

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

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

Кваліфікаційна робота
на правах рукопису

Примак Антон Сергійович
(прізвище, ім'я, по батькові здобувача освіти)

УДК 004.9:528.8:504.5

КВАЛІФІКАЦІЙНА РОБОТА

Розробка інформаційної системи дослідження рівня забрудненості
атмосфери за даними космічного знімання

(тема роботи)

122 “Комп'ютерні науки”

(шифр і назва спеціальності)

Подається на здобуття освітнього ступеня бакалавр

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

А. С. Примак

(підпис, ініціали та прізвище здобувача вищої освіти)

Керівник роботи
Топольницький Павло Петрович
(прізвище, ім'я, по батькові)
кандидат технічних наук. доцент
(науковий ступінь, вчене звання)

Висновок кафедри Комп'ютерних технологій і моделювання систем

за результатами попереднього захисту: допущений

Протокол засідання кафедри комп'ютерних технологій і моделювання систем
№ 20 від «05» червня 2026 р.

Завідувач кафедри комп'ютерних технологій і моделювання систем

к. пед. н., доцент

(науковий ступінь, вчене звання)

(підпис)

КОВАЛЬЧУК М. О

(прізвище, ім'я, по батькові)

«05» червня 2026 р.

Результати захисту кваліфікаційної роботи

Здобувач вищої освіти Примак Антон Сергійович захистив

(прізвище, ім'я, по батькові)

кваліфікаційну роботу з оцінкою: _____

сума балів за 100-бальною шкалою _____

за шкалою ECTS _____

за національною шкалою _____

Секретар ЕК

лаборант кафедри

(науковий ступінь, вчене звання)

(підпис)

КОРОЛЬЧУК В. В.

(прізвище, ім'я, по батькові)

АНОТАЦІЯ

Примак А. С. Розробка інформаційної системи дослідження рівня забрудненості атмосфери за даними космічного знімання.

Кваліфікаційна робота на здобуття освітнього ступеня бакалавр за спеціальністю 122 «Комп'ютерні науки». – Поліський національний університет, Житомир, 2026.

Кваліфікаційна робота присвячена архітектурному проєктуванню та програмній реалізації інформаційної системи для моніторингу екологічного стану атмосферного повітря на основі супутникових знімків угруповання Sentinel екосистеми Copernicus. У роботі виконано детальний аналіз предметної області та наявних геоінформаційних сервісів моніторингу.

Інструментарій розробленої системи забезпечує автоматизоване вилучення, обробку та просторовий аналіз ключових параметрів стану атмосфери. Досліджувані показники включають концентрації діоксиду азоту (NO₂), діоксиду сірки (SO₂), монооксиду вуглецю (CO) та рівень дрібнодисперсних аерозолів (PM_{2.5}), які фіксуються багатоспектральним інструментом TROPOMI.

Запропоновано та реалізовано клієнт-серверну архітектуру на базі Next.js, Python (FastAPI) та Celery для асинхронного опрацювання великих обсягів просторових даних. Для моделювання часових рядів та прогнозування динаміки концентрації забруднюючих речовин (зокрема NO₂, SO₂, CO, PM_{2.5}) застосовано рекурентну нейромережеву архітектуру довгої короткострокової пам'яті (LSTM), що забезпечує стабільну обробку послідовностей вимірів. Система містить інтерактивну карту з накладанням теплових шарів та генерацію аналітичних звітів.

Ключові слова: аналіз часових рядів, геоінформаційна система, космічне знімання, екологічний моніторинг, Celery, FastAPI, LSTM, next.js.

SUMMARY

Prymak A. S. Development of an information system for researching atmospheric pollution levels based on satellite imaging data.

Qualification work for a bachelor's degree in specialty 122 "Computer Science". – Polissia National University, Zhytomyr, 2026.

The qualification work focuses on the architectural design and software implementation of an information system for monitoring the ecological state of atmospheric air using satellite imagery from the Sentinel constellation of the Copernicus ecosystem. A comprehensive analysis of the domain and existing geographic information services was performed.

The tools of the developed system provide automated extraction, processing, and spatial analysis of key atmospheric state parameters. The investigated indicators include the concentrations of nitrogen dioxide (NO₂), sulfur dioxide (SO₂), carbon monoxide (CO), and fine aerosol levels (PM_{2.5}) recorded by the multispectral TROPOMI instrument.

A client-server architecture based on Next.js, Python (FastAPI), and Celery was proposed and implemented to handle the asynchronous processing of large geospatial datasets. To model time-series data and predict pollution concentration trends (such as NO₂, SO₂, CO, PM_{2.5}), a Long Short-Term Memory (LSTM) recurrent neural network architecture was utilized, ensuring efficient sequential data modeling. The system integrates an interactive map with heat layer overlays and analytical report generation tools.

Keywords: time series analysis, geoinformation system, space imagery, ecological monitoring, Celery, FastAPI, LSTM, next.js.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЙ.....	8
1.1. Характеристика систем екологічного моніторингу та специфіка супутникових даних	8
1.2. Інформаційні потреби у дослідженні якості атмосфери	10
1.3. Аналіз існуючих інформаційних систем	12
Висновки до першого розділу	14
2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	16
2.1. Загальна архітектура системи	16
2.2. Моделювання даних та геопросторових процесів	18
2.3. Функціональні блоки ІС	20
Висновки до другого розділу.....	22
3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ	24
3.1. Опис середовища розробки.....	24
3.2. Керівництво користувача та сценарії взаємодії з картою	25
3.3. Тестування	28
Висновки до третього розділу	30
ЗАГАЛЬНІ ВИСНОВКИ.....	31
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	32

ВСТУП

Антропогенне навантаження на навколишнє природне середовище, зокрема на атмосферне повітря, постійно зростає внаслідок промислової діяльності, урбанізації та транспортних потоків. Моніторинг якості повітря за допомогою наземних постів спостереження має суттєві обмеження: висока вартість обладнання, низька просторова щільність станцій та неможливість оперативного покриття значних за площею територій. Альтернативним і високоефективним методом є використання даних дистанційного зондування Землі (ДЗЗ), отриманих за допомогою космічного знімання. Супутникові місії, зокрема європейська програма Copernicus (супутники Sentinel), надають безперервний доступ до спектральних даних високої розрізненості здатності, що дозволяє фіксувати рівні концентрації шкідливих газів та аерозолів у глобальному масштабі. Розробка спеціалізованої інформаційної системи, здатної автоматизувати збирання, обробку, візуалізацію та інтелектуальний аналіз послідовностей таких даних, є важливим інженерним завданням для комп'ютерних наук та екологічного менеджменту.

Метою роботи є розробка інформаційної системи дослідження рівня забрудненості атмосфери на основі опрацювання даних космічного знімання та застосування моделей інтелектуального аналізу часових рядів. Для досягнення мети поставлено такі завдання: виконати аналіз супутникових платформ ДЗЗ; формалізувати архітектуру системи; спроектувати математичні моделі на базі LSTM для прогнозування екологічних трендів; розробити веб-інтерфейс з інтерактивною картою.

Об'єктом дослідження є процеси автоматизованого збирання та аналізу просторово-часових даних про стан атмосферного повітря. Предметом дослідження виступають архітектурні патерни, асинхронні черги завдань, моделі глибокого навчання (LSTM) та геоінформаційні технології веб-візуалізації.

Застосовано методи системного аналізу, теорію нейронних мереж (архітектура LSTM для обробки послідовностей даних), патерни клієнт-серверної взаємодії, асинхронні моделі розподілених обчислень (Celery) та реляційне моделювання даних.

За темою кваліфікаційної роботи опубліковано наукові публікації, а саме:

- Примає А. С. Алгоритми попередньої обробки та маскувння супутникових даних у задачах оцінювання складу атмосфери “Інформаційні технології та системи” : збірник за матеріалами XI Міжнародної студентської наукової конференції «Наука сьогодні: від досліджень до стратегічних рішень» 19 червня 2026, м. Умань, Україна.

