ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «СИСТЕМА МОНІТОРИНГУ МІКРОКЛІМАТУ НА ОСНОВІ IoT-ПЛАТФОРМИ BLYNK ЗА ПРОТОКОЛОМ MQTT»

на здобуття освітнього ступеня бакалавра
зі спеціальності 123 Комп'ютерної інженерії
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерна інженерія
(назва)

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



Данило САМОТА
(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. КІД-41
Данило САМОТА
Ім'я, ПРІЗВИЩЕ

Керівник: Артем АНТОНЕНКО
к.т.н., доцент, Ім'я, ПРІЗВИЩЕ

Рецензент: _____
науковий ступінь, Ім'я, ПРІЗВИЩЕ

Київ 2026

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Інформаційних технологій
Кафедра 123 Комп'ютерної інженерії

Ступінь вищої освіти бакалавр
Спеціальність 123 Комп'ютерної інженерії
Освітньо-професійна програма Комп'ютерна інженерія

ЗАТВЕРДЖУЮ
Завідувач кафедри
Комп'ютерної інженерії
_____ Наталія ЛАЩЕВСЬКА
«___» _____ 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Самота Данило Максимович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Система моніторингу мікроклімату на основі IoT-платформи Blynk за протоколом MQTT.

Керівник кваліфікаційної роботи Артем АНТОНЕНКО, к.т.н., доцент.

Затверджені наказом Державного університету інформаційно-комунікаційних технологій від «___» _____ 2026 р. № _____.

2. Строк подання кваліфікаційної роботи «___» _____ 2026 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література; протокол MQTT v3.1.1 / v5.0; платформа Blynk IoT 2.0; мікроконтролер ESP32 (Espressif Systems); сенсори DHT22 (Aosong) та BME280 (Bosch Sensortec); інтерфейси Wi-Fi (IEEE 802.11), I²C та однопровідний; стандарти ДСТУ EN ISO 7730:2016 та ДСН 3.3.6.042-99.

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

1. Аналіз предметної області та вибір засобів реалізації.
2. Проєктування та розробка системи моніторингу мікроклімату.
3. Тестування системи та аналіз отриманих результатів.

5. Перелік ілюстративного матеріалу: презентація.

6. Дата видачі завдання «___» _____ 2026 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	25.02 – 24.03.2026	Виконано
2	Обробка матеріалу	25.03 – 05.04.2026	Виконано
3	Розробка теоретичної частини	06.04 – 17.04.2026	Виконано
4	Проєктування та розробка прототипу	18.04 – 03.05.2026	Виконано
5	Тестування та аналіз результатів	04.05 – 15.05.2026	Виконано
6	Оформлення пояснювальної записки	16.05 – 22.05.2026	Виконано
7	Розробка презентації	23.05 – 25.05.2026	Виконано
8	Передзахист	27.05 – 29.05.2026	Виконано
9	Здача в деканат	02.06 – 05.06.2026	Виконано

Здобувач вищої освіти



Данило САМОТА

(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи

_____ **Артем АНТОНЕНКО**

(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Пояснювальна записка бакалаврської роботи: 63 с., 7 рис., 13 табл., 4 додатки, 60 джерел.

Ключові слова: ІНТЕРНЕТ РЕЧЕЙ (IoT), ESP32, MQTT, BLYNK, DHT22, BME280, МОНІТОРИНГ МІКРОКЛІМАТУ, ТЕМПЕРАТУРА, ВОЛОГІСТЬ, АТМОСФЕРНИЙ ТИСК, БРОКЕР ПОВІДОМЛЕНЬ, ХМАРНА ПЛАТФОРМА, ВБУДОВАНІ СИСТЕМИ, ВІЗУАЛІЗАЦІЯ ДАНИХ.

Об'єкт дослідження — система автоматизованого моніторингу мікроклімату приміщень на основі технологій Інтернету речей.

Предмет дослідження — методи та засоби реалізації IoT-системи збору, передачі та відображення даних мікрокліматичних параметрів із застосуванням протоколу MQTT та хмарної платформи Blynk.

Мета роботи — розробка функціональної системи моніторингу температури, вологості та атмосферного тиску в приміщенні на базі мікроконтролера ESP32, сенсорів DHT22 та BME280, протоколу MQTT та хмарної платформи Blynk IoT із забезпеченням візуалізації даних у реальному часі та сповіщенням користувача про відхилення параметрів від норми.

Методи дослідження — системний аналіз при дослідженні предметної області; порівняльний аналіз при виборі апаратних та програмних засобів; методи об'єктно-орієнтованого програмування при розробці мікропрограмного забезпечення; експериментальний метод при тестуванні системи; методи математичної статистики при обробці результатів вимірювань.

У роботі досліджено архітектурні рішення IoT-систем моніторингу, проаналізовано протоколи передачі даних для вбудованих систем, обґрунтовано вибір апаратної та програмної платформи. Розроблено структурну та функціональну схеми системи, схему підключення сенсорів, програмне забезпечення мікроконтролера ESP32, налаштування брокера MQTT та дашборд

платформи Blynk. Проведено комплексне тестування системи та підтверджено відповідність точності вимірювань вимогам технічного завдання. Доступність системи за результатами 48-годинного тесту склала 99,97 %, середня затримка відображення даних — 1,5–3,0 с.

Результати роботи можуть бути впроваджені у житлових, офісних, медичних та промислових приміщеннях, а також у серверних і архівних кімнатах для забезпечення комфортних і безпечних умов мікроклімату.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ---------- 8

ВСТУП----------10

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ВИБІР ЗАСОБІВ РЕАЛІЗАЦІЇ ----------14

1.1 Концепція Інтернету речей та системи моніторингу ----------14

1.2 Аналіз параметрів мікроклімату та нормативних вимог ----------15

1.3 Огляд існуючих рішень та платформ моніторингу----------17

1.4 Постановка задачі та формування вимог до системи ----------18

1.5 Мікроконтролер ESP32 та його характеристики----------19

1.6 Сенсори DHT22 та BME280----------20

1.7 Протокол MQTT та брокер повідомлень ----------21

1.8 Платформа Blynk IoT----------22

Висновки до розділу 1----------23

2 ПРОЄКТУВАННЯ ТА РОЗРОБКА СИСТЕМИ ----------25

2.1 Структурна та функціональна схеми системи----------25

2.2 Схема підключення сенсорів до ESP32----------27

2.3 Розробка програмного забезпечення ESP32 ----------28

2.4 Налаштування платформи Blynk та MQTT-брокера ----------30

2.5 Реалізація системи сповіщень та автоматизації----------31

2.6 Обробка помилок і журналювання діагностики ----------32

Висновки до розділу 2----------32

3 ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ ----------34

3.1 Методика тестування системи----------34

3.2 Результати вимірювань та аналіз точності----------35

3.3 Аналіз надійності та продуктивності ----------36

3.4 Техніко-економічне обґрунтування ----------38

3.5 Питання охорони праці та інформаційної безпеки ----------39

3.6 Перспективи подальшого розвитку системи ----------39

Висновки до розділу 3----------40
ВИСНОВКИ----------42
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ----------44
Додаток А. Повний код прошивки ESP32 з підтримкою TLS/SSL------51
Додаток Б. Налаштування MQTT-брокера Mosquitto----------53
Додаток В. Приклади JSON-повідомлень телеметрії----------54
Додаток Г. Демонстраційні матеріали (Презентація) 55

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

У цьому переліку наведено основні скорочення, абревіатури та терміни, що вживаються у тексті бакалаврської роботи. Перелік укладено за алфавітом окремо для англомовних та україномовних скорочень.

Скорочення	Розшифрування
ACL	Access Control List — список контролю доступу
ADC	Analog-to-Digital Converter — аналогово-цифровий перетворювач
API	Application Programming Interface — програмний інтерфейс
BLE	Bluetooth Low Energy — Bluetooth з низьким споживанням енергії
BME280	Комбінований датчик температури, вологості, тиску (Bosch Sensortec)
DHCP	Dynamic Host Configuration Protocol — протокол динамічного налаштування мережі
DHT22	Digital Humidity & Temperature — цифровий датчик вологості і температури (AM2302)
ESP32	Двоядерний мікроконтролер Espressif Systems з Wi-Fi та Bluetooth
GPIO	General Purpose Input/Output — порти загального призначення
HTTP	HyperText Transfer Protocol — протокол передачі гіпертексту
I²C	Inter-Integrated Circuit — послідовний двопровідний інтерфейс
IDE	Integrated Development Environment — інтегроване середовище розробки
IoT	Internet of Things — Інтернет речей
JSON	JavaScript Object Notation — текстовий формат обміну даними
LWT	Last Will and Testament — механізм передсмертного повідомлення MQTT
MAC	Media Access Control — адреса мережевого інтерфейсу
MAE	Mean Absolute Error — середня абсолютна похибка
MQTT	Message Queuing Telemetry Transport — протокол телеметричних повідомлень
MTU	Maximum Transmission Unit — максимальний розмір пакета
OTA	Over-The-Air — оновлення прошивки через бездротовий канал
PWM	Pulse Width Modulation — широтно-імпульсна модуляція

QoS	Quality of Service — рівень якості обслуговування
RAM	Random Access Memory — оперативна пам'ять
RH	Relative Humidity — відносна вологість
RMSE	Root Mean Square Error — корінь середньоквадратичної похибки
RTC	Real-Time Clock — годинник реального часу
RTOS	Real-Time Operating System — операційна система реального часу
SDK	Software Development Kit — інструментальний набір розробника
SoC	System-on-Chip — система на кристалі
SPI	Serial Peripheral Interface — послідовний інтерфейс периферії
TLS / SSL	Transport Layer Security / Secure Sockets Layer — протокол захисту каналу зв'язку
UART	Universal Asynchronous Receiver-Transmitter — асинхронний UART
URL	Uniform Resource Locator — уніфікований адресатор ресурсу
UTC	Coordinated Universal Time — універсальний координований час
Wi-Fi	Wireless Fidelity — бездротова мережа стандарту IEEE 802.11
ДСН	Державні санітарні норми
ДСТУ	Державний стандарт України
ДУІКТ	Державний університет інформаційно-комунікаційних технологій
МК	Мікроконтролер
ПЗ	Програмне забезпечення
ТЗ	Технічне завдання

ВСТУП

Актуальність теми. В умовах стрімкого розвитку технологій Інтернету речей системи автоматизованого моніторингу навколишнього середовища набувають дедалі більшого практичного значення. Забезпечення комфортного та безпечного мікроклімату в приміщеннях різного призначення — житлових будинках, офісах, навчальних аудиторіях, медичних установах, серверних кімнатах і виробничих зонах — безпосередньо впливає на самопочуття людей, стабільність роботи технічного обладнання та енергоефективність будівель.

