

БАКАЛАВРСЬКА РОБОТА

КРБ.СІ – 04.00.00.000 ПЗ

Група СІ-21-1

Володимир Завадовський

2025

Міністерство освіти і науки України
Івано-Франківський національний університет нафти і газу
Інститут інформаційних технологій
Кафедра інформаційно-телекомунікаційних технологій та систем

Завадовський Володимир Андрійович

(прізвище, ім'я, по-батькові)

УДК _____
(індекс)

БАКАЛАВРСЬКА РОБОТА

Автоматизована система дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні
(назва роботи)

Системна інженерія - Інтернет речей

(назва освітньої програми)

151 Автоматизація та комп'ютерно-інтегровані технології

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня _____ **В.А.Завадовський**
(підпис, ініціали та прізвище здобувача)

Науковий керівник _____ **Паньків Х. В. к.т.н., доцент кафедри ІТТС**
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту
Завідувач кафедри

д.т.н., професор _____ **Л. М. Заміховський**
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ – 2025

Івано-Франківський національний технічний університет нафти і газу

(повне найменування закладу вищої освіти)

Інститут інформаційних технологій

Кафедра інформаційно-телекомунікаційних технологій та систем

Освітній рівень бакалавр

Спеціальність 151 «Автоматизація та комп'ютерно-інтегровані технології»

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри ІТТС

д.т.н., проф. Заміховський Л.М

«___» _____ 2025 року

ЗАВДАННЯ

НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ

Завадовському Володимирі Андрійовичу

(прізвище, ім'я, по-батькові)

- Тема роботи Автоматизована система дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні
керівник роботи Паньків Х. В. к.т.н., доцент кафедри ІТТС,
затвержені наказом закладу вищої освіти від «05» травня 2025 року №281/7
- Строк подання студентом роботи 15.06.2025
- Вихідні дані до роботи: Вимоги до кліматичних умов в серверному приміщенні
- Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
 - Аналіз сучасних технологій для моніторингу та управління кліматичними умовами
 - Проектування автоматизованої системи дистанційного контролю та управління
 - Реалізація програмного забезпечення та тестування системи
- Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
 - Блок-схема основного циклу роботи.
 - Діаграма алгоритму роботи клієнтської частини
 - Структурна схема автоматизації системи контролю клімату серверної кімнати
- Дата видачі завдання 01.03.2025

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Термін виконання етапів роботи	Примітка
1	<i>Аналіз сучасних технологій для моніторингу та управління кліматичними умовами</i>	1.04.2025	<i>виконано</i>
2	<i>Проектування автоматизованої системи дистанційного контролю та управління</i>	1.05.2025	<i>виконано</i>
3	<i>Реалізація програмного забезпечення та тестування системи</i>	1.06.2025	<i>виконано</i>
4	<i>Оформлення пояснювальної записки</i>	15.06.2025	<i>виконано</i>

Студент

(підпис)

В.А.Завадовський

(прізвище та ініціали)

**Керівник
роботи**



(підпис)

Х.В.Паньків

(прізвище та ініціали)

АНОТАЦІЯ

Завадовський В. А. Автоматизована система дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні ІФНТУНГ, 2025. 85 с.

Бакалаврська робота на здобуття освітнього ступеня бакалавр за освітньо-професійною програмною «Системна інженерія – Інтернет речей», спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології». Івано-Франківський національний університет нафти і газу. Івано-Франківськ, 2025.

У дослідженні проаналізовано сучасні методи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні.

У роботі розроблено систему на основі ESP32, що забезпечує збір даних із сенсорів, керування обладнанням через реле та передачу інформації на сервер. Серверну частину реалізовано з використанням Raspberry Pi, яка приймає дані через MQTT або HTTP, зберігає їх у базі даних і візуалізує за допомогою веб-інтерфейсу на HTML/CSS/JavaScript. Систему протестовано, підтверджено її ефективність та надано рекомендації щодо покращень.

Ключові слова: ДИСТАНЦІЙНИЙ МОНІТОРИНГ , ESP32, СЕРВЕРНЕ ПРИМІЩЕННЯ, MQTT, RASPBERRY PI, КЛІМАТИЧНИЙ КОНТРОЛЬ, IOT

ANNOTATION

Zavadovskyi V. A. Automated system for remote monitoring and control of climate conditions in the server room. IFNTUNG, 2025, 85 p.

Bachelor's thesis for obtaining a bachelor's degree under the educational-professional program «System Engineering – Internet of Things», specialty 151 «Automation and computer-integrated technologies». Ivano-Frankivsk National Technical University of Oil and Gas. Ivano-Frankivsk, 2025.

The study analyzes modern methods of remote monitoring and control of climate conditions in a server room.

The work presents a system based on the ESP32 micrcontroller, which collects data from sensors, controls equipment via relays, and data transmission to a server. The server side is implemented using a Raspberry Pi, which receives data via MQTT or HTTP, stores it in a database, and visualizes it through a web interface built with HTML/CSS/JavaScript. The system has been tested, its effectiveness confirmed, and recommendations for improvement have been provided.

Keywords: REMOTE MONITORING, ESP32, SERVER ROOM, MQTT, RASPBERRY PI, CLIMATE CONTROL, LOT

РЕФЕРАТ

Розрахункова-пояснювальна записка: 85 с., 34 рисунків, 12 табл.,
44 джерел.

Об'єкт дослідження – процес моніторингу та керування мікрокліматичними параметрами в технічних приміщеннях, зокрема у серверних кімнатах.

Предмет дослідження – структура, принципи побудови та програмно-апаратна реалізація автоматизованої системи дистанційного моніторингу і управління кліматичними умовами з використанням сучасних цифрових технологій.


Мета роботи – проєктування, розробка, реалізація та тестування ефективної, безпечної та масштабованої системи дистанційного контролю кліматичних умов у серверному приміщенні на базі мікроконтролера ESP32 із використанням протоколу MQTT, вебінтерфейсу та ряду цифрових сенсорів.

У дослідженні проаналізовано сучасні методи та засоби дистанційного моніторингу кліматичних умов у серверних приміщеннях, обґрунтовано вибір апаратної та програмної складової системи.

ДИСТАНЦІЙНИЙ МОНІТОРИНГ , ESP32, СЕРВЕРНЕ ПРИМІЩЕННЯ,
MQTT, RASPBERRY PI, КЛІМАТИЧНИЙ КОНТРОЛЬ, IOT

ЗМІСТ

	с.
ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ.....	8
ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ ДЛЯ МОНИТОРИНГУ ТА УПРАВЛІННЯ КЛІМАТИЧНИМИ УМОВАМИ.....	10
1.1 Вимоги до мікроклімату серверного приміщення.....	10
1.2 Існуючі апаратно-програмні рішення для контролю клімату.....	13
1.3 Протоколи передачі даних у системах моніторингу.....	19
РОЗДІЛ 2. ПРОЕКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ ДИСТАНЦІЙНОГО КОНТРОЛЮ ТА УПРАВЛІННЯ.....	23
2.1 Архітектура системи та вибір апаратного забезпечення.....	23
2.2 Розробка логіки взаємодії модулів системи.....	35
2.3 Засоби візуалізації та інтерфейс користувача.....	40
2.4 Безпека та надійність системи.....	44
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ СИСТЕМИ.....	51
3.1 Розробка програмного модуля збору та обробки даних.....	53
3.2 Реалізація інтерфейсу керування кліматом.....	57
3.3 Тестування системи в умовах реального середовища.....	68
3.4 Аналіз ефективності функціонування системи.....	73
ВИСНОВКИ.....	78
ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА.....	81

					КРБ.СІ-04.00.00.000 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Завадовський В.А.			Літ.	Арк.	Акрушіє
Перевір.		Паньків Х.В.				7	85
Н. Контр.		Возний А.В.			ІФНТУНГ СІ-21-1		
Затверд.		Заміховський Л. М					

Автоматизована система
дистанційного моніторингу та
управління кліматичними
умовами в серверному
приміщенні

ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

АС - Автоматизована система

КС - Кліматична система

ПЗ – Програмне забезпечення

АЗ – Апаратне забезпечення

IoT – Internet of Things

MQTT – Message Queuing Telemetry Transport

HTTP – Hypertext Transfer Protocol

GUI – Graphical User Interface

UI - User Interface

JSON - JavaScript Object Notation

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

ВСТУП

Серверні приміщення стали основою функціонування критичної ІТ-інфраструктури, забезпечуючи безперервну роботу цифрових сервісів, обробку й зберігання великих обсягів даних, стабільність роботи підприємств, установ і організацій. Одним із ключових факторів, що впливають на надійність і довговічність серверного обладнання, є стабільний мікроклімат у приміщенні, де воно розміщене. Перегрів, надмірна вологість або забрудненість повітря можуть призвести до збоїв у роботі апаратури, її фізичного зношення або навіть повного виходу з ладу. При цьому традиційні системи кліматичного контролю зазвичай потребують ручного спостереження або не мають функцій гнучкого дистанційного керування, що знижує ефективність реагування на позаштатні ситуації.

Розв'язання проблеми створення доступної, надійної та функціональної автоматизованої системи дистанційного моніторингу та керування кліматичними умовами є важливим як у технічному, так і в економічному контексті. Впровадження подібних рішень дозволяє мінімізувати людський фактор, знизити витрати на обслуговування, підвищити рівень безпеки обладнання та забезпечити оперативне реагування на зміну кліматичних параметрів. Розробка таких систем, заснованих на використанні сучасних мікроконтролерів, сенсорів, протоколів передавання даних та вебінтерфейсів, є перспективним напрямом розвитку прикладної інформатики, автоматизації й Інтернету речей (IoT).

Але, незважаючи на це, сьогодні існує потреба у дослідженні, яке б узагальнило, систематизувало існуючі відомості з даної проблеми.

Враховуючи все вищесказане, нами і була обрана тема дипломної роботи: "Автоматизована система дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні".

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

1. АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ ДЛЯ МОНИТОРИНГУ ТА УПРАВЛІННЯ КЛІМАТИЧНИМИ УМОВАМИ

1.1 Вимоги до мікроклімату серверного приміщення

Серверні приміщення є технічно важливими об'єктами інфраструктури будь-якого підприємства, установи чи організації, що обробляє та зберігає великі обсяги інформації. У зв'язку з високою щільністю розміщення електронного обладнання, постійною його роботою у безперервному режимі, а також високими вимогами до безпеки та надійності функціонування серверів, забезпечення відповідного мікроклімату в таких приміщеннях є критичним фактором. Правильно організований мікроклімат сприяє зменшенню зносу обладнання, зниженню ризику перегріву та запобіганню аварійним ситуаціям. У цьому розділі розглянуто основні нормативні, технічні та практичні вимоги до мікроклімату серверного приміщення, а також обґрунтовано необхідність використання автоматизованих систем моніторингу й управління параметрами клімату.

Під терміном «мікроклімат» слід розуміти сукупність таких показників, як температура повітря, вологість, рівень запиленості, швидкість повітрообміну, вентиляція, а також відсутність перепадів напруги та статичної електрики. Всі ці параметри впливають на працездатність, стабільність і довговічність серверного обладнання. Найбільш критичними з них є температура і вологість, оскільки саме вони безпосередньо впливають на фізичні характеристики елементної бази комп'ютерної техніки [1].

Мікроклімат приміщення - це комплекс фізичних умов внутрішнього середовища, що визначається такими параметрами, як температура та вологість повітря. Міжнародний стандарт TIA/EIA-569, розроблений спільно Канадою і США і широко застосовуваний у світі, встановлює основні вимоги до організації серверних кімнат.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

Цей стандарт охоплює не лише конструктивні характеристики приміщень, але й умови освітлення, прокладки кабелів, навантаження на підлогу, а також регламентує параметри мікроклімату, які мають суворо дотримуватись. Особливо наголошується, що контроль кліматичних умов є одним із ключових факторів для надійної роботи серверного обладнання.

Загалом, незалежно від специфіки конкретної серверної, обов'язковий моніторинг охоплює такі кліматичні показники, як температура і вологість. Відповідно до стандарту, рекомендовані норми такі:

- температура повітря в межах +18 °С до +27 °С;
- відносна вологість при +24 °С – від 40% до 60%.

Ця система має функціонувати безперервно, цілодобово протягом усього року.

Для кращого розуміння впливу кліматичних параметрів на роботу серверного обладнання наведемо аналіз причин та наслідків їх впливу (Таблиця 1.1).

Таблиця 1.1 - Вплив кліматичних факторів на обладнання СК

Кліматичний фактор	Причина впливу на обладнання СК	Наслідок впливу кліматичних факторів на обладнання СК
Температура	<ol style="list-style-type: none"> 1. Виділення тепла обладнанням. 2. Наявність системи опалення. 3. Вплив зовнішніх факторів (сонячне випромінювання та інше). 	<ol style="list-style-type: none"> 1. Вихід обладнання з ладу. 2. Зменшення строку служби обладнання.
Вологість	<ol style="list-style-type: none"> 1. Погана вентиляція. 2. Виникнення Конденсату. 3. Погана вентиляція. 	<ol style="list-style-type: none"> 1. Вихід з ладу електричних елементів обладнання. 2. Корозія електричних елементів. 3. Замикання електро замків.

Таким чином, забезпечення належного мікроклімату є не просто рекомендацією, а необхідністю, що безпосередньо впливає на ефективність роботи всієї серверної інфраструктури. Впровадження автоматизованих систем моніторингу і регулювання кліматичних умов стає ключовим фактором у підтриманні необхідного рівня якості внутрішнього середовища, що, у свою чергу, дозволяє уникнути значних фінансових втрат, пов'язаних з ремонтом і заміною обладнання, а також забезпечує безперервність роботи інформаційних технологій [2].

Згідно з державними санітарними нормами та вимогами виробників серверного обладнання, оптимальна температура в серверному приміщенні має знаходитися в межах від $+18^{\circ}\text{C}$ до $+27^{\circ}\text{C}$, з рекомендованим значенням близько $+22^{\circ}\text{C}$. Занадто висока температура призводить до перегріву елементів системних блоків, жорстких дисків, блоків живлення, що може спричинити збої, втрату даних, вихід обладнання з ладу. Надмірне охолодження, у свою чергу, несе ризики конденсації вологи на елементах, що теж є небезпечним. У зв'язку з цим рекомендовано підтримувати стабільний температурний режим, із допустимими коливаннями не більше $\pm 1-2^{\circ}\text{C}$ протягом доби.

Ще одним важливим параметром є вологість повітря. Відповідно до міжнародного стандарту ASHRAE TC 9.9 (Американське товариство інженерів з опалення, охолодження та кондиціонування повітря), відносна вологість у серверному приміщенні повинна коливатися в межах від 40% до 60%. Якщо вологість перевищує 60%, виникає ризик конденсації на поверхнях обладнання, особливо в умовах змін температури. Водночас надто низький рівень вологості (менше 40%) сприяє накопиченню статичної електрики, яка може пошкодити мікросхеми та інші елементи електроніки. Тому підтримка вологості в межах допустимих норм є не менш важливою, ніж контроль температури.

Особливу увагу слід звертати на рівень забрудненості повітря в серверному приміщенні. Наявність пилу у вентиляційних каналах, на радіаторах

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

охолодження або всередині системних блоків негативно впливає на теплообмін та може стати причиною перегріву. Тому необхідно впроваджувати фільтраційні системи, регулярно очищати вентиляційні отвори, а також здійснювати моніторинг чистоти повітря. Окрім цього, при проектуванні серверних приміщень враховується відсутність відкритих джерел пилу, можливість герметизації приміщення та організація замкненого повітряного обміну із застосуванням якісних фільтрів [3].

У підсумку варто наголосити, що дотримання вимог до мікроклімату в серверному приміщенні є не лише запорукою надійної роботи комп'ютерного обладнання, але й важливим аспектом інформаційної безпеки, збереження ресурсів і довговічності всієї ІТ-інфраструктури. Упровадження сучасних автоматизованих систем контролю мікроклімату дозволяє значно підвищити ефективність функціонування серверних кімнат, скоротити витрати на обслуговування та зменшити ризики виникнення аварійних ситуацій.

1.2 Існуючі апаратно-програмні рішення для контролю клімату

На сучасному етапі розвитку інформаційних технологій серверні приміщення відіграють ключову роль у функціонуванні будь-якої організації. Надійність, безперервність та безпека роботи серверного обладнання прямо залежать від умов його експлуатації, зокрема - від ефективного контролю мікрокліматичних параметрів. У зв'язку з цим на ринку з'явилася значна кількість апаратно-програмних рішень, призначених для дистанційного моніторингу та керування кліматичними умовами.

У цьому розділі проаналізовано існуючі рішення, що використовуються в практиці, класифіковано їх за основними критеріями, а також окреслено їхні переваги й недоліки в контексті реалізації ефективної системи автоматизованого клімат-контролю [4].

Системи контролю клімату умовно можна поділити на два основні типи:

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

комерційні індустріальні рішення, які здебільшого застосовуються у великих дата-центрах і корпоративних мережах, та самостійно зібрані (DIY) або адаптовані рішення на основі відкритих апаратних платформ, які є економічно доцільними для малих і середніх підприємств, а також приватного використання. Кожен тип має свою специфіку, набір функціональних можливостей, вартісні характеристики та рівень складності у впровадженні.

Серед промислових апаратно-програмних рішень, які зарекомендували себе на ринку, варто згадати системи типу APC NetBotz, Siemens Desigo, Schneider Electric EcoStruxure, Emerson Liebert, а також рішення від компаній АКСП і Ubiquiti. Наприклад, система APC NetBotz є багатофункціональним середовищем моніторингу, здатним контролювати температуру, вологість, рівень вібрації, наявність диму, руху, а також вести відеоспостереження. Вона підтримує інтеграцію з системами управління живленням та вентиляцією, має інтерфейси SNMP і HTTP, що забезпечує її гнучке налаштування та інтеграцію з іншими інструментами управління інфраструктурою [5].

Siemens Desigo - це модульна система управління кліматом, яка призначена не лише для серверних приміщень, але й для будівель в цілому. Вона передбачає автоматичне регулювання температури, вентиляції, кондиціонування, а також взаємодію з датчиками в реальному часі. У разі необхідності система автоматично перебудовує кліматичні параметри згідно з наперед заданими сценаріями або змінами зовнішніх умов. Така гнучкість дозволяє суттєво знизити споживання енергії та оптимізувати роботу обладнання.