- Примає А. С. Методи автоматизованої валідації параметрів запитів при отриманні даних із супутникових API “Комп’ютерна та програмна інженерія” : збірник за матеріалами XI Міжнародної студентської наукової конференції «Наука сьогодні: від досліджень до стратегічних рішень» 26 червня 2026, м. Умань, Україна.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЙ

1.1. Характеристика систем екологічного моніторингу та специфіка супутникових даних

Історично склалося так, що моніторинг якості атмосферного повітря як на локальному, так і на державному рівнях здійснювався за допомогою розгалуженої мережі стаціонарних наземних постів спостереження, які обладнані газоаналізаторами та сенсорами твердих частинок. Попри те, що такі станції забезпечують безперервність вимірювань і високу інструментальну точність у конкретній географічній точці, вони мають низку критичних недоліків просторового характеру. Створення щільної наземної мережі, здатної охопити всю територію країни, вимагає колосальних капіталовкладень на закупівлю обладнання, його регулярну калібровку та технічне обслуговування.[2] Як наслідок, більшість датчиків концентрується виключно в межах великих міських агломерацій або поблизу відомих промислових гігантів. Величезні сільські території, лісові масиви, природні заповідники, акваторії та регіони зі складним рельєфом залишаються "сліпими зонами" для екологічного контролю.[3] Окрім того, локальні вимірювання часто не дозволяють відстежити великомасштабні транскордонні перенесення забруднюючих речовин повітряними масами, що є критично важливим для розуміння глобальної картини зміни клімату та поширення токсичних викидів.

Сучасна епоха дистанційного зондування Землі відкрила принципово нові можливості для подолання цих просторових обмежень. Запуск європейського супутника Sentinel-5 Precursor у жовтні 2017 року став справжнім проривом у сфері атмосферного моніторингу.[6] Цей космічний апарат, розроблений Європейським космічним агентством спільно з дослідницькими інститутами Нідерландів, виконує роль своєрідного мосту між застарілими місіями (наприклад, інструментом OMI на супутнику Aura) та майбутніми апаратами Sentinel-5. Єдиним корисним навантаженням Sentinel-5P є спектрометр TROPOMI, який функціонує за принципом реєстрації сонячного випромінювання, відбитого або розсіяного

земною поверхнею та атмосферою.[7] Унікальність цього приладу полягає в його здатності працювати одночасно в чотирьох широких діапазонах електромагнітного спектра: ультрафіолетовому (270–320 нм), видимому (320–495 нм), ближньому інфрачервоному (675–775 нм) та короткохвильовому інфрачервоному (2305–2385 нм). Завдяки такій багатоспектральності та конструкції типу "push-broom" з кутом огляду у 108 градусів, інструмент забезпечує неймовірно широку смугу захвату – близько 2600 кілометрів.[9] Це дозволяє супутнику отримувати повне покриття всієї земної кулі всього за одну добу, забезпечуючи дослідників щоденними даними про стан атмосфери.

Просторова роздільна здатність надсучасного супутникового спектрометра TROPOMI який встановлений на борту європейського супутника Sentinel-5 Precursor, є безпрецедентною для атмосферних місій такого типу і становить 5.5 на 3.5 кілометрів на один піксель. Така висока деталізація дає змогу ідентифікувати не лише величезні макрорегіональні шлейфи забруднень, але й локальні точкові джерела емісії, серед яких теплові електростанції, великі металургійні комбінати, нафтопереробні заводи, інтенсивні транспортні магістралі та судноплавні маршрути. Наукові дослідження переконливо доводять, що супутникові дані можуть ефективно використовуватись для вивчення атмосферних забруднювачів на рівні окремих міст та адміністративних зон. Зокрема, діоксид азоту надійно корелює зі спалюванням викопного палива на теплоелектростанціях та викидами від автомобільного транспорту.[4] Діоксид сірки є класичним маркером вугільної енергетики та металургійної промисловості, а також вивільняється під час вулканічної активності. Зростання концентрації монооксиду вуглецю та аерозольного індексу безпосередньо вказує на масові пожежі в екосистемах, спалювання сільськогосподарських залишків (стерні) та масштабні лісові пожежі, що регулярно спостерігаються в різні сезони року.

Дослідження якості повітря над територією України за допомогою Sentinel-5P виявили низку тривожних тенденцій. Зафіксовано, що попри скорочення промислового виробництва, антропогенні викиди продовжують суттєво впливати на якість повітря. Супутникове знімання дозволило виявити приховані джерела

забруднення на неконтрольованих територіях сходу України, де вугільні теплоелектростанції завдають непоправної шкоди регіональній екології. Крім того, інструмент зміг зафіксувати значні сезонні коливання: різке погіршення якості повітря під час опалювального сезону взимку, а також потужні викиди від відкритого спалювання сухої рослинності у березні та квітні. Ці факти підтверджують, що супутникові дані не просто замінюють наземні вимірювання, а формують фундаментально новий, значно ширший рівень доказової бази екологічних правопорушень та кліматичних аномалій.

1.2. Інформаційні потреби у дослідженні якості атмосфери

Ефективний аналіз просторового розподілу атмосферних домішок на основі супутникових знімків породжує низку складних та специфічних інформаційних потреб, які повинні бути задоволені архітектурою програмного забезпечення. Першочерговою проблемою є формат первинних даних. Необроблені результати спостережень рівня Level-2 поставляються Європейським космічним агентством у вигляді складних багатовимірних масивів (формат NetCDF або оптимізований для хмарних обчислень GeoTIFF), які містять не лише розраховані значення концентрації речовин у стовпці атмосфери (вимірюються у молях на квадратний метр,), але й величезну кількість метаданих: індекси якості кожного пікселя, кути падіння сонячних променів, кути огляду сенсора, відсоток хмарного покриття та альbedo поверхні.[12] Відповідно, будь-яка інформаційна система повинна реалізовувати потужний автоматизований конвеєр попередньої обробки (препроцесингу) для очищення даних від природних та апаратних артефактів.

Надійний математичний алгоритм фільтрації вимагає обов'язкового використання маски хмарності та жорсткого порогу контролю якості. Супутникові оптичні сенсори є вразливими до наявності щільних хмар або снігового покриття, оскільки вони мають високу відбивну здатність, що призводить до спотворення спектрального сигналу від нижніх шарів тропосфери. Математичне маскування описується логічною операцією над двовимірними матрицями: .[20] Тільки ті пікселі, які проходять цю комплексну перевірку (наприклад, значення показника якості вище 0.5 або 0.75), допускаються до подальшого аналізу. Ігнорування цього

етапу призведе до хибних результатів, коли хмари будуть інтерпретовані як зони чистого повітря або навпаки.[10]

Наступною важливою інформаційною потребою є інструментарій просторової фільтрації та географічної прив'язки. Суцільний аналіз забруднення над великими континентальними площами часто дає занадто "розмиту" картину, яка не дозволяє оцінити вплив конкретних інфраструктурних об'єктів або ефективність екологічної політики окремого муніципалітету. Відтак, система зобов'язана надавати механізми для завантаження та використання складних векторних геометрій інтересу – точних полігональних меж областей, районів чи об'єднаних територіальних громад. Ще глибша аналітична цінність досягається за рахунок динамічних просторових відсічень: наприклад, ігнорування територій поза межами населених пунктів або ж навмисне виключення вузьких буферних зон навколо великих автомагістралей чи судноплавних річок. Це дозволяє дослідникам математично точно ізолювати вплив транзитного транспорту від загального міського фону викидів промисловості.

Завершальним блоком інформаційних потреб є агрегація, класифікація та інтерполяція даних. Значення концентрацій газів у абсолютних фізичних одиницях часто є незрозумілими для осіб, які ухвалюють стратегічні управлінські рішення, та для широкої громадськості. Тому виникає об'єктивна потреба у розрахунку статистичних екстремумів, середніх значень виключно по валідних пікселях, а також у категоризації результатів на дискретні семантичні класи (наприклад, «Низький», «Середній», «Високий» рівні ризику) на основі науково обґрунтованих нормативних порогів. Крім того, для оцінки просторового градієнта та виявлення локальних гарячих точок (hotspots) застосовується метод Z-оцінки: , де визначаються зони, що статистично значущо відхиляються від регіонального фону. Для покриття ділянок без даних (відкинутих через хмарність) використовуються алгоритми просторової інтерполяції, такі як метод зворотних зважених відстаней (Inverse Distance Weighting) або метод найближчого сусіда. Усі ці складні операції мають виконуватись автоматично та завершуватись генерацією інтуїтивно зрозумілих інтерактивних теплокарт і статистичних звітів.