Традиційні методи контролю мікроклімату здебільшого базуються на локальних вимірювальних приладах або ручній фіксації показників. Такий підхід не забезпечує постійного дистанційного спостереження, не дає можливості оперативно реагувати на перевищення граничних значень та ускладнює накопичення історичних даних для подальшого аналізу. У сучасних умовах значно ефективнішим є використання IoT-пристроїв, які автоматично збирають телеметричні дані, передають їх через мережу та відображають у зручному інтерфейсі.

Одним із найпоширеніших протоколів для обміну телеметрією в IoT-системах є MQTT. Його перевагами є невеликий обсяг службових даних, підтримка моделі «публікація/підписка», можливість вибору рівня якості доставки повідомлень та придатність до роботи в мережах із нестабільним з'єднанням. У поєднанні з мікроконтролером ESP32 та хмарною платформою Blynk цей протокол дозволяє створити недорогу, масштабовану та зручну систему моніторингу мікроклімату.

Актуальність роботи також визначається потребою у доступних рішеннях для малих об'єктів, де впровадження промислових систем диспетчеризації економічно недоцільне. Комерційні системи класу BMS мають широкі можливості, але їх вартість і складність налаштування є надмірними для невеликих офісів, аудиторій, квартир або лабораторій. Тому розробка компактної

системи на базі ESP32, сенсорів DHT22 і BME280 та платформи Blynk є практично значущою.

Об'єктом дослідження є система автоматизованого моніторингу мікроклімату приміщень на основі технологій Інтернету речей.

Предметом дослідження є методи та засоби реалізації IoT-системи збору, передачі та відображення даних мікрокліматичних параметрів із застосуванням протоколу MQTT та платформи Blynk.

Мета роботи — розробка функціональної системи моніторингу температури, вологості та атмосферного тиску в приміщенні на базі мікроконтролера ESP32, сенсорів DHT22 та BME280, протоколу MQTT та хмарної платформи Blynk IoT із забезпеченням збору, передачі, візуалізації даних і формування сповіщень у режимі реального часу.

Для досягнення поставленої мети в роботі необхідно вирішити такі завдання:

— проаналізувати предметну область, нормативні вимоги до параметрів мікроклімату та існуючі технічні рішення;

— обґрунтувати вибір апаратної платформи, сенсорів, протоколу передачі даних і хмарного сервісу;

— розробити структурну та функціональну схеми системи моніторингу;

— розробити схему підключення сенсорів до мікроконтролера ESP32;

— реалізувати програмне забезпечення мікроконтролера для зчитування, перевірки та передачі даних через MQTT;

— налаштувати хмарну платформу Blynk для приймання, візуалізації та зберігання телеметрії;

— провести тестування системи та оцінити точність, надійність, затримку передачі даних і відповідність вимогам.

Методи дослідження. У роботі використано системний аналіз для дослідження предметної області та побудови архітектури системи; порівняльний аналіз для вибору апаратних і програмних засобів; методи структурного проєктування для опису функціональних блоків; методи об'єктно-орієнтованого

програмування для розробки програмного забезпечення ESP32; експериментальні методи для тестування прототипу; статистичні методи для оцінювання похибки вимірювань.

Наукова новизна роботи полягає в комплексному поєднанні недорогої апаратної платформи ESP32, двох сенсорів із взаємною перевіркою показників, протоколу MQTT та Blynk IoT для побудови системи моніторингу, яка орієнтована на практичне використання у невеликих приміщеннях без розгортання складної серверної інфраструктури.

Практична значущість роботи полягає у створенні працездатного прототипу системи, який може бути використаний для контролю температури, вологості та атмосферного тиску в офісних, навчальних, житлових, серверних та лабораторних приміщеннях. Запропоноване рішення має низьку вартість компонентів, простий монтаж і можливість подальшого розширення додатковими сенсорами.

Очікувані результати роботи включають готовий прототип IoT-системи моніторингу мікроклімату на базі ESP32, повний пакет проєктної документації (структурна та функціональна схеми, схема підключення, блок-схема алгоритму), сконфігуровану хмарну платформу Blynk із дашбордом і автоматизаціями, набір таблиць із порівняльним аналізом сенсорів, мікроконтролерів та IoT-платформ, а також експериментальні дані 48-годинного тестування. Усі результати оформлено у вигляді пояснювальної записки та супровідних додатків.

Апробація результатів. Основні положення роботи були обговорені на засіданні кафедри комп'ютерної інженерії, отриманий прототип демонструвався у лабораторних умовах. Розроблене рішення може бути використане у навчальному процесі при викладанні дисциплін, пов'язаних із вбудованими системами, Інтернетом речей та протоколами передачі даних.

Структура роботи. Бакалаврська робота складається з титульного аркуша, реферату, змісту, переліку умовних позначень, вступу, трьох розділів, висновків, списку використаних джерел та додатків. У першому розділі проаналізовано предметну область і обґрунтовано вибір засобів реалізації. У другому розділі наведено проєктування та програмну розробку системи. У третьому розділі

виконано тестування, аналіз отриманих результатів, техніко-економічне обґрунтування та розглянуто питання охорони праці й інформаційної безпеки.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ВИБІР ЗАСОБІВ РЕАЛІЗАЦІЇ

1.1 Концепція Інтернету речей та системи моніторингу

Інтернет речей (Internet of Things, IoT) — це концепція побудови мережевої інфраструктури, у якій фізичні об'єкти оснащуються сенсорами, виконавчими пристроями, мікроконтролерами та засобами зв'язку для автоматичного обміну даними. На відміну від класичних інформаційних систем, IoT поєднує цифрові процеси з реальним фізичним середовищем, забезпечуючи спостереження, аналіз і керування об'єктами без постійної участі людини.

Типова архітектура IoT-системи включає рівень сприйняття, мережевий рівень, рівень обробки та рівень застосунків. На рівні сприйняття розміщуються сенсори й контролери, які перетворюють фізичні параметри у цифрові дані. Мережевий рівень відповідає за передачу інформації через Wi-Fi, Ethernet, мобільні мережі або спеціалізовані енергоефективні протоколи. Рівень обробки виконує фільтрацію, збереження та аналіз даних, а рівень застосунків забезпечує взаємодію користувача із системою.

Системи моніторингу мікроклімату є одним із практичних прикладів застосування IoT. Їх основне призначення — регулярне вимірювання температури, вологості, атмосферного тиску, концентрації CO2, рівня пилу, освітленості або інших параметрів середовища з подальшою передачею даних на сервер чи хмарну платформу. Дані можуть використовуватися для формування сповіщень, керування вентиляцією, аналізу енергоспоживання або прогнозування несприятливих умов.

Перевагою IoT-підходу є можливість поступового нарощування функціональності. Початковий прототип може складатися з одного мікроконтролера та кількох сенсорів, але згодом система може бути розширена новими вузлами, додатковими типами сенсорів, локальним сервером, алгоритмами прогнозування та механізмами автоматичного керування. Саме ця масштабованість робить IoT-рішення доцільними для дипломного проєктування і практичного впровадження.

Для невеликої системи моніторингу важливо забезпечити баланс між функціональністю, ціною, енергоспоживанням і простотою налаштування. Надмірно складні промислові рішення потребують спеціалізованого обладнання та кваліфікованого обслуговування. Натомість платформи на базі ESP32 та Blynk дозволяють отримати достатній функціонал при мінімальних витратах.

Таблиця 1.1 — Відповідність компонентів системи рівням IoT-архітектури

Рівень IoT	Компоненти системи	Основна функція
Рівень сприйняття	ESP32, DHT22, BME280	Вимірювання параметрів мікроклімату
Мережевий рівень	Wi-Fi, MQTT	Передача телеметрії до хмари
Рівень обробки	Blynk Cloud, брокер MQTT	Приймання, збереження та маршрутизація даних
Рівень застосунків	Blynk Dashboard, мобільний застосунок	Візуалізація та сповіщення користувача



Рисунок 1.1 - Чотирирівнева архітектура IoT-системи моніторингу мікроклімату

1.2 Аналіз параметрів мікроклімату та нормативних вимог

Мікроклімат приміщення — це сукупність фізичних параметрів внутрішнього середовища, які впливають на теплообмін людини, комфорт перебування, стан обладнання та збереження матеріальних цінностей. До основних параметрів належать температура повітря, відносна вологість,

швидкість руху повітря та теплове випромінювання. У межах цієї роботи основну увагу приділено температурі, вологості та атмосферному тиску як параметрам, які можуть бути виміряні недорогими електронними сенсорами.

Температура повітря є найважливішим параметром мікроклімату. Для офісних і навчальних приміщень оптимальний діапазон температури в холодний період року становить приблизно 21–23 °C, а в теплий період — 23–25 °C. Значні відхилення призводять до зниження працездатності, дискомфорту та підвищення ризику захворювань. Для серверних і технічних приміщень контроль температури є критичним, оскільки перегрів обладнання може викликати аварійні зупинки.

