

Міністерство освіти і науки України

Луцький національний технічний університет

(повне найменування закладу вищої освіти)

Факультет комп'ютерних та інформаційних технологій

(повне найменування факультету)

Кафедра комп'ютерної інженерії та безпеки

(повне найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «МАГІСТР»**

**ІНФОРМАЦІЙНО-АНАЛІТИЧНА СИСТЕМА МОНІТОРИНГУ
МІКРОКЛІМАТУ ПРИМІЩЕННЯ В РЕАЛЬНОМУ ЧАСІ**

**INFORMATION AND ANALYTICAL SYSTEM FOR MONITORING
INDOOR MICROCLIMATE IN REAL TIME**

спеціальність 123 Комп'ютерна інженерія

(шифр і назва спеціальності)

освітня програма Комп'ютерна інженерія

(назва освітньої програми)

Виконав: здобувач вищої освіти

групи КІм-21

Скиба Владислав Олегович

(підпис)

Керівник:

к.т.н., доцент

Христинець Наталія Анатоліївна

(підпис)

Кваліфікаційну роботу

допущено до захисту

« » 2025 р.

Гарант освітньої програми:

к.т.н., доцент

Гринюк Сергій Васильович

(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра комп'ютерної інженерії та безпеки

Ступінь вищої освіти: магістр

Галузь знань: 12 Інформаційні технології

Спеціальність: 123 Комп'ютерна інженерія

Освітня програма: «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

доц. Т.ТЕРЛЕЦЬКИЙ

« _____ » _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Скибі Владиславу Олегович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи Інформаційно-аналітична система моніторингу мікроклімату приміщення в реальному часі

Керівник роботи к.т.н., доцент Христинець Наталія Анатоліївна

затверджені наказом закладу вищої освіти від «17» червня 2025 року № 290/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи 09.12.2025р.

3. Вихідні дані до роботи Джерелом розробки є науково-технічна література та публікації в періодичних виданнях з даного питання, опубліковані зарубіжні роботи в даній області, різні інтернет-ресурси технічного спрямування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

Вступ

Дослідити сучасний стан технологій моніторингу мікроклімату

Дослідити та реалізувати технічні рішення для системи моніторингу мікроклімату

Розробити тестовий стенд моніторингу мікроклімату

Розробити інформаційно-аналітичну систему моніторингу мікроклімату

Зробити висновки на основі розробленої системи

5. Перелік графічного (ілюстративного) матеріалу:

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
<i>Сучасний стан технологій моніторингу мікроклімату</i>	<i>Христинець Н. А., доцент</i>		
<i>Дослідження та реалізація технічних рішень для системи моніторингу мікроклімату</i>	<i>Христинець Н. А., доцент</i>		
<i>Розробка інформаційно-аналітичної системи моніторингу мікроклімату</i>	<i>Христинець Н. А., доцент</i>		
<i>Нормоконтроль</i>	<i>Багнюк Н. В., доцент</i>		
<i>Гарант ОП</i>	<i>Гринюк С. В., доцент</i>		
<i>Показник запозичень тексту</i>		%	
<i>Академічна доброчесність</i>	<i>Міскевич О. І., ст.викладач</i>		

7. Дата видачі завдання 18.06.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	<i>Огляд літератури із досліджуваної проблеми</i>	До 15.09.2025 р.	
2.	<i>Дослідження сучасного стану технологій моніторингу мікроклімату та постановка завдань дослідження</i>	До 20.10.2025 р.	
3.	<i>Дослідження та реалізація технічних рішень для системи моніторингу мікроклімату</i>	До 25.10.2025 р.	
4.	<i>Розробка інформаційно-аналітичної системи моніторингу мікроклімату</i>	До 05.11.2025 р.	
5.	<i>Висновки та пропозиції</i>	До 07.11.2025 р.	
6.	<i>Формування списку використаних джерел</i>	До 10.11.2025 р.	
7.	<i>Формування додатків</i>	До 15.11.2025 р.	
8.	<i>Оформлення ілюстративного матеріалу</i>	До 17.11.2025 р.	
9.	<i>Нормоконтроль</i>	До 25.11.2025 р.	
10.	<i>Інструментальна перевірка на академічний плагіат</i>	До 02.12.2025 р.	
11.	<i>Представлення кваліфікаційної роботи бакалавра до захисту</i>	До 09.12.2025 р.	

Здобувач вищої освіти

(підпис)

Скиба В. О.

(прізвище, ініціали)

Керівник кваліфікаційної роботи

(підпис)

Христинець Н. А.

(прізвище, ініціали)

АНОТАЦІЯ

Скиба В. О. Інформаційно-аналітична система моніторингу мікроклімату приміщення в реальному часі. Рукопис.

Кваліфікаційна робота магістра ОП «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія. Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота складається з вступу, трьох розділів, висновків, списку використаних джерел, додатків.

Перший розділ присвячено огляду предметної області моніторингу мікроклімату приміщень: подано основні параметри мікроклімату та нормативні вимоги до них, проаналізовано сучасні апаратно-програмні рішення контролю якості повітря й виконано порівняння протоколів передавання даних HTTP REST, MQTT, WebSocket, Server-Sent Events щодо придатності для моніторингу в реальному часі.

У другому розділі обґрунтовано вибір апаратних і програмних засобів: мікроконтролера ESP32 та комплексу датчиків DHT22, BMP280, BH1750, PMS5003, MQ-2, MQ-7, MQ-135, визначено структуру прошивки для ініціалізації сенсорів, збирання та попередньої обробки даних, а також обрано платформу ASP.NET Core MVC і СУБД PostgreSQL та спроектовано інформаційну й логічну структури бази даних.

Третій розділ присвячено опису розробленої інформаційно-аналітичної системи: подано структуру веб-додатка, взаємодію API-контролерів з ESP32 через HTTP REST і механізм потокового оновлення інтерфейсу, реалізацію веб-інтерфейсу з графіками, індикаторами та системою порогових оповіщень, а також результати експериментальної перевірки швидкодії й точності та окреслено напрями подальшого розвитку.

Ключові слова: мікроклімат, моніторинг, інформаційно-аналітична система, ESP32, датчики, IoT, ASP.NET Core, Server-Sent Events, HTTP REST.

ANNOTATION

Skyba V. Information-Analytical System for Real-Time Indoor Microclimate Monitoring. Manuscript.

Master's qualification thesis in the educational programme «Computer Engineering», specialty 123 Computer Engineering. Lutsk National Technical University. Lutsk, 2025.

The master's thesis consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The first chapter is devoted to a review of the subject area of indoor microclimate monitoring: the main microclimate parameters and regulatory requirements for them are presented, modern hardware and software solutions for air-quality control are analysed, and data transmission protocols HTTP REST, MQTT, WebSocket and Server-Sent Events are compared in terms of their suitability for real-time monitoring.

The second chapter substantiates the choice of hardware and software tools: the ESP32 microcontroller and a set of sensors DHT22, BMP280, BH1750, PMS5003, MQ-2, MQ-7 and MQ-135 are selected; the firmware structure for sensor initialisation, data acquisition and pre-processing is defined; the ASP.NET Core MVC platform and PostgreSQL DBMS are chosen; and the information and logical structures of the database are designed.

The third chapter describes the developed information-analytical system: it presents the structure of the web application, the interaction of API controllers with the ESP32 via HTTP REST and the mechanism of streaming interface updates using Server-Sent Events, the implementation of the web interface with charts, indicators and a threshold-based alert system, as well as the results of experimental evaluation of performance and accuracy and outlines directions for further development.

Keywords: microclimate, monitoring, information-analytical system, ESP32, sensors, IoT, ASP.NET Core, Server-Sent Events, HTTP REST.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 СУЧАСНИЙ СТАН ТЕХНОЛОГІЙ МОНІТОРИНГУ МІКРОКЛІМАТУ	10
1.1 Загальні положення моніторингу мікроклімату.....	10
1.2 Використання IoT-рішення у системах моніторингу мікроклімату	16
1.3 Огляд існуючих систем моніторингу мікроклімату у приміщенні	18
РОЗДІЛ 2 ДОСЛІДЖЕННЯ ТА РЕАЛІЗАЦІЯ ТЕХНІЧНИХ РІШЕНЬ ДЛЯ СИСТЕМИ МОНІТОРИНГУ МІКРОКЛІМАТУ	27
2.1 Вимоги до інформаційно-аналітичної системи	27
2.2 Визначення допустимих параметрів мікроклімату	30
2.3 Огляд компонентів для системи моніторингу мікроклімату	33
2.3.1 Огляд мікроконтролерів	33
2.3.2 Огляд датчиків температури та вологості.....	39
2.3.3 Огляд датчиків пилу.....	44
2.3.4 Огляд датчиків чадного газу, якості повітря та горючих газів	48
2.3.5 Огляд датчиків атмосферного тиску та освітленості.....	52
2.4 Огляд мов програмування для розробки системи	56
2.4.1 Огляд Python	57
2.4.2 Огляд C#.....	58
2.4.3 Огляд C/C++.....	60
2.5 Дослідження протоколів зв'язку	63
2.5.1 Огляд HTTP.....	63
2.5.2 Огляд MQTT	66
2.5.3 Огляд WebSocket.....	68
2.5.4 Огляд Server-Sent Events.....	70
2.5.5 Обґрунтування вибору протоколу зв'язку	71
2.6 Реалізація тестового стенду системи.....	72
2.6.1 Програмне забезпечення розробленої IoT-системи	75

	7
2.6.2 Інформаційне забезпечення системи.....	76
РОЗДІЛ 3 РОЗРОБКА ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ.....	78
3.1 Структура інформаційно-аналітичної системи	78
3.2 Інформаційне забезпечення системи	80
3.3 Програмне забезпечення інформаційно-аналітичної системи.....	81
3.3.1 Програмне забезпечення мікроконтролера.....	81
3.3.2 Програмна реалізація веб-інтерфейсу	84
3.3.3 Комунікаційний рівень системи	87
3.4 Порівняння системи з існуючими рішеннями	89
3.5 Інтерфейс користувача.....	92
ВИСНОВКИ	105
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	108
ДОДАТКИ	113

ВСТУП

Актуальність теми. Забезпечення моніторингу мікроклімату в реальному часі є важливим чинником збереження здоров'я людей, їхньої працездатності та енергоефективності будівель. Використання вимірювальних приладів не дає змоги організувати безперервний контроль параметрів повітря, накопичення історичних даних і оперативне інформування користувача про відхилення від нормативних значень. Розвиток IoT-пристроїв, недорогих сенсорів та веб-технологій створює передумови для побудови інтегрованих інформаційно-аналітичних систем моніторингу мікроклімату в реальному часі. Тому актуальним є дослідження та практична реалізація ефективних рішень для збору, передавання, зберігання, аналізу й візуалізації параметрів мікроклімату із застосуванням сучасних протоколів обміну даними та веб-платформ.

Метою роботи є розроблення програмно-апаратного комплексу моніторингу мікроклімату приміщення в реальному часі, який забезпечує автоматизований збір, передавання, зберігання, аналіз і наочне подання параметрів повітря користувачеві.

Об'єкт дослідження – процес контролю параметрів мікроклімату у приміщеннях, що охоплює вимірювання, передавання, накопичення та використання даних про стан повітря.

Предмет дослідження – методи й засоби побудови інформаційно-аналітичної системи реального часу для моніторингу мікроклімату приміщення на основі мікроконтролера, комплексу сенсорів, веб-технологій та сучасних протоколів передавання даних.

Завдання, які необхідно виконати:

- дослідити існуючі системи моніторингу мікроклімату у приміщенні в реальному часі;
- визначити вимоги до інформаційно-аналітичної системи моніторингу мікроклімату;
- дослідити й порівняти протоколи передачі даних;

- спроектувати архітектуру системи, структуру бази даних та комунікаційного рівня моніторингу;

- розробити апаратно-програмний прототип системи та комплексу сенсорів;

- розробити веб-інтерфейс користувача;

- візуалізувати параметри мікроклімату в режимі реального часу та за обрані історичні періоди у веб-інтерфейсі;

- реалізувати систему оповіщення про перевищення порогових значень параметрів мікроклімату та виконати тестування й аналіз отриманих результатів.

Апробація результатів. Результати роботи представлені на IV Міжнародній конференції «Період трансформаційних процесів в світовій науці: задачі та виклики», яка проходила 28 листопада 2025 року у м. Полтава [1]. Матеріали апробацій подано в додатку А кваліфікаційної роботи.

РОЗДІЛ 1

СУЧАСНИЙ СТАН ТЕХНОЛОГІЙ МОНІТОРИНГУ МІКРОКЛІМАТУ

1.1 Загальні положення моніторингу мікроклімату

Поняття мікроклімату у сучасній науці охоплює набір метеорологічних та гігієнічних параметрів, які формують середовище на обмеженій території. У природних умовах мікроклімат описує відмінності кліматичних характеристик на невеликій ділянці порівняно із середнім для ширшої місцевості. В закритих приміщеннях поняття мікроклімату тлумачать як сукупність параметрів внутрішнього середовища – температури повітря, відносної вологості, швидкості руху повітря, атмосферного тиску, складу та чистоти повітря, а також освітленості та рівня шуму. Ці фактори впливають на теплообмін людини з оточенням і визначають її самопочуття, працездатність та безпеку [2].

Температурний режим повітря є ключовим чинником, що формує тепловий стан організму людини. Його оптимальні межі варіюються залежно від пори року та функціонального призначення приміщення. Наприклад, у житлових просторах та офісах нормативно встановлено інтервал 18-22 градуси, тоді як у закладах охорони здоров'я – 20-25. Відхилення від цих значень може призводити до переохолодження або теплового перевантаження організму [3-4].

Вологість повітря, зокрема її відносний показник, відображає вміст водяної пари в повітрі. Оптимальний діапазон 40-60 % сприяє захисту слизових оболонок від пересихання та запобігає розмноженню патогенних мікроорганізмів.

Повітряна рухливість безпосередньо впливає на теплове самопочуття людини й ефективність відведення надлишкового тепла. У зимовий період надмірна швидкість повітряного потоку небажана, натомість влітку допускається інтенсивніше переміщення повітряної маси для поліпшення комфортності.

Рівень вуглекислого газу є показником якості та свіжості повітря. У межах житлових або офісних приміщень він не повинен перевищувати 1000 ppm. Концентрація, що сягає 1600 ppm, свідчить про мінімальний приплив свіжого

повітря, еквівалентний приблизно 15 м³ на годину на людину. Перевищення цих меж, а також присутність летких органічних речовин, можуть спричинити головний біль, втому та зниження когнітивної активності.

Крім основних характеристик мікроклімату, на самопочуття також впливають освітленість, акустичне середовище, атмосферний тиск та електрофізичні властивості повітря, зокрема рівень іонізації. Ці параметри враховуються у розширених системах оцінювання мікроклімату.

Удосконалені системи формування мікроклімату одночасно контролюють термічні, гігromетричні, аеродинамічні та хімічні параметри. Взаємодія цих факторів визначає тепловий комфорт людини; його оцінюють за індексами PMV та PPD, які враховують температуру, вологість, швидкість руху повітря, теплове випромінювання та теплоізоляцію одягу. Окрім того, на сприйняття мікроклімату впливають індивідуальні особливості – вік, фізичний стан, рівень метаболізму, одяг, культурні звички [5].

Внутрішній мікроклімат має безпосередній вплив на фізіологічний стан людини. За даними гігієнічних досліджень, екстремальні значення температури, вологості або концентрації вуглекислого газу призводять до підвищення артеріального тиску, зміни серцевого ритму та частоти дихання. Окрім того, підвищений рівень CO₂ та низька іонізація повітря викликають головний біль, сонливість, зниження працездатності та концентрації уваги. Довготривале перебування у середовищі з підвищеною вологістю сприяє розвитку цвілі та грибків, що є причиною алергічних реакцій і захворювань дихальної системи [6].

Люди проводять до 22 годин на день у приміщеннях, тому якість повітря та термічні умови мають вирішальне значення для здоров'я, продуктивності й добробуту. Епідеміологічні дослідження показують, що незадовільний мікроклімат пов'язаний з підвищеним ризиком респіраторних, серцево-судинних та нейродегенеративних захворювань. Водночас близько 35 % працівників не задоволені термічними умовами на робочому місці, що призводить до зниження ефективності праці та додаткових економічних втрат.

Забезпечення комфортного мікроклімату має також енергетичний аспект. Близько 40 % світового енергоспоживання припадає на системи опалення, охолодження та вентиляції для підтримання необхідних параметрів внутрішнього середовища. Неефективні або застарілі системи не лише збільшують енергозатрати, а й часто не забезпечують належної якості повітря, що зумовлює необхідність оптимізації та сучасного моніторингу [7].

Мікроклімат є динамічною системою, на яку впливають змінні зовнішні та внутрішні фактори: погодні умови, архітектурні характеристики будівлі, наявність теплоізоляції, кількість та активність людей, робота електроприладів, процеси горіння чи приготування їжі, а також режими вентиляції. Через це параметри середовища можуть змінюватися протягом дня. Наприклад, швидке підвищення концентрації вуглекислого газу при перебуванні великої кількості людей у приміщенні може відбуватися протягом десятків хвилин, і без оперативного контролю це залишається непоміченим. Тому системи моніторингу мікроклімату мають забезпечувати безперервний або періодичний збір даних та їх аналіз.

Необхідність впровадження систем моніторингу мікроклімату обумовлюється низкою важливих причин.

Передусім це пов'язано із забезпеченням здоров'я та безпеки людей. Своєчасне виявлення відхилень від нормативних параметрів середовища дає змогу уникнути небезпечних станів, таких як перегрів, переохолодження, дефіцит рідини в організмі, а також отруєння токсичними газами або продуктами згорання.

Підтримка теплового комфорту є ще одним критично важливим аспектом. Стабільні мікрокліматичні умови сприяють зниженню психофізіологічного навантаження, покращують розумову працездатність і загальний добробут людини в умовах навчання, праці чи відпочинку.

З погляду енергоефективності системи моніторингу, дозволяють гнучко регулювати роботу опалювального, кондиціонувального обладнання відповідно до актуальних потреб. Контроль концентрації вуглекислого газу може бути

використаний для оптимізації повітрообміну залежно від кількості присутніх осіб, що сприяє зниженню енергоспоживання [8].

Моніторинг також відіграє важливу роль у збереженні матеріальних об'єктів. Установи, що опікуються збереженням культурних цінностей – музеї, архіви, бібліотеки, історичні споруди – вимагають стабільного мікроклімату, адже підвищена вологість і температурні коливання прискорюють деградацію деревини, паперу, тканин та інших чутливих матеріалів. Тривалий контроль параметрів середовища є запорукою збереження цих об'єктів у належному стані.

Окрім того, наявність системи моніторингу часто обумовлена вимогами чинного законодавства та регламентів. Санітарно-гігієнічні норми та будівельні стандарти передбачають контроль за дотриманням допустимих значень параметрів повітряного середовища, що має реалізовуватися відповідними технічними засобами [9].

Перші спроби впливати на мікроклімат приміщень пов'язані з регулюванням вентиляції та вологості у житлових спорудах і сховищах. У ХХ столітті з розвитком промисловості виникла потреба захищати музейні та бібліотечні колекції від шкідливих факторів середовища. Ці примітивні рішення ізолювали експонати від зовнішнього середовища, але не дозволяли точно керувати параметрами повітря.

Згодом із появою механічних систем кондиціювання та опалення у ХХ столітті з'явилися можливості активного регулювання температури й вологості. Проте вартість обладнання та складність експлуатації обмежували його застосування переважно у великих музеях та театрах. Інженери усвідомили, що мікроклімат є окремим, відокремленим середовищем, і що його потрібно ізолювати від амбієнтного середовища для досягнення стабільності [10]. У середині ХХ століття розвиток вентиляційних установок та кондиціонерів у житлових будинках поклав початок масовому регулюванню внутрішнього середовища.

Останнє десятиліття характеризується стрімким розвитком систем Інтернету речей та бездротових сенсорних мереж, що призвело до кардинальних

змін у моніторингу мікроклімату. Порівняно з традиційними стаціонарними датчиками, IoT-пристрої компактні, енергоефективні та здатні передавати дані у хмарні сервіси або системи управління у реальному часі. Висока щільність розміщення сенсорів дозволяє виявляти локальні коливання, спричинені змінами погоди або поведінкою користувачів.

Приклад сучасного рішення – бездротова мережа для контролю якості внутрішнього середовища у хмарочосі у Гонконгу, описана у 2024 р. Дванадцять сенсорних блоків збирали дані про температуру повітря, випромінювальну температуру поверхонь, відносну вологість, швидкість повітря, концентрацію вуглекислого газу, кількість твердих частинок, освітленість та рівень шуму. Датчики знімали покази щохвилини, передаючи їх Wi-Fi на сервер, де здійснювалася обробка та візуалізація; користувачі могли отримувати рекомендації та подавати відгуки через веб-інтерфейс. Подібні системи дозволяють адаптувати режими кондиціонування під фактичні потреби, підвищувати комфорт та скорочувати енергоспоживання.

У побутових умовах набувають популярності розумні термостати та датчики, інтегровані у системи розумного будинку. Вони можуть автоматично регулювати опалення, вентиляцію та освітлення з урахуванням присутності людей та показників мікроклімату. Використання мобільних застосунків дозволяє користувачам відстежувати стан середовища у реальному часі та отримувати попередження про небезпечні відхилення, наприклад щодо концентрації вуглекислого газу або твердих частинок.

Сучасні системи моніторингу збирають великі обсяги даних, що відкриває можливості застосування машинного навчання та прогнозування. Алгоритми аналізують часові ряди показників, визначають кореляції між параметрами, виявляють аномалії та прогнозують майбутні зміни на основі історичних даних. Це дозволяє оптимізувати управління HVAC-системами, планувати енергообмін, здійснювати діагностику обладнання та попереджати аварійні ситуації. Наприклад, за рівнем вуглекислого газу та температурою можна оцінити кількість людей у приміщенні та заздалегідь підсилити вентиляцію.

Провідні дослідники пропонують розширювати перелік контрольованих параметрів, включаючи концентрацію дрібнодисперсного пилю, леткі органічні сполуки, іонізацію повітря, рівень шуму та освітленість. Це пов'язано з розумінням, що фізичний та психічний комфорт формуються не лише температурою та вологістю, а й якістю повітря, звуковою та світловою обстановкою. Відповідно, сучасні стандарти Indoor Environmental Quality розглядають мікроклімат як комплексну характеристику, що об'єднує параметри повітряного середовища, теплового, акустичного та світлового комфорту [11].

Системи моніторингу мікроклімату використовуються у різних категоріях приміщень, наприклад де від параметрів повітря залежить безпека, комфорт та ефективність діяльності людини.

Моніторинг параметрів мікроклімату має важливе значення для широкого спектра об'єктів з огляду на їх функціональні особливості та вимоги до якості середовища.

У житлових приміщеннях контроль температури, вологості та вмісту вуглекислого газу сприяє запобіганню утворенню плісняви, зменшенню накопичення шкідливих сполук та забезпеченню здорового мікроклімату, що позитивно впливає на самопочуття мешканців і профілакує захворювання.

У службових та офісних приміщеннях мікрокліматичні умови безпосередньо пов'язані з продуктивністю праці, рівнем концентрації та загальним станом здоров'я працівників.

Для навчальних закладів дотримання нормативів температури, вологості й вентиляції є критичним чинником. Незбалансоване середовище, зокрема дефіцит кисню чи надлишок вологи, погіршує самопочуття учнів, знижує здатність до навчання та сприяє поширенню респіраторних захворювань.

У закладах охорони здоров'я – особливо в операційних блоках, палатах інтенсивної терапії та реанімаційних відділеннях – забезпечення чистоти повітря, стабільної температури та вологості є невід'ємною умовою як безпеки пацієнтів, так і ефективності медичних втручань.

На промислових підприємствах, зокрема в галузях, що пов'язані з високим рівнем ризику, таких як хімічна, металургійна чи будівельна промисловість, постійний контроль параметрів повітря дозволяє запобігати небажаному підвищенню температури, забезпечувати своєчасне виявлення токсичних речовин і гарантувати безпечні умови праці.

Складські комплекси, особливо ті, що призначені для зберігання продукції з особливими вимогами: продукти харчування, фармацевтична продукція, електроніка, потребують підтримання строго визначених мікрокліматичних умов для забезпечення збереження якості та властивостей товарів.

У сфері збереження культурної спадщини – музеях, бібліотеках, архівах – стабільність мікроклімату є ключовим фактором у попередженні деградації цінних експонатів, стародруків і рукописів.

У транспортній інфраструктурі належні мікрокліматичні умови створюють додатковий рівень комфорту та безпеки для пасажирів.

У серверних приміщеннях і центрах обробки даних температурно-вологісний режим повинен суворо контролюватися, оскільки будь-яке відхилення може призвести до збоїв у роботі чутливого обладнання або повної втрати працездатності систем.

1.2 Використання IoT-рішення у системах моніторингу мікроклімату

Концепція Інтернету речей передбачає створення мережі фізичних об'єктів, здатних збирати, передавати й аналізувати дані за допомогою датчиків та програмного забезпечення. Такі «розумні речі» охоплюють як побутові прилади, так і складні промислові системи. Однією з ключових особливостей IoT є взаємодія «машина-машина», що дозволяє пристроям обмінюватися інформацією без участі людини, підвищуючи ефективність роботи та рівень автоматизації. Важливу роль тут відіграють бездротові мережі та сучасні мікропроцесори, завдяки яким з'явилися компактні й доступні рішення для практичного впровадження IoT [12].

Застосування IoT суттєво розширює можливості моніторингу мікроклімату у приміщеннях, пропонуючи автоматизовані й «розумні» рішення для підтримання комфортних та безпечних умов. Користувачі можуть віддалено відстежувати температуру, вологість, рівень вуглекислого газу та інші параметри через смартфони чи комп'ютери. Система надсилає повідомлення при відхиленнях від норми, що дозволяє швидко реагувати на проблеми й запобігати шкоді для людей, обладнання чи матеріалів [13].

IoT-пристрої легко інтегруються з системами HVAC, зволожувачами та осушувачами повітря. На основі даних датчиків система автоматично регулює параметри: наприклад, підвищення рівня вуглекислого газу активує вентиляцію, а перевищення температури – кондиціонування. Це особливо важливо для офісів, навчальних та медичних закладів.

Ще одна перевага – накопичення та аналіз даних. Тривале збирання інформації дозволяє виявляти закономірності та оптимізувати роботу кліматичних систем залежно від часу доби, сезону чи активності користувачів. Використання алгоритмів штучного інтелекту й машинного навчання робить системи здатними прогнозувати зміни умов і адаптуватися до них автоматично.

Інтеграція з іншими компонентами «розумного будинку» чи офісу забезпечує додаткові можливості. Наприклад, система моніторингу може автоматично відкрити вікна або змінити положення жалюзі для підтримання комфортної температури та економії енергії.

Підвищують енергоефективність будівель, адже кліматичні системи можуть автоматично знижувати інтенсивність роботи у неробочий час або за відсутності людей. Це зменшує споживання енергії й витрати на утримання. Такі пристрої також здійснюють діагностику обладнання: при виявленні несправностей система може зупинити його роботу й повідомити користувача.

Отже, переваги впровадження IoT у моніторинг мікроклімату можна узагальнити так:

- віддалений контроль і керування параметрами середовища;
- автоматизація роботи кліматичних систем;

- аналіз даних та виявлення закономірностей;
- інтеграція з іншими «розумними» системами;
- підвищення енергоефективності та зниження витрат;
- профілактика аварій і зростання рівня безпеки.