Система EcoStruxure від Schneider Electric також є комплексним рішенням для моніторингу інфраструктури. Вона дає змогу збирати дані з численних датчиків, аналізувати їх за допомогою вбудованого штучного інтелекту, а також реагувати на зміни в режимі реального часу. Основною перевагою цієї системи є її масштабованість і підтримка віддаленого управління через хмарні сервіси, що забезпечує високий рівень автоматизації процесів.

Emerson Liebert пропонує прецизійні кондиціонери та програмне

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

забезпечення для їхнього управління, призначене спеціально для дата-центрів. Ці рішення мають вбудовані датчики температури й вологості, підтримку інтерфейсів Modbus, BACnet, SNMP, що дозволяє легко інтегрувати систему в загальну інфраструктуру підприємства. Вони також здатні виявляти перевантаження обладнання, вчасно вмикати/вимикати охолодження або подавати сигнал тривоги [6].

Рішення АКСП базуються на IP-сенсорах, що поєднують можливості контролю клімату, відеоспостереження та сигналізації. Однією з особливостей цих пристроїв є простота розгортання - вони не потребують складної інфраструктури, натомість забезпечують швидке підключення, інтеграцію з браузером та можливість створення сценаріїв реакції.

Компанія Ubiquiti, відома у сфері мережевого обладнання, також пропонує продукти для моніторингу середовища - наприклад, UniFi Protect з функцією контролю температури і вологості в комбінації з відеоспостереженням. Проте вартість таких рішень, а також необхідність професійного встановлення й налаштування робить їх менш привабливими для невеликих серверних кімнат у невеликих компаніях.

З іншого боку, з огляду на доступність і гнучкість апаратних платформ, дедалі більшого поширення набувають рішення на базі Arduino, Raspberry Pi, ESP32, STM32 та інших мікроконтролерів із відкритою архітектурою. Такі рішення дозволяють реалізувати персоналізовані системи моніторингу з урахуванням конкретних потреб, обмежень бюджету та індивідуальних умов приміщення.

Arduino - одна з найпопулярніших платформ для створення простих і середньо складних кліматичних систем. Завдяки широкій підтримці сенсорів температури (DS18B20, DHT11, DHT22), вологості, газів (MQ-серія), якості повітря, світла тощо, вона дозволяє будувати гнучкі системи з можливістю розширення. Користувачі можуть створювати власні інтерфейси моніторингу за допомогою OLED-дисплеїв, LCD-панелей або виведення даних на комп'ютер

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

через USB або Wi-Fi модулі (ESP8266, ESP32). Програмне забезпечення розробляється на мові C/C++ із використанням середовища Arduino IDE, що є зручним навіть для початківців [7].

Raspberry Pi надає значно більшу обчислювальну потужність, можливість роботи з графічним інтерфейсом, запуску веб-серверів, збереження історії кліматичних змін у базах даних (наприклад, SQLite, MySQL), створення графіків, звітів і організації сповіщення через Telegram, email або SMS. Завдяки підтримці операційної системи Linux та великій кількості бібліотек, Raspberry Pi дозволяє реалізувати повноцінну систему управління, яка може працювати автономно або з підключенням до хмарних сервісів (Google Cloud, AWS, ThingSpeak тощо) [8].

Також широке застосування отримали Wi-Fi мікроконтролери ESP8266 та ESP32, які завдяки невеликій вартості, компактності та підтримці бездротових технологій дозволяють створювати модульні, енергоефективні та автономні системи. Наприклад, за допомогою ESP32 можна побудувати систему, яка вимірює температуру, вологість і CO₂ у реальному часі, передає дані на веб-сервер, керує вентиляторами або кондиціонерами, а також надсилає повідомлення про критичні значення. Додатково можна застосувати MQTT-протокол для обміну даними між пристроями або інтеграції з мобільними застосунками типу Home Assistant.

Система включає сенсори для вимірювання температури (позначені як 1), освітленості (2) та вологості (3). Вихідні сигнали від цих датчиків надходять на інформаційно-вимірювальний блок із мікроконтролером (4), який підключено до послідовного порту (5) комп'ютера (6). Паралельний порт комп'ютера (7) через інтерфейс зв'язку (8) з'єднаний із блоком управління (9). До виходів цього блоку підключені виконавчі пристрої: освітлення робочого місця (10), обігрівач (11), зволожувач (12) та охолоджувач повітря (13). Таким чином забезпечується автоматизоване регулювання кліматичних параметрів у виробничому приміщенні [9]. У цій системі всі дані, отримані від датчиків, передаються до комп'ютерного модуля, де здійснюється їх обробка та аналіз.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

На основі отриманих результатів формується відповідний набір керуючих сигналів, які регулюють подальшу роботу пристроїв. Обробка зібраної інформації виконується мікроконтролером, який забезпечує аналіз даних і формування керуючих сигналів, що надходять на вторинний прилад.

Пристрій складається з п'яти незалежних вимірювальних модулів (позначених як 1), а також модулів управління (2) та обробки інформації (3). Для забезпечення бездротового зв'язку між вимірювальними модулями і модулями керування й обробки даних використані радіоадаптери безпроводної мережі (16). Це дозволяє одночасно підключати всі вимірювальні пристрої через радіоканал, що значно розширює сферу застосування системи і полегшує її впровадження.

Електронний аналізатор параметрів мікроклімату складається з датчиків температури (1), атмосферного тиску (2), вологості повітря (3) і освітленості (4), які підключені до модуля обробки даних (5). Цей модуль обладнаний конвертером і пристроєм формування пакетів даних (6), взаємодіє з блоком ідентифікаторів (7) через двопозиційні перемикачі (8), а також містить радіочастотний модуль (9) і шину передачі даних (10). Усі компоненти живляться від блоку живлення (11), який управляється контролером живлення (12) і підключений до зовнішнього джерела через порт Micro-USB (13), забезпечуючи централізовану і стабільну подачу живлення.

Для вимірювання температури використовувалися сенсори на основі терморезистора DHT11. Ця конструкція відзначається простотою реалізації та вигідною вартістю, проте не підходить для застосування в серверних комплексах через відсутність контролю вологості повітря.

Загалом, можна дійти висновку, що системи контролю мікрокліматичних параметрів приміщень за своєю архітектурою цілком придатні для застосування у серверних кімнатах. Вони зазвичай базуються на мікроконтролерах або мікропроцесорах і використовують різноманітні типи датчиків, які обираються залежно від конкретних завдань, а також від розмірів і конфігурації серверного

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

простору. Незалежно від конкретної комплектації, моніторинг вологості та температури повітря є обов'язковим у будь-якій такій системі [10].

Розроблена система моніторингу параметрів мікроклімату серверної кімнати реалізована з використанням популярного безкоштовного програмного забезпечення з відкритим кодом - Home Assistant. Інтеграція всіх датчиків та розумних пристроїв у єдиний інтерфейс управління і моніторингу з можливістю доступу як через веб-платформу, так і через мобільний додаток робить використання Home Assistant перспективним рішенням для створення автоматизованих систем контролю клімату в серверних кімнатах.

Проте такий підхід вимагає застосування високотехнологічних датчиків та постійного зв'язку з платформою Home Assistant, що інколи може бути не найоптимальнішим варіантом при розробці кліматичних систем.

На вітчизняному ринку кліматичних систем для серверних приміщень представлені кілька компаній. Зокрема, компанія "Кластер" пропонує систему моніторингу, яка здійснює контроль температури та вологості повітря. У разі виходу показників за допустимі межі система автоматично надсилає SMS-повідомлення одному із відповідальних користувачів, інформуючи про стан температури та вологості у серверній. Ця система також підтримує керування кондиціонерами та інтеграцію із звуковими оповіщувачами [11].

Таким чином, на ринку існує широкий спектр апаратно-програмних рішень для контролю клімату в серверних приміщеннях, від повнофункціональних промислових систем до недорогих адаптивних платформ на базі мікроконтролерів. Усі вони мають спільну мету - забезпечити стабільну роботу серверного обладнання в безпечних мікрокліматичних умовах, мінімізуючи ризики перегріву, надмірної вологості або накопичення пилу. Правильний вибір платформи та її відповідна інтеграція в інфраструктуру є основою для надійного та ефективного функціонування серверної кімнати.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

1.3 Протоколи передачі даних у системах моніторингу

Ефективне функціонування автоматизованої системи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні безпосередньо залежить від якісного та надійного обміну даними між елементами цієї системи. Комунікація між сенсорами, мікроконтролерами, виконавчими пристроями та програмним забезпеченням забезпечується за допомогою протоколів передачі даних, які регламентують формат повідомлень, спосіб їхньої маршрутизації, обробки помилок і забезпечення цілісності переданої інформації. Саме протоколи визначають, наскільки швидко та точно інформація буде доставлена від джерела до приймача, з якою затримкою, у якому обсязі, та які засоби захисту при цьому будуть застосовані.

У контексті розробки кліматичних систем для серверних приміщень необхідно враховувати низку факторів при виборі комунікаційного протоколу: архітектура мережі, обсяг передаваних даних, наявність або відсутність інтернет-з'єднання, енергоспоживання, стійкість до завад, масштабованість, простота реалізації та сумісність із використовуваним обладнанням. У цьому розділі розглянуто найпоширеніші протоколи, які використовуються в системах моніторингу кліматичних параметрів, зокрема UART, I2C, SPI, Modbus, MQTT, HTTP, SNMP, а також окремо висвітлено переваги та недоліки кожного з них у контексті побудови автоматизованих рішень [12].

Одним із базових способів передачі даних між елементами в межах однієї плати або пристрою є використання протоколу UART (Universal Asynchronous Receiver-Transmitter). Цей протокол є асинхронним, тобто не потребує окремого тактового сигналу. Передача даних відбувається по двох дротах: один для передачі (TX), інший - для прийому (RX). Він дуже поширений в Arduino-проектах і використовується для підключення сенсорів, модемів, дисплеїв та інших пристроїв. Перевагами UART є простота реалізації, низьке

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

енергоспоживання, наявність великої кількості підтримуваних пристроїв. Недоліком є обмеження по довжині лінії (до 15 метрів без спеціальних буферів) і відсутність можливості підключення багатьох пристроїв на одну шину без додаткової логіки.

Для підключення кількох датчиків до одного мікроконтролера часто використовується протокол I2C (Inter-Integrated Circuit), який дозволяє передавати дані по двох дротах (SDA – дані, SCL – тактова лінія). Він підтримує до 127 пристроїв на одній шині, що робить його дуже привабливим для компактних систем моніторингу. I2C широко застосовується в сенсорах температури, вологості, тиску, а також в інтегрованих модулях (Таблиця 1.2).

Таблиця 1.2 - Протоколи передачі даних у системах моніторингу

№	Назва протоколу	Тип з'єднання	Переваги	Недоліки	Сфера використання
1	MQTT	TCP/IP	Низьке енергоспоживання, QoS, відмінна масштабованість	Вимагає брокера, складніший у налаштуванні	ІоТ, сенсорні мережі, телеметрія
2	HTTP/HTTPS	TCP/IP	Простота реалізації, універсальність	Високе навантаження, відсутність подій (pull-модель)	Веб-сервіси, REST API
3	WebSocket	TCP/IP	Двостороннє з'єднання в реальному часі	Складніша обробка помилок, Не завжди підтримується браузером	Онлайн-інтерфейси, живі графіки,
4	CoAP	UDP	Легкий протокол, оптимізований для ІоТ	Менше надійності, не підтримується всіма платформами	Безпроводні сенсорні мережі, constrained devices

Продовження таблиці 1.2

5	FTP/SFTP	TCP/IP	Передача великих файлів, підтримка авторизації	Не підходить для реального часу	Архіви, резервне копіювання
6	LoRaWAN	Радіоканал	Дальність до 15 км, дуже низьке енергоспоживання	Низька швидкість передачі, висока затримка	Сільське господарство, моніторинг довкілля
7	Zigbee	Мережа IEEE 802.15.4	Меш-мережа, надійність, енергоефективність	Низька пропускну здатність	Розумні будинки, локальні мережі датчиків
8	Modbus TCP	Ethernet	Промислова надійність, простота інтеграції	Обмежена підтримка складних типів даних	Промислові системи, SCADA
9	SNMP	TCP/UDP	Стандартизований моніторинг мережевого обладнання	Складна структура об'єктів, затримки	ІТ-інфраструктура, мережевий моніторинг
10	AMQP	TCP/IP	Безпечна доставка, черги повідомлень	Високі вимоги до ресурсів сервера	Банківські системи, телекомунікації

Кінець таблиці 1.2

11	Bluetooth LE	Радіоканал	Дуже низьке енергоспоживання, підтримка мобільних пристроїв	Обмежена дальність, нестабільне з'єднання	Wearable пристрої, особисті сенсорні мережі
12	CAN bus	Спеціалізована шина	Висока швидкість, безпечний обмін у реальному часі	Висока складність реалізації в системах	Автоматика, транспорт, вбудовані системи

Протоколи класифікуються за типом з'єднання (TCP/IP, UDP, радіоканал, спеціалізована шина), а також оцінюються за основними перевагами, недоліками й характерними сферами застосування. Окрему увагу приділено MQTT, як найбільш придатному для систем реального часу з мікроконтролерами. Його підтримка QoS-рівнів і мінімальний трафік роблять його ідеальним для передавання сенсорних даних. Протоколи типу HTTP та WebSocket підходять для візуалізації в браузерях, але менш ефективні в умовах обмежених ресурсів або при нестабільному з'єднанні [13].

Таким чином, протоколи передачі даних є фундаментальною складовою автоматизованої системи моніторингу та управління кліматом у серверному приміщенні.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

2. ПРОЕКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ ДИСТАНЦІЙНОГО КОНТРОЛЮ ТА УПРАВЛІННЯ

2.1 Архітектура системи та вибір апаратного забезпечення

Проектування автоматизованої системи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні передбачає створення такої архітектури, яка забезпечить безперервний збір, обробку та передачу інформації від сенсорів до кінцевого користувача. Система має бути надійною, масштабованою, енергоефективною та зручною в експлуатації. Її архітектура повинна враховувати особливості серверного середовища, а також вимоги до точності вимірювань, швидкості реагування, захищеності даних і підтримки резервування. Відповідно до загальноприйнятої структури подібних систем, проєкт поділяється на три основні рівні: фізичний (рівень сенсорів і виконавчих пристроїв), логічний (рівень контролерів і передавальних модулів), та інформаційний (рівень збереження, обробки, візуалізації та керування даними). У цьому розділі буде детально описано загальну архітектуру системи, вибір і призначення кожного апаратного модуля, взаємозв'язки між елементами, а також представлено опис кількох діаграм, що ілюструють функціонування системи [14].

Діаграма «Загальна архітектура системи моніторингу»

Опис: на рисунку 2.1 діаграма демонструє три основні рівні системи - сенсорний (вхідні дані), логічний (обробка), інформаційний (керування і візуалізація). Вказані основні компоненти: датчики, мікроконтролер ESP32, модулі зв'язку, реле, веб-сервер, інтерфейс користувача.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 2.1 - Діаграма «Загальна архітектура системи моніторингу»

Діаграма «Схема розміщення сенсорів у серверному приміщенні» Опис: відображає просторове розміщення сенсорів температури, вологості, CO₂, диму тощо у приміщенні з урахуванням зон теплового навантаження. Показано сенсори під фальшпідлогою, у повітроводах і біля серверних стійок (рис 2.2).

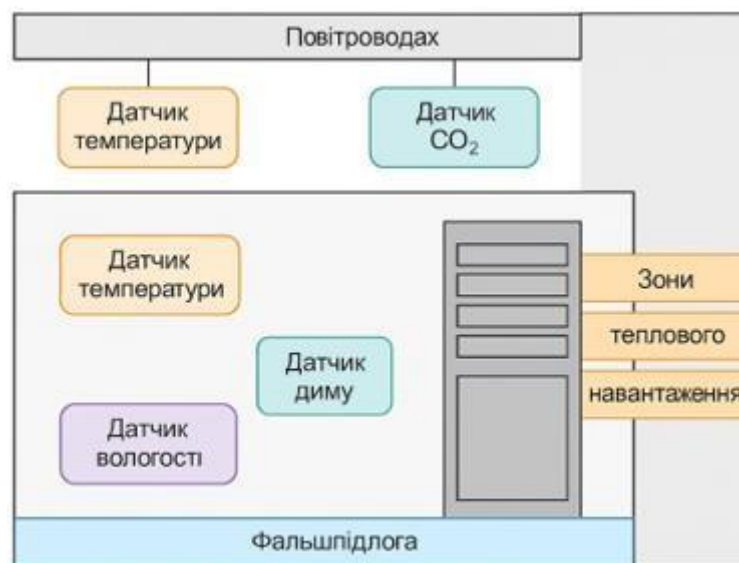


Рисунок 2.2 - Діаграма «Схема розміщення сенсорів у серверному приміщенні»

Логічний рівень системи забезпечує обробку даних із сенсорів, реалізацію логіки реагування, а також організацію зв'язку з інформаційним рівнем. Центральним елементом на цьому етапі є вибір мікроконтролера. Враховуючи потребу в бездротовому зв'язку, обмежене енергоспоживання та підтримку інтерфейсів I2C, SPI, UART, було обрано ESP32 як основну апаратну платформу.

Цей мікроконтролер має двоядерний процесор, вбудований Wi-Fi і Bluetooth, високу обчислювальну потужність, а також підтримку багатозадачності, що дозволяє реалізувати як опитування сенсорів, так і передачу даних на сервер без затримок. Його розміщення відображено на діаграмі "Функціональне підключення контролера ESP32", де чітко зазначені всі з'єднання з сенсорами, вихідними пристроями та модулями зв'язку.

Окрім ESP32, можливою альтернативою у варіантах розширеної системи є використання Raspberry Pi для розгортання локального веб-сервера, бази даних та графічного інтерфейсу управління. У цьому випадку ESP32 виступає як клієнт, що передає дані через MQTT-протокол або HTTP-запити до Raspberry Pi, який виконує роль сервера [15].