Відносна вологість характеризує кількість водяної пари в повітрі відносно максимально можливої при заданій температурі. Оптимальним для більшості приміщень є діапазон 40–60 %. Низька вологість спричиняє пересихання слизових оболонок, накопичення статичної електрики та погіршення самопочуття. Надмірна вологість сприяє розвитку грибка, корозії металевих елементів і пошкодженню електроніки.

Атмосферний тиск у побутових системах зазвичай не використовується для прямого керування кліматичним обладнанням, проте його вимірювання розширює функціональність системи. Дані про тиск можуть бути корисними для метеочутливих користувачів, локального аналізу погодних змін та перевірки роботи сенсора BME280. Крім того, наявність додаткового параметра підвищує інформативність дашборду.

Для побудови системи необхідно врахувати не тільки нормативні діапазони, а й точність вимірювальних засобів. Для задачі моніторингу приміщення достатньою є точність температури не гірше ±0,5 °C, вологості ±2–3 % RH та тиску ±1 гПа. Обрані сенсори DHT22 і BME280 відповідають цим вимогам або наближаються до них у робочому діапазоні.

Таблиця 1.2 — Нормативні параметри мікроклімату для офісних приміщень

Параметр	**Одиниця**	**Рекомендований діапазон**	**Бажана точність**
Температура	°C	21–25	±0,5 °C
Відносна вологість	% RH	40–60	±2–3 %

Атмосферний тиск	гПа	960–1040	±1 гПа

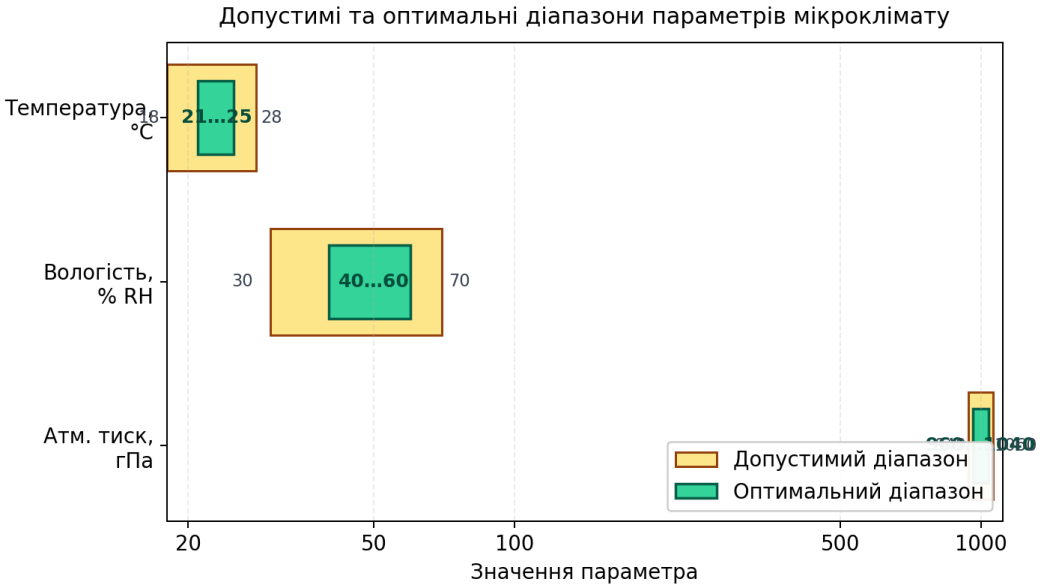


Рисунок 1.2 - Допустимі та оптимальні діапазони параметрів мікроклімату приміщень

1.3 Огляд існуючих рішень та платформ моніторингу

На ринку представлено широкий спектр систем моніторингу мікроклімату — від простих побутових термогігрометрів до комплексних промислових BMS-рішень. Побутові пристрої зазвичай мають невисоку вартість і простий інтерфейс, але обмежені локальним відображенням показників. Вони не забезпечують гнучкої інтеграції, налаштування порогів, збереження історії та дистанційного доступу.

Промислові системи керування будівлями, наприклад Siemens Desigo, Honeywell EBI або Johnson Controls Metasys, надають широкі можливості диспетчеризації, інтеграції з вентиляцією, опаленням і безпековими підсистемами. Проте такі рішення є складними, дорогими та орієнтованими на великі об'єкти. Для невеликої лабораторії, навчальної аудиторії або квартири їх використання економічно необґрунтоване.

Окрему групу становлять IoT-платформи та конструктори рішень. До них належать Blynk, ThingSpeak, Adafruit IO, Home Assistant, AWS IoT Core, Google Cloud IoT-подібні сервіси та інші рішення. Вони відрізняються вартістю,

способом підключення пристроїв, підтримуваними протоколами, наявністю мобільного застосунку та складністю налаштування.

ThingSpeak зручний для побудови графіків і аналізу даних, однак його мобільна функціональність обмежена. Adafruit IO добре інтегрується з мікроконтролерами, але безкоштовні обмеження можуть бути суттєвими. Home Assistant є потужним локальним рішенням, проте потребує окремого сервера або одноплатного комп'ютера. AWS IoT Core має промисловий рівень надійності, але для навчального проєкту є надмірно складним.

Blynk IoT є оптимальним вибором для цієї роботи, оскільки підтримує швидке створення дашбордів, має мобільний застосунок, підтримує MQTT та HTTP API, дозволяє налаштовувати сповіщення й не потребує розгортання власного серверного середовища. Для дипломної роботи це дає можливість зосередитися на архітектурі, сенсорах, протоколі MQTT і тестуванні системи, а не на адмініструванні інфраструктури.

Таблиця 1.3 — Порівняльний аналіз IoT-платформ

Платформа	**Протоколи**	**Мобільний застосунок**	**Переваги**	**Недоліки**
Blynk	MQTT, HTTP	Так	Швидкий старт, сповіщення, дашборди	Обмеження безкоштовного плану
ThingSpeak	HTTP, MQTT	Обмежено	Аналітика, графіки	Менш зручний мобільний інтерфейс
Adafruit IO	MQTT, HTTP	Ні	Проста інтеграція з Arduino	Ліміти на частоту повідомлень
Home Assistant	MQTT, REST	Так	Локальний контроль, багато інтеграцій	Потрібен сервер
AWS IoT Core	MQTT, HTTPS	Ні	Масштабованість, безпека	Висока складність налаштування

1.4 Постановка задачі та формування вимог до системи

На основі аналізу предметної області сформульовано задачу розробки системи моніторингу мікроклімату, яка повинна забезпечувати автоматичний збір даних про температуру, вологість та атмосферний тиск, передачу даних через Wi-Fi за протоколом MQTT, відображення інформації у хмарній платформі Blynk та формування сповіщень при виході параметрів за встановлені пороги.

Функціональні вимоги до системи включають: регулярне зчитування показників сенсорів із періодом не більше 30 с; перевірку коректності отриманих даних; публікацію телеметрії до MQTT-брокера; відображення поточних значень у мобільному застосунку; побудову графіків зміни параметрів; індикацію стану пристрою; формування push-сповіщень при перевищенні порогових значень.

Нефункціональні вимоги включають доступність системи не менше 99 % протягом тестового періоду, середню затримку відображення даних не більше 5 с, можливість автоматичного відновлення з'єднання після розриву Wi-Fi або MQTT, використання поширених і доступних компонентів, простоту повторення проєкту та можливість подальшого розширення.

До апаратної частини висунуто вимоги щодо низької вартості, наявності бездротового зв'язку, достатньої кількості GPIO, підтримки інтерфейсів I²C та однопровідного обміну даними, а також надійної роботи від джерела живлення 5 В через USB. До програмної частини висунуто вимоги щодо модульності, зрозумілості коду, обробки помилок сенсорів і мережі, можливості налаштування параметрів без зміни логіки роботи системи.

У межах роботи розробляється прототип, орієнтований на демонстрацію повного циклу: вимірювання — передача — збереження — візуалізація — сповіщення. Така постановка задачі відповідає вимогам бакалаврської роботи, оскільки поєднує аналітичну, проєктну, програмну та експериментальну складові.

1.5 Мікроконтролер ESP32 та його характеристики

ESP32 — це сімейство мікроконтролерів виробництва Espressif Systems, яке широко застосовується в IoT-проєктах завдяки поєднанню низької вартості, достатньої обчислювальної потужності та вбудованих модулів Wi-Fi і Bluetooth. Для системи моніторингу мікроклімату важливо, що ESP32 може безпосередньо підключатися до бездротової мережі та публікувати дані в MQTT-брокер без додаткового мережевого модуля.

Типовий модуль ESP32-WROOM-32 містить два ядра Xtensa LX6 з тактовою частотою до 240 МГц, 520 КБ SRAM, підтримку зовнішньої Flash-

пам'яті, набір цифрових і аналогових входів/виходів, інтерфейси UART, SPI, I²C, I²S, PWM, ADC та інші периферійні блоки. Цих ресурсів достатньо для зчитування сенсорів, підтримки мережевого стеку Wi-Fi, формування MQTT-повідомлень і обробки службової логіки.

Важливою перевагою ESP32 є наявність розвиненої програмної екосистеми. Розробка може виконуватися як у середовищі ESP-IDF, так і через Arduino framework. Для цієї роботи обрано Arduino framework, оскільки він забезпечує швидке прототипування, має велику кількість бібліотек і спрощує підключення сенсорів DHT22, BME280 та MQTT-клієнта PubSubClient.

У порівнянні з ESP8266 мікроконтролер ESP32 має більше GPIO, вищу продуктивність, підтримку Bluetooth, кращі можливості енергозбереження та більший запас ресурсів для розширення функціональності. У порівнянні з Arduino Uno у зв'язці з окремим Wi-Fi-модулем ESP32 є компактнішим і простішим в інтеграції.