1.3 Огляд існуючих систем моніторингу мікроклімату у приміщенні

Для того щоб забезпечити ефективний моніторинг мікроклімату та підтримання оптимальних умов розроблено низку систем, що відстежують температуру, вологість, рівень вуглекислого газу, леткі органічні сполуки та інші параметри.

Проаналізувавши сучасні рішення у цій сфері можна вибрати найбільш поширені пристрої, які відображають поточний стан галузі: Aqara TVOC AAQS-S01, Xiaomi MIIW, Bosean T-Z01, PROTMECH PTH-5 та IQAir AirVisual Pro. Вони відрізняються за характеристиками та функціональністю, але всі забезпечують вимірювання й аналіз умов у приміщенні.

Огляд розпочнемо з пристрою Aqara TVOC AAQS-S01, зовнішній вигляд якого показано на рисунку 1.1 [14].



Рисунок 1.1 – Монітор якості повітря Aqara TVOC AAQS-S01 [14]

Пристрій вимірює температуру, вологість та рівень летких органічних сполук з точністю до 0,01 мг/м³. Оснащений екраном E-Ink, що забезпечує чітке зображення навіть на сонці, має магнітне кріплення та живиться від двох батарейок CR2450. Для роботи потрібне підключення до Aqara Hub із підтримкою Zigbee 3.0; сумісний із Apple Homekit, Mi Home та іншими системами «розумного дому».

Недоліки: відстежує лише TVOC як показник якості повітря, не має сповіщень про порушення температури чи вологості та вбудованої пам'яті для збереження даних.

Пристрій Xiaomi MHPW оснащений кнопками керування, за допомогою яких користувач може змінювати одиниці вимірювання температури, налаштовувати поточний час і вибирати формат його відображення. Живлення забезпечує батарея типу CR2032 із орієнтовним терміном роботи до 12 місяців, яку за потреби можна легко замінити. На рисунку 1.2 подано задню панель термогігрометра Xiaomi MHPW. Пристрій має рідкокристалічний дисплей, на якому відображаються значення температури, відносної вологості, а також час і дата; у центральній частині екрана розміщено індикатор рівня комфорту, що ґрунтується на поточному значенні вологості повітря. Монітор підтримує три стани індикації рівня комфорту [15].

- сухо (жовтогарячий) – вологість нижче 30 %;
- комфортно (зелений) – вологість у межах 30-69 %;
- волого (синій) – вологість понад 70 %.



Рисунок 1.2 – Датчик температури та вологості Xiaomi MHPW [15]

Основними недоліками пристрою є відсутність контролю параметрів якості повітря: вуглекислий газ, ЛОС, пил та вбудованої пам'яті для збереження даних.

Далі розглянемо монітор якості повітря компанії PROTMECH моделі PTH-5. Зовнішній вигляд приладу наведений на рисунку 1.3 [16].



Рисунок 1.3 – Монітор якості повітря PROTMECH PTH-5 [16]

Портативний пристрій PROTMECH PTH-5 призначений для визначення основних показників повітря: температури, вологості та концентрації вуглекислого газу. Він обладнаний літій-іонним акумулятором 3,7 В формату 18650, який за потреби можна замінити у випадку виходу з ладу. Для вимірювання вуглекислого газу використовується сенсор, що працює на принципі недисперсійного інфрачервоного поглинання з температурною корекцією, що забезпечує точність у діапазоні від 0 до 5000 ppm. Це вигідно відрізняє його від монітора Bosean T-Z01, де концентрація вуглекислого газу визначалася опосередковано.

Рідкокристалічний дисплей з підсвіткою відображає температуру, вологість, рівень вуглекислого газу та заряд акумулятора. Додатково над

екраном розташовано світлодіодний індикатор, який у поєднанні з вбудованим п'єзоелементом попереджає користувача про перевищення граничних значень вуглекислого газу.

На зворотній стороні PROTMEХ PTH-5 знаходиться USB – порт для заряджання акумулятора, зворотню сторону зображено на рисунку 1.4.

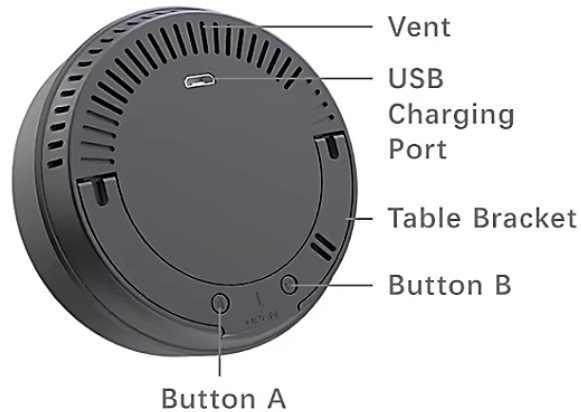


Рисунок 1.4 – Зворотня сторона PROTMEХ PTH-5 [16]

Наступним пристроєм для огляду буде монітор якості повітря компанії Bosean моделі T-Z01. Зовнішній вигляд пристрою моніторингу Bosean T-Z01 наведений на рисунку 1.5 [17].



Рисунок 1.5 – Монітор якості повітря Bosean T-Z01 [17]

Монітор Bosean T-Z01 здійснює вимірювання температури й вологості повітря, концентрації летких органічних сполук, а також рівня зважених твердих частинок. Прилад настільного типу, живлення забезпечується від вбудованого літій-іонного акумулятора з напругою 3,7 В та ємністю 1500 мА-год, що заряджається через USB-кабель. Оскільки акумулятор має нестандартний формфактор і не передбачає заміни користувачем, термін служби пристрою обмежується ресурсом батареї.

На рисунку 1.6 представлений графічний інтерфейс Bosean T-Z01. Інформація виводиться на дисплей з роздільною здатністю 320×240 пікселів. Порівняно з моніторами Aqara TVOC AAQS-S01 та Xiaomi MIIW, інтерфейс цього пристрою є більш інформативним, оскільки дозволяє переглядати статистику за останні 8 годин роботи. Крім того, на головному екрані відображається кольорова індикація рівня якості повітря у вигляді індексу якості повітря.



Рисунок 1.6 – Графічний інтерфейс Bosean T-Z01 [17]

Система індикації має чотири рівні:

- 0-50: Good (зелений) – повітря чисте, ризик для здоров'я відсутній;
- 51-100: Slight (жовтий) – загалом прийнятна якість, можливий дискомфорт лише для окремих чутливих осіб;
- 101-150: Moderate (помаранчевий) – потенційний негативний вплив на людей із хронічними захворюваннями, дітей та літніх осіб;
- 151-300: Serious (фіолетовий) – погіршення здоров'я можливе для всіх груп населення.

При відхиленні показників від норми активується звуковий сигнал. У меню налаштувань можна змінювати граничні значення, мову інтерфейсу (англійська або китайська), яскравість дисплея та час переходу в режим сну.

Завершуючи огляд, розглянемо монітор якості повітря IQAir AirVisual Pro. Пристрій обладнаний кольоровим LCD-дисплеєм (рис. 1.7), на якому відображаються дані за чотирма основними параметрами: температура, вологість, концентрація дрібнодисперсних частинок та рівень вуглекислого газу. Аналізатор працює в режимі реального часу й додатково зберігає показники за останні 24 години. На відміну від інших розглянутих моделей, AirVisual Pro підтримує мобільні додатки для iOS та Android, які через Wi-Fi дають змогу переглядати збережені дані й керувати пристроєм [18].



Рисунок 1.7 – Монітор якості повітря IQAir AirVisual Pro [18]

До недоліків IQAir AirVisual Pro можна віднести обмежений період збереження статистики лише 24 години.

Порівняємо основні характеристики пристроїв в табличному вигляді, зображено на таблиці 1.1.

Таблиця 1.1 – Основні характеристики пристроїв якості повітря

	Параметри які вимірюються пристроями						Зв'язок	Запис даних, годин	Ціна, грн.
	PM2,5	HCH O	TVOC	CO ₂	t, °C	%RH			
Aqara TVOC AAQS-S01	-	-	+	-	+	+	Zigbee	-	999
Xiaomi MIIW	-	-	-	-	+	+	-	-	460
Bosean T-Z01	-	+	+	-	+	+	-	8	5495
PROTMEX PTH-5	-	-	-	+	+	+	-	-	2070
IQAir AirVisual Pro	+	+	-	+	+	+	Wi-Fi	24	18000

Отже, розглянуті монітори мають низку обмежень, що знижують їхню ефективність і зручність використання. Більшість пристроїв або не зберігають статистику, або обмежуються лише однією добою, що унеможлиблює аналіз довготривалих змін та виявлення проблемних зон у приміщеннях. Крім того, вони потребують постійного ручного контролю.

Доступні за ціною моделі зазвичай вимірюють лише кілька параметрів, мають невисоку точність, не підтримують калібрування датчиків і не забезпечують дистанційного доступу до даних. Прилади з ширшим функціоналом коштують надто дорого, що ускладнює їхнє масове впровадження.

Існуючі недоліки підкреслюють потребу у створенні доступних та багатофункціональних рішень на базі технологій Інтернету речей, здатних відповідати сучасним вимогам до моніторингу мікроклімату в приміщеннях різного типу.

Розробка пристрою моніторингу якості повітря має забезпечити вимірювання найважливіших параметрів, які впливають як на комфорт, так і на безпеку перебування в приміщенні. Розробка повинна забезпечити моніторинг: рівня пилу, температури, вологості, вуглекислого газу, летких органічних сполук, чадного газу, а також загального тиску повітря. Ці показники дозволять своєчасно виявляти перевищення норм і запобігати негативному впливу мікроклімату на здоров'я.

Окрему увагу слід приділити системі збереження та передачі даних. Для повноцінного функціонування моніторингу недостатньо лише фіксації показників на екрані. Пристрій повинен мати можливість локального збереження вимірювань, що гарантує автономність роботи у разі відсутності доступу до мережі. Одночасно важливо реалізувати передачу даних у хмару через бездротове з'єднання, що відкриває можливість віддаленого доступу до статистики та актуальної інформації з будь-якого пристрою через вебінтерфейс або мобільний додаток.

Таким чином, функціональність пристрою повинна включати як технічні можливості вимірювання основних параметрів повітря, так і сучасну архітектуру зберігання та візуалізації даних, що дозволяє здійснювати моніторинг у реальному часі, вести облік динаміки змін і оперативно реагувати на критичні значення. Такий підхід не тільки підвищує зручність користування, а й суттєво розширює сферу практичного застосування пристрою в системах розумного дому, офісах, навчальних чи медичних закладах.

Дослідження наявних рішень у сфері моніторингу мікроклімату показало, що більшість пристроїв мають суттєві недоліки, які знижують їхню ефективність та зручність експлуатації. Зокрема, значна частина моделей не зберігає статистику або обмежується лише добовим проміжком, що унеможливорює аналіз довготривалих тенденцій і виявлення проблемних зон. Доступні для широкого кола користувачів прилади зазвичай характеризуються обмеженим функціоналом, невисокою точністю, відсутністю калібрування та дистанційного доступу до даних, що звужує сферу їх практичного застосування. Водночас

більш функціональні моделі відзначаються надто високою вартістю, яка обмежує їх масове використання для моніторингу мікроклімату у різних типах приміщень.

Тому створення інформаційно-аналітичної системи моніторингу мікроклімату в реальному часі є актуальним завданням, спрямованим на поєднання даних із різних датчиків, їх аналітичну обробку та підтримання оптимальних умов для людей і матеріальних цінностей. Розроблення таких систем потребує врахування нормативних вимог, історичного досвіду та сучасних технологічних можливостей.

РОЗДІЛ 2

ДОСЛІДЖЕННЯ ТА РЕАЛІЗАЦІЯ ТЕХНІЧНИХ РІШЕНЬ ДЛЯ СИСТЕМИ МОНІТОРИНГУ МІКРОКЛІМАТУ

2.1 Вимоги до інформаційно-аналітичної системи

У процесі проєктування та створення інформаційно-аналітичної системи моніторингу мікроклімату першочерговим завданням є формування технічних, функціональних та експлуатаційних вимог, від яких залежать архітектура рішення, вибір апаратних компонентів, програмних засобів і підхід до інтеграції системи в цільове середовище. Такі вимоги базуються на нормативно встановлених межах мікрокліматичних показників, потребах користувача в оперативному й достовірному контролі параметрів середовища, а також на принципах побудови сучасних IoT-систем. З огляду на необхідність роботи в режимі реального часу, автоматизованого збирання, передавання, зберігання та обробки даних, можливості масштабування, енергоефективності й кросплатформності, доцільним є використання інформаційної системи з веб-інтерфейсом, доступ до якої може здійснюватися з будь-якого пристрою, підключеного до мережі. При цьому важливо забезпечити чіткий поділ системи на рівень збору даних, рівень їх обробки та зберігання, а також рівень представлення результатів кінцевому користувачеві. Дотримання такої багаторівневої архітектури спрощує модернізацію окремих компонентів, дає змогу розширювати систему новими сенсорами або аналітичними модулями без суттєвого втручання в загальну структуру.

Було розроблено схему функціонування інформаційно-аналітичної системи моніторингу мікроклімату, яка зображена на рисунку 2.1. На ній відображено взаємодію трьох основних підсистем: мікроконтролера, серверної частини та веб-інтерфейсу користувача. На стороні мікроконтролера послідовно виконуються ініціалізація сенсорних модулів, зчитування поточних значень параметрів та формування JSON-пакета вимірювань, після чого сформовані дані передаються на сервер за допомогою HTTP POST-запиту.

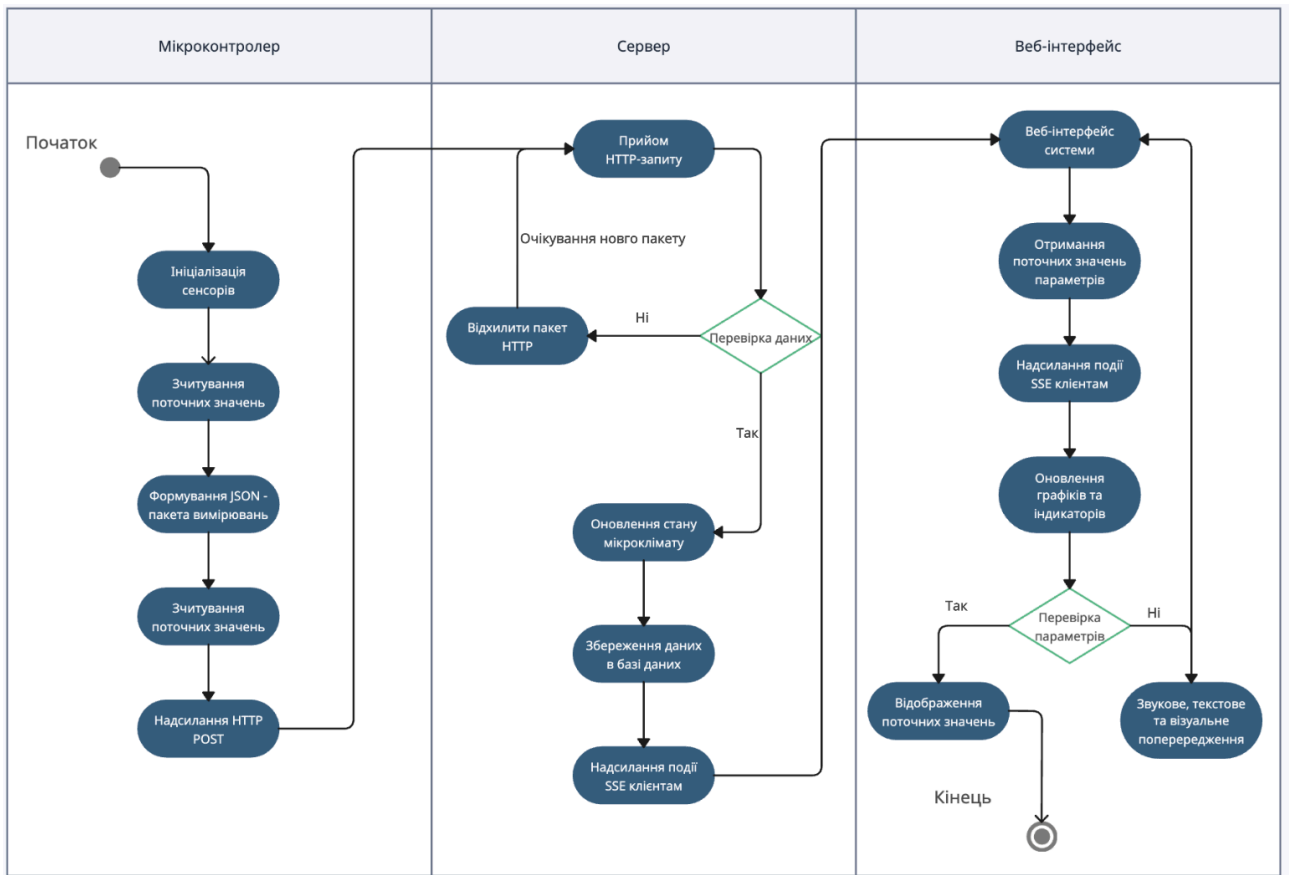


Рисунок 2.1 – Діаграма активності функціонування інформаційно-аналітичної системи моніторингу мікроклімату

Дана схема відображає системи моніторингу мікроклімату з використанням веб-інтерфейсу. Серверна частина приймає HTTP-пакет, та отримує дані з мікроконтролера, оновлює поточний стан мікроклімату, зберігає зібрані показники в базу даних та надсилає події SSE підключеним клієнтам. Веб-інтерфейс приймає цей потік подій, оновлює графіки та індикатори і перевіряє параметри на перевищення заданих порогових значень. За умови дотримання нормативів користувачу відображаються актуальні показники, а при виявленні відхилень формується звукове, текстове та візуальне попередження, що забезпечує оперативне реагування на зміну умов мікроклімату.

На основі діаграми активності можна виділити основні вимоги до інформаційно-аналітичної системи:

- забезпечення моніторингу параметрів у режимі реального часу;
- перевірка отриманих даних на коректність;

- обробка та реєстрація помилок під час передавання або зчитування даних;
- оповіщення користувача про вихід параметрів за встановлені допустимі межі;
- можливість ручного задання та коригування порогових значень;
- можливість переглядати дані в табличному вигляді;
- збереження і накопичення історії зміни параметрів з подальшою візуалізацією статистичних показників.

Розробка та реалізація зазначених вимог дозволяє створити адаптивну та функціонально систему, яка ефективно демонструє мікроклімат у режимі реального часу, сприяти підвищенню комфорту перебування в приміщенні, енергоефективності будівлі та своєчасному реагуванню на потенційно небезпечні зміни середовища.

Масштабованість є однією з ключових вимог до сучасної IoT-системи моніторингу мікроклімату. Архітектура повинна забезпечувати можливість підключення додаткових сенсорних модулів і розширення обчислювальних чи мережевих ресурсів без зміни основних функціональних блоків і без зупинки роботи системи. Для цього доцільно застосовувати модульний підхід, за якого окремі компоненти можуть незалежно оновлюватися, замінюватися або масштабуватися залежно від умов експлуатації та потреб користувачів.

Апаратне забезпечення має гарантувати достатні ресурси для збору, попередньої обробки та передавання даних, при цьому зберігаючи прийнятний рівень енергоспоживання. Для базового збору вимірювань доцільно використовувати мікроконтролери середнього класу, а в разі потреби у складніших алгоритмах аналізу або локальній аналітиці – одноплатні комп'ютери з розширеними обчислювальними можливостями. Важливими залишаються також зручність використання для некваліфікованого користувача, інтуїтивний інтерфейс перегляду показників і швидке виявлення відхилень, а також надійність і захищеність каналу передавання та доступу до системи.

Розробка та реалізація зазначених вимог дозволяє створити адаптивну та функціональну систему, яка ефективно відображає параметри мікроклімату в режимі реального часу, підвищує комфорт перебування в приміщенні, сприяє енергоефективності будівлі та забезпечує своєчасне реагування на потенційно небезпечні зміни середовища.

2.2 Визначення допустимих параметрів мікроклімату

Інформаційно-аналітична система моніторингу мікроклімату повинна працювати на основі чітко визначених нормативно допустимих меж параметрів середовища. Саме ці значення використовуються для налаштування порогів сповіщення, алгоритмів реагування та формування рекомендацій щодо поліпшення умов у приміщенні. До ключових показників належать температура та відносна вологість повітря, вміст зважених частинок, концентрації шкідливих газоподібних домішок, рівень освітленості та атмосферний тиск, що регламентуються чинними санітарними та будівельними нормами, а також міжнародними рекомендаціями.

Температура повітря та відносна вологість є базовими параметрами теплового комфорту. Для житлових, офісних і навчальних приміщень у холодний період року рекомендовано підтримувати температуру повітря в межах 20-22° С, у теплий – 22-25° С; значення нижче 18° С вважаються дискомфорними, а перевищення понад 26° С може спричиняти теплове перевантаження. Оптимальний діапазон відносної вологості становить 40-60 %, тоді як показники нижче 30 % призводять до пересихання слизових оболонок, а надмірна вологість сприяє розвитку плісняви та алергенів. Саме ці діапазони покладено в основу порогів контролю температури й вологості в системі моніторингу [19].

Для оцінювання запиленості повітря використовуються концентрації дрібнодисперсних фракцій PM_{2,5} і PM₁₀. За рекомендаціями ВООЗ значення не повинні перевищувати 15 мкг/м³ для PM_{2,5} та 45 мкг/м³ для PM₁₀, а

короткочасні пікові рівні – 25 мкг/м³ і 50 мкг/м³ відповідно; перевищення цих меж пов'язується з підвищеним ризиком серцево-судинних і респіраторних захворювань. Українські норми для РМ10 є менш жорсткими, однак для побутових приміщень доцільно орієнтуватися на більш консервативні міжнародні стандарти. У системі моніторингу як пороги сповіщення прийнято значення 25 мкг/м³ для РМ2,5 та 50 мкг/м³ для РМ10 [20-21].

Концентрації шкідливих газоподібних домішок визначають рівень хімічного забруднення повітря. Для оксиду вуглецю середньодобова концентрація в житлових приміщеннях не повинна перевищувати 2 мг/м³, а короткочасно допускається до 10 мг/м³; вищі рівні супроводжуються симптомами отруєння. Для сумарних летких органічних сполук бажаним вважається рівень до 300 мкг/м³, 300-1000 мкг/м³ відповідає помірному забрудненню, а понад 1000 мкг/м³ є небезпечним. Допустима концентрація формальдегіду у житлових приміщеннях не має перевищувати 0,1 мг/м³. Для горючих газів спрацювання сигналізації повинно відбуватися на рівні 5-10 % від нижньої межі вибухонебезпечної концентрації, що для метану становить приблизно 0,22-0,44 %. Зазначені значення прийнято як орієнтири для налаштування порогів газових датчиків у системі [22].

Освітленість та атмосферний тиск доповнюють картину мікроклімату, впливаючи на зоровий комфорт та самопочуття користувачів. Нормативи передбачають освітленість не менше 150 лк у житлових кімнатах, близько 300 лк у навчальних аудиторіях і 300-500 лк в офісах, тоді як у зонах з підвищеними візуальними навантаженнями вона може досягати 1000 лк. Надто низька чи надмірна освітленість призводить до перевтоми зору та зниження працездатності [23].

Нормальний атмосферний тиск становить близько 1013 гПа с, а прийнятним для більшості людей є діапазон 735-785 мм рт. ст.; вихід за ці межі може негативно впливати на метеочутливі групи населення. Тому в системі моніторингу реалізовано контроль освітленості та тиску з можливістю фіксації відхилень від зазначених діапазонів [24].

Загальні допустимі значення для всіх показників наведені в таблиці 2.1.

Таблиця 2.1 – допустимі значення параметрів мікроклімату.

Параметр	Допустимі значення
Температура повітря (зима)	20-22° С
Температура повітря (літо)	22-25° С
Відносна вологість	40-60 %
PM _{2,5}	≤ 15 мкг/м ³
PM ₁₀	≤ 45 мкг/м ³
PM _{2,5} пікове	≤ 25 мкг/м ³
PM ₁₀ пікове	≤ 50 мкг/м ³
СО	≤ 2 мг/м ³
СО короткочасне	≤ 10 мг/м ³
НСНО	≤ 0,1 мг/м ³
TVOC	≤ 300 мкг/м ³
TVOC помірне	300-1000 мкг/м ³
TVOC небезпечне	> 1000 мкг/м ³
СН ₄	0,22-0,44 % або 2200-4400 ppm
Освітленість (житло)	≥ 150 люкс
Освітленість (навчання)	≥ 300 люкс
Освітленість (офіс)	300-500 люкс
Освітленість (точні роботи)	до 1000 люкс
Атмосферний тиск нормальний	760 мм рт. ст. / 1013 гПа
Атмосферний тиск допустимий	735-785 мм рт. ст. / 980-1047 гПа

Узагальнюючи наведені допустимі значення параметрів мікроклімату, можна зазначити, що вони охоплюють як показники теплового комфорту, так і критерії запиленості, хімічного забруднення повітря, освітленості та атмосферного тиску, які безпосередньо впливають на здоров'я й працездатність людини. Сукупність цих діапазонів формує нормативну «зону комфорту», вихід за межі якої асоціюється з підвищенням ризиків респіраторних, серцево-судинних та професійних захворювань, а також з погіршенням самопочуття й зниженням продуктивності праці. Саме тому зазначені граничні значення покладено в основу налаштування порогів сповіщення в розробленій

інформаційно-аналітичній системі моніторингу мікроклімату, що дає змогу не лише фіксувати фактичні параметри середовища, а й оперативно виявляти небезпечні відхилення та формувати рекомендації щодо оптимізації умов у приміщенні.

2.3 Огляд компонентів для системи моніторингу мікроклімату

Огляд компонентів для системи моніторингу мікроклімату є важливим етапом для створення інформаційно-аналітичної системи, оскільки саме правильний вибір апаратних засобів визначає точність, стабільність і функціональність усієї системи. До складу таких компонентів входять датчики фізичних величин, мікроконтролери, модулі зв'язку, джерела живлення та допоміжні елементи, які разом утворюють апаратну платформу для збору, обробки та передавання даних. Кожен із зазначених елементів повинен відповідати специфічним технічним вимогам, які враховують характер контролюваного середовища, необхідну частоту вимірювань, точність, тип середовища передачі даних, а також вимоги до енергоспоживання та масштабу системи. Категорії компонентів, що використовуються у сучасних системах моніторингу мікроклімату, а також здійснено порівняльний аналіз найбільш розповсюджених варіантів для подальшого обґрунтованого вибору оптимальної конфігурації.

2.3.1 Огляд мікроконтролерів

Почнемо огляд з мікроконтролера Arduino MKR1000, який є одним із представників лінійки Arduino, орієнтованої на проєкти Інтернету речей. Цей мікроконтролер поєднує в собі простоту використання платформи Arduino з потужністю та можливостями бездротового з'єднання, що робить його привабливим варіантом для побудови систем моніторингу мікроклімату [25].