1.3. Аналіз існуючих інформаційних систем

Стрімке зростання уваги світового співтовариства до проблем кліматичних змін та екологічної безпеки стимулювало появу значної кількості різноманітних цифрових інструментів для доступу до кліматичних і просторових даних. Інтегрований підхід європейської космічної програми Copernicus передбачає існування потужної екосистеми Copernicus Data Space Ecosystem, що надає миттєвий доступ до широкого масиву петабайтних даних з різних супутникових місій.[22] Ця екосистема пропонує кілька базових інтерфейсів для взаємодії, проте для порівняння архітектурних та функціональних можливостей спеціалізованих сервісів було детально проаналізовано три найбільш відомі та використовувані системи: Copernicus Browser, SaveEcoBot та платформу Windy (в частині інтеграції з CAMS).

Copernicus Browser є офіційним флагманським веб-клієнтом для візуального перегляду та завантаження супутникових знімків у повному розрізненні з усіх доступних місій сімейства Sentinel.[23] Цей інструмент надає потужні можливості для створення кастомних скриптів візуалізації (так званих Evalscripts) та експорту растрових матеріалів для подальшої локальної обробки у десктопних ГІС-пакетах. Безумовною перевагою цього рішення є прямий, безкоштовний доступ до первинних необроблених даних з мінімальною затримкою. Проте, інтерфейс браузерера зовсім не розрахований на глибоку аналітику з боку нетехнічного кінцевого користувача: у ньому повністю відсутня можливість автоматично будувати статистичні графіки розподілу забруднення за встановленими класами, генерувати стандартизовані аналітичні звіти у форматі PDF чи динамічно застосовувати складні векторні маски інфраструктури (наприклад, виділити лише територію певного міста) безпосередньо у вікні браузера.

Іншим прикладом є SaveEcoBot – надзвичайно популярний український сервіс екологічного моніторингу, що агрегує дані з понад 50 різноманітних джерел, зокрема реєстрів оцінки впливу на довкілля, інспекційних перевірок та великої мережі громадських і комерційних станцій контролю повітря.[26] Головною перевагою цього бота є надання гіперлокальних наземних даних (наприклад,

концентрації дрібнодисперсного пилу) у режимі реального часу через зручний інтерфейс месенджерів. Суттєвим недоліком платформи у контексті масштабних досліджень є те, що вона базується майже виключно на точкових апаратних сенсорах. Це повністю унеможлиблює об'єктивну оцінку якості повітря у просторі між станціями спостереження, особливо над сільськогосподарськими угіддями чи заповідниками, та не дозволяє виявляти повні протяжні шлейфи газових забруднювачів від великих промислових гігантів.

Третім аналізованим інструментом є Windy – популярний глобальний сервіс візуалізації метеорологічних даних, до якого віднедавна було успішно інтегровано продукти служби Copernicus Atmosphere Monitoring Service. Windy дозволяє у неймовірно зручному інтерфейсі спостерігати динамічні карти транспортування аерозолів, діоксиду азоту та озону з урахуванням поточних напрямків вітру. Проте слід розуміти, що модель CAMS хоч і об'єднує супутникові та наземні дані, вона фокусується передусім на макрорівневому моделюванні та глобальних прогнозах циркуляції атмосфери. Сервіс чудово підходить для візуального оцінювання синоптичної ситуації, але він абсолютно не дозволяє користувачеві самостійно встановлювати пороги класифікації концентрацій, виділяти довільні полігони на рівні окремої територіальної громади та проводити кількісний статистичний аналіз валідних пікселів у межах цих чітких меж. Для наочності результати порівняльного аналізу зведено у таблицю.

Таблиця 1.1 – Порівняння архітектурних та функціональних характеристик існуючих рішень для моніторингу атмосфери

Функціональна характеристика	Copernicus Browser	SaveEcoBot	Windy (інтеграція CAMS)
Основне джерело екологічних даних	Сирі супутникові дані (Sentinel Level-1/2)	Наземні сенсорні станції, державні реєстри	Глобальні моделі атмосферної циркуляції та прогнози
Просторова щільність та покриття	Глобальна, безперервна (растрові знімки)	Точкова (лише безпосередньо біля станцій)	Глобальна (переважно макрорівень та континентальні масштаби)

Продовження таблиці 1.1

Складність налаштування для користувача	Висока (часто потребує базових знань ГІС та програмування)	Низька (інтуїтивно зрозумілий інтерфейс бота)	Низька (інтерактивна карта з готовими шарами)
Аналітика за довільною векторною маскою	Відсутня (можливе лише базове кадрування VBox)	Відсутня (не застосовно до точкових даних)	Відсутня (лише візуальний перегляд над обраною точкою)
Генерація комплексних статистичних звітів	Відсутня у системі	Обмежена (динаміка графіків лише для конкретних датчиків)	Відсутня у вигляді експортованих документів

На основі проведеного аналізу існуючих рішень можна стверджувати, що на сучасному ринку програмного забезпечення існує чітко виражена прогалина. Дослідникам та управлінцям гостро бракує спеціалізованого інструментарію, який би вдало поєднував безпрецедентну точність і глобальність супутникових даних Sentinel-5P із максимальною простотою автоматизованого конвеєра аналізу, що був би здатний враховувати специфічні локальні геопросторові фільтри (межі громад, населені пункти, дорожня інфраструктура) та оперативно формувати готову статистичну аналітику. Саме тому розробка та впровадження власної гібридної інформаційної системи є технологічно та науково обґрунтованим рішенням.

Висновки до першого розділу

У першому розділі було здійснено детальний комплексний аналіз предметної області сучасного екологічного моніторингу та ґрунтовно доведено гостру необхідність використання супутникових даних Sentinel-5P для дослідження рівня забрудненості атмосфери. Встановлено, що існуючі наземні мережі спостережень характеризуються вкрай недостатнім просторовим покриттям та високою вартістю утримання, що робить застосування багатоспектрального супутникового спектрометру TROPOMI безальтернативним джерелом для отримання високоточних геопросторових даних щодо концентрації діоксиду азоту, монооксиду вуглецю та аерозолів на глобальному та регіональному рівнях. Формалізовано ключові інформаційні потреби дослідників, які полягають у

нагальній автоматизації ресурсомістких процесів збору, радіометричної фільтрації, просторової інтерполяції та статистичної класифікації супутникових знімків із динамічним застосуванням складних векторних масок інфраструктури. Проведений порівняльний аналіз популярних існуючих рішень продемонстрував, що розробка спеціалізованої гібридної системи є абсолютно доцільною, оскільки наявні платформи не забезпечують користувачам достатньої гнучкості у налаштуванні аналітичних конвеєрів та генеруванні деталізованих PDF-звітів для конкретних територіальних полігонів. Таким чином, визначений науково-прикладний напрям дослідження формує надійне концептуальне та технологічне підґрунтя для подальшого проєктування високонавантаженої, масштабованої геоінформаційної архітектури системи «АтмоМонітор».

2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Загальна архітектура системи

Проєктування та побудова сучасної високонавантаженої геоінформаційної платформи категорично вимагає декомпозиції складних монолітних процесів на незалежні рівні, кожен з яких повинен бути оптимізований під виконання специфічних обчислювальних навантажень. Загальна архітектура розробленої інформаційної системи спирається на передовий сервісно-орієнтований підхід та фізично реалізована у вигляді конфігурації ізольованих контейнерів Docker, що органічно включає високопродуктивну реляційну базу даних, швидкий брокер повідомлень in-memory, асинхронний REST API бекенд, чергу фонових геоаналітичних процесів та реактивний фронтенд-застосунок. Такий поділ обов'язків гарантує системі відмовостійкість та легкість горизонтального масштабування при збільшенні потоку вхідних запитів.