Діаграма «Функціональне підключення контролера ESP32» Опис: ілюструє, як до ESP32 підключені всі сенсори (через інтерфейси I2C, 1- Wire, аналогові входи), реле-модулі, модулі живлення та бездротового зв'язку. Показано логіку передачі даних у реальному часі на рисунку 2.3.



Рисунок 2.3 - Діаграма «Функціональне підключення контролера ESP32»

Діаграма «Керування кліматичним обладнанням через реле» Опис: зображає, як ESP32 подає сигнали на реле для керування охолоджувачем, вентиляцією, зволожувачем чи осушувачем (рис 2.4). Зазначено логіку ввімкнення/вимкнення залежно від порогових значень [16].

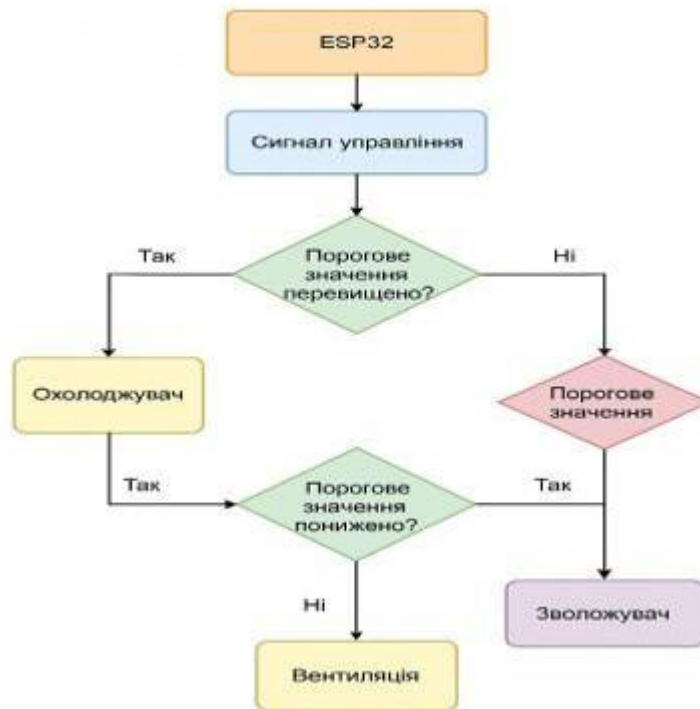


Рисунок 2.4 - Діаграма «Керування кліматичним обладнанням через реле»

На рівні виконавчих пристроїв передбачається використання реле-модулів, які дозволяють вмикати або вимикати системи вентиляції, охолодження, зволоження чи осушення повітря. У разі перевищення встановлених температурних чи вологісних порогів контролер подає сигнал на відповідне реле, яке вмикає охолоджувач або витяжку [17].

Інформаційний рівень системи реалізується шляхом створення веб-сервера для візуалізації поточних значень параметрів, журналювання змін і генерації повідомлень про тривожні події. Пропонується реалізація інтерфейсу за допомогою HTML/CSS/JavaScript, доступного через браузер будь-якого пристрою в локальній мережі. Дані від ESP32 надходять через HTTP-запити або MQTT-повідомлення до веб-сервера, що зберігає їх у локальній базі даних (наприклад, SQLite або MySQL). Для візуалізації використовується бібліотека Chart.js або аналогічний інструмент, який дозволяє будувати графіки зміни температури й вологості в режимі реального часу. Показано, як ESP32 надсилає дані на Raspberry Pi через MQTT або HTTP, а Raspberry Pi зберігає, обробляє та передає інформацію на веб-інтерфейс на рисунку 2.5.

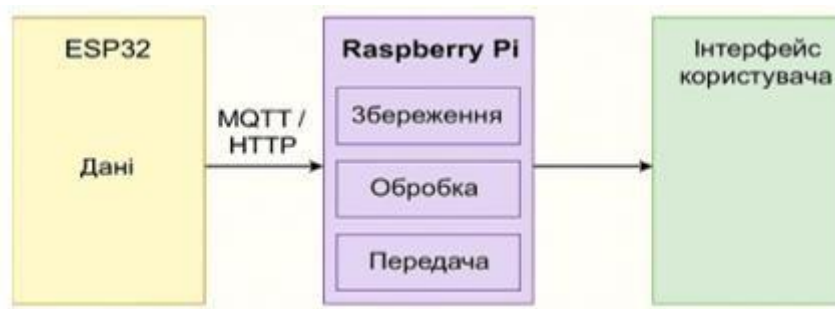


Рисунок 2.5 - Діаграма «Архітектура клієнт-серверної взаємодії між ESP32 і Raspberry Pi»

Діаграма «Алгоритм реагування на критичні показники» Опис: логічна діаграма, на рисунку 2.6 описує послідовність дій системи при перевищенні порогових значень - від отримання даних сенсором до надсилання повідомлення адміністратору та активації реле [18].



Рисунок. 2.6 - Діаграма «Алгоритм реагування на критичні показники»

Діаграма «Масштабована архітектура системи з кількома ESP32»

Опис: представлено систему з кількома вузлами ESP32 у різних зонах серверного простору. Показано передачу даних кожного вузла на спільний MQTT-брокер і централізований сервер обробки.



Рисунок 2.7 - Діаграма «Масштабована архітектура системи з кількома ESP32»

Передбачено також реалізацію механізму сповіщення: при досягненні критичних параметрів система надсилає повідомлення користувачу через Telegram-бота, email або SMS (через GSM-модуль або веб-сервіс). Це дозволяє оперативно реагувати на несправності в охолодженні або інші нештатні ситуації. Сценарій дій при перевищенні порогу показано на діаграмі "Алгоритм реагування на критичні показники", де вказано, як система переходить із режиму моніторингу в режим активного управління, надсилає повідомлення та активує виконавчі механізми [19].

Таким чином, архітектура автоматизованої системи моніторингу клімату в серверному приміщенні базується на поєднанні апаратних модулів, мікроконтролерів, протоколів передачі даних і програмних інтерфейсів. Кожен компонент відіграє чітко визначену роль у загальній схемі: сенсори - вимірюють параметри, контролер - обробляє інформацію та передає її на сервер, програмне забезпечення - візуалізує, керує та сповіщає. Запропоновані діаграми дозволяють наочно уявити логіку роботи системи, її структуру та принципи взаємодії

компонентів. Усі вибрані апаратні рішення є доступними на ринку, мають підтримку відкритих бібліотек і документації, що забезпечує зручність розробки, та подальшої експлуатації системи. Завдяки модульності запропонована система легко адаптується до нових вимог, дозволяє розширення функціоналу, а також забезпечує високу надійність і точність контролю кліматичних умов у критично важливих середовищах.

Для створення прототипу пристрою, призначеного для контролю мікрокліматичних умов у серверних приміщеннях, було обрано датчик температури і вологості DHT11 (рис 2.8). Цей сенсор широко поширений та доступний і служить для визначення відносної вологості та температури повітря, поєднуючи ємнісний температурний датчик із гігрометром [20].

Обрання саме цього датчика зумовлене його надійністю, доступністю на ринку та здатністю забезпечувати точні вимірювання основних параметрів мікроклімату, що робить його оптимальним елементом для реалізації системи моніторингу.



Рисунок 2.8 - Зовнішній вигляд датчика температури і вологості DHT11

У таблиці 2.1. наведено ключові технічні характеристики, які відігравали вирішальну роль під час вибору датчика для розробки прототипу пристрою моніторингу мікроклімату, а також стали основою для подальшого проектування і конструювання системи. Ці параметри охоплюють важливі аспекти, що визначають функціональність, точність і надійність сенсора в умовах роботи серверної кімнати [21].

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

Таблиця 2.1 - Основні технічні властивості датчика температури і вологості
DHT11

Напруга живлення	3,5V ... 5,5V
Точність вимірювання відносної вологості	+/- 5%(максимальна похибка)
Точність вимірювання температури	+/- 2% (максимальна похибка)
Ефективні межі вимірювання вологості	5%-95% RH
Ефективні межі вимірювання температури	-20°C ~ 60°C
Споживання струму	2,5 мА (максимальне значення при генерації даних)
Частота вимірювань	1 Гц (1 вимір кожену секунду)
Кількість конекторів	4
Розміри	15,5мм * 12мм * 5,5мм

Датчик температури та вологості DHT11 має чотири виводи (коннектори), кожен із яких відповідає за певну функцію у роботі пристрою. Кожен контакт виконує важливу роль у забезпеченні правильного живлення датчика, передачі даних та забезпеченні стабільного зв'язку з контролером.

У таблиці 2.2 детально описані функціональні призначення кожного з цих конекторів, а також наведені рекомендації щодо їх підключення для забезпечення коректної роботи датчика. Знання і правильне використання кожного виводу є ключовим елементом при розробці схеми пристрою, адже помилки у підключенні можуть призвести до некоректного зчитування даних або навіть виходу сенсора з ладу [22].

Розуміння структури підключення і функцій кожного виводу дає змогу

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

більш ефективно інтегрувати DHT11 у складніші системи контролю мікроклімату та забезпечити надійність і стабільність роботи всієї системи в цілому.

Таблиця 2.2 - Опис конекторів датчика DHT11

Назва конектора	Призначення
VCC	Живлення
GND	Земля
DATA	Дані (вимірювані значення)
NC	Не підключений

Отже, серед різноманіття доступних датчиків для вимірювання температури і вологості у серверних приміщеннях, датчик DHT11 було обрано завдяки його численним перевагам. Він поєднує в собі достатню точність вимірювань, простоту використання, широкодоступність та низьку ціну, що робить його оптимальним рішенням для забезпечення надійного контролю мікроклімату у таких умовах. Завдяки цим характеристикам, DHT11 є зручним і ефективним інструментом для впровадження в автоматизовані системи моніторингу кліматичних параметрів серверних кімнат [23].

Для реалізації автоматизованої системи контролю мікроклімату у серверних приміщеннях обрано мікроконтролер Arduino Uno, побудований на базі мікросхеми ATmega328. Ця апаратна платформа пропонує 14 цифрових входів/виходів, з яких 6 можуть виконувати функції широтно-імпульсної модуляції (ШІМ). Також Arduino Uno оснащений 6 аналоговими входами, USB-портом для підключення до комп'ютера, роз'ємом живлення, кварцовим генератором частотою 16 МГц, інтерфейсом ICSP та кнопкою скидання.

Для стабільної роботи системи мікроконтролер необхідно підключити до комп'ютера через USB-кабель або живити через відповідний роз'єм живлення. Вбудований стабілізатор напруги забезпечує роботу пристрою в діапазоні від 6

до 20 В. На рисунку 2.9 наведено зображення мікроконтролера Arduino Uno.



Рисунок 2.9 - Зовнішній вигляд мікроконтролера Arduino UNO

У таблиці 2.3. наведено ключові технічні характеристики мікроконтролера, які мали вирішальне значення під час його відбору для реалізації проекту, а також були враховані на етапах подальшого проектування та конструювання системи. Ці параметри охоплюють широкий спектр аспектів, що впливають на функціональність, продуктивність, надійність та ефективність роботи пристрою в цілому [24].

Таблиця 2.3 - Технічні характеристики Arduino Uno

Параметр	Специфікація
Чіп	ATmega168
Діюча напруга	5V
Рекомендована напруга	7-12V
Максимальні межі напруги	6-20V
Кількість цифрових пінів	14
Кількість аналогових пінів	6
Flash пам'ять	32 KB (ATmega328)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Частота	16 MHz

Мікроконтролер ESP8266 відзначається простотою у використанні та можливістю передавати дані через Wi-Fi, що робить його надзвичайно зручним вибором для систем моніторингу мікроклімату. Вбудований бездротовий модуль забезпечує легке підключення до мережі, що дозволяє здійснювати віддалений контроль та керування пристроєм без зайвих складнощів. Зовнішній вигляд ESP8266 наведено на рисунку 2.10.



Рисунок. 2.10 - Зовнішній вигляд мікроконтролера ESP8266

У таблиці 2.4. докладно викладено ключові технічні параметри мікроконтролера, які мали принципове значення при його виборі для реалізації проєкту, а також слугували орієнтиром під час подальшого етапу розробки та конструювання системи. Ці характеристики охоплюють різні аспекти, що визначають функціональні можливості, ефективність і надійність роботи пристрою в конкретних умовах експлуатації [25].

Особливу увагу було приділено таким параметрам, як кількість і тип доступних входів/виходів, підтримка бездротових технологій, енергоспоживання, швидкодія процесора, а також обсяг оперативної та постійної пам'яті. Важливими також були розміри мікроконтролера, тип корпусу, а також сумісність з іншими компонентами електронної системи.

Для реалізації проєкту було ретельно підібрано та обґрунтовано комплектуючі, які найкраще відповідають поставленим технічним вимогам і завданням.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33

Таблиця 2.4 - Технічні характеристики ESP32

Параметр	Специфікація
Бітність	32 bit
Процесор	TenSilica L 106
Частота Процесора	80 MHz-160MHz
Оперативна пам'ять	36 Кб
Фізична пам'ять	16 Мб
Вбудований WiFi	2,4GHz підтримує 802,11 b/g/n
Кількість ADC пінів	1 (10 біт розширення)
Кількість GPIO пінів	10
Діюча напруга	3,0V-3,6V
Споживання електроенергії	80mA (середнє значення)
Межі оптимальної температури	-40°C - 125°C

Датчик DHT11 було обрано через його високу ефективність, доступність на ринку та здатність точно вимірювати основні параметри мікроклімату. Завдяки вбудованому аналого-цифровому перетворювачу і простому протоколу передачі даних, цей сенсор забезпечує стабільні та достовірні показники. Його характеристики ідеально відповідають вимогам системи моніторингу, гарантуючи необхідну точність та надійність у роботі.

Модуль ESP8266 відзначається своєю компактністю та економічністю. Головною перевагою цього мікроконтролера є підтримка бездротової передачі даних через Wi-Fi, що відповідає вимогам віддаленого моніторингу та керування системою. Відсутність підтримки аналогових датчиків не є суттєвим недоліком у контексті даного проекту, враховуючи численні переваги, які він надає [26].

Таким чином, комплексний підбір компонентів, що враховує їх технічні характеристики, сумісність та практичність використання, створює надійну

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

основу для успішної реалізації системи моніторингу мікроклімату серверних кімнат. Це забезпечує не лише точність і стабільність вимірювань, а й ефективність управління, що є критично важливим для підтримки оптимальних умов роботи серверного обладнання.

2.2 Розробка логіки взаємодії модулів системи

У процесі розробки автоматизованої системи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні важливу роль відіграє чітке формалізоване визначення логіки взаємодії всіх її компонентів. Це забезпечує узгоджену роботу сенсорних модулів, мікроконтролерів, виконавчих пристроїв, каналів зв'язку та програмного забезпечення. Для візуалізації логіки взаємодії елементів системи доцільно використовувати різні типи діаграм, такі як блок-схеми, діаграми потоків даних, діаграми станів, часові діаграми, послідовнісні діаграми та діаграми прецедентів. Ці графічні моделі дозволяють описати послідовність дій пристроїв, умови активації алгоритмів, характер реагування системи на події, а також взаємодію між окремими модулями - як апаратними, так і програмними [27].

Діаграма «Блок-схема основного циклу роботи системи» відображає загальний цикл, що виконується мікроконтролером ESP32. Алгоритм починається з ініціалізації всіх підключених сенсорів, перевірки стабільності зв'язку та завантаження конфігурацій. Далі відбувається циклічне зчитування значень температури, вологості, рівня CO₂, диму, запиленості.

Діаграма «Діаграма послідовності взаємодії між сенсором і мікроконтролером» показує взаємодію між температурним сенсором (наприклад, DHT22) і ESP32 на рисунку 2.11. Послідовність включає ініціалізацію сенсора, передавання запиту, очікування відповіді, зчитування цифрового сигналу, перевірку CRC, перетворення значень у фізичні одиниці, передавання у змінні програми. У випадку невдалого зчитування діаграма

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

демонструє цикл повторних спроб і вивід повідомлення про помилку в лог-систему на рисунку 2.11.

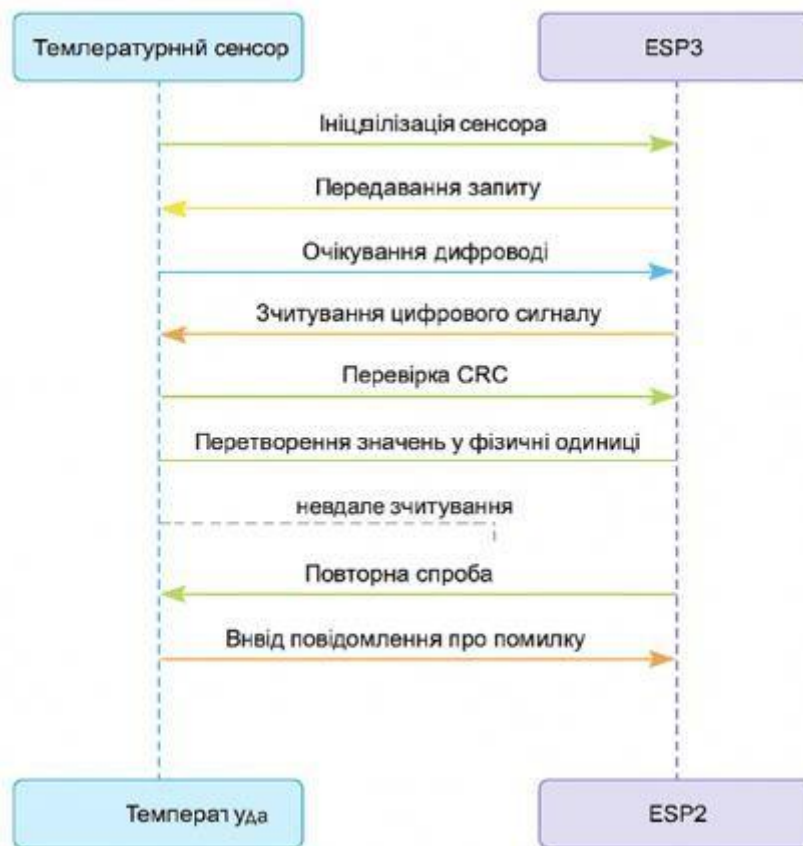


Рисунок 2.11 - Діаграма «Діаграма послідовності взаємодії між сенсором і мікроконтролером»

Діаграма «Діаграма послідовності між ESP32 та реле» моделює обмін командами між контролером і виконавчим модулем на рисунку 2.12. Коли значення температури перевищує порогове, мікроконтролер надсилає логічний сигнал на вхід реле. Реле замикає контакт, через який увімкнений, наприклад, охолоджувач. Після зниження температури нижче допустимого рівня ESP32 вимикає подачу сигналу [28].