Arduino MKR1000 побудований на основі 32-бітного мікроконтролера ATSAMW25 від Microchip, який інтегрує ARM Cortex-M0+ та модуль Wi-Fi з підтримкою IEEE 802.11 b/g/n. Завдяки вбудованому криптографічному чипу,

пристрій підтримує апаратне шифрування, що дозволяє реалізувати безпечну передачу даних у мережах. Плата має компактні розміри та може працювати як від USB, так і від батареї, що є особливо важливо для автономних пристроїв моніторингу в умовах обмеженого доступу до джерел живлення. Мікроконтролер зображено на рисунку 2.2 [25].



Рисунок 2.2 – Мікроконтролер Arduino MKR1000 [25]

До ключових характеристик та переваг Arduino MKR1000 належать:

- 32-бітний процесор ARM Cortex-M0 до 48 МГц;
- 256 КБ Flash-пам'яті, 32 КБ SRAM;
- вбудований модуль Wi-Fi WINC1500;
- підтримка протоколів TLS/SSL;
- апаратне шифрування (CryptoAuthentication чип АТЕСС508);
- живлення: 3,7 В або через micro-USB;
- розміри: 61,5 × 25 мм;
- енергозберігаючі режими;
- сумісність з Arduino IDE та бібліотеками.

Завдяки вказаним технічним характеристикам Arduino MKR1000 є ефективною платформою для реалізації інтелектуальних систем збору та передавання даних. Наявність бездротового зв'язку, низьке енергоспоживання та

підтримка сучасних протоколів безпеки дозволяють використовувати цей мікроконтролер у проєктах, де важливо забезпечити стабільність, автономність та захищеність як у побутових, так і в промислових умовах.

Ще одним ефективним варіантом для побудови високопродуктивної системи моніторингу мікроклімату є мікроконтролер STM32F103C8T6 (рис. 2.3), який належить до 32-розрядної лінійки STM32 від компанії STMicroelectronics. Цей мікроконтролер реалізовано на основі ядра ARM Cortex-M3 і він має суттєву перевагу в обчислювальній потужності та енергоефективності порівняно з 8-бітними контролерами, такими як Arduino Nano або Uno.

STM32F103C8T6, відомий також під неофіційною назвою «Blue Pill», є одним із найпопулярніших варіантів для застосування в системах з високими вимогами до обробки даних, точності, швидкості реакції та масштабованості. Він підтримує сучасні комунікаційні інтерфейси, зокрема UART, I2C, SPI, CAN, USB, що дозволяє інтегрувати його як у прості локальні системи, так і в комплексні розподілені рішення інтернету речей [26].

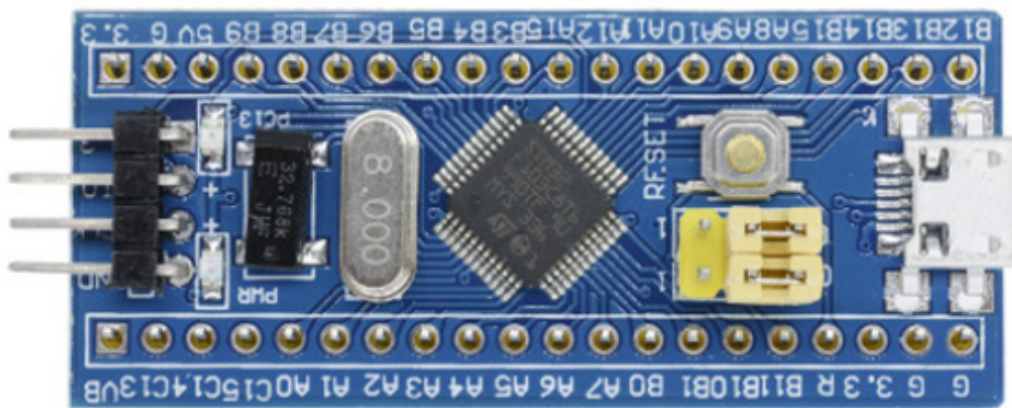


Рисунок 2.3 – Мікроконтролер STM32F03C8T6 ARM 32 Cortex-M3 STM32 [26]

Технічні характеристики:

- ядро: ARM Cortex-M3, 32-біт;
- тактова частота: до 72 МГц;
- Flash-пам'ять: 64 КБ;
- оперативна пам'ять: 20 КБ;

- кількість цифрових входів/виходів: до 37;
- кількість аналогових входів: до 10 каналів ADC;
- підтримка інтерфейсів: UART, I2C, SPI, USB, CAN;
- робоча напруга: 3,3 В;
- габарити плати: приблизно 52 x 22 мм.

STM32F103C8T6 є вигідним вибором для побудови масштабованих і надійних систем моніторингу мікроклімату з високими вимогами до точності вимірювань, стабільності та можливості розширення. Його технічні характеристики дозволяють не лише реалізувати збір та обробку екологічних параметрів, а й інтегрувати систему у ширший контекст автоматизованого керування будівлею або «розумного» середовища. Наявність багатоканальних АЦП, таймерів, інтерфейсів UART, I²C, SPI, CAN та USB спрощує під'єднання великої кількості датчиків і виконавчих пристроїв, а підтримка енергоощадних режимів дає змогу використовувати мікроконтролер у автономних або резервованих вузлах.

Серед сучасних мікроконтролерів, що знайшли широке застосування у системах моніторингу мікроклімату, особливе місце посідає ESP32 – високопродуктивна платформа з інтегрованими модулями бездротового зв'язку. Вона розроблена компанією Espressif Systems і є наступником популярного ESP8266, пропонуючи значно розширені функціональні можливості, мікроконтролер зображено на рисунку 2.4 [27].



Рисунок 2.4 – Мікроконтролер ESP32 [27]

ESP32 є універсальним рішенням, яке поєднує у собі двоядерний процесор, великий обсяг оперативної пам'яті, апаратні інтерфейси, а також вбудовану підтримку Wi-Fi та Bluetooth. Завдяки цьому мікроконтролер ідеально підходить для побудови автономних інтернет-речей систем, які вимагають бездротового з'єднання з сервером або хмарною платформою без використання зовнішніх модулів зв'язку. Його можливості забезпечують передачу даних у реальному часі, що критично важливо для своєчасного реагування системи моніторингу [27].

Крім того, ESP32 відзначається широкою підтримкою спільноти розробників, наявністю численних бібліотек для взаємодії з датчиками температури, вологості, тиску, концентрації пилу та газів, а також повною інтеграцією з платформами Arduino IDE, PlatformIO, ESP-IDF та іншими. Це спрощує процес розробки прототипу та масштабування системи.

Основні технічні характеристики ESP32:

- процесор: двоядерний Tensilica Xtensa LX6 до 240МГц;
- пам'ять: 520 КБ SRAM, до 16 МБ зовнішньої Flash;
- вбудовані інтерфейси: Wi-Fi 802.11 b/g/n, Bluetooth v4.2 BR/EDR та BLE;
- кількість GPIO: 34;
- аналогові входи: 18 каналів ADC 12-біт;
- цифрові інтерфейси: UART, SPI, I2C, PWM, I2S, CAN, SDIO;
- робоча напруга: 3,3 В;
- споживання енергії: підтримка режимів глибокого сну.

Серед основних переваг мікроконтролера ESP32 слід відзначити наявність вбудованих модулів Wi-Fi та Bluetooth, що усуває потребу у використанні додаткових плат для бездротового зв'язку. Завдяки високій обчислювальній потужності при відносно низькому енергоспоживанні він ідеально підходить для автономних систем. Також ESP32 має велику кількість цифрових і аналогових входів та виходів, що розширює можливості підключення різноманітних датчиків та виконавчих пристроїв. Мікроконтролер підтримує широкий спектр протоколів зв'язку, забезпечуючи гнучкість інтеграції в різні системи. Його

перевагами також є наявність відкритого, добре документованого середовища розробки, компактні габарити, а також доступна вартість, що робить його надзвичайно привабливим вибором для систем моніторингу мікроклімату.

Порівняння вище наведених мікроконтролерів зображено в таблиці 2.2.

Таблиця 2.2 Порівняння мікроконтролерів та їх характеристик

Модель	CPU	Частота	Ядер	RAM	FLASH	IO	Зв'язок	Ціна
Raspberry Pi Zero W	Broadcom BCM2835, ARM11	1 ГГц	1	512 МБ DDR2	micro-SD	2x UART, 2x SPI, 3x I2C, 2x PWM, 1x ADC, 6x GPIO	Wi-Fi 4, Bluetooth 4.1	~\$10
Radxa Zero	Rockchip RK3328, Cortex-A53	1,5 ГГц	4	512Мб-4ГБ LPDDR4	eMMC / micro SD	2x UART, 2x SPI, 3x I2C, 2x PWM, 1x ADC, 6x GPIO	Wi-Fi 4, Bluetooth 4.0	~\$15-50
Arduino MKR1000	ARM Cortex-M0+	48 МГц	1	32 КБ	256 КБ	1x UART, 1x SPI, 1x I2C, 12x PWM, 7x ADC, 7x GPIO	Wi-Fi 4	~\$30
STM32L475EI	ARM Cortex-M4	80 МГц	1	128 КБ	1 МБ	5x UART, 4x SPI, 3x I2C, 2x PWM, 3x ADC, 2x DAC, USB, CAN	Wi-Fi 4, Bluetooth 4.1, NFC	~\$50
ESP8266	Tensilica LX106	80-160 МГц	1	50 КБ	0,5-4 МБ	1x ADC, 1x UART, 1x I2C, 1x I2S, 1x SPI, 11x GPIO	Wi-Fi 4	~\$4
ESP32	Xtensa LX6	240 МГц	2	520 КБ	4-16 МБ	18x ADC, 3x SPI, 3x UART, 2x I2C, 2x DAC, 2x I2S, 1x CAN, 36x GPIO	Wi-Fi 4, Bluetooth 4.2	~\$6-9

На основі проведеного порівняння технічних характеристик мікроконтролерів, що можуть бути використані для реалізації системи моніторингу мікроклімату, найбільш доцільним вибором є ESP32. Цей мікроконтролер поєднує в собі високу обчислювальну потужність завдяки

двоядерному процесору Xtensa LX6 з тактовою частотою до 240 МГц, значний обсяг оперативної пам'яті та широкий набір периферійних інтерфейсів, зокрема 18 каналів ADC, 3 UART, 3 SPI, 2 I2C, 2 DAC, 2 I2S, CAN, 36 GPIO тощо.

Ключовою перевагою ESP32 є наявність вбудованих модулів Wi-Fi та Bluetooth v4.2, що дозволяє реалізувати бездротовий зв'язок без додаткових зовнішніх модулів, що знижує загальні витрати та спрощує апаратну реалізацію. Підтримка сучасних протоколів зв'язку, енергоефективні режими роботи та активна спільнота розробників роблять ESP32 ідеальним варіантом для проєктів Інтернету речей, які потребують автономності, стабільності та розширюваності.

2.3.2 Огляд датчиків температури та вологості

Одним із ключових завдань системи моніторингу мікроклімату є точне та стабільне вимірювання температури й відносної вологості повітря в режимі реального часу. Ці параметри безпосередньо впливають на комфорт перебування людини в приміщенні, працездатність, збереження технічного обладнання, а також є важливими індикаторами енергетичної ефективності будівлі. Тому вибір відповідного сенсора температури та вологості повинен ґрунтуватися не лише на точності та діапазоні вимірювання, а й на таких факторах, як стабільність показників, швидкість відгуку, тип інтерфейсу, сумісність із мікроконтролером, енергоспоживання та експлуатаційна надійність. У цьому підпункті буде розглянуто та порівняно найбільш поширені варіанти сенсорів для вимірювання температури й вологості, що використовуються в сучасних системах Інтернету речей.

Почнемо огляд з датчика DHT11, одного з найпоширеніших цифрових сенсорів для вимірювання температури та вологості, що широко використовується в недорогих проєктах мікрокліматичного моніторингу. Цей сенсор є популярним вибором серед початківців завдяки своїй простоті підключення, наявності цифрового виходу, а також низькій вартості.

DHT11 має вбудований NTC-термістор для вимірювання температури, емнісний сенсор для вимірювання вологості та мікроконтролер для обробки сигналу й передачі даних. Комунікація з основним мікроконтролером

відбувається через однопровідний цифровий інтерфейс, що дозволяє зменшити кількість необхідних з'єднань. Завдяки інтеграції основних компонентів, DHT11 може забезпечувати базові вимірювання температури та вологості з прийнятною точністю для побутових і навчальних застосунків.

Хоча DHT11 поступається більш точним та стабільним сенсорам, його характеристик достатньо для використання у простих системах, де не вимагається висока точність, але важливими є доступність, компактність і легкість у застосуванні. Датчик зображено на рисунку 2.5 [28].

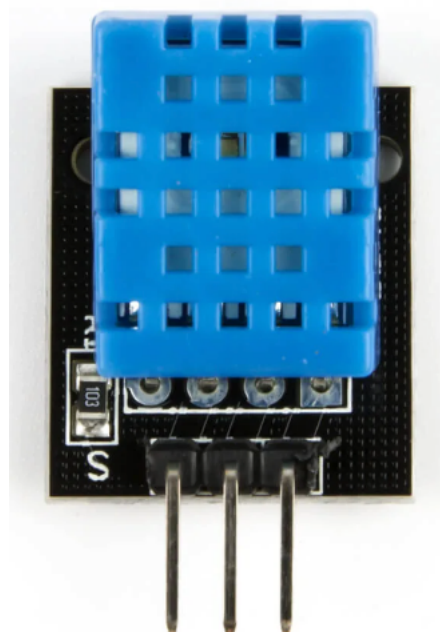


Рисунок 2.5 – Датчик DHT11 [28]

Основні технічні характеристики DHT11:

- діапазон вимірювання температури: від 0°C до $+50^{\circ}\text{C}$;
- точність температури: $\pm 2^{\circ}\text{C}$;
- діапазон вимірювання вологості: 20-90 % RH;
- точність вологості: $\pm 5\%$ RH;
- частота оновлення даних: 1 Гц;
- інтерфейс передачі даних: однопровідний цифровий;
- живлення: 3,3-5,5 В;

- споживання струму: $\sim 2,5$ мА під час вимірювання;
- форм-фактор: модуль з 3 або 4 контактами.

Його простота підключення, низьке енергоспоживання та широке поширення в навчальних наборах роблять його придатним для навчальних цілей, тестових макетів, а також домашніх систем з низькими вимогами до точності.

Наступним оглянемо DHT22, також відомий як AM2302 – цифровий датчик температури та вологості, який є вдосконаленою версією DHT11 і забезпечує вищу точність та ширший діапазон вимірювання. Він належить до сенсорів середнього класу та поєднує у собі доступність, точність і надійність, що робить його оптимальним варіантом для більшості проєктів моніторингу мікроклімату як побутового, так і напівпромислового призначення.

Як і DHT11, DHT22 використовує ємнісний сенсор вологості та термістор для температури, але має вдосконалений внутрішній мікроконтролер, що дозволяє зчитувати дані з точністю до десяткових дробів. Комунікація також здійснюється через однопровідний цифровий інтерфейс, проте на відміну від DHT11, DHT22 має довший інтервал передачі даних – приблизно кожні 2 секунди. Це слід враховувати при проєктуванні системи з високою частотою опитування датчиків. На відміну від DHT11. Живлення DHT22 може здійснюватися як від 3,3 В, так і від 5 В, що спрощує інтеграцію з більшістю популярних мікроконтролерних платформ, зокрема ESP32 та Arduino. Важливо також дотримуватися рекомендацій виробника щодо довжини з'єднувальних провідників та використання підтягувального резистора на лінії даних, оскільки це напряму впливає на стабільність обміну даними.

Датчик доступний у вигляді голого модуля або вже інтегрованого в пластиковий корпус з трьома або чотирма виводами для зручного підключення. Завдяки кращим характеристикам вимірювання, DHT22 може використовуватись у лабораторіях, медичних приміщеннях, системах «розумного дому», теплицях, архівах, де є підвищені вимоги до точності параметрів мікроклімату. Датчик зображено на рисунку 2.6 [29].

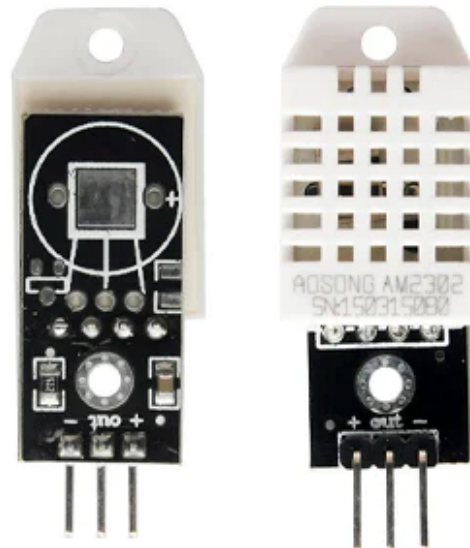


Рисунок 2.6 – Датчик DHT22 [29]

Основні технічні характеристики DHT22:

- діапазон вимірювання температури: від -40°C до $+80^{\circ}\text{C}$;
- точність температури: $\pm 0,5^{\circ}\text{C}$;
- діапазон вимірювання вологості: 0-100 %;
- точність вологості: $\pm 2-5\%$;
- частота оновлення даних: 0,5 Гц;
- інтерфейс передачі даних: однопровідний цифровий;
- живлення: 3,3-6,0 В;
- споживання струму: $\sim 1,5\text{ mA}$ при вимірюванні;
- форм-фактор: модуль або корпусований варіант AM2302.

DHT22 є значно точнішим і функціонально ширшим у порівнянні з DHT11, при цьому залишається простим у використанні та легко інтегрується в проекти з мікроконтролерами, зокрема ESP32, STM32 або Arduino. Це робить його одним із найбільш збалансованих рішень для систем екологічного моніторингу, де важлива не лише доступність, а й достовірність отриманих даних.

Далі розглянемо BME280 – високоточний цифровий датчик від компанії Bosch Sensortec, що поєднує в собі можливість одночасного вимірювання температури, відносної вологості та атмосферного тиску. Це один із

найсучасніших сенсорів для екологічного моніторингу, який забезпечує високу точність, енергоефективність та компактність, що робить його ідеальним рішенням для інтеграції в інформаційно-аналітичні системи мікроклімату.

Датчик BME280 використовує сучасну технологію MEMS та побудований на базі CMOSens-архітектури, яка дозволяє досягти стабільної точності при мінімальному енергоспоживанні. Завдяки наявності трьох сенсорів в одному корпусі, BME280 дозволяє значно скоротити кількість компонентів на платі, спростити збирання системи та забезпечити синхронний збір трьох ключових параметрів мікроклімату в одній точці. Це особливо важливо для задач довготривалого моніторингу та енергозбереження.

Комунікація з мікроконтролерами здійснюється через інтерфейси I2C або SPI, що робить BME280 сумісним із більшістю сучасних платформ, включаючи Arduino, ESP32, Raspberry Pi. Датчик зображено на рисунку 2.7 [30].

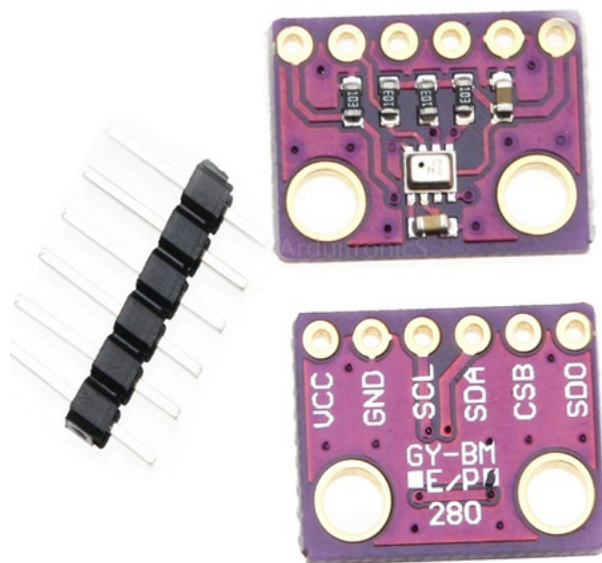


Рисунок 2.7 – Датчик BME280 [30]

Основні технічні характеристики BME280:

- діапазон вимірювання температури: -40°C до $+85^{\circ}\text{C}$;
- точність температури: $\pm 1,0^{\circ}\text{C}$;
- діапазон вимірювання вологості: 0-100 % RH;

- точність вологості: ± 3 % RH;
- діапазон атмосферного тиску: 300-1100 гПа;
- точність тиску: ± 1 гПа;
- інтерфейси: I2C (до 3,4 МГц) або SPI;
- живлення: 1,8-3,6 В;
- розміри корпусу: $2,5 \times 2,5 \times 0,93$ мм;
- енергоспоживання: в режимі сну < 1 μ А, у робочому режимі ~ 0.6 мА.

BME280 є одним із найкращих варіантів для створення точних, компактних та надійних систем екологічного моніторингу. Він забезпечує одночасне вимірювання трьох найважливіших параметрів мікроклімату, а саме: температури, вологості та атмосферного тиску – з високою точністю, стабільністю та низьким енергоспоживанням.

2.3.3 Огляд датчиків пилу

Першим розглянемо датчик GP2Y1010AU0F – один із найпоширеніших побутових оптичних сенсорів для виявлення пилу у повітрі, який виробляється компанією Sharp. Він працює на основі інфрачервоного випромінювання та здатен виявляти зважені частинки, що розсіюють світло у вимірювальній камері. Завдяки своїй доступності, простоті інтеграції та надійності, цей сенсор широко застосовується у проєктах екологічного моніторингу, системах очищення повітря та домашньої автоматизації [31].

Принцип роботи GP2Y1010AU0F полягає у виявленні кількості розсіяного світла, яке генерується частинками пилу під дією інфрачервоного світлодіода, розміщеного в оптичній камері. Змінюючи інтенсивність розсіяного сигналу, сенсор формує аналоговий вихідний сигнал, пропорційний концентрації пилу. Незважаючи на те, що він не забезпечує розділення за фракціями, його дані можуть бути використані для орієнтовної оцінки чистоти повітря.

Датчик має низьке енергоспоживання, що дозволяє використовувати його в автономних системах, зокрема в комбінації з мікроконтролерами типу ESP32 або Arduino Nano. Його невеликі розміри, простота підключення до аналогового входу, та відсутність необхідності в складних протоколах обміну роблять

GP2Y1010AU0F привабливим рішенням для базового пилового моніторингу. Датчик зображено на рисунку 2.8.



Рисунок 2.8 – Датчик пилу GP2Y1010AU0F [31]

Основні технічні характеристики:

- принцип дії: інфрачервоне розсіювання світла;
- тип виходу: аналоговий;
- діапазон виявлення: від 0 до 0,5 мг/м³;
- напруга живлення: 5 В;
- споживання струму: до 20 мА;
- час відгуку: ≤ 1 с;
- робоча температура: від -10 до $+65^{\circ}$ С;
- габарити: 46 × 30 × 17,6 мм.

GP2Y1010AU0F є економічно доцільним вибором для нескладних систем моніторингу, де важливими є низька ціна, енергоефективність та проста інтеграція. Водночас для завдань, що потребують більшої точності або класифікації фракцій пилу, доцільніше використовувати лазерні сенсори.

Наступним розглянемо датчик DSM501A – недорогий сенсор для виявлення зважених частинок у повітрі, який часто застосовується в побутових системах очищення повітря та проектах Інтернету речей, зображено на. Цей

сенсор призначений для виявлення частинок пилу розміром приблизно від 1 до 2,5 мкм, а також може частково фіксувати більші фракції до PM10 [31].

Принцип роботи DSM501A базується на методі інфрачервоного розсіювання світла: коли пил проходить через оптичну камеру, він розсіює інфрачервоне світло, що фіксується фотодетектором. Сенсор видає цифровий ШІМ-сигнал, де тривалість імпульсу прямо пропорційна концентрації пилу. Це дає змогу мікроконтролеру зчитувати частоту та тривалість імпульсів для оцінки рівня забруднення.

На відміну від аналогових сенсорів, таких як DSM501A (рис. 2.9) не потребує ADC-перетворення, що спрощує підключення до широкого кола мікроконтролерів, зокрема ESP32, Arduino Uno/Nano та STM32. Завдяки простоті інтерфейсу, компактним розмірам і низькому енергоспоживанню, датчик ідеально підходить для малопотужних автономних систем.



Рисунок 2.9 – Датчик пилу DSM501A [32]

Основні технічні характеристики:

- принцип дії: інфрачервоне розсіювання світла;
- тип виходу: цифровий;
- діапазон виявлення: частинки >1 мкм;
- вихідна напруга: 0-5 В;
- споживання струму: до 90 мА;

- робоча напруга: 5 В;
- робоча температура: від -10°C до $+65^{\circ}\text{C}$;
- частота PWM: приблизно 1 Гц;
- розміри: $59 \times 46 \times 18\text{мм}$.

Завершимо огляд сенсорів пилу описом PMS5003 – одного з найпопулярніших лазерних датчиків для вимірювання концентрації дрібнодисперсних частинок у повітрі. Цей сенсор розроблений компанією Plantower і широко застосовується в системах екологічного моніторингу, очищення повітря, IoT-рішеннях, а також у сертифікованих комерційних пристроях для оцінки якості повітря [32].

Принцип роботи базується на лазерному розсіюванні світла – частинки пилу, що проходять через оптичну камеру сенсора, розсіюють лазерне світло, яке детектується фотодіодом. Це дає змогу визначати концентрацію частинок PM_{1,0}, PM_{2,5} та PM₁₀ з високою точністю. PMS5003 оснащено вбудованим мікропроцесором, який обробляє дані в реальному часі та передає готові результати по UART.

На відміну від дешевших сенсорів, PMS5003 забезпечує стабільні й достовірні вимірювання завдяки температурній компенсації, захисту від пилу та наявності фільтрації. Підключення до мікроконтролера виконується через UART, що забезпечує надійну інтеграцію у складні системи. Зображено на рисунку 2.10.



Рисунок 2.10 – Датчик пилу PMS5003 [32]

Основні технічні характеристики:

- принцип дії: лазерне розсіювання;
- діапазон вимірювання: 0-1000мкг/м³;
- дискретність: 1 мкг/м³;
- показники: PM1,0, PM2,5, PM10;
- інтерфейс: UART TTL;
- робоча напруга: 5 В;
- споживання струму: ~100 мА;
- час відгуку: ≤ 1 с;
- робоча температура: –10° С до +60 ° С;
- термін служби: ≥ 3 роки;
- габарити: 50 × 38 × 21мм.

Цей сенсор є ідеальним для проєктів, де потрібна достовірна оцінка стану повітря, включаючи критичні додатки в медичних установах, лабораторіях, серверних кімнатах і розумних будівлях.

2.3.4 Огляд датчиків чадного газу, якості повітря та горючих газів

Одним із найпоширеніших датчиків для виявлення диму та горючих газів у повітрі є MQ-2. Він використовується у системах пожежної сигналізації, побутових детекторах газу, а також в IoT-рішеннях для контролю мікроклімату. Завдяки високій чутливості до пропану, бутану, водню та інших летких сполук, MQ-2 дає змогу своєчасно виявляти потенційно небезпечні концентрації газів і підвищувати рівень техногенної та побутової безпеки приміщень.

MQ-2 здатен виявляти такі речовини, як пропан, метан, бутан, водень, дим, спирти та LPG. Датчик має високу чутливість, короткий час відгуку та аналоговий вихід, який легко інтерпретується мікроконтролером. Працює за допомогою напівпровідникового сенсора з нагрівачем, який активує реакцію діоксиду олова на присутність горючих речовин у повітрі (рис. 2.11) [33].