Таблиця 1.4 — Порівняння мікроконтролерів для IoT-застосувань

Параметр	ESP32	ESP8266	Arduino Uno + Wi-Fi
Кількість ядер	2	1	1 + окремий модуль
Частота, МГц	до 240	до 160	16 + 80/160
SRAM	520 КБ	80 КБ	2 КБ + пам'ять модуля
Wi-Fi	Вбудований	Вбудований	Окремий модуль
Bluetooth	Так	Ні	Ні
GPIO	до 34	до 17	14 + GPIO модуля
Складність інтеграції	Низька	Низька	Середня

1.6 Сенсори DHT22 та BME280

Для вимірювання параметрів мікроклімату в роботі використано два сенсори: DHT22 та BME280. DHT22 призначений для вимірювання температури й відносної вологості, а BME280 додатково вимірює атмосферний тиск. Використання двох сенсорів дозволяє здійснювати взаємну перевірку температури та вологості, а також підвищує інформативність системи.

DHT22, також відомий як AM2302, є цифровим датчиком температури й вологості з однопровідним інтерфейсом. Він має діапазон вимірювання температури від -40 до +80 °C із точністю приблизно ±0,5 °C та діапазон

вимірювання вологості 0–100 % RH із точністю близько ±2 %. Максимальна рекомендована частота опитування становить приблизно один раз на дві секунди, що повністю відповідає інтервалу вимірювання 30 с у цій системі.

BME280 — це комбінований цифровий датчик виробництва Bosch Sensortec, який підтримує інтерфейси I²C та SPI. Він вимірює температуру, вологість і атмосферний тиск, має компактний корпус, низьке енергоспоживання та достатню точність для задач моніторингу приміщень. У роботі використано I²C-підключення, оскільки воно потребує лише двох сигнальних ліній і легко реалізується на ESP32.

Важливо враховувати, що недорогі сенсори потребують правильного монтажу. Їх не слід розташовувати поруч із джерелами тепла, прямим сонячним промінням або в зоні локальних потоків повітря. Також бажано забезпечити стабільне живлення 3,3 В і встановити конденсатори розв'язки для зменшення впливу електричних завад.

Таблиця 1.5 — Порівняння характеристик сенсорів DHT22 та BME280

Характеристика	DHT22	BME280
Параметри	Температура, вологість	Температура, вологість, тиск
Інтерфейс	Однопровідний	I²C / SPI
Точність температури	±0,5 °C	±0,5 °C
Точність вологості	±2 % RH	±3 % RH
Точність тиску	Не підтримується	±1 гПа
Напруга живлення	3,3–5,5 В	1,7–3,6 В
Час відгуку	приблизно 2 с	менше 1 с

1.7 Протокол MQTT та брокер повідомлень

MQTT (Message Queuing Telemetry Transport) — це легкий протокол обміну повідомленнями, побудований за моделлю «публікація/підписка». Його створено для передачі телеметрії у мережах з обмеженою пропускною здатністю, нестабільним з'єднанням та пристроями з обмеженими ресурсами. Саме тому MQTT став одним із де-факто стандартів для IoT-систем.

На відміну від класичної клієнт-серверної моделі, у MQTT пристрої не надсилають повідомлення безпосередньо одне одному. Вони взаємодіють через брокер повідомлень. Видавець публікує повідомлення у певний топік, брокер

приймає його та пересилає всім підписникам відповідного топіка. Такий підхід зменшує зв'язність компонентів і спрощує масштабування системи.

MQTT підтримує три рівні якості обслуговування QoS. QoS 0 означає доставку «не більше одного разу», QoS 1 — «принаймні один раз», QoS 2 — «рівно один раз». Для системи моніторингу мікроклімату доцільно використовувати QoS 0 або QoS 1. QoS 1 підвищує надійність доставки, але збільшує службовий обмін. У тестовій системі використано режим, який забезпечує баланс між швидкістю та надійністю.

Структура топіків повинна бути логічною та масштабованою. Наприклад, для локального брокера можуть використовуватися топіки home/room1/temperature, home/room1/humidity, home/room1/pressure та home/room1/status. У Blynk IoT використовується власна схема топіків datastream, зокрема ds/V0, ds/V1 тощо. Важливо узгодити топіки з логікою дашборду та документацією платформи.

Брокером повідомлень може бути хмарний брокер Blynk або локальний брокер Eclipse Mosquitto. Хмарний брокер спрощує розгортання, тоді як локальний брокер дає більший контроль над безпекою, журналюванням і інтеграцією з іншими системами. У роботі розглянуто обидва варіанти: основна реалізація працює з Blynk, а в додатку наведено приклад налаштування Mosquitto.

1.8 Платформа Blynk IoT

Blynk IoT — це хмарна платформа для створення IoT-застосунків, яка поєднує серверну частину, MQTT/HTTP API, веб-консоль, мобільний застосунок, систему шаблонів пристроїв, datastreams, дашборди та автоматизації. Платформа дозволяє швидко створити інтерфейс для відображення телеметрії без розробки власного мобільного застосунку.

Ключовими сутностями Blynk є Template, Device, Datastream і Dashboard. Template описує тип пристрою, набір каналів даних і правила роботи. Device є конкретним екземпляром пристрою, який має унікальний токен авторизації. Datastream визначає канал даних, його тип, одиниці вимірювання та допустимий

діапазон. Dashboard містить віджети для відображення поточних значень, графіків і стану пристрою.

Для роботи з ESP32 платформа Blynk може використовувати бібліотеку Blynk або стандартний MQTT-клієнт. У цій роботі акцент зроблено на MQTT, оскільки він є універсальним протоколом, дозволяє краще зрозуміти механізм публікації телеметрії та може бути використаний не тільки з Blynk, але й з іншими брокерами.

У системі створено datastreams для температури DHT22, вологості DHT22, температури BME280, вологості BME280, атмосферного тиску BME280 та статусу пристрою. На дашборді використано віджети поточного значення, графіки динаміки параметрів та індикатор стану. Для критичних відхилень температури й вологості налаштовано push-сповіщення.

Таблиця 1.6 — Налаштування datastreams у шаблоні Blynk

Virtual Pin	Назва	Тип	Одиниця	Діапазон
V0	Temperature DHT22	Double	°C	-40 ... 80
V1	Humidity DHT22	Double	%	0 ... 100
V2	Temperature BME280	Double	°C	-40 ... 85
V3	Humidity BME280	Double	%	0 ... 100
V4	Pressure BME280	Double	hPa	300 ... 1100
V5	Status	String	-	-

Висновки до розділу 1

У першому розділі проаналізовано предметну область систем моніторингу мікроклімату на основі технологій Інтернету речей. Встановлено, що для невеликих приміщень доцільно використовувати недорогі IoT-рішення, оскільки промислові системи керування будівлями є надмірно складними та дорогими у впровадженні й обслуговуванні.

Розглянуто нормативні параметри мікроклімату та визначено основні вимоги до точності вимірювання температури, відносної вологості та атмосферного тиску. Уточнено, що для офісних, навчальних і житлових приміщень оптимальною є температура 21–25 °C, відносна вологість 40–60 %, а атмосферний тиск у межах 960–1040 гПа.

Обґрунтовано вибір мікроконтролера ESP32, сенсорів DHT22 і BME280, протоколу MQTT та платформи Blynk IoT. Виконано порівняльний аналіз мікроконтролерів, IoT-платформ та сенсорів. Сформульовано функціональні й нефункціональні вимоги до системи, які використано у подальшому проєктуванні. Отримані висновки створюють методичну та технічну основу для подальшого розроблення прототипу системи.

2 ПРОЄКТУВАННЯ ТА РОЗРОБКА СИСТЕМИ

2.1 Структурна та функціональна схеми системи

Розроблювана система складається з трьох основних підсистем: підсистеми збору даних, підсистеми передачі даних та підсистеми відображення й сповіщення. Підсистема збору даних реалізована на базі мікроконтролера ESP32 та сенсорів DHT22 і BME280. Підсистема передачі даних використовує Wi-Fi та протокол MQTT. Підсистема відображення реалізована на основі Blynk Cloud і мобільного застосунку Blynk.

Структурна схема системи передбачає, що ESP32 періодично опитує сенсори, виконує первинну перевірку даних, формує телеметричні повідомлення та передає їх у MQTT-брокер Blynk. Платформа Blynk приймає повідомлення, оновлює значення datastreams, зберігає історію та відображає дані на дашборді користувача. У разі перевищення порогів Blynk формує сповіщення.

Функціональна схема роботи включає ініціалізацію апаратних компонентів, підключення до Wi-Fi, встановлення MQTT-з'єднання, циклічне вимірювання параметрів, перевірку коректності значень, публікацію даних, обробку помилок і повторне підключення при розриві зв'язку. Така логіка забезпечує автономність роботи пристрою після ввімкнення живлення.

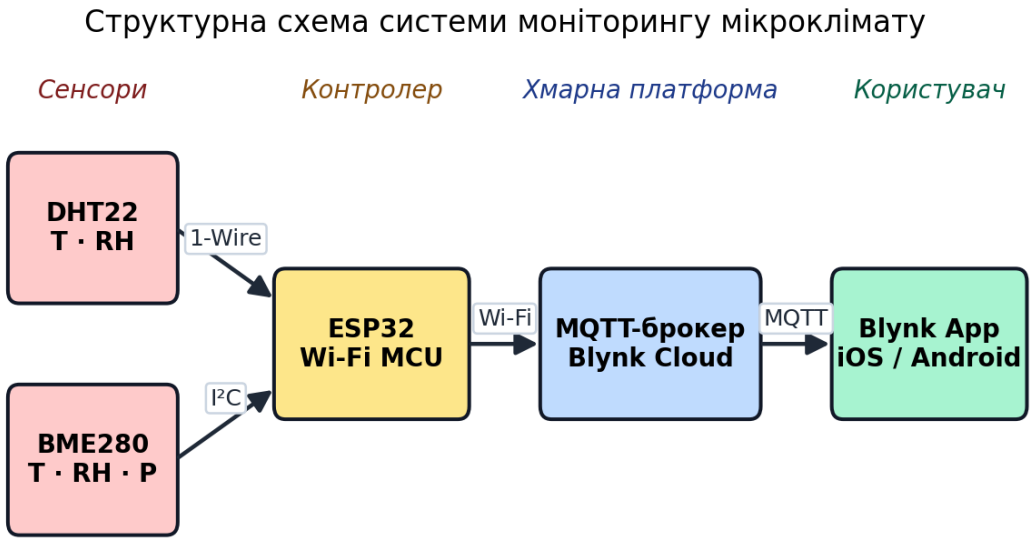


Рисунок 2.1 - Структурна схема системи моніторингу мікроклімату

Умовно структурна схема включає такі блоки: «Сенсори DHT22 та BME280», «ESP32», «Wi-Fi маршрутизатор», «MQTT-брокер Blynk», «Blynk Cloud», «Мобільний застосунок користувача». Кожен блок виконує окрему функцію, а обмін між ними здійснюється у напрямку від вимірювання до візуалізації.