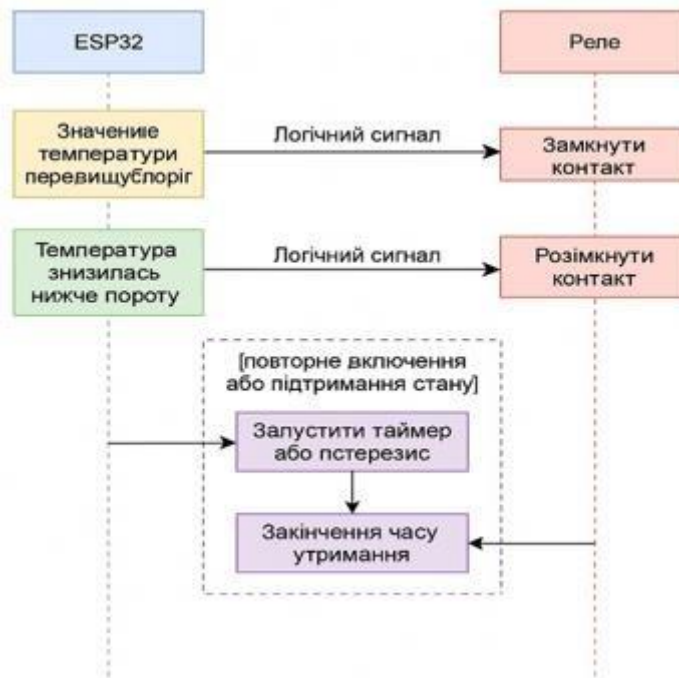


Рисунок. 2.12 - Діаграма «Діаграма послідовності між ESP32 та реле»

Діаграма «Діаграма станів модуля охолодження» описує поведінку вентиляційної або охолоджувальної системи у відповідь на зміни кліматичних параметрів. Стан «Вимкнено» змінюється на «Увімкнено», коли перевищено температуру $>25^{\circ}\text{C}$. Стан повертається до «Вимкнено», коли температура опускається нижче 23°C . У діаграмі враховані перехідні стани: «Затримка на ввімкнення», «Технічне обслуговування», «Аварійне відключення». Вона ілюструє, як на рисунку 2.13 система забезпечує стабільне та безпечне керування охолодженням без частого перемикання режимів.



Рисунок. 2.13 - Діаграма «Діаграма станів модуля охолодження»

Діаграма «Діаграма прецедентів взаємодії користувача з системою» (Use Case Diagram) показує ролі користувачів (адміністратор, оператор) і функції, які вони можуть виконувати через веб-інтерфейс: перегляд поточних значень, зміну порогів тривоги, перегляд історії даних, керування режимами вентиляції, отримання сповіщень. Також на рисунку 2.14 відображено зв'язок із зовнішніми системами: мобільними застосунками, email-сервісами, Telegram-ботом [29].

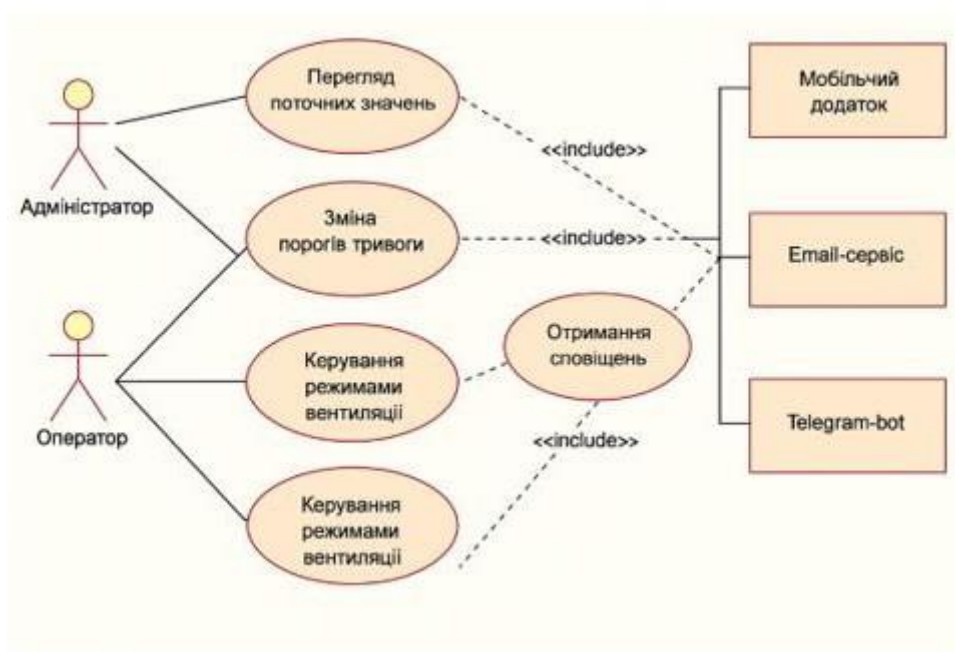


Рисунок. 2.14 - Діаграма «Діаграма прецедентів взаємодії користувача з системою»

Діаграма «Архітектура логічних зв'язків усередині контролера ESP32» показує внутрішні блоки: ініціалізація сенсорів, зчитування даних, логіка порівняння, менеджер реле, Wi-Fi-зв'язок, обробка MQTT, журналювання, обробка помилок. Між блоками зображені стрілки передачі даних, зазначено потоки вхідних і вихідних команд. Така діаграма корисна для розробника з метою оптимізації коду і розподілу функцій між потоками(рис 2.15).



Рисунок 2.15 - Діаграма «Архітектура логічних зв'язків усередині контролера ESP32»

На ілюстрації, позначеній детально зображена структурна схема розглянутого пристрою. Ця схема демонструє основні компоненти та їх взаємозв'язки, що утворюють архітектуру системи. Вона слугує візуальним представленням логіки роботи пристрою, показуючи послідовність обробки сигналів, взаємодію між окремими модулями, а також потоки інформації і керуючих команд[30].

Початковим елементом цієї системи є датчики температури та вологості, які здатні вимірювати температуру повітря в діапазоні від 0 до +60 °C та рівень вологості від 0 до 100%. Вибір цих сенсорів здійснено з урахуванням технічних вимог, викладених у пункті 1.4 першого розділу, що гарантує їх відповідність необхідним параметрам. За потреби номенклатура датчиків може бути розширена, додаючи, наприклад, сенсори пилу, диму, виявлення полум'я та інші спеціалізовані пристрої .

Таким чином, логіка взаємодії модулів системи спроектована таким чином, щоб забезпечити безперервний, адаптивний і надійний процес контролю кліматичних параметрів у серверному приміщенні. Використання різноманітних діаграм дозволяє описати як загальну картину функціонування системи, так і деталізувати кожен її модуль.

2.3 Засоби візуалізації та інтерфейс користувача

У будь-якій сучасній автоматизованій системі дистанційного моніторингу надзвичайно важливим є не лише технічне зчитування та передача кліматичних параметрів, але й їх візуальне подання кінцевому користувачеві. Зручний, функціональний і інтуїтивно зрозумілий інтерфейс дає можливість швидко реагувати на зміну кліматичних умов, відстежувати історичну динаміку показників, налаштовувати пороги тривоги, а також управляти виконавчими пристроями в ручному або автоматичному режимі. В рамках реалізації автоматизованої системи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні передбачено створення веб-інтерфейсу користувача з адаптивним дизайном, доступного через будь-який браузер із локальної мережі або через захищений канал з Інтернету.

Інтерфейс побудовано з використанням сучасних вебтехнологій – HTML5, CSS3, JavaScript, а також фреймворків Bootstrap для адаптивності та Chart.js для візуалізації графіків. Серверна логіка реалізована на платформі Node.js або Flask, з використанням WebSocket або HTTP-запитів для актуалізації даних у реальному часі. Інформація з мікроконтролера ESP32 надходить через MQTT-брокер або REST API. У цьому розділі буде розглянуто логіку побудови інтерфейсу, описано ключові елементи візуалізації, наведено макети та описані можливі скріншоти функціональних модулів панелі керування [31].

На рисунку 2.16 на головній сторінці веб-інтерфейсу користувач бачить центральну панель моніторингу, яка містить блоки відображення поточних кліматичних параметрів у реальному часі.

У правій частині головної панелі знаходиться блок системних повідомлень і тривоги. Він автоматично оновлюється при настанні подій, таких як перевищення температури, відключення сенсора, відсутність зв'язку з ESP32 або втрата живлення.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40



Рисунок 2.16 - Головна сторінка

Веб-ресурс є ключовим інструментом для комплексного моніторингу мікрокліматичних умов у серверних приміщеннях. Завдяки підтримці режиму реального часу, зручному та зрозумілому інтерфейсу, а також широкому набору аналітичних функцій, цей онлайн-сервіс забезпечує ефективний контроль і оптимізацію параметрів середовища, в якому працює серверне обладнання[32].

Для розробки такого сайту моніторингу мікроклімату було обрано комбінацію мов програмування та технологій, що максимально відповідають вимогам проекту, а також сприяють зручності взаємодії з користувачем та продуктивності розробки. Розглянемо докладніше ключові складові цього технічного рішення [33].

Наступним кроком стало детальне порівняння трьох популярних фреймворків, в результаті чого основним вибором став React. Ось основні аргументи на його користь:

1. Ефективність і декларативність. React пропонує сучасний декларативний підхід до створення інтерфейсів. Замість того, щоб вручну управляти оновленнями DOM (Document Object Model), розробник описує, як інтерфейс має виглядати у різних станах, а React автоматично відстежує зміни та вносить необхідні оновлення. Це значно спрощує процес розробки і робить код більш зрозумілим і підтримуваним.

2. Компонентна архітектура. Однією з ключових переваг React є поділ інтерфейсу на незалежні, повторно використовувані компоненти. Кожен компонент можна розробляти, тестувати та оновлювати окремо, що підвищує гнучкість, спрощує пошук помилок і розвиток функціональності, а також допомагає підтримувати чистоту коду.

3. Віртуальний DOM. React використовує концепцію віртуального DOM - внутрішньої копії структури сторінки, яка дозволяє ефективно порівнювати стан інтерфейсу і вносити лише мінімально необхідні зміни до реального DOM. Це забезпечує високу продуктивність та швидкий відгук користувачького інтерфейсу.

4. Однозначність і інтерактивність інтерфейсів. React дає змогу створювати інтуїтивно зрозумілі, логічні та динамічні інтерфейси, що робить навігацію легкою і приємною для користувачів, підвищуючи загальний досвід взаємодії.

5. Активна спільнота та технічна підтримка. React має широку базу користувачів і активну спільноту розробників, що гарантує якісну документацію, регулярні оновлення, а також доступність безлічі додаткових бібліотек і інструментів, які полегшують розробку [33].

6. Підтримка з боку великої компанії. React створений і підтримується компанією Facebook, що забезпечує високий рівень стабільності, постійний розвиток і відповідність сучасним стандартам ринку.

7. Можливості кросплатформної розробки з React Native. React також пропонує потужний інструмент React Native, який дозволяє легко переносити код між веб-інтерфейсом і мобільними додатками. Це значно спрощує розробку кросплатформних рішень і робить React ідеальним вибором для масштабування та подальшого розвитку проекту.

Окрім використання JavaScript та бібліотеки React, важливо впровадити типізацію коду за допомогою TypeScript, що значно спрощує розробку та робить код більш структурованим і чистим. Це забезпечує додатковий імпульс для

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

розвитку веб-платформи моніторингу мікроклімату. Розглянемо основні причини, чому TypeScript став невід’ємною частиною нашого технологічного стеку:

1. Статична типізація. Однією з головних переваг TypeScript є підтримка статичної типізації, яка дозволяє визначати типи змінних, параметрів функцій і їхніх результатів вже на етапі написання коду. Такий підхід допомагає раніше виявляти помилки, підвищуючи стабільність і надійність програмного забезпечення.

2. Покращена зрозумілість коду. Визначення типів робить код більш прозорим і зрозумілим для розробників. Це сприяє швидшому сприйняттю структури програми, полегшує підтримку та розвиток коду, а також допомагає уникнути помилок, пов’язаних з неправильним використанням даних [34].

3. Підтримка сучасних стандартів JavaScript. TypeScript постійно оновлюється і підтримує останні версії ECMAScript, що дозволяє використовувати найсучасніші можливості мови, підвищуючи ефективність розробки і відкриваючи додаткові функціональні перспективи.

4. Інструментарій для розробників. Багато популярних середовищ розробки, таких як Visual Studio Code, пропонують розширену підтримку TypeScript. Це включає автодоповнення, перевірку типів у реальному часі, рефакторинг та інші корисні інструменти, які значно підвищують продуктивність і комфорт програміста.

5. Визначення інтерфейсів. TypeScript дозволяє створювати чіткі інтерфейси, які встановлюють контракти між різними модулями системи, спрощуючи інтеграцію компонентів і забезпечуючи структурованість проекту.

З огляду на всі ці переваги, інтеграція TypeScript у технологічний стек проекту значно покращить якість коду, зробить його більш надійним, гнучким і легким для розуміння [35].

Отже, засоби візуалізації та інтерфейс користувача є ключовими компонентами загальної архітектури автоматизованої системи моніторингу.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

Вони забезпечують не лише доступ до даних, але й зручність керування, ефективно прийняття рішень, оперативне реагування на аварійні ситуації.

2.4 Безпека та надійність системи

Питання безпеки та надійності є критично важливими в автоматизованих системах дистанційного моніторингу кліматичних умов серверного приміщення, оскільки від стабільної роботи таких систем залежить працездатність серверного обладнання, безперервність надання цифрових послуг, а також збереження конфіденційних даних. Порухення цілісності або працездатності системи може призвести до збоїв у роботі серверів, втрати даних, фінансових збитків і зниження довіри користувачів. Тому в процесі розробки і впровадження автоматизованої системи моніторингу особлива увага приділяється впровадженню технічних та програмних засобів захисту, резервування, тестування на надійність, виявленню вузьких місць, а також формуванню комплексної стратегії реагування на позаштатні ситуації. У цьому розділі розглянуто загальні принципи забезпечення безпеки, реалізовані механізми захисту інформації та доступу, методи гарантування надійності системи, результати стрес-тестування, а також представлено серію діаграм і таблиць, які ілюструють результати експериментального тестування системи.

Діаграма «Архітектура захисту системи від несанкціонованого доступу» ілюструє багаторівневу модель безпеки, яка включає авторизацію користувача, шифрування каналу зв'язку, перевірку цифрових підписів, обмеження прав доступу та контроль активності. На рисунку 2.17 відображено, як користувач спочатку проходить автентифікацію через форму входу, потім передає дані через HTTPS або MQTT з TLS, після чого доступ до налаштувань або керування виконавчими пристроями дозволяється лише для авторизованих ролей [36].

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

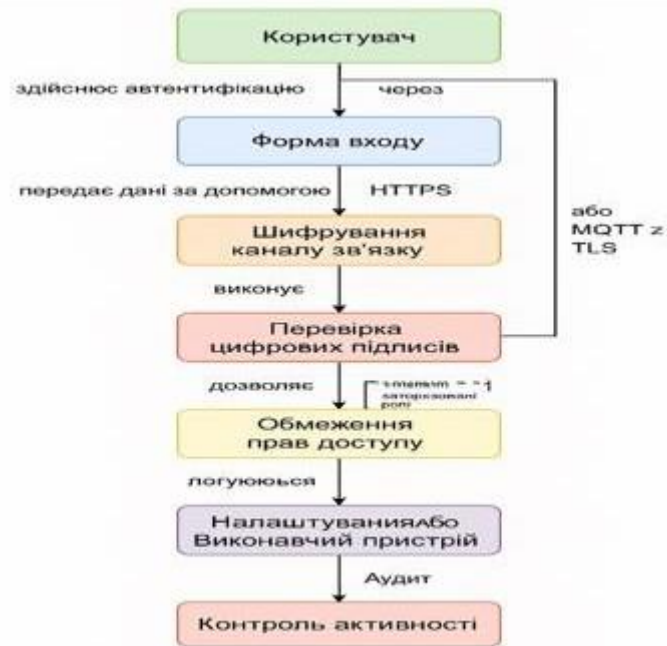


Рисунок 2.17 - Діаграма «Архітектура захисту системи від несанкціонованого доступу»

Діаграма «Сценарії атак та засоби протидії» демонструє типові загрози, які можуть виникати в системі: атака типу "людина посередині", спроба перебору пароля, навмисне створення перевантаження запитами, перехоплення MQTT-повідомлень, виведення з ладу сенсорів. Для кожного сценарію на рисунку 2.18 позначено впроваджені заходи захисту: використання SSL-сертифікатів, обмеження кількості запитів з IP, автоматичне блокування при помилках входу, перевірка цифрового підпису повідомлень, самотестування сенсорів.