Рисунок 2.11 – Датчик MQ-2 [33]

Основні характеристики:

- виявлювані гази: дим, водень, метан, пропан, бутан, алкоголь;
- напруга живлення: 5 В;
- вихід: аналоговий та цифровий;
- час прогріву: 20 секунд;
- робоча температура: від -10 до $+50^{\circ}$ С;
- чутливість: від 300 до 10 000 ppm.

Датчик MQ-7 призначений для виявлення монооксиду вуглецю – одного з найнебезпечніших забруднювачів повітря, який не має запаху чи кольору. Його використання є особливо актуальним у приміщеннях з газовими плитами, камінами, а також з газовими котлами або поганою вентиляцією [34]. Завдяки можливості безперервного моніторингу концентрації СО цей датчик дає змогу своєчасно формувати попереджувальні сигнали та знижувати ризик отруєння людей чадним газом.

Особливістю MQ-7 (рис. 2.12) є двофазний режим нагріву, де сенсор чергує фази високої температури для очищення чутливого шару та низької температури для вимірювання. Це забезпечує кращу селективність до СО порівняно з іншими сенсорами.



Рисунок 2.12 – Датчик MQ-7 [34]

Основні характеристики:

- виявлюваний газ: CO;
- діапазон вимірювання: 10-1000 ppm;
- напруга живлення: 5 В;
- вихід: аналоговий;
- робоча температура: від -10°C до $+50^{\circ}\text{C}$;
- час прогріву: 60 секунд;
- висока точність у виявленні CO з мінімальним впливом інших газів.

Датчик MQ-135 (рис. 3.13) призначений для моніторингу загальної якості повітря та виявлення летких органічних сполук, а також аміаку, бензолу, чадного газу та диму. Завдяки широкому спектру виявлюваних речовин, MQ-135 часто використовується у системах вентиляції, клімат-контролю та екологічного моніторингу зображено на.

Сенсор має високу чутливість до змін складу повітря і забезпечує стабільний вихідний сигнал у режимі аналогового напруженого виходу. Його опір змінюється залежно від концентрації шкідливих речовин, що дозволяє побудувати просту систему порогового контролю [35].

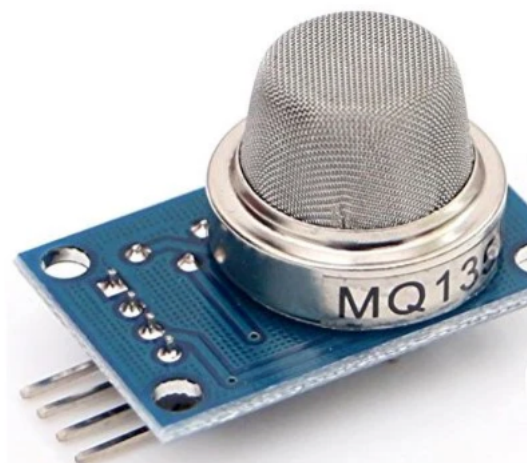


Рисунок 2.13 – Датчик MQ-135 [35]

Основні характеристики:

- виявлювані гази: аміак, бензол, спирти, чадний газ, дим, TVOC;
- напруга живлення: 5 В;
- вихід: аналоговий;
- діапазон вимірювання: від 10 до 1000 ppm;
- робоча температура: від -10°C до $+50^{\circ}\text{C}$;
- час прогріву: до 60 секунд;
- широка чутливість до різних газів, зручний для попередньої оцінки повітря.

Описані вище датчики серії MQ: MQ-2, MQ-7 та MQ-135 – демонструють високу ефективність і доцільність застосування в контексті побудови інформаційно-аналітичної системи моніторингу мікроклімату. Їх здатність виявляти широкий спектр шкідливих газів і домішок, включаючи чадний газ, леткі органічні сполуки, дим і горючі гази, дозволяє забезпечити всебічний контроль якості повітря в контрольованому середовищі. Простота інтеграції з мікроконтролерами, низька вартість і підтримка аналогового виходу роблять ці сенсори оптимальним вибором для реалізації надійної, масштабованої та енергоефективної системи моніторингу як у побутових, так і в промислових умовах.

2.3.5 Огляд датчиків атмосферного тиску та освітленості

Почнемо огляд з датчика BMP280 призначений для точного вимірювання барометричного тиску та температури. Він відрізняється більшою точністю та енергоефективністю в порівнянні з попередником BMP180 (рис. 2.14) [36]. BMP280 активно використовується в мобільних, метеостанційних та IoT-рішеннях для моніторингу зміни тиску в приміщенні або на вулиці, з метою аналізу погоди або контролю умов навколишнього середовища.

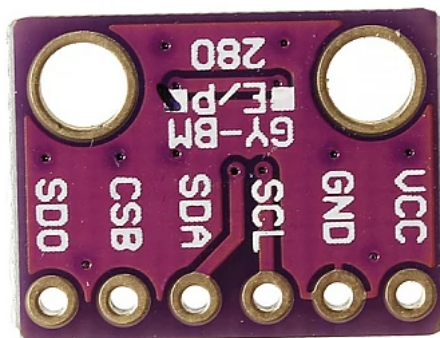


Рисунок 2.14 – Датчик BMP280 [36]

Основні характеристики BMP280:

- тиск: 30-1100 hPa, точність ± 1 hPa;
- вологість: 0-100 %, точність ± 3 %;
- температура: від -40° C до $+85^{\circ}$ C, точність $\pm 1^{\circ}$ C;
- напруга живлення: 1,8-3,6 В;
- інтерфейси: I²C, SPI;
- розмір: 3,6 × 3,8 мм;
- низьке енергоспоживання: до 3,6 μ A.

BH1750 – це цифровий сенсор освітленості, що вимірює рівень освітлення в люксах. Його основною перевагою є простота інтеграції завдяки інтерфейсу I²C та виведенню результатів без потреби в складних перерахунках, датчик зображено на рисунку 2.15. BH1750 забезпечує високу точність навіть за

низького або змінного освітлення, що робить його придатним для моніторингу денного світла в приміщеннях, офісах або лабораторіях [37].

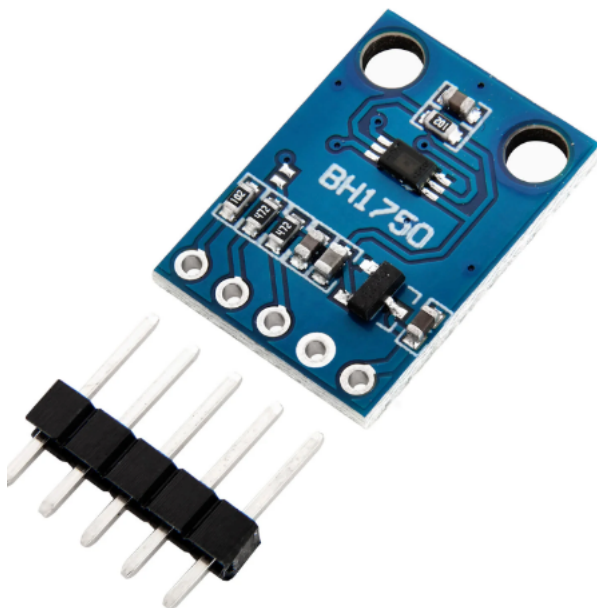


Рисунок 2.15 – Датчик BH1750 [37]

Основні характеристики:

- тип сенсора: цифровий фотометр для вимірювання освітленості;
- діапазон вимірювання: 1-65535 люкс;
- роздільна здатність: 1 люкс;
- інтерфейс зв'язку: I²C;
- напруга живлення: 2,4-3,6 В;
- струм споживання: ~0,12 мА;
- час вимірювання: приблизно 120 мс;
- кут огляду: близько 120°;
- робоча температура: від -40° С до 85° С;
- розміри корпусу: приблизно 3,0 × 3,0 мм.

TSL2561 – це точний датчик освітлення, який має два фотодіоди: один чутливий до інфрачервоного світла, інший – до видимого спектра. Таке поєднання дозволяє точно визначати рівень освітленості та компенсуючи вплив ІЧ-випромінювання, зображено на рисунку 2.16. Він широко використовується в

системах автоматичного керування освітленням, мобільних пристроях, а також у «розумному» середовищі для коректного вимірювання світлових умов [38].



Рисунок 2.16 – Датчик TSL2561 [38]

Основні характеристики:

- тип сенсора: цифровий датчик освітленості з подвійним фотодіодом;
- діапазон вимірювання: 0,1-40000 люкс;
- роздільна здатність: залежить від режиму підсилення і часу інтеграції, до 0,1 люкс;
- канали вимірювання: два незалежні канали;
- інтерфейс зв'язку: I²C;
- напруга живлення: 2,7-3,6 В;
- споживання струму: ~0,6 мА у робочому режимі, ~2 мкА у режимі очікування;
- час інтеграції: 13,7 мс, 101 мс або 402 мс;
- робоча температура: від -30° С до +70° С;
- вбудований переривання для сигналізації перевищення порогів освітленості.

Проведений огляд апаратних компонентів для побудови інформаційно-аналітичної системи моніторингу мікроклімату дозволив визначити оптимальні рішення серед широкого спектра доступних

мікроконтролерів та датчиків. Найбільш доцільним для реалізації даної системи визнано мікроконтролер ESP32. Він забезпечує високу обчислювальну потужність, вбудовані бездротові модулі Wi-Fi та Bluetooth, великий набір апаратних інтерфейсів, підтримку різноманітних протоколів зв'язку та широкі можливості для автономної роботи. Також ESP32 має добре підтримуване програмне середовище з відкритим кодом, компактні розміри та низьке енергоспоживання, що робить його універсальним рішенням для систем Інтернету речей.

Щодо сенсорної частини, для вимірювання температури та вологості обрано датчики DHT22 і BME280, які забезпечують високу точність і стабільність вимірювань. Для контролю концентрації пилу було проаналізовано кілька моделей, серед яких найбільш точним та сучасним є лазерний сенсор PMS5003, що дозволяє вимірювати фракції PM1,0, PM2,5 та PM10. Контроль забруднювальних газоподібних речовин, зокрема чадного газу, летких органічних сполук, аміаку, диму та горючих газів, реалізується за допомогою сенсорів MQ-серії, зокрема MQ-135, MQ-2 та MQ-7, які є простими у використанні, доступними за ціною та забезпечують необхідний рівень чутливості до відповідних речовин. Для моніторингу освітленості у приміщенні були відібрані датчики BH1750 та TSL2561, які дозволяють точно визначати рівень освітлення в люксах, адаптуються до зовнішніх умов та мають низьке енергоспоживання. Контроль атмосферного тиску, який є важливою частиною мікрокліматичного моніторингу, забезпечується сенсорами BMP280 та BME280.

На основі технічного, функціонального та економічного аналізу було сформовано остаточну конфігурацію системи, яка базується на мікроконтролері ESP32 з інтегрованими датчиками, що охоплюють всі ключові параметри мікроклімату. Обрані компоненти дозволяють забезпечити безперервний моніторинг, високу точність, енергоефективність та можливість масштабування системи в рамках розумного середовища або автоматизованого контролю стану приміщень.

2.4 Огляд мов програмування для розробки системи

У процесі розробки інформаційно-аналітичної системи моніторингу мікроклімату ключовим рішенням є вибір основної мови програмування для серверної частини та веб-інтерфейсу. У даній роботі пріоритетною мовою програмування буде C# на платформі .NET з використанням архітектурного патерну ASP.NET Core MVC, саме ця зв'язка найкраще відповідає вимогам до продуктивності, масштабованості, підтримки сучасних веб-технологій та зручності супроводу, оскільки моніторинг мікроклімату потребує часте оновлення даних, Python для цього не зовсім підходить, оскільки ця мова програмування дуже повільна, хоча в ній є багато корисних бібліотек, але для моніторингу в реальному часі пріоритетним буде швидкість передача даних, та простота реалізації. Патерн MVC забезпечує чітке розділення рівнів представлення, бізнес-логіки та доступу до даних, що спрощує розширення функціональності, впровадження нових сенсорів, модулів аналітики та засобів візуалізації без зміни базової архітектури, чого C/C++ не може забезпечити, що ускладнює розробку, а також виникають проблеми з кешуванням даних, в C# це робиться автоматично, а в C/C++, це робиться в ручну. Платформа .NET надає розвинену бібліотеку Entity Framework Core, засоби роботи з HTTP, JSON, реальним часом, потужні інструменти розробки Visual Studio, Visual Studio Code та високу продуктивність завдяки JIT-компіляції й оптимізованому виконанню коду.

Порівняно з Python та C/C++, C# .NET є більш збалансованим вибором для серверної та веб-частини системи моніторингу. Python добре підходить для швидкого прототипування та аналізу даних, проте як інтерпретована динамічно типізована мова він суттєво поступається C# за швидкістю в обчислювально інтенсивних та високонавантажених веб-сервісах, складніше забезпечує статичний контроль типів і жорстку структурування великих проєктів. Мови C/C++ забезпечують максимальну продуктивність і тому доцільні на мікроконтролерному рівні, але їх використання для розробки веб-застосунку

потребувало б значно більших витрат часу на реалізацію мережевої логіки, безпеки, ORM та інтерфейсу користувача. Таким чином, комбінація C/C++ на рівні мікроконтролера та C# .NET з шаблоном MVC на рівні серверної й клієнтсько-серверної частини забезпечує оптимальне поєднання низькорівневої ефективності й високорівневої гнучкості, що робить C# практично ідеальним вибором для реалізації інформаційно-аналітичної системи моніторингу мікроклімату, для того щоб більш детально порівняти ці мови програмування, оглянемо їх окремо.

2.4.1 Огляд Python

Python є однією з найпоширеніших мов програмування загального призначення, що активно використовується у наукових дослідженнях, автоматизації, обробці даних та машинному навчанні. Це мова високого рівня з динамічною типізацією та лаконічним синтаксисом, завдяки чому вона є зручною як для швидкого прототипування, так і для написання складних скриптів інтеграції. Розвинена екосистема бібліотек NumPy, Pandas, Matplotlib, SciPy робить Python природним вибором для задач аналізу та візуалізації даних, у тому числі даних, що надходять від сенсорів мікроклімату [39].

У систем Інтернету речей Python часто використовується як проміжна ланка між пристроями збору даних та серверною частиною. Мова надає зручні засоби для роботи з HTTP та MQTT-інтерфейсами, взаємодії з базами даних, побудови REST-API, а також інтеграції алгоритмів інтелектуального аналізу. Для вбудованих платформ розроблено MicroPython – полегшений інтерпретатор Python, який дозволяє програмувати мікроконтролери на знайомій високорівневій мові замість традиційних C/C++, що спрощує зміну логіки роботи пристрою та експериментування з алгоритмами.

Водночас ключові особливості Python накладають обмеження при виборі його як основної платформи для розроблення інформаційно-аналітичної системи моніторингу мікроклімату. Інтерпретована природа мови та наявність GIL зумовлюють нижчу продуктивність і гіршу масштабованість порівняно з компільованими мовами на кшталт C# .NET, особливо в частині обробки великої

кількості одночасних запитів у режимі реального часу. Крім того, екосистема Python для веб-розробки менш тісно інтегрована з інструментами корпоративного стеку Microsoft, ORM Entity Framework Core та шаблоном ASP.NET Core MVC, обраними для реалізації веб-інтерфейсу й серверної логіки в даному проєкті. Саме тому Python доцільно розглядати як допоміжний інструмент для експериментів, аналітики або можливого майбутнього впровадження моделей прогнозування, але не як основну мову реалізації інформаційно-аналітичної системи моніторингу мікроклімату.

2.4.2 Огляд C#

C# – це сучасна об'єктно-орієнтована мова програмування, яка була створена корпорацією Microsoft у межах платформи .NET і стала однією з найпопулярніших мов для розробки різноманітного програмного забезпечення, включаючи вебсайти, десктопні додатки, серверні сервіси, мобільні застосунки та IoT-системи. Завдяки постійній підтримці з боку Microsoft і величезній спільноті розробників, мова C# не лише еволюціонувала до високопродуктивного інструменту, але й стала універсальним рішенням для побудови комплексних, масштабованих і безпечних систем, включно з інформаційно-аналітичними платформами моніторингу навколишнього середовища [40].

У контексті побудови системи моніторингу мікроклімату в реальному часі C# відіграє ключову роль у реалізації як бекенд-частини обробка запитів, зберігання даних, взаємодія з мікроконтролерами, так і фронтенд-інтерфейсів вебпанелі керування, графіки, таблиці, дашборди. Зокрема, екосистема ASP.NET Core забезпечує потужну основу для створення вебінтерфейсів, що можуть функціонувати як на локальному сервері, так і в хмарі, що забезпечує мобільність, надійність та зручність доступу до системи. ASP.NET Core має підтримку мультиплатформності: Windows, Linux, macOS, що дозволяє розгорнути систему на будь-якому серверному середовищі, включно з контейнерами Docker або віртуальними машинами.

Однією з найсильніших сторін C# у веброзробці є застосування архітектурного патерну Model-View-Controller, що розділяє додаток на три логічні компоненти. Модель відповідає за роботу з даними зокрема, структура сенсорних показників, їх зберігання в базі даних, перевірка порогових значень, представлення – за візуальне подання інформації користувачеві графіки температури, таблиці вологості, діаграми перевищення рівня вуглекислого газу, а контролер – за логіку обробки взаємодії між користувачем і системою отримання нових даних, запуск обробки, створення звітів тощо. Цей підхід дозволяє розділити зони відповідальності між різними частинами коду, що значно полегшує підтримку, розширення та масштабування програми.

Окрім MVC, C# у середовищі ASP.NET Core дозволяє реалізовувати додаткові патерни проектування, які особливо актуальні в системах моніторингу:

- Repository – для абстрагованої роботи з базою даних, що дозволяє легко змінити тип сховища без зміни логіки програми;
- Dependency Injection – для ефективного управління залежностями між компонентами, що підвищує тестованість і зменшує зв'язність системи;
- Observer – для реалізації підписки на події зміни параметрів мікроклімату;
- Singleton – для забезпечення єдиного екземпляру сервісу реального часу, що збирає та розподіляє дані;
- Factory – для гнучкого створення об'єктів сенсорів або з'єднань з мікроконтролерами в залежності від конфігурації системи.

Важливою перевагою C# порівняно з іншими мовами є інтеграція з ORM-технологією Entity Framework Core, яка дає змогу зручно працювати з базами даних, здійснюючи запити мовою LINQ, з повною підтримкою міграцій, валідації, конфігурації сутностей і зв'язків. Для систем моніторингу це означає можливість створення складної структури з логами, профілями користувачів, історією показників, правилами обробки даних – без потреби вручну писати SQL-запити.

Завдяки бібліотеці SignalR, яка також підтримується в ASP.NET Core, можна реалізовувати передачу даних у реальному часі без постійного оновлення сторінки, що особливо важливо в інформаційно-аналітичних системах, де дані оновлюються щосекунди або щохвилини. SignalR підтримує WebSocket, Server-Sent Events та Long Polling, що дозволяє адаптуватися до будь-якого типу клієнтського середовища. Таким чином, користувач бачить зміни показників одразу після надходження нових значень із мікроконтролера, без затримки або перезавантаження сторінки.

Також C# дозволяє легко реалізувати безпечну автентифікацію та авторизацію, підтримуючи стандартні механізми реєстрації, ролей, політик доступу, OAuth2, JWT, інтеграцію з Active Directory, що критично важливо у системах з обмеженим доступом до аналітичних даних, зокрема в навчальних закладах, медичних установах або підприємствах.

На рівні інтерфейсу C# дозволяє використовувати такі технології, як Blazor WebAssembly – інноваційна модель вебпрограмування, яка дає змогу створювати клієнтські додатки на C# замість JavaScript. Це означає, що весь стек може бути реалізований однією мовою, що значно спрощує розробку, супровід і командну співпрацю.

У підсумку, мова програмування C# в поєднанні з ASP.NET Core, MVC, SignalR, Blazor, Entity Framework та іншими технологіями екосистеми .NET є ідеальним вибором для побудови складних, надійних та масштабованих систем моніторингу мікроклімату. Вона забезпечує зручний, безпечний та швидкий розвиток проекту з повною інтеграцією серверної логіки, бази даних та користувацького інтерфейсу. Завдяки цьому розробник має у своєму розпорядженні всі необхідні засоби для створення ефективної та зручно керованої інформаційно-аналітичної системи нового покоління.

2.4.3 Огляд C/C++

C та C++ є одними з найдавніших і водночас найпотужніших мов програмування, які й сьогодні залишаються базовими для розроблення програмного забезпечення вбудованих систем, драйверів пристроїв,

мікроконтролерів та систем реального часу. Вони забезпечують прямий доступ до апаратних ресурсів, роботу з пам'яттю на низькому рівні та тонке керування регістрами периферійних модулів, що робить ці мови особливо актуальними в галузі Інтернету речей і побудови вузлів збору даних [41].

У системах моніторингу мікроклімату C/C++ зазвичай застосовуються на рівні прошивки мікроконтролера, зокрема під час програмування плат ESP32, STM32, Arduino Uno, Nano чи MKR1000. Саме за допомогою цих мов реалізується зчитування показників із датчиків, їхня попередня фільтрація та нормалізація, передавання даних через інтерфейси UART, SPI, I²C, формування пакетів HTTP або MQTT, а також алгоритми енергозбереження й переходу пристрою в режими сну. Робота безпосередньо з портами вводу/виводу і регістрами дає змогу досягти високої швидкодії та мінімальних затримок, що є критичним для систем реального часу.

Мова C традиційно використовується для створення мінімалістичних, максимально компактних прошивок, де визначальними є обсяг коду, стабільність і низьке енергоспоживання. C++ як надмножина C підтримує об'єктно-орієнтований підхід, що дозволяє структурувати прошивку у вигляді класів і модулів, спрощуючи повторне використання коду й розширення функціональності. Це зручно для побудови бібліотек роботи з різними сенсорами, протоколами зв'язку та конфігураційними підсистемами.

Важливою перевагою C/C++ є широка підтримка в інструментах розробки для мікроконтролерів та наявність великої кількості готових бібліотек для роботи з сенсорами, мережевими модулями, Wi-Fi, Bluetooth, HTTP, MQTT. Завдяки цьому розробник може поєднувати низькорівневий код з використанням готових компонентів, що прискорює створення прошивки. Крім того, мінімальні накладні витрати під час виконання роблять C/C++ придатними для пристроїв з живленням від батареї, де критичною є кожна мілісекунда роботи процесора та кожен байт пам'яті [41].

Незважаючи на переваги C/C++ для мікроконтролерного рівня, ці мови є менш придатними для реалізації серверної частини інформаційно-аналітичної

системи, веб-інтерфейсу та аналітики даних. Розроблення повноцінного вебзастосування на C/C++ потребує значно більше зусиль, не має такої глибокої інтеграції з сучасними веб-фреймворками, ORM, системами автентифікації та інструментами розгортання, як це забезпечує платформа C# .NET з шаблоном MVC. Крім того, ручне керування пам'яттю та відсутність високорівневих засобів для швидкої побудови REST-API, роботи з базами даних і шаблонізаторами інтерфейсу ускладнюють супровід і масштабування серверної частини. Тому в межах даної роботи C/C++ доцільно використовуються лише на рівні прошивки мікроконтролера, тоді як для реалізації серверної логіки та веб-інтерфейсу оптимальним є вибір C# .NET та архітектури MVC.

2.4.4 Обґрунтування вибору мови програмування

Python, безперечно, є потужною мовою для обробки даних, швидкого прототипування та взаємодії з API. Однак, у масштабованих багатофункціональних системах з чітко структурованою архітектурою, суворими вимогами до стабільності та безпеки, вона поступається за гнучкістю у веб-розробці та модульності.

C і C++ залишаються незамінними для низькорівневого програмування мікроконтролерів, забезпечуючи максимальну ефективність, контроль над ресурсами та роботу у реальному часі. Водночас ці мови не підходять для реалізації серверної логіки, взаємодії з базами даних та побудови сучасного користувацького інтерфейсу, особливо якщо йдеться про хмарні або веборієнтовані системи.

Тому для розробки серверної частини системи моніторингу мікроклімату, зокрема веб-інтерфейсу, обробки запитів, зберігання даних та реалізації бізнес-логіки, доцільно обрати мову програмування C#. Вона дозволяє реалізувати архітектуру за шаблоном MVC, що забезпечує зручне розділення логіки, подання та доступу до даних. Завдяки платформі ASP.NET Core, C# забезпечує високу продуктивність, кросплатформність, широкі можливості масштабування, надійний захист, а також повну підтримку сучасних вебтехнологій. Крім того, C# чудово інтегрується з базами даних, сторонніми

API, хмарними платформами та має розвинену екосистему інструментів розробки. Таким чином, мова C# у поєднанні з C/C++ на апаратному рівні становить ідеальну основу для побудови надійної, сучасної та функціональної інформаційно-аналітичної системи моніторингу мікроклімату.

2.5 Дослідження протоколів зв'язку

Було вибрано комбінований протокол зв'язку, а саме HTTP та Server-Sent Events. Вони забезпечують передачу вимірювань від периферійних вузлів до центрального сервера, де здійснюються їх збереження, аналіз і візуалізація.

Також було з урахуванням обмежених ресурсів IoT-пристроїв та особливостей трафіку, було досліджено, які найчастіше застосовуються спеціалізовані протоколи: HTTP, MQTT, CoAP, WebSocket, Server-Sent Events, кожен з яких має власний діапазон застосувань, переваги та обмеження. У подальших підпунктах подано стислий огляд і порівняння найпоширеніших із них з метою обґрунтування вибору оптимального поєднання протоколів для взаємодії мікроконтролера, серверної частини та вебінтерфейсу у розроблюваній системі моніторингу.

2.5.1 Огляд HTTP

Одним із найпоширеніших і базових протоколів зв'язку в архітектурі «клієнт-сервер» є HTTP. Цей протокол використовується в переважній більшості вебзастосунків та IoT-рішень завдяки своїй простоті, відкритості та сумісності з існуючими інтернет-інфраструктурами. У контексті систем моніторингу мікроклімату HTTP дозволяє мікроконтролеру надсилати дані на сервер у вигляді HTTP-запитів, зокрема методом POST, а також отримувати конфігурації або команди через GET-запити [42].

Основною перевагою HTTP є його простота реалізації: практично кожен сучасний мікроконтролер з підтримкою TCP/IP-стека може надсилати дані через HTTP без потреби у складному налаштуванні. Протокол добре підтримується численними мовами програмування, а також легко інтегрується з вебсервером

або RESTful-API, що дозволяє будувати масштабовану серверну частину системи [42].

Слід також відзначити, що HTTP є орієнтованим на запити, тобто мікроконтролер ініціює комунікацію лише за потреби - наприклад, через певний часовий інтервал або після фіксації змін параметрів мікроклімату. У системах, де важлива періодичність та стабільність передачі даних, це дозволяє точно контролювати мережеву активність та зменшити енергоспоживання.

З метою кількісної оцінки доцільності використання HTTP розглянемо реальну конфігурацію розроблюваної системи, у якій контролер ESP32 надсилає на сервер пакет вимірювань із періодом:

$$T_{\text{post}} = 1\text{c (POST_PERIOD_MS = 1000)},$$

де T_{post} – інтервал між POST-запитами.