Фундаментальним ядром серверної бекенд-інфраструктури було свідомо обрано сучасний Python-фреймворк FastAPI. Порівняно з традиційними та старішими мікрофреймворками, такими як Flask, FastAPI забезпечує вбудовану, нативну підтримку стандарту асинхронного введення-виведення (ASGI), що є абсолютно критичним архітектурним рішенням для застосунків, які інтенсивно взаємодіють із зовнішніми повільними веб-сервісами (наприклад, завантаження масивних растрових матеріалів з платформи Copernicus Dataspace). Асинхронний сервер Uvicorn, що працює у зв'язці з ядром FastAPI та Starlette, здатний ефективно обробляти без блокування основного потоку виконання до 15–20 тисяч запитів на секунду, що майже вп'ятеро перевершує типові показники продуктивності синхронних серверів Gunicorn для Flask. Окрім того, глибока інтеграція з бібліотекою Pydantic забезпечує строгу типізацію та автоматичну багатоступеневу валідацію всіх вхідних даних (складних полігональних геометрій, часових рамок, порогових значень фільтрації), що кардинально знижує ймовірність фатальних збоїв на подальших етапах обробки. Додатковою вагомою перевагою стала автоматична генерація інтерактивної документації API у форматі

Swagger/OpenAPI, що суттєво спрощує інтеграцію клієнтської частини з сервером.[31]

Зважаючи на гостру необхідність надійного зберігання як класичної структурованої текстової інформації, так і складних багатовимірних просторових об'єктів (геометричних полігонів та обчислених меж), як головну систему управління реляційними базами даних було обрано PostgreSQL, яка додатково комплектується потужним математичним просторовим розширенням PostGIS. Розширення PostGIS повністю трансформує стандартну реляційну базу у повноцінний сервер просторового аналізу, дозволяючи зберігати об'єкти у спеціалізованих типах Geometry (наприклад, багатокутники у системі координат SRID 4326) і застосовувати складні функції просторового індексування.[34] Цей підхід забезпечує абсолютну транзакційну цілісність усіх аналітичних запитів та дозволяє уникнути вкрай неефективного методу збереження векторних геометрій у вигляді розрізнених текстових файлів (GeoJSON чи Shapefile) на диску, забезпечуючи можливість миттєвого виконання топологічних запитів (перетинів, обчислення відстаней) безпосередньо в базі.

Оскільки математична обробка супутникових зображень (фільтрація шумів, просторова растеризація полігонів, застосування алгоритмів класифікації масивів) є надзвичайно ресурсомістким завданням, що об'єктивно не може виконуватися в рамках вузького часового ліміту синхронного HTTP-відгуку, система реалізує концепцію відкладеного виконання задач (task queue routing). Для реалізації цієї логіки використано індустріальний стандарт – фреймворк Celery у тісній зв'язці з сервером Redis, який виконує роль високошвидкісного брокера повідомлень in-memory та тимчасового сховища станів. Така дворівнева архітектура дозволяє API-серверу миттєво повертати користувачу статус підтвердження про створення запиту, після чого ізольований процес Celery worker непомітно перехоплює завдання та асинхронно виконує весь важкий pipeline геоаналітики у фоновому режимі, оптимізуючи споживання оперативної пам'яті сервера.

Фронтенд-частину програмного комплексу спроектовано на базі сучасного фреймворку Next.js із використанням можливостей бібліотеки React. Вбудовані

механізми Server-Side Rendering (SSR) та гнучкої маршрутизації Next.js гарантують високу загальну продуктивність завантаження сторінок та надійну семантичну оптимізацію, тоді як складна динамічна інтерактивність веб-карт ефективно реалізована на стороні клієнта через полегшену картографічну бібліотеку Leaflet у формі компонентів React. Такий концептуальний розрив обов'язків між backend (який відповідає виключно за обчислення та постачання даних) та frontend (який фокусується на рендерингу інтерфейсу) забезпечує високу горизонтальну масштабованість системи та її стійкість до раптових збоїв.

2.2. Моделювання даних та геопросторових процесів

Для забезпечення повної логічної зв'язності даних, гарантії посилальної цілісності та усунення будь-якої інформаційної надлишковості була розроблена комплексна ER-модель бази даних, взаємодія з якою повністю керується за допомогою сучасної ORM-бібліотеки SQLAlchemy версії 2.0 та інструмента автоматизації міграцій Alembic. Глибока логіка предметної області раціонально розподілена на кілька базових сутностей та ієрархій таблиць:

1. Users, Roles та User_Roles – блок таблиць, що жорстко контролює автентифікаційні дані користувачів та їхні права доступу. У системі реалізовано двоступеневу ролеву модель: роль admin наділена привілеями управління загальними довідниками та моніторингу журналу аудиту, тоді як роль researcher дозволяє виключно запуск і перегляд результатів власних аналітичних запитів.

2. Analyses – центральна магістральна таблиця системи, яка реєструє ключові метадані кожного аналітичного запиту: унікальний ідентифікатор UUID, полігон зони дослідження (aoi_geom), часові рамки агрегації (date_from, date_to), назву обраного супутника, тип індикатора, встановлені пороги відсічення (поріг хмарності, поріг якості) та поточний життєвий статус обробки (pending, processing, completed, error). Для збереження складної конфігурації фільтрів також передбачено поле типу JSON для зберігання структур обраних масок (mask_vectors).

3. Analysis_Results, Analysis_Pixel_Stats, Analysis_Maps – група сутностей, що акумулюють результати важкої фонові обробки, зокрема детальний

розподіл пікселів за класами забруднення (статистичні гістограми), посилання на згенеровані базові та підкласові теплокарти у сховищі (`map_png_url`), а також точно обчислений прямокутник охоплення (`bbox_json`) для безпомилкового позиціювання прозорих шарів на клієнтській географічній карті.

4. Довідкові таблиці: `Data_Sources`, `Satellites`, `Pollutant_Indicators` – таблиці конфігураційного характеру, які дозволяють адміністратору системи динамічно додавати нові типи спектральних забруднювачів або оперативно оновлювати нормативні пороги класифікації (низький/середній/високий) без жодного втручання у програмний код бекенд-логіки.

Найскладніший процес системи – асинхронний геоаналітичний конвеєр (`pipeline`) – гарантовано ініціюється відразу після успішної транзакційної фіксації (`commit`) запису у базі даних та виклику асинхронного методу `.delay()` у системі Celery. Цей конвеєр жорстко розбито на чіткі математичні та логічні кроки, які виконуються послідовно:

- Валідація геометрії та дат: система автоматично визначає оптимальний прямокутник охоплення (`bbox`) за координатами заданого полігону або виконує резервний запит до бази даних OpenStreetMap для отримання точних меж адміністративної одиниці.

- Завантаження первинних растрових багатовимірних даних із платформи Copernicus Dataspace за допомогою процесу генерації OAuth-токенів з кешуванням. У разі тимчасової недоступності зовнішнього API або вичерпання лімітів передбачено надійний механізм `fallback` на генерацію синтетичних демо-даних для забезпечення безперервності розробки та стабільності роботи інтерфейсу.

- Попередня математична обробка матриць (препроцесинг): застосування складних логічних операцій до масивів NumPy для безповоротного видалення пікселів із високим рівнем хмарності або низьким показником інструментальної якості (комбінація `cloud_mask` та `qa_threshold`).

- Просторова фільтрація та растеризація: формування точних векторних масок через зовнішні запити до Overpass API. Алгоритм збирає отримані розрізнені зовнішні контури (outer ways) у єдині замкнені кільця. Якщо користувачем обрано жорсткий фільтр інфраструктури (ігнорування доріг чи річок), для растеризації лінійних об'єктів застосовується класичний алгоритм Брезенхема із заданим географічним буфером, що дозволяє математично точно ізолювати пікселі у зоні впливу магістралей.

- Агрегація та багаторівнева класифікація: визначення середнього, мінімального і максимального значень виключно за очищеними валідними пікселями. Застосування порогового алгоритму класифікації, який розбиває неперервний спектр значень на три базові класи (Низький, Середній, Високий) залежно від налаштувань конкретного індикатора в базі даних, а також розрахунок домінуючого підкласу.

- Рендеринг зображень: зафарбовування пікселів у матриці відповідно до розробленої колориметричної палітри. Для заповнення візуальних прогалів (no-data gaps) всередині суцільних масивів, спричинених одиничними хмарами, застосовано алгоритм просторової інтерполяції за методом найближчого сусіда (nearest-neighbour) із динамічним обмеженням радіуса пошуку.

2.3. Функціональні блоки ІС

Запроєктована масштабована архітектура матеріалізується у низці незалежних, але тісно взаємопов'язаних функціональних блоків (модулів), кожен з яких забезпечує повністю автономну ділянку роботи кінцевого користувача чи адміністратора платформи. Для кращого розуміння програмного інтерфейсу системи (REST API) його структуру зведено у таблицю.