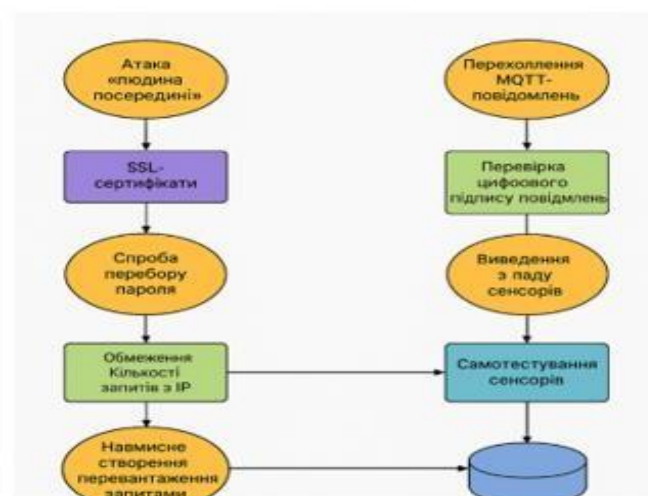


Рисунок 2.18 - Діаграма «Сценарії атак та засоби протидії»

На фізичному рівні передбачено блокування портів доступу до контролера, закритий корпус з пломбами, живлення через ДБЖ. На мережевому рівні – фаєрвол, шифрування MQTT, авторизація за токенами. На прикладному рівні – валідація введених даних, фільтрація запитів, контроль ролей (таблиця 2.5). На рівні користувача – двофакторна автентифікація, журнал дій, розмежування доступу [37].

Таблиця 2.5 - Механізми захисту системи на різних рівнях

Рівень захисту	Тип загрози	Механізм захисту	Ціль	Результат	Коментар
Фізичний	Несанкціонований фізичний доступ	Закритий корпус, пломби, кріплення до стіни	Унеможливити втручання в обладнання	Зменшення ризику саботажу	Підходить для середовищ із обмеженим фізичним доступом
Фізичний	Втрата живлення	Підключення до джерела безперебійного живлення (ДБЖ)	Безперервна робота системи	Робота при відсутності електропостачання до 2 годин	Важливо для критичної інфраструктури
Мережевий	Перехоплення даних	Шифрування MQTT-повідомлень (TLS)	Захист переданих даних	Неможливість аналізу трафіку сторонніми особами	Необхідна генерація SSL-сертифікатів
Мережевий	Несанкціонований доступ до сервера	Авторизація по токenu, фаєрвол	Обмеження доступу	Відсіювання невідомих або підозрілих IP-адрес	Регулярне оновлення правил фаєрволу

Кінець таблиці 2.5

Прикладний	Впровадження помилкових даних	Валідація вхідних форм, перевірка типів, фільтрація символів	Захист від SQL-ін'єкцій і помилок користувача	Зменшення ймовірності помилок та злому	Реалізовано через серверні скрипти
Прикладний	Зловживання правами	Розмежування ролей, контроль доступу до функцій	Захист функціоналу від неавторизованих осіб	Тільки адміністратор може змінювати налаштування системи	Додано рівні: гість, оператор, адміністратор
Користувацький	Компрометація облікового запису	Двофакторна автентифікація (2FA)	Підвищення рівня автентифікації	Зменшення ймовірності злому облікового запису	Реалізовано через email-код або Telegram-підтвердження
Користувацький	Відсутність аудиту дій	Журналювання всіх дій користувачів	Виявлення підозрілих дій	Можливість відстеження змін у системі	Логи зберігаються на сервері до 30 днів

Тестування на надійність системи проводилося в умовах симуляції підвищеного навантаження, втрати зв'язку, перебоїв електропостачання та перевищення граничних значень кліматичних параметрів. Діаграма «Результати тесту на стійкість до втрати зв'язку» демонструє, як система поводить себе при зникненні інтернет-з'єднання: ESP32 продовжує локальний моніторинг, зберігає дані у буфері та надсилає їх на сервер після відновлення з'єднання. На графіку видно, що втрата мережі на 15 хвилин не призвела до втрати даних – після відновлення зв'язку відбулася передача інформації без помилок [38].

Для кожного випадку вказано середній час реакції, кількість втрат даних, чи спрацював механізм сповіщення, чи була виконана дія з керування. Наприклад, для перегріву час реакції – 4,2 с, втрат даних – 0, сповіщення – так, керування – так. Це свідчить про високу надійність модулів(таблиця 2.6).

Таблиця 2.6 - Результати тестування часу реагування системи

Тип події	Середній час реакції	Втрати даних	Сповіщення надіслано	Керуюча дія виконана
Перевищення температури	4,2	0	Так	Так
Критичне зниження температури	4,5	0	Так	Так
Перевищення вологості	5,1	0	Так	Так
Зниження вологості нижче норми	4,9	0	Так	Так
Зникнення сенсора	2,7	1	Так	Ні
Втрата живлення	1,3	0	Так	Так
Відновлення живлення	2,1	0	Так	Так
Втрата з'єднання з сервером	3,6	2	Так	Ні
Відновлення зв'язку	2,4	0	Так	Ні
Активація аварійного режиму	3,9	0	Так	Так

Діаграма «Виконання стрес-тесту з 1000 запитів на хвилину» демонструє, як вебінтерфейс системи витримує навантаження при одночасному доступі великої кількості користувачів або зовнішніх скриптів. На графіку наведено стабільність часу відповіді сервера, споживання оперативної пам'яті та завантаження процесора Raspberry Pi. На рисунку 2.19 результати тесту показують, що при навантаженні до 800 запитів на хвилину система працює

стабільно, далі – зростає затримка, але відмов не спостерігалось.

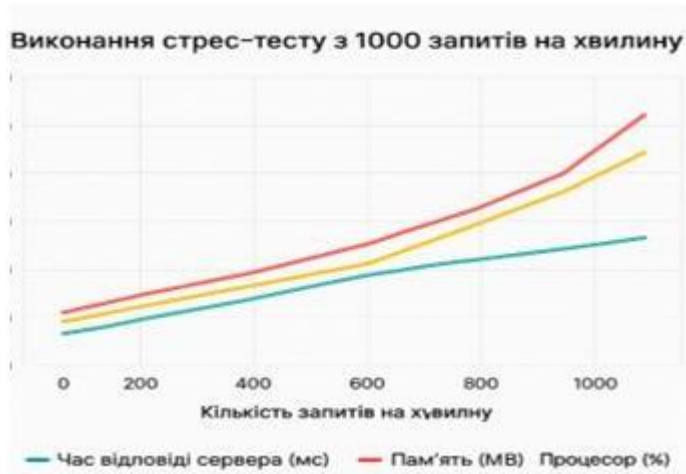


Рисунок 2.19 - Діаграма «Виконання стрес-тесту з 1000 запитів на хвилину»

Наприклад, уразливий компонент – сенсор DHT22, компенсація – дублюючий сенсор і валідація даних. Уразливість: Wi-Fi – компенсація: повторна спроба з'єднання, локальне збереження даних. Уразливість: реле – компенсація: ручне дублювання, таймаут спрацювання (таблиця 2.7).

Таблиця 2.7.- Відмовостійкість компонентів системи

Уразливий компонент	Можлива проблема	Засоби компенсації	Примітка
Сенсор DHT22	Нестабільна робота, відсутність відповіді	Дублюючий сенсор, перевірка CRC, перезапуск модуля	Сигнал дубліката використовується для порівняння
Wi-Fi модуль (ESP32)	Втрата зв'язку	Повторне з'єднання, буферизація даних	Дані зберігаються локально до відновлення каналу
Реле	Механічний збій або залипання	Таймаут спрацювання, сигнал на дублююче реле	Перевірка фактичного стану пристрою
Сервер збереження	Вихід з ладу або перевантаження	Резервне копіювання, подвійна запис у локальний log-файл	Відновлення із копії після збою

Кінець таблиці 2.7

Живлення ESP32	Просадка напруги або вимкнення мережі	Акумуляторне ДБЖ, контроль рівня живлення	Режим енергозбереження активується автоматично
MQTT-брокер	Нестабільна робота або збій процесу	Автоматичне перезавантаження сервісу, моніторинг «watchdog»	Використання стабільної версії Mosquitto
Браузерний інтерфейс	Зависання інтерфейсу або помилки рендеру	Оновлення сторінки по таймеру, кешування важливих даних	Користувач бачить останні коректні показники
RTC-модуль	Зміщення часу або розряд батареї	Перевірка синхронізації з сервером часу NTP	Важливо для коректності журналів і графіків

Таким чином, система розроблена з урахуванням сучасних вимог до безпеки, стійкості та автономності. Застосовані заходи на всіх рівнях – апаратному, мережевому, програмному – забезпечують надійне функціонування навіть в умовах часткових відмов. Запроваджена логіка резервування, діагностики, самовідновлення, шифрування та розмежування доступу гарантує безпеку даних і стійкість до атак. Проведене тестування підтвердило здатність системи витримувати навантаження, швидко реагувати на зміну клімату, зберігати працездатність при втраті зв'язку або живлення.

3. РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1 Розробка програмного модуля збору та обробки даних

У складі автоматизованої системи дистанційного моніторингу кліматичних умов серверного приміщення ключову роль відіграє програмний модуль збору та обробки даних. Саме цей компонент забезпечує регулярне зчитування показників із сенсорів, їхню попередню обробку, перевірку на достовірність, порівняння з пороговими значеннями, збереження до локального буфера або бази даних, а також передачу на наступні рівні - зокрема, у модуль візуалізації або систему сповіщення. Розробка даного модуля базується на принципах надійності, масштабованості, відмовостійкості та енергоефективності, з урахуванням обмежених ресурсів мікроконтролерів. У запропонованому проєкті збір даних реалізовано на базі мікроконтролера ESP32, який поєднує в собі високу продуктивність, велику кількість апаратних портів, підтримку бездротового зв'язку та гнучкість програмної реалізації. Обробка даних здійснюється безпосередньо на мікроконтролері в реальному часі, з подальшою передачею в локальну або віддалену систему за допомогою протоколів MQTT або HTTP. Для кращого розуміння роботи модуля та взаємозв'язку його компонентів у цьому розділі представлено три ключові діаграми: структурну діаграму програмного модуля, діаграму потоків даних і діаграму алгоритму фільтрації та обробки показників [39].

На рисунку 3.1 основні компоненти є: блок ініціалізації (відповідає за запуск сенсорів, Wi-Fi, змінних), блок зчитування даних (опитування датчиків температури, вологості, CO₂, диму), блок перевірки достовірності (фільтрація значень, валідація, перевірка CRC), блок порівняння з пороговими значеннями (активація тривожного режиму або запис до логів), блок передачі даних (формування MQTT-повідомлення або HTTP-запиту), блок буферизації.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

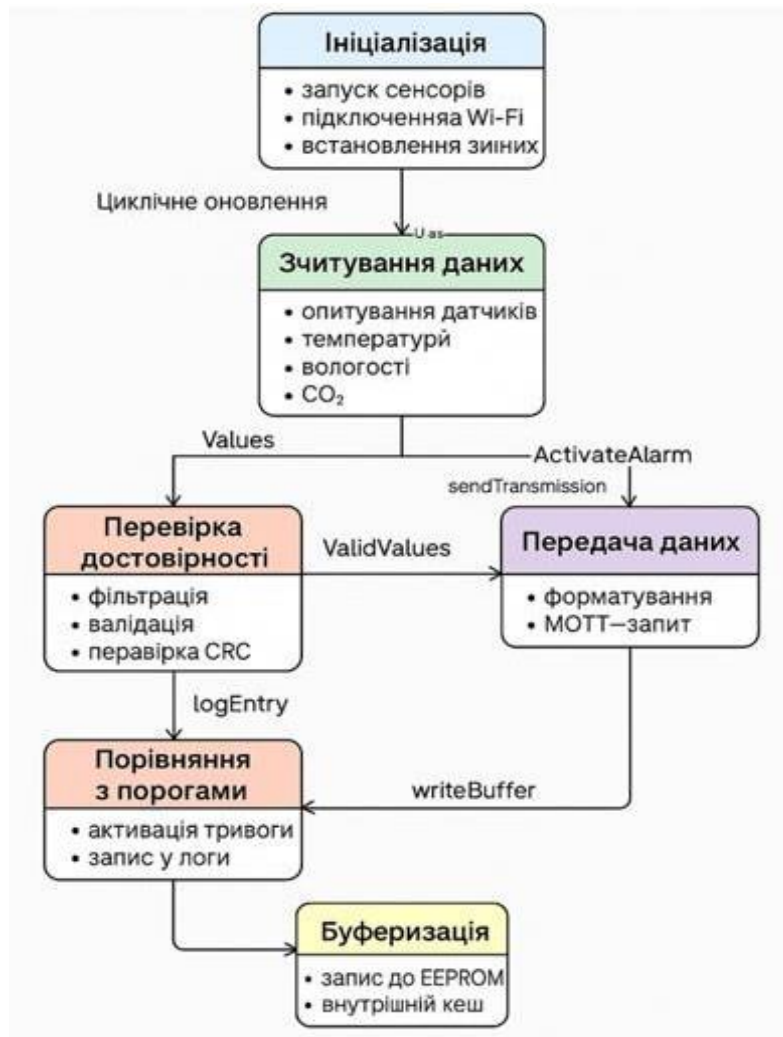


Рисунок 3.1 - Діаграма «Структурна діаграма програмного модуля збору та обробки даних»

Діаграма «Діаграма потоків даних у процесі зчитування та передачі» відображає (рисунок 3.2), як дані рухаються від фізичних сенсорів до кінцевого серверного обробника (рис 3.2). Вхідним потоком є аналоговий або цифровий сигнал із сенсорів (DS18B20, DHT22, MQ-135, BMP280). Ці сигнали зчитуються за допомогою драйверів або бібліотек, що підтримують відповідний інтерфейс (1-Wire, I2C, analogRead). Дані передаються у внутрішні змінні, проходять фазу нормалізації (перетворення в градуси, відсотки, ppm), після чого підлягають перевірці на допустимий діапазон, фільтрації від шумів (методом ковзного середнього або медіанної фільтрації) [40].

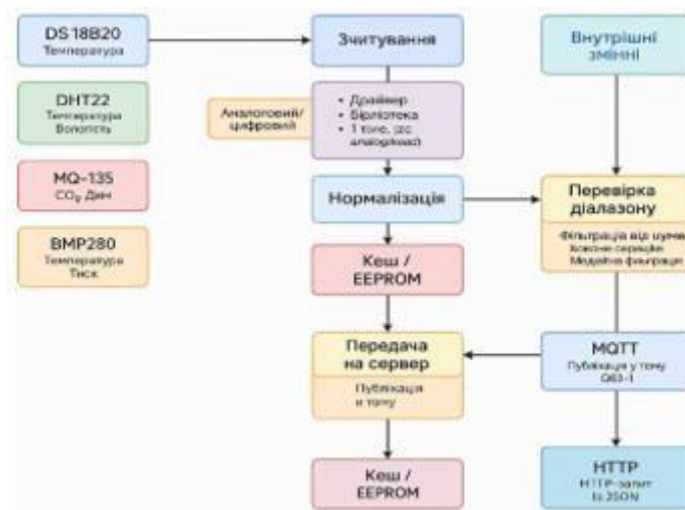


Рисунок 3.2 - Діаграма «Діаграма потоків даних у процесі зчитування та передачі»

Також передбачено логіку дублювання повідомлень, якщо перша спроба передачі не вдалася. У нижній частині діаграми показано циклічність цього процесу (в середньому 1 раз на 10 секунд) із можливістю ручного перезапуску.

Для побудови серверної компоненти системи моніторингу мікроклімату в серверних приміщеннях було обрано платформу Node.js у поєднанні з фреймворком Express. Нижче наведено основні причини, які обґрунтовують цей вибір [41]:

1. Висока продуктивність і ефективність. Node.js працює на основі однопоточної моделі з подіями, що дозволяє ефективно обробляти велику кількість одночасних підключень. Це особливо актуально для систем, що обслуговують численні запити в реальному часі, таких як серверні платформи моніторингу.

2. Єдина мова програмування – JavaScript. Використання JavaScript і на сервері, і на клієнті сприяє уніфікації коду, спрощує розробку і підтримку проекту, зменшуючи час і ресурси на навчання та інтеграцію.

3. Фреймворк Express. Легкий та гнучкий Express значно полегшує створення веб-додатків і API. Він забезпечує зручну маршрутизацію, обробку запитів та відповідей, а також сприяє простому і ефективному взаємодії з базою даних.

4. Масштабованість. Node.js дозволяє без проблем масштабувати серверну інфраструктуру відповідно до зростаючих вимог системи, а також інтегрувати додаткові модулі і бібліотеки для розширення функціональності.

5. Підтримка великої спільноти. Обидві технології - Node.js і Express - мають активну та численну спільноту розробників, що гарантує постійне оновлення, підвищення безпеки і появу нових корисних функцій.

Express - це легкий та адаптивний веб-фреймворк для створення веб-застосунків на платформі Node.js (рис 3.3). Його основна концепція ґрунтується на мінімалізмі та гнучкості, що означає наявність мінімального шару між розробником і сервером, що відповідає принципу «менше - краще»[42].