Тоді кількість $N_{\text{hour}} = \text{HTTP-запитів від одного вузла становить:}$

$$N_{\text{hour}} = \frac{3600}{T_{\text{post}}} = \frac{3600}{1} = \text{запитів/год},$$

$$N_{\text{day}} = 24 \cdot N_{\text{hour}} = 24 \cdot 3600 = 86400 \text{ запитів/добу},$$

де N_{day} – кількість запитів надісланих за один день;

N_{hour} – кількість запитів надісланих за одну годину.

$$\frac{N_{\text{day}}}{N_{\text{save}}}, \quad (2.1)$$

де N_{save} – кількість збережених запитів в базу даних.

Оскільки ми зберігаємо в базу даних кожен $N_{\text{save}} = 5$, запитів, то за (2.1):

$$\frac{86400}{5} = 17280 \text{ запитів/збережено.}$$

Середній розмір одного запиту можна подати як:

$$L_{\text{total}} = L_{\text{data}} + L_{\text{http}}, \quad (2.2)$$

де L_{total} – сумарний обсяг переданих даних;

L_{data} – корисне навантаження JSON;

L_{http} – службові заголовки HTTP-протоколу.

Кожне з цих значень має різну кількість байт, а саме:

– $L_{data} \approx 250$ байт;

– $L_{http} \approx 650$ байт;

– $L_{total} \approx 900$ байт.

Тоді обсяг мережевого трафіку за годину одного вузла за формулою (2.3) дорівнює:

$$V_{hour} = L_{total} \cdot N_{hour} = 3600 \cdot 900 = 3,24 \cdot 10^6 \text{ байт} = 3,1 \text{ МБ/год}, \quad (2.3)$$

де V_{hour} – кількість МБ/год.

Тоді обсяг мережевого трафіку за день від одного вузла за формулою (2.4) дорівнює:

$$V_{day} = N_{day} \cdot L_{total} = 17280 \cdot 900 = 7,78 \cdot 10^6 \text{ байт} = 74 \text{ МБ/добу}, \quad (2.4)$$

де V_{day} – кількість МБ/день.

Кількість збережених запитів за добу за формулою (2.5), при умові, що ми зберігаємо кожен 5 запит:

$$V_{save} = L_{total} \cdot N_{day} = 900 \cdot 17280 = \frac{1,5552 \cdot 10^6 \text{ байт}}{5} = 14,8 \text{ МБ/добу}. \quad (2.5)$$

Отримані значення показують, що за періоду надсилання 1 с обсяг трафіку для одного вузла є прийнятним для локальної мережі та типового інтернет-каналу. Це підтверджує практичну придатність HTTP для періодичної передачі вимірювань у розроблюваній системі без перевантаження мережевої інфраструктури, а також без перевантаження бази даних. Для декількох вузлів трафік масштабується лінійно, що дає змогу оцінити граничну кількість

пристроїв у мережі. За потреби збільшення кількості вузлів або розширення номенклатури параметрів інтервал передавання даних може бути адаптивно змінений, а пакети вимірювань – агреговані, що забезпечує додатковий резерв для масштабування системи.

Таким чином, HTTP є доцільним вибором для побудови інфраструктури збору та обміну даними між мікроконтролерами і сервером у розроблюваній системі, а виконані кількісні оцінки мережевого навантаження підтверджують його відповідність експлуатаційним вимогам.

2.5.2 Огляд MQTT

Message Queuing Telemetry Transport – це легкий протокол обміну повідомленнями, спеціально розроблений для систем з обмеженими ресурсами та нестабільним або ненадійним з'єднанням. Завдяки низьким накладним витратам та публікаційно-підписній моделі, MQTT став одним із найпопулярніших стандартів у сфері Інтернету речей, зокрема для побудови розподілених систем моніторингу мікроклімату. У такій моделі пристрої або публікують повідомлення у певну тему, або підписуються на неї й отримують оновлення від брокера, не знаючи IP-адрес інших вузлів мережі [43].

MQTT підтримує різні рівні якості обслуговування:

- QoS 0 – повідомлення доставляється щонайбільше один раз;
- QoS 1 – повідомлення доставляється щонайменше один раз;
- QoS 2 – повідомлення доставляється тільки один раз.

Це дає змогу балансувати між надійністю доставки, затримками та мережевим навантаженням відповідно до потреб застосування. У системі моніторингу мікроклімату, де важливе своєчасне надходження даних без дублювання, перевагу можна надати QoS 1 або QoS 2 для критичних параметрів.

З метою кількісної оцінки мережевих витрат MQTT розглянемо той самий сценарій, що й для HTTP: контролер ESP32 публікує пакет вимірювань з періодом:

$$T_{\text{pub}} = 1\text{c} \text{ (POST_PERIOD_MS = 1000)},$$

де T_{pub} – інтервал між публікаціями MQTT-повідомлень.

Тоді кількість повідомлень від одного вузла за годину:

$$N_{\text{hour}} = \frac{3600}{T_{\text{post}}} = \frac{3600}{1} = 3600 \text{ запитів/год,}$$

$$N_{\text{day}} = 24 \cdot N_{\text{hour}} = 24 \cdot 3600 = 86400 \text{ запитів/добу,}$$

де N_{hour} – кількість повідомлень за одну годину;

N_{day} – кількість повідомлень за день.

При умові що кожен 5 запит зберігається в базу даних тоді $N_{\text{save}} = 5$, згідно формули (2.6), як і в випадку з HTTP-POST в підпункті 2.5.1, $N_{\text{save}} = 17280$ записі/добу:

$$L_{\text{total}} = L_{\text{data}} + L_{\text{mqtt}}, \quad (2.6)$$

де L_{total} – сумарний обсяг переданих даних;

L_{data} – корисне навантаження JSON;

L_{mqtt} – службові поля MQTT-протоколу.

Кожне з цих значень має різну кількість байт, а саме:

– $L_{\text{data}} \approx 250$ байт,

– $L_{\text{mqtt}} \approx 300$ байт,

– $L_{\text{total}} = 550$ байт.

Тоді обсяг мережевого трафіку згідно формули (2.7) від одного вузла за годину дорівнює:

$$V_{\text{hour}} = L_{\text{total}} \cdot N_{\text{hour}} = 550 \cdot 3600 = 1,98 \cdot 10^6 \text{ байт} = 1,9 \text{ МБ/год,} \quad (2.7)$$

де V_{hour} – кількість МБ/год.

Тоді обсяг мережевого трафіку згідно формули (2.8) від одного вузла за день дорівнює:

$$V_{\text{day}} = N_{\text{day}} \cdot L_{\text{total}} = 17280 \cdot 550 = 9,50 \cdot 10^6 \text{ байт} = 45,6 \text{ МБ/добу,} \quad (2.8)$$

де V_{day} – кількість МБ/день.

Кількість збережених запитів за добу, при умові що ми зберігаємо кожен 5 запит:

$$V_{\text{save}} = L_{\text{total}} \cdot N_{\text{day}} = 550 \cdot 17280 = \frac{9,50 \cdot 10^6 \text{ байт}}{5} = 9,12 \text{ МБ/добу.}$$

На основі виконаних розрахунків можна зробити висновок, що використання MQTT дає відчутний вигоду за обсягом мережевого трафіку порівняно з HTTP. За однакового періоду передавання $T=1\text{с}$. і тієї самої кількості повідомлень N_{hour} та N_{day} сумарна довжина одного пакета для MQTT становить близько 550 байт проти ~900 байт для HTTP, тобто службове навантаження зменшується орієнтовно у 1,6 рази. Відповідно, годинний і добовий трафік від одного вузла при застосуванні MQTT є меншим.

Разом з тим зменшення трафіку досягається ціною ускладнення архітектури: для роботи MQTT необхідно розгорнути окремий брокер, підтримувати його конфігурацію та забезпечувати надійність цього центрального вузла. Для розроблюваної системи моніторингу мікроклімату з одним вимірювальним вузлом і веб-інтерфейсом на базі HTTP/SSE отримані оцінки показують, що навіть у випадку класичного HTTP обсяг трафіку й навантаження на базу даних залишаються в допустимих межах. Тому MQTT доцільно розглядати як перспективний варіант для сценаріїв подальшого масштабування системи, тоді як на поточному етапі використання HTTP як основного транспортного протоколу є технічно виправданим і достатнім.

2.5.3 Огляд WebSocket

WebSocket – це сучасний мережевий протокол, який забезпечує двостороннє з'єднання між клієнтом і сервером у межах одного TCP-з'єднання. Його поява стала відповіддю на обмеження класичного HTTP-протоколу, який є однонапрямленим і передбачає постійне ініціювання запитів клієнтом для отримання оновлень. WebSocket дозволяє обоє сторонам надсилати дані у

будь-який момент, що робить його ідеальним рішенням для застосунків реального часу [44].

У систем моніторингу мікроклімату WebSocket забезпечує мінімальну затримку при передаванні інформації від мікроконтролера або сервера до вебінтерфейсу користувача. Наприклад, коли ESP32 надсилає дані на сервер, останній може миттєво транслювати їх користувачу без потреби у періодичному опитуванні. Це суттєво знижує навантаження на мережу та сервер, особливо у порівнянні з традиційним AJAX-полінгом.

Підключення WebSocket починається з рукостискання через HTTP-запит, після чого відбувається оновлення протоколу до WebSocket і встановлюється постійне з'єднання. Комунікація після цього відбувається у вигляді легких фреймів, що значно скорочує обсяг службової інформації у порівнянні з HTTP-запитами [44].

Основною перевагою WebSocket у системах моніторингу є висока інтерактивність і реактивність. У користувацькому інтерфейсі це дозволяє реалізовувати оновлення графіків, показників, тривог і станів обладнання в режимі реального часу без перезавантаження сторінки чи повторних запитів.

Недоліком WebSocket є дещо більша складність реалізації як на клієнтській, так і на серверній стороні, а також вища вимога до постійного утримання з'єднання, що може бути критичним для мікроконтролерів з обмеженими ресурсами або енергоавтономією. Для повноцінного функціонування потрібна підтримка WebSocket-серверу, наприклад, з використанням бібліотек SignalR, ws, Socket.IO або FastAPI WebSocket.

У розробці інформаційно-аналітичних систем, які мають вебінтерфейс з динамічним оновленням, WebSocket є одним із найбільш ефективних протоколів зв'язку. Він забезпечує постійну синхронізацію станів між сервером і користувачем, що критично важливо у випадках, коли необхідно оперативно реагувати на зміну мікрокліматичних параметрів або перевищення допустимих норм.

2.5.4 Огляд Server-Sent Events

Server-Sent Events – це протокол односторонньої передачі даних від сервера до клієнта, що базується на стандарті HTTP. SSE дає змогу серверу надсилати оновлення в браузер у режимі реального часу без потреби в періодичному опитуванні або встановленні складного двостороннього з'єднання, як це відбувається з WebSocket. Такий підхід є надзвичайно ефективним для систем, де ініціатором передавання даних завжди є сервер, у випадках моніторингу змін стану сенсорів [45].

Протокол SSE працює через стандартне HTTP-з'єднання, використовуючи MIME-тип `text/event-stream`. Браузер або інший клієнт ініціює запит до сервера, після чого сервер підтримує відкритий канал і періодично надсилає нові дані у вигляді подій. Це робить SSE природною та простою технологією для реалізації стрімінгу даних без використання сторонніх бібліотек чи спеціального середовища розробки.

У контексті інформаційно-аналітичних систем моніторингу мікроклімату SSE може використовуватися для оновлення вебінтерфейсу, наприклад, графіків, індикаторів, табличних звітів у реальному часі. Завдяки своїй енергоефективності, SSE особливо добре підходить для мікроконтролерів із обмеженими ресурсами, таких як ESP32, де важливо зменшити обсяг службового трафіку та спростити логіку взаємодії з сервером.

До основних переваг SSE можна віднести простоту реалізації, використання стандартного HTTP-порту, автоматичне повторне з'єднання у разі втрати зв'язку та можливість надсилання іменованих подій. SSE також підтримується всім сучасними браузерами, що спрощує інтеграцію у вебсистеми. Крім того, оскільки зв'язок ініціюється клієнтом, його можна безперешкодно реалізувати на серверах, які обслуговують одночасно багато підключень, без істотного навантаження [45].

Проте SSE має і певні обмеження. Зокрема, він працює лише в напрямку сервер-клієнт, тому не підходить для повноцінної двосторонньої комунікації, як

це можливо у WebSocket. Також він не підтримується у старіших браузерах Internet Explorer та має обмежену підтримку на деяких мобільних пристроях.

Загалом, SSE є оптимальним вибором для вебзастосунків, які потребують регулярного оновлення даних у реальному часі без потреби зворотного зв'язку від клієнта. У системі моніторингу мікроклімату цей протокол забезпечує надійну, просту й масштабовану модель для передавання екологічних показників з серверної частини до користувацького інтерфейсу.

2.5.5 Обґрунтування вибору протоколу зв'язку

Вибір протоколу зв'язку між мікроконтролером та сервером для побудови інформаційно-аналітичної системи моніторингу мікроклімату було здійснено на основі детального аналізу характеристик, можливостей і обмежень кожного з розглянутих рішень. У процесі дослідження були вивчені такі протоколи, як HTTP, MQTT, WebSocket та Server-Sent Events, кожен із яких має свої переваги у специфічних сценаріях використання. Однак, з урахуванням вимог до стабільної та ефективної передачі даних у режимі реального часу, пріоритетів у простоті реалізації, сумісності з обраними технологіями розробки інтерфейсу та підтримки з боку браузерів, найбільш доцільним було визнано застосування технології Server-Sent Events.

SSE забезпечує односторонній стрімінг даних від сервера до клієнта з використанням звичайного HTTP-з'єднання, що значно спрощує реалізацію і підтримку системи. Цей протокол ідеально підходить для задач постійного оновлення показників із сенсорів, оскільки дозволяє надсилати повідомлення в реальному часі без необхідності встановлення повноцінного двостороннього з'єднання, як у випадку з WebSocket. Завдяки підтримці автоматичного відновлення з'єднання після його розриву та механізму повторної передачі останніх даних, SSE забезпечує високу надійність у нестабільних мережах, що є критично важливим для безперервного моніторингу.

Вибір SSE зумовлений високою сумісністю з технологічним стеком, використаним у проєкті, зокрема ASP.NET Core та C#, які дозволяють без особливих труднощів реалізувати серверну підтримку даного протоколу. З боку

клієнта SSE підтримується усіма сучасними браузерами без потреби у додаткових бібліотеках чи складному налаштуванні. Це дає змогу спростити як розгортання системи, так і її подальшу експлуатацію.

Тому можна сказати, що протокол Server-Sent Events був обраний як основний механізм комунікації між сервером і клієнтською частиною інформаційно-аналітичної системи моніторингу мікроклімату завдяки його стабільності, простоті інтеграції, низькому споживанню ресурсів і здатності забезпечити швидке та надійне оновлення даних у реальному часі. Це рішення дозволяє реалізувати ефективну, зручну та масштабовану архітектуру передачі екологічних параметрів з пристроїв до інтерфейсу користувача.

2.6 Реалізація тестового стенду системи

Для реалізації тестового стенду інформаційно-аналітичної системи мікроклімату були використані такі компоненти:

- плата розробника DOIT DEVKIT V1 на базі ESP32;
- цифровий датчик температури та вологості DHT22;
- барометричний датчик атмосферного тиску BMP280;
- датчик освітленості BH1750;
- лазерний оптичний датчик пилу PMS5003 для вимірювання PM1,0, PM2,5, PM10;
- макетна плата;
- джампери;
- резистори;
- перехідники;
- газовий датчик MQ-135 для виявлення ЛОС, аміаку, бензолу, чадного газу;
- газовий датчик MQ-2 для виявлення диму та горючих газів;
- газовий датчик MQ-7 для вимірювання концентрації чадного газу.

Всі датчики та модулі підключені до плати мікроконтролера ESP32 наступним чином:

- 1) комбінований датчик температури та вологості DHT22:
 - вхід GND до виходу GND контролера;
 - вхід VCC до виходу 3v3 контролера;
 - вихід out до входу D4 контролера;
- 2) датчик атмосферного тиску BMP280:
 - вхід GND до виходу GND контролера;
 - вхід VCC до виходу 3v3 контролера;
 - вихід SCL до входу D22 (GPIO22) контролера;
 - вихід SDA до входу D21 (GPIO21) контролера;
- 3) датчик освітленості BH1750:
 - вхід GND до виходу GND контролера;
 - вхід VCC до виходу 3v3 контролера;
 - вихід SCL до входу D22 (GPIO22) контролера;
 - вихід SDA до входу D21 (GPIO21) контролера;
- 4) датчик твердих часток PMS5003 (UART2):
 - вихід TX PMS5003 до входу RX2 (GPIO16);
 - вхід GND до виходу GND контролера;
 - вхід VCC до виходу 5 V;
- 5) датчик якості повітря MQ-135:
 - вихід АО до входу A0 (GPIO36) контролера;
 - вхід GND до виходу GND контролера;
 - вхід VCC до виходу 5 V;
- б) датчик диму та горючих газів MQ-2:
 - вихід АО до входу A2 (GPIO34) контролера;
 - вхід GND до виходу GND контролера;
 - вхід VCC до виходу 5 V;
- 7) датчик чадного газу MQ-7:

- вихід АО до входу А3 (GPIO39) контролера;
- вхід GND до виходу GND контролера;
- вхід VCC до виходу 5 V.

Підключено живлення ESP32 через USB Type-C кабель, а датчики живляться від esp32 від виходів 3,3v та 5 v.

Після визначення всіх компонентів було побудовано функціональну схему, яку зображено на рисунку 2.17.

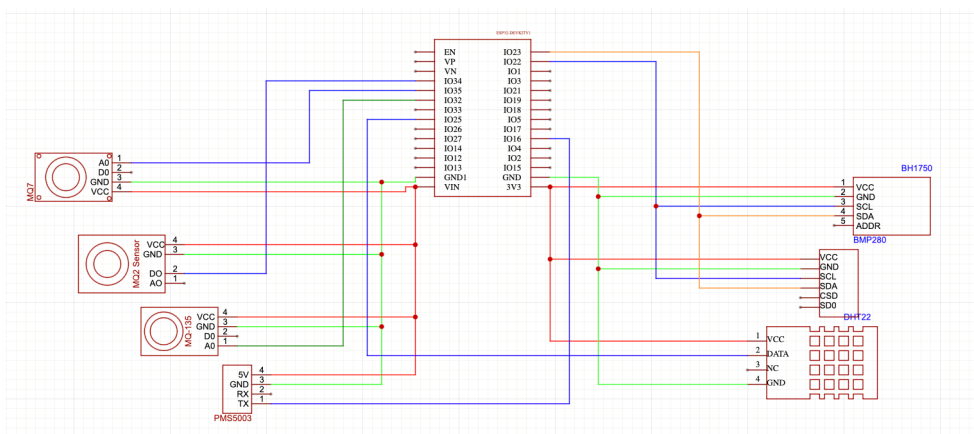


Рисунок 2.17 – Функціональна схема системи моніторингу мікроклімату

Після того, як була побудована функціональна система моніторингу мікроклімату на макетній платі було підключено всі компоненти відповідно до схеми, тестовий стенд зображено на рисунку 2.18.

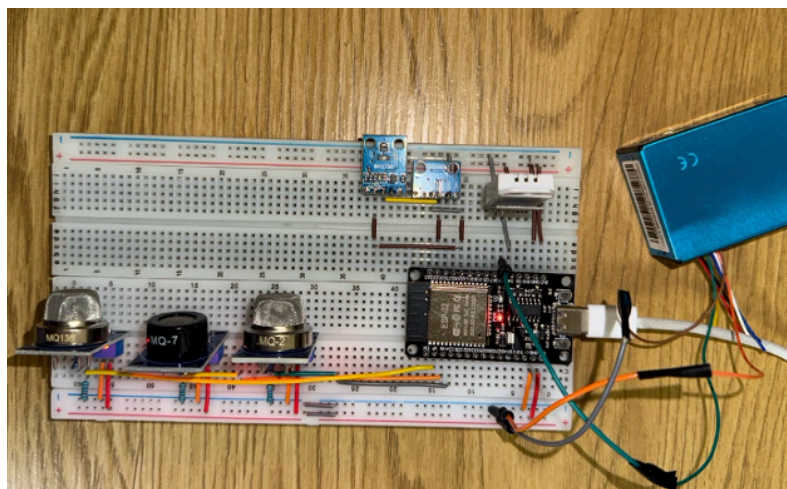


Рисунок 2.18 – Тестовий стенд системи моніторингу мікроклімату

Для того, щоб все запрацювало, потрібно прошити мікроконтролер ESP32 відповідною прошивкою, яка містить програмний код для ініціалізації всіх підключених датчиків, зчитування показників, їх обробки та подальшої передачі даних до вебсервера. Тому далі будуть розглянуті способи прошивки мікроконтролера для подальшого дослідження.

2.6.1 Програмне забезпечення розробленої IoT-системи

Для того щоб вся система запрацювала належним чином, необхідно прошити мікроконтролер ESP32. Для цього використовувалось ArduinoIDE. Саме програмне забезпечення виконує роль «мозку» IoT-системи, забезпечуючи логіку її функціонування та взаємодії з середовищем.

Мова програмування C/C++ з використанням Arduino IDE – найбільш популярний варіант завдяки простоті, великій кількості бібліотек та активній спільноті. За допомогою бібліотек, таких як DHT.h, Adafruit_BMP280.h, Wire.h, WiFi.h тощо, реалізується підключення до сенсорів і передача даних через мережу.

Платформа PlatformIO – розширена система розробки, що дозволяє ефективно працювати з великими проєктами, має підтримку автозбирання прошивок, інтеграцію з Git, гнучке налаштування середовищ та зручну роботу з бібліотеками.

MicroPython – підходить для простих та навчальних проєктів, дозволяє писати код на мові Python, що особливо зручно для тих, хто не має глибоких знань C++.

Espressif IoT Development Framework – офіційний фреймворк від виробника ESP32. Забезпечує доступ до низькорівневих можливостей контролера, надає гнучкість та високу продуктивність. Рекомендований для промислових та комерційних розробок.

У межах цієї роботи для реалізації прошивки була використана среда Arduino IDE як найбільш доступна, стабільна та зручна для розгортання прототипу. Завдяки відкритому коду, великій кількості прикладів та доступних

бібліотек, розробка функціональної частини програмного забезпечення здійснювалась швидко й ефективно.

2.6.2 Інформаційне забезпечення системи

Інформаційне забезпечення є фундаментальною складовою інформаційно-аналітичної системи моніторингу мікроклімату, оскільки визначає можливості зберігання, структурування, доступу та обробки даних, що надходять від сенсорних модулів у режимі реального часу. Для системи, що оперує великими обсягами телеметрії IoT-пристроїв, важливо обрати таку систему керування базами даних, яка забезпечить високу швидкість запису, стабільність, цілісність даних, можливість масштабування та зручність інтеграції з веб-сервером.

Сучасні реляційні СУБД забезпечують строгість даних, підтримку транзакцій та ефективну роботу зі структурованою інформацією, що робить їх особливо придатними для систем моніторингу. У цьому підрозділі розглянемо дві найпоширеніші СКБД – MySQL та PostgreSQL – і проведемо їх детальне порівняння з точки зору застосування у розробленій системі [46].

MySQL є однією з найбільш розповсюджених реляційних СУБД, яка історично набула популярності завдяки простоті розгортання, надійності та високій швидкості виконання операцій читання. Її нерідко застосовують у веб-проєктах, де переважають операції SELECT, а структура даних є відносно стабільною та не потребує складних SQL-конструкцій [46].

Однією з ключових переваг MySQL є її продуктивність у випадках, коли система працює з великою кількістю запитів на читання. Завдяки оптимізованому механізму індексування та ефективному кешуванню ця СУБД добре масштабується у середовищах, де домінують операції вибірки.

Однак при роботі з інтенсивними вставками даних, що характерно для IoT-платформ, MySQL демонструє певні обмеження. Зокрема, використання рушія InnoDB хоча й забезпечує транзакційність, але у високонавантажених сценаріях INSERT-операції можуть поступатися за швидкістю PostgreSQL. Додатковим недоліком є частково обмежена підтримка складних

SQL-конструкцій, відсутність повноцінної реалізації віконних функцій у старіших версіях та більш скромні можливості роботи з JSON у порівнянні з конкурентами [46].

PostgreSQL є об'єктно-реляційною СУБД, яка відома своєю потужністю, стабільністю та відповідністю стандартам SQL. На відміну від MySQL, PostgreSQL орієнтована не лише на високу продуктивність, але й на підтримку складної бізнес-логіки, гнучких типів даних та розширених можливостей аналітики [47].

Порівнюючи MySQL та PostgreSQL, можна зазначити, що обидві СУБД забезпечують якісну роботу з реляційними даними, однак орієнтовані на різні типи застосувань. MySQL більше підходить для систем, де домінує читання та де структура даних стабільна й не потребує складних обчислень на рівні СУБД. PostgreSQL, своєю чергою, пропонує значно ширші можливості для аналітики, гнучкі типи даних та розширені інструменти оптимізації запитів.

У випадку інформаційно-аналітичної системи моніторингу мікроклімату PostgreSQL демонструє явну перевагу. Система оперує показниками, що надходять у режимі реального часу та формують історичні набори даних, які потребують подальшої статистичної обробки, фільтрування, групування за часовими інтервалами та побудови графіків. PostgreSQL краще справляється з такими задачами завдяки підтримці часових рядів, ефективній роботі з великими таблицями, коректній транзакційності та стабільній продуктивності під високими навантаженнями при вставці даних.

Ця СУБД забезпечує найкраще поєднання продуктивності, можливостей аналітики, гнучкості роботи з різними типами даних та надійності. PostgreSQL дозволяє ефективно реалізувати як базове зберігання сенсорних показників, так і розширені можливості для подальшого аналізу, що є важливим для повноцінного функціонування інформаційно-аналітичної системи моніторингу мікроклімату.