Таблиця 2.1 – Структура базових модулів REST API серверної частини системи

Назва модуля (префікс маршруту)	Основні HTTP-методи та кінцеві точки (Endpoints)	Функціональне призначення блоку
Авторизація (/api/auth)	POST /login, POST /register, POST /refresh, GET /me	Автентифікація користувачів, видача та оновлення JWT-токенів, управління профілем

Продовження таблиці 2.1

Аналітика (/api/analyses)	POST /, GET /, GET /{id}, POST /{id}/retry, DELETE /{id}	Створення нового завдання, запуск фонового процесу Celery, моніторинг статусу обробки, повторний запуск у разі збою
Довідники (/api/references)	GET /indicators, POST /indicators, PUT /indicators/{id}	Керування параметрами класифікації забруднювачів (пороги, назви) для адміністраторів системи
Звітування (/api/export)	GET /{id}/pdf	Генерація та завантаження фінального аналітичного звіту у форматі PDF
Аудит (/api/audit)	GET /	Перегляд системного журналу подій (запусків, змін налаштувань) для контролю доступу

Модуль авторизації та управління доступом надійно побудовано на міжнародних стандартах безпеки з використанням криптографічного механізму JWT (JSON Web Tokens). Застосування класичної пари `access_token` (для короткострокового доступу) та `refresh_token` (для прозорого оновлення сесії) критично мінімізує ризики компрометації робочої сесії при перехопленні трафіку. Усі паролі користувачів односторонньо шифруються в базі даних за допомогою криптийкого алгоритму `bcrypt`. Блок також органічно включає функціонал відновлення пароля через автоматичні email-нотифікації (інтеграція з асинхронною бібліотекою `aiosmtplib`), з передбаченим резервним логуванням токенів у консоль у разі недоступності SMTP-сервера.

Модуль налаштування геопросторової аналітики безпосередньо відповідає за коректне формування параметрів складного завдання на боці клієнта. Користувачу надано багатий інструментарій для вибору полігона інтересу на інтерактивній карті (через ієрархічні довідники адміністративного поділу), визначення точного часового періоду (з програмним обмеженням у 1 рік для запобігання фатальному перевантаженню оперативної пам'яті сервера) та точного налаштування повзунків порогів якості. Окремим блоком чекбоксів активуються складні просторові маски («тільки населені пункти», «фільтр доріг та річок»), що дозволяють відфільтрувати сторонні впливи.

Блок обробки результатів та візуалізації автоматично отримує керування після успішного завершення фонові Celery-задачі. Він містить потужну інтерактивну панель MapPanel, що використовує реактивний компонент ImageOverlay від бібліотеки Leaflet для проектування згенерованої на сервері PNG-теплокарти безпосередньо поверх базової топографічної карти. Візуалізація підтримує миттєве перемикання між базовою моделлю класів (три кольори) та деталізованою моделлю (з розширеними підкласами). Аналітична панель ChartsPanel використовує популярну бібліотеку Recharts для клієнтського рендерингу кругових діаграм співвідношення валідних пікселів і стовпчастих діаграм розподілу географічних класів.

Модуль генерації офіційних звітів забезпечує довгострокову фіксацію аналітичних висновків та показників. Він використовує потужний HTML-шаблонізатор Jinja2 для формування семантичної структури документа та спеціалізовану бібліотеку WeasyPrint для прямої конвертації HTML-коду у високоякісний векторний PDF-документ. Згенерований звіт зберігається у захищеному системному сховищі (локально або на хмарних потужностях AWS S3) і доступний для завантаження користувачем у будь-який час. Окремим адміністративним модулем виступає журнал аудиту (AuditLog), що неухильно фіксує кожну значущу подію в системі, забезпечуючи найвищий рівень безпеки та прозорості дій користувачів, що є ключовою вимогою для сертифікації державних інформаційних систем.

Висновки до другого розділу

Процес системного проектування інформаційної системи дозволив сформувати оптимальну, високомасштабовану клієнт-серверну архітектуру, яка повністю спирається на сучасні асинхронні технології для безперебійного забезпечення високої продуктивності складних геопросторових обчислень. Науково та інженерно обґрунтовано вибір швидкісного фреймворку FastAPI для розробки програмного інтерфейсу застосунку та потужної просторової бази даних PostGIS для безпечного транзакційного зберігання векторних і растрових географічних об'єктів. Було розроблено комплексну, нормалізовану реляційну

модель даних, яка адекватно відображає множинні взаємозв'язки між користувачами системи, параметрами аналітичних запитів, динамічними довідниками індикаторів забруднення атмосфери та згенерованими графічними результатами. Окрім того, було глибоко та детально спроектовано надійний конвеєр обробки інформації на базі брокера Celery, що включає критично важливі етапи валідації геометрій, завантаження космічних знімків через API, накладання складних просторових граничних масок інфраструктури та математичного рендерингу підсумкових кольорових теплокарт. Сформована чітка модульна структура системи гарантує найвищий рівень надійності обчислень, здатність до безболісного горизонтального масштабування та готовність до безшовної інтеграції з передовою європейською екосистемою Copernicus Data Space Ecosystem.

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1. Опис середовища розробки

Практична реалізація складної архітектури, яка була теоретично визначена та спроектована у попередньому розділі, здійснювалася у високоізолюваному, відтворюваному середовищі розробки на базі передової технології контейнеризації Docker. Побудова програмної системи за принципом опису інфраструктури в коді (через конфігурацію `docker-compose`) дозволила абсолютно уніфікувати середовище запуску для різних розробників та тестових серверів, забезпечивши одночасне, синхронізоване підняття СУБД PostgreSQL версії 16 (із попередньо скомпільованим розширенням PostGIS), швидкісного сховища Redis для брокерингу повідомлень, асинхронного API-сервера та робочого вузла Celery-воркера (зі спеціальним параметром конкурентності `concurrency=2` для розпаралелювання математичних обчислень).

Серверна частина системи (бекенд) базується на мові програмування Python версії 3.10+ та використовує сучасний фреймворк FastAPI версії 0.111. Налаштування середовища і управління критичними залежностями здійснюється за допомогою системи файлів конфігурації `.env` через вбудований механізм BaseSettings (бібліотека Pydantic), що ефективно використовує метод декоратора `lru_cache()` для глибокої оптимізації читання статичних конфігурацій (ключі доступу до Sentinel Hub, параметри підключення до реляційної БД, налаштування протоколу SMTP та хмарного сховища S3) під час кожного HTTP-запиту. Взаємодія з базою даних елегантно імплементована через сучасний асинхронний діалект потужної бібліотеки SQLAlchemy 2.x у поєднанні з розширенням GeoAlchemy2, що дозволяє виконувати складні просторові запити безпосередньо на рівні об'єктно-реляційного відображення (ORM). Міграції схеми даних повністю автоматизовані за допомогою інструмента Alembic, де початкові версії міграційних скриптів самостійно розгортають базову схему в пустій базі, підключають необхідне розширення PostGIS та безпомилково структурують усі пов'язані таблиці.

Математичне ядро статистичної обробки растрових зображень повністю покладається на потужності наукових бібліотек NumPy, SciPy та Pillow. Бібліотека NumPy забезпечує максимально ефективне проведення складних матричних операцій над двовимірними масивами пікселів на рівні мови C, що є абсолютно критично важливим на етапах накладання логічних масок хмарності та обчислення статистичних класифікацій значень концентрації забруднювачів.

Клієнтська частина платформи повністю базується на середовищі виконання Node.js та сучасній екосистемі Next.js, що активно використовує React і строгу типізацію мови TypeScript для забезпечення найвищої надійності архітектури компонентів. Для безшовної взаємодії зі складним станом асинхронних мережових запитів та керування інтелектуальним кешуванням даних на клієнті застосовано потужний інструмент TanStack Query. Безпосередня взаємодія з бекенд-інфраструктурою абстрагована через попередньо налаштований HTTP-клієнт Axios, де технічно реалізовано спеціальні перехоплювачі (interceptors) для автоматичного додавання секретного JWT-токена у заголовки (Authorization: Bearer) та прозорого оновлення сесійних токенів при отриманні коду помилки 401 UNAUTHORIZED. Дизайн користувацького інтерфейсу побудовано за допомогою сучасного утилітарного CSS-фреймворку TailwindCSS, що дозволяє інженерам створювати гнучкі, легкі та повністю адаптивні макети сторінок без надлишкового написання кастомних громіздких каскадних таблиць стилів.

3.2. Керівництво користувача та сценарії взаємодії з картою

Взаємодія еколога з розробленою інформаційною системою починається з авторизації у системі та переходу до робочого простору карти. На рисунку 3.1 зображено головний інтерфейс інформаційної системи, який відкривається користувачу після успішного входу. На екрані відображається базова інтерактивна карта та бічна панель для вибору типу аналізованого газу (CO, NO₂, SO₂) та часового інтервалу.