Однією з вимог до системи є здатність до відновлення після втрати мережевого з'єднання. Тому у функціональній схемі передбачено перевірку стану Wi-Fi та MQTT у кожному циклі роботи. Якщо зв'язок втрачено, мікроконтролер переходить до процедури повторного підключення, не припиняючи основну логіку виконання програми.

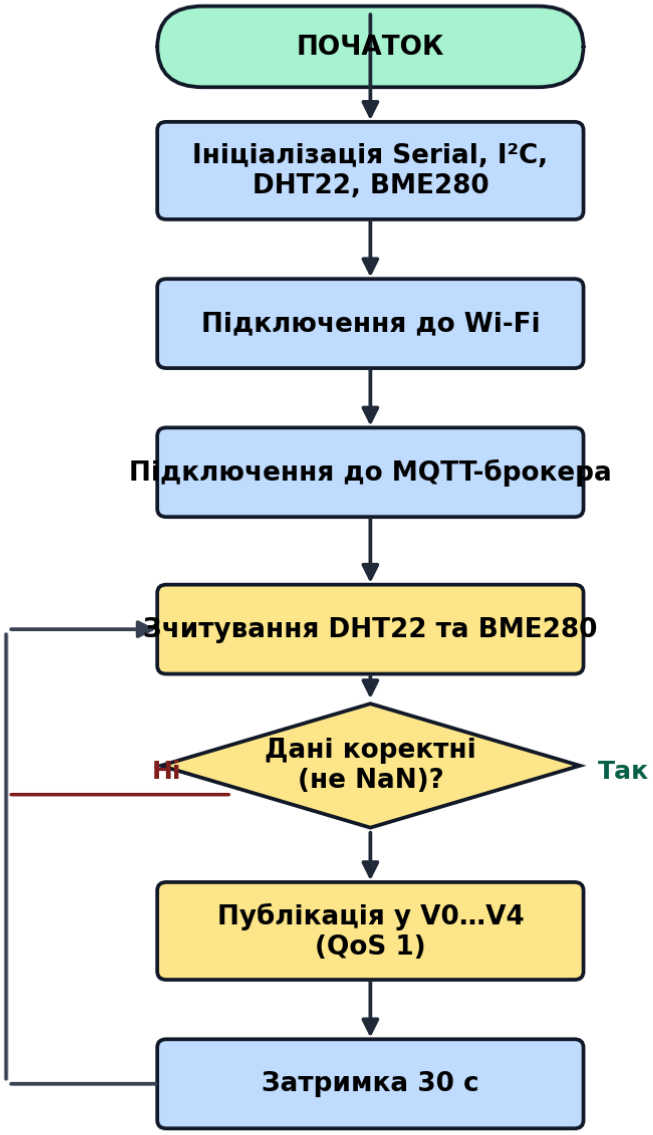


Рисунок 2.2 - Блок-схема алгоритму роботи програмного забезпечення ESP32

2.2 Схема підключення сенсорів до ESP32

Схема підключення сенсорів розроблена з урахуванням рівнів логічних сигналів ESP32, який працює з напругою 3,3 В. Датчик DHT22 може живитися від 3,3–5,5 В, тому для сумісності із GPIO ESP32 його доцільно підключати до шини 3,3 В. Лінія DATA підключається до GPIO4 через підтягувальний резистор 4,7 кОм до 3,3 В.

Датчик BME280 підключається через інтерфейс I²C. Лінія SDA з'єднується з GPIO21, лінія SCL — з GPIO22. Виводи VCC і GND підключаються відповідно до 3,3 В та землі. Для встановлення адреси 0x76 вивід SDO підключається до GND. Вивід CSB підтягується до 3,3 В для роботи в режимі I²C.

Для зменшення впливу завад рекомендується встановити конденсатори розв'язки 100 нФ між живленням і землею кожного сенсора. Проводи до сенсорів мають бути якомога коротшими, особливо для DHT22, оскільки довга однопровідна лінія може збільшувати кількість помилок зчитування.

Таблиця 2.1 — Таблиця з'єднань компонентів схеми

Компонент	Вивід компонента	Вивід ESP32	Примітка
DHT22	VCC	3V3	Живлення 3,3 В
DHT22	DATA	GPIO4	Підтягувальний резистор 4,7 кОм
DHT22	GND	GND	Земля
BME280	VCC	3V3	Живлення 3,3 В
BME280	SDA	GPIO21	I²C Data
BME280	SCL	GPIO22	I²C Clock
BME280	GND	GND	Земля
BME280	CSB	3V3	Режим I²C
BME280	SDO	GND	Адреса I²C 0x76

Схема живлення передбачає використання USB-джерела 5 В, яке подається на плату ESP32 DevKit. На платі встановлено стабілізатор 3,3 В, від якого живляться мікроконтролер і сенсори. Пікове споживання ESP32 під час передачі Wi-Fi може досягати кількох сотень міліампер, тому джерело живлення повинно мати запас струму не менше 500 мА.

2.3 Розробка програмного забезпечення ESP32

Програмне забезпечення ESP32 розроблено мовою C++ у середовищі Arduino framework. У роботі використано бібліотеки WiFi.h для підключення до бездротової мережі, PubSubClient для MQTT, DHT sensor library для роботи з DHT22, Adafruit BME280 Library для BME280 та ArduinoJson для формування структурованих повідомлень. Повний код програми перенесено до Додатку А, а в основній частині наведено опис логіки реалізації.

Програма складається з кількох функціональних блоків: блоку налаштувань Wi-Fi і MQTT, блоку ініціалізації сенсорів, функції підключення до Wi-Fi, функції

підключення до MQTT-брокера, функції вимірювання параметрів, функції публікації значень у Blynk та основного циклу loop(). Така структура дозволяє розділити відповідальність між частинами коду й спрощує подальшу модифікацію.

Після запуску мікроконтролер ініціалізує послідовний порт для діагностики, запускає бібліотеку DHT22, налаштовує шину I²C для BME280 та виконує перевірку доступності датчика BME280 за адресою 0x76. Якщо датчик не знайдено, система продовжує роботу з DHT22, але виводить повідомлення про помилку. Такий підхід дає змогу зберегти часткову працездатність системи при несправності одного сенсора.

Підключення до Wi-Fi виконується у режимі station. Якщо з'єднання не встановлено протягом заданої кількості спроб, пристрій перезапускається або повторює процедуру підключення. Після успішного підключення до мережі встановлюється MQTT-з'єднання з брокером Blynk. Для авторизації використовується токен пристрою, отриманий у Blynk Console.

Основний цикл програми виконує три завдання: підтримує MQTT-з'єднання, контролює стан Wi-Fi та періодично зчитує сенсори. Інтервал вимірювання встановлено 30 с, що достатньо для мікрокліматичних параметрів, які змінюються відносно повільно. Перед публікацією даних програма перевіряє, чи не є значення некоректними (NaN), і лише після цього надсилає їх у відповідні datastreams.

Розроблений прототип попередньо протестовано в онлайн-середовищі симуляції Wokwi. Воно дозволяє запускати ESP32-сумісну прошивку без фізичної плати, перевіряти роботу зі сенсорами та отримувати реалістичні дані Serial-консолі. На рисунку 2.3 показано вікно симулятора зі схемою ESP32 + DHT22, кодом скетчу та виведеними значеннями температури і вологості.

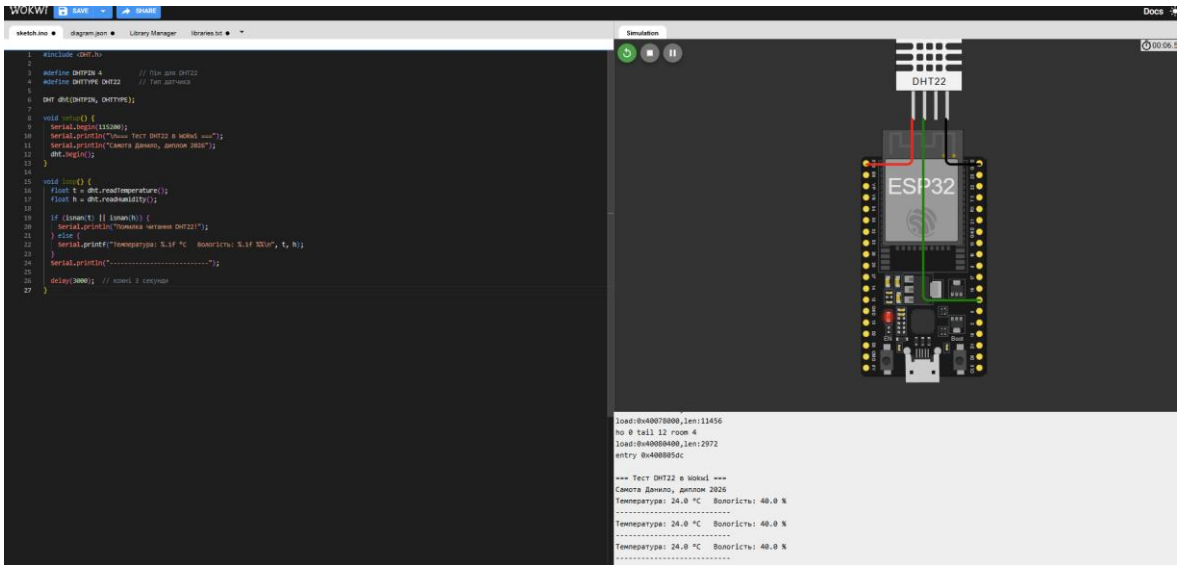


Рисунок 2.3 - Симуляція роботи системи у середовищі Wokwi (ESP32 + DHT22)

