інформації відкриває можливості для всебічного аналізу, досліджень та співпраці в рамках наукової спільноти. Загалом, система моніторингу слугує цінним інструментом для оцінки сейсмічної активності, раннього попередження та прийняття рішень, сприяючи підвищенню загальної безпеки та стійкості громад, що зазнають сейсмічної небезпеки.

На закінчення, розробка системи моніторингу регіональної сейсмічної активності є значним досягненням у сфері моніторингу землетрусів. Успішна робота програми, продемонстрована її здатністю виконувати поставлені завдання, аналізувати сейсмічні дані та зберігати інформацію, підкреслює її ефективність у підвищенні якості та ефективності операцій сейсмічного моніторингу. З подальшим розвитком та інтеграцією нових технологій система має величезний потенціал для покращення управління сейсмічними ризиками та надання цінної інформації про сейсмічні явища. Внесок програми в сейсмічну науку та її потенційний вплив на безпеку і добробут громад, що піддаються сейсмічній небезпеці, роблять її цінним надбанням у сфері моніторингу та реагування на землетруси.

Пропозиції щодо майбутніх удосконалень:

Інтеграція даних у режимі реального часу: Удосконалити систему моніторингу шляхом інтеграції даних у реальному часі з додаткових сейсмічних станцій і датчиків. Це забезпечить більш повне та сучасне розуміння сейсмічної активності, що дасть змогу швидше реагувати та робити точніші прогнози.

Передові методи аналізу даних: Дослідити впровадження передових методів аналізу даних, таких як алгоритми машинного навчання і розпізнавання образів, щоб покращити здатність системи виявляти і класифікувати сейсмічні події. Ці методи можуть підвищити точність ідентифікації подій і сприяти більш точному прогнозуванню.

Географічне розширення: Розширити географічне покриття системи моніторингу шляхом встановлення додаткових сейсмічних станцій у районах, схильних до сейсмічної активності, але наразі не охоплених достатнім моніторингом. Це забезпечить ширше розуміння сейсмічних подій і дасть змогу точніше оцінювати потенційні ризики в регіонах, які раніше не були охоплені моніторингом.

Покращена візуалізація та звітність: Розробити інтуїтивно зрозумілі та інтерактивні інструменти візуалізації для представлення сейсмічних даних у зручний для користувача спосіб. Це дозволить операторам і зацікавленим сторонам легко інтерпретувати і аналізувати інформацію, сприяючи прийняттю обґрунтованих рішень і ефективному інформуванню громадськості про сейсмічні ризики.

Налагодити інтеграцію між системою сейсмічного моніторингу та існуючими системами реагування на надзвичайні ситуації, такими як управління надзвичайними ситуаціями та системи раннього попередження. Така інтеграція забезпечить ефективну координацію та реагування під час сейсмічних подій, що призведе до покращення готовності, своєчасної евакуації та ефективного розгортання ресурсів.

Впроваджувати ініціативи, спрямовані на підвищення обізнаності та розуміння громадськістю сейсмічної активності та її потенційних ризиків. Це може включати освітні кампанії, семінари та інформаційно-просвітницькі програми для поширення інформації про готовність до землетрусів, заходи безпеки та стратегії реагування. Підвищення рівня обізнаності та готовності населення сприятиме мінімізації впливу сейсмічних подій на життя людей та інфраструктуру.

Постійне оцінювання та вдосконалення системи: Регулярно оцінюйте результативність та ефективність системи моніторингу за допомогою процесів моніторингу та оцінки. Це допоможе визначити сфери для вдосконалення, вирішити будь-які обмеження або проблеми, а також впровадити технологічні досягнення, щоб система залишалася надійною та сучасною.

Реалізація цих пропозицій може сприяти подальшому вдосконаленню системи моніторингу регіональної сейсмічної активності, забезпечуючи більш точний і надійний збір, аналіз і поширення даних. Ці вдосконалення в кінцевому підсумку сприятимуть прийняттю більш обґрунтованих рішень, покращенню реагування на стихійні лиха та підвищенню стійкості до сейсмічної небезпеки.

### Перелік джерел

1. Джон Скит. С# для професіоналів: тонкощі програмування, 3-тє видання, новий переклад = C# in Depth, 3rd ed. - М.: «Вільямс», 2014. - 608 с. - ISBN 978-5-8459-1909-0.
2. Наконечний, М. В., and Ю. М. Наконечний. "ОСОБЛИВОСТІ ІДЕНТИФІКАЦІЇ ДИНАМІЧНИХ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ РЕКУРЕНТНИХ НЕЙРОННИХ МЕРЕЖ." Л.: НУЛПІ (2009)
3. А. Хейлсберг, М. Торгерсен, С. Вілтамут, П. Голд. Мова програмування С#. Класика Computers Science. 4-тє видання = C# Programming Language (Covering C# 4.0), 4th Ed. - СПб.: "Пітер", 2012. - 784 с. - ISBN 978-5-459-00283-6. Архівна копія від 10 жовтня 2011 року на Wayback Machine
4. Lee, Jangsoo, et al. "An Open Dataset for Deep Learning-based Earthquake Detection using MEMS Sensors." 2022 IEEE International Conference on Big Data (Big Data). IEEE, 2022.
5. Ендрю Троелсен. Мова програмування С# 5.0 і платформа .NET 4.5, 6-тє видання = Pro C# 5.0 і .NET 4.5 Framework, 6th edition. - М.: "Вільямс", 2013. - 1312 с. - ISBN 978-5-8459-1814-7.
6. Wiszniowski, J., Plesiewicz, B.M. & Trojanowski, J. Application of real time recurrent neural network for detection of small natural earthquakes in Poland. Acta Geophys. 62, 469–485 (2014).
7. Анфімова, Г. В. Актуальність і перспективи створення технічних засобів моніторингу сейсмічної активності в проблемних регіонах. Науковопрактична конференція «Безпека енергетики в епоху цифрової трансформації», ІПМЕ ім. ГЕ Пухова НАН України, Київ, 20.12. 2019р. С36-38.
8. Агазаде, Т., Тютюник, В., Черногор, Л., & Тютюник, О. (2022). ОСОБЛИВОСТІ ПІДТРИМКИ ПРИЙНЯТТЯ АНТИКРИЗОВИХ РІШЕНЬ В УМОВАХ ВИНИКНЕННЯ ГЕОФІЗИЧНИХ НАДЗВИЧАЙНИХ СИТУАЦІЙ. Науковий вісник: Цивільний захист та пожежна безпека, (2 (14)), 65-79.