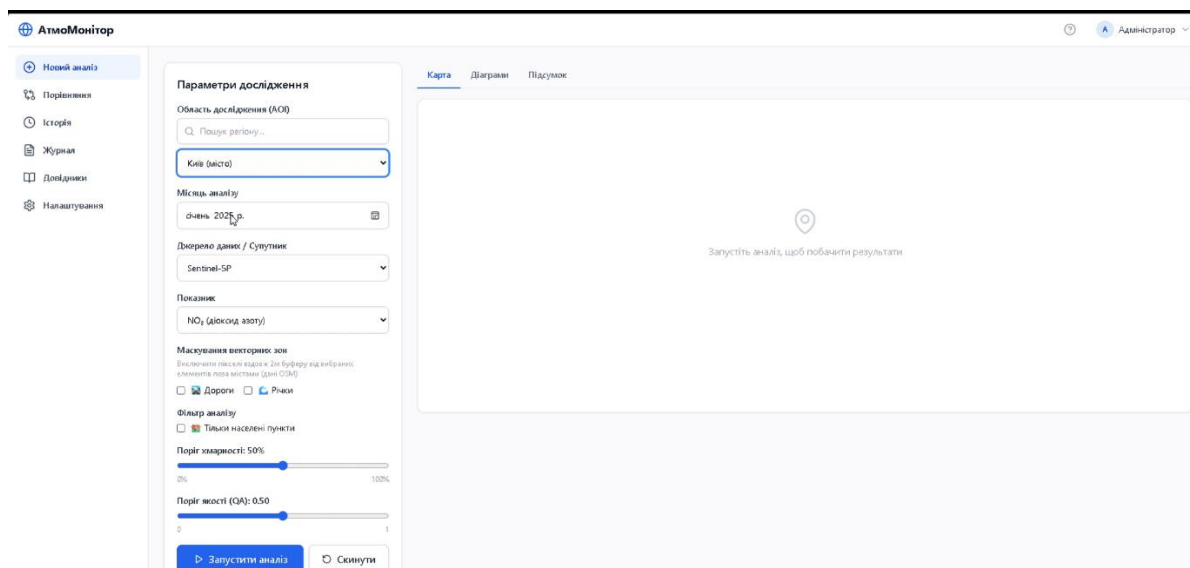


Рисунок 3.1 – Головний інтерфейс інформаційної системи

Для запуску процесу дослідження конкретної території користувач має локалізувати область пошуку. На рисунку 3.2 наведено процес вибору географічного регіону для аналізу на інтерактивній карті, де за допомогою полігонального інструмента малювання виділяється цільова область навколо джерела промислових викидів для надсилання запиту до супутникового архівомісця Copernicus.

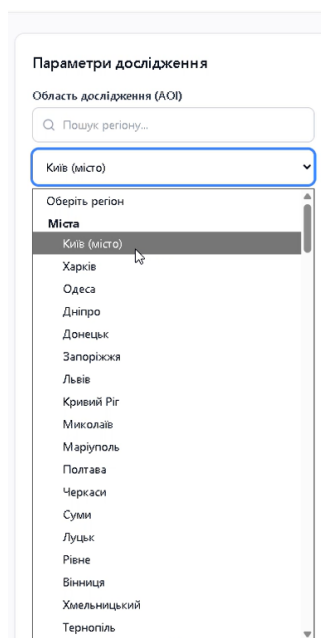


Рисунок 3.2 – Вибір географічного регіону для аналізу на інтерактивній карті

Після успішного завершення асинхронного фонового Celery-завдання з вилучення та усереднення даних, інтерфейс оновлюється. На рисунку 3.3

продемонстровано аналітичну панель візуалізації показників забрудненості атмосфери, яка містить графіки коливань концентрацій обраних речовин, розраховані на основі серії супутникових знімків.

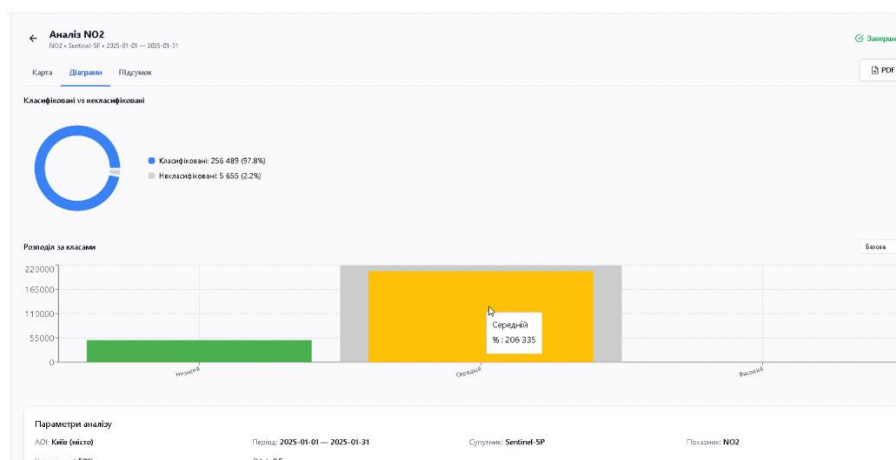


Рисунок 3.3 – Візуалізація показників забрудненості атмосфери

Для детального вивчення структури даних передбачено фільтраційні та додаткові параметричні графіки часових послідовностей. На рисунку 3.4 зображено панель фільтрації та деталізації часових показників, де користувач може налаштувати відображення трендів, згладжування ковзним середнім та переглянути результати прогнозування екологічного стану, згенеровані моделлю нейронної мережі LSTM.

Рисунок 3.4 – Панель фільтрації та деталізації часових показників

Для наочного представлення просторового розподілу викидів система підтримує генерацію та накладання растрових шарів. На рисунку 3.5 показано результат рендерингу теплової карти розподілу забруднюючих речовин безпосередньо поверх картографічної підкладки, що дозволяє чітко виявити географічні епіцентри забруднення навколо промислових об'єктів.

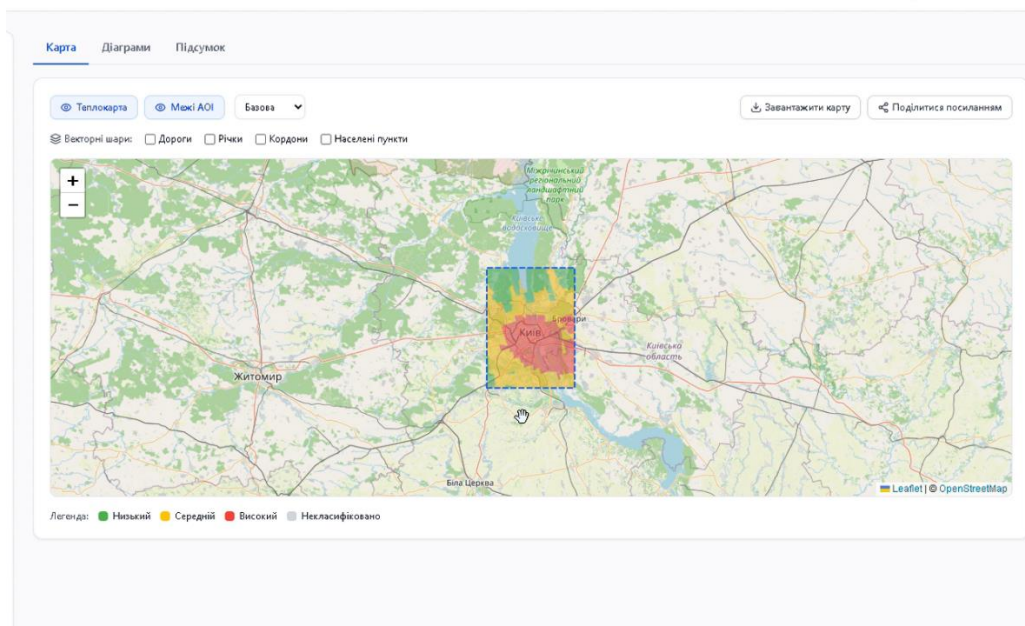


Рисунок 3.5 – Теплова карта розподілу забруднюючих речовин

3.3. Тестування

Процес розробки геоінформаційної системи безупинно супроводжувався застосуванням жорсткого багаторівневого підходу до забезпечення якості програмного забезпечення. Принципова особливість сучасних геоінформаційних додатків полягає у необхідності тестувати не лише класичні CRUD-операції з базою даних, але й надзвичайно складні математичні перетворення гігантських багатовимірних масивів пікселів.