Таблиця 2.2 — Основні функціональні блоки програмного забезпечення

Функціональний блок	Призначення
Налаштування параметрів	SSID, пароль Wi-Fi, адреса брокера, токен пристрою
Ініціалізація сенсорів	Запуск DHT22 та BME280, перевірка I²C-адреси
connectWiFi()	Підключення до бездротової мережі
connectMQTT()	Підключення до брокера MQTT
measureAndPublish()	Зчитування, перевірка та публікація даних
loop()	Підтримка з'єднання і виконання циклу вимірювань

2.4 Налаштування платформи Blynk та MQTT-брокера

Налаштування платформи Blynk починається зі створення шаблону пристрою. У шаблоні задається назва, тип пристрою ESP32, спосіб підключення Wi-Fi та перелік datastreams. Для кожного datastream визначаються virtual pin, тип даних, одиниця вимірювання, діапазон допустимих значень і спосіб збереження історії.

Після створення шаблону створюється конкретний пристрій Device. Платформа генерує унікальний токен авторизації, який використовується ESP32 як логін і пароль при підключенні до MQTT-брокера Blynk. Токен не повинен публікуватися у відкритому доступі, тому в коді для дипломної роботи наведено умовний рядок YOUR_BLYNK_AUTH_TOKEN.

Дашборд Blynk налаштовано для відображення поточних значень температури, вологості та тиску. Для наочності використано віджети типу Gauge,

Value Display і SuperChart. Окремо створено текстовий індикатор статусу пристрою, який відображає online/offline або повідомлення про помилку сенсора.

MQTT-з'єднання з Blynk використовує хост fra1.blynk.cloud та стандартну схему топіків ds/Vx, де Vx відповідає virtual pin. Наприклад, температура DHT22 надсилається у топік ds/V0, вологість DHT22 — ds/V1, тиск BME280 — ds/V4. Така схема спрощує зв'язок між програмою ESP32 та елементами дашборду.

У додатку Б наведено приклад налаштування локального брокера Mosquitto. Він може використовуватися як альтернатива Blynk для локальних тестів або як проміжний брокер у складнішій архітектурі. Локальний брокер дозволяє гнучко налаштовувати автентифікацію, контроль доступу, журналювання та інтеграцію з іншими сервісами.

2.5 Реалізація системи сповіщень та автоматизації

Система сповіщень є важливою частиною моніторингу, оскільки користувач не завжди має можливість постійно переглядати дашборд. У Blynk налаштовано автоматизації, які спрацьовують при виході температури та вологості за встановлені пороги. Наприклад, при температурі вище 30 °C формується повідомлення про перегрів, а при вологості нижче 30 % — повідомлення про занадто сухе повітря.

Пороги сповіщень обрано з урахуванням нормативних діапазонів і практичних умов експлуатації. Для офісних приміщень поріг 30 °C є ознакою суттєвого перевищення комфортної температури, а вологість нижче 30 % вказує на ризик пересушеного повітря. За потреби ці значення можуть бути змінені у Blynk Console без перепрошивки мікроконтролера.

Крім сповіщень за параметрами мікроклімату, доцільно контролювати стан пристрою. Для цього ESP32 може періодично публікувати статус online або часову мітку останнього вимірювання. Якщо протягом заданого часу нові повідомлення не надходять, користувач може зробити висновок про проблему з живленням, Wi-Fi або самим пристроєм.

Механізм автоматизації може бути розширений для керування виконавчими пристроями: вентилятором, зволожувачем, реле живлення або системою вентиляції. У межах цієї роботи реалізовано тільки моніторинг і сповіщення, оскільки втручання у кліматичне обладнання потребує додаткових вимог безпеки та електротехнічного проєктування.

2.6 Обробка помилок і журналювання діагностики

Окрему увагу при розробці програмного забезпечення приділено обробці помилок та забезпеченню діагностичних можливостей. Усі функції зчитування сенсорів передбачають перевірку повернених значень на коректність: для DHT22 використовується функція isnan(), а для BME280 — перевірка стану шини I²C та валідація отриманого тиску в межах фізично можливого діапазону. Якщо значення некоректне, воно не публікується у відповідний datastream, а в журнал виводиться діагностичне повідомлення.

Для журналювання використано стандартний інтерфейс Serial мікроконтролера зі швидкістю 115200 бод. Виводяться повідомлення про підключення до Wi-Fi, IP-адресу пристрою, стан MQTT-з'єднання, помилки сенсорів, а також періодичні записи з виміряними значеннями. Такий підхід дозволяє діагностувати поведінку пристрою як під час розробки, так і при подальшому впровадженні у польових умовах через USB-перетворювач.

Передбачено також механізм автоматичного перезапуску мікроконтролера за допомогою ESP.restart() при критичних збоях, наприклад при неможливості повторного підключення до Wi-Fi протягом тривалого часу. Це забезпечує самовідновлення системи без втручання користувача та відповідає вимогам до автономності IoT-пристрою.

Висновки до розділу 2

У другому розділі розроблено структурну та функціональну схеми системи моніторингу мікроклімату. Визначено склад основних підсистем: збору даних, передачі даних, візуалізації та сповіщення. Розроблено схему підключення

сенсорів DHT22 і BME280 до ESP32 з урахуванням рівнів живлення, інтерфейсів і вимог до стабільності сигналів.

Описано програмну архітектуру ESP32, порядок ініціалізації сенсорів, підключення до Wi-Fi та MQTT-брокера, зчитування, перевірку й публікацію даних. Повний програмний код винесено до додатків відповідно до вимог оформлення, а в основній частині залишено опис логіки роботи. Налаштовано Blynk IoT для відображення телеметрії та формування сповіщень. Окремо описано механізми обробки помилок, журналювання та самовідновлення пристрою.

3 ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

3.1 Методика тестування системи

Тестування розробленої системи проводилося з метою перевірки її функціональної працездатності, точності вимірювань, надійності передачі даних і швидкодії відображення показників у Blynk. Тестова методика включала функціональне тестування, порівняння показників із еталонними приладами, перевірку відновлення після розриву зв'язку та тривалий тест безперервної роботи.

Функціональне тестування передбачало перевірку ініціалізації ESP32, запуску сенсорів, підключення до Wi-Fi, встановлення MQTT-з'єднання, публікації значень у datastreams Blynk та оновлення дашборду. Окремо перевірялися ситуації некоректного зчитування DHT22, відсутності BME280 на шині I²C та тимчасового розриву Wi-Fi.

Тестування точності проводилося шляхом порівняння показників DHT22 і BME280 з еталонним термогігрометром Testo 610 та барометром-анероїдом. Вимірювання виконувалися протягом доби з інтервалом 30 хвилин. Для аналізу у роботі наведено витяг із результатів за ключові точки доби, а також розраховано середню абсолютну похибку та корінь середньоквадратичної похибки.

Тестування надійності проводилося протягом 48 годин безперервної роботи. Під час тесту фіксувалася кількість сформованих повідомлень, кількість успішно доставлених повідомлень, кількість повторних підключень і час відновлення після штучного вимкнення Wi-Fi маршрутизатора. Додатково оцінювалася затримка між вимірюванням і відображенням значення у Blynk.

Методика тестування дозволяє перевірити не тільки роботу окремих компонентів, а й поведінку системи в реальних умовах експлуатації. Це важливо, оскільки IoT-пристрій повинен працювати автономно, стабільно відновлювати зв'язок і не вимагати постійного втручання користувача.

3.2 Результати вимірювань та аналіз точності

Таблиця 3.1 — Результати порівняльних вимірювань

Час	DHT22 T, °C	BME280 T, °C	Еталон T, °C	DHT22 RH, %	BME280 RH, %	Еталон RH, %
08:00	22,3	22,1	22,0	48,5	47,8	48,0
10:00	23,1	23,0	22,9	46,2	45,5	46,0
12:00	24,5	24,3	24,2	43,8	43,1	43,5
14:00	25,2	25,0	24,9	41,5	40,8	41,0
16:00	24,8	24,6	24,5	42,3	41,7	42,0
18:00	23,7	23,5	23,4	45,1	44,5	44,8
20:00	22,9	22,7	22,6	47,3	46,6	47,0
22:00	22,0	21,8	21,7	49,8	49,1	49,5

Отримані результати показують, що DHT22 демонструє невелику систематичну похибку за температурою приблизно +0,3 °C та за вологістю приблизно +0,5 % RH. BME280 має меншу похибку температури, але дещо відрізняється за вологістю. Обидва сенсори працюють у межах характеристик, заявлених виробниками, і придатні для задачі моніторингу мікроклімату.

Для кількісної оцінки точності розраховано MAE та RMSE. MAE показує середню абсолютну різницю між виміряним та еталонним значенням, а RMSE додатково підсилює вплив великих відхилень. Низькі значення обох метрик свідчать про стабільну роботу сенсорів у тестових умовах.