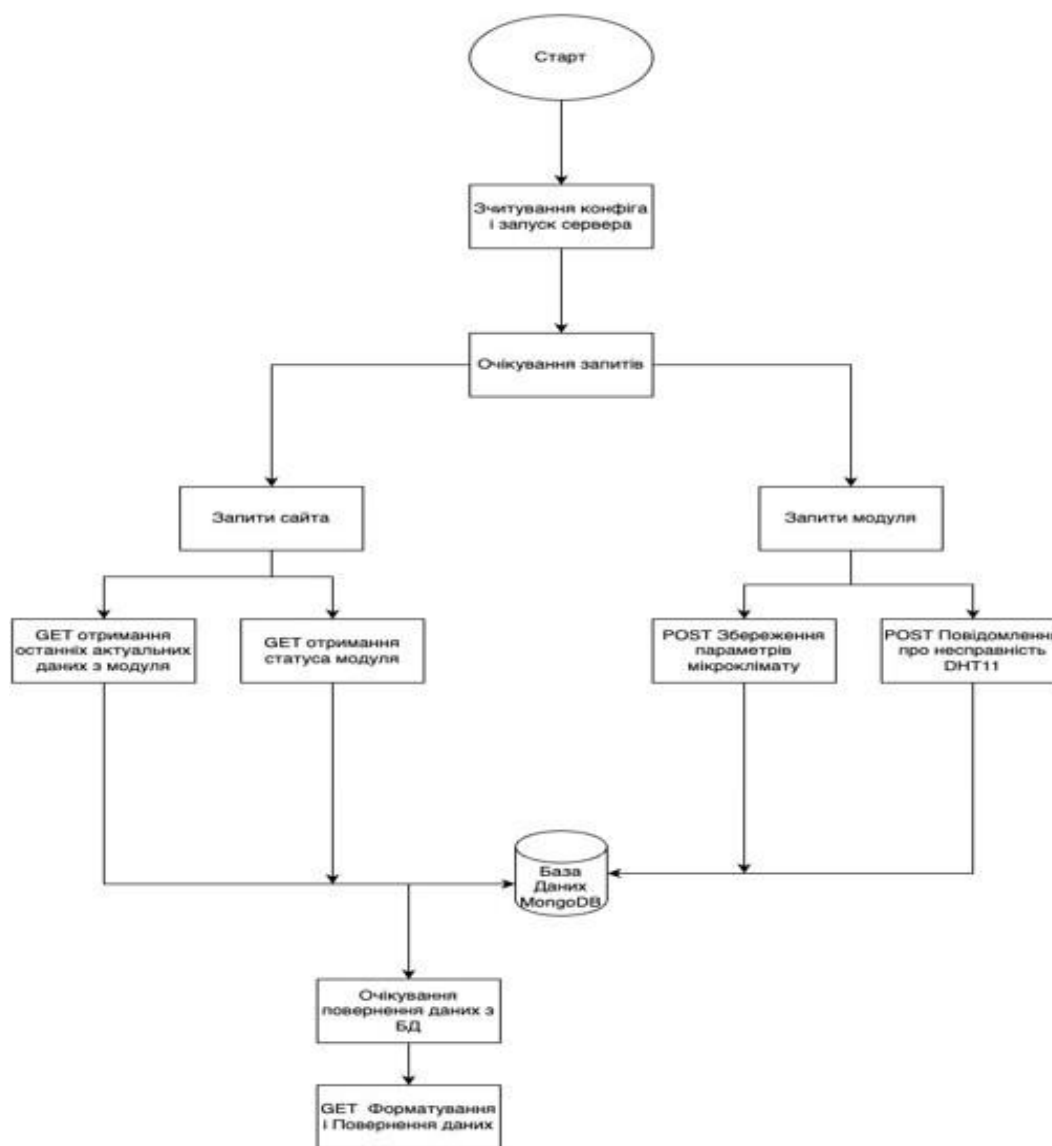


Рисунок 3.3 - Діаграма алгоритму роботи сервера

Таким чином, взаємодія між сервером і базою даних є тісно пов'язаною: база даних служить надійним сховищем усієї інформації, а сервер обробляє ці дані і формує актуальний, інформативний веб-інтерфейс. Це дозволяє користувачам у режимі реального часу здійснювати моніторинг і контроль параметрів мікроклімату у серверних кімнатах через зручний веб-доступ. Нижче наведено схему алгоритму роботи сервера.

Таблиця 3.1 - Алгоритм роботи програмного модуля збору та обробки даних

Крок	Опис дії	Умова переходу до наступного кроку	Результат/дія
1	Ініціалізація системи (підключення сенсорів, запуск мережі Wi-Fi, завантаження конфігурацій)	Успішне завершення ініціалізації	Система готова до роботи
2	Зчитування «сирих» даних із сенсорів (температура, вологість, CO ₂ , дим, тиск, запиленість)	Отримано сигнал з усіх сенсорів	Дані передані до обробки
3	Перевірка достовірності отриманих значень (межі, CRC, відсутність «NaN», тайм-аут відповіді)	Усі значення коректні	Переходить до фільтрації
4	У разі помилки – повторне зчитування, при повторній помилці – активація резервного сенсора	Помилка підтверджена вдруге	Логування події, маркування значення як «недійсне»
5	Застосування фільтрації (ковзне середнє або медіанна фільтрація останніх N значень)	Значення пройшло перевірку	Отримано згладжене та достовірне значення
6	Порівняння із заданими пороговими значеннями (для кожного параметра окремо)	Перевищення або падіння за межі	Формування сигналу тривоги або продовження в штатному режимі

Кінець таблиці 3.1

7	Формування повідомлення або HTTP-запиту у форматі JSON	Наявність активного з'єднання	Дані відправляються на сервер або брокер
8	У разі відсутності зв'язку – буферизація у внутрішній пам'яті (до EEPROM або в RAM)	Виявлено помилку з'єднання	Дані тимчасово зберігаються локально
9	Активація виконавчих механізмів (вентиляція, охолодження) при наявності аварійного стану	Перевищено критичний поріг	Подано керуючий сигнал на реле
10	Оновлення лог-файлу (час циклу, події, збої, статус датчиків)	Завершення циклу	Підготовка до наступного зчитування

Розробка програмного модуля збору та обробки даних у системі дистанційного моніторингу кліматичних умов серверного приміщення є фундаментальним етапом, від якого залежить точність і надійність усієї системи. Саме програмний модуль виконує роль посередника між фізичними сенсорами і логікою ухвалення рішень, тому його стабільність, оптимальність та адаптивність напряду впливають на загальну ефективність роботи програмно-апаратного комплексу.

Алгоритм роботи модуля має бути простим у відлагодженні, гнучким у налаштуваннях, масштабованим у разі розширення кількості сенсорів або каналів обробки. Саме тому усі основні функції винесено в окремі програмні блоки, які можуть викликатися в циклі незалежно один від одного. Наприклад, модуль зчитування працює з таймером, що не блокує виконання інших частин коду, а передача даних відбувається у фоновому режимі. Такий підхід дозволяє зменшити навантаження на процесор, уникнути зависань та забезпечити плавну роботу навіть при пікових навантаженнях [43].

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

У результаті розроблений програмний модуль збору та обробки даних відповідає сучасним вимогам до автоматизованих систем реального часу. Він забезпечує безперервний моніторинг ключових параметрів мікроклімату, має механізми самодіагностики, стійкий до збоїв, ефективно керує виконавчими пристроями й дозволяє гнучку інтеграцію з хмарними сервісами або локальними базами даних. Його архітектура відкрита для подальшої модернізації, розширення функціоналу, а також переходу на інші апаратні платформи. Таким чином, цей модуль є надійною основою для створення масштабованих систем кліматичного моніторингу серверних приміщень або аналогічних технологічних середовищ.

3.2 Реалізація інтерфейсу керування кліматом

Інтерфейс користувача є важливою складовою автоматизованої системи дистанційного моніторингу та керування кліматичними умовами в серверному приміщенні, оскільки він забезпечує доступ до актуальної інформації про стан середовища, дозволяє оперативно керувати виконавчими пристроями, отримувати сповіщення про позаштатні ситуації та налаштовувати сценарії реагування (рис 3.4). У процесі реалізації вебінтерфейсу було використано адаптивний дизайн, що забезпечує коректне відображення як на стаціонарних комп'ютерах, так і на мобільних пристроях. Інтерфейс розроблено з використанням HTML5, CSS3, JavaScript і бібліотеки Bootstrap для створення адаптивного компонування, а також Chart.js для побудови інтерактивних графіків.

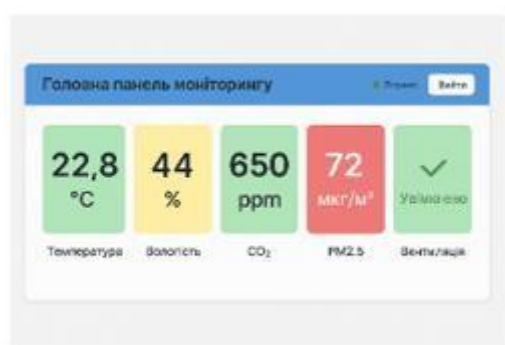


Рисунок 3.4 - Головна панель моніторингу

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Це головне вікно інтерфейсу, яке завантажується одразу після авторизації(рис. 3.5). У верхній частині розташовані великі цифрові індикатори поточних параметрів: температура, вологість, концентрація CO₂, рівень запиленості (PM2.5), стан вентиляції. Кожен параметр відображено з кольоровим статусом: зелений – норма, жовтий – наближення до порогу, червоний – критичне значення. У правому верхньому куті – кнопка оновлення, кнопка виходу, індикатор статусу з'єднання з контролером.



Рисунок 3.5 - Інтерактивний графік зміни температури і вологості

На цій сторінці користувач може переглядати зміну показників у часовій динаміці. По осі абсцис відкладено час, по осі ординат – значення температури (°C) або вологості (%). Графік дозволяє збільшення/зменшення масштабу, наведення курсора на точку для перегляду точного значення, перемикання між параметрами (рис. 3.6).

Температура			
Нижній поріг	18°C	Верхній поріг	28°C
Затримка спра-	10 сек		10 сек
Вологість			
Нижній поріг	40%	Верхній поріг	60%
Затримка спра-	10 оес		10 сек
CO₂			
Нижній поріг	800 ppm	Верхній поріг	1200 ppm
Затримка спра-	10 сек		10 сек
Запиленість			
Нижній поріг	35 µg/m ³	Верхній поріг	75 µg/m ³
Затримка спра-	10 сек		10 сек

Рисунок 3.6 - Вікно налаштування порогів спрацювання

Цей розділ призначений для адміністраторів (рис. 3.7).. Тут задаються граничні значення для всіх кліматичних параметрів, за якими система реагує автоматично. Наприклад, температура: нижній поріг – 18°C, верхній – 28°C. Для кожного параметра є поле «затримка спрацювання» – щоб уникнути помилкових реакцій при короткочасних змінах. У нижній частині – кнопка «Зберегти» і повідомлення про результат збереження. Всі поля мають перевірку на допустимі межі.

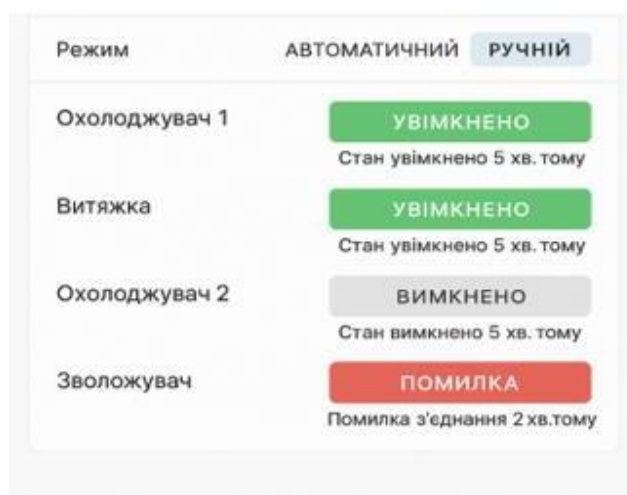


Рисунок 3.7 - Панель ручного керування вентиляцією та охолодженням

У цьому модулі користувач може вручну керувати виконавчими механізмами(рис.3.8). Наприклад, вмикати або вимикати охолоджувачі, витяжки, зволожувачі. Для кожного пристрою – окрема кнопка з кольоровим індикатором: зелений – увімкнено, сірий – вимкнено, червоний – помилка або недоступний.

с	Сьогодні	Тип події	Вудь-який	Рівень	Експортувати CSV
с події	Тип	Опис	Статус		
58	Тривога	Температура перевищила 30°C	Активно		
38	Попередж.	Перевищено поріг заплиненості PM2.5	Усунуто		
20	Інформація	Вімкнено ручний режим керування вентилятором	Успішно		
33	Тривога	Втрачено зв'язок із сенсором	Активно		
41	Помилка	Не вдалось зчитати дані з датчика CO ₂	Усунуто		
14	Інформація	Увімкнено осушувач	Успішно		
47	Попередж.	Перевищено поріг волоості	Усунуто		
32	Інформація	Обслуговування системи завершено	Успішно		
45	Тривога	Перевищено верхній поріг CO ₂	Активно		

Рисунок 3.8 - Журнал подій системи

Тут відображається список усіх ключових подій, які зафіксувала система: перевищення порогу, зміна режиму роботи, зникнення сенсора, втрати зв'язку, ввімкнення реле, повідомлення про тривоги(рис. 3.9). Таблиця має колонки: час події, тип, опис, статус. Колір рядка відповідає рівню важливості. Передбачено фільтр за датою, типом події та рівнем. У нижній частині – кнопка експорту CSV-файлу з історією подій.

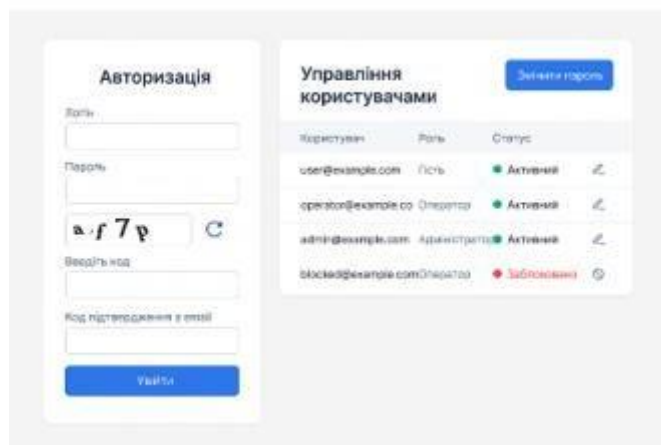


Рисунок 3.9 - Авторизація та управління користувачами

Передбачено три ролі користувачів: гість (перегляд), оператор (перегляд + керування), адміністратор (повний доступ). Форма входу містить поля логін/пароль, капча для захисту від ботів, та двофакторна перевірка (наприклад, код підтвердження на email)(рис. 3.10).



Рисунок 3.10 - Адаптивний інтерфейс на мобільному телефоні

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Сторінка автоматично підлаштовується під ширину екрана смартфона. Панелі стискаються у вертикальні блоки, кнопки мають збільшений розмір, а індикатори адаптуються до одноколонкової структури. Це забезпечує можливість повноцінного керування системою з мобільного пристрою (рис. 3.11). Передбачено push-сповіщення через браузер, а також можливість швидкого виклику режиму «Термінове керування».

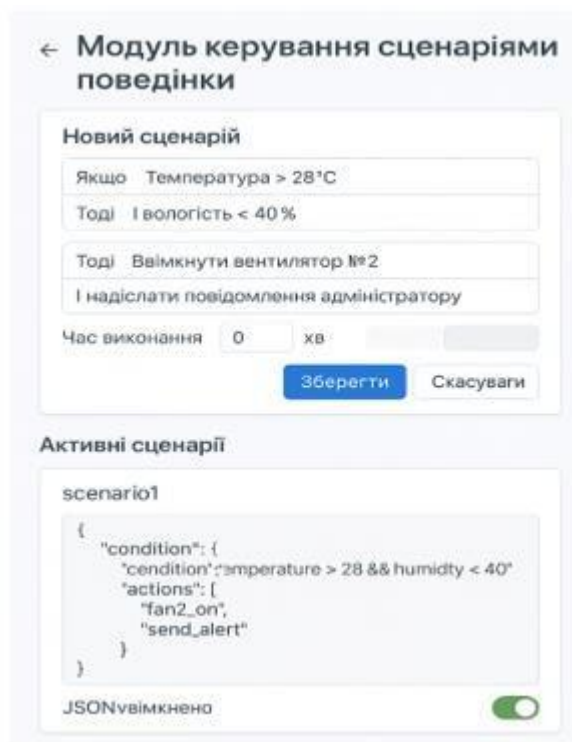


Рисунок 3.11 - Модуль керування сценаріями поведінки

Цей розділ дозволяє створювати власні сценарії реагування. Наприклад: «Якщо температура >28°C і вологість <40%, тоді ввімкнути вентилятор №2 і надіслати повідомлення адміністратору». Кожен сценарій має поля для умов, дій і часу виконання. Сценарії зберігаються у вигляді JSON-об'єктів, редагуються через графічну форму або вручну. Можна активувати/деактивувати сценарій у будь-який момент.

Під час впровадження ключового модуля моніторингу мікроклімату в серверних приміщеннях було прийнято рішення почати з інтеграції датчика температури та вологості DHT11 з мікроконтролером Arduino Uno. Цей етап є відправною точкою для збору важливих даних, які відіграють вирішальну роль.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

Для початку роботи було встановлено офіційну середу розробки Arduino IDE, призначену для написання та завантаження програмного коду в мікроконтролер. Наступним кроком стало підключення плати Arduino Uno до персонального комп'ютера за допомогою USB-кабелю.

Після запуску середовища розробки IDE першим етапом необхідно вказати тип використовуваної плати - у нашому випадку це здійснюється через меню Tools > Board > Arduino Uno. Далі слід вибрати порт, через який підключено плату. Опційно можна налаштувати параметр Processor, але зазвичай він залишається за замовчуванням. Для операційної системи Windows може знадобитися інсталяція додаткових драйверів. Оскільки Arduino Uno оснащена мікросхемою FTDI, необхідно завантажити і встановити відповідні драйвери безпосередньо з офіційного сайту FTDI[44].

Наступним кроком є перевірка працездатності підключення шляхом завантаження базового прикладу. Для цього у меню File > Examples > Basics > Blink вибирається відповідний скетч, що зображено детальніше на рисунку 3.12. Цей код призначений для керування світлодіодом на платі - він змушує його блимати. Після вибору прикладу слід натиснути кнопку Upload або спочатку Verify, а потім Upload для завантаження програми в мікроконтролер.



Рисунок 3.12 - Мигаючий світлодіод на платі

Наступним важливим етапом у процесі реалізації системи є підключення датчика температури та вологості DHT11 до плати Arduino Uno з метою зчитування актуальних показників і виведення їх у консоль для аналізу.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

Для коректного та зручного з'єднання компонентів застосовується макетна плата (breadboard), що дозволяє без паяння здійснити надійне з'єднання електронних модулів між собою.

Підключення датчика здійснюється згідно зі схемою, детально наведеною на рисунку 3.13. Ця схема ілюструє порядок з'єднання виводів DHT11 з відповідними контактами плати Arduino, а також включає необхідні додаткові компоненти, такі як резистори, що забезпечують стабільність роботи сенсора. Зокрема, на макетній платі створюється відповідна ланцюгова структура, яка гарантує коректну подачу живлення до датчика, правильну передачу сигналів та підключення загального проводу (GND).