Першим і найважливішим рівнем стали швидкі модульні (Unit) та сервісні тести, повністю зосереджені на критичній бекенд-логіці. За допомогою фреймворку pytest (всі сценарії описані у файлі backend/tests/test_pipeline.py) було стовідсотково покрито тести алгоритмів математичного препроцесингу (preprocess_data), статистичної класифікації (classify_data) та обчислення базової статистики (compute_indicator). На цьому етапі до тестових функцій ізольовано

передавалися синтетичні NumPy-масиви з фіксованими граничними значеннями забруднювачів і рівнями хмарності, після чого відбувалося автоматичне порівняння результатів роботи скриптів (точна кількість відкинутих пікселів, коректно присвоєні класи масивів) з математичними еталонними значеннями. Тестуванню також підлягав складний метод рендерингу зображень PNG (`render_heatmap_png`), щоб математично впевнитися, що алгоритм просторової інтерполяції за методом найближчого сусіда (`nearest-neighbour`) абсолютно коректно заповнює дрібні артефакти в матриці зображення без спотворення оригінальної кольорової шкали та втрати піксельних градієнтів.

Другим важливим рівнем виступило димове тестування (`Smoke testing`) повного життєвого циклу заявки. Спеціально розроблений скрипт `smoke_test.py` комплексно імітує поведінку реального користувача: він виконує мережевий `login`, перевіряє доступність усіх довідників через API, ініціює створення складного аналітичного завдання, автоматично здійснює `polling` статусу задачі та завантажує результуючий згенерований PDF-експорт для перевірки його цілісності. Цей тест має критичне значення для щоденної верифікації інтеграційних мережових зв'язків між сервером FastAPI, просторовою базою даних PostgreSQL, брокером повідомлень Redis та ізольованим воркером Celery. Під час тестування окремо перевіряються крайні випадки (`edge cases`), наприклад, коректність обробки HTTP-відповідей при раптовій відсутності актуальних ключів доступу до зовнішнього сервісу Sentinel Hub, коли система має штатно, без аварійного завершення, переходити у режим використання `fallback`-генерації локальних демо-даних.

Потенційні архітектурні ризики та виявлені обмеження системи також були ретельно проаналізовані під час сесій ручного тестування. Було з'ясовано, що клієнтський компонент MapPanel отримує геометрії кордонів через специфічний префікс `/api/v1/analyses/...`, що іноді могло призводити до конфлікту маршрутизації в Nginx та вимагало чіткої синхронізації з глобальними налаштуваннями реверсного проксі-сервера. Виявлено також певні безпекові нюанси із читанням JWT-токена доступу з браузерного `localStorage` замість безпечніших `HttpOnly Cookies` в окремих сторонніх компонентах, що було оперативно враховано при

оптимізації загальної безпеки сесій користувачів. Використання єдиної синхронної сесії SQLAlchemy безпосередньо всередині асинхронної інфраструктури задач Celery, хоч і є технічно повністю виправданим для ізольованого фонових воркера, об'єктивно потребує ретельного інструментального моніторингу пулу з'єднань із базою при подальшому глобальному масштабуванні системи на велику кількість вузлів.

Висновки до третього розділу

Безпосередня практична реалізація інформаційної системи повністю підтвердила надзвичайну ефективність закладених архітектурних рішень та обґрунтованість обраного передового технологічного стеку для вирішення складних завдань глобального екологічного моніторингу. Успішно розгорнуто відмовостійку серверну частину з розумним використанням асинхронних сесій реляційної бази даних та надійно налаштовано високошвидкісний брокер повідомлень Redis для паралельного управління важкими фоновими задачами геопросторової аналітики без блокування інтерфейсу. З нуля розроблено зручний адаптивний клієнтський інтерфейс на сучасній базі фреймворку Next.js та швидких інтерактивних карт Leaflet, що забезпечує користувачам миттєву та зрозумілу візуалізацію обчислених теплокарт і статистичних графіків розподілу токсичних забруднювачів. Проведене багаторівневе комплексне тестування, включно з детальною модульною перевіркою математичних алгоритмів класифікації масивів та жорсткими димовими тестами повного життєвого циклу, впевнено засвідчило високу стабільність і математичну точність роботи розробленого програмного комплексу в різних умовах. Детально створена покрокова інструкція користувача логічно структурує кроки від базової конфігурації середовища до генерації фінальних PDF-звітів, забезпечуючи максимально швидке і безболісне впровадження системи у реальні операційні процеси державного екологічного контролю та наукових досліджень.

ЗАГАЛЬНІ ВИСНОВКИ

У межах успішного виконання цієї кваліфікаційної роботи було повністю спроектовано, розроблено та успішно впроваджено комплексну інноваційну інформаційну систему дослідження рівня атмосферної забрудненості на основі безперервних даних космічного знімання європейського апарата Sentinel-5P. Використання надчутливого спектрометра TROPOMI у тісному поєднанні з розробленими потужними геоінформаційними алгоритмами фільтрації дозволило ефективно подолати традиційні географічні обмеження наземних станцій, забезпечивши безпрецедентно високу просторову точність аналізу навіть для важкодоступних територій зі складним антропогенним навантаженням. Спроектowana масштабована асинхронна мікросервісна архітектура на технічній базі FastAPI, брокера Celery та просторової СУБД PostGIS практично продемонструвала абсолютну стійкість до пікових навантажень під час потокової обробки великих багатовимірних масивів растрових даних та застосування складних векторних масок інфраструктури. Зручний клієнтський інтерфейс, побудований на сучасних технологіях React та Leaflet, надає кінцевим користувачам інтуїтивний, але надзвичайно потужний інструментарій для гнучкого формування аналітичних запитів, візуалізації кольорових теплокарт і автоматичної генерації стандартизованих PDF-звітів. Завдяки цим технічним досягненням, створений масштабований програмний продукт становить неабияку об'єктивну практичну цінність для державних екологічних організацій та дослідницьких інститутів, виступаючи надійним і прозорим інструментом для довгострокової підтримки стратегічного прийняття рішень у глобальній сфері охорони навколишнього середовища.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Atmospheric air quality monitoring over the territory of Ukraine with specification over the cities using Sentinel-5P satellite data - ResearchGate, доступ отримано травня 2, 2026,
https://www.researchgate.net/publication/350384830_Atmospheric_air_quality_monitoring_over_the_territory_of_Ukraine_with_specification_over_the_cities_using_Sentinel-5P_satellite_data
2. Identifying Micro-Level Pollution Hotspots Using Sentinel-5P for the Spatial Analysis of Air Quality Degradation in the National Capital Region, India - MDPI, доступ отримано травня 2, 2026, <https://www.mdpi.com/2071-1050/17/5/2241>
3. Яким повітрям ми дихаємо в Україні, та як на нього вплинув карантин, доступ отримано травня 3, 2026,
<https://www.undp.org/uk/ukraine/blog/yakym-povitryam-my-dykhayemo-v-ukrayini-ta-yak-na-noho-vplynuv-karantyn>
4. Air pollution in Ukraine: a view from the Sentinel-5P satellite - Semantic Scholar, доступ отримано травня 3, 2026, <https://www.semanticscholar.org/paper/Air-pollution-in-Ukraine%3A-a-view-from-the-satellite-Savenets/8107b2421858cb72c7b299bd947f51915a35ec96>
5. Тема 1. Історія розвитку та застосування аерокосмічних методів у лісовому господарстві, доступ отримано травня 3, 2026,
<https://ecology.udau.edu.ua/assets/files/024.pdf>
6. Sentinel-5P: for the air we breathe - Copernicus, доступ отримано травня 4, 2026, <https://www.copernicus.eu/en/sentinel-5p-air-we-breathe>
7. Comparing Sentinel-5P TROPOMI NO 2 column observations with the CAMS regional air quality ensemble - GMD, доступ отримано травня 4, 2026,
<https://gmd.copernicus.org/articles/16/509/2023/>
8. Sentinel-5P air quality data look promising - ECMWF, доступ отримано травня 5, 2026, <https://www.ecmwf.int/en/newsletter/157/news/sentinel-5p-air-quality-data-look-promising>