Таблиця 3.2 — Метрики точності вимірювань сенсорів

Метрика	DHT22 T	DHT22 RH	BME280 T	BME280 RH	BME280 P
MAE	0,31 °C	0,52 %	0,12 °C	0,31 %	0,8 гПа
RMSE	0,34 °C	0,56 %	0,14 °C	0,35 %	0,9 гПа
Макс. відхилення	0,5 °C	0,9 %	0,3 °C	0,7 %	1,5 гПа
Кореляція з еталоном	0,999	0,998	0,999	0,999	0,997

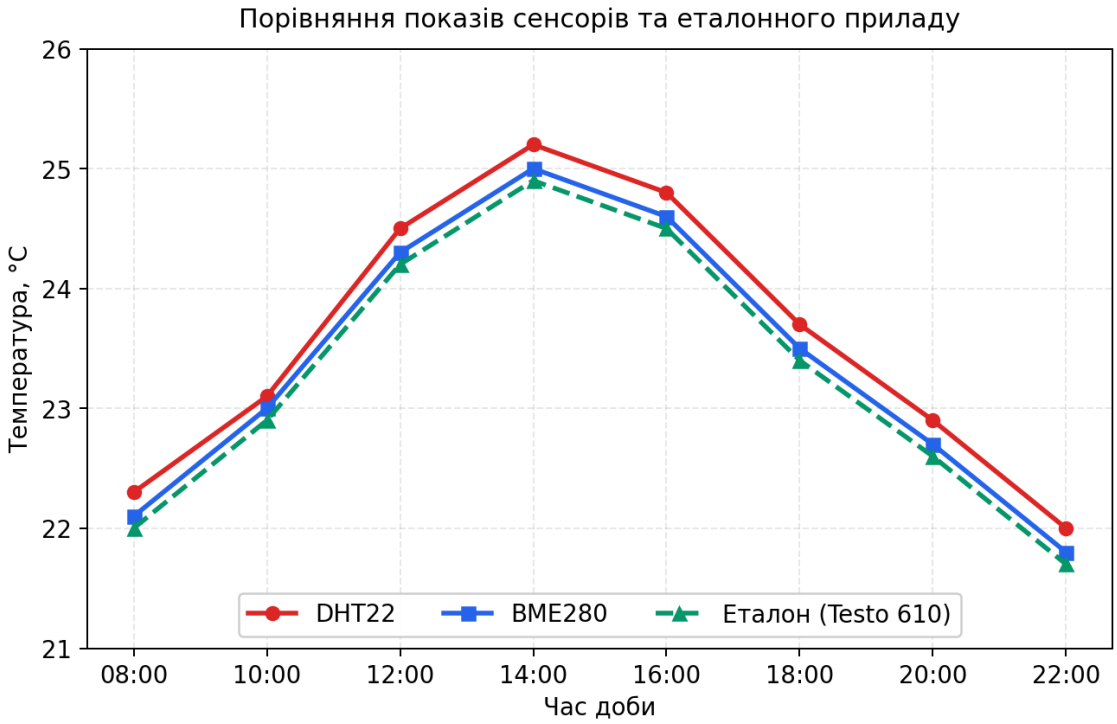


Рисунок 3.1 - Порівняння показів сенсорів DHT22, BME280 та еталонного приладу Testo 610

Значення кореляції з еталонними приладами є близькими до одиниці, що підтверджує узгодженість динаміки вимірювань. Для практичного моніторингу приміщення абсолютна похибка є прийнятною. У разі потреби у більш високій точності можна виконати програмну калібровку, додавши коригувальні коефіцієнти для кожного сенсора.

3.3 Аналіз надійності та продуктивності

За 48-годинний тест із періодом вимірювання 30 с система сформувала 5760 пакетів телеметрії. Успішно доставлено 5758 пакетів, що відповідає доступності 99,97 %. Два пакети були повторно передані після короткочасної втрати Wi-Fi, після чого значення з'явилися на дашборді. Жодного критичного зависання мікроконтролера під час тесту не зафіксовано.

Середній час відновлення після розриву Wi-Fi склав 3,2 с для повторного підключення до мережі та додатково 0,8–2,2 с для відновлення MQTT-з'єднання. Загальний середній час відновлення становив приблизно 5,1 с, що відповідає вимогам до системи. Затримка від моменту вимірювання до появи значення на дашборді Blynk становила 1,5–3,0 с.

Таблиця 3.3 — Результати тестування надійності системи

Показник	Отримане значення	Вимога
Доступність системи	99,97 %	≥ 99,0 %
Час відновлення зв'язку	5,1 с	≤ 10 с
Затримка відображення даних	1,5–3,0 с	≤ 5 с
Використання Flash	68 %	наявний резерв
Використання RAM	34 %	наявний резерв
Пікове споживання	≈0,8 Вт	прийнятно для USB-живлення

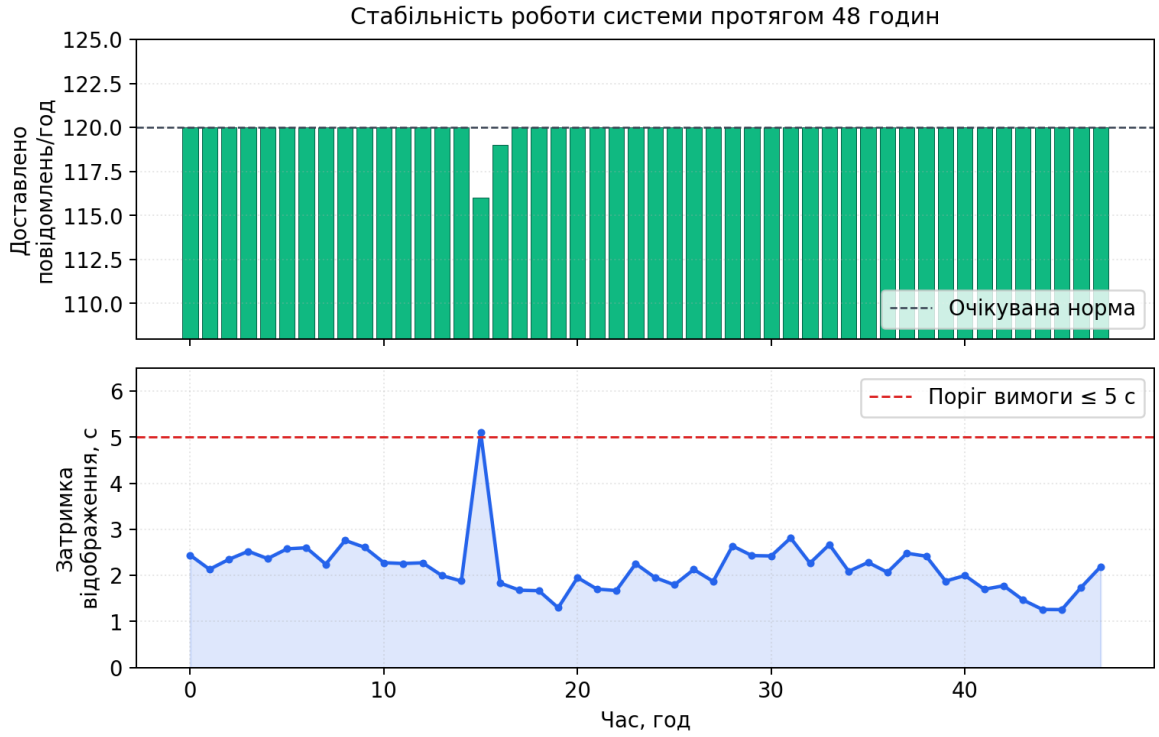


Рисунок 3.2 - Стабільність роботи системи протягом 48 годин: доставка пакетів та затримка відображення

Використання Flash-пам'яті та оперативної пам'яті залишає резерв для подальшого розширення функціональності, зокрема додавання OTA-оновлень, TLS-сертифікатів, локального буферу повідомлень або підтримки додаткових сенсорів. Найбільше навантаження на мікроконтролер створює Wi-Fi-з'єднання, однак для обраного інтервалу вимірювання продуктивність ESP32 є достатньою.

Результати тестування підтверджують, що система здатна працювати автономно протягом тривалого часу та автоматично відновлювати мережеві з'єднання. Це є важливою умовою для практичного використання у приміщеннях, де пристрій має працювати без постійного контролю користувача.

3.4 Техніко-економічне обґрунтування

Техніко-економічне обґрунтування виконано з метою оцінки доцільності розробки системи на базі ESP32 порівняно з готовими комерційними рішеннями. Основними критеріями є вартість компонентів, простота впровадження, функціональність, можливість масштабування та витрати на обслуговування.

Орієнтовна вартість апаратних компонентів прототипу є невисокою: плата ESP32 DevKit — 250–350 грн, сенсор DHT22 — 120–180 грн, модуль BME280 — 100–180 грн, макетна плата, дроти, корпус і дрібні компоненти — 150–250 грн. Загальна вартість базового пристрою становить приблизно 650–950 грн без урахування блоку живлення, який часто вже наявний у користувача.

Комерційні Wi-Fi-термогігрометри з мобільним застосунком коштують від 800 до 2500 грн, але зазвичай мають обмежені можливості інтеграції та не завжди підтримують відкриті протоколи. Промислові системи моніторингу коштують значно дорожче і потребують монтажу та налаштування спеціалістом. Розроблена система займає проміжну позицію: вона є дешевою, але гнучкою і придатною до модифікації.

Таблиця 3.4 — Орієнтовна вартість компонентів системи

Компонент	Кількість	Орієнтовна ціна, грн	Сума, грн
ESP32 DevKit	1	300	300
DHT22	1	150	150
BME280	1	150	150
Макетна плата та дроти	1 компл.	120	120
Корпус	1	150	150
Блок живлення USB	1	150	150
Разом	-	-	1020

Економічна доцільність також визначається можливістю повторного використання програмного забезпечення. Один раз розроблена прошивка може бути використана для кількох пристроїв із мінімальними змінами параметрів підключення. У разі масштабування витрати збільшуються переважно за рахунок апаратних компонентів, тоді як програмна частина залишається спільною.

Таким чином, розробка системи є економічно обґрунтованою для навчальних, офісних і побутових сценаріїв, де потрібен контроль мікроклімату без

дорогого промислового обладнання. При цьому система зберігає можливість подальшого розширення та інтеграції з іншими IoT-сервісами.