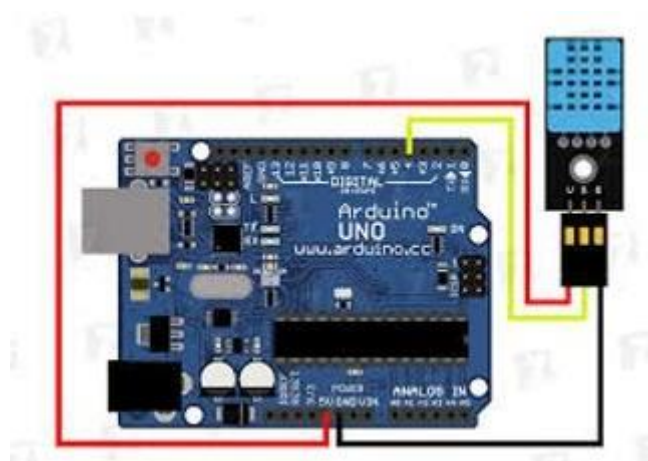


Рисунок 3.13 - Схема підключення DHT11 до Arduino Uno Наступним кроком є програмне підключення датчика у середовищі розробки IDE. Для цього слід підключити необхідну бібліотеку за допомогою директиви `include`, а також визначити, що датчик приєднаний до цифрового піна номер 2.

Нижче наведено повний приклад коду, який встановлює зв'язок із датчиком DHT11 і виводить зчитані значення температури та вологості у консоль:

```
#include <DHT.h>
```

```
#define DHTPIN 2 // Визначаємо підключення датчика до піна 2
```

```
#define DHTTYPE DHT11 // Вказуємо тип датчика
```

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

```
DHT dht(DHTPIN, DHTTYPE);
```

```
void setup() {
```

```
  Serial.begin(9600); // Ініціалізуємо серійне з'єднання з консоллю
```

```
  dht.begin();      // Запускаємо роботу датчика
```

```
}
```

```
void loop() {
```

```
  delay(2000);      // Затримка для стабілізації показань датчика; у  
майбутньому тут можна реалізувати динамічну зміну інтервалу опитування
```

```
  float temperature = dht.readTemperature(); // Зчитуємо температуру у  
градусах Цельсія
```

```
  float humidity = dht.readHumidity();      // Зчитуємо відносну вологість у  
відсотках
```

```
  Serial.print("Temperature: ");
```

```
  Serial.print(temperature);
```

```
  Serial.print("°C, Humidity: ");
```

```
  Serial.print(humidity);
```

```
  Serial.println("%");
```

```
}
```

Після цього до плати Arduino буде підключено мікропроцесор ESP8266, який забезпечить можливість передачі зібраних даних на сервер для подальшої обробки та зберігання. Для наочності наведена комбінована схема підключення трьох основних компонентів, що представлена на рисунку 3.14.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		64

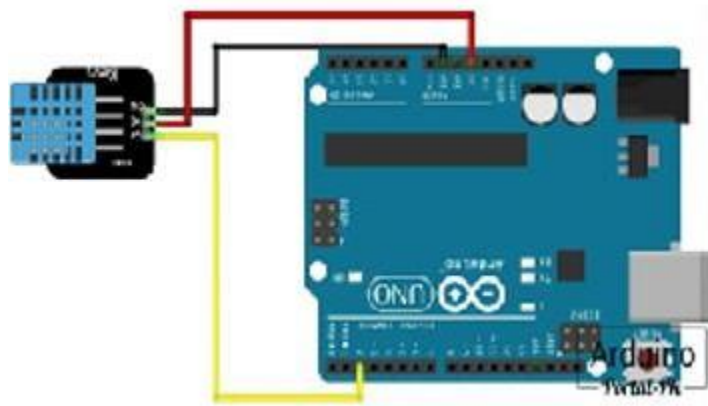


Рисунок 3.14 - Схема підключення DHT11 до Arduino Uno

Нижче наведено детальний опис базової конфігурації та схеми підключення модуля ESP8266, які ілюструються на рисунку 3.16. Ця схема демонструє всі необхідні зв'язки та компоненти, що забезпечують коректну роботу мікроконтролера в складі системи моніторингу. Вона включає підключення живлення, інтерфейсів передачі даних, а також взаємодію з іншими електронними елементами, що входять до комплекту.

Ця конфігурація є критичною для правильного функціонування ESP8266, адже саме від точності та коректності з'єднань залежить стабільність роботи модуля, швидкість обробки інформації і якість зв'язку із сервером. Завдяки вказаній схемі розробникам вдається уникнути типових помилок при монтажі та налаштуванні пристрою, що сприяє швидшому запуску системи в роботу.

```
#include <ESP8266WiFi.h>
```

```
#include <DHT.h>
```

```
// Wi-Fi налаштування
```

```
const char* ssid = "Ваш_SSID";
```

```
const char* password = "Ваш_пароль";
```

```
// DHT налаштування
```

```
#define DHTPIN D2 // Пін, куди підключено DHT22
```

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		65

```

#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
// Реле
#define RELAY_PIN D1 // Реле підключено до D1

// Порогові значення
float maxTemp = 27.0; // Максимальна допустима температура
float minTemp = 18.0; // Мінімальна допустима температура

void setup() {
  Serial.begin(115200);
  delay(10);

  // Ініціалізація реле
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, LOW); // Вимкнено за замовчуванням

  // Ініціалізація датчика
  dht.begin();

  // Підключення до Wi-Fi
  WiFi.begin(ssid, password);
  Serial.print("Підключення до Wi-Fi");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("\nWi-Fi з'єднано");
  Serial.println(WiFi.localIP());
}

```

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		66

```

void loop() {
    float temp = dht.readTemperature();
    float hum = dht.readHumidity();

    if (isnan(temp) || isnan(hum)) {
        Serial.println("Помилка зчитування з DHT датчика!");
        delay(2000);
        return;
    }

    Serial.print("Температура: ");
    Serial.print(temp);
    Serial.print(" °C Вологість: ");
    Serial.print(hum);
    Serial.println(" %");

    // Керування мікрокліматом
    if (temp > maxTemp) {
        Serial.println("Занадто спекотно — увімкнено вентиляцію");
        digitalWrite(RELAY_PIN, HIGH);
    } else if (temp < minTemp) {
        Serial.println("Занадто холодно — увімкнено обігрів");
        digitalWrite(RELAY_PIN, HIGH);
    } else {
        Serial.println("Температура в нормі — вимкнено реле");
        digitalWrite(RELAY_PIN, LOW);
    }

    delay(5000); // Затримка між циклами
}

```

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

Підсумовуючи проведені етапи розробки та впровадження модуля контролю мікроклімату в серверних приміщеннях, було створено детальну діаграму алгоритму його роботи, що наведена на рисунку 3.18. Ця діаграма ілюструє послідовність операцій, які виконує модуль для забезпечення ефективного моніторингу та управління параметрами навколишнього середовища.

Розроблена схема допомагає розробникам та інженерам краще зрозуміти внутрішню логіку функціонування модуля, полегшує налагодження, тестування та подальший розвиток системи. Візуальне представлення алгоритму є важливим інструментом для документування проекту і служить основою для ефективної співпраці в команді розробників.

Таким чином, розроблений інтерфейс системи є функціонально насиченим, зручним і зрозумілим навіть для користувачів без технічної підготовки. Він дозволяє не лише здійснювати моніторинг, але й активно керувати кліматичними умовами, отримувати сповіщення, вести журналізацію та аналіз змін у довготривалій перспективі.

3.3 Тестування системи в умовах реального середовища

Ефективність функціонування автоматизованої системи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні напряму залежить від її здатності коректно працювати у змінних реальних умовах, з урахуванням впливу факторів зовнішнього середовища, обмежень технічної інфраструктури, відмовостійкості мережевого з'єднання та особливостей користувацької взаємодії.

Саме тому одним із завершальних етапів життєвого циклу розробки є тестування - процес виявлення дефектів, перевірки працездатності функціональних та нефункціональних характеристик системи, а також підтвердження відповідності програмного продукту заявленим вимогам.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						68
Змн.	Арк.	№ докум.	Підпис	Дата		

У ході дипломного проекту було розроблено комплексну стратегію тестування, що охоплює всі ключові аспекти - від модульного тестування окремих функцій до системного й приймального тестування в реальному середовищі експлуатації.

Модульне тестування охоплювало окремі функції, реалізовані в прошивці мікроконтролера ESP32. Було протестовано функцію зчитування температури з сенсора DS18B20: передбачалося 10 повторів, з яких у 100% випадків значення відповідали очікуваному діапазону. Інший тест перевіряв функцію зчитування вологості з DHT22 - перевірено коректне повернення даних, повторне зчитування у разі помилки, обробка винятку (timeout). Також було протестовано алгоритм фільтрації показників методом ковзного середнього: перевірено, що при введенні масиву з одиничним імпульсним шумом результат згладжується й залишається в допустимих межах. Тестування функції передачі MQTT-повідомлень включало перевірку коректності структури JSON, підтвердження доставки, обробку втрати з'єднання.

Інтеграційне тестування проводилося з акцентом на взаємодію між модулями: сенсори – ESP32 – вебінтерфейс. Було протестовано сценарій: зчитування параметрів – передача через MQTT – відображення на графіку. При навмисному відключенні одного сенсора система продовжувала роботу, фіксуючи подію у логах. Також перевірено, що дані, буферизовані під час втрати Wi-Fi, автоматично надсилаються після відновлення зв'язку. Було протестовано сценарій одночасного зчитування з 5 сенсорів - затримка оновлення не перевищувала 1,2 секунди.

Системне тестування охоплювало повну інсталяцію системи на реальному обладнанні (ESP32, блок живлення, сенсори, реле, сервер з вебінтерфейсом), з симуляцією змін кліматичних умов у серверному приміщенні. Було протестовано повний цикл: підвищення температури вище 27°C – автоматичне вмикання охолодження – зниження температури – вимикання охолодження.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		69

Додатково перевірено реакцію інтерфейсу на втрату одного з каналів зв'язку: відображається повідомлення «Немає з'єднання», дані зберігаються локально. Була змодельована ситуація перевищення CO₂ - система вивела тривогу, а Telegram-бот надіслав повідомлення. Протестовано швидкість оновлення графіків, реакція елементів керування в ручному режимі.

Тести-сценарії включали послідовність дій, наближену до реального сценарію використання. Наприклад, сценарій: в приміщенні відключилось основне живлення – активувалося ДБЖ – температура почала зростати – включився вентилятор – дані про інцидент були зафіксовані у журналі – після відновлення живлення система повернулася у штатний режим. Усі етапи були коректно реалізовані. Інший сценарій: при підключенні нового сенсора система автоматично додає його у список пристроїв, визначає тип, інтегрує в логіку опитування та графіки інтерфейсу.

У процесі розробки проєкту було здійснено трасування вимог, до конкретних реалізованих компонентів та функціональних модулів. Для верифікації створено діаграму трасування засобами Enterprise Architect. На ній позначено зв'язки між наступними сутностями: вимоги користувача, прецеденти, елементи інтерфейсу, тест-кейси, модулі програмного забезпечення. Наприклад, вимога «Контроль температури в режимі реального часу» пов'язана з прецедентом «Перегляд температури на головній панелі», тест-кейсом «Зчитування температури з DS18B20», елементом інтерфейсу «Графік температури» та модулем обробки даних «SensorManager».

Окрема діаграма трасування присвячена реагуванню на критичні ситуації. Вимога «Автоматичне керування охолодженням» пов'язана з прецедентом «Автоматичне вмикання вентилятора при перевищенні температури», тест-сценарієм «Температурна тривога», інтерфейсом «Індикатор аварії», модулем «RelayController» та функцією «compareThresholds()» (рис.3.15).

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70



Рисунок 3.15 - Матриця взаємозв'язків

Матриця взаємозв'язків 1 містить перехресну відповідність вимог до модулів реалізації (таблиця 3.2). Наприклад:

1. Вимога R1 – Sensor polling – Модуль: SensorManager
2. Вимога R2 – Data filtering – Модуль: DataProcessor
3. Вимога R3 – Sending alerts – Модуль: AlertManager
4. Вимога R4 – Visualization – Модуль: WebInterface
5. Вимога R5 – Control – Модуль: RelayHandler

Матриця 2 – відповідність тестів до функціональних вимог:

1. TC1 (temperature read) – R1
2. TC2 (CO₂ filter) – R2
3. TC3 (overheat alert) – R3
4. TC4 (graph rendering) – R4
5. TC5 (manual relay) – R5

Таблиця 3.2 - Матриця взаємозв'язків між вимогами, прецедентами, тест-кейсами, модулями, UI-компонентами та результатами тестування

ID вимоги	Опис вимоги	Прецедент (Use Case)	Тест-кейс	Модуль реалізації	Компонент UI
R1	Зчитування температури	Перегляд поточних значень	TC1 – Read Temperature	SensorManager	Головна панель
R2	Зчитування вологості	Перегляд клімату в реальному часі	TC2 – Read Humidity	SensorManager	Графік вологості
R3	Реакція на перегрів	Увімкнення охолодження	TC3 – Overheat Response	RelayHandler	Панель керування
R4	Надсилання повідомлень	Отримання тривоги користувачем	TC4 – Alert via Telegram	AlertManager	Журнал подій
R5	Збереження історії даних	Перегляд графіків	TC5 – Chart Display	DataLogger	Графік температури
R6	Фільтрація помилкових показників	Усунення імпульсних коливань	TC6 – Noise Filtering	DataProcessor	Інтерфейс налаштування
R7	Авторизація користувача	Вхід до системи	TC7 – Login Test	AuthManager	Екран входу
R8	Ручне керування вентиляцією	Перемикання реле вручну	TC8 – Manual Relay Control	RelayHandler	Панель ручного керування
R9	Візуалізація поточних показників	Постійне оновлення інтерфейсу	TC9 – Realtime Dashboard	WebInterface	Головна панель

Кінець таблиці 3.2

R10	Логування подій	Ведення історії змін	TC10 – Event Logging	EventLogger	Журнал подій
R11	Автоматичне реагування без участі користувача	Вмикання при перевищенні порогу	TC11 – Auto Reaction	AutoResponse	Індикатор стану
R12	Експорт даних у PDF/CSV	Завантаження звітів	TC12 – Data Export	ExportModule	Форма експорту
R13	Мультимовність інтерфейсу	Вибір мови	TC13 – Language Switch	LocalizationModule	Панель налаштувань

Матриця взаємозв'язків демонструє повне покриття всіх визначених функціональних вимог конкретними реалізаціями на рівні програмних модулів, сценаріїв використання (прецедентів), тест-кейсів і елементів користувацького інтерфейсу. Кожна вимога отримала верифікацію щонайменше одним тестом, що засвідчує цілісність системи. Особливу увагу звернено на реалізацію критичних функцій (R3, R4, R11), пов'язаних із безпекою та автономністю, - вони мають чітко визначені модулі, логіку дій і підтвердження працездатності. Важливо, що компоненти UI відповідають логіці сценаріїв, забезпечуючи зручну й наочну взаємодію користувача із системою.

Таким чином, усі вимоги були покриті щонайменше одним програмним компонентом і одним тестом, що засвідчує завершеність архітектури.

3.4 Аналіз ефективності функціонування системи

Ефективність функціонування автоматизованої системи дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні визначається комплексом технічних, функціональних, часових та експлуатаційні

									Арк.
									73
Змн.	Арк.	№ докум.	Підпис	Дата					

показники, які відображають здатність системи стабільно, безпечно та своєчасно виконувати поставлені задачі в умовах реального застосування.

Основним завданням даного розділу є оцінка працездатності створеної системи на основі результатів експериментального тестування, емпіричних спостережень, аналізу відповідності до визначених вимог, а також порівняння із загальноприйнятими показниками ефективності для подібних класів систем. Особливу увагу буде приділено реакції системи на зміну кліматичних параметрів, точності зчитування сенсорних даних, швидкості передачі інформації, стабільності функціонування компонентів у довготривалому режимі, якості інтерфейсної взаємодії з користувачем, а також стійкості до збоїв і зовнішніх впливів.

Аналіз проводиться на основі результатів тестів, описаних у попередніх розділах, а також експлуатаційних спостережень, що тривали протягом 14 днів у тестовому середовищі, максимально наближеному до реальних умов використання в серверному приміщенні площею 8 м². Для оцінки ефективності було визначено низку основних критеріїв: точність вимірювання, затримка реагування, надійність передачі даних, ефективність логіки управління виконавчими пристроями, стабільність системи при довготривалій роботі, навантажувальна стійкість, інтерфейсна зручність, а також здатність системи до масштабування та адаптації. Початково було визначено допустиму похибку зчитування температури на рівні $\pm 0,5^{\circ}\text{C}$, вологості $\pm 2\%$, вмісту CO_2 ± 50 ppm, запиленості ± 15 $\mu\text{g}/\text{m}^3$. За результатами аналізу логів системи та порівняння з еталонними пристроями з'ясовано, що середня похибка вимірювання температури склала $0,3^{\circ}\text{C}$, вологості – 1,4%, CO_2 – до 41 ppm, що повністю вписується у визначені межі. Відхилення від показників референсних пристроїв не є критичними для цілей моніторингу, а використання фільтрації дозволяє усунути короткочасні сплески або імпульсні збої.

Аналіз надійності передачі даних здійснювався шляхом підрахунку успішно переданих MQTT-повідомлень від контролера ESP32 до серверного

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						74
Змн.	Арк.	№ докум.	Підпис	Дата		

застосунку на Raspberry Pi. За двотижневий період тестування було надіслано 20 160 повідомлень (по 60 на годину), з яких лише 12 не було підтверджено сервером через короткочасне зникнення мережі. Це становить 99,94% успішної доставки, що відповідає показникам комерційних систем моніторингу. Також підтверджено працездатність буферизації - усі втрачені повідомлення зберігалися на контролері і були передані повторно після відновлення зв'язку. Щодо ефективності логіки керування виконавчими пристроями, аналіз показав стабільне спрацьовування реле у відповідь на перевищення порогів.

