

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Комп'ютерних наук і технологій
(повне найменування факультету)

Комп'ютерні системи та мережі
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалаврський
(ступінь вищої освіти)

на тему: РОЗРОБКА ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ МОНІТОРИНГУ
СТАНУ ПРИРОДНОГО ДОВКІЛЛЯ

Виконав(ла): студент(ка) 4 курсу,
групи КНТз-512

Спеціальності

123 Комп'ютерна інженерія

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерна інженерія

(назва освітньої програми (спеціалізації))

БОРОВИК В.Ю.

(ПРИЗВИЩЕ та ініціали)

Керівник ЗЕЛЕНЬОВА І.Я.

(ПРИЗВИЩЕ та ініціали)

Рецензент ЩЕРБАКОВ В.І.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет Комп'ютерних наук і технологій
Кафедра комп'ютерних систем та мереж
Ступінь вищої освіти бакалаврський
Спеціальність 123 Комп'ютерна інженерія
(код і найменування)
Освітня програма (спеціалізація) Комп'ютерна інженерія
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри КУДЕРМЕТОВ Р.К.

«_» _____ 2026 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

БОРОВИКА Володимира Юрійовича

(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) розробка інформаційної підсистеми моніторингу стану природного довкілля

керівник проєкту (роботи) к.т.н., доцент, Зеленьова І.Я.

(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від « 14 » квітня 2026 року № 160

2. Строк подання студентом проєкту (роботи) 01.06.2026 р.

3. Вихідні дані до проєкту (роботи) опис предметної області, технології розробки клієнтської та серверної частини

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1) Опис предметної області;

2) Розробка вимог до програмного забезпечення;

3) Моделювання програмного забезпечення;

4) Проєктування архітектури програмного забезпечення;

5) Проєктування інтерфейсу користувача

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-5	ЗЕЛЕНЬОВА І.Я.		
Нормоконтроль	ДЬЯЧУК Т.С.		

7. Дата видачі завдання «14» квітня 2025 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз предметної області	до 18.04.2025	
2	Визначення та аналіз вимог до програмного забезпечення	до 22.04.2026	
3	Розробка специфікації вимог	до 24.04.2026	
4	Розробка діаграми варіантів використання	до 28.04.2026	
5	Розробка діаграми послідовності	до 01.05.2026	
6	Розробка описів прецедентів	до 05.05.2026	
7	Проєктування архітектури програмного забезпечення	до 12.05.2026	
8	Проєктування інтерфейсу користувача	до 22.05.2026	
9	Оформлення пояснювальної записки	до 25.05.2026	
10	Проходження нормоконтролю	до 01.06.2026	
11	Перевірка на наявність академічного плагіату	до 03.06.2026	
12	Проходження рецензування	до 10.06.2026	

Студент(ка)

(підпис)

Володимир БОРОВИК

(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

(підпис)

Ірина ЗЕЛЕНЬОВА

(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

ПЗ: 94 с., 8 рис., 15 табл., 4 лістинги, 2 додатки, 25 джерел.

МОНІТОРИНГ ДОВКІЛЛЯ, ІНФОРМАЦІЙНА ПІДСИСТЕМА, ESP32, ДАТЧИК, ІНТЕРНЕТ РЕЧЕЙ, MQTT, БАЗА ДАНИХ, ВЕБ-ДОДАТОК, ЯКІСТЬ ПОВІТРЯ, ТЕЛЕМЕТРІЯ

Об'єкт розробки – інформаційна підсистема моніторингу стану природного довкілля, що забезпечує автоматизований збір, передавання, збереження та візуалізацію показників якості повітря, води та ґрунту.

Метою роботи є розробка інформаційної підсистеми моніторингу стану природного довкілля на основі мікроконтролерних вузлів збору даних та серверного веб-додатку, проведення аналізу сучасних технологій екологічного моніторингу, проектування апаратної і програмної частин, а також моделювання й дослідження розробленої підсистеми.

Пояснювальна записка складається з трьох розділів.

Перший розділ присвячено аналізу предметної галузі: розглянуто завдання екологічного моніторингу, проаналізовано існуючі системи-аналоги, технології передавання даних та сформульовано технічне завдання на розробку підсистеми.

У другому розділі виконано розробку апаратного та програмного забезпечення: обґрунтовано вибір елементної бази вузлів збору даних на основі мікроконтролера ESP32 та комплекту датчиків, розроблено структурну й функціональну схеми, спроектовано структуру бази даних і серверний веб-додаток, наведено алгоритми та фрагменти програмного коду.

Третій розділ присвячено моделюванню та експериментальній перевірці працездатності підсистеми, аналізу отриманих результатів вимірювань та оцінці передавання даних. Надано рекомендації щодо подальшого розвитку підсистеми.

У результаті виконання дипломної роботи розроблено, змодельовано та досліджено інформаційну підсистему моніторингу стану природного довкілля та сформульовано рекомендації щодо її практичного впровадження.

ЗМІСТ

Скорочення та умовні позначки	6
Вступ.....	7
1 Аналіз предметної галузі та постановка задачі.....	10
1.1 Призначення та завдання моніторингу стану довкілля.....	10
1.2 Огляд та аналіз існуючих систем-аналогів.....	18
1.3 Аналіз технологій передавання та оброблення даних.....	21
1.4 Постановка задачі та технічне завдання	26
1.5 Обґрунтування архітектурних рішень	29
1.6 Техніко-економічне обґрунтування	31
2 Розробка апаратного та програмного забезпечення	33
2.1 Розробка структурної та функціональної схем	33
2.2 Вибір та обґрунтування елементної бази.....	36
2.3 Проектування бази даних	46
2.4 Розробка програмного забезпечення.....	50
2.5 Забезпечення захисту інформації	60
2.6 Тестування програмного забезпечення.....	61
2.7 Особливості розгортання та експлуатації підсистеми	63
2.8 Узагальнена характеристика розробленої підсистеми	65
3 Моделювання та експериментальна перевірка	67
3.1 Формування моделі та умови експерименту	67
3.2 Інтерпретація результатів	70
3.3 Рекомендації щодо подальшого розвитку	74
3.4 Оцінка результатів та перспективи впровадження.....	76
3.5 Обмеження проведеного дослідження.....	78
Висновки	81
Перелік джерел посилання	84
Додаток А Технічне завдання на розробку підсистеми	87
Додаток Б Фрагмент сценарію створення бази даних.....	88
Додаток В Слайди презентації.....	89

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

АЦП	– аналого-цифровий перетворювач
БД	– база даних
ЕОМ	– електронна обчислювальна машина
ІС	– інформаційна система
КС	– комп'ютерна система
МК	– мікроконтролер
ПЗ	– програмне забезпечення
СКБД	– система керування базами даних
API	– програмний інтерфейс застосунку, Application Programming Interface
GUI	– графічний інтерфейс користувача, Graphical User Interface
HTTP	– протокол передавання гіпертексту, HyperText Transfer Protocol
IoT	– інтернет речей, Internet of Things
JSON	– об'єктний запис JavaScript, JavaScript Object Notation
MQTT	– протокол передавання телеметрії, Message Queuing Telemetry Transport
PM	– тверді зважені частинки, Particulate Matter
REST	– передавання репрезентативного стану, Representational State Transfer
SQL	– мова структурованих запитів, Structured Query Language
TLS	– захист транспортного рівня, Transport Layer Security

ВСТУП

Антропогенне навантаження на навколишнє природне середовище постійно зростає, що зумовлює необхідність безперервного спостереження за станом атмосферного повітря, водних об'єктів та ґрунтів. Своєчасне виявлення перевищень гранично допустимих концентрацій забруднювальних речовин є передумовою прийняття обґрунтованих управлінських рішень у сфері охорони довкілля, а також інформування населення про екологічну ситуацію.

Традиційні системи екологічного моніторингу базуються на стаціонарних постах спостереження, обладнаних дорогими стаціонарними аналізаторами. Такі системи характеризуються високою точністю вимірювань, проте мають значну вартість, низьку щільність розміщення постів та обмежену оперативність надання даних. Розвиток технологій інтернету речей (IoT), поява недорогих мікроконтролерів та електрохімічних і оптичних датчиків уможливили побудову розподілених підсистем моніторингу з високою щільністю вузлів збору даних за прийнятною вартістю.

Актуальність роботи зумовлена потребою у створенні гнучкої, масштабованої та економічно доступної інформаційної підсистеми, що здатна автоматично збирати показники стану довкілля з територіально рознесених вузлів, передавати їх до центрального сервера, зберігати у базі даних та надавати користувачеві у зручній формі через веб-інтерфейс.

Розвиток мікроелектроніки та технологій інтернету речей створив передумови для побудови систем моніторингу принципово нового типу. Сучасні мікроконтролери поєднують у єдиному кристалі обчислювальне ядро, пам'ять, засоби вводу-виводу та радіомодуль, що дає змогу будувати компактні автономні вузли вимірювання за невисокої вартості. Поява масових недорогих датчиків фізичних та хімічних величин уможливила вимірювання широкого кола екологічних показників без застосування дорогого лабораторного обладнання. Поєднання цих чинників робить актуальним створення розподілених підсистем

моніторингу, що доповнюють традиційні стаціонарні системи та забезпечують детальніше покриття території спостереження.

Об'єктом розробки є процес автоматизованого збору, передавання, збереження та відображення даних екологічного моніторингу.

Предметом розробки є апаратні та програмні засоби інформаційної підсистеми моніторингу стану природного довкілля.

Метою дипломної роботи є розробка інформаційної підсистеми моніторингу стану природного довкілля на основі мікроконтролерних вузлів збору даних та серверного веб-додатку.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- проаналізувати предметну галузь та завдання екологічного моніторингу;
- виконати огляд існуючих систем-аналогів та технологій передавання даних;
- сформулювати технічне завдання на розробку підсистеми;
- обґрунтувати вибір елементної бази вузлів збору даних та датчиків;
- розробити структурну та функціональну схеми підсистеми;
- спроектувати структуру бази даних і серверний веб-додаток;
- розробити алгоритми роботи вузлів та програмне забезпечення;
- виконати моделювання та експериментальну перевірку працездатності підсистеми;
- проаналізувати отримані результати та сформулювати рекомендації.

Практичне значення роботи полягає у можливості застосування розробленої підсистеми для організації локального екологічного моніторингу на території підприємств, населених пунктів, навчальних закладів та природоохоронних територій.

Методи дослідження, застосовані у роботі, включають аналіз науково-технічної літератури та нормативних документів для вивчення предметної галузі; методи порівняльного аналізу для оцінювання систем-аналогів та вибору технічних рішень; методи проектування інформаційних систем для розроблення структури підсистеми та бази даних; методи об'єктно-орієнтованого

програмування для реалізації програмного забезпечення; а також методи експериментального дослідження та статистичного оброблення даних для перевірки працездатності підсистеми.

Пояснювальна записка складається зі вступу, трьох розділів, висновків, переліку джерел посилання та додатків. У першому розділі виконано аналіз предметної галузі та сформульовано технічне завдання. Другий розділ присвячено розробленню апаратного й програмного забезпечення підсистеми. У третьому розділі наведено результати моделювання та експериментального дослідження. У висновках узагальнено основні результати роботи.

Очікуваним результатом виконання роботи є створення працездатної інформаційної підсистеми моніторингу стану природного довкілля, що забезпечує автоматизований збір, передавання, збереження та візуалізацію показників якості повітря, води та ґрунту, а також підтвердження її працездатності за результатами експериментального дослідження. Розроблені технічні рішення та отримані результати мають бути придатними для практичного впровадження у сфері локального екологічного моніторингу та для подальшого розвитку в межах наукових і навчальних досліджень.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Призначення та завдання моніторингу стану довкілля

Моніторинг стану природного довкілля – це система регулярних спостережень, збирання, оброблення, передавання, збереження та аналізу інформації про стан навколишнього природного середовища, прогнозування його змін та розроблення науково обґрунтованих рекомендацій для прийняття рішень щодо запобігання негативним змінам і дотримання вимог екологічної безпеки. Залежно від об'єкта спостереження розрізняють моніторинг атмосферного повітря, поверхневих і підземних вод, ґрунтів, рослинного й тваринного світу, а також радіаційний та шумовий моніторинг.

Концепція моніторингу довкілля сформувалася як відповідь на потребу систематичного спостереження за станом середовища в умовах зростання антропогенного навантаження. Ключовою ідеєю моніторингу є не лише фіксація поточного стану, а й відстеження його зміни у часі, що дає змогу виявляти тенденції та прогнозувати розвиток ситуації. Це відрізняє моніторинг від одноразових вимірювань та зумовлює вимогу до регулярності й безперервності спостережень, яку забезпечують автоматизовані системи збору даних.

За ступенем автоматизації системи моніторингу поділяють на ручні, у яких відбір проб та вимірювання виконуються людиною, автоматизовані, у яких частину функцій виконує людина, та автоматичні, у яких увесь цикл від вимірювання до подання даних виконується без участі людини. Розроблювана підсистема належить до автоматичних, оскільки вимірювання, передавання, збереження та відображення даних відбуваються автоматично за заданим алгоритмом. Це забезпечує безперервність спостережень, виключає вплив людського чинника на процес збору даних та дає змогу отримувати дані у режимі, близькому до реального часу, що є суттєвою перевагою для оперативного контролю стану довкілля.

Інформаційна підсистема моніторингу є складовою частиною загальної системи екологічного моніторингу та реалізує функції автоматизованого

вимірювання фізичних і хімічних показників, передавання первинних даних до центру оброблення, їх накопичення та подання у формі, придатній для аналізу. На відміну від повної автоматизованої системи екологічного моніторингу, підсистема охоплює визначений набір показників та обмежену територію, проте побудована за принципами, що допускають подальше масштабування.

В Україні функціонування системи моніторингу довкілля регламентується Законом України «Про охорону навколишнього природного середовища» та Положенням про державну систему моніторингу довкілля. Згідно з цими документами державний моніторинг здійснюється низкою суб'єктів, кожен з яких відповідає за певний компонент середовища, а зібрана інформація підлягає узагальненню та оприлюдненню. Розроблювана підсистема не замінює державну систему моніторингу, а доповнює її на рівні окремого об'єкта, забезпечуючи більш щільне у просторі та часі покриття території спостереження [1,2].

За цільовим призначенням розрізняють кілька рівнів моніторингу: локальний (імпактний), що охоплює зони впливу окремих джерел забруднення; регіональний, що поширюється на територію адміністративної або природної області; а також національний і глобальний рівні. Розроблювана підсистема належить до локального рівня, оскільки призначена для контролю стану довкілля на території конкретного об'єкта – підприємства, населеного пункту, навчального закладу чи природоохоронної території.

За способом отримання інформації моніторинг поділяють на контактний, що передбачає безпосереднє вимірювання показників датчиками, та дистанційний, що ґрунтується на даних аерокосмічного зондування. У межах роботи реалізовано контактний автоматизований моніторинг із застосуванням стаціонарних вузлів вимірювання, рознесених у просторі. Такий підхід забезпечує безперервність спостережень та високу часову роздільну здатність даних, що є важливим для виявлення короткочасних епізодів забруднення [3].

Сучасна екологічна ситуація в Україні та світі характеризується значним антропогенним навантаженням на навколишнє середовище, зумовленим розвитком промисловості, транспорту та урбанізацією. Забруднення

атмосферного повітря у великих містах часто перевищує гранично допустимі рівні, що негативно впливає на здоров'я населення та якість життя. Стан водних об'єктів та ґрунтів також зазнає погіршення внаслідок скидів стічних вод, застосування добрив і пестицидів та накопичення відходів. У цих умовах систематичний контроль стану довкілля набуває особливого значення як інструмент виявлення проблем та обґрунтування природоохоронних заходів.

Розгортання щільної мережі недорогих вузлів моніторингу дає змогу не лише фіксувати середні рівні забруднення, а й виявляти локальні зони підвищеного ризику, простежувати поширення забруднень та оцінювати ефективність вжитих заходів. Це робить розроблювану підсистему корисним інструментом як для природоохоронних органів та підприємств, так і для громадськості, зацікавленої в інформації про стан довкілля у місцях проживання. Зростання суспільного запиту на достовірну екологічну інформацію додатково підкреслює актуальність створення доступних та відкритих засобів моніторингу.

Функціонально процес моніторингу можна подати як послідовність етапів: вимірювання фізичних та хімічних величин первинними перетворювачами; перетворення сигналів датчиків у цифрову форму; первинне оброблення та фільтрацію даних; передавання даних каналом зв'язку; збереження у сховищі даних; вторинне оброблення, аналіз та подання результатів користувачеві. Кожен з цих етапів реалізується відповідним апаратним або програмним компонентом підсистеми, а якість роботи підсистеми загалом визначається узгодженістю всіх етапів.

1.1.1 Джерела та види забруднення довкілля

Для коректного вибору контрольованих показників необхідно розуміти основні джерела та види забруднення навколишнього середовища. Забруднення атмосферного повітря поділяють на природне, спричинене виверженнями вулканів, пиловими бурями та лісовими пожежами, та антропогенне, пов'язане з діяльністю людини. Основними антропогенними джерелами є транспорт, енергетика, промислові підприємства та опалення житла. Транспорт є провідним

джерелом твердих частинок та оксидів азоту у містах, тоді як промисловість і енергетика додатково спричиняють викиди оксидів сірки та інших сполук.

Тверді зважені частинки класифікують за аеродинамічним діаметром: частинки розміром до 10 мкм (PM10) та до 2,5 мкм (PM2.5). Саме дрібнодисперсна фракція PM2.5 становить найбільшу небезпеку, оскільки здатна проникати у альвеоли легень та потрапляти у кровоносну систему. Газоподібні забруднювачі, такі як діоксид азоту, оксид вуглецю та діоксид сірки, утворюються переважно під час згоряння палива. Контроль саме цих речовин є першочерговим завданням систем моніторингу якості повітря.

Концентрація забруднювальних речовин у повітрі залежить не лише від інтенсивності джерел, а й від метеорологічних умов. Швидкість та напрямок вітру визначають перенесення та розсіювання забруднень, температурна стратифікація атмосфери впливає на вертикальне перемішування повітря, а опади сприяють вимиванню частинок з атмосфери. Тому для коректної інтерпретації даних моніторингу якості повітря важливо одночасно реєструвати метеорологічні параметри, що зумовлює доцільність включення до складу вузла датчика температури, вологості та тиску. Це дає змогу враховувати вплив погодних умов під час аналізу динаміки забруднення.

Забруднення водних об'єктів характеризується зміною фізико-хімічних показників води: водневого показника рН, електропровідності, вмісту розчиненого кисню, концентрації завислих речовин. Забруднення ґрунтів виявляється у зміні їх кислотності, засоленості, вологості та накопиченні важких металів. Комплексний характер розроблюваної підсистеми передбачає контроль показників усіх трьох компонентів середовища – повітря, води та ґрунту, що відрізняє її від вузькоспеціалізованих систем моніторингу лише якості повітря.

Комплексний підхід до моніторингу має суттєву перевагу, оскільки стан різних компонентів довкілля взаємопов'язаний. Наприклад, атмосферні опади переносять забруднювальні речовини з повітря у ґрунт та водойми, а стан ґрунту впливає на якість підземних вод. Одночасне спостереження за повітрям, водою та ґрунтом дає змогу виявляти такі взаємозв'язки та формувати цілісну картину

екологічного стану території. Водночас комплексний моніторинг висуває підвищені вимоги до уніфікації засобів збору й оброблення даних різної фізичної природи, що враховано у структурі розроблюваної підсистеми за рахунок єдиного формату подання даних незалежно від типу показника.

1.1.2 Нормативно-правова база моніторингу

Здійснення екологічного моніторингу в Україні регламентується низкою нормативно-правових актів. Базовим є Закон України «Про охорону навколишнього природного середовища», що визначає загальні засади охорони довкілля та обов'язковість спостережень за його станом. Порядок функціонування державної системи моніторингу довкілля встановлюється відповідним Положенням, затвердженим постановою Кабінету Міністрів України, яке визначає суб'єктів моніторингу та порядок обміну інформацією між ними.

Значення гранично допустимих концентрацій забруднювальних речовин встановлюються санітарними нормами та державними стандартами. Орієнтиром для встановлення цільових показників якості повітря слугують також настанови Всесвітньої організації охорони здоров'я, що визначають рекомендовані рівні концентрацій з огляду на захист здоров'я населення. Розроблювана підсистема використовує ці значення як порогові для формування попереджень, проте архітектура передбачає можливість їх настроювання відповідно до чинних нормативів [1].

1.1.3 Структура даних екологічного моніторингу

Дані екологічного моніторингу мають характерну структуру, що визначає вимоги до засобів їх збору, зберігання та оброблення. Кожне окреме вимірювання являє собою значення певного показника, зареєстроване визначеним засобом вимірювання у конкретний момент часу та у конкретній точці простору. Таким чином, елементарною одиницею даних є кортеж, що містить ідентифікатор показника, числове значення, мітку часу та просторову прив'язку. Сукупність таких вимірювань утворює багатовимірний масив даних, упорядкований за часом, простором та типом показника [4].

Часовий характер даних моніторингу зумовлює потребу у зберіганні рядів значень з регулярними або нерегулярними інтервалами між вимірюваннями. Аналіз таких рядів дає змогу виявляти тренди, періодичні коливання та аномалії. Просторова складова даних уможлиблює побудову карт розподілу показників та виявлення зон підвищеного забруднення. Поєднання часового та просторового вимірів робить дані моніторингу придатними для просторово-часового аналізу, що є потужним інструментом дослідження стану довкілля [5].

Важливою характеристикою даних моніторингу є їх достовірність, що може знижуватися внаслідок похибок вимірювання, збоїв обладнання чи завад. Тому кожне вимірювання доцільно супроводжувати ознакою достовірності, яка враховується під час подальшого аналізу. Структура даних підсистеми, розглянута у другому розділі, відображає зазначені особливості: дані зберігаються з прив'язкою до датчика, часу та з ознакою достовірності, що забезпечує повноту та коректність інформації для аналізу.

Основними завданнями інформаційної підсистеми моніторингу стану довкілля є:

- безперервне або періодичне вимірювання показників якості повітря (концентрація твердих частинок PM2.5 і PM10, рівень діоксиду вуглецю, оксидів азоту, летких органічних сполук);
- вимірювання показників водного середовища (температура, водневий показник pH, електропровідність, рівень розчиненого кисню);
- вимірювання показників стану ґрунту (вологість, температура, кислотність);
- реєстрація супутніх метеорологічних параметрів (температура, вологість і тиск повітря);
- автоматичне передавання вимірюваних даних до центрального сервера;
- накопичення даних у базі даних з прив'язкою до часу та географічних координат вузла;
- формування попереджень у разі перевищення гранично допустимих концентрацій (ГДК);

– візуалізація поточного стану та динаміки показників через веб-інтерфейс.

Найважливіші показники якості атмосферного повітря та їх орієнтовні гранично допустимі значення наведено у таблиці 1.1. Саме ці параметри обрано як базовий набір для розроблюваної підсистеми.

Таблиця 1.1 – Контрольовані показники та орієнтовні ГДК

Показник	Позначення	Одиниця	Орієнтовна ГДК
Тверді частинки 2.5 мкм	PM2.5	мкг/м ³	25 (доб.)
Тверді частинки 10 мкм	PM10	мкг/м ³	50 (доб.)
Діоксид вуглецю	CO ₂	ppm	1000 (приміщ.)
Діоксид азоту	NO ₂	мкг/м ³	200 (год.)
Водневий показник води	pH	од.	6,5 – 8,5
Вологість ґрунту	W	%	40 – 70

Значущість завдання моніторингу довкілля зумовлена кількома чинниками. По-перше, забруднення атмосферного повітря, води та ґрунту безпосередньо впливає на здоров'я населення, що підтверджено численними епідеміологічними дослідженнями. За оцінками Всесвітньої організації охорони здоров'я, забруднення повітря є одним з провідних чинників передчасної смертності у світі. По-друге, своєчасне виявлення перевищень дає змогу оперативно вживати заходів щодо обмеження впливу джерел забруднення. По-третє, накопичені ряди спостережень є основою для аналізу тенденцій, оцінювання ефективності природоохоронних заходів та планування забудови територій.

Зазначені завдання підсистеми тісно взаємопов'язані та утворюють єдиний технологічний цикл. Вимірювання показників є джерелом первинних даних, без яких неможливі подальші етапи. Передавання даних забезпечує їх централізоване накопичення, а накопичення створює основу для аналізу та формування попереджень. Візуалізація завершує цикл, надаючи користувачеві результати у зручній формі. Якість виконання кожного завдання впливає на ефективність підсистеми загалом, тому під час проектування приділено увагу всім етапам без винятку.

Окремою проблемою традиційного моніторингу є низька просторова роздільна здатність даних. Стаціонарні пости розташовані на значній відстані один від одного, тому дані одного поста екстраполюються на велику територію, що не враховує локальних особливостей розподілу забруднення поблизу автомагістралей, промислових об'єктів чи у внутрішньоквартальних зонах. Розроблювана підсистема покликана подолати це обмеження за рахунок розгортання значної кількості недорогих вузлів вимірювання, що забезпечують детальну картину просторового розподілу показників.

Не менш важливою є оперативність надання даних. Традиційні методи лабораторного аналізу проб передбачають значний час між відбором проби та отриманням результату, що унеможливорює оперативне реагування на епізоди забруднення. Автоматизована підсистема забезпечує практично безперервне вимірювання та негайне передавання даних, завдяки чому перевищення граничних значень виявляються у режимі, близькому до реального часу. Це створює основу для оперативного інформування населення та вжиття невідкладних заходів, що є суттєвою перевагою автоматизованих систем моніторингу.

Тверді зважені частинки PM_{2.5} та PM₁₀ є одним з найбільш небезпечних для здоров'я людини видів забруднення повітря, оскільки дрібнодисперсні частинки здатні проникати у глибокі відділи дихальної системи та кровоносне русло. Концентрація діоксиду вуглецю характеризує якість повітря у приміщеннях та опосередковано – інтенсивність провітрювання. Для водного середовища ключовими показниками є водневий показник pH, що визначає кислотність води, та електропровідність, пов'язана з мінералізацією. Стан ґрунту характеризується насамперед вологістю, що впливає на родючість та ерозійні процеси. Саме ці показники, як такі, що мають найбільше практичне значення та можуть бути виміряні доступними датчиками, покладено в основу набору контрольованих параметрів підсистеми [14,15].

1.2 Огляд та аналіз існуючих систем-аналогів

Системи екологічного моніторингу можна класифікувати за кількома ознаками. За масштабом охоплення території розрізняють глобальні, національні, регіональні та локальні системи. За складом контрольованих показників – спеціалізовані (контроль одного компонента середовища) та комплексні. За способом розміщення засобів вимірювання – стаціонарні, мобільні та змішані. За ступенем автоматизації – ручні, автоматизовані та автоматичні. За способом передавання даних – з провідними, бездротовими та комбінованими каналами зв'язку. Розроблювана підсистема за цією класифікацією є локальною, комплексною, стаціонарною, автоматичною з бездротовим каналом зв'язку.

На ринку та в дослідницькому середовищі представлено значну кількість систем екологічного моніторингу, які різняться вартістю, точністю, способом передавання даних та ступенем відкритості. Для обґрунтування проектних рішень доцільно розглянути найхарактерніші з них.

Під час аналізу систем-аналогів доцільно оцінювати їх за такими критеріями: вартість окремого вузла та системи в цілому, метрологічна точність вимірювань, просторова щільність розміщення вузлів, оперативність надання даних, ступінь відкритості даних та програмного забезпечення, можливість автономної роботи вузлів, а також залежність від зовнішньої інфраструктури. Саме за цими критеріями нижче розглянуто основні класи систем.

Стаціонарні державні пости спостереження обладнані сертифікованими стаціонарними газоаналізаторами та забезпечують високу метрологічну точність. Їхніми недоліками є висока вартість, складність обслуговування та низька щільність розміщення, унаслідок чого дані характеризують лише окремі точки міста.

Принцип дії стаціонарних аналізаторів ґрунтується на високоточних фізико-хімічних методах вимірювання – хемілюмінесцентному, ультрафіолетовому флуоресцентному, інфрачервоному абсорбційному та інших. Ці методи

забезпечують високу селективність та точність, проте відповідне обладнання має значні габарити, вартість та потребує регулярного технічного обслуговування й перевірки кваліфікованим персоналом. Унаслідок цього кількість стаціонарних постів навіть у великих містах є обмеженою, що не дає змоги отримати детальну просторову картину забруднення.

Комерційні мережі недорогих датчиків (наприклад, платформи на зразок PurpleAir, Luftdaten / Sensor.Community). Вони базуються на масових оптичних датчиках пилу та мікроконтролерах, передають дані через Wi-Fi до хмарних сервісів і надають відкриті карти забруднення. Перевагами є низька вартість і висока щільність, недоліком – обмежена точність та залежність від зовнішньої хмарної інфраструктури.

Такі платформи продемонстрували, що мережі недорогих датчиків здатні забезпечувати корисну інформацію про просторово-часовий розподіл забруднення за умови належного оброблення даних та калібрування. Водночас залежність від зовнішньої хмарної інфраструктури означає, що працездатність вузлів та доступність даних визначаються політикою стороннього сервісу, а самі дані зберігаються поза контролем користувача. Для організацій, що потребують повного контролю над власними даними та незалежності від зовнішніх сервісів, цей підхід має суттєві обмеження, які усуваються застосуванням власного серверного застосунку.

Промислові SCADA-рішення застосовуються на підприємствах для контролю викидів. Вони мають високу надійність та інтеграцію з виробничими процесами, проте є закритими, дорогими й надлишковими для завдань локального моніторингу.

Академічні та аматорські розробки на основі мікроконтролерів. Значна кількість досліджень присвячена побудові вузлів моніторингу на платформах Arduino, ESP8266 та ESP32 з оптичними датчиками пилу та електрохімічними газовими датчиками. Такі розробки відрізняються відкритістю програмного коду й апаратних рішень, низькою вартістю та гнучкістю. Їхнім обмеженням є потреба у самостійному калібруванні датчиків та забезпеченні надійності передавання

даних. Саме цей клас рішень є найближчим за призначенням до розроблюваної підсистеми, тому досвід відповідних робіт враховано під час проектування.

Аналіз академічних публікацій з тематики недорогих систем моніторингу якості повітря свідчить про усталеність низки технічних рішень: застосування мікроконтролерів сімейства ESP, оптичних датчиків пилу серії PMS, передавання даних мережею Wi-Fi та збереження у базах даних з подальшою візуалізацією. Водночас у багатьох роботах недостатньо уваги приділяється забезпеченню надійності передавання даних за нестабільного зв'язку та захисту інформації. У розроблюваній підсистемі ці аспекти враховано шляхом реалізації механізму буферизації та застосування захищеного каналу передавання, що є її відмінною рисою порівняно з типовими аматорськими рішеннями.

Порівняльний аналіз основних характеристик розглянутих систем наведено у таблиці 1.2.

Таблиця 1.2 – Порівняння систем моніторингу довкілля

Характеристика	Держ. пости	Мережі IoT	SCADA	Розробка
Вартість вузла	Висока	Низька	Висока	Низька
Точність	Висока	Середня	Висока	Середня
Щільність	Низька	Висока	Середня	Висока
Відкритість даних	Часткова	Висока	Низька	Повна
Автономність вузла	Ні	Частк.	Ні	Так

Аналіз показав, що для завдань локального екологічного моніторингу найбільш доцільною є архітектура на основі недорогих мікроконтролерних вузлів з власним серверним застосунком. Такий підхід поєднує низьку вартість і високу щільність мереж IoT з повним контролем над даними та незалежністю від сторонніх хмарних платформ, що й покладено в основу розроблюваної підсистеми.

Слід зазначити, що знижена точність недорогих датчиків порівняно зі стаціонарними аналізаторами не є критичною для завдань, які вирішує розроблювана підсистема. Для виявлення тенденцій, відносного порівняння стану

довкілля у різних точках території та оперативного попередження про значні перевищення достатньо тієї точності, яку забезпечують сучасні масові датчики. За потреби підвищення точності окремих вимірювань підсистема може бути доповнена еталонними приладами, дані яких використовуватимуться для калібрування мережі недорогих вузлів.

1.3 Аналіз технологій передавання та оброблення даних

Вибір технологій передавання даних суттєво впливає на енергоспоживання вузлів, надійність та масштабованість підсистеми. Розглянемо доступні рішення.

1.3.1 Технології бездротового зв'язку

Бездротовий зв'язок є основою побудови розподілених систем моніторингу, оскільки забезпечує передавання даних від територіально рознесених вузлів без прокладання кабельних ліній. Вибір технології зв'язку визначається співвідношенням дальності, швидкості передавання, енергоспоживання та вартості, а також наявністю відповідної інфраструктури на об'єкті спостереження.

Для передавання даних від вузлів до шлюзу найчастіше застосовують такі технології: Wi-Fi, Bluetooth Low Energy, ZigBee, а також технології далекого зв'язку LoRaWAN та стільникові NB-IoT. Wi-Fi забезпечує високу пропускну здатність та повсюдну інфраструктуру, проте має відносно велике енергоспоживання. LoRaWAN дає значну дальність за малого споживання, але обмежений у швидкості передавання. З урахуванням наявності Wi-Fi-інфраструктури на більшості об'єктів спостереження та вбудованого модуля Wi-Fi у мікроконтролері ESP32 для розроблюваної підсистеми обрано саме Wi-Fi [7,21].

Порівняння технологій бездротового зв'язку за основними параметрами наведено у таблиці 1.3. Параметри дальності та енергоспоживання наведено

орієнтовно, оскільки вони суттєво залежать від умов поширення сигналу та режиму роботи.

Таблиця 1.3 – Порівняння технологій бездротового зв'язку

Технологія	Дальність	Швидкість	Споживання
Wi-Fi	до 100 м	висока	високе
BLE	до 50 м	середня	низьке
ZigBee	до 100 м	низька	низьке
LoRaWAN	до 10 км	дуже низька	дуже низьке
NB-IoT	до 10 км	низька	низьке

Вибір Wi-Fi є компромісним рішенням: попри вище енергоспоживання порівняно з технологіями далекого зв'язку, він забезпечує достатню швидкість для передавання телеметрії, не потребує розгортання окремої інфраструктури за наявності точок доступу та має вбудовану підтримку у мікроконтролері. Підвищене енергоспоживання компенсується застосуванням режиму глибокого сну мікроконтролера між циклами вимірювання, коли радіомодуль вимкнено.

Слід зауважити, що архітектура підсистеми не виключає застосування інших технологій зв'язку для окремих вузлів. Для вузлів, розміщених у зоні дії Wi-Fi, доцільним є використання цієї технології, тоді як для віддалених вузлів на відкритих територіях може застосовуватися технологія далекого зв'язку LoRaWAN. Оскільки на рівні обміну повідомленнями використовується протокол MQTT, незалежний від технології канального рівня, поєднання різних технологій зв'язку у межах однієї підсистеми не потребує зміни серверної частини, що підвищує гнучкість розгортання.

Передавання даних у підсистемі здійснюється на основі стеку протоколів TCP/IP. На канальному та фізичному рівнях застосовується технологія Wi-Fi (стандарт IEEE 802.11), на мережевому – протокол IP, на транспортному – протокол TCP, що гарантує надійне доставлення даних. Прикладний рівень реалізовано протоколами MQTT та HTTP. Така багаторівнева організація відповідає принципам побудови комп'ютерних мереж та забезпечує сумісність з

наявною мережевою інфраструктурою, оскільки використовуються стандартні загальноприйняті протоколи [25].

1.3.2 Протоколи прикладного рівня

Протоколи прикладного рівня визначають правила обміну даними між програмними компонентами підсистеми. На відміну від протоколів нижчих рівнів, що забезпечують доставлення даних між вузлами мережі, протоколи прикладного рівня визначають структуру та семантику переданих повідомлень. Вибір протоколу прикладного рівня суттєво впливає на ефективність, надійність та масштабованість обміну даними у системах інтернету речей.

Для передавання телеметрії у системах інтернету речей найпоширенішими є протоколи HTTP/REST та MQTT. Протокол MQTT побудований за моделлю «видавець – передплатник» (publish/subscribe) через центрального брокера, характеризується малим обсягом службових даних та низьким енергоспоживанням, що робить його оптимальним для вузлів з обмеженими ресурсами. Протокол HTTP/REST зручний для взаємодії веб-клієнта із сервером під час отримання історичних даних та налаштувань. У підсистемі застосовано комбінований підхід: вузли публікують дані через MQTT, а веб-клієнт отримує їх через REST API сервера [8].

Протокол MQTT передбачає три рівні якості обслуговування (QoS): рівень 0 – доставлення без підтвердження (не більше одного разу), рівень 1 – доставлення з підтвердженням (принаймні один раз) та рівень 2 – гарантоване одноразове доставлення. Для передавання даних моніторингу обрано рівень QoS 1, що гарантує доставлення кожного повідомлення та водночас не створює надмірного навантаження на канал зв'язку, характерного для рівня QoS 2. Можливі повторні доставлення повідомлень обробляються на сервері шляхом виявлення дублікатів за міткою часу.

Архітектурний стиль REST, застосований для взаємодії веб-клієнта із сервером, ґрунтується на поданні даних у вигляді ресурсів, доступних за унікальними адресами, та використанні стандартних методів протоколу HTTP для

операцій над ними. Перевагами REST є простота, незалежність від мови програмування клієнта, можливість кешування відповідей та масштабованість. Поєднання MQTT для приймання даних від вузлів та REST для їх надання клієнту дає змогу використати сильні сторони обох протоколів [9,24].

Доцільність комбінованого підходу пояснюється відмінністю вимог до двох напрямів обміну даними. Передавання даних від численних автономних вузлів потребує економного за ресурсами протоколу з підтримкою асинхронної моделі та гарантованого доставлення, чому відповідає MQTT. Натомість взаємодія веб-клієнта із сервером має характер запит-відповідь та потребує зручного доступу до структурованих даних за різними критеріями, чому краще відповідає REST. Застосування кожного протоколу за його призначенням забезпечує ефективність підсистеми загалом та відповідає сучасній практиці побудови систем інтернету речей [24].

Дані передаються у форматі JSON, що забезпечує структурованість, читабельність та простоту розбору як на боці сервера, так і на боці клієнта.

Модель взаємодії «видавець – передплатник», на якій ґрунтується MQTT, має суттєву перевагу для систем моніторингу: вузли-видавці не мають знати про конкретних споживачів даних, а лише публікують повідомлення у відповідні теми (topics) брокера. Це забезпечує слабку зв'язаність компонентів та спрощує масштабування – додавання нових вузлів або споживачів не потребує зміни наявних. Брокер також бере на себе функції тимчасового зберігання повідомлень та контролю якості доставлення за рахунок механізму рівнів QoS, що дає змогу гарантувати доставлення кожного повідомлення принаймні один раз.

Для оброблення накопичених даних застосовується пакетний підхід: первинні виміри зберігаються у базі даних, після чого над ними можуть виконуватися операції агрегування, усереднення за інтервалами часу, виявлення трендів та порівняння з граничними значеннями. Такий поділ на оперативне передавання та подальше аналітичне оброблення є характерним для систем моніторингу та забезпечує як оперативність реагування, так і можливість ретроспективного аналізу.

1.3.3 Методи оброблення та подання даних моніторингу

Дані екологічного моніторингу проходять кілька стадій оброблення. На стадії первинного оброблення, що виконується на вузлі, здійснюється фільтрація шумів, усереднення відліків та перетворення сигналів датчиків у фізичні величини. На стадії вторинного оброблення, що виконується на сервері, дані перевіряються на достовірність, порівнюються з граничними значеннями та агрегуються за часовими інтервалами для зменшення обсягу та виявлення тенденцій.

Важливим завданням оброблення є контроль достовірності даних. Недостовірні значення, що виникають унаслідок збоїв датчиків або завад, можуть бути виявлені шляхом перевірки на належність до фізично можливого діапазону та на різке відхилення від попередніх значень. Виявлені недостовірні значення позначаються відповідною ознакою та виключаються з аналітичного оброблення, що підвищує якість підсумкових даних.

Подання даних користувачеві здійснюється у кількох формах: у вигляді поточних числових значень показників, у вигляді графіків динаміки показників за обраний період та у вигляді карти просторового розподілу. Графічне подання є найбільш наочним та дає змогу швидко оцінити стан довкілля й виявити несприятливі тенденції. Поєднання різних форм подання забезпечує зручність роботи з підсистемою для різних категорій користувачів.

1.3.4 Формати подання даних

Важливим аспектом обміну даними між компонентами підсистеми є вибір формату їх подання. До поширених форматів обміну даними у системах інтернету речей належать текстові формати JSON та XML, а також бінарні формати. Текстові формати є самоописовими та зручними для читання й налагодження, проте мають більший обсяг порівняно з бінарними. Бінарні формати компактніші, але потребують узгодженої схеми даних та ускладнюють налагодження.

Для розроблюваної підсистеми обрано формат JSON, що поєднує самоописовість, читабельність та достатню компактність для обсягів даних,

характерних для телеметрії. Формат JSON має широку підтримку у мовах програмування та середовищах розроблення, що спрощує реалізацію розбору й формування повідомлень на всіх рівнях підсистеми. Самоописовість формату дає змогу легко розширювати набір переданих показників без зміни структури повідомлення загалом, що є важливим для подальшого розвитку підсистеми [10].

Порівняно з форматом XML, що також є самоописовим текстовим форматом, JSON має простіший синтаксис та менший обсяг службових даних, що робить його кращим вибором для передавання телеметрії від вузлів з обмеженими ресурсами. Застосування єдиного формату на всіх рівнях підсистеми – від вузла до веб-клієнта – забезпечує узгодженість обміну даними та спрощує розроблення, оскільки усуває потребу у перетворенні даних між різними форматами на межах компонентів [23].

1.4 Постановка задачі та технічне завдання

На підставі проведеного аналізу сформульовано постановку задачі: розробити інформаційну підсистему моніторингу стану природного довкілля, що складається з територіально рознесених вузлів збору даних на базі мікроконтролера ESP32 та комплекту датчиків, центрального сервера з базою даних і MQTT-брокером, а також веб-додатку для візуалізації даних.

Перед формулюванням технічного завдання виконано аналіз вимог до підсистеми з урахуванням потреб потенційних користувачів та умов експлуатації. Користувачами підсистеми є фахівці екологічних служб, адміністрації підприємств та установ, а також, у разі публічного доступу, населення. Кожна категорія користувачів має власні потреби: фахівцям потрібні детальні дані та можливість аналізу, адміністрації – узагальнені показники та попередження, населенню – наочна інформація про поточний стан довкілля. Підсистема має

задовольняти потреби всіх категорій за рахунок гнучкого подання даних та розмежування доступу.

Аналіз умов експлуатації показав, що вузли підсистеми можуть розміщуватися як у приміщеннях, так і на відкритому повітрі, зокрема на віддалених територіях без доступу до електромережі та зі змінним покриттям мережею Wi-Fi. Це зумовлює вимоги до автономності живлення, стійкості до погодних умов та надійності передавання даних за нестабільного зв'язку. Зазначені вимоги враховано під час формування технічного завдання та проектування підсистеми.

Узагальнену структуру підсистеми, що відображає основні елементи та зв'язки між ними, наведено на рисунку 1.1.

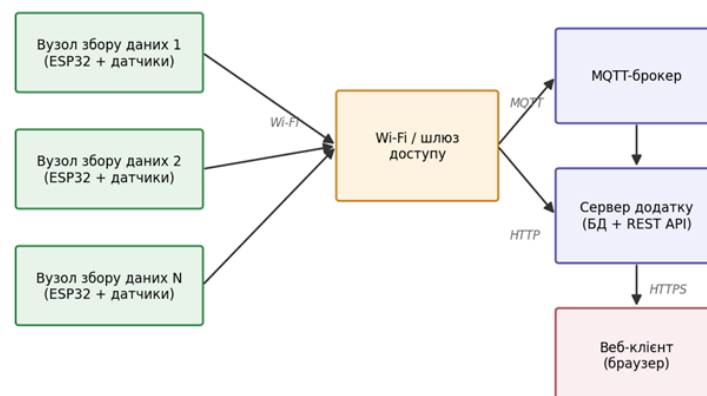


Рисунок 1.1 – Узагальнена структурна схема підсистеми

Технічне завдання на розробку підсистеми сформульовано відповідно до структури, наведеної у додатку Л методичних вказівок, і містить такі основні вимоги.

Підсистема призначена для автоматизованого контролю стану природного довкілля на території об'єкта спостереження. Вона має забезпечувати збирання даних від територіально рознесених вузлів, їх централізоване оброблення,

зберігання та подання користувачеві. Підсистема розрахована на цілодобову безперервну роботу з мінімальним втручанням обслуговного персоналу.

Вимоги до функцій підсистеми:

- підсистема має забезпечувати періодичне вимірювання показників якості повітря, води та ґрунту з настроюваним інтервалом опитування;
- передавання даних від вузлів до сервера має виконуватися автоматично без втручання оператора;
- у разі тимчасової недоступності каналу зв'язку вузол має зберігати дані у локальному буфері з подальшим відправленням;
- сервер має забезпечувати накопичення даних у базі даних та надання їх через програмний інтерфейс;
- веб-додаток має відображати поточні значення, динаміку показників у вигляді графіків та розташування вузлів на карті.

Вимоги до інформаційного забезпечення:

- дані мають зберігатися у структурованому вигляді у реляційній базі даних з можливістю вибірки за вузлом, датчиком та періодом часу;
- кожне виміряне значення має супроводжуватися міткою часу та ознакою достовірності;
- обмін даними між компонентами має здійснюватися у відкритому форматі JSON.

Вимоги до інтерфейсу користувача:

- інтерфейс має бути веб-орієнтованим та доступним через стандартний браузер без встановлення додаткового програмного забезпечення;
- інтерфейс має забезпечувати наочне подання поточного стану довкілля та зручну навігацію за історичними даними;
- відображення попереджень про перевищення граничних значень має бути візуально вирізненим.

Вимоги до надійності та умов експлуатації:

- вузли мають зберігати працездатність у діапазоні температур від мінус 10 до плюс 45 °С;

- підсистема має передбачати автономне живлення вузлів від акумулятора із сонячною панеллю;
- передавання даних користувачеві має здійснюватися захищеним каналом (HTTPS/TLS);
- підсистема має допускати збільшення кількості вузлів без зміни архітектури сервера.

Вимоги до програмного забезпечення:

- вбудоване програмне забезпечення вузла має реалізовувати зчитування датчиків, первинне оброблення, формування повідомлень, буферизацію та передавання даних;
- серверне програмне забезпечення має забезпечувати приймання, оброблення та збереження даних, а також надання їх через програмний інтерфейс;
- програмне забезпечення має бути модульним та допускати розвиток без суттєвої переробки наявних компонентів.

Вимоги до безпеки інформації:

- доступ користувачів до підсистеми має регламентуватися механізмом автентифікації;
- передавання даних користувачеві має здійснюватися захищеним каналом;
- паролі користувачів мають зберігатися у захищеному вигляді.

1.5 Обґрунтування архітектурних рішень

На основі сформульованих вимог обрано архітектурний стиль побудови підсистеми. Розглянуто два альтернативні підходи: монолітну архітектуру, за якої всі функції зосереджено в єдиному застосунку, та розподілену архітектуру з виділеним рівнем обміну повідомленнями. Перший підхід простіший у реалізації, проте погано масштабується та має низьку стійкість до збоїв окремих компонентів. Другий підхід складніший, але забезпечує слабку зв'язаність

компонентів, масштабованість та відмовостійкість, тому саме його покладено в основу підсистеми.

Порівняння розглянутих архітектурних підходів за основними критеріями наведено у таблиці 1.4. Оцінювання виконано якісно з огляду на специфіку завдань екологічного моніторингу.

Таблиця 1.4 – Порівняння архітектурних підходів

Критерій	Монолітна	Розподілена
Складність реалізації	Низька	Середня
Масштабованість	Низька	Висока
Відмовостійкість	Низька	Висока
Слабка зв'язаність	Ні	Так
Простота розвитку	Низька	Висока

Як видно з таблиці, розподілена архітектура переважає за більшістю критеріїв, важливих для систем моніторингу, поступаючись лише у простоті початкової реалізації. З огляду на вимоги масштабованості та відмовостійкості, сформульовані у технічному завданні, обрано саме розподілену архітектуру з виділеним рівнем обміну повідомленнями на основі MQTT-брокера.

Підсистема будується за трирівневою схемою. Перший рівень – рівень збору даних – утворюють вузли вимірювання, що виконують зчитування показників та їх передавання. Другий рівень – рівень обміну та оброблення – охоплює MQTT-брокер та сервер з базою даних, що приймає, обробляє та зберігає дані. Третій рівень – рівень подання – реалізовано веб-додатком, що відображає дані користувачеві. Такий поділ на рівні забезпечує незалежність розроблення та можливість заміни окремих компонентів без впливу на інші.

Важливим архітектурним принципом є масштабованість підсистеми. Завдяки застосуванню моделі «видавець – передплатник» додавання нового вузла зводиться до його налаштування на публікацію у відповідну тему брокера та реєстрації у базі даних; жодних змін у наявних компонентах при цьому не потрібно. Аналогічно, за зростання навантаження сервер та брокер можуть бути

перенесені на потужніше обладнання або масштабовані горизонтально. Це задовольняє вимогу технічного завдання щодо збільшення кількості вузлів без зміни архітектури.

Ще одним принципом є відмовостійкість, що забезпечується дворівневим механізмом захисту від втрати даних: локальною буферизацією на вузлах та проміжним зберіганням повідомлень брокером. Завдяки цьому тимчасова недоступність каналу зв'язку або сервера не призводить до втрати вимірюваних даних, а лише до відкладеного їх доставлення, що є прийнятним для завдань екологічного моніторингу.

1.6 Техніко-економічне обґрунтування

Однією з ключових переваг розроблюваної підсистеми є її низька вартість порівняно зі стаціонарними системами моніторингу. Орієнтовну вартість одного вузла збору даних формують вартість мікроконтролера, комплекту датчиків, елементів живлення та корпусу. Сумарна вартість компонентів одного вузла є на кілька порядків нижчою за вартість стаціонарного аналізатора, що уможливорює розгортання значної кількості вузлів у межах обмеженого бюджету.

Економічна доцільність підсистеми визначається співвідношенням витрат на її створення та експлуатацію з корисним ефектом від отриманої інформації. До витрат належать одноразові витрати на придбання компонентів та розроблення програмного забезпечення, а також періодичні витрати на обслуговування та електроенергію. Корисний ефект полягає у можливості своєчасного виявлення несприятливих змін стану довкілля, що дає змогу запобігти збиткам для здоров'я населення та довкілля, а також обґрунтовано планувати природоохоронні заходи.

Застосування відкритого програмного забезпечення та доступних апаратних компонентів дає змогу мінімізувати витрати на створення підсистеми. Серверну частину може бути розгорнуто на наявному обладнанні організації, що додатково

знижує витрати. Завдяки автономному живленню вузлів від акумулятора із сонячною панеллю експлуатаційні витрати на електроенергію є мінімальними. Зазначені чинники свідчать про економічну доцільність розроблення та впровадження підсистеми.

Орієнтовну структуру витрат на створення підсистеми наведено у таблиці 1.5. Витрати поділено на капітальні (одноразові) та експлуатаційні (періодичні). Значення наведено у відносних одиницях для порівняння часток окремих складових без прив'язки до конкретних цін, що змінюються у часі.

Таблиця 1.5 – Орієнтовна структура витрат

Стаття витрат	Тип	Частка, %
Компоненти вузлів	капітальні	45
Серверне обладнання	капітальні	20
Розроблення ПЗ	капітальні	25
Обслуговування	експлуатаційні	7
Електроенергія, зв'язок	експлуатаційні	3

З наведеної структури видно, що основну частку капітальних витрат становить вартість компонентів вузлів, яка прямо пропорційна їх кількості. Це підкреслює важливість вибору недорогої елементної бази для забезпечення економічної доступності підсистеми за масштабного розгортання. Експлуатаційні витрати є незначними завдяки автономності вузлів та використанню наявної мережевої інфраструктури, що робить підсистему економічно привабливою у тривалій перспективі.

У розділі розглянуто призначення та завдання моніторингу стану природного довкілля, виконано огляд і порівняльний аналіз систем-аналогів, проаналізовано технології бездротового зв'язку та протоколи передавання даних. Обґрунтовано вибір архітектури на основі недорогих вузлів ESP32, технології Wi-Fi та протоколу MQTT у поєднанні з REST API. Сформульовано постановку задачі та технічне завдання на розробку підсистеми, що є основою для подальшого проектування.

Проведений аналіз показав, що традиційні стаціонарні системи моніторингу, попри високу точність, не забезпечують потрібної просторової щільності спостережень за прийнятною вартістю, тоді як комерційні хмарні мережі недорогих датчиків обмежують контроль користувача над даними. Це обґрунтовує доцільність розроблення власної підсистеми на основі недорогих вузлів з власним серверним застосунком, що поєднує переваги обох підходів. Обрані технічні рішення – мікроконтролер ESP32, технологія Wi-Fi, протоколи MQTT і REST, формат JSON та реляційна база даних – є усталеними, добре підтримуваними та сумісними між собою, що знижує ризики реалізації та забезпечує можливість подальшого розвитку підсистеми.

2 РОЗРОБКА АПАРАТНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Розробка структурної та функціональної схем

Проектування підсистеми розпочинається з розроблення її структурної схеми, що визначає склад основних компонентів та зв'язки між ними, а потім функціональних схем окремих компонентів, що деталізують їх внутрішню організацію та порядок взаємодії елементів. Структурна схема відображає загальну організацію підсистеми на рівні функціональних блоків, тоді як функціональна схема розкриває реалізацію функцій окремого блока.

Узагальнену структурну схему підсистеми було наведено у розділі 1 (рисунок 1.1). Вона відображає три рівні організації: рівень збору даних, що утворюють вузли вимірювання; рівень обміну та оброблення, що охоплює MQTT-брокер і сервер з базою даних; та рівень подання, реалізований веб-додатком. Зв'язки між рівнями реалізовано стандартними протоколами: вузли передають дані шлюзу мережею Wi-Fi, шлюз спрямовує їх до брокера протоколом MQTT, сервер взаємодіє з брокером та надає дані веб-клієнту захищеним протоколом HTTPS.

Кількість вузлів збору даних у структурній схемі не обмежена та визначається потребами конкретного об'єкта моніторингу. Усі вузли є однотипними за структурою та відрізняються лише місцем розміщення й набором встановлених датчиків, що спрощує тиражування та обслуговування підсистеми. Шлюз доступу може бути реалізований як наявною точкою доступу Wi-Fi, так і спеціалізованим пристроєм залежно від умов розгортання. Сервер та брокер можуть розміщуватися як на одному фізичному обладнанні, так і на окремих машинах, що забезпечує гнучкість розгортання залежно від навантаження.

Розроблювана підсистема має дворівневу клієнт-серверну архітектуру з проміжним рівнем обміну повідомленнями. Нижній рівень утворюють вузли збору даних, які виконують вимірювання та публікують результати; проміжний рівень реалізовано MQTT-брокером, що забезпечує розв'язання вузлів та сервера в часі; верхній рівень становить сервер з базою даних та веб-додаток.

Функціональну схему окремого вузла збору даних наведено на рисунку 2.1. Вузол об'єднує комплект датчиків, мікроконтролер ESP32, модуль зв'язку та джерело живлення.

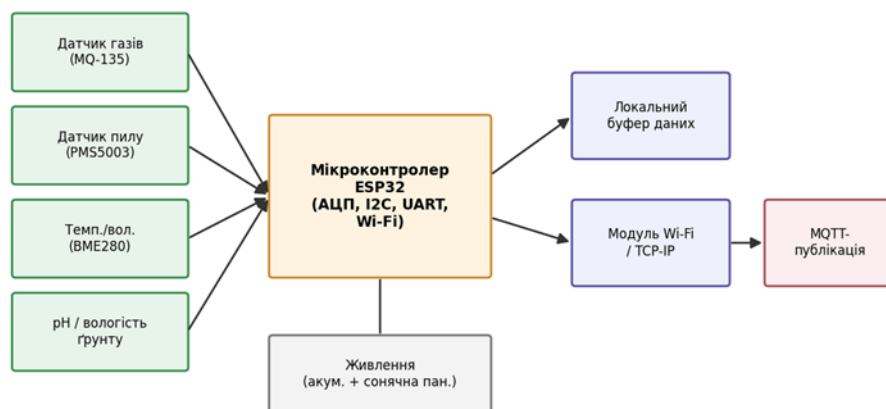


Рисунок 2.1 – Функційна схема вузла збору даних

Мікроконтролер циклічно опитує датчики через відповідні інтерфейси (АЦП для аналогових датчиків газів, шину I2C для цифрового датчика

температури, вологості й тиску, інтерфейс UART для оптичного датчика пилю), виконує первинне оброблення сигналів, формує повідомлення у форматі JSON та передає його через модуль Wi-Fi до MQTT-брокера. За відсутності зв'язку дані тимчасово зберігаються у локальному буфері.

Розподіл функцій між рівнями підсистеми обрано так, щоб мінімізувати обчислювальне навантаження на вузли та забезпечити централізоване оброблення й зберігання даних. На рівні вузла виконуються лише операції, що не потребують значних ресурсів: вимірювання, первинна фільтрація та перетворення сигналів, формування повідомлень та керування передаванням. Ресурсомісткі операції – накопичення великих обсягів даних, аналітичне оброблення, побудова графіків та перевірка складних правил – покладено на сервер. Такий розподіл відповідає принципу «тонкого» вузла та «товстого» сервера, що є типовим для систем інтернету речей.

Обґрунтуванням такого розподілу є обмеженість обчислювальних ресурсів та енергії вузлів порівняно з сервером. Виконання ресурсомістких операцій на вузлі призвело б до зростання енергоспоживання та скорочення тривалості автономної роботи, а також ускладнило б програмне забезпечення вузла. Натомість сервер не має таких обмежень та може ефективно обробляти дані багатьох вузлів. Винесення логіки оброблення на сервер також спрощує її модифікацію: зміна правил перевірки або алгоритмів аналізу не потребує оновлення програмного забезпечення всіх вузлів, що є суттєвою перевагою за масштабного розгортання підсистеми.

Проміжний рівень, реалізований MQTT-брокером, виконує роль буфера між вузлами та сервером. Він забезпечує приймання повідомлень від вузлів незалежно від готовності сервера їх обробити, що підвищує стійкість підсистеми до тимчасових збоїв. У разі короткочасної недоступності сервера повідомлення зберігаються брокером і доставляються після відновлення з'єднання, що разом із локальною буферизацією на вузлах утворює дворівневий механізм захисту від втрати даних.

Функціональна схема вузла, наведена на рисунку 2.1, відображає внутрішню організацію вузла та потоки даних між його елементами. Чутливі елементи (датчики) перетворюють фізичні та хімічні величини у електричні сигнали, які надходять до мікроконтролера відповідними інтерфейсами. Мікроконтролер виконує перетворення сигналів, їх оброблення та формування повідомлень, після чого через модуль зв'язку передає дані до брокера. Підсистема живлення забезпечує електроживлення всіх елементів вузла.

Потоки даних у межах вузла організовано так, щоб мінімізувати час активної роботи та, відповідно, енергоспоживання. Зчитування всіх датчиків виконується послідовно одразу після пробудження вузла, після чого зібрані дані об'єднуються в єдине повідомлення. Це дає змогу виконати лише один сеанс передавання за цикл замість окремого передавання кожного показника, що скорочує час роботи радіомодуля та обсяг службових даних. Після успішного передавання вузол одразу переходить у режим сну, не очікуючи додаткових подій, що додатково знижує споживання енергії.

2.2 Вибір та обґрунтування елементної бази

Елементна база вузла збору даних визначає його функціональні можливості, точність вимірювань, енергоспоживання та вартість. До складу вузла входять мікроконтролер, комплект датчиків, модуль зв'язку та підсистема живлення. Вибір кожного з елементів виконано на основі аналізу вимог технічного завдання та порівняння доступних варіантів.

Загальними вимогами до елементної бази є доступність компонентів на ринку, помірна вартість, наявність технічної документації та підтримки у середовищі розроблення, а також сумісність елементів між собою. Перевага надається компонентам, що набули широкого поширення та мають підтверджену надійність, оскільки це знижує ризики реалізації та спрощує подальше

тиражування й обслуговування вузлів. Усі обрані компоненти задовольняють зазначені вимоги та є доступними для придбання, що забезпечує можливість практичного відтворення розробленого вузла.

2.2.1 Вибір мікроконтролера

Центральним елементом вузла є мікроконтролер. До нього висуваються вимоги: наявність вбудованого модуля Wi-Fi, достатня кількість аналогових і цифрових каналів вводу-виводу, підтримка інтерфейсів I2C та UART, низьке енергоспоживання та доступна вартість. Порівняння поширених мікроконтролерних платформ наведено у таблиці 2.1.

Таблиця 2.1 – Порівняння мікроконтролерних платформ

Параметр	Arduino Uno	ESP8266	ESP32
Розрядність ядра	8 біт	32 біт	32 біт
Тактова частота	16 МГц	80 МГц	240 МГц
Вбудований Wi-Fi	Ні	Так	Так
Кількість каналів АЦП	6	1	18
ОЗП	2 КБ	80 КБ	520 КБ
Орієнтовна вартість	Середня	Низька	Низька

За результатами порівняння обрано мікроконтролер ESP32. Він має вбудований модуль Wi-Fi, два процесорні ядра, достатній обсяг пам'яті для буферизації даних та значну кількість аналогових каналів, що дає змогу під'єднати весь необхідний комплект датчиків без додаткових розширювачів портів [6].

Додатковими перевагами ESP32, що зумовили його вибір, є наявність апаратної підтримки криптографічних операцій, потрібних для організації захищеного передавання даних, підтримка кількох режимів зниженого енергоспоживання, а також розвинене середовище розроблення з відкритими бібліотеками для роботи з датчиками, протоколами Wi-Fi та MQTT. Мікроконтролер підтримує програмування у середовищі Arduino, що значно спрощує розроблення та налагодження програмного забезпечення вузла.

Наявність вбудованого годинника реального часу та можливості синхронізації часу мережею забезпечує коректну прив'язку вимірювань до часової шкали.

Порівняння з мікроконтролером ESP8266, що також має вбудований модуль Wi-Fi, показало переваги ESP32 за кількістю аналогових каналів, обсягом пам'яті та обчислювальною продуктивністю за збісованої вартості. Платформа Arduino Uno, попри простоту та поширеність, не має вбудованого модуля Wi-Fi та має обмежений обсяг пам'яті, що унеможлиблює реалізацію буферизації даних та потребує додаткових модулів зв'язку. З огляду на це вибір ESP32 є обґрунтованим компромісом між функціональними можливостями, енергоспоживанням та вартістю, що повною мірою задовольняє вимоги до вузла збору даних розроблюваної підсистеми.

2.2.2 Вибір датчиків

Комплект датчиків обрано виходячи з контрольованих показників, визначених у розділі 1. Перелік обраних датчиків та їхні характеристики наведено у таблиці 2.2.

Таблиця 2.2 – Комплект датчиків вузла

Датчик	Показник	Інтерфейс	Діапазон
PMS5003	PM2.5, PM10	UART	0–500 мкг/м³
MQ-135	CO ₂ , NO _x , VOC	АЦП	10–1000 ppm
BME280	Т, вол., тиск	I2C	–40...85 °C
Аналог. рН-модуль	рН води	АЦП	0–14 рН
Ємнісний датчик	вологість ґрунту	АЦП	0–100 %

Під час вибору датчиків враховувалися такі критерії: відповідність вимірюваної величини контрольованому показнику, прийнятний діапазон та точність вимірювання, тип вихідного інтерфейсу та сумісність з мікроконтролером, енергоспоживання, вартість та доступність на ринку. Перевага надавалася датчикам з цифровим інтерфейсом, що передають готові значення, оскільки вони знижують навантаження на мікроконтролер та зменшують вплив

завад порівняно з передаванням аналогового сигналу. Водночас для показників, для яких цифрові датчики є надто дорогими або недоступними, застосовано аналогові датчики з під'єднанням до АЦП.

Оптичний датчик PMS5003 застосовано для вимірювання концентрації твердих частинок, оскільки він має вбудований вентилятор і лазерний випромінювач та видає цифрове значення концентрації через UART, що знижує навантаження на мікроконтролер. Датчик BME280 обрано для одночасного вимірювання температури, відносної вологості та атмосферного тиску цифровим інтерфейсом I2C. Аналогові датчики під'єднано до каналів АЦП мікроконтролера [11,12].

Принцип дії оптичного датчика пилу ґрунтується на вимірюванні розсіювання лазерного променя частинками, що проходять через вимірювальну камеру разом із потоком повітря, створюваним вентилятором. За інтенсивністю та розподілом розсіяного світла визначається кількість та розмір частинок, на основі чого обчислюється масова концентрація PM2.5 і PM10. Перевагою такого датчика є відсутність потреби у первинному обробленні сигналу на мікроконтролері, оскільки готові значення концентрації передаються у цифровій формі.

Оптичний датчик пилу передає дані у вигляді кадрів фіксованого формату, що містять значення концентрації частинок різних розмірних фракцій, контрольну суму та службові поля. Мікроконтролер приймає ці кадри інтерфейсом UART, перевіряє контрольну суму для виявлення помилок передавання та виокремлює потрібні значення концентрації. Застосування контрольної суми підвищує достовірність даних, оскільки кадри з помилками відкидаються та не потрапляють до подальшого оброблення. Вбудований вентилятор датчика забезпечує стабільний потік повітря через вимірювальну камеру, що є необхідною умовою коректності вимірювань.

Газовий датчик MQ-135 належить до напівпровідникових датчиків, опір чутливого шару яких змінюється залежно від концентрації газів. Цей датчик чутливий до низки забруднювальних речовин, зокрема діоксиду вуглецю, аміаку та летких органічних сполук. Оскільки напівпровідникові датчики мають

нелінійну характеристику та залежність від температури й вологості, для отримання достовірних значень застосовано програмну корекцію та калібрування за відомими концентраціями. Вихідний аналоговий сигнал датчика подається на канал АЦП мікроконтролера [13].

Принцип дії напівпровідникового газового датчика ґрунтується на зміні провідності оксиду металу під дією газів-відновників за підвищеної температури чутливого елемента. Для забезпечення робочої температури датчик містить вбудований нагрівальний елемент, що споживає певну потужність, яку враховано під час розрахунку енергоспоживання вузла. Час виходу датчика на робочий режим після ввімкнення становить кілька десятків секунд, тому в алгоритмі роботи вузла передбачено відповідну затримку перед зчитуванням показань для забезпечення достовірності вимірювань.

Розрядність вбудованого АЦП мікроконтролера ESP32 становить 12 біт, що забезпечує 4096 рівнів квантування у діапазоні входної напруги. Похибка квантування для аналогових датчиків за такої розрядності не перевищує однієї тисячної частки діапазону, що є достатнім для завдань екологічного моніторингу з огляду на власну похибку самих датчиків, яка зазвичай значно більша.

Датчик BME280 є комбінованим цифровим датчиком, що в одному корпусі поєднує чутливі елементи температури, відносної вологості та атмосферного тиску. Він обмінюється даними з мікроконтролером цифровим інтерфейсом I2C, передаючи вже відкалібровані значення, що спрощує програмне забезпечення вузла. Окрім самостійного значення метеорологічних параметрів, дані цього датчика використовуються для корекції показань інших датчиків, чутливих до температури та вологості, що підвищує загальну достовірність вимірювань вузла.

Для контролю стану водного середовища застосовано аналоговий модуль вимірювання водневого показника рН, що перетворює різницю потенціалів вимірювального електрода у напругу, пропорційну кислотності розчину. Вологість ґрунту вимірюється ємнісним датчиком, принцип дії якого ґрунтується на зміні діелектричної проникності ґрунту залежно від вмісту води. Ємнісні датчики мають перевагу над резистивними завдяки відсутності прямого контакту

електродів з ґрунтом, що подовжує термін їх служби та підвищує стабільність показань.

Вимірювання водневого показника потребує періодичного калібрування електрода за буферними розчинами з відомими значеннями рН, оскільки характеристика електрода змінюється з часом. Аналоговий сигнал рН-модуля подається на канал АЦП мікроконтролера, після чого перетворюється у значення рН за калібрувальною залежністю. Ємнісний датчик вологості ґрунту також потребує калібрування для конкретного типу ґрунту, оскільки діелектрична проникність залежить не лише від вологості, а й від складу ґрунту. Калібрування виконується шляхом вимірювання показань за відомих значень вологості, зокрема для сухого та насиченого вологою ґрунту.

Слід зазначити, що датчики контролю води та ґрунту, на відміну від датчиків якості повітря, можуть експлуатуватися періодично, оскільки показники цих середовищ змінюються повільніше. Це дає змогу зменшити енергоспоживання вузла за рахунок рідшого опитування відповідних датчиків. Гнучкість налаштування інтервалів опитування для різних груп датчиків є важливою особливістю розроблюваного вузла, що дає змогу оптимізувати співвідношення між інформативністю та енергоспоживанням залежно від контрольованого середовища.

2.2.3 Оціночні розрахунки

Перед виконанням розрахунків енергоспоживання доцільно розглянути режими роботи вузла, що визначають його споживання. Вузол функціонує у двох основних режимах: активному, коли виконується вимірювання, оброблення та передавання даних, та режимі глибокого сну, коли більшість підсистем вимкнено. Споживання у цих режимах відрізняється на кілька порядків, тому середнє споживання суттєво залежить від співвідношення тривалості активної фази та фази сну, яке визначається інтервалом опитування.

Для підтвердження працездатності обраного варіанту виконано оціночні розрахунки енергоспоживання вузла та обсягу переданих даних. Середній струм споживання вузла визначається сумою струмів окремих компонентів:

$$I_{\text{сер}} = I_{\text{МК}} + \Sigma I_{\text{дат}} + I_{\text{Wi-Fi}}, \quad (2.1)$$

де $I_{\text{МК}}$ – струм споживання мікроконтролера, мА;

$\Sigma I_{\text{дат}}$ – сумарний струм датчиків, мА;

$I_{\text{Wi-Fi}}$ – струм у режимі передавання, мА.

За паспортними даними компонентів у режимі вимірювання вузол споживає близько 180 мА, а у режимі глибокого сну – менше 1 мА. За інтервалу опитування 5 хвилин та тривалості активної фази близько 10 секунд середньодобове споживання становить приблизно 12 мА·год на цикл, що дає змогу жити вузол від акумулятора ємністю 2000 мА·год протягом кількох діб навіть без підзаряджання сонячною панеллю. Обсяг одного JSON-повідомлення не перевищує 300 байт, тому добовий трафік одного вузла не перевищує 100 КБ, що є несуттєвим для типового Wi-Fi-каналу.

Найбільший внесок у споживання активної фази робить радіомодуль Wi-Fi під час передавання даних, а також нагрівальний елемент газового датчика. Тому ключовим засобом зниження середнього споживання є скорочення тривалості активної фази та максимальне використання режиму сну. За збільшення інтервалу опитування частка часу, що вузол перебуває у режимі сну, зростає, а середнє споживання знижується, що подовжує тривалість автономної роботи. Це дає змогу гнучко налаштовувати вузол залежно від пріоритету: за потреби частого оновлення даних обирається менший інтервал опитування, за пріоритету тривалої автономності – більший.

Тривалість автономної роботи вузла від акумулятора без підзаряджання можна оцінити за співвідношенням:

$$T_{\text{авт}} = Q / I_{\text{сер}}, \quad (2.2)$$

де Q – ємність акумулятора, $\text{мА} \cdot \text{год}$;

$I_{\text{сер}}$ – середній струм споживання, мА .

За середнього струму споживання близько 6 мА (з урахуванням переважання режиму сну) та ємності акумулятора $2000 \text{ мА} \cdot \text{год}$ розрахункова тривалість автономної роботи перевищує 13 діб. Застосування сонячної панелі потужністю кілька ват забезпечує практично необмежену тривалість роботи за умови достатньої освітленості, що підтверджує доцільність обраного варіанту живлення.

Для забезпечення стабільної роботи від сонячної панелі необхідно узгодити її потужність із середнім споживанням вузла та умовами освітленості у регіоні розгортання. Сонячна панель має виробляти за світловий день кількість енергії, достатню для живлення вузла протягом доби та підзаряджання акумулятора для роботи у темний час та у похмурі дні. Акумулятор виконує роль буфера енергії, згладжуючи нерівномірність надходження сонячної енергії. Розрахунок параметрів підсистеми живлення виконується з урахуванням запасу на несприятливі погодні умови, що забезпечує безперебійну роботу вузла протягом тривалого часу без обслуговування.

Окремо оцінено навантаження на сервер за повномасштабного розгортання підсистеми. За інтервалу опитування 5 хвилин один вузол формує 288 повідомлень на добу. Для мережі зі 100 вузлів сумарна кількість повідомлень становить близько $28,8$ тисяч на добу, або в середньому менше одного повідомлення за секунду. Таке навантаження є незначним для сучасного серверного обладнання та звичайного MQTT-брокера, що підтверджує здатність підсистеми обслуговувати велику кількість вузлів без ускладнення архітектури.

Обсяг даних, що накопичуються у базі даних, також піддається оцінюванню. За припущення, що один запис вимірювання займає близько 30 байт, мережа зі 100 вузлів за рік формує приблизно один гігабайт даних. Цей обсяг легко обслуговується сучасними системами керування базами даних, а за потреби

довготривалого зберігання історичні дані можуть бути агреговані або перенесені до архівного сховища.

Наведені оціночні розрахунки підтверджують, що обрані технічні рішення забезпечують працездатність підсистеми як на рівні окремого вузла, так і на рівні системи загалом. Енергоспоживання вузла дає змогу реалізувати автономне живлення, обсяг трафіку є несуттєвим для каналу зв'язку, а навантаження на сервер та обсяг даних залишаються прийнятними навіть за значної кількості вузлів. Це свідчить про масштабованість підсистеми та її придатність до розгортання мереж спостереження різного масштабу без зміни базової архітектури.

2.2.4 Організація інтерфейсів та підсистеми живлення

Під'єднання датчиків до мікроконтролера організовано з урахуванням типу їх вихідного сигналу та інтерфейсу. Цифровий датчик BME280 під'єднано до шини I2C, що дає змогу обмінюватися даними лише двома сигнальними лініями та за потреби під'єднувати кілька пристроїв до однієї шини за рахунок адресації. Оптичний датчик пилу PMS5003 під'єднано до інтерфейсу UART, яким він передає кадри даних із готовими значеннями концентрації. Аналогові датчики газів, кислотності та вологості ґрунту під'єднано до каналів вбудованого АЦП мікроконтролера.

Підсистема живлення вузла побудована за схемою з акумулятором та сонячною панеллю. Контролер заряду забезпечує підзарядження акумулятора від сонячної панелі та захист від надмірного розрядження. Стабілізатор напруги формує стабілізовану напругу живлення мікроконтролера та датчиків. Така організація живлення забезпечує автономність вузла та можливість його розгортання у місцях без доступу до електромережі, що є важливим для моніторингу віддалених природних територій.

Для забезпечення коректної роботи аналогових датчиків передбачено стабілізацію опорної напруги АЦП, оскільки коливання напруги живлення безпосередньо впливають на результат вимірювання. Цифрові датчики менш

чутливі до таких коливань завдяки внутрішньому формуванню результату у цифровій формі. Зазначені особливості враховано під час проектування схеми вузла.

2.2.5 Калібрування та корекція показань датчиків

Недорогі датчики, що застосовуються у вузлах, характеризуються індивідуальним розкидом характеристик та залежністю показань від умов навколишнього середовища. Для отримання достовірних значень необхідне калібрування датчиків, що полягає у визначенні залежності між показаннями датчика та дійсним значенням вимірюваної величини. Калібрувальна характеристика для більшості датчиків апроксимується лінійною або поліноміальною залежністю, коефіцієнти якої визначаються за результатами вимірювань за відомих значень величини.

Для напівпровідникового газового датчика калібрування ускладнюється залежністю опору чутливого шару від температури та вологості. Тому застосовується багатовимірна корекція, що враховує не лише сирий сигнал датчика, а й поточні значення температури та вологості, виміряні датчиком BME280. Фрагмент функції корекції показань наведено у лістингу 2.3.

Лістинг 2.3 – Корекція показань газового датчика

```
float correctGas(float raw, float temp, float hum) {
    // нормування за температурою та вологістю
    float kt = 1.0 + ALPHA_T * (temp - T_REF);
    float kh = 1.0 + ALPHA_H * (hum - H_REF);
    float corrected = raw / (kt * kh);
    // перетворення у концентрацію (ppm)
    return A_COEF * corrected + B_COEF;
}
```

Калібрування датчиків доцільно повторювати періодично, оскільки з часом характеристики чутливих елементів змінюються внаслідок старіння та забруднення. Для мережі вузлів ефективним підходом є взаємне калібрування за даними еталонного приладу або стаціонарного поста, поблизу якого тимчасово розміщується вузол. Такий підхід дає змогу підтримувати достовірність даних

мережі недорогих датчиків на прийнятному рівні протягом тривалого часу експлуатації.

2.3 Проектування бази даних

База даних є центральним сховищем інформації підсистеми, від структури якого залежать продуктивність та зручність роботи з даними. Проектування бази даних виконано у три етапи: інфологічне проектування, на якому визначено сутності предметної області та зв'язки між ними; даталогічне проектування, на якому інфологічну модель перетворено на структуру реляційних таблиць; та фізичне проектування, на якому визначено типи даних, індекси та обмеження цілісності.

Для накопичення даних застосовано реляційну СКБД. Реляційна модель забезпечує цілісність даних, підтримку транзакцій та зручне виконання аналітичних запитів засобами мови SQL. Інфологічну модель предметної області подано у вигляді сутностей: вузли спостереження, датчики, виміри, користувачі та попередження.

Вибір реляційної моделі замість нереляційних (документоорієнтованих, часових рядів) обґрунтовано структурованим характером даних моніторингу та потребою у складних аналітичних запитах з об'єднанням даних різних сутностей. Хоча для зберігання часових рядів вимірювань існують спеціалізовані бази даних, реляційна СКБД забезпечує універсальність, широку підтримку та достатню продуктивність для обсягів даних, характерних для локального моніторингу. За потреби обслуговування дуже великих обсягів даних реляційну СКБД може бути доповнено спеціалізованим сховищем часових рядів без зміни загальної архітектури підсистеми.

Логічну структуру бази даних із зазначенням таблиць, їхніх атрибутів та зв'язків наведено на рисунку 2.2.

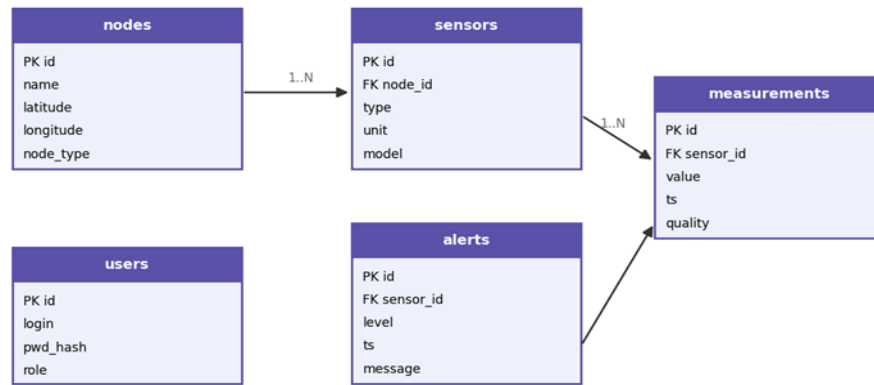


Рисунок 2.2 – Структура бази даних підсистеми

Інфологічна модель предметної області будується на основі аналізу об'єктів, дані про які необхідно зберігати, та зв'язків між ними. Основними сутностями є вузол спостереження, датчик, вимірювання, користувач та попередження. Вузол спостереження характеризується унікальним ідентифікатором, назвою, географічними координатами та типом. Датчик належить конкретному вузлу та характеризується типом вимірюваної величини, одиницею вимірювання та моделлю. Вимірювання є фактом реєстрації значення показника певним датчиком у визначений момент часу.

Зв'язки між сутностями відображають логіку предметної області. Вузол та датчик пов'язані відношенням «один до багатьох»: один вузол може містити декілька датчиків, тоді як кожен датчик належить лише одному вузлу. Аналогічно, датчик та вимірювання пов'язані відношенням «один до багатьох»: кожен датчик формує множину вимірювань, а кожне вимірювання належить одному датчику. Попередження пов'язане з датчиком, на якому зафіксовано перевищення. Така структура зв'язків забезпечує однозначну інтерпретацію даних та можливість виконання різноманітних аналітичних запитів, зокрема вибірки всіх вимірювань певного вузла за заданий період.

Таблиця `nodes` зберігає опис вузлів спостереження з географічними координатами. Таблиця `sensors` пов'язана з вузлами відношенням «один до багатьох» і описує встановлені на вузлі датчики. Центральною є таблиця

measurements, що накопичує виміряні значення з прив'язкою до датчика та мітки часу. Таблиця alerts реєструє факти перевищення граничних значень, а таблиця users зберігає облікові записи користувачів із розмежуванням ролей доступу. Опис основних полів центральної таблиці наведено у таблиці 2.3.

Таблиця 2.3 – Структура таблиці measurements

Поле	Тип	Призначення
id	BIGSERIAL	Первинний ключ
sensor_id	INTEGER	Зовнішній ключ на sensors
value	REAL	Виміряне значення
ts	TIMESTAMP	Час вимірювання
quality	SMALLINT	Ознака достовірності

Під час проектування бази даних застосовано нормалізацію до третьої нормальної форми, що усуває надмірність даних та аномалії оновлення. Відомості про вузли та датчики винесено в окремі таблиці, тому за зміни характеристик вузла не потрібно оновлювати численні записи вимірювань. Для пришвидшення вибірки даних за часовими інтервалами на полі мітки часу таблиці measurements створено індекс, що суттєво підвищує швидкодію аналітичних запитів на великих обсягах даних.

Розглянемо застосування нормальних форм докладніше. Перша нормальна форма вимагає атомарності значень атрибутів – кожне поле таблиці містить лише одне неподільне значення, що виконується для всіх таблиць проектованої бази даних. Друга нормальна форма вимагає повної функційної залежності неключових атрибутів від первинного ключа, що забезпечується винесенням характеристик вузлів та датчиків в окремі таблиці. Третя нормальна форма усуває транзитивні залежності між неключовими атрибутами, чого досягнуто за рахунок відокремлення довідкової інформації від даних вимірювань.

Дотримання нормальних форм забезпечує цілісність та несуперечливість даних, проте у деяких випадках може знижувати продуктивність аналітичних запитів через потребу у з'єднанні таблиць. Для часто виконуваних запитів,

зокрема отримання останніх значень показників за вузлами, передбачено можливість застосування матеріалізованих подань або кешування результатів на рівні сервера, що поєднує переваги нормалізованої структури з високою швидкодією подання даних користувачеві.

Окрім структури таблиць, під час фізичного проектування бази даних визначено типи даних полів з урахуванням характеру зберезуваних значень та вимог до точності й обсягу. Для ідентифікаторів застосовано цілочисельні типи з автоінкрементом, для вимірних значень – тип з рухомою комою, для міток часу – спеціалізований тип, що зберігає дату й час з потрібною точністю. Правильний вибір типів даних впливає як на обсяг бази даних, так і на швидкість операцій, тому йому приділено належну увагу. Для забезпечення цілісності визначено обмеження на обов’язковість заповнення ключових полів та допустимі значення окремих атрибутів [17].

Оскільки таблиця вимірювань є найбільш динамічною та швидко зростає у розмірі, передбачено можливість її секціонування за періодами часу, що спрощує обслуговування та архівування історичних даних. Реляційна модель забезпечує цілісність даних за рахунок обмежень зовнішніх ключів: запис вимірювання не може посылатися на неіснуючий датчик, а датчик – на неіснуючий вузол.

Таблиця попереджень alerts відіграє важливу роль у реалізації функції оповіщення про перевищення граничних значень. Кожен запис цієї таблиці містить посилання на датчик, рівень критичності, мітку часу та текстовий опис події. Накопичення історії попереджень дає змогу аналізувати частоту та характер перевищень, виявляти проблемні ділянки території та оцінювати ефективність вжитих заходів. Таблиця користувачів users забезпечує розмежування доступу: поле ролі визначає набір дозволених операцій, що реалізує модель керування доступом на основі ролей.

2.4 Розробка програмного забезпечення

Програмне забезпечення підсистеми поділяється на дві основні частини: вбудоване програмне забезпечення вузлів збору даних та серверне програмне забезпечення разом із веб-додатком. Обидві частини розроблено з урахуванням обраної архітектури та протоколів взаємодії. Нижче розглянуто кожну з них окремо.

2.4.1 Програмне забезпечення вузла збору даних

Програмне забезпечення вузла реалізовано мовою C++ у середовищі Arduino для ESP32. Алгоритм роботи вузла побудовано як циклічний процес із переходом у режим зниженого енергоспоживання між циклами вимірювання. Схему алгоритму функціонування вузла наведено на рисунку 2.3.

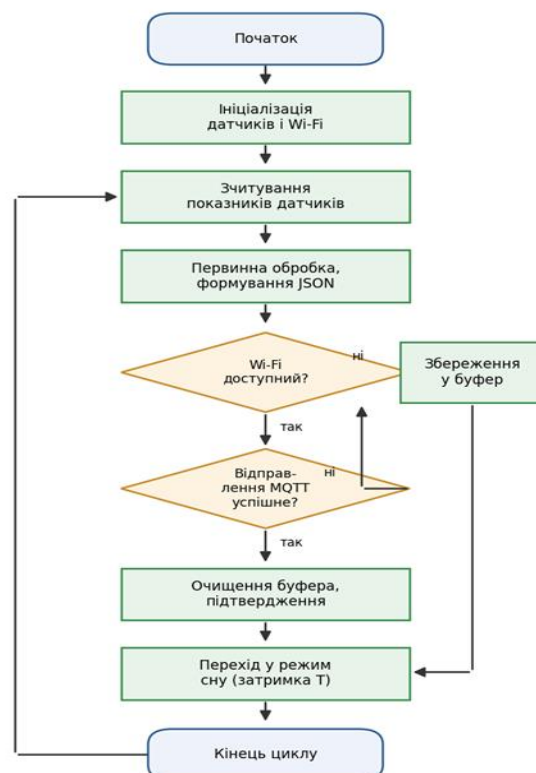


Рисунок 2.3 – Схема алгоритму роботи вузла

Вибір мови C++ та середовища Arduino зумовлений кількома чинниками. Мова C++ забезпечує ефективне використання обмежених ресурсів мікроконтролера та прямий доступ до апаратних засобів, що є важливим для роботи з датчиками й периферією. Середовище Arduino надає велику кількість готових бібліотек для роботи з типовими датчиками, протоколами зв'язку та форматом JSON, що суттєво скорочує час розроблення. Для роботи з датчиками застосовано спеціалізовані бібліотеки, для формування повідомлень JSON – бібліотеку серіалізації, а для взаємодії з брокером – бібліотеку клієнта MQTT [20].

Після ввімкнення вузол ініціалізує датчики та встановлює з'єднання з мережею Wi-Fi. Далі циклічно зчитуються показники датчиків, виконується їх первинне оброблення та формується JSON-повідомлення. За наявності зв'язку повідомлення публікується у MQTT-брокер; у разі невдачі або відсутності мережі дані зберігаються у локальному буфері та надсилаються за першої нагоди. Між циклами вузол переходить у режим сну на заданий інтервал. Фрагмент функції формування та публікації повідомлення наведено у лістингу 2.1.

Лістинг 2.1 – Формування та публікація повідомлення

```
void publishMeasurement() {
    StaticJsonDocument<256> doc;
    doc["node"] = NODE_ID;
    doc["pm25"] = pms.getPM25();
    doc["pm10"] = pms.getPM10();
    doc["co2"] = mq135.readCO2();
    doc["temp"] = bme.readTemperature();
    doc["hum"] = bme.readHumidity();
    doc["soil"] = readSoilMoisture();
    doc["ts"] = getEpochTime();

    char buf[256];
    size_t n = serializeJson(doc, buf);
    if (!mqtt.publish(TOPIC, buf, n)) {
        buffer.store(buf, n); // зберегти у буфер
    }
}
```

Первинне оброблення сигналів датчиків на вузлі включає кілька операцій. Для аналогових датчиків виконується усереднення кількох послідовних відліків АЦП з метою зменшення впливу шумів. Сирі значення АЦП перетворюються у фізичні величини за калібрувальними характеристиками датчиків. Для напівпровідникового газового датчика додатково застосовується корекція показань залежно від температури та вологості, вимірюаних датчиком VME280. Такий підхід підвищує достовірність даних ще на етапі їх формування, до передавання на сервер.

Усереднення відліків АЦП є простим та ефективним методом цифрової фільтрації, що зменшує вплив випадкових завад. Кількість усереднюваних відліків обирається як компроміс між якістю фільтрації та часом вимірювання: збільшення кількості відліків покращує згладжування, але подовжує активну фазу та збільшує енергоспоживання. Для розроблюваного вузла обрано усереднення кількох десятків відліків, що забезпечує достатнє згладжування за прийняттого часу вимірювання. За потреби можуть застосовуватися й складніші методи фільтрації, зокрема медіанна фільтрація для усунення поодиноких викидів.

Перехід вузла у режим глибокого сну між циклами вимірювання є ключовим для забезпечення енергоефективності. У цьому режимі більшість підсистем мікроконтролера вимикається, а споживання струму зменшується на кілька порядків. Пробудження відбувається за сигналом вбудованого таймера через заданий інтервал, після чого цикл вимірювання повторюється. Тривалість інтервалу сну є налаштованим параметром, що дає змогу обирати компроміс між частотою оновлення даних та тривалістю автономної роботи.

Окремої уваги потребує організація надійного встановлення з'єднання з мережею Wi-Fi після пробудження вузла. Оскільки після виходу з режиму глибокого сну стан мережевого з'єднання втрачається, вузол має повторно під'єднатися до точки доступу та брокера. Для підвищення надійності реалізовано механізм повторних спроб з'єднання з обмеженою кількістю повторів та зростаючою затримкою між ними. Якщо встановити з'єднання не вдається протягом визначеного часу, вузол зберігає виміряні дані у буфері та переходить у

режим сну, відкладаючи передавання до наступного циклу. Це запобігає надмірному споживанню енергії на безуспішні спроби з'єднання за тривалої відсутності мережі.

Алгоритм роботи вузла також передбачає оброблення позаштатних ситуацій, зокрема відмови окремих датчиків. Якщо під час зчитування датчика виявлено помилку або отримано фізично неможливе значення, відповідний показник позначається як недостовірний, проте це не перериває роботу вузла – решта показників вимірюється та передається у звичайному режимі. Такий підхід підвищує відмовостійкість вузла та забезпечує неперервність моніторингу навіть за часткової несправності обладнання.

Структуру вбудованого програмного забезпечення вузла організовано за модульним принципом. Виокремлено модуль роботи з датчиками, що інкапсулює деталі взаємодії з кожним типом датчика та надає уніфікований інтерфейс зчитування показань; модуль зв'язку, що відповідає за встановлення з'єднання та передавання даних; модуль буферизації, що керує тимчасовим зберіганням даних; та головний модуль, що координує роботу решти модулів відповідно до загального алгоритму. Такий поділ спрощує розроблення, тестування та подальше розширення програмного забезпечення, зокрема додавання нових типів датчиків.

Конфігураційні параметри вузла, такі як ідентифікатор вузла, параметри підключення до мережі, адреса брокера та інтервал опитування, винесено в окремий конфігураційний блок, що дає змогу налаштовувати вузол без зміни основного коду. Це спрощує розгортання багатьох однотипних вузлів, оскільки для кожного вузла достатньо змінити лише конфігураційні параметри. Зазначені рішення відповідають сучасним практикам розроблення вбудованого програмного забезпечення для систем інтернету речей.

2.4.2 Серверне програмне забезпечення та веб-додаток

Серверну частину реалізовано як веб-додаток. Сервер підписується на повідомлення MQTT-брокера, виконує їх розбір, перевіряє на перевищення граничних значень та зберігає у базі даних. Для взаємодії з веб-клієнтом сервер

надає REST API, що повертає дані у форматі JSON. Фрагмент обробника MQTT-повідомлень наведено у лістингу 2.2.

Серверне програмне забезпечення реалізовано мовою Python, що забезпечує високу швидкість розроблення, наявність розвинених бібліотек для роботи з протоколом MQTT, базами даних та побудови веб-інтерфейсів, а також зручність супроводження коду. Як система керування базами даних обрано реляційну СКБД з підтримкою мови SQL та механізмів забезпечення цілісності даних. MQTT-брокер реалізовано на основі відкритого програмного забезпечення, що забезпечує надійне функціонування та широкі можливості налаштування. Усі застосовані компоненти є відкритими та безкоштовними, що відповідає вимозі економічної доступності підсистеми.

Лістинг 2.2 – Обробник вхідного повідомлення на сервері

```
def on_message(client, userdata, msg):
    data = json.loads(msg.payload)
    sensor_ids = resolve_sensors(data["node"])
    for key, sid in sensor_ids.items():
        value = data.get(key)
        if value is None:
            continue
        save_measurement(sid, value, data["ts"])
        if exceeds_limit(key, value):
            create_alert(sid, value)
```

Веб-додаток реалізовано як односторінковий застосунок, що звертається до REST API сервера та відображає дані у вигляді карток поточних значень, графіків динаміки показників і карти розташування вузлів. Загальний вигляд інтерфейсу панелі спостереження наведено на рисунку 2.4.

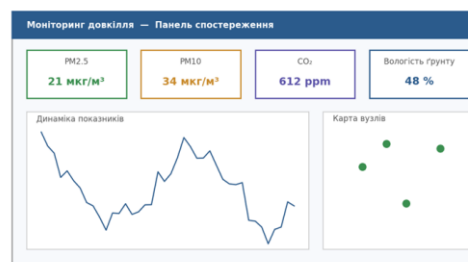


Рисунок 2.4 – Вигляд інтерфейсу панелі спостереження

Програмний інтерфейс сервера побудовано за принципами REST: кожен ресурс (вузли, датчики, виміри, попередження) доступний за власною адресою, а операції над ним виконуються стандартними методами протоколу HTTP. Основні кінцеві точки API наведено у таблиці 2.4. Доступ до даних, що змінюють стан системи, захищено механізмом автентифікації, а передавання здійснюється захищеним каналом.

Таблиця 2.4 – Основні кінцеві точки REST API

Метод і шлях	Призначення
GET /api/nodes	Перелік вузлів спостереження
GET /api/sensors	Перелік датчиків вузла
GET /api/measurements	Виміри за період і датчиком
GET /api/alerts	Перелік активних попереджень
POST /api/auth	Автентифікація користувача

Серверний застосунок складається з кількох логічних модулів: модуля приймання даних, що підписується на повідомлення брокера; модуля доступу до бази даних, що інкапсулює операції збереження та вибірки; модуля бізнес-логіки, що реалізує перевірку граничних значень та формування попереджень; та модуля програмного інтерфейсу, що обслуговує запити веб-клієнта. Такий поділ на модулі спрощує супроводження та розвиток застосунку, оскільки зміни в одному модулі мінімально впливають на інші.

Модуль приймання даних працює асинхронно: він постійно очікує на нові повідомлення від брокера та обробляє їх у міру надходження. Для кожного отриманого повідомлення виконується послідовність дій – розбір формату JSON, зіставлення показників з датчиками, перевірка достовірності, збереження у базі даних та, за потреби, формування попередження. Завдяки асинхронній обробці сервер здатний приймати повідомлення від багатьох вузлів практично одночасно, не блокуючи оброблення на час виконання операцій з базою даних.

Перевірка достовірності даних на сервері є важливим етапом оброблення, що доповнює первинну перевірку на вузлі. Сервер контролює належність

отриманих значень до фізично можливих діапазонів для кожного типу показника та виявляє різкі відхилення від попередніх значень, що можуть свідчити про несправність датчика. Значення, що не пройшли перевірку, позначаються відповідною ознакою достовірності, але зберігаються у базі даних для подальшого аналізу можливих причин аномалій. Такий підхід забезпечує повноту даних та водночас дає змогу виключати недостовірні значення з аналітичного оброблення.

Формування попереджень виконується модулем бізнес-логіки шляхом порівняння вимірних значень із граничними. У разі перевищення формується запис у таблиці попереджень із зазначенням датчика, рівня критичності та часу події. Для запобігання надмірній кількості однотипних попереджень за тривалого перевищення передбачено логіку, що формує нове попередження лише за переходу показника через граничне значення, а не за кожного вимірювання. Це підвищує інформативність системи попереджень та зменшує навантаження на користувача.

Модуль доступу до бази даних реалізує шаблон проектування, що відокремлює бізнес-логіку від деталей роботи з конкретною системою керування базами даних. Це забезпечує можливість заміни системи керування базами даних без зміни інших модулів та спрощує тестування застосунку. Модуль бізнес-логіки містить правила перевірки граничних значень, які зберігаються у конфігурації та можуть бути змінені без перекомпіляції програми, що забезпечує гнучкість налаштування підсистеми під конкретні нормативні вимоги.

Модуль програмного інтерфейсу обслуговує запити веб-клієнта, виконуючи вибірку даних з бази даних відповідно до параметрів запиту та повертаючи результат у форматі JSON. Перед обробленням запиту перевіряються права доступу користувача, що забезпечує захист даних та операцій. Для запитів історичних даних передбачено можливість фільтрації за вузлом, датчиком та періодом часу, а також агрегування даних за заданими інтервалами, що зменшує обсяг переданих даних та пришвидшує побудову графіків на боці клієнта. Така організація програмного інтерфейсу забезпечує ефективну взаємодію клієнта із сервером за різних сценаріїв використання [18].

2.4.3 Структура повідомлень та взаємодія компонентів

Обмін даними між компонентами підсистеми побудовано на основі формату JSON. Повідомлення, що формується вузлом, містить ідентифікатор вузла, набір пар «показник – значення» для кожного з вимірних параметрів та мітку часу вимірювання. Така структура є самоописовою та легко розширюваною: додавання нового показника не потребує зміни формату повідомлення загалом, що спрощує розвиток підсистеми.

Вузли публікують повідомлення у визначені теми MQTT-брокера, ієрархічна структура яких відображає належність даних до конкретного вузла та типу показника. Сервер, підписавшись на відповідні теми, отримує повідомлення, виконує їх розбір, зіставляє показники з відповідними датчиками у базі даних та зберігає вимірні значення. Веб-клієнт, своєю чергою, отримує дані не безпосередньо від брокера, а через REST API сервера, що забезпечує єдину точку доступу до даних та можливість застосування контролю доступу.

Така організація взаємодії забезпечує чіткий поділ відповідальності між компонентами: вузли відповідають за вимірювання та публікацію, брокер – за маршрутизацію повідомлень, сервер – за оброблення та зберігання, веб-клієнт – за подання. Кожен компонент може бути розроблений, протестований та замінений незалежно від інших за умови дотримання узгоджених форматів обміну, що відповідає сучасним принципам побудови розподілених інформаційних систем.

Послідовність обміну даними між компонентами підсистеми від моменту вимірювання до надання даних користувачеві наведено на рисунку 2.6. Діаграма відображає основні повідомлення, якими обмінюються компоненти, та порядок їх передавання.

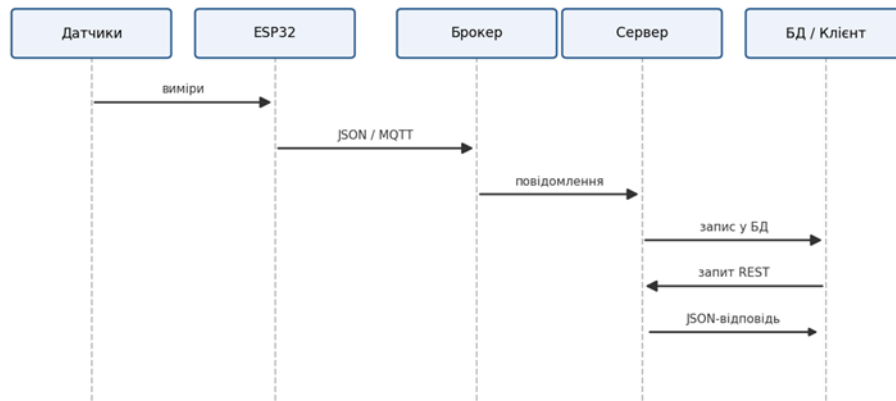


Рисунок 2.6 – Діаграма взаємодії компонентів підсистеми

Як видно з діаграми, дані послідовно проходять через усі рівні підсистеми. Датчики передають виміряні значення мікроконтролеру, який формує повідомлення у форматі JSON та публікує його у брокер за протоколом MQTT. Брокер передає повідомлення серверу, що зберігає дані у базі даних. Веб-клієнт за запитом через REST API отримує дані від сервера у форматі JSON для відображення користувачеві. Така послідовність забезпечує впорядковане проходження даних та чітке розмежування функцій компонентів на кожному етапі.

2.4.4 Програмне забезпечення веб-клієнта

Веб-клієнт реалізовано як односторінковий застосунок, що виконується у браузері користувача та взаємодіє із сервером через REST API. Такий підхід забезпечує доступність підсистеми з будь-якого пристрою, оснащеного браузером, без потреби у встановленні спеціалізованого програмного забезпечення. Інтерфейс побудовано з урахуванням принципів зручності використання та наочності подання інформації.

Головна сторінка веб-клієнта містить панель спостереження, на якій відображаються картки поточних значень основних показників, графіки динаміки показників за обраний період та карта розташування вузлів. Картки поточних значень візуально вирізняють показники, що перевищують граничні значення, кольоровим індикатором, що привертає увагу користувача до проблемних

ділянок. Графіки динаміки будуються на основі історичних даних, отриманих від сервера, та дають змогу простежити зміну показників у часі.

Інтерфейс веб-клієнта побудовано за принципом адаптивності, що забезпечує коректне відображення на пристроях з різним розміром екрана – від настільних комп'ютерів до мобільних пристроїв. Це дає змогу користувачам отримувати доступ до даних моніторингу у будь-якому місці за наявності з'єднання з мережею. Навігація між поточними даними, історичними графіками та картою організована інтуїтивно зрозумілим чином, що не потребує спеціальної підготовки користувача. Для зручності аналізу передбачено можливість вибору періоду відображення історичних даних та конкретних показників, що цікавлять користувача.

Оновлення даних на панелі спостереження виконується періодично шляхом повторних запитів до сервера, що забезпечує відображення актуального стану довкілля у режимі, близькому до реального часу. Для зменшення навантаження на сервер та мережу інтервал оновлення обрано співмірним з інтервалом надходження нових вимірювань. Карта розташування вузлів дає змогу оцінити просторовий розподіл показників та швидко визначити вузли, на яких зафіксовано перевищення.

Фрагмент коду веб-клієнта, що виконує запит поточних даних до сервера та оновлює інтерфейс, наведено у лістингу 2.4.

Лістинг 2.4 – Запит даних веб-клієнтом

```
async function refreshData() {
  try {
    const resp = await fetch("/api/measurements/latest");
    const data = await resp.json();
    data.forEach(m => updateCard(m.sensor, m.value));
    renderChart(data);
  } catch (e) {
    showError("Не вдалося отримати дані");
  }
}
setInterval(refreshData, REFRESH_INTERVAL);
```

2.5 Забезпечення захисту інформації

Захист інформації у підсистемі моніторингу реалізовано на кількох рівнях. На рівні передавання даних від веб-клієнта до сервера застосовано протокол захищеного передавання гіпертексту HTTPS, що використовує криптографічний протокол TLS для шифрування трафіку та запобігання його перехопленню й модифікації. Це унеможливорює несанкціонований доступ до даних під час їх передавання відкритими мережами.

Протокол TLS забезпечує конфіденційність даних шляхом їх шифрування, цілісність – за допомогою кодів автентифікації повідомлень, та автентифікацію сервера – на основі цифрових сертифікатів. Завдяки цьому веб-клієнт може пересвідчитися, що взаємодіє саме із сервером підсистеми, а не зі зловмисником, що підмінив його, а передані дані не можуть бути прочитані чи змінені третьою стороною. Застосування захищеного протоколу є обов’язковим за передавання даних відкритими мережами та особливо важливим у разі публічного доступу до підсистеми.

Доступ користувачів до підсистеми регламентується механізмом автентифікації та розмежування прав. У базі даних зберігаються облікові записи користувачів із зазначенням ролі: спостерігач, який має право лише переглядати дані, та адміністратор, якому доступні також операції налаштування вузлів та керування користувачами. Паролі зберігаються не у відкритому вигляді, а у формі криптографічних хешів, що захищає їх навіть у разі несанкціонованого доступу до бази даних.

Застосування криптографічного хешування паролів передбачає перетворення пароля у незворотну послідовність фіксованої довжини за допомогою стійкої хеш-функції з додаванням випадкової солі для кожного запису. Під час автентифікації пароль, уведений користувачем, хешується та порівнюється зі збереженим значенням, тому сам пароль ніколи не зберігається у відкритому вигляді та не може бути відновлений зі збереженого хешу. Додавання

солі унеможлиблює застосування заздалегідь обчислених таблиць для відновлення паролів, що суттєво підвищує захищеність облікових даних користувачів.

На рівні обміну повідомленнями між вузлами та брокером передбачено можливість автентифікації вузлів за обліковими даними та обмеження доступу до тем брокера. Це запобігає публікації недостовірних даних сторонніми пристроями. Сукупність зазначених заходів забезпечує конфіденційність, цілісність та доступність даних підсистеми відповідно до вимог технічного завдання.

Окрему увагу приділено захисту від несанкціонованого втручання у роботу підсистеми. Оскільки вузли можуть бути розміщені у загальнодоступних місцях, передбачено заходи з обмеження фізичного доступу до них та контролю цілісності переданих даних. Застосування захищеного каналу зв'язку та автентифікації вузлів унеможлиблює підміну даних зловмисником, а перевірка достовірності на сервері дає змогу виявити аномальні значення, що можуть свідчити про несправність або втручання.

Резервне копіювання бази даних забезпечує збереження накопичених даних у разі апаратних збоїв сервера. Періодичність резервного копіювання обирається залежно від цінності даних та допустимого обсягу їх втрати. Сукупність організаційних та технічних заходів захисту забезпечує надійне функціонування підсистеми та збереження зібраної інформації.

2.6 Тестування програмного забезпечення

Розроблене програмне забезпечення підлягало поетапному тестуванню. На рівні вузла перевірялася коректність зчитування показань датчиків шляхом порівняння з контрольними вимірюваннями, а також правильність формування

JSON-повідомлень. Окремо тестувалася робота механізму буферизації за штучного відключення мережі.

На рівні сервера виконувалося модульне тестування функцій розбору повідомлень, збереження вимірювань та перевірки граничних значень. Для цього сформовано набір тестових повідомлень, що містили як коректні дані, так і граничні й помилкові значення, та перевірено відповідність реакції сервера очікуваним результатам. Перелік тестових сценаріїв наведено у таблиці 2.5.

Таблиця 2.5 – Сценарії тестування програмного забезпечення

Сценарій	Очікуваний результат
Коректне повідомлення	Збереження у БД
Перевищення ГДК	Збереження + попередження
Неповне повідомлення	Пропуск відсутніх полів
Помилковий формат JSON	Відхилення повідомлення
Дублікат за міткою часу	Ігнорування дублікату

На рівні веб-додатку перевірялася коректність відображення даних, оновлення показників у реальному часі та робота з різними веб-браузерами. За результатами тестування виявлені недоліки було усунуто, а підсистему визнано працездатною та готовою до експериментального дослідження.

Окрім функціонального тестування, проведено перевірку поведінки підсистеми за граничних та позаштатних умов. Зокрема, перевірено реакцію сервера на надходження повідомлень з некоректними або відсутніми полями, на спроби несанкціонованого доступу до програмного інтерфейсу, а також на одночасне надходження великої кількості повідомлень. Результати показали, що сервер коректно відхиляє некоректні повідомлення, обмежує доступ до захищених ресурсів та зберігає працездатність за підвищеного навантаження, що підтверджує надійність реалізованого програмного забезпечення.

Тестування механізму буферизації вузла полягало у відключенні мережі під час роботи та подальшому її відновленні з контролем повноти переданих даних. У всіх проведених випробуваннях дані, накопичені у буфері під час відсутності

зв'язку, було повністю передано після його відновлення без втрат та порушення хронології, що підтверджує коректність реалізації цього критично важливого механізму.

Результати проведеного тестування узагальнено та використано для усунення виявлених недоліків програмного забезпечення. Зокрема, було уточнено оброблення граничних значень показників, удосконалено логіку повторного з'єднання з мережею та оптимізовано формування запитів до бази даних. Після внесення відповідних змін тестування повторювалося до досягнення стабільної та коректної роботи всіх компонентів підсистеми. Такий ітеративний підхід до тестування та вдосконалення забезпечив високу якість кінцевого програмного забезпечення та його готовність до експериментального дослідження у складі підсистеми.

2.7 Особливості розгортання та експлуатації підсистеми

Розгортання підсистеми передбачає кілька етапів: установлення серверного програмного забезпечення та бази даних, налаштування MQTT-брокера, конфігурування й розміщення вузлів збору даних та налаштування веб-клієнта. Сервер, брокер та база даних розгортаються на обладнанні організації або на виділеному сервері, що забезпечує контроль над даними та незалежність від зовнішніх сервісів. Вузли конфігуруються шляхом задання їх ідентифікаторів, параметрів підключення до мережі Wi-Fi та адреси брокера, після чого розміщуються у точках спостереження.

Вибір місць розміщення вузлів суттєво впливає на репрезентативність даних. Вузли контролю якості повітря доцільно розташовувати на висоті дихання людини, уникаючи безпосередньої близькості до локальних джерел, що можуть спотворити загальну картину. Вузли контролю стану ґрунту та води розміщуються безпосередньо у досліджуваному середовищі. Під час розміщення

враховуються також умови електроживлення, доступність мережі Wi-Fi та захищеність вузла від несприятливих погодних впливів і несанкціонованого доступу.

Типову топологію розгортання підсистеми на території спостереження наведено на рисунку 2.5. Вузли збору даних розподілено по території з урахуванням розташування джерел забруднення та зон, що потребують контролю, а їхні дані через шлюз Wi-Fi надходять до сервера з базою даних і брокером, звідки стають доступними користувачам.

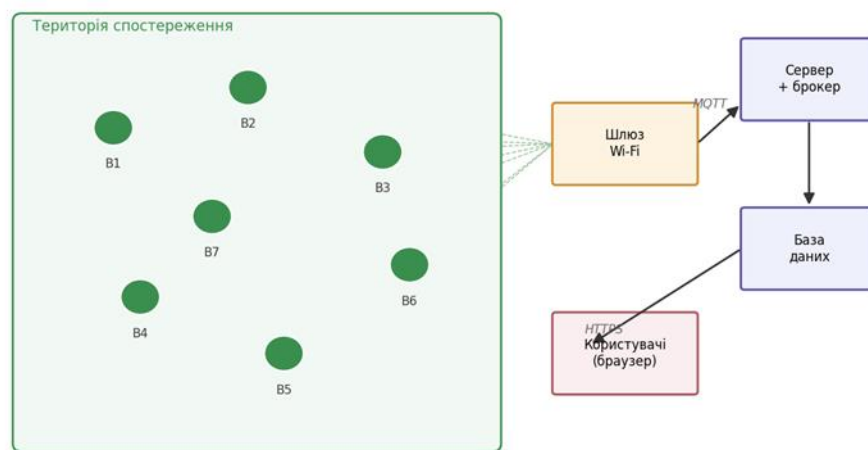


Рисунок 2.5 – Типова топологія розгортання підсистеми

Кількість та розташування вузлів визначаються індивідуально для кожного об'єкта на основі попереднього обстеження території, аналізу розташування потенційних джерел забруднення та цілей моніторингу. Щільність розміщення вузлів може бути нерівномірною: у зонах підвищеного ризику чи поблизу важливих об'єктів вузли розташовуються щільніше, тоді як на фонових ділянках достатньо рідшого розміщення. Така гнучкість розгортання є важливою перевагою підсистеми, що дає змогу оптимально розподілити обмежені ресурси та забезпечити необхідну детальність контролю саме там, де це найбільш потрібно.

Експлуатація підсистеми передбачає періодичне технічне обслуговування, що включає перевірку працездатності вузлів, контроль рівня заряду акумуляторів, очищення датчиків від забруднень та періодичне калібрування. Завдяки автоматичному передаванню даних та централізованому контролю стану вузлів обсяг обслуговування є незначним, а більшість операцій може виконуватися дистанційно через серверний інтерфейс. Це забезпечує низькі експлуатаційні витрати та простоту супроводження підсистеми.

Для контролю стану вузлів доцільно передбачити передавання разом із даними вимірювань службової інформації – рівня заряду акумулятора, рівня сигналу Wi-Fi та ознаки справності датчиків. Це дає змогу адміністратору дистанційно відстежувати стан кожного вузла та своєчасно реагувати на несправності чи розрядження акумулятора, не очікуючи повної втрати зв'язку з вузлом. Відображення стану вузлів у веб-інтерфейсі спрощує обслуговування мережі, особливо за значної кількості територіально рознесених вузлів, та підвищує загальну надійність функціонування підсистеми.

2.8 Узагальнена характеристика розробленої підсистеми

За результатами проектування сформовано завершену апаратно-програмну реалізацію підсистеми. Апаратну частину становлять вузли збору даних на основі мікроконтролера ESP32 з комплектом датчиків для вимірювання показників якості повітря, води та ґрунту. Програмну частину утворюють вбудоване програмне забезпечення вузлів, серверний застосунок з базою даних та MQTT-брокером, а також веб-додаток для подання даних користувачеві.

Узагальнені технічні характеристики розробленої підсистеми наведено у таблиці 2.6. Вони відображають основні параметри, що визначають функціональні можливості та умови застосування підсистеми.

Таблиця 2.6 – Узагальнені характеристики підсистеми

Характеристика	Значення
Контрольовані середовища	повітря, вода, ґрунт
Тип мікроконтролера вузла	ESP32
Технологія зв'язку	Wi-Fi (IEEE 802.11)
Протокол передавання даних	MQTT, REST
Формат обміну даними	JSON
Тип бази даних	реляційна
Інтерфейс користувача	веб-додаток
Захист передавання	HTTPS / TLS

Наведені характеристики підтверджують відповідність розробленої підсистеми вимогам технічного завдання. Підсистема забезпечує контроль усіх передбачених компонентів довкілля, використовує усталені та сумісні технології, реалізує захищене передавання даних та зручний веб-інтерфейс. Розроблені рішення утворюють основу для виготовлення дослідного зразка та проведення експериментального дослідження, результати якого наведено у наступному розділі.

Висновки до розділу 2. У розділі розроблено структурну й функціональну схеми підсистеми, обґрунтовано вибір елементної бази – мікроконтролера ESP32 та комплекту датчиків, виконано оціночні розрахунки енергоспоживання й трафіку, що підтверджують працездатність обраного варіанту. Спроектовано структуру реляційної бази даних, розроблено алгоритм роботи вузла та програмне забезпечення вузла й сервера, наведено фрагменти програмного коду та вигляд інтерфейсу веб-додатку.

Розроблені рішення утворюють завершену апаратно-програмну основу підсистеми. Обраний мікроконтролер ESP32 у поєднанні з комплектом датчиків забезпечує вимірювання всіх передбачених технічним завданням показників, а реалізований розподіл функцій між рівнями підсистеми відповідає принципам побудови систем інтернету речей. Спроекована база даних у третій нормальній формі забезпечує цілісність та ефективне зберігання даних, а модульна структура серверного програмного забезпечення створює основу для подальшого розвитку

підсистеми. Передбачені заходи захисту інформації та проведене тестування підтверджують готовність підсистеми до експериментального дослідження, результати якого наведено у наступному розділі.

3 МОДЕЛЮВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА

3.1 Формування моделі та умови експерименту

Перевірка працездатності розробленої підсистеми передбачає дослідження її поведінки за різних умов експлуатації. Оскільки розгортання повномасштабної мережі вузлів у межах дипломної роботи є недоцільним, дослідження проведено на макеті, що відтворює всі основні компоненти підсистеми у мінімальній конфігурації, достатній для перевірки коректності їх взаємодії та основних характеристик.

Для перевірки працездатності розробленої підсистеми проведено експериментальне дослідження на макеті, що складався з одного вузла збору даних на базі ESP32, локального MQTT-брокера та сервера з базою даних, розгорнутих на персональному комп'ютері. Вузол було розміщено у приміщенні та на відкритому повітрі для оцінювання поведінки підсистеми за різних умов.

Під моделюванням у цьому контексті розуміється відтворення роботи підсистеми у контрольованих умовах з метою дослідження її характеристик. Застосовано натурне моделювання на фізичному макеті, що дає змогу врахувати реальні характеристики апаратних компонентів та умов передавання даних, на відміну від суто програмного імітаційного моделювання. Натурний макет забезпечує найбільш достовірну оцінку працездатності, оскільки відтворює всі етапи роботи підсистеми – від фізичного вимірювання до відображення даних користувачеві.

Критеріями оцінювання працездатності підсистеми обрано такі показники: повнота та коректність зібраних даних, що характеризує здатність підсистеми

реєструвати та зберігати вимірювання без пропусків та спотворень; надійність передавання, що визначається часткою успішно доставлених повідомлень; затримка доставлення, що характеризує оперативність надання даних; точність вимірювань, що оцінюється порівнянням з контрольними значеннями; та коректність формування попереджень. Обрані критерії всебічно характеризують роботу підсистеми та відповідають вимогам технічного завдання.

Метою експерименту була перевірка таких аспектів роботи підсистеми: коректність вимірювання та передавання показників; надійність доставлення повідомлень за умов нестабільного зв'язку; правильність роботи механізму буферизації; коректність формування попереджень у разі перевищення граничних значень. Параметри проведення експерименту наведено у таблиці 3.1.

Інтервал опитування для експерименту обрано меншим за робочий (30 секунд замість 5 хвилин) з метою накопичення достатнього обсягу даних за обмежений час спостереження та більш детального відстеження динаміки показників. Тривалість спостереження у 24 години обрано для охоплення повного добового циклу, протягом якого виявляються характерні коливання показників. Граничне значення концентрації PM_{2.5} встановлено відповідно до рекомендованих нормативів для перевірки коректності формування попереджень.

Програмне середовище експерименту складалося з MQTT-брокера, серверного застосунку та бази даних, розгорнутих на персональному комп'ютері, що виконував роль сервера. Вузол збору даних під'єднувався до мережі Wi-Fi та публікував повідомлення у брокер, а сервер приймав їх, зберігав у базі даних та відображав через веб-додаток. Для імітації нестабільного зв'язку періодично відключалася точка доступу Wi-Fi, після чого перевірялася поведінка механізму буферизації.

Апаратна частина макета включала мікроконтролер ESP32 з під'єднаними датчиками якості повітря, температури та вологості. Аналогові датчики під'єднувалися до каналів АЦП, цифровий датчик BME280 – до шини I2C, а оптичний датчик пилу – до інтерфейсу UART. Живлення макета здійснювалося від стабілізованого джерела, що імітувало роботу від акумулятора. Така

конфігурація макета відтворює всі ключові елементи реального вузла та дає змогу дослідити його поведінку у наближених до експлуатаційних умовах.

Достовірність вимірювань оцінювалася шляхом порівняння показань вузла з контрольними значеннями: концентрація діоксиду вуглецю звірялася з показаннями повіреного приладу, а температура й вологість – з показаннями лабораторного термогігрометра. Це дало змогу оцінити не лише функціональну коректність підсистеми, а й орієнтовну точність вимірювань обраними датчиками.

Методика експерименту передбачала кілька серій вимірювань за різних умов. У першій серії вузол працював за стабільного зв'язку для оцінювання базових характеристик. У другій серії штучно вносилися перерви у роботі мережі для перевірки механізму буферизації. У третій серії змінювалися умови навколишнього середовища (концентрація вуглекислого газу, вологість) для оцінювання реакції датчиків. Така організація експерименту забезпечила всебічну перевірку працездатності підсистеми за різних сценаріїв експлуатації.

Під час проведення експерименту дані безперервно фіксувалися у базі даних, що дало змогу після завершення спостереження виконати їх ретроспективний аналіз. Для кожної серії вимірювань реєструвалися як власне значення показників, так і службові параметри – час доставлення повідомлень, факти спрацювання механізму буферизації та формування попереджень. Це забезпечило повноту вихідних даних для подальшого оброблення та дало змогу оцінити роботу підсистеми за всіма обраними критеріями. Накопичені у ході експерименту дані використано для побудови наведених далі графіків та таблиць.

Таблиця 3.1 – Параметри експерименту

Параметр	Значення
Інтервал опитування	30 с
Тривалість спостереження	24 год
Кількість вимірювань	2880
Контрольовані показники	PM2.5, PM10, CO ₂ , Т, вологість
Граничне значення PM2.5	25 мкг/м ³

3.2 Інтерпретація результатів

За результатами добового спостереження отримано часові ряди значень контрольованих показників. На рисунку 3.1 наведено динаміку концентрації твердих частинок PM2.5 та PM10 протягом доби разом із граничним значенням ГДК для PM2.5.

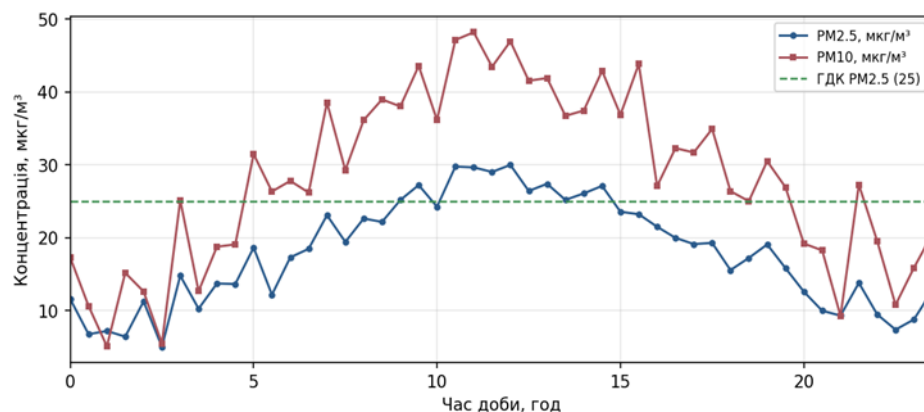


Рисунок 3.1 – Динаміка концентрації твердих частинок за добу

За добу спостереження вузол сформував та передав на сервер близько 2880 повідомлень, кожне з яких містило значення п'яти показників, що загалом склало понад 14 тисяч окремих вимірювань. Усі дані було коректно збережено у базі даних з відповідними мітками часу, що дало змогу побудувати неперервні часові ряди для кожного показника. Перевірка цілісності даних показала відсутність пропусків та порушень хронології, що підтверджує надійність ланцюга збору та збереження даних.

Аналіз отриманих даних показує характерне добове коливання концентрації пилу з мінімумом у нічні години та підвищенням у періоди ранкової й вечірньої активності. У кілька проміжків часу спостерігалось короткочасне перевищення граничного значення PM2.5, на що підсистема коректно відреагувала формуванням відповідних попереджень, збережених у таблиці alerts бази даних.

Виявлений добовий хід концентрації твердих частинок узгоджується з відомими закономірностями динаміки забруднення повітря у населених пунктах. Ранковий та вечірній максимуми пов'язані з підвищенням інтенсивності транспортного руху, тоді як нічний мінімум зумовлений зниженням активності джерел забруднення. Така відповідність отриманих даних очікуваним закономірностям непрямо підтверджує адекватність роботи підсистеми та достовірність вимірювань. Концентрація PM10 у середньому перевищує концентрацію PM2.5, що відповідає фізичній природі цих показників, оскільки фракція PM10 включає й дрібніші частинки фракції PM2.5.

Для оцінювання надійності передавання даних проаналізовано частку успішно доставлених повідомлень за умов штучно внесених перерв у роботі мережі Wi-Fi. Результати наведено у таблиці 3.2.

Таблиця 3.2 – Показники надійності передавання даних

Сценарій	Надіслано	Доставлено	Втрачено, %
Стабільний зв'язок	720	720	0,0
Періодичні перерви	720	718	0,3
З буферизацією	720	720	0,0

Механізм формування попереджень перевірявся шляхом створення умов, за яких контрольований показник перевищував встановлене граничне значення. Під час перевищення концентрації PM2.5 над пороговим значенням сервер коректно реєстрував подію у таблиці попереджень із зазначенням датчика, рівня та часу, а веб-додаток відображав відповідне візуальне сповіщення. Після зниження показника нижче порогового значення нові попередження не формувалися, що свідчить про правильну роботу логіки порівняння з граничними значеннями. Перевірено також відсутність хибних попереджень за нормальних значень показників протягом тривалого спостереження.

Результати підтверджують, що застосування механізму локальної буферизації даних дає змогу повністю уникнути втрат повідомлень навіть за умов нестабільного зв'язку: дані, які не вдалося передати під час перерви, було успішно

надіслано після відновлення з'єднання. За стабільного зв'язку втрат не зафіксовано, а середня затримка доставлення повідомлення від моменту вимірювання до збереження у базі даних не перевищувала 0,8 с.

Додатково проаналізовано поведінку підсистеми за тривалих перерв зв'язку. Установлено, що обсяг локального буфера вузла достатній для зберігання даних за кілька годин роботи за обраного інтервалу опитування. Після відновлення зв'язку накопичені дані передаються пакетно, причому коректність хронології зберігається завдяки індивідуальним міткам часу кожного вимірювання. Це підтверджує придатність підсистеми до експлуатації в умовах нестабільного або періодичного покриття мережею Wi-Fi, характерного для віддалених територій.

Оцінено також час відновлення роботи підсистеми після перезавантаження сервера. Завдяки проміжному зберігання повідомлень брокером та локальній буферизації на вузлах короткочасна недоступність сервера не призводить до втрати даних: після відновлення роботи сервер обробляє накопичені брокером повідомлення та продовжує штатне функціонування. Зазначені властивості свідчать про високу відмовостійкість розробленої підсистеми.

Окремо досліджено вплив температури та вологості на показання напівпровідникового газового датчика. Установлено, що за підвищення відносної вологості показання датчика зростають, що потребує програмної корекції за даними датчика BME280. Після застосування корекції відносна похибка вимірювання концентрації знизилася приблизно вдвічі, що підтверджує важливість сумісного оброблення показань різних датчиків вузла.

Для дослідження впливу умов навколишнього середовища проведено серію вимірювань за різних значень температури та вологості зі сталою концентрацією контрольного газу. Результати показали систематичне відхилення показань датчика залежно від цих параметрів, що узгоджується з відомими характеристиками напівпровідникових газових датчиків. На основі отриманих даних визначено коефіцієнти корекційної функції, що враховує температуру та вологість, після чого корекцію застосовано до показань датчика. Порівняння

скоригованих та нескоригованих значень з контрольними підтвердило ефективність корекції у зменшенні похибки вимірювання.

Перевірено також коректність роботи веб-додатку: поточні значення показників оновлювалися у реальному часі, графіки динаміки відображали накопичені дані, а на карті коректно позначалося розташування вузла. Сформовані попередження про перевищення граничних значень своєчасно з'являлися в інтерфейсі, що свідчить про правильну взаємодію всіх компонентів підсистеми – від вузла до веб-клієнта.

Для кількісного оцінювання отриманих рядів даних обчислено основні статистичні характеристики показників за період спостереження: середнє значення, мінімум, максимум та середньоквадратичне відхилення. Середньодобова концентрація PM_{2.5} становила близько 18 мкг/м³ за максимуму близько 34 мкг/м³, що узгоджується з типовими значеннями для міського середовища. Середньоквадратичне відхилення відображає добову мінливість показника, зумовлену зміною інтенсивності джерел забруднення та метеорологічних умов. Обчислені характеристики є основою для подальшого порівняння стану довкілля у різні періоди та в різних точках території.

Аналіз кореляції між показниками показав очікуваний зв'язок між концентраціями PM_{2.5} та PM₁₀, що пояснюється спільним походженням цих фракцій твердих частинок. Виявлено також зворотну залежність концентрації пилу від швидкості провітрювання приміщення, що підтверджує фізичну обґрунтованість отриманих даних. Зазначені результати свідчать про придатність підсистеми не лише для контролю поточних значень, а й для аналітичного оброблення накопичених даних.

Узагальнюючи результати інтерпретації даних, можна стверджувати, що розроблена підсистема забезпечує отримання достовірної інформації про стан довкілля, придатної як для оперативного контролю, так і для подальшого аналізу. Отримані часові ряди коректно відображають динаміку показників, узгоджуються з відомими закономірностями та фізичними залежностями, а виявлені перевищення граничних значень своєчасно реєструються підсистемою. Це

підтверджує виконання підсистемою основної функції – інформаційного забезпечення контролю стану природного довкілля.

Для оцінювання точності вимірювань виконано порівняння показань вузла з контрольними значеннями. Результати порівняння для основних показників наведено у таблиці 3.3. Відносна похибка визначалася як відношення модуля різниці між показанням вузла та контрольним значенням до контрольного значення.

Таблиця 3.3 – Оцінка точності вимірювань

Показник	Контр. знач.	Показ. вузла	Похибка, %
Температура, °C	22,0	22,4	1,8
Вологість, %	45,0	47,1	4,7
CO ₂ , ppm	600	638	6,3
PM2.5, мкг/м ³	20,0	21,5	7,5

Отримані значення відносної похибки є типовими для недорогих датчиків та прийнятними для завдань локального моніторингу. Найменшу похибку має вимірювання температури, найбільшу – вимірювання концентрації твердих частинок, що пояснюється складністю оптичного методу та залежністю показань від характеристик аерозолі. Застосування калібрування та програмної корекції дає змогу зменшити похибку, проте повне наближення до точності еталонних приладів за використання масових датчиків є недосяжним та не вимагається для поставлених завдань.

3.3 Рекомендації щодо подальшого розвитку

Проведене дослідження підтвердило працездатність підсистеми, проте також виявило напрями її вдосконалення. Аналіз отриманих даних показав, що основним чинником, який обмежує точність вимірювань, є характеристики

недорогих датчиків, зокрема їх залежність від умов навколишнього середовища та дрейф показань з часом. Це зумовлює потребу у періодичному калібруванні та програмній корекції показань.

Іншим напрямом удосконалення є підвищення енергоефективності вузлів, що дасть змогу збільшити інтервал між обслуговуванням та розширити можливості розгортання на віддалених територіях. Це може бути досягнуто оптимізацією алгоритму роботи вузла, зниженням споживання у режимі сну та застосуванням енергоощадних режимів передавання даних. Перспективним є також застосування адаптивного інтервалу опитування, за якого частота вимірювань збільшується у періоди швидкої зміни показників та зменшується за стабільного стану, що оптимізує співвідношення між інформативністю та енергоспоживанням.

На основі проведеного дослідження сформульовано рекомендації щодо вдосконалення підсистеми:

- для підвищення точності вимірювань доцільно застосувати калібрування недорогих датчиків за еталонними приладами та програмну корекцію показань залежно від температури й вологості;
- для збільшення дальності розгортання вузлів на відкритих територіях варто передбачити альтернативний канал зв'язку на основі технології LoRaWAN;
- для зменшення обсягу збережуваних даних доцільно застосувати агрегування історичних вимірювань та їх періодичне усереднення;
- для підвищення інформативності веб-додатку можна додати прогнозування рівня забруднення засобами машинного навчання;
- для зручності користувачів доцільно реалізувати мобільний застосунок та механізм сповіщень про перевищення граничних значень електронною поштою або повідомленнями;
- для підвищення надійності живлення вузлів варто оптимізувати алгоритм енергоспоживання та передбачити контроль рівня заряду акумулятора з передаванням цього показника на сервер.

Перспективним напрямом є також інтеграція розробленої підсистеми з державною системою моніторингу довкілля та відкритими платформами обміну екологічними даними, що дало б змогу узгоджувати локальні спостереження з даними офіційних постів та підвищувати достовірність загальної картини стану довкілля. Реалізація такої інтеграції потребує дотримання узгоджених форматів обміну даними, що враховано у відкритій архітектурі підсистеми.

3.4 Оцінка результатів та перспективи впровадження

Узагальнення результатів експериментального дослідження свідчить, що розроблена підсистема відповідає вимогам технічного завдання та забезпечує виконання покладених на неї функцій. Підтверджено коректність вимірювання, передавання, збереження та відображення даних, а також надійність роботи механізмів захисту від втрати даних. Орієнтовна точність вимірювань є достатньою для завдань локального моніторингу, а виявлені обмеження можуть бути усунуті застосуванням калібрування та програмної корекції показань.

Зведену оцінку відповідності підсистеми вимогам технічного завдання за основними критеріями наведено у таблиці 3.4. Оцінювання виконано за результатами проведеного експериментального дослідження.

Таблиця 3.4 – Оцінка відповідності вимогам технічного завдання

Вимога	Результат перевірки
Вимірювання показників	Виконано
Автоматичне передавання	Виконано
Буферизація за відсутності зв'язку	Виконано
Збереження у базі даних	Виконано
Формування попереджень	Виконано
Візуалізація даних	Виконано
Захищене передавання	Виконано

Як видно з таблиці, всі основні вимоги технічного завдання виконано, що підтверджує успішність розроблення підсистеми. Жодну з вимог не залишено нереалізованою, а працездатність відповідних функцій підтверджено експериментально. Це дає підстави вважати поставлену в роботі задачу повністю вирішеною.

Розроблена підсистема придатна до впровадження на території підприємств для контролю стану довкілля у зонах впливу виробництва, у населених пунктах для інформування населення про якість повітря, у навчальних закладах для організації навчальних та дослідницьких робіт, а також на природоохоронних територіях для спостереження за станом природних комплексів. Масштабованість підсистеми дає змогу розпочати впровадження з невеликої кількості вузлів та поступово розширювати мережу спостереження відповідно до потреб та наявних ресурсів.

Особливу цінність підсистема становить для навчальних закладів, оскільки поєднує практичну корисність із навчальним потенціалом. Студенти можуть брати участь у розгортанні та обслуговуванні вузлів, аналізі зібраних даних та вдосконаленні програмного забезпечення, що сприяє формуванню практичних навичок у галузях комп'ютерної інженерії, програмування та екології. Відкритість архітектури та використання доступних компонентів роблять підсистему зручною платформою для навчальних і дослідницьких проєктів, а також для подальших дипломних і курсових робіт.

Економічна доступність підсистеми та використання відкритих технологій роблять її привабливою для організацій з обмеженим бюджетом, для яких придбання стаціонарних систем моніторингу є недоцільним. Таким чином, розроблена підсистема має практичну цінність та потенціал для широкого застосування у сфері локального екологічного моніторингу.

Порівняння розробленої підсистеми з розглянутими у першому розділі системами-аналогами підтверджує досягнення поставлених цілей. Порівняно зі стаціонарними постами підсистема забезпечує значно вищу просторову щільність спостережень за суттєво нижчої вартості, поступаючись лише у метрологічній

точності, що є прийнятним для завдань локального моніторингу. Порівняно з комерційними хмарними мережами недорогих датчиків підсистема забезпечує повний контроль над даними та незалежність від зовнішніх сервісів за рахунок власного серверного застосунку. Реалізований механізм буферизації даних вирізняє підсистему серед типових аматорських рішень, що часто не забезпечують надійності передавання за нестабільного зв'язку.

Таким чином, розроблена підсистема поєднує переваги розглянутих класів систем: низьку вартість та високу щільність мереж недорогих датчиків, контроль над даними промислових рішень та відкритість академічних розробок. Це підтверджує обґрунтованість обраних на етапі аналізу технічних рішень та свідчить про практичну цінність виконаної роботи.

Слід зазначити, що розроблена підсистема не позиціонується як заміна високоточних стаціонарних систем моніторингу, а розглядається як їх доповнення на рівні локального спостереження. Оптимальним є комбіноване застосування, за якого нечисленні високоточні стаціонарні пости забезпечують еталонні вимірювання та слугують для калібрування, а щільна мережа недорогих вузлів розроблюваної підсистеми забезпечує детальну просторово-часову картину розподілу показників. Таке поєднання дає змогу досягти як високої точності, так і високої щільності спостережень, що неможливо у межах будь-якого з підходів окремо. Це визначає місце розробленої підсистеми у загальній структурі засобів екологічного моніторингу та окреслює перспективний напрям її практичного застосування.

3.5 Обмеження проведеного дослідження

Об'єктивна оцінка результатів вимагає врахування обмежень проведеного дослідження. Експеримент виконувався на макеті з одним вузлом збору даних, тоді як реальна підсистема передбачає розгортання мережі багатьох вузлів. Хоча

однотипність вузлів дає підстави поширювати отримані висновки на мережу загалом, окремі аспекти, пов'язані з одночасною роботою великої кількості вузлів та масштабуванням сервера, потребують додаткового дослідження на повномасштабній мережі.

Тривалість безперервного спостереження у межах експерименту обмежувалася добовим циклом, що достатньо для виявлення добових закономірностей, проте не охоплює сезонних змін та довготривалого дрейфу характеристик датчиків. Оцінювання довготривалої стабільності показань та визначення оптимальної періодичності калібрування потребують спостережень протягом значно тривалішого часу, що виходить за межі дипломної роботи та становить напрям подальших досліджень.

Оцінка точності вимірювань виконувалася шляхом порівняння з контрольними приладами обмеженого класу точності, а не з еталонними засобами вимірювальної техніки. Тому отримані значення похибки слід розглядати як орієнтовні, що характеризують відносну точність підсистеми, а не як результати метрологічної атестації. Для застосувань, що потребують гарантованої точності вимірювань, необхідне калібрування вузлів за повіреними еталонними приладами у спеціалізованій лабораторії. Зазначені обмеження не впливають на основні висновки щодо працездатності підсистеми, проте окреслюють умови коректної інтерпретації отриманих результатів та напрями подальшого вдосконалення методики дослідження.

Висновки до розділу 3. У розділі описано формування моделі та умови експериментального дослідження розробленої підсистеми, наведено й проаналізовано результати добового спостереження за показниками якості повітря. Підтверджено коректність вимірювання, передавання та збереження даних, а також надійність роботи механізму буферизації, що забезпечує відсутність втрат повідомлень за нестабільного зв'язку. Сформульовано рекомендації щодо подальшого розвитку підсистеми.

Результати експериментального дослідження засвідчили відповідність розробленої підсистеми вимогам технічного завдання. Отримані ряди даних

узгоджуються з відомими закономірностями динаміки забруднення, що підтверджує достовірність вимірювань, а оцінена точність є прийнятною для завдань локального моніторингу. Підсистема продемонструвала високу відмовостійкість завдяки дворівневому механізму захисту від втрати даних. Сформульовані рекомендації окреслюють напрями подальшого вдосконалення підсистеми, спрямовані на підвищення точності, розширення функціональних можливостей та зручності використання.

Проведене дослідження має не лише підтверджувальний, а й практичний характер, оскільки в його ході було відпрацьовано методику розгортання, налаштування та експлуатації підсистеми, що може бути використана під час її практичного впровадження. Отриманий досвід та виявлені особливості роботи підсистеми за різних умов становлять цінну інформацію для подальшого вдосконалення як апаратної, так і програмної частин. Таким чином, третій розділ завершує цикл розроблення підсистеми, підтверджуючи працездатність та практичну придатність створеного рішення для вирішення завдань моніторингу стану природного довкілля.

ВИСНОВКИ

У дипломній роботі вирішено актуальне завдання розробки інформаційної підсистеми моніторингу стану природного довкілля на основі недорогих мікроконтролерних вузлів та серверного веб-додатку. Поставлену мету досягнуто, а всі визначені завдання виконано.

Основні результати роботи полягають у такому:

- проаналізовано предметну галузь, призначення та завдання моніторингу стану довкілля, визначено базовий набір контрольованих показників якості повітря, води та ґрунту;
- виконано огляд і порівняльний аналіз систем-аналогів, що дав змогу обґрунтувати архітектуру на основі недорогих вузлів IoT з власним серверним застосунком;
- проаналізовано технології бездротового зв'язку та протоколи передавання даних, обрано Wi-Fi і протокол MQTT у поєднанні з REST API та форматом JSON;
- сформульовано технічне завдання та постановку задачі на розробку підсистеми;
- розроблено структурну й функціональну схеми підсистеми, обґрунтовано вибір мікроконтролера ESP32 та комплекту датчиків, виконано оціночні розрахунки енергоспоживання й трафіку;
- спроектовано структуру реляційної бази даних, розроблено алгоритм роботи вузла та програмне забезпечення вузла й сервера;
- проведено експериментальну перевірку підсистеми, що підтвердила коректність її роботи та надійність передавання даних із застосуванням буферизації.

Результати експериментального дослідження показали, що розроблена підсистема забезпечує коректне вимірювання й передавання показників стану довкілля, своєчасне формування попереджень у разі перевищення граничних значень та відсутність втрат даних за умов нестабільного зв'язку завдяки механізму локальної буферизації.

Розроблена підсистема може бути впроваджена для організації локального екологічного моніторингу на території підприємств, населених пунктів та природоохоронних об'єктів. Подальший розвиток підсистеми пов'язаний із калібруванням датчиків, розширенням каналів зв'язку та застосуванням методів прогнозування рівня забруднення.

Наукова новизна роботи полягає у застосуванні комбінованого підходу до передавання даних, що поєднує протокол MQTT для приймання телеметрії від вузлів з інтерфейсом REST для надання даних користувачеві, а також у реалізації дворівневого механізму захисту від втрати даних на основі локальної буферизації та проміжного зберігання повідомлень. Практична цінність роботи полягає у створенні економічно доступної масштабованої підсистеми, придатної до впровадження за обмежених ресурсів.

Отримані результати підтверджують досягнення поставленої мети та можуть бути використані як основа для подальших досліджень у сфері автоматизованого екологічного моніторингу, а також для практичного розгортання систем спостереження за станом довкілля на конкретних об'єктах.

Виконана робота має комплексний характер та охоплює всі етапи створення інформаційної підсистеми – від аналізу предметної галузі та формування технічного завдання до розроблення апаратного й програмного забезпечення та його експериментальної перевірки. Поєднання теоретичного аналізу, інженерного проектування та експериментального дослідження забезпечило обґрунтованість прийнятих рішень та достовірність отриманих результатів. Розроблена підсистема є завершеним технічним рішенням, придатним до практичного впровадження та подальшого розвитку відповідно до сформульованих рекомендацій.

Під час виконання роботи застосовано та закріплено знання з низки дисциплін фахової підготовки: комп'ютерних систем та мереж, мікропроцесорної техніки, баз даних, програмування, комп'ютерних мереж та захисту інформації. Розв'язання поставленої задачі потребувало комплексного застосування цих знань, що відповідає кваліфікаційним вимогам до фахівця спеціальності «Комп'ютерна інженерія». Робота поєднує апаратний та програмний аспекти, що

є характерним для задач комп'ютерної інженерії, та демонструє здатність до самостійного вирішення інженерних задач від постановки до експериментальної перевірки результату.

Підсумовуючи, можна стверджувати, що мету дипломної роботи досягнуто повною мірою: розроблено, реалізовано та досліджено інформаційну підсистему моніторингу стану природного довкілля, що відповідає сформульованим вимогам та підтвердила працездатність за результатами експериментального дослідження. Усі поставлені завдання виконано, а отримані результати мають як теоретичну, так і практичну цінність та можуть бути використані для подальшого розвитку засобів автоматизованого екологічного моніторингу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Закон України «Про охорону навколишнього природного середовища» від 25.06.1991 № 1264-XII. URL: <https://zakon.rada.gov.ua/laws/show/1264-12> (дата звернення: 20.05.2025).
2. Положення про державну систему моніторингу довкілля : затв. постановою Кабінету Міністрів України від 30.03.1998 № 391. URL: <https://zakon.rada.gov.ua/laws/show/391-98-п> (дата звернення: 20.05.2025).
3. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. Вид. офіц. Київ : ДП «УкрНДНЦ», 2016. 32 с.
4. Клименко М. О., Прищепя А. М., Вознюк Н. М. Моніторинг довкілля : підручник. Київ : Академія, 2006. 360 с.
5. Бабіченко В. М., Ніколаєва Н. В. Основи екологічного моніторингу : навч. посіб. Харків : ХНУ, 2014. 248 с.
6. Kolban N. Kolban's Book on ESP32. Texas : Leanpub, 2018. 567 p.
7. Espressif Systems. ESP32 Technical Reference Manual. Version 4.8. 2023. URL: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf (дата звернення: 20.05.2025).
8. OASIS. MQTT Version 5.0. OASIS Standard. 2019. URL: <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html> (дата звернення: 20.05.2025).
9. Fielding R. T. Architectural Styles and the Design of Network-based Software Architectures : dissertation. Irvine : University of California, 2000. 162 p.
10. Bray T. The JavaScript Object Notation (JSON) Data Interchange Format. RFC 8259. 2017. URL: <https://www.rfc-editor.org/rfc/rfc8259> (дата звернення: 20.05.2025).

11. Plantower. PMS5003 Digital Universal Particle Concentration Sensor : datasheet. 2016. 12 p.
12. Bosch Sensortec. BME280 Combined humidity and pressure sensor : datasheet. Rev. 1.6. 2018. 60 p.
13. Hanwei Electronics. MQ-135 Gas Sensor : technical data. 2014. 4 p.
14. World Health Organization. WHO global air quality guidelines. Geneva : WHO, 2021. 273 p.
15. Snyder E. G. et al. The Changing Paradigm of Air Pollution Monitoring. Environmental Science & Technology. 2013. Vol. 47, No. 20. P. 11369–11377.
16. Castell N. et al. Can commercial low-cost sensor platforms contribute to air quality monitoring? Environment International. 2017. Vol. 99. P. 293–302.
17. PostgreSQL Global Development Group. PostgreSQL 16 Documentation. 2023. URL: <https://www.postgresql.org/docs/16/> (дата звернення: 20.05.2025).
18. Date C. J. An Introduction to Database Systems. 8th ed. Boston : Addison-Wesley, 2004. 1024 p.
19. Flanagan D. JavaScript: The Definitive Guide. 7th ed. Sebastopol : O'Reilly Media, 2020. 706 p.
20. Stroustrup B. The C++ Programming Language. 4th ed. Boston : Addison-Wesley, 2013. 1368 p.
21. Schwartz M. Internet of Things with ESP32. Birmingham : Packt Publishing, 2018. 230 p.
22. Кафедра комп'ютерних систем та мереж Національного університету «Запорізька політехніка» : вебсайт. URL: <http://csn.zntu.edu.ua> (дата звернення: 20.05.2025).

23. Eclipse Foundation. Eclipse Mosquitto: An Open Source MQTT Broker : documentation. 2023. URL: <https://mosquitto.org/documentation/> (дата звернення: 20.05.2025).

24. Richardson L., Ruby S. RESTful Web Services. Sebastopol : O'Reilly Media, 2007. 446 p.

25. Hunt R. TCP/IP Network Administration. 3rd ed. Sebastopol : O'Reilly Media, 2002. 746 p.

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ПІДСИСТЕМИ

1 Повна назва підсистеми – інформаційна підсистема моніторингу стану природного довкілля.

2 Призначення підсистеми – автоматизований збір, передавання, збереження та візуалізація показників якості повітря, води та ґрунту.

3 Мета створення – забезпечення оперативного контролю стану довкілля на території об'єкта спостереження з можливістю масштабування.

4 Показники досягнення мети – повнота й достовірність зібраних даних, відсутність втрат повідомлень, оперативність формування попереджень.

5 Вимоги до підсистеми в цілому:

- дворівнева клієнт-серверна архітектура з проміжним рівнем обміну повідомленнями;
- автоматичне передавання даних від вузлів до сервера;
- збереження даних у реляційній базі даних;
- захищений канал доступу користувача до даних.

5.1 Вимоги до функцій – вимірювання показників із настроюваним інтервалом, буферизація за відсутності зв'язку, формування попереджень, візуалізація даних.

5.2 Вимоги до умов експлуатації – діапазон температур від мінус 10 до плюс 45 °С, автономне живлення вузлів, можливість збільшення кількості вузлів без зміни архітектури сервера.

ДОДАТОК Б

ФРАГМЕНТ СЦЕНАРІЮ СТВОРЕННЯ БАЗИ ДАНИХ

Лістинг Б.1 – Фрагмент сценарію створення бази даних

```
CREATE TABLE nodes (  
    id          SERIAL PRIMARY KEY,  
    name        VARCHAR(64) NOT NULL,  
    latitude    REAL,  
    longitude   REAL,  
    node_type   VARCHAR(16)  
);  
  
CREATE TABLE sensors (  
    id          SERIAL PRIMARY KEY,  
    node_id     INTEGER REFERENCES nodes(id),  
    type        VARCHAR(16),  
    unit        VARCHAR(16),  
    model       VARCHAR(32)  
);  
  
CREATE TABLE measurements (  
    id          BIGSERIAL PRIMARY KEY,  
    sensor_id   INTEGER REFERENCES sensors(id),  
    value       REAL NOT NULL,  
    ts          TIMESTAMP NOT NULL,  
    quality     SMALLINT DEFAULT 1  
);  
  
CREATE INDEX idx_meas_ts ON measurements(ts);
```

ДОДАТОК В

СЛАЙДИ ПРЕЗЕНТАЦІЇ

Національний університет «Запорізька політехніка»



ДИПЛОМНА РОБОТА

Розробка інформаційної підсистеми моніторингу стану природного довкілля

Виконав:
ст. групи КНТ-512з
Керівник:
доцент, к.т.н.

В.Ю. Боровик
І.Я. Зеленцова

Рисунок В.1 – Слайд 1

Актуальність теми

1. Зростання антропогенного навантаження на довкілля.
2. Необхідність безперервного контролю якості повітря, води та ґрунту.
3. Недоліки традиційних систем моніторингу:
 - висока вартість;
 - мала кількість постів спостереження;
 - низька оперативність.
4. Використання IoT-технологій дозволяє створювати дешеві та масштабовані системи моніторингу.

Рисунок В.2 – Слайд 2

Мета та завдання роботи

Мета:

Розробка інформаційної підсистеми моніторингу стану природного довкілля на основі ESP32 та серверного веб-додатку.

Основні завдання:

- Аналіз предметної галузі та систем-аналогів
- Формування технічного завдання
- Вибір елементної бази й розробка схем
- Проєктування бази даних і серверного веб-додатку
- Розробка ПЗ вузлів і сервера, експериментальна перевірка

Рисунок В.3 – Слайд 3

Архітектура підсистеми

Схема:



Функції:

- збір даних;
- передача телеметрії;
- збереження;
- аналіз;
- візуалізація.

Рисунок В.4 – Слайд 4

Контрольовані параметри

Повітря

- PM2.5
- PM10
- CO₂
- NO₂

Вода

- pH
- температура
- електропровідність
- розчинений кисень

Ґрунт

- вологість
- температура
- кислотність

Метеодані

- температура
- вологість
- атмосферний тиск

Рисунок В.5 – Слайд 5

Вибір технологій

Апаратна платформа

ESP32

- вбудований Wi-Fi
- низька вартість
- достатня продуктивність
- достатній обсяг пам'яті й портів вводу-виводу

Передача даних

- Wi-Fi
- MQTT (QoS 1)

Веб-система

- REST API
- JSON формат даних

Рисунок В.6 – Слайд 6

Структура бази даних

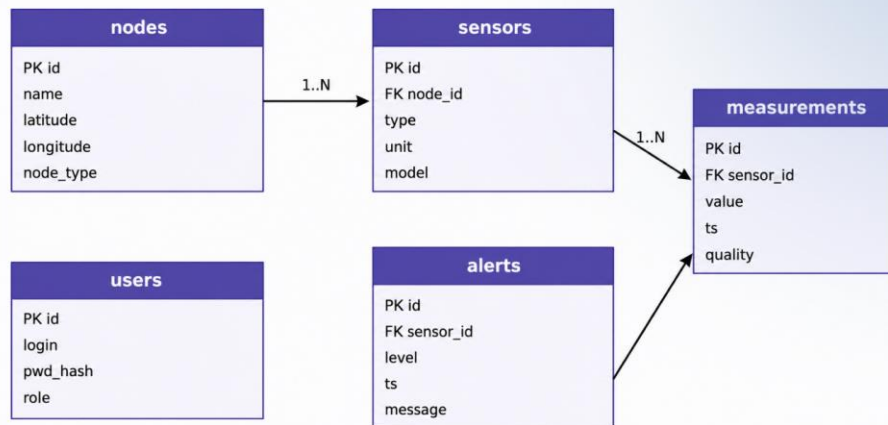


Рисунок В.7 – Слайд 7

Функціональні можливості:

- перегляд поточних показників;
- графіки змін параметрів;
- карта розташування вузлів;
- повідомлення про перевищення ГДК;
- історія вимірювань.



Рисунок В.8 – Слайд 8

Результати тестування

Перевірено:

- коректність збору даних;
- передачу через MQTT;
- запис у БД;
- відображення через веб-інтерфейс;
- формування попереджень.

Отримано стабільну роботу підсистеми та успішну передачу телеметрії.

Рисунок В.9 – Слайд 9

Переваги розробки

- Низька вартість вузлів.
- Масштабованість.
- Автоматизований збір даних
- Власний сервер і контроль даних.
- Відкрита архітектура.
- Робота в режимі, близькому до реального часу.

Рисунок В.10 – Слайд 10

Висновки

- Розроблено інформаційну підсистему моніторингу довкілля.
- Реалізовано вузли збору даних на ESP32.
- Створено серверну частину, БД та веб-додаток.
- Підтверджено працездатність системи експериментально.
- Підсистема придатна для практичного впровадження.

Рисунок В.11 – Слайд 11

ДЯКУЮ ЗА УВАГУ!

Рисунок В.12 – Слайд 12