РОЗДІЛ 3

РОЗРОБКА ІНФОРМАЦІЙНО-АНАЛІТИЧНОЇ СИСТЕМИ

3.1 Структура інформаційно-аналітичної системи

Проект розроблявся в програмному середовищі Visual Studio Code із використанням платформи ASP.NET Core MVC. Така структура дає змогу чітко розмежовувати рівні представлення, бізнес-логіки та доступу до даних, спрощуючи подальший супровід та масштабування системи моніторингу мікроклімату. Структуру інформаційно-аналітичної системи зображено на рисунку 3.1

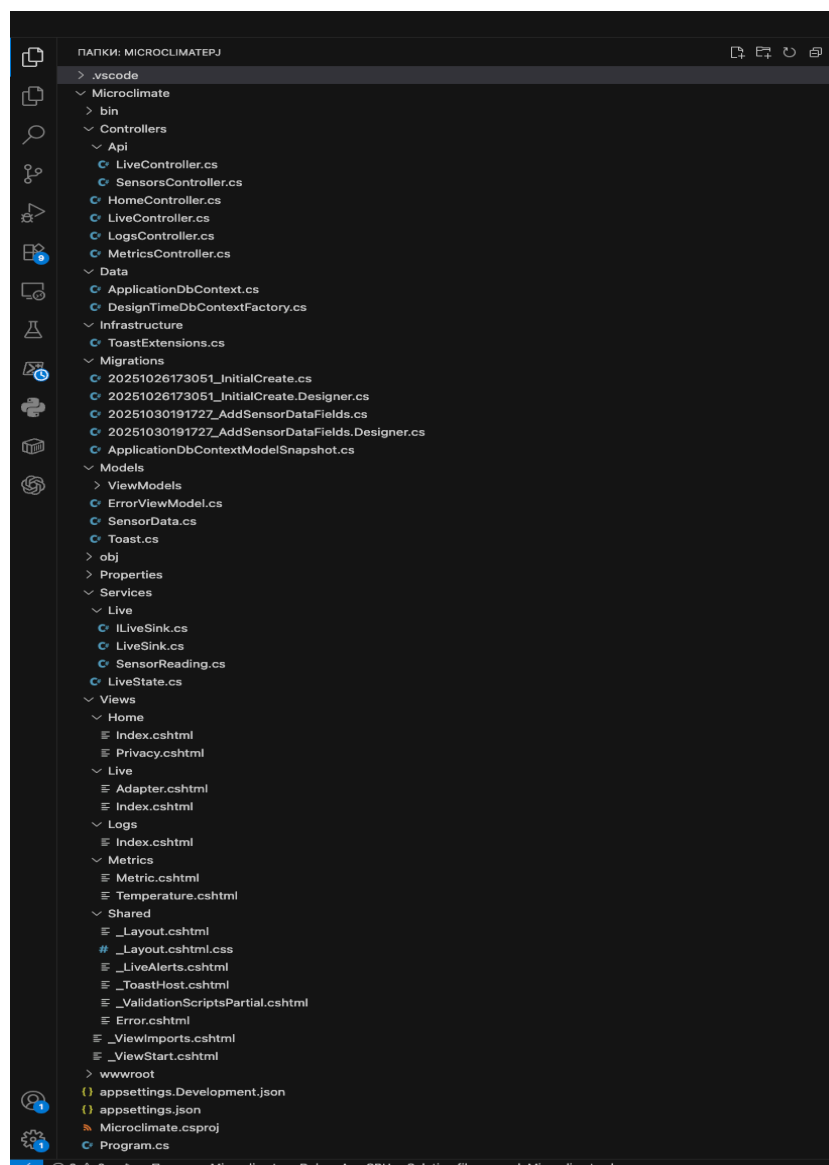


Рисунок 3.1 – Структура інформаційно-аналітичної системи

Основна директорія Microclimate містить усі вихідні компоненти застосунку: контролери, моделі, представлення, сервіси та конфігураційні файли. Файл Microclimate.csproj визначає параметри збірки та підключені бібліотеки, а Program.cs є вхідною точкою, де налаштовуються служби, підключення до бази даних, маршрутизація та запуск вебсерверу.

Контролери реалізують бізнес-логіку та посередницьку роль між інтерфейсом користувача, сервісами й шаром даних. У проєкті виділено дві групи: API-контролери для інтеграції із зовнішніми пристроями та застосунками, і MVC-контролери для формування веб-інтерфейсу.

До API-групи належать Api/LiveController.cs та Api/SensorsController.cs. Перший приймає POST-запити від ESP32 на маршрут /api/live, десеріалізує JSON із показниками сенсорів, зберігає їх у БД через ApplicationDbContext і передає останні значення у сервіс даних, а також організовує потік Server-Sent Events для онлайн-оновлення інтерфейсу. Api/SensorsController.cs надає ендпоінти для отримання поточних та історичних вимірювань у форматі JSON, що використовується мобільними клієнтами та зовнішніми сервісами.

MVC-групу утворюють HomeController.cs, LiveController.cs, LogsController.cs та MetricsController.cs. HomeController обслуговує стартові та загальні сторінки. LiveController відображає режим реального часу, отримуючи актуальні дані з сервісу «живих» показників і SSE-потіку. LogsController працює з історичними записами, забезпечуючи фільтрацію та перегляд логів. MetricsController формує аналітичні сторінки, де обчислюються статистичні характеристики та готуються дані для графіків.

Моделі в структурі проєкту описують дані, з якими працює система. SensorData.cs представляє запис вимірювань і використовується для збереження в базі та подальшої обробки. ErrorViewModel.cs застосовується для відображення сторінок помилок, Toast.cs – для передачі повідомлень про події й попередження в інтерфейс. Окрема папка ViewModels містить моделі представлення, які об'єднують кілька сутностей і додаткові поля, необхідні для зручного відображення журналів та інших даних на конкретних сторінках.

Toast.cs модель призначена для передачі повідомлень про підвищення показників та передає їх до інтерфейсу користувача.

3.2 Інформаційне забезпечення системи

У структурі проєкту окремий рівень становлять компоненти доступу до даних, які інкапсують роботу з базою даних і конфігурацію ORM-фреймворку Entity Framework Core. Ці компоненти зосереджені в каталозі Data і відповідають за створення контексту даних, опис наборів сутностей та налаштування підключення до СУБД як під час роботи програми, так і на етапі проєктування.

Файл ApplicationDbContext.cs містить клас контексту даних, який успадковується від DbContext і описує, з якими таблицями та сутностями працює застосунок. Оголошується набір записів журналу вимірювань сенсорів та задаються правила їх конфігурації. Зокрема, для сутності вимірювань встановлюється індекс за полем часової позначки, що прискорює виконання запитів, пов'язаних із пошуком та сортуванням даних за часом.

Файл DesignTimeDbContextFactory.cs містить фабрику контексту даних, яка використовується на етапі розробки. Цей клас створює екземпляр контексту ApplicationDbContext для інструментів Entity Framework Core під час виконання команд міграцій. У межах фабрики налаштовується підключення до СУБД PostgreSQL та параметри контексту, що дозволяє коректно генерувати та застосовувати зміни схеми бази даних без запуску основного веб-застосунку.

Підключення до бази даних PostgreSQL відбувається через рядок підключення, заданий у файлі конфігурації appsettings.json у секції ConnectionStrings. У цьому файлі описується параметр DefaultConnection, який містить усі необхідні налаштування для доступу до СУБД. Фрагмент коду зображено на рисунку 3.2.

```
Microclimate > {} appsettings.json > ...
1
2   "Logging": {
3     "LogLevel": {
4       "Default": "Information",
5       "Microsoft.AspNetCore": "Warning"
6     }
7   },
8   "ConnectionStrings": {
9     "DefaultConnection": "Host=localhost;Port=5432;Database=sensors;Username=postgres;Password=1111"
10  },
11  "AllowedHosts": "*"
12 }
```

Рисунок 3.2 – Рядок підключення до бази даних

База даних була побудована за допомогою міграцій ORM-фреймворку Entity Framework Core. На основі опису моделей у кодї було сформовано початкову міграцію, яка створила структуру таблиць, їх поля, типи даних та необхідні обмеження.

Узгодженість між об'єктною моделлю застосунку та фізичною структурою бази даних PostgreSQL, спростив розгортання системи на різних середовищах.

3.3 Програмне забезпечення інформаційно-аналітичної системи

Програмне забезпечення відіграє ключову роль у функціонуванні інформаційно-аналітичної системи моніторингу мікроклімату, забезпечуючи приймання, обробку, зберігання та візуалізацію даних, отриманих від мікроконтролера з підключеними датчиками. На цьому етапі реалізується логіка взаємодії між вебінтерфейсом, серверною частиною та системою керування базами даних, а також механізми забезпечення надійності, масштабованості та зручності користування.

3.3.1 Програмне забезпечення мікроконтролера

Програмне забезпечення мікроконтролера ESP32 розроблено у середовищі Arduino IDE. Програма має модульну архітектуру: програмний код поділено на окремі логічні фрагменти відповідно до їх функціонального призначення, що підвищує структурованість, читабельність та зрозумілість реалізації. Схема взаємозв'язків модулів прошивки зображено на рисунку 3.3.

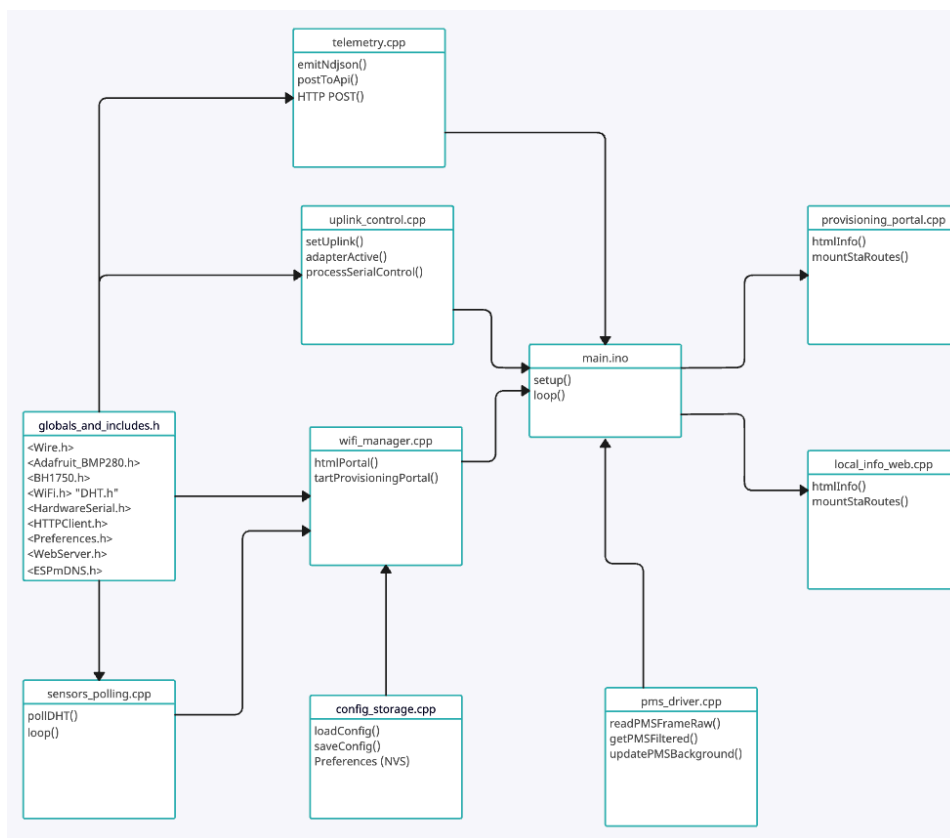


Рисунок 3.3 – Схема взаємозв'язків модулів прошивки системи моніторингу мікроклімату

Розглянемо функціонал кожного модуля прошивки окремо. Модуль `globals_and_includes.h` містить підключення всіх необхідних бібліотек, оголошення глобальних змінних та допоміжних функцій, спільних для всієї прошивки. У ньому задаються інтервали опитування сенсорів `POST_PERIOD_MS`, `PMS_POLL_PERIOD_MS`, `DHT_PERIOD_MS`, створюються об'єкти датчиків DHT22, BMP280, BH1750, PMS5003, визначаються піни газових сенсорів MQ-135, MQ-2, MQ-7, оголошуються мережеві параметри `wifiSsid`, `wifiPass`, `apiUrl`, `apiToken`, об'єкт `Preferences` для NVS-пам'яті та вебсервер `WebServer` `cfg(80)`. Тут же описано перерахування Uplink для вибору каналу передавання даних та реалізовано допоміжні функції `num()` і `emaInt()` для форматування чисел і згладжування шумних вимірювань.

Модуль `config_storage.cpp` реалізує збереження конфігурації мережі та API. Функції `loadConfig()` і `saveConfig()` відповідають за читання та запис SSID, пароля, URL API та токена авторизації у NVS-пам'ять мікроконтролера,

забезпечуючи збереження налаштувань між перезавантаженнями. Ці функції викликаються з `main.ino` під час запуску системи та з порталу налаштування при збереженні нових параметрів.

Модулі `wifi_manager.cpp` та `provisioning_portal.cpp` відповідають за підключення до Wi-Fi та роботу вебпорталу налаштування. Функція `connectSTA()` виконує підключення ESP32 до заданої мережі в режимі станції WIFI_STA і повідомляє про результат через послідовний порт. Функція `startProvisioningPortal()` вмикає режим точки доступу, піднімає SoftAP, запускає вбудований HTTP-сервер, реєструє маршрути і через `htmlPortal()` надає користувачу вебформу для введення SSID, пароля та параметрів API з подальшим збереженням їх у NVS.

Модуль `local_info_web.cpp` реалізує локальний вебінтерфейс для перегляду поточних показників сенсорів у STA-режимі. Функція `htmlInfo()` формує HTML-сторінку з відображенням значень температури й вологості DHT22, тиску BMP280, освітленості BH1750, концентрації твердих частинок PMS5003 і показників газових датчиків MQ. Функція `mountStaRoutes()` реєструє маршрути / і /info, що використовуються для діагностики та налагодження роботи вузла через браузер.

Модуль `pms_driver.cpp` інкапсулює роботу з датчиком пилу PMS5003. Функції `readPMSFrameRaw()` і `getPMSFiltered()` відповідають за читання та валідацію кадрів з UART, фільтрацію аномальних значень і згладжування вимірювань за допомогою ЕМА. Функція `updatePMSBackground()` періодично оновлює глобальні змінні `last_pm1`, `last_pm25`, `last_pm10`, які потім використовуються вебінтерфейсом і підсистемою телеметрії.

Модуль `sensors_polling.cpp` відповідає за опитування DHT22 та газових сенсорів MQ. Функція `pollDHT()` із заданою періодичністю оновлює глобальні значення температури та вологості, а показники MQ-датчиків зчитуються з АЦП у головному циклі, додатково згладжуються й зберігаються у відповідних змінних. Ці дані використовуються як для локального відображення, так і для відправлення на сервер.

Модуль `uplink_control.cpp` реалізує керування каналом передавання даних. За допомогою перерахування `Uplink`, функцій `setUplink()` і `adapterActive()` та обробника `processSerialControl()` модуль перемикає систему між роботою через Wi-Fi та спеціальним «адаптерним» режимом по UART, інтерпретуючи службові команди, отримані з послідовного порту.

Модуль `telemetry.cpp` відповідає за формування й відправлення телеметрії. Функція `emitNdjson()` виводить виміряні параметри у форматі JSON-об'єкта в NDJSON, а функція `postToApi()` формує HTTP-запит з тими ж даними й надсилає його на `apiUrl`, враховуючи стан Wi-Fi та активність адаптера. Обидві функції періодично викликаються з `main.ino`.

Модуль `main.ino` виконує процеси всієї прошивки. У `setup()` він послідовно викликає `loadConfig()`, `connectSTA()` або `startProvisioningPortal()`, ініціалізує датчики та маршрути вебсерверу. У `loop()` модуль обробляє команди адаптера, керує станом Wi-Fi, обслуговує HTTP-запити, періодично опитує сенсори `pollDHT()`, `updatePMSBackground()` та викликає функції `emitNdjson()` і `postToApi()` для передавання даних до серверної частини інформаційно-аналітичної системи.

3.3.2 Програмна реалізація веб-інтерфейсу

Схема веб-інтерфейсу інформаційно-аналітичної системи моніторингу мікроклімату побудована за шаблоном ASP.NET Core MVC і відображена на рисунку 3.4. Вона включає окремі підсистеми для конфігурації застосунку, доступу до бази даних, обробки даних від сенсорів, реалізації HTTP-API для мікроконтролера ESP32, формування веб-інтерфейсу та обслуговування статичних ресурсів. Такий поділ на модулі дає змогу чітко розмежувати відповідальність між компонентами, спростити супровід та подальше розширення функціональності системи, а також забезпечує надійне та масштабоване опрацювання телеметрії в реальному часі.

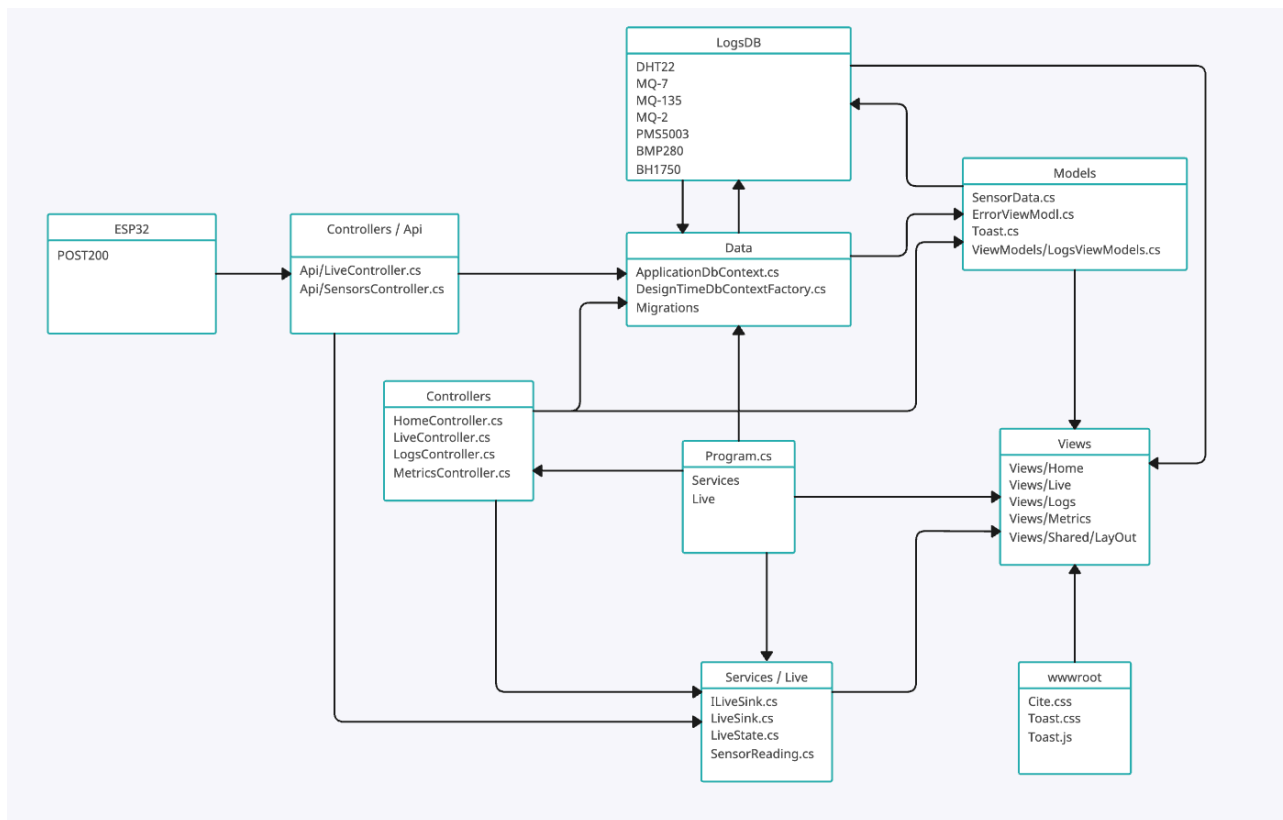


Рисунок 3.4 – Схема взаємозв'язків файлів веб-інтерфейсу користувача

Модуль Program.cs виконує роль точки входу в веб-застосунку. У ньому налаштовується конвеєр обробки запитів ASP.NET Core, реєструються служби в DI-контейнері, вмикається підтримка MVC-контролерів і представлень, а також конфігурується обслуговування статичних файлів із каталогу wwwroot. Таким чином, Program.cs пов'язує між собою всі інші підсистеми серверної частини.

Підсистема Data містить класи ApplicationDbContext.cs, DesignTimeDbContextFactory.cs та набір міграцій. Модуль ApplicationDbContext описує модель бази даних і забезпечує доступ до неї через Entity Framework Core. Міграції фіксують історію змін схеми БД та дозволяють відтворити структуру таблиць. DesignTimeDbContextFactory використовується інструментами EF Core для генерування міграцій на етапі розробки.

Модулі Models та ViewModels інкапсулюють інформаційні структури, з якими працює система. Клас SensorData.cs відображає запис у таблиці логів, ErrorViewModel.cs і Toast.cs використовуються для представлення службових повідомлень та помилок, а ViewModels/LogsViewModels.cs містить

спеціалізовані моделі для відображення журналів у зручному для користувача вигляді. Ці типи спільно використовуються як шаром доступу до даних, так і MVC-контролерами.

Підсистема `Services / Live` `ILiveSink.cs`, `LiveSink.cs`, `LiveState.cs`, `SensorReading.cs` реалізує сервіс роботи з даними від сенсорів. Інтерфейс `ILiveSink` визначає контракт запису нових вимірювань, а `LiveSink` разом із `LiveState` підтримують у пам'яті актуальний стан сенсорів і перетворюють сирі записи `SensorReading` у структуру, зручну для подальшого відображення на live-сторінці. Ця підсистема є проміжною ланкою між API-контролерами та представленнями моніторингу в реальному часі.

Група `Controllers / Api` включає контролери `Api/LiveController.cs` та `Api/SensorsController.cs`, які реалізують HTTP-інтерфейс для зовнішніх клієнтів. Саме сюди ESP32 надсилає POST-запити з вимірними параметрами. API-контролери приймають JSON-дані, виконують базову валідацію, звертаються до `ApplicationDbContext` для збереження записів у `LogsDB` та паралельно передають їх у сервіс `ILiveSink` для оновлення «живого» стану.

Група `Controllers MVC`-контролери `HomeController.cs`, `LiveController.cs`, `LogsController.cs`, `MetricsController.cs` відповідає за формування сторінок користувацького інтерфейсу. `HomeController` обслуговує головну сторінку та допоміжні представлення. `LiveController` звертається до сервісів із простору `Services.Live`, отримує поточний стан сенсорів і передає його у представлення live-моніторингу. `LogsController` та `MetricsController` працюють із `ApplicationDbContext` і моделями/в'юмоделями для побудови таблиць логів та аналітичних показників.

Підсистема `Views` об'єднує Razor-представлення, в яких формується HTML-інтерфейс для користувача. У підкаталогах `Views/Home`, `Views/Live`, `Views/Logs`, `Views/Metrics` розміщені сторінки для відповідних контролерів: головна сторінка, сторінка живих даних, журнал показників, сторінки з графіками та метриками. Каталог `Views/Shared` містить спільні частини інтерфейсу: макет `_Layout.cshtml`, часткові представлення `_LiveAlerts.cshtml`,

_ToastHost.cshtml, а також технічні файли _ViewImports.cshtml і _ViewStart.cshtml. Саме через _Layout у представлення підключаються стилі й скрипти.

Каталог wwwroot зберігає статичні ресурси клієнтської частини – каскадні таблиці стилів Site.css, Toast.css та JavaScript-файли Toast.js. Ці файли підключаються в макеті _Layout.cshtml та інших представленнях і відповідають за візуальне оформлення сторінок, а також відображення спливаючих повідомлень і динамічних елементів інтерфейсу.

Зовнішній компонент ESP32 в цій схемі виступає джерелом даних для веб-системи: він надсилає періодичні POST-запити на контролери групи Controllers / Api, після чого вимірні значення потрапляють у Services / Live та в Data / LogsDB для довгострокового зберігання й подальшої аналітики. Дані використовуються MVC-контролерами та Razor-представленнями для побудови інтерфейсу інформаційно-аналітичної системи моніторингу мікроклімату.

3.3.3 Комунікаційний рівень системи

Також було досліджено комунікаційний рівень, та способи передачі інформації, на стороні мікроконтролера усі датчики опитуються асинхронно з власними періодами, після чого їхні значення буферизуються й усереднюються. Для температури та вологості період опитування становить 2500 мс, для датчика пилу PMS5003 – близько 1200 мс, для ВН1750 освітленість оновлюється в режимі безперервних вимірювань із інтервалом порядку сотень мілісекунд, а газові сенсори MQ-135, MQ-2 та MQ-7 зчитуються з АЦП ESP32 практично в реальному часі. Отримані миттєві значення проходять через експоненційні фільтри згладжування, що зменшує шум без суттєвого збільшення затримки.

Після оновлення буферів ESP32 формує JSON-повідомлення, куди входять усі актуальні параметри мікроклімату, та надсилає його на серверний endpoint api/live методом HTTP POST. Інтервал відправлення задається константою POST_PERIOD_MS = 1000, тобто дані передаються раз на секунду, оскільки датчик температури та вологості передає дані раз в 2500 мс. то реалізація прошивки мікроконтролера передбачає це, тому він надсилає не пустий рядок, а

останні зчитані дані декілька раз. Об'єм одного JSON-пакета становить порядку 200-300 байт, тому навіть у разі передачі раз на секунду сумарне навантаження на локальну Wi-Fi-мережу не перевищує кількох кілобіт за секунду. У локальній мережі типові мережеві затримки для одного POST-запиту становлять в середньому 20-60 мс, залежно від якості сигналу Wi-Fi та завантаженості точки доступу.

На сервері ASP.NET Core отриманий HTTP-запит десеріалізується в об'єкт доменної моделі, проходить базову валідацію і передається до рівня доступу до даних. Час обробки на цьому етапі, зазвичай, становить 1-3 мс, оскільки операції обмежуються перетворенням JSON та простими перевірками. Після цього застосунок через ORM-фреймворк Entity Framework Core виконує вставку нового запису в таблицю журналу показників PostgreSQL. З урахуванням індексації за часовою міткою й оптимізованої схеми таблиць час виконання транзакції вставки одного рядка знаходиться в межах приблизно 5-15 мс. Сумарний час від моменту приходу HTTP-запиту до фактичного збереження вимірювань у базі даних не перевищує 20-30 мс і є істотно меншим за період між послідовними вимірюваннями.

Для відображення актуальних значень на клієнтській стороні використовується механізм Server-Sent Events. Веб-браузер відкриває довготривале HTTP-з'єднання з сервером і отримує потік повідомлень у текстовому форматі щоразу, коли на сервері з'являються нові вимірювання. У типовому сценарії між моментом запису показників до СКБД та їхнім надсиланням у SSE-потоці минає не більше 10-20 мс, оскільки сервер просто віддає останні значення з пам'яті або з кешу в межах поточного запиту. Подальше оновлення DOM-елементів у браузері відбувається в межах одного кадру перерисовки інтерфейсу приблизно 16-33 мс. У підсумку сумарний час від фактичного вимірювання датчиками до появи оновлених значень на екрані користувача становить близько 1,05-1,2 с, де домінуючою складовою є саме інтервал відправлення даних з ESP32, а власне комунікаційні та обчислювальні затримки займають кілька мілісекунд.

Порівняно з іншими підходами до організації комунікацій, обрана архітектура поєднання HTTP-POST і SSE є компромісом між простотою реалізації та ефективністю. Використання класичного періодичного опитування з боку браузера через AJAX-запити означало б, що клієнт повинен сам ініціювати запит із певним інтервалом, дублюючи вже наявний потік даних від мікроконтролера. Це збільшило б кількість HTTP-з'єднань, навантаження на сервер і додало б додаткову затримку, рівну періоду опитування. Застосування WebSocket забезпечує найменший протокол-оверга й повноцінний двосторонній канал, однак потребує складнішої реалізації як на стороні серверу, так і на вбудованому контурі.

Також є можливість використовувати MQTT-протокол з брокером повідомлень. MQTT працює в умовах обмежених ресурсів і підтримує масштабування на десятки й сотні вузлів, але вимагає окремого брокера, додаткового компонента інтеграції між брокером та веб-сервером, а також адаптації браузерної частини, без адаптації для запису даних в базу даних вкладка моніторингу завжди повинна бути відкрита, що в свою чергу ускладнює процес збору даних.

Комунікаційний рівень побудований таким чином, щоб забезпечити регулярну передачу агрегованих даних з періодом 1 с, гарантований запис до бази даних із затримкою порядку кількох десятків мілісекунд і миттєве оновлення інтерфейсу користувача.