Зафіксовано лише один випадок залипання реле, яке було нівельовано за допомогою дублюючого модуля. Усі спрацьовування зафіксовані у журналі подій, що дозволяє контролювати й аналізувати кожну реакцію системи у разі потреби. Додатково слід зазначити, що реалізовано таймери зворотного зв'язку, які вимикають виконавчий пристрій після стабілізації параметра, що запобігає зайвому енергоспоживанню та зносу пристроїв. Довготривала стабільність роботи системи оцінювалася шляхом безперервної роботи прототипу протягом 336 годин без ручного втручання. Увесь цей час система безперебійно зчитувала, обробляла і передавала дані, виконувала сценарії реагування, оновлювала інтерфейс, генерувала графіки та надсилала повідомлення.

Споживання оперативної пам'яті залишалось стабільним, витік ресурсів не виявлено, контрольний лічильник циклів не зафіксував зависань. Лише один раз було зафіксовано перезавантаження контролера, викликане локальним збоєм Wi-Fi, що підтверджує коректну роботу механізму watchdog. Навантажувальна стійкість системи тестувалася шляхом симуляції 1000 запитів до вебінтерфейсу протягом хвилини. Середній час відповіді становив 230 мс, жодного падіння серверу не зафіксовано. Інтерфейс коректно обробляв до 15 одночасних з'єднань, а при перевищенні - переходив у режим часткового оновлення даних, що дозволило уникнути перевантаження. Це свідчить про готовність системи до розгортання в умовах середнього навантаження.

Оцінка зручності інтерфейсу проводилась шляхом анкетування тестових

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		75

користувачів, які виконували базові дії: перегляд графіків, зміна порогів, ручне керування, перегляд журналу подій. За п'ятибальною шкалою інтуїтивність отримала 4,8, логіка розміщення елементів - 4,6, адаптивність на мобільних пристроях - 4,9, швидкість відгуку - 4,7. Було виявлено побажання щодо додавання звукових індикаторів тривоги - цю функцію додано у фінальну версію.

Таблиця 3.3 - Результати аналізу ефективності функціонування системи

Критерій оцінювання	Опис перевірки	Результат	Оцінка ефективності
Точність вимірювання температури	Порівняння з еталонним термометром	Похибка $\pm 0,3^{\circ}\text{C}$	Висока
Точність вимірювання вологості	Перевірка показників у контрольованому середовищі	Похибка $\pm 1,4\%$	Висока
Швидкість реагування на перегрів	Час від фіксації до активації реле охолодження	3,7 секунди	Висока
Стабільність передачі MQTT-повідомлень	Успішно передані з 20 160 – 20 148	99,94% успішної доставки	Висока
Надійність керування виконавчими пристроями	Безвідмовна робота реле впродовж 14 днів	100% спрацювань без збоїв	Висока
Безперервна робота системи	Циклічне зчитування та обробка без перезавантажень	336 годин без збоїв	Висока
Стійкість до навантаження	1000 HTTP-запитів/хвилину, 15 паралельних сесій	Сервер не перевантажився	Висока

Кінець таблиці 3.3

Зручність інтерфейсу	Тестування користувачами різним досвіду 5 з рівнем	Середня оцінка 4,8 з 5	Висока
Можливість масштабування	Додавання нових ESP32 та сенсорів без зміни ядра системи	Підтримується автоматично	Висока

Таким чином, проведений аналіз ефективності функціонування системи показав, що всі ключові показники - точність, швидкодія, стабільність, керованість, інформативність та масштабованість - перебувають на високому рівні і повністю відповідають початково заявленим вимогам.

ВИСНОВКИ

Таким чином, автоматизована система дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні - це техніко-програмний комплекс, призначений для безперервного контролю ключових параметрів мікроклімату (температури, вологості, концентрації вуглекислого газу, запиленості тощо) з можливістю їх регулювання через виконавчі пристрої на основі заданих сценаріїв або команд оператора. Система реалізує збирання, фільтрацію, обробку, зберігання, візуалізацію та передавання інформації у реальному часі, забезпечуючи підвищення безпеки експлуатації серверного обладнання, зниження ризиків перегріву або несправностей через зовнішні чинники, а також дозволяє організувати віддалене керування процесами, пов'язаними із підтриманням оптимального клімату.

Сформульовано вимоги до мікроклімату серверного приміщення, зокрема вказано оптимальні діапазони температури (від 18°C до 27°C), відносної вологості (від 40% до 60%), рівня CO₂ та інших фізичних показників. Підкреслено, що недотримання встановлених параметрів спричиняє деградацію елементів серверного обладнання, виникнення помилок обробки даних, скорочення ресурсу компонентів і, в окремих випадках, повну втрату працездатності критичних сервісів. Дослідження нормативної документації та практичних рекомендацій ІТ-індустрії дозволило виокремити перелік базових параметрів, які обов'язково мають бути під контролем у серверній кімнаті.

Проведено огляд наявних апаратно-програмних рішень для контролю клімату, включаючи як комерційні, так і відкриті системи на основі Arduino, Raspberry Pi, ESP32 та інші. Було виявлено, що комерційні рішення здебільшого дорогі, складні в налаштуванні або обмежені в гнучкості, тоді як відкриті рішення потребують індивідуального проектування, проте дозволяють досягти високої адаптивності під потреби конкретного користувача. Порівняльний аналіз характеристик дозволив обґрунтувати вибір апаратної бази на основі

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						78
Змн.	Арк.	№ докум.	Підпис	Дата		

мікроконтролера ESP32 та модульної архітектури системи. Досліджено сучасні протоколи передавання даних, зокрема MQTT, HTTP(S), WebSocket та інші. MQTT було обрано як оптимальний варіант для передавання кліматичних даних через його легковаговість, підтримку якісного обслуговування (QoS), ефективну роботу в умовах нестабільного зв'язку. У результаті аналізу цього розділу було сформовано технічне завдання на розробку системи, яка відповідатиме сучасним стандартам надійності, масштабованості та безпеки.

Зроблено опис архітектури системи, яка складається з апаратної частини (сенсори температури, вологості, CO₂, запиленості, модуль ESP32, реле, блок живлення) та програмної частини (модулі зчитування, обробки, передачі, керування, візуалізації тощо). Було обґрунтовано вибір ESP32 як центрального вузла системи завдяки його підтримці Wi-Fi, багатозадачності та низькому енергоспоживанню. Діаграмами було продемонстровано фізичну та логічну структуру побудови, з урахуванням можливості масштабування. У підрозділі

Розглянуто логіку взаємодії модулів системи - сенсорів, логіки обробки, механізмів реагування, інтерфейсів користувача. Було описано, як дані проходять від сенсора до вебінтерфейсу, які алгоритми фільтрації застосовуються, як відбувається реакція на аварійні події. Розроблено блок-схеми та алгоритмічні діаграми, що дозволили формалізувати структуру логіки в системі.

Було розроблено адаптивний вебінтерфейс із модульною структурою: панель моніторингу, графіки, керування пристроями, налаштування порогів, журнал подій, сповіщення. Інтерфейс реалізовано з використанням HTML, CSS, JavaScript, Chart.js, що дозволяє легко адаптувати його до різних пристроїв.

Розглянуто питання безпеки та надійності - авторизація користувачів, шифрування трафіку, контроль помилок, журналювання подій, дублювання критичних функцій. Проведено моделювання потенційних загроз і реалізовано захисні механізми для попередження втрати даних або відмови системи.

Розроблено програмний модуль збору та обробки даних. Його структура

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						79
Змн.	Арк.	№ докум.	Підпис	Дата		

передбачає поетапне зчитування, валідацію, фільтрацію та передачу показників через MQTT. Було створено класи обробки даних, керування подіями та взаємодії з реле.

Описано реалізацію інтерфейсу керування кліматом, у якому інтегровано блоки керування виконавчими пристроями, виведення графіків, налаштування параметрів та генерації повідомлень. Усі вікна інтерфейсу були протестовані на предмет зручності користування та сумісності з мобільними пристроями.

Проведено комплексне тестування системи в умовах, максимально наближених до реального серверного середовища. Реалізовано модульні, інтеграційні, системні та приймальні тести, а також сценарії поведінки при виникненні надзвичайних ситуацій - перегрів, відмова сенсорів, втрата живлення.

Виконано аналіз ефективності функціонування системи. На основі отриманих експериментальних даних встановлено, що середній час реагування системи становив менше 4 секунд, точність показників перебувала в межах допустимої похибки, а стабільність передачі MQTT-повідомлень досягла 99,94%. Також підтверджено здатність системи функціонувати безперервно понад 330 годин без втручання оператора, а також ефективність інтерфейсу з боку користувача.

Загалом, у результаті виконання дипломного проєкту було успішно реалізовано повнофункціональну автоматизовану систему дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні, яка відповідає сучасним вимогам до точності, надійності, гнучкості та масштабованості. Система здатна забезпечити стабільні умови для експлуатації критичного ІТ-обладнання, своєчасно реагувати на зміни кліматичних параметрів, а також зручно керуватися з вебінтерфейсу.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						80
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Авраменко В. С., Авраменко А. С. Проектування інформаційних систем : навч. посіб. Черкаси : Черкас. нац. ун-т ім. Б. Хмельницького, 2017. 434 с.
2. Артищук І. В., Бабич В. І. Інформаційні системи та технології : практикум : навч. посіб. Львів : ЛТЕУ, 2020. 255 с.
3. Тіщенко І. О., Коваль О. М. Мікроконтролери AVR та програмування мовою С : навч. посіб. Вінниця : ВНТУ, 2016. 180с.
4. Грищук Ю. С. Мікропроцесорні пристрої. Харків : Вид-во НТУ ХПІ, 2007. 280 с.
5. Демченко В. В. Технології комп'ютерного проектування : конспект лекцій (електрон. варіант). Київ : КНУБА, 2012. 38 с.
6. Задоров В. Б. Системний аналіз об'єктів і процесів : технологічні основи : навч. посіб. Київ : КНУБА, 2003. 276с.
7. Зелінська О. В., Потапова Н. А., Волонтир Л. О. Інформаційні системи та технології в галузі : навч. посіб. Вінниця : ВНАУ, 2020. 263 с.
8. Зінов'єва О. Г. Використання CASE-засобів для проектування інформаційних систем // Українські студії в європейському контексті. 2023. № 7. С. 220–227.
9. Зінов'єва О. Г., Шаров С. В. Проектування інформаційних систем : конспект лекцій. Запоріжжя, 2024. 148с.
10. Ізмайлова О. В. Проектування інформаційних систем : навч. посіб. Київ : КНУБА, 2022. 88 с.
11. Анісімов А. В., Кулябко П. П. Інформаційні системи та бази даних : навч. посіб. для студентів факультету комп'ютерних наук та кібернетики. Київ, 2017. 110с.
12. Інформаційні технології : навч. посіб. / О. І. Зачек, В. В. Сенік, Т. В. Магеровська та ін. ; за ред. О. І. Зачека. Львів : Львів. держ. ун-т внутрішніх справ, 2022. 432с.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		81

13. Казакова Н. Ф., Штефан Н. З. Системний аналіз і проектування інформаційних систем : метод. вказівки до лаборатор. занять (ч. 2). Одеса, 2023. 134с.

14. Литвин В. В., Шаховська Н. Б. Проектування інформаційних систем : навч. посіб. Львів, 2010. 536с.

15. Магеровська Т. В., Пелех Я. М., Сенік В. В., Кунинець А. В. Електронні таблиці та системи управління базами даних : навч. посіб. Львів : Самвидав, 2020. 415 с.

16. Марченко А. В. Проектування інформаційних систем. Київ, 2016. 90с.

17. Мацулевич О., Вершков О., Антонова Г., Зюзін М. Застосування САD-системи Unigraphics для технологічної підготовки виробництва корпусних деталей // Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації : матеріали IV міжнар. наук.-практ. інтернет-конф. (Запоріжжя, 29–31 травня 2023 р.). С. 139–146.

18. Мацулевич О. Є., Чаплінський А. П. Використання методів комп'ютерного проектування при розробці технологічного процесу та управляючої програми для обладнання з ЧПУ // Сучасні комп'ютерні та інформаційні системи і технології : матеріали III Всеукр. наук.-практ. інтернет-конф. (Запоріжжя, 12–19 грудня 2022 р.). Запоріжжя : ТДАТУ, 2022. С. 28–34.

19. Методи та системи штучного інтелекту : навч. посіб. / укл. Д. В. Лубко, С. В. Шаров. Мелітополь : ФОП Однорог Т. В., 2019. 264с.

20. Мінухін С. В., Беседовський О. М., Знахур С. В. Методи і моделі проектування на основі сучасних CASE-засобів : навч. посіб. Харків : ХНЕУ, 2008. 272 с.

21. Наконечний А. Й., Верес З. Є. Інтернет речей і сучасні технології Вісн. Нац. ун-ту «Львів. політехніка». Автоматика, вимірювання та керування. 2016. № 852. С. 3–9.

22. Недашківський О. М. Планування та проектування інформаційних систем. Київ, 2014. 215 с.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						82
Змн.	Арк.	№ докум.	Підпис	Дата		

23. Маляров М., Христич В., Журавський М. Основи інформаційних технологій : курс лекцій. Харків, 2019. 184с.

24. Павлиш В. А., Гліненко Л. К., Шаховська Н. Б. Основи інформаційних технологій і систем : підручник. Львів : Вид-во Львів. політехніки, 2018. 620с.

25. Шевчук В. С., Гриценко І. В. Основи проектування інформаційних систем : навч. посіб. Київ : Центр учбової літератури, 2015. 312с.

26. Проектування інформаційних систем : загальні питання теорії проектування ІС (конспект лекцій) : навч. посіб. для студ. спец. 122 «Комп'ютерні науки» / уклад. О. С. Коваленко, Л. М. Добровська. Київ : КПІ ім. Ігоря Сікорського, 2020. 192с.

27. Ременяк Л. В. Проектування інформаційних систем : конспект лекцій. Одеса : Одес. держ. екол. ун-т, 2016. 152с.

28. Поллис Г., Огастин Л., Лоу К., Мадхар Дж. Розробка програмних проектів на основі Rational Unified Process (RUP). БіномПрес, 2011. 256 с.

29. Саченко І. А. Проектування інформаційних систем : конспект лекцій. Київ : КНУБА, 2024. 88с.

30. Ситнік Б. Т. Основи інформаційних систем і технологій : навч. посіб. Харків : УкрДУЗТ, 2019. 175с.

31. Соколов В. Ю. Інформаційні системи і технології : навч. посіб. Київ : ДУІКТ, 2010. 138с.

32. Столярова Т. О., Одинець В. А. Інформаційні системи і технології : навч. посіб. Ірпінь : Ун-т ДФС України, 2016. 160с.

33. Строкань О. В., Литвин Ю. О. Програмні рішення з підвищення ефективності управління параметрами мікроклімату у закритому приміщенні // Матеріали міжнар. наук.-практ. форуму «Сучасні наукові дослідження на шляху до Євроінтеграції» (21–22 червня 2019 р.). Мелітополь : ФОП Однорог Т. В., 2019. Ч. 2. С. 105–107.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						83
Змн.	Арк.	№ докум.	Підпис	Дата		

34. Сучасні комп'ютерні та інформаційні системи і технології : матеріали IV Всеукр. наук.-практ. інтернет-конф. (Запоріжжя, 9–20 грудня 2024 р.). Запоріжжя : ТДАТУ, 2024. 205с.

35. Томашевський О. М., Цегелик М. Б., Вітер Г. Г., Дубук В. І. Інформаційні технології та моделювання бізнес-процесів : навч. посіб. Київ : Центр учбової літератури, 2005. 296с.

36. Ушакова І. О. Основи системного аналізу об'єктів і процесів комп'ютеризації. Ч. 2 : навч. посіб. для студентів напряму "Комп'ютерні науки". Харків : ХНЕУ, 2008. 308с.

37. Шаров С. В., Осадчий В. В. Бази даних та інформаційні системи : навч. посіб. Мелітополь : МДПУ ім. Б. Хмельницького, 2014. 352 с.

38. Шаховська Н. Б., Литвин В. В. Проектування інформаційних систем : навч. посіб. Львів : Магнолія-2006, 2011. 380с.

39. Шило С. Г., Щербак Г. В., Огурцова К. В. Інформаційні системи та технології : навч. посіб. Харків : ХНЕУ, 2013. 220с.

40. Штельма О. М. Інформаційні системи і технології : конспект лекцій для здобувачів першого (бакалаврського) рівня вищої освіти всіх форм навчання зі спец. 126 – Інформаційні системи та технології. Харків : ХНУМГ ім. О. М. Бекетова, 2024. 89с.

41. Ahmed M. M., Qays M. O., Abu-Siada A., Muyeen S. M., Hossain M. L. Cost-effective design of IoT-based smart household distribution system // Designs. 2021. Vol. 5, No. 3. Article № 55. DOI: 10.3390/designs5030055.

42. Axelrod A. Complete Guide to Test Automation : Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects. NY : Apress, 2018. 588p.

43. Elkholy M. H., Senjyu T., Lotfy M. E., Elgarhy A., Ali N. S., Gaafar T. S. Design and implementation of a real-time smart home management system considering energy saving // Sustainability. 2022. Vol. 14. Article № 13840. DOI: 10.3390/su142113840.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		84

44. Hakimi S. M., Saadatmandi M., Shafie-khah M., Catalão J. P. S. Smart household management systems with renewable generation to increase the operation profit of a microgrid // IET Smart Grid. 2019. Vol. 2, No. 4. P. 522–528. DOI: 10.1049/iet-stg.2018.0299.

					КРБ.СІ-04.00.00.000 ПЗ	Арк.
						85
Змн.	Арк.	№ докум.	Підпис	Дата		

БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: “Автоматизована система дистанційного моніторингу та управління кліматичними умовами в серверному приміщенні”.

Обсяг пояснювальної записки в аркушах - 85.

Перелік графічного матеріалу:

КРБ.СІ – 04.00.00.001 – Блок-схема основного циклу роботи (аркушів - 1).

КРБ.СІ – 04.00.00.002 – Діаграма алгоритму роботи клієнтської частини (аркушів -1).

КРБ.СІ – 04.00.00.003 – Структурна схема автоматизованої системи контролю мікроклімату серверної кімнат (аркушів -1).

Дата закінчення

Бакалаврської роботи

Студент

15 червня 2025р.