3.5 Питання охорони праці та інформаційної безпеки

Під час експлуатації системи необхідно враховувати вимоги електробезпеки. Пристрій живиться від низьковольтного джерела USB 5 В, що зменшує ризик ураження електричним струмом. Однак блок живлення повинен бути сертифікованим і справним, а монтаж проводів має виключати коротке замикання. Не рекомендується розміщувати пристрій у місцях з високою вологістю без захисного корпусу.

Під час розміщення сенсорів потрібно забезпечити їх нормальну вентиляцію та захист від механічних пошкоджень. Корпус пристрою має мати отвори для доступу повітря до сенсорів, але водночас захищати електронні компоненти від випадкового дотику. У серверних або виробничих приміщеннях пристрій слід закріплювати так, щоб він не заважав обслуговуванню обладнання.

Інформаційна безпека IoT-системи включає захист Wi-Fi-мережі, токена Blynk, MQTT-з'єднання та доступу до дашборду. Токен пристрою є секретним параметром, тому його не слід розміщувати у відкритих репозиторіях. Для продукційного використання доцільно застосовувати TLS-з'єднання з MQTT-брокером, складні паролі Wi-Fi та окрему гостьову мережу для IoT-пристроїв.

У роботі розглянуто базову реалізацію та варіант із TLS/SSL у додатку. Для реального впровадження рекомендовано використовувати захищений порт MQTT 8883, перевірку сертифіката сервера, обмеження доступу за ACL для локального брокера та регулярне оновлення прошивки. Такі заходи знижують ризик несанкціонованого доступу до телеметрії та пристрою.

3.6 Перспективи подальшого розвитку системи

Незважаючи на досягнуті результати, розроблена система має значний потенціал подальшого вдосконалення. Перший напрям розвитку — розширення переліку контрольованих параметрів. До системи можна додати датчики концентрації вуглекислого газу CCS811 або SCD30, датчики дрібнодисперсного

пилу PMS5003, газові датчики MQ-серії, а також сенсори рівня освітленості BH1750. Така інтеграція дозволить отримати комплексну картину якості повітря у приміщенні.

Другий напрям — впровадження елементів локального інтелекту. Завдяки розвитку напряму TinyML на мікроконтролер ESP32 можна завантажити невелику нейронну модель, яка прогнозуватиме зміну параметрів мікроклімату або визначатиме аномальні значення без передачі сирих даних у хмару. Такий підхід зменшує навантаження на мережу та підвищує конфіденційність даних.

Третій напрям — реалізація механізму OTA-оновлень. ESP32 підтримує оновлення прошивки «по повітрю», що дозволить вносити зміни до програмного забезпечення без фізичного доступу до пристрою. Це особливо актуально при масовому впровадженні системи у різних приміщеннях. Четвертий напрям — інтеграція з системою керування виконавчими пристроями: реле, вентиляторами, зволожувачами або кондиціонерами. Це перетворює систему з пасивного моніторингу в активного учасника керування мікрокліматом.

Окремо варто відзначити можливість мережевого масштабування. Кілька пристроїв на основі ESP32 можуть утворювати розподілену мережу з обміном даними через спільний MQTT-брокер, що дозволяє контролювати мікроклімат у різних приміщеннях єдиного об'єкта — школи, лікарні чи офісної будівлі. У цьому випадку Blynk виконує роль централізованого пункту візуалізації, а локальні пристрої забезпечують збір даних на місцях.

Висновки до розділу 3

У третьому розділі розроблено методику тестування системи, виконано функціональні перевірки, проаналізовано точність вимірювань, надійність передачі даних і продуктивність. Результати підтвердили, що система забезпечує прийнятну точність вимірювання температури, вологості та тиску, а також стабільну передачу телеметрії до Blynk.

Доступність системи за 48-годинний тест становила 99,97 %, середня затримка відображення даних — 1,5–3,0 с, а час відновлення зв'язку — близько

5,1 с. Проведене техніко-економічне обґрунтування показало, що рішення на базі ESP32 є економічно доцільним для невеликих об'єктів. Також визначено основні заходи електробезпеки та інформаційної безпеки при експлуатації системи.

Окреслено перспективні напрями подальшого розвитку: розширення переліку сенсорів, впровадження локального машинного навчання, реалізація OTA-оновлень, керування виконавчими пристроями та масштабування системи у мережу пристроїв. Це підтверджує гнучкість обраної архітектури та її здатність еволюціонувати разом із вимогами користувача.

ВИСНОВКИ

Проаналізувавши предметну область та виконавши поставлені у вступі завдання, можна зробити такі висновки.

1. Проведено аналіз сучасних підходів до побудови IoT-систем моніторингу мікроклімату. Встановлено, що для невеликих приміщень найбільш доцільними є недорогі системи на базі мікроконтролерів із бездротовим зв'язком, оскільки промислові BMS-рішення мають надмірну вартість і складність.

2. Визначено основні параметри мікроклімату, які потребують контролю: температура, відносна вологість та атмосферний тиск. Обґрунтовано, що для задачі моніторингу приміщення достатньою є точність вимірювання температури близько ±0,5 °C, вологості ±2–3 % RH і тиску ±1 гПа.

3. Обґрунтовано вибір апаратної та програмної платформи. Мікроконтролер ESP32 обрано завдяки вбудованому Wi-Fi, достатній продуктивності, доступній вартості та широкій бібліотечній підтримці. Сенсори DHT22 і BME280 забезпечують вимірювання необхідних параметрів, а їх спільне використання дозволяє виконувати взаємну перевірку показників.

4. Обґрунтовано використання протоколу MQTT як легкого й ефективного протоколу передачі телеметрії для IoT-систем. Платформу Blynk IoT обрано завдяки швидкому налаштуванню дашборду, підтримці MQTT, мобільному застосунку та можливості формування сповіщень без розробки власного серверного програмного забезпечення.

5. Розроблено структурну та функціональну схеми системи, схему підключення сенсорів до ESP32, логіку програмного забезпечення мікроконтролера та налаштування Blynk. Повний код програми винесено до додатків відповідно до вимог оформлення, а в основній частині подано опис алгоритмів і функціональних блоків.

6. Проведено тестування системи. За результатами порівняння з еталонними приладами середня абсолютна похибка DHT22 становила 0,31 °C за температурою та 0,52 % RH за вологістю, а BME280 — 0,12 °C за температурою, 0,31 % RH за

вологістю та 0,8 гПа за тиском. Такі показники відповідають вимогам до системи моніторингу мікроклімату.

7. Перевірено надійність і продуктивність системи. За 48 годин тестування доступність становила 99,97 %, середня затримка відображення даних у Blynk — 1,5–3,0 с, а середній час відновлення після розриву зв'язку — близько 5,1 с. Це підтверджує придатність системи до автономної експлуатації.

8. Виконано техніко-економічне обґрунтування. Орієнтовна вартість прототипу становить близько 1000 грн, що робить його економічно доцільним для навчальних, офісних, житлових і лабораторних приміщень. Система може бути легко повторена та розширена без значних додаткових витрат.

Розроблена система може бути рекомендована до практичного використання для базового моніторингу мікроклімату у житлових, офісних, навчальних, медичних та лабораторних приміщеннях. Подальший розвиток роботи може включати додавання сенсорів CO2 та PM2.5, реалізацію OTA-оновлень, використання TLS за замовчуванням, інтеграцію з Home Assistant, а також додавання алгоритмів прогнозування і автоматизованого керування кліматичним обладнанням.

Наукова значущість роботи полягає у комплексному підході до проєктування доступної IoT-системи моніторингу: обґрунтовано вибір апаратних і програмних засобів, описано архітектуру системи, експериментально підтверджено її працездатність і визначено напрями розвитку. Практична значущість підтверджується низькою вартістю компонентів, простотою повторення прототипу та можливістю масштабування у мережу пристроїв.

Мета бакалаврської роботи досягнута: створено повноцінний прототип IoT-системи моніторингу мікроклімату на базі ESP32, DHT22, BME280 з використанням MQTT та Blynk IoT, а також проведено комплексне тестування, що підтвердило відповідність системи технічним вимогам.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Лемешко А. В., Антоненко А. В., Петрик А. В. Нейроморфні системи як інструмент реалізації штучного інтелекту. Вчені записки ТНУ імені В. І. Вернадського. Серія: Технічні науки. 2023. Том 34 (73), № 3. С. 175–183. URL: https://doi.org/10.32782/2663-5941/2023.3.1/28.

2. Лемешко А. В., Антоненко А. В., Балвак А. А., Новіченко Є. О. Актуальні засади створення алгоритмів обробки інформації для логістичних центрів. Таврійський науковий вісник. Серія: Технічні науки. 2023. № 1. С. 25–32. URL: https://doi.org/10.32851/tnv-tech.2023.1.3.

3. Гакман Д. В., Антоненко А. В. Особливості розробки та використання фреймворків для автоматизованого тестування. Таврійський науковий вісник. Серія: Технічні науки. 2023. № 2. С. 21–32. URL: https://doi.org/10.32782/tnv-tech.2023.2.3.

4. Сорочинський О. О., Гунько О. С., Антоненко А. В., Шульженко К. Ю., Христенко Д. О. Моделювання та проектування приватного мережево-серверного середовища з використанням технології OpenNebula. Таврійський науковий вісник. Серія: Технічні науки. 2023. № 3. С. 3–12. URL: https://doi.org/10.32782/tnv-tech.2023.3.1.

5. Лемешко А., Добровольський О., Антоненко А., Завадський В. Оптимізація часу реагування веб-серверу Apache. Вісник Хмельницького національного університету. 2023. Том 1, № 2 (319). С. 189–201. URL: https://doi.org/10.31891/2307-5732-2023-319-1-189-201.

6. Лемешко А., Антоненко А., Цвик О. Аналіз