3.4 Порівняння системи з існуючими рішеннями

Для того щоб продемонструвати доцільність розробки в таблиці 3.1 наведено порівняння пристроїв для моніторингу якості повітря в приміщеннях: Aqara TVOC AAQS-S01, Xiaomi MIIIW, Bosean T-Z01, PROTMECH PTH-5, IQAir AirVisual Pro з розробленою інформаційно-аналітичною системою. У таблиці узагальнено, які параметри вимірює кожен прилад, який тип зв'язку

використовується, чи підтримується запис даних та з якою періодичністю оновлюються показники.

Таблиця 3.1 – Порівняння існуючих рішень з розробленою системою

	Параметри які вимірюються пристроями						Зв'язок	Запис даних	Оновлення даних секунд
	PM2,5	HCHO	TVOC	CO ₂	t, °C	%RH			
Aqara TVOC AAQS-S01	-	-	+	-	+	+	Zigbee	-	2 с
Xiaomi MIIIW	-	-	-	-	+	+	local	-	1 с
Bosean T-Z01	-	+	+	-	+	+	local	8	1-1,5 с
PROTMEX PTH-5	-	-	-	+	+	+	local	-	2 с
IQAir AirVisual Pro	+	+	-	+	+	+	Wi-Fi	24	1 с
Розроблена система	+	+	+	+	+	+	Wi-Fi/local	Обмежен о пам'яттю	1 с

Розглянемо переваги та недоліки описаних систем, які були продемонстровані. Ключовими факторами, які впливають на придатність пристрою до побудови повноцінної системи моніторингу мікроклімату, є кількість контрольованих параметрів, наявність запису даних, тип каналу зв'язку та швидкість оновлення показників. Більшість з наведених моделей, орієнтовані переважно на локальний контроль обмеженого набору параметрів і не підтримують довготривале зберігання вимірювань, що суттєво звужує можливості подальшого аналізу та побудови статистики. Пристрої Bosean T-Z01 та IQAir AirVisual Pro демонструють кращі можливості щодо частоти оновлення, однак або не мають повноцінного віддаленого доступу, або працюють у межах

закритої екосистеми виробника з фіксованими обмеженнями щодо обсягу історичних даних.

Залежність між кількістю вимірюваних параметрів і практичною цінністю системи, де для кожної моделі відображено кількість підтримуваних показників мікроклімату. Найменшу інформативність мають пристрої, що вимірюють лише TVOC або базові параметри t , % RH, тоді як IQAir AirVisual Pro та розроблена система забезпечують комплексний контроль декількох критично важливих показників якості повітря. Разом з тим, навіть за співставного набору параметрів розроблена система має перевагу завдяки можливості гнучкого розширення переліку сенсорів та їх конфігурації, що надає їй кращий потенціал для адаптації під конкретні умови експлуатації.

Другу важливу залежність демонструє порівняння швидкості оновлення та можливостей збереження даних. Пристрої з локальним відображенням показників і відсутністю повноцінного запису даних виявляються придатними лише для моментного контролю стану повітря, але не дозволяють відстежувати тенденції, виявляти аномалії. Натомість розроблена система, яка забезпечує інтервал оновлення близько однієї секунди та накопичення історії, обмежене лише ресурсами серверної бази даних, створює передумови для глибокого аналітичного опрацювання інформації та підтримки прийняття рішень щодо керування мікрокліматом.

Додатковою перевагою розробленої системи є використання веб-інтерфейсу з доступом через стандартний браузер. Це забезпечує кросплатформність і віддалений моніторинг, користувач може працювати з системою з персонального комп'ютера, планшета чи смартфона, перебуваючи поза межами контрольованого приміщення. На відміну від більшості розглянутих пристроїв, які обмежені локальним екраном або прив'язані до мобільного застосунку виробника, запропоноване рішення дає змогу масштабувати систему, інтегрувати її з іншими сервісами та будувати єдиний інформаційно-аналітичний простір для контролю мікроклімату в режимі реального часу.

3.5 Інтерфейс користувача

Інтерфейс користувача також є важливим елементом інформаційно-аналітичної системи моніторингу мікроклімату, оскільки забезпечує зручний доступ до результатів вимірювань та аналітичних даних для кінцевого користувача. Добре продуманий інтерфейс дозволяє оперативно оцінювати поточний стан середовища, швидко виявляти відхилення від нормативних значень та своєчасно реагувати на критичні ситуації. Для підвищення інформативності застосовуються графічні індикатори, кольорні маркери та текстові повідомлення про перевищення порогів. Веб-інтерфейс реалізовано з урахуванням адаптивної верстки, що забезпечує коректне відображення як на стаціонарних комп'ютерах, так і на мобільних пристроях.

На рисунку 3.5 зображено головну сторінку інформаційно-аналітичної системи моніторингу мікроклімату, а саме «Показники в реальному часі». У верхній частині інтерфейсу розташовано горизонтальне меню навігації з назвою системи та посиланнями на інші розділи веб-застосунку, а саме:

- мікроклімат;
- температура;
- вологість;
- пил;
- якість повітря;
- горючі гази;
- чадний газ;
- тиск;
- освітленість;
- база даних;
- інструкція користувача.

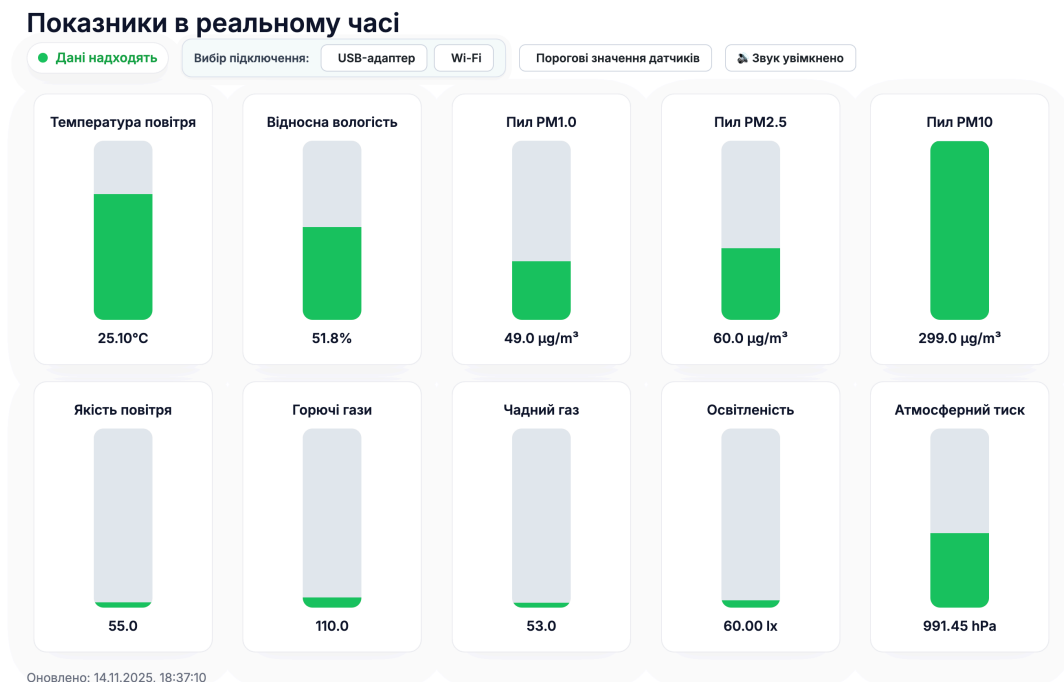


Рисунок 3.5 – Головна сторінка веб-інтерфейсу

Також на головній сторінці можна побачити індикатор чи надходять дані, якщо дані надходять індикатор буде зеленого кольору, зображено на рисунку 3.6, та з текстовим повідомленням «Дані надходять».

Показники в реальному часі

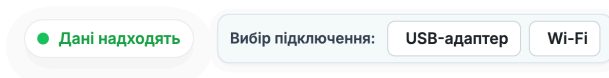


Рисунок 3.6 – Фрагмент інтерфейсу моніторингу в реальному часі з індикатором стану «Дані надходять»

Якщо дані не надходять індикатор змінюється на червоний колір (рис.3.7), що свідчить про відсутність надходження даних з ESP32 і з текстовим повідомленням «Дані не надходять».

Показники в реальному часі

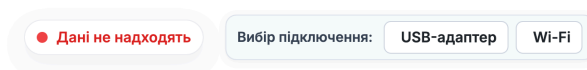


Рисунок 3.7 – Фрагмент інтерфейсу моніторингу в реальному часі з індикатором стану «Дані не надходять»

Якщо індикатор сірого кольору та текстовим повідомленням «Підключення...» сервер очікує підключення ESP32, зображено на рисунку 3.8.

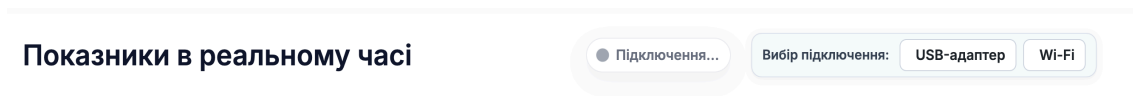


Рисунок 3.8 – Фрагмент інтерфейсу моніторингу в реальному часі з індикатором стану «Очікування підключення»

На поданому рисунку можна побачити вибір підключення «USB-адаптер», «Wi-Fi», ESP32 може підключатись через USB-адаптер, та через Wi-Fi. Для того щоб підключитись за допомогою адаптера USB-TypeC, потрібно підключити пристрій до ПК, після чого перейди в відповідну вкладку, натиснувши кнопку «USB-адаптер», посилає на дану сторінку, зображено на рисунку 3.9.

Мікроклімат приміщення Мікроклімат Температура Вологість Пил Якість повітря Горючі гази Чадний газ Тиск Освітленість База даних Інструкція користувача

USB → API міст

Натисніть "Підключити", оберіть порт до якого підключений ESP32

Підключити Відключити Порт відкрито

Чекаю рядки від ESP32...

Рисунок 3.9 – Підключення до сервера через USB-адаптер

Для того щоб ESP32 передавало дані, потрібно натиснути «Підключитись» та вибрати відповідний порт, до якого підключений мікроконтролер. Після цього з'явиться повідомлення «Порт відкрито». Оскільки реалізовано фільтрацію портів, необхідно обрати саме той, який відображається у відповідній вкладці (рис 3.10). Після успішного відкриття порту застосунок переходить у режим приймання даних, а користувач може відстежувати статус підключення за індикаторами в інтерфейсі.

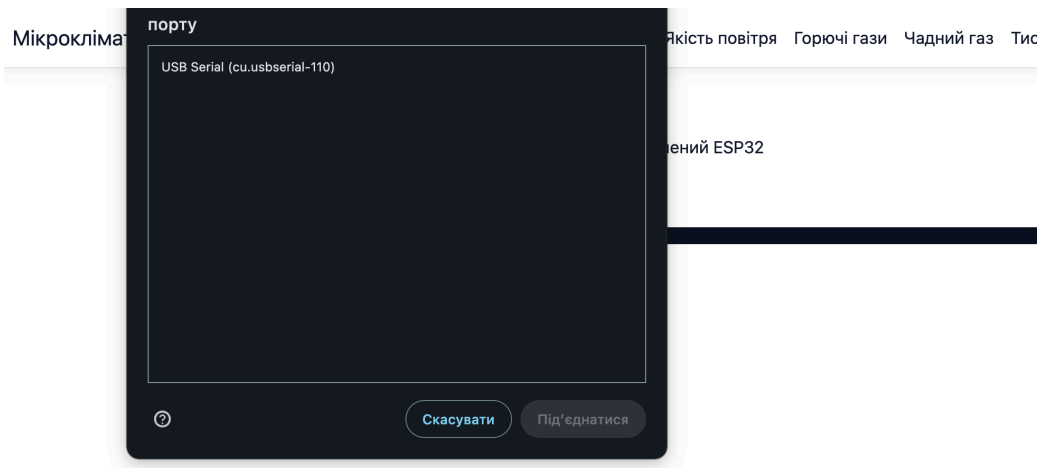


Рисунок 3.10 – Вибір порту для підключення

Після того, як було вибрано потрібний порт та підключено його, на вкладці буде повідомлення «очікування HTTP-POST». Коли датчики пройдуть ініціалізацію і дані будуть надходити на сервер, повідомлення зміниться на «Дані надходять».

Для того щоб підключитись по Wi-Fi, потрібно заживити ESP32 після чого, підключитись до Wi-Fi самого мікроконтролера під назвою «ESP32-Provision» з паролем «12345678», після чого потрібно натиснути «Wi-Fi», з вибору підключення, якщо цього не зробити, сторінка не відкриється, оскільки ця сторінка знаходиться не на основному сервері, а в самому мікроконтролері, після чого можна побачити сторінку підключення, зображено на рисунку 3.11.

Налаштування Wi-Fi та API

Сканувати доступні мережі

🔒 Моє добро (-78dBm) 🔒 TP-Link_DAA0 (-90dBm) 🔒 TP-Link_8A52 (-93dBm)

SSID

Пароль
(може бути порожнім)

API URL
http://MacBook-Pro-Vlad.local:5264/api/live

API Token (опційно)

Зберегти і перезапустити

Рисунок 3.11 – Підключення мікроконтролера через Wi-Fi

Для того, щоб підключитись, потрібно натиснути кнопку «Сканувати доступні мережі», після чого буде оновлено доступні мережі, потрібно вибрати потрібну мережу ввести пароль, та натиснути «Зберегти і перезапустити», якщо ESP32 підключиться до мережі то вас поверне на головну сторінку і дані будуть надходити на сервер.

Основну область екрану займає сітка карток з вертикальними індикаторами, кожна з яких відповідає окремому параметру мікроклімату. Для кожного показника температура повітря, відносна вологість, концентрації пилу PM1,0, PM2,5, PM10, якість повітря за MQ-135, горючі гази за MQ-2, чадний газ за MQ-7, освітленість, атмосферний тиск, відображається графічний стовпчиковий індикатор із заповненням зеленим кольором та відповідне числове значення з одиницями вимірювання, якщо норму перевищено то індикатор змінює колір на червоний, та буде звукове та текстове повідомлення, що параметри конкретного датчика завищені, також є можливість виключити або включити звукове сповіщення за допомогою кнопки «Звук увімкнено», зображено на рисунку 3.12.

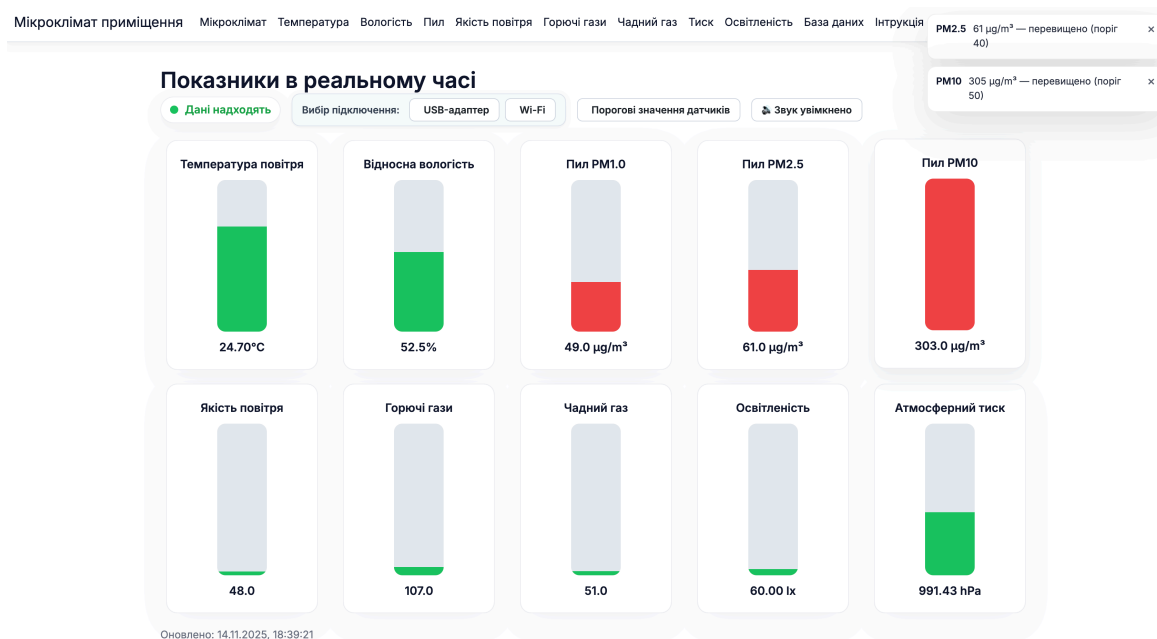


Рисунок 3.12 – Попередження про перевищення порогового значення

Натиснувши в навігаційному меню пункт «Вологість», можна побачити дані в графічному вигляді, інтерфейс якої показано на рисунку 3.14. Елементи фільтрації ідентичні як на вкладці «Температура» і працюють так само.

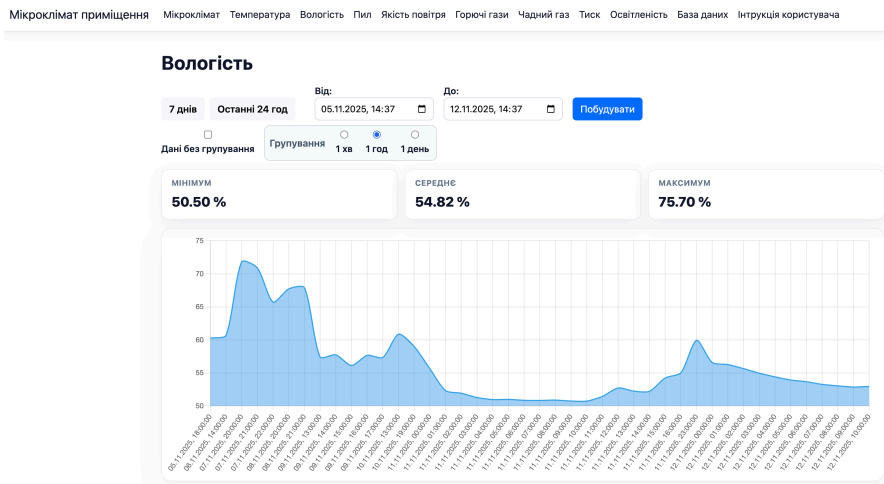


Рисунок 3.14 – Графічне представлення даних «Вологість»

Натиснувши в навігаційному меню «Пил», можна побачити сторінку де зображено дані в графічному вигляді, а також мінімальні, середні та максимальні значення в періоді, також на даній вкладці можна виділити конкретне значення та забрати його з відображення на графіку, для цього потрібно натиснути на PM1,0, PM2,5 або PM10 після чого на графіку зникнуть дані які були вибрані, зображено на рисунку 3.15.

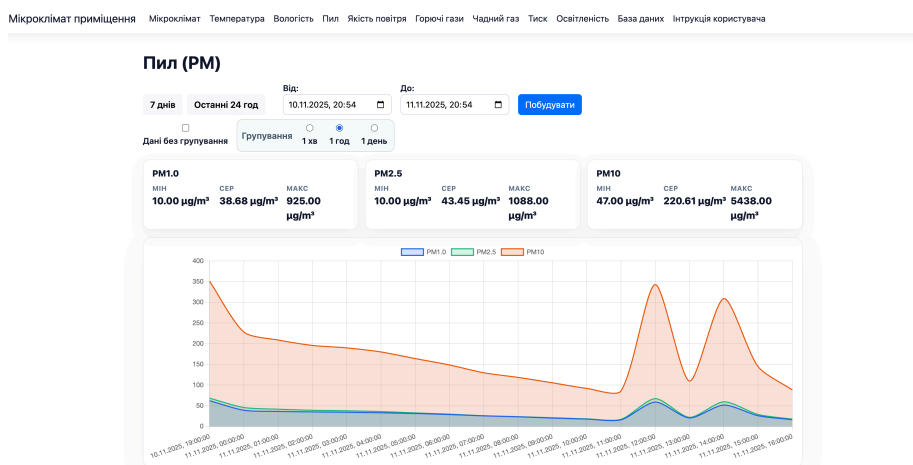


Рисунок 3.15 – Графічне представлення даних «Пил»

Графічне представлення даних «Якість повітря», можна побачити перейшовши на відповідну вкладку, дані в графічному вигляді зображено на рисунку 3.16.

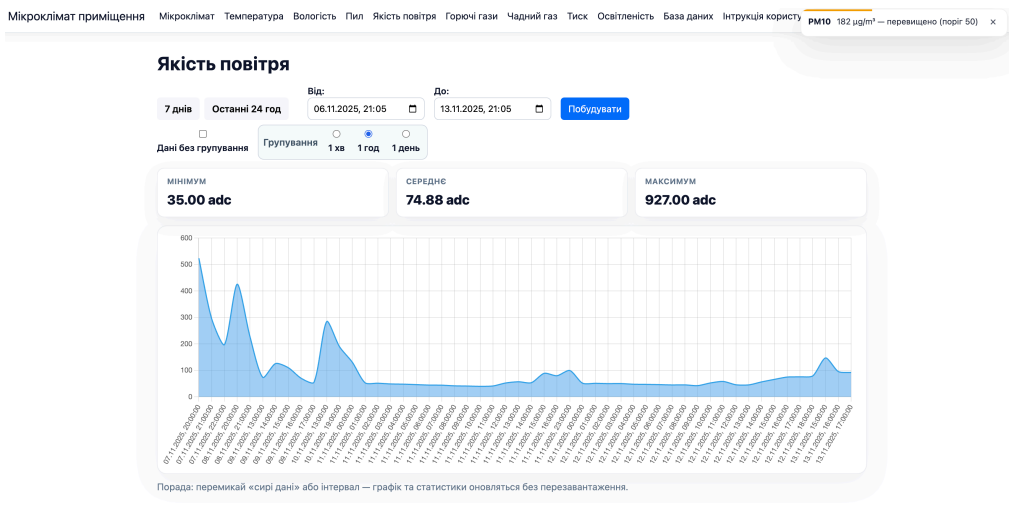


Рисунок 3.16 – Графічне представлення даних «Якість повітря»

Натиснувши в навігаційному меню «Горючі гази», можна побачити дані в графічному вигляді, які зображено на рисунку 3.17.

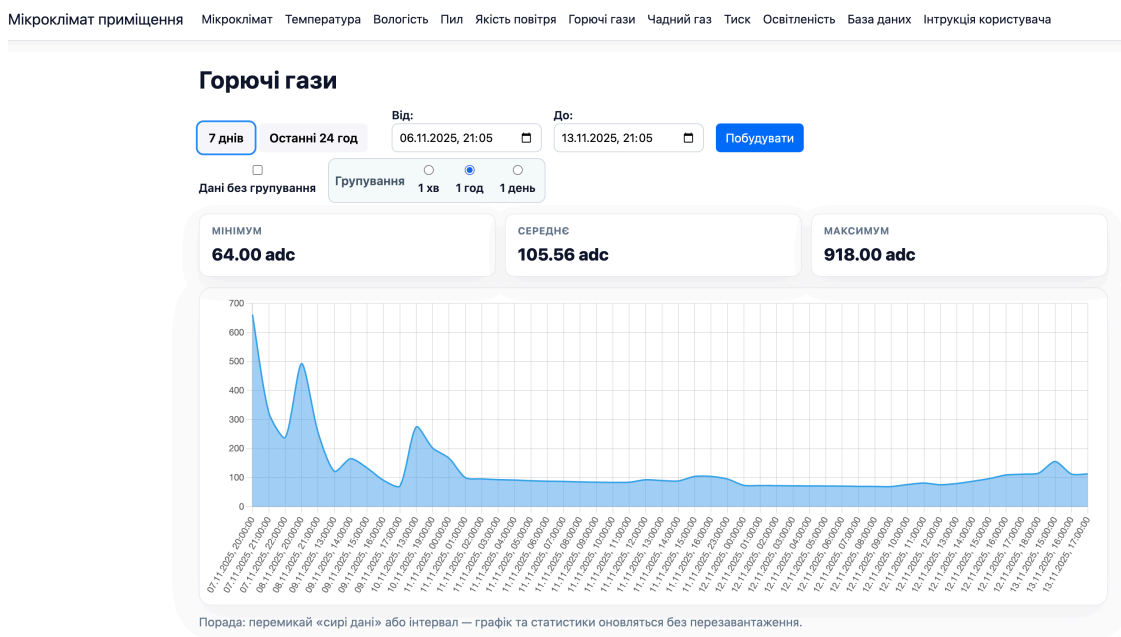


Рисунок 3.17 – Графічне представлення даних «Горючі гази»

Концентрацію чадного газу, отриману в ході вимірювань, можна побачити перейшовши на відповідну вкладку, яку зображено на рисунку 3.18.

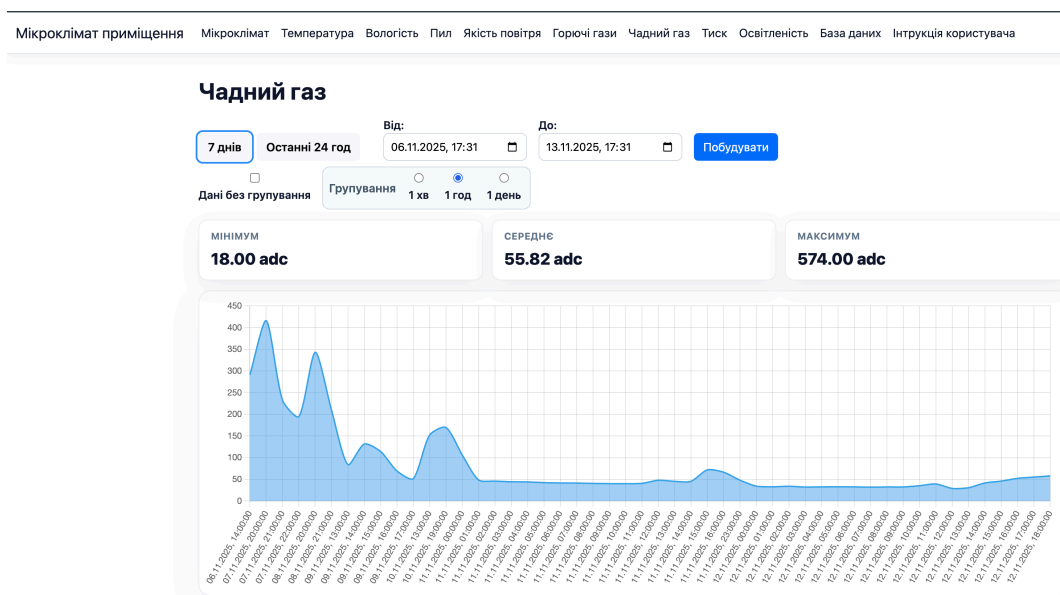


Рисунок 3.18 – Графічне представлення даних «Чадний газ»

На рисунку 3.19 зображено інтерфейс перегляду динаміки атмосферного тиску за вибраний період, де відображаються мінімальне, середнє та максимальне значення тиску, а також графік зміни показника в часі.