9. Air Quality Satellite Monitoring by TROPOMI on Sentinel-5P | ESS Open Archive, доступ отримано травня 6, 2026,
<https://essopenarchive.org/doi/10.1002/essoar.10500849>
10. “Моніторинг якості атмосферного повітря над територією України з деталізацією для міст за даними супутника Sentinel-5P”, доступ отримано травня 6, 2026, <https://www.old.nas.gov.ua/UA/Messages/Pages/View.aspx?MessageID=7671>
11. Застосування методів дистанційного зондування у моніторингу навколишнього середовища, доступ отримано травня 7, 2026,
<https://science.lpnu.ua/sites/default/files/journal-paper/2017/may/1645/gka72200921.pdf>
12. Air Quality Satellite Monitoring by TROPOMI on Sentinel-5P, доступ отримано травня 7, 2026,
<https://ntrs.nasa.gov/api/citations/20180008621/downloads/20180008621.pdf>
13. Air quality impacts of COVID-19 lockdown measures detected from space using high spatial resolution observations of multiple trace gases from Sentinel-5P/TROPOMI - Copernicus ACP, доступ отримано травня 7, 2026,
<https://acp.copernicus.org/articles/22/10319/2022/>
14. Current state of atmospheric air pollution in Ukraine based on Sentinel-5P satellite data, доступ отримано травня 7, 2026,
https://www.researchgate.net/publication/343446592_Current_state_of_atmospheric_air_pollution_in_Ukraine_based_on_Sentinel-5P_satellite_data
15. The system for near-real time air pollution monitoring over cities based on the Sentinel-5P satellite data | Visnyk of V. N. Karazin Kharkiv National University. Series Geology. Geography. Ecology, доступ отримано травня 8, 2026,
<https://periodicals.karazin.ua/geoecono/article/view/21383>
16. First Copernicus Sentinel-5 images signal a new era in global air-quality monitoring, доступ отримано травня 8, 2026, <https://www.eumetsat.int/first-copernicus-sentinel-5-images-signal-new-era-global-air-quality-monitoring>

17. Spatio-Temporal Monitoring of Atmospheric Pollutants Using Earth Observation Sentinel 5P TROPOMI Data: Impact of Stubble Burning a Case Study - MDPI, доступ отримано травня 9, 2026, <https://www.mdpi.com/2220-9964/11/5/301>
18. Стаття про моніторинг якості повітря за супутниковими даними в журналі «Remote Sensing - Кафедра математичного моделювання та аналізу даних, доступ отримано травня 9, 2026, <https://mmda.ipt.kpi.ua/2021/11/26i>
19. Новий рівень супутникового моніторингу посилює докази екологічних злочинів, доступ отримано травня 9, 2026, https://nesu.org.ua/novyj-riven-suputnykovogo-monitoryngu-posylyuye-dokazy-ekologichnyh-zlochyniv/detalnyi_opys_funktsionalu_ta_osnovnykh_moduliv.txt
20. Sentinel-5P - Copernicus Data Space Ecosystem, доступ отримано травня 11, 2026, <https://dataspace.copernicus.eu/explore-data/data-collections/sentinel-data/sentinel-5p>
21. Copernicus Services, доступ отримано травня 11, 2026, <https://dataspace.copernicus.eu/copernicus-services>
22. Copernicus Data Space Ecosystem | Europe's eyes on Earth, доступ отримано травня 12, 2026, <https://dataspace.copernicus.eu/>
23. From open data to empowering global Insights with the Copernicus Data Space Ecosystem, доступ отримано травня 12, 2026, <https://earthobservations.org/about-us/news/from-open-data-to-empowering-global-insights-with-the-copernicus-data-space-ecosystem>
24. Copernicus Browser, доступ отримано травня 12, 2026, <https://browser.dataspace.copernicus.eu/>
25. SaveEcoBot: Єдина в Україні екологічна система, доступ отримано травня 13, 2026, <https://www.saveecobot.com/>
26. Windy - Easy access air quality forecasts - Copernicus Atmosphere Monitoring Service, доступ отримано травня 13, 2026, <https://atmosphere.copernicus.eu/windy-easy-access-air-quality-forecasts>
27. Copernicus Atmosphere Monitoring Service (CAMS) data and forecast are now accessible on Windy., доступ отримано травня 14, 2026,

<https://community.windy.com/topic/8577/copernicus-atmosphere-monitoring-service-cams-data-and-forecast-are-now-accessible-on-windy>

29. Designing Scalable Geospatial Systems: From FastAPI & PostGIS to Real-Time GIS Platforms | by Ashly W Ochwada | Mar, 2026 | Medium, доступ отримано травня 14, 2026, <https://medium.com/@ashlywochwada/designing-scalable-geospatial-systems-from-fastapi-postgis-to-real-time-gis-platforms-0c61ec9d8f63>

30. Flask vs. FastAPI: Which One to Choose - GeeksforGeeks, доступ отримано травня 14, 2026, <https://www.geeksforgeeks.org/blogs/flask-vs-fastapi/>

31. FastAPI vs Flask 2025: Performance, Speed & When to Choose - Strapi, доступ отримано травня 15, 2026, <https://strapi.io/blog/fastapi-vs-flask-python-framework-comparison>

32. FastAPI vs Flask: Key Differences, Performance, and Use Cases | Codecademy, доступ отримано травня 15, 2026, <https://www.codecademy.com/article/fastapi-vs-flask-key-differences-performance-and-use-cases>

33. Designing High-Performance Scalable REST APIs using FastAPI and PostgreSQL, доступ отримано травня 16, 2026, https://www.researchgate.net/publication/404277008_Designing_High-Performance_Scalable_REST_APIs_using_FastAPI_and_PostgreSQL

34. What Is PostGIS? The Open-Source Spatial Database Powering Modern GIS and AI, доступ отримано травня 17, 2026, <https://forrest.nyc/what-is-postgis-the-open-source-spatial-database-powering-modern-gis-and-ai/>

35. 2. Introduction - PostGIS, доступ отримано травня 17, 2026, <https://postgis.net/workshops/postgis-intro/introduction.html>

36. Why should you care about PostGIS? – A gentle introduction to spatial databases - Medium, доступ отримано травня 19, 2026, <https://medium.com/@tjukanov/why-should-you-care-about-postgis-a-gentle-introduction-to-spatial-databases-9eccd26bc42b>

37. Maptainer: High Performance and Technical Differentiation with Vector Tiles, PostGIS, and FastAPI, доступ отримано травня 20, 2026,

<https://maptainer.com/en/blog/maptainer-high-performance-and-technical-differentiation-with-vector-tiles-postgis-and-fastapi>

38. Хлопці, знаєте якісь реальні приклади використання Celery в реальному житті? : r/django - Reddit, доступ отримано травня 20, 2026, https://www.reddit.com/r/django/comments/gn190e/do_you_guys_know_any_realworld_use_cases_for/?tl=uk

39. Data pipelines with Celery: modular, signal-driven and manageable – Marin Aglić Čuvrić, доступ отримано травня 20, 2026, <https://www.youtube.com/watch?v=g-3KClG-2xc>

40. Data pipelines: what, why and which ones - UbiOps, доступ отримано травня 20, 2026, <https://ubiops.com/data-pipelines-what-why-and-which-ones/>

41. Interactive Dashboard with Next.js and Python | 2025 Guide - August Infotech, доступ отримано травня 21, 2026, <https://www.augustinfotech.com/blogs/building-an-interactive-dashboard-with-next-js-and-python/>

42. Create an Asset Tracker with Next.js and React Leaflet - DEV Community, доступ отримано травня 21, 2026, <https://dev.to/paigen11/create-an-asset-tracker-map-with-nextjs-and-react-leaflet-42d2>

43. Copernicus Data Space Ecosystem establishes public cloud processing for earth observation data - PMC, доступ отримано травня 23, 2026, <https://pmc.ncbi.nlm.nih.gov/articles/PMC13057113/>

44. 75+ Geospatial Python and Spatial Data Science Resources and Guides - Matt Forrest, доступ отримано травня 23, 2026, <https://forrest.nyc/75-geospatial-python-and-spatial-data-science-resources-and-guides/>

45. Building a Next.js Dashboard with Dynamic Charts and SSR - Cube Blog, доступ отримано травня 25, 2026, <https://cube.dev/blog/building-nextjs-dashboard-with-dynamic-charts-and-ssr>