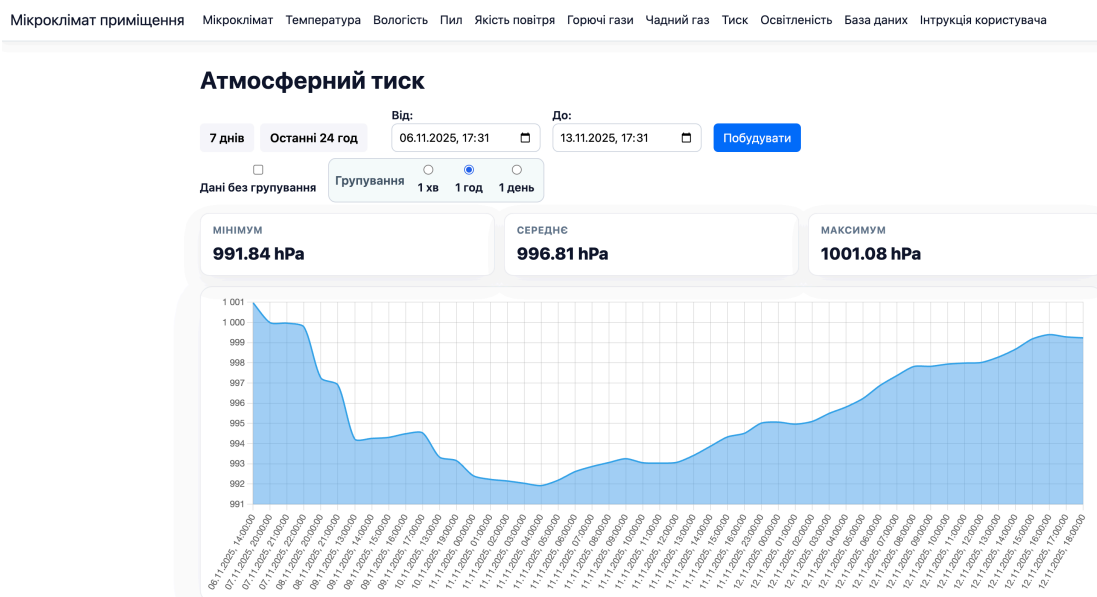


Рисунок 3.19 – Графічне представлення даних «Атмосферний тиск»

На рисунку 3.20 представлено інтерфейс перегляду динаміки освітленості за обраний період, де відображаються мінімальне, середнє та максимальне значення освітленості, а також графік зміни цього показника в часі.

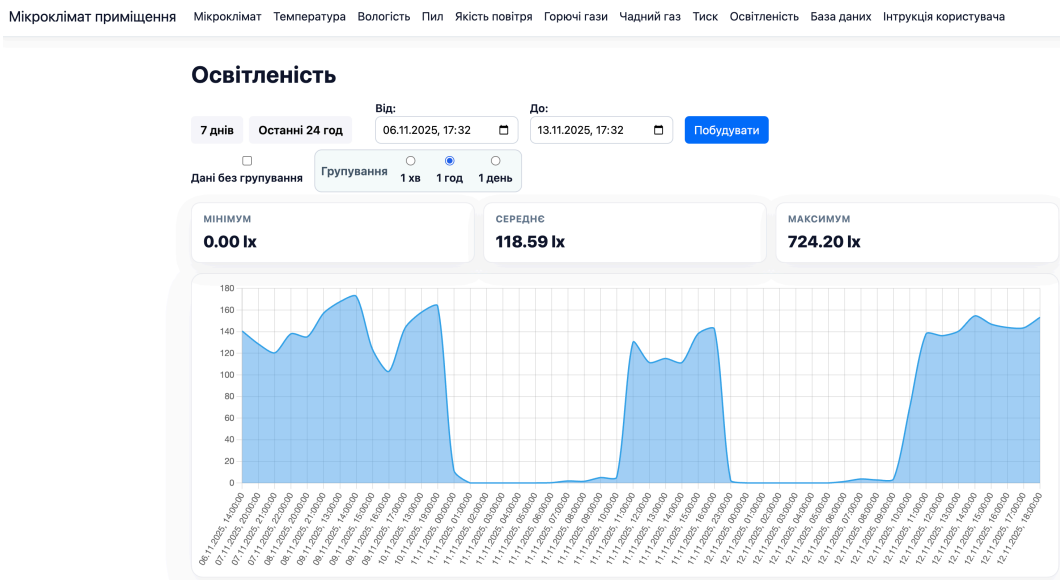


Рисунок 3.20 – Графічне представлення даних «Освітленість»

Натиснувши на головній сторінці пункт «Порогові значення датчиків», користувач переходить на сторінку, на якій можна задати порогові значення для кожного датчика відповідно до поточних потреб та умов експлуатації системи, дану сторінку зображено на рисунку 3.21.

Порогові значення

<p>Температура повітря Рекомендовано: від 18 °C до 26 °C</p> <p>Мінімум <input type="text" value="18"/> °C</p> <p>Максимум <input type="text" value="26"/> °C</p> <p>Відносна вологість Рекомендовано: від 30 % до 60 %</p> <p>Мінімум <input type="text" value="30"/> %</p> <p>Максимум <input type="text" value="60"/> %</p>	<p>Атмосферний тиск Рекомендовано: від 980 до 1035 hPa</p> <p>Мінімум <input type="text" value="980"/> hPa</p> <p>Максимум <input type="text" value="1035"/> hPa</p> <p>Освітленість Рекомендовано: від 10 до 1200 lx</p> <p>Мінімум <input type="text" value="10"/> lx</p> <p>Максимум <input type="text" value="1200"/> lx</p>	<p>Пил (PM) Приклад рекомендацій верхніх меж: PM1.0 ≤ 40, PM2.5 ≤ 50, PM10 ≤ 80 µg/m³</p> <p>PM1.0 (макс) <input type="text" value="30"/> µg/m³</p> <p>PM2.5 (макс) <input type="text" value="40"/> µg/m³</p> <p>PM10 (макс) <input type="text" value="50"/> µg/m³</p> <p>Газові датчики Значення вказуються в умовних одиницях ADC. Чим менше подій — тим рідше спрацюватиме попередження.</p> <p>Якість повітря (MQ-135) <input type="text" value="900"/></p> <p>Горючі газу (MQ-2) <input type="text" value="900"/></p> <p>Чадний газ (MQ-7) <input type="text" value="900"/></p>
--	--	--

[Зберегти порогові значення](#)

Рисунок 3.21 – Встановлення порогових значень мікроклімату

На рисунку 3.22 зображено інтерфейс журналу показників, у якому в табличному вигляді відображаються усі зафіксовані вимірювання з датою та часом, значеннями концентрацій газів, пилю, температури, вологості та атмосферного тиску. У верхній частині форми реалізовано фільтри за датою та діапазоном записів, що дає змогу обирати потрібний період спостережень і аналізувати детальні сирі дані сенсорів.

Мікроклімат приміщення Мікроклімат Температура Вологість Пил Якість повітря Горючі гази Чадний газ Тиск Освітленість База даних Інструкція користувача

Журнал показників

Від: 12.11.2025, 17:34 До: 13.11.2025, 17:34 Рядків: 50 Показати Останні 24 год

ID	DATE	Smoke/Gas (MQ-2)	CO (MQ-7)	VOCs (MQ-135)	PM1.0	PM2.5	PM10	°C	H%	Lux	Pressure
8712	2025-11-13 17:34:12	167	135	158	4	4	22	24,6	55,0	51	995,7
8711	2025-11-13 17:34:07	169	136	158	5	5	24	24,6	55,1	151	995,75
8710	2025-11-13 17:34:02	167	134	157	4	4	22	24,6	55,1	152	995,72
8709	2025-11-13 17:33:57	166	133	156	4	4	20	24,6	55,1	152	995,7
8708	2025-11-13 17:33:52	167	135	158	4	4	21	24,6	55,2	152	995,69
8707	2025-11-13 17:33:47	172	137	164	4	4	22	24,6	55,3	152	995,72
8706	2025-11-13 17:33:42	171	135	165	4	5	25	24,6	55,3	152	995,7
8705	2025-11-13 17:33:37	170	133	164	4	5	25	24,6	55,3	152	995,7
8704	2025-11-13 17:33:32	168	133	162	3	5	25	24,6	55,5	152	995,67
8703	2025-11-13 17:33:27	170	134	163	3	5	24	24,6	55,4	152	995,66
8702	2025-11-13 17:33:22	171	135	164	4	5	23	24,5	55,6	152	995,66
8701	2025-11-13 17:33:17	169	132	165	3	4	21	24,5	55,6	152	995,64
8700	2025-11-13 17:33:12	169	129	163	3	4	20	24,5	55,7	152	995,64
8699	2025-11-13 17:33:07	169	131	165	3	4	19	24,5	55,8	152	995,68
8698	2025-11-13 17:33:02	172	132	168	3	3	17	24,5	55,8	152	995,66

Рисунок 3.22 – Дані в табличному вигляді

На рисунку 3.23 зображено веб-сторінку «Інструкція користувача», що слугує довідковим розділом системи моніторингу мікроклімату. У верхній частині розміщено інформаційний банер з коротким поясненням призначення системи, нижче – блок «Основні розділи системи» з кнопками швидкого переходу до сторінок з поточними показниками, журналом даних та окремими графіками параметрів. Далі наведено розділ «Рекомендовані попередні налаштування» у вигляді маркованого списку, де описано, як підключити датчики, налаштувати мережу й параметри мікроконтролера. Блок «Оновлення даних» пояснює принцип роботи потоку вимірювань та значення індикаторів стану. Наприкінці сторінки подано короткий опис журналу показників та порад

користувачу, що допомагають інтерпретувати вимірювання та працювати із системою без спеціальної технічної підготовки.

я Горючі гази Чадний газ Тиск Освітленість База даних Інструкція користувача

Інструкція користувача

Як користуватися системою моніторингу мікроклімату

Призначення сторінки: Ця інструкція допоможе зорієнтуватися новим користувачам, пояснює як переглядати, аналізувати та розуміти показники сенсорів.

Основні розділи системи

Оберіть потрібний розділ, щоб перейти:

Показники в реальному часі

Температура

Вологість

Пил (PM)

Якість повітря

Горючі гази

Чадний газ

Освітленість

Атмосферний тиск

Журнал показників (База даних)

Рекомендовані порогові значення

- Температура:** комфортна 18–25°C; нижче — прохолодно, вище — спекотно.
- Вологість:** 40–60% — норма, <30% — занадто сухо, >70% — ризик конденсату.
- PM1.0, PM2.5, PM10:** тверді частинки пилу.
— **PM2.5** ≤ 35 µg/m³ — добре, **PM10** ≤ 50 µg/m³ — межа норми.
- MQ-135:** показник якості повітря (ЛОС, аміак, дим).
Менше значення = чистіше повітря.
- MQ-2:** горючі гази або дим. Ріст = небезпека витоку газу чи задимлення.
- MQ-7:** чадний газ (CO). Високі значення небезпечні — провітрити негайно.
- Освітленість:** 300–500 lx — робота; 100 lx — відпочинок; 50 lx — вечір.
- Атмосферний тиск:** ≈1013 hPa — норма; зниження → зміна погоди.

Оновлення даних

Система використовує технологію **Server-Sent Events (SSE)**, що забезпечує миттєве оновлення показників. Якщо SSE недоступне, сторінка переходить у **резервний режим опитування** кожні 2 секунди. Індикатор у верхній частині сторінки показує статус підключення:

- — дані надходять у реальному часі;
- — Очікує підключення.
- — підключення втрачено.

Журнал показників

У вкладці **"База даних"** можна переглядати історію вимірювань. Використовуйте фільтри "Від / До" для вибору періоду та кнопку **"Показати"**. Таблиця сортується за часом у порядку спадання (останні дані зверху).

Поради користувачеві

- На графіках можна змінювати масштаб, обирати період 24 год / 7 днів.
- Для тривалого аналізу вмикай **групування** за годиною або днем.
- При перевищенні порогових з'являється звукове та текстове сповіщення у правому верхньому куті.
- Усі дані зберігаються в базі даних

Рисунок 3.23 – Інструкція користувача

У результаті розроблено повноцінну інформаційно-аналітичну систему моніторингу мікроклімату приміщення в реальному часі. Спроектовано структуру вебзастосунку на платформі ASP.NET Core MVC, сформовано окремі шари представлення, бізнес-логіки та доступу до даних, а також побудовано базу даних PostgreSQL із використанням міграцій Entity Framework Core. Реалізовано програмне забезпечення мікроконтролера ESP32 з модульною прошивкою, яка забезпечує опитування сенсорів, попередню обробку вимірювань та передавання телеметрії на сервер через HTTP. Описано та налаштовано комунікаційний рівень системи з використанням поєднання HTTP POST і Server-Sent Events, що

забезпечує регулярну передачу даних та їх оперативне відображення в інтерфейсі. Створено зручний веб-інтерфейс із режимом «живих» показників, аналітичними графіками, журналом вимірювань, інструкцією користувача та системою оповіщення про перевищення порогових значень параметрів мікроклімату.

Отримані результати роботи системи показують коректну реакцію датчиків на реальні зміни умов у приміщенні: на графіках температури та вологості фіксуються очікувані коливання при провітрюванні та зміні режимів опалення, криві концентрації пилу та газів зростають у моменти появи відповідних домішок у повітрі, а показники освітленості узгоджуються зі зміною природного та штучного освітлення. Встановлені порогові значення параметрів мікроклімату забезпечують своєчасне спрацьовування системи оповіщення без надмірної кількості хибних спрацьовувань, що підтверджується побудованими графіками та журналом вимірювань. Це дає підстави вважати налаштування системи адекватними та придатними для подальшої експлуатації в умовах реального приміщення.

ВИСНОВКИ

За результатами проведеного дослідження варто зробити відповідні висновки. У роботі послідовно вирішено комплекс теоретичних та практичних завдань, пов'язаних із аналізом предметної області, вибором апаратно-програмних засобів, проектуванням архітектури та реалізацією інформаційно-аналітичної системи моніторингу мікроклімату приміщення в реальному часі.

Досліджено існуючі систем моніторингу мікроклімату у приміщенні в реальному часі. Показано, що більшість готових рішень або орієнтовані на обмежений набір параметрів, або не забезпечують накопичення історії вимірювань та гнучкої візуалізації. Виявлено проблеми, пов'язані з високою вартістю готових комплексів, закритістю протоколів обміну даними та обмеженими можливостями розширення функціоналу, що обґрунтовує доцільність створення власної відкритої системи.

Визначено вимоги до інформаційно-аналітичної системи моніторингу мікроклімату. Визначено функціональні та нефункціональні вимоги щодо переліку контрольованих параметрів: температура, відносна вологість, атмосферний тиск, освітленість, концентрація пилу, газоподібні домішки, періодичності вимірювань, необхідної точності сенсорів, способів передавання й зберігання даних, а також вимоги до масштабованості, енергоефективності та зручності користування системою. На основі цих вимог сформовано технічне завдання на розроблення системи.

Досліджено й порівняно протоколи передавання даних HTTP REST, MQTT, WebSocket та Server-Sent Events. Показано, що MQTT доцільно застосовувати для багатоточкових IoT-мереж з брокером, WebSocket – для двостороннього інтенсивного обміну, тоді як для задач односторонньої телеметрії від одного вузла до сервера достатньо комбінації HTTP REST та Server-Sent Events для потокового оновлення інтерфейсу користувача.

Обґрунтовано вибір саме такого підходу як компромісу між простотою реалізації, швидкодією та стабільністю роботи.

Спроектовано структуру бази даних та комунікаційного рівня системи моніторингу. Розроблено інформаційну та логічну моделі БД у СУБД PostgreSQL для зберігання показників сенсорів, параметрів пристроїв та порогових значень. Описано схему взаємодії між мікроконтролером ESP32, сервером ASP.NET Core MVC та веб-клієнтом, визначено формати JSON-повідомлень, маршрути REST-API та кінцеві точки SSE-потоків, що забезпечують цілісний цикл вимірювання від передачі даних до візуалізації.

Розроблено апаратно-програмний прототип системи на базі мікроконтролера ESP32 та комплексу сенсорів, а саме: DHT22, BMP280, BH1750, PMS5003, MQ-2, MQ-7, MQ-135. Прошивка організовує ініціалізацію обладнання, періодичне опитування сенсорів з різними інтервалами, фільтрацію та усереднення даних, формування пакетів телеметрії у форматі JSON і їх передавання на сервер. Отримано стабільну роботу вузла збору даних і забезпечено контроль усіх заданих параметрів мікроклімату.

Розроблено серверну частину та інтерфейс користувача інформаційно-аналітичної системи. Виконано реалізацію веб-застосунку на платформі ASP.NET Core MVC з використанням Entity Framework Core для роботи з PostgreSQL. Створено контролери API для приймання даних від ESP32, MVC-контролери для формування сторінок, а також Razor-представлення, що відображають поточний стан мікроклімату та журнали вимірювань. Розроблений інтерфейс є адаптивним, структурованим за окремими показниками та зрозумілим для кінцевого користувача.

Візуалізовано параметри мікроклімату в режимі реального часу та за обрані історичні періоди у веб-інтерфейсі. На основі отриманих даних формуються графіки, індикатори та агреговані показники мінімум, максимум, середнє значення для заданих відрізків часу, що дозволяє аналізувати динаміку змін мікроклімату, виявляти пікові навантаження та можливі порушення

нормативів. Використання SSE забезпечує майже миттєве оновлення візуальних компонентів без перезавантаження сторінки.

Реалізовано систему оповіщення про перевищення порогових значень параметрів мікроклімату та виконано тестування роботи комплексу. При виході вимірювань за допустимі межі індикатори на панелі змінюють колір, у користувача з'являються текстові попередження та спливаючі повідомлення. Як показали експериментальні дослідження, сумарні комунікаційні та обчислювальні затримки не перевищують декількох десятків мілісекунд і є значно меншими за період опитування сенсорів, що відповідає вимогам до систем реального часу.

На основі отриманих результатів можемо зробити висновок, що поставлена мета – розроблення інформаційно-аналітичної системи моніторингу мікроклімату приміщення в реальному часі – досягнута повністю. Розроблену систему можна використовувати як прототип для впровадження в навчальних аудиторіях, офісних та житлових приміщеннях, серверних кімнатах та інших об'єктах, де необхідний безперервний контроль параметрів повітря. Подальший розвиток роботи доцільно спрямувати на розширення мережі сенсорних вузлів, інтеграцію з системами керування мікрокліматом, впровадження інтелектуальних методів аналізу часових рядів.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Скиба В. О. Багаторівнева веб-система збору та аналітики параметрів середовища в реальному часі. Період трансформаційних процесів в світовій науці: задачі та виклики: Зб. наук. пр. з матеріалами VI Міжнар. наук. конф., м. Полтава, 28 листоп. 2025 р. С. 436-443.
2. Balocco C., Vestrucci S. An experimental study of museum showcases in Florence under real operating conditions. *International Journal of Conservation Science*. 2020. Vol. 11, № 4. P. 979-996.
3. ІПС ЛІГА:ЗАКОН – система пошуку, аналізу та моніторингу нормативно-правової бази. Державні медико-санітарні нормативи. URL: <https://surl.li/qkzpzs> (дата звернення: 08.09.2025).
4. ДСН 3.3.6.042-99. Санітарні норми мікроклімату виробничих приміщень. [Чинний від 1999-12-01]. URL: <https://zakon.rada.gov.ua/go/va042282-99> (дата звернення: 08.09.2025).
5. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Офіційний вебпортал парламенту України. URL: <https://surl.lt/jodvqf> (дата звернення: 08.11.2025).
6. Про охорону праці : Закон України від 14 жовтня 1992 р. № 2694-ХІІ. URL: <https://zakon.rada.gov.ua/laws/show/2694-12> (дата звернення: 08.09.2025).
7. ДБН В.2.5-67:2013. Опалення, вентиляція та кондиціонування. [Чинний від 2014-01-01]. Київ, 2013. URL: <https://surl.li/nzoczt> (дата звернення: 09.09.2025).
8. Омелянов О. М. Безпека праці та життєдіяльності. Безпека праці та життєдіяльності : навч. посіб. / О. М. Омелянов, А. В. Спірін, І. В. Твердохліб; Вінниц. нац. аграр. ун-т. Вінниця, 2020. 334 с. URL: <http://repository.vsau.org/getfile.php/25530.pdf> (дата звернення: 09.09.2025).
9. ІПС ЛІГА:ЗАКОН - система пошуку, аналізу та моніторингу нормативно-правової бази. URL: <https://surl.lt/dabgpg> (дата звернення: 09.09.2025).

10. Flammable Gas Sensor. Manual. Version 1.4. Zhengzhou Winsen Electronics Technology Co., Ltd., 2020. 7 p. URL: <https://surl.li/ffqrrx> (date of access: 10.09.2025).

11. Жураковський Б. Ю., Зенів І. О. Технології інтернету речей : навчальний посібник для здобувачів ступеня бакалавра за освітньою програмою «Інформаційне забезпечення робототехнічних систем». Київ : КПІ ім. Ігоря Сікорського, 2021. 271 с. URL: <https://surl.li/tqwsir> (дата звернення: 10.09.2025).

12. Farsi M., Daneshkhah A., Hosseinian-Far A., Jahankhani H. Technology and Applications of the Internet of Things: Smart Devices, Smart Environments and Smart Cities. Springer, 2020. 289 p.

13. Jo J., Jo B., Kim J., Kim S., Han W. Development of an IoT-Based Indoor Air Quality Monitoring Platform. Journal of Sensors. 2020. Vol. 2020. Article ID 8749764. 14 p.

14. MPLC. Датчик якості повітря Aqara TVOC Air Quality Monitor AAQS-S01. URL: <https://epicentrk.ua/ua/shop/mplc-datchik.html> (дата звернення: 10.09.2025).

15. Датчик температури і вологості Xiaomi MiiW S210 (MW22S06). URL: <https://surl.li/dlndhz> (дата звернення: 10.09.2025).

16. PROTMEХ PTH-5 – цифровий термогігрометр із вимірюванням рівня вуглекислого газу. URL: <https://surl.li/ogoovt> (дата звернення: 11.09.2025).

17. Детектор якості повітря з дисплеєм для вимірювання CO₂, температури та вологості. URL: <https://surl.li/whorbd> (дата звернення: 11.09.2025).

18. Метеостанція Носо D5 з кольоровим дисплеєм та сенсорами температури, вологості й. URL: <https://surl.li/pmykan> (дата звернення: 11.09.2025).

19. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Офіційний вебпортал парламенту України. URL: <https://surl.li/ongyzv> (дата звернення: 12.09.2025).

20. Допустимі концентрації пилу у приміщенні. StudFiles. URL: <https://studfile.net/preview/15500927/page:20/> (дата звернення: 12.09.2025).

21. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99 : Постанова МОЗ України від 01.12.1999 № 42. База даних «Законодавство України» / Верховна Рада України. URL: <https://zakon.rada.gov.ua/go/va042282-99> (дата звернення: 12.09.2025).

22. Про затвердження гігієнічних регламентів допустимого вмісту хімічних і біологічних речовин у повітрі робочої зони. Офіційний вебпортал парламенту України. URL: <https://zakon.rada.gov.ua/go/z0741-20> (дата звернення: 12.09.2025).

23. ДБН В.2.5-28:2018. Природне і штучне освітлення. Державні будівельні норми України. URL: <https://surl.li/ratqci> (дата звернення: 12.09.2025).

24. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Офіційний вебпортал парламенту України. URL: <https://surl.li/ldxcwm> (дата звернення: 12.09.2025).

25. Arduino MKR1000 WiFi. Arduino Documentation. URL: <https://www.arduino.cc/en/Guide/MKR1000> (date of access: 12.09.2025).

26. STM32F103C8T6 – 32-bit ARM Cortex-M3 Microcontroller. STMicroelectronics – Official Website. URL: <https://surl.li/clwgqb> (date of access: 12.09.2025).

27. ESP32 Series – Wi-Fi & Bluetooth Microcontrollers. Espressif Systems – Official Website. URL: <https://surl.li/iruwsh> (date of access: 13.09.2025).

28. DHT11 – Digital Temperature and Humidity Sensor : Product Manual. Aosong (Guangzhou) Electronics Co., Ltd. URL: <https://surl.li/seeiie> (date of access: 13.09.2025).

29. DHT22 – Digital Humidity and Temperature Sensor : Datasheet. Aosong (Guangzhou) Electronics Co., Ltd. URL: <https://surl.li/qjpovz> (date of access: 13.09.2025).

30. BME280 – Combined Digital Humidity, Pressure and Temperature Sensor : Datasheet. Bosch Sensortec GmbH. URL: <https://surl.li/bpyptq> (date of access: 13.09.2025).

31. GP2Y1010AU0F – Dust Sensor, Optical Type : Datasheet. SHARP Corporation. URL: <https://surl.li/ldyuol> (date of access: 13.09.2025).
32. PMS5003 – Digital Universal Particle Concentration Sensor : Product Manual. Plantower. URL: <https://surl.li/txycwq> (date of access: 14.09.2025).
33. MQ-2 Semiconductor Sensor for Combustible Gas : Technical Data. Hanwei Electronics Co., Ltd. URL: <https://surl.lt/eauark> (date of access: 14.09.2025).
34. MQ-7 Semiconductor Sensor for Carbon Monoxide : Technical Data. Hanwei Electronics Co., Ltd. URL: <https://surl.li/iajqtu> (date of access: 14.09.2025).
35. MQ-135 Semiconductor Gas Sensor : Manual. Zhengzhou Winsen Electronics Technology Co., Ltd. URL: <https://surl.li/zaswfp> (date of access: 14.09.2025).
36. BMP280 – Digital Pressure Sensor : Datasheet. Bosch Sensortec GmbH. URL: <https://surl.li/cnexpa> (date of access: 14.09.2025).
37. BH1750FVI – Digital 16-bit Serial Output Ambient Light Sensor IC : Datasheet. ROHM Co., Ltd. URL: <https://surl.li/bkrdyr> (date of access: 15.11.2025).
38. TSL2561 – Ambient Light Sensor / Luminosity Sensor : Datasheet. ams-OSRAM / TAOS Inc. URL: <https://surl.li/bnkhkp> (date of access: 15.11.2025).
39. Python Software Foundation. The Python Tutorial. Офіц. сайт Python. URL: <https://docs.python.org/uk/3/tutorial/> (date of access: 15.09.2025).
40. Microsoft. Тип мовою C# (C# documentation). Microsoft Learn. URL: <https://surl.li/zewlgt> (date of access: 15.11.2025).
41. cppreference.com. C and C++ reference. URL: <https://surl.li/hhqezs> (date of access: 15.09.2025).
42. MDN Web Docs. HTTP: Hypertext Transfer Protocol. Mozilla Foundation. URL: <https://developer.mozilla.org/en-US/docs/Web/HTTP> (date of access: 15.09.2025).
43. EMQ Technologies. MQTT 5.0: 7 New Features and a Migration Checklist. EMQ Blog. – 09.06.2025. URL: <https://surl.lu/lgjizq> (date of access: 15.11.2025).
44. MDN Web Docs. The WebSocket API (WebSockets). Mozilla Foundation. URL: <https://surli.cc/xuvobf> (date of access: 15.09.2025).

45. MDN Web Docs. Server-sent events. Mozilla Foundation. URL: <https://surl.li/kertxu> (date of access: 15.09.2025).

46. Oracle Corporation. MySQL 8.0 Reference Manual. – Oracle, 2025. URL: https://docs.oracle.com/cd/E17952_01/mysql-8.0-en/ (date of access: 15.11.2025).

47. The PostgreSQL Global Development Group. PostgreSQL Documentation. Офіц. сайт PostgreSQL. URL: <https://www.postgresql.org/docs/> (date of access: 15.09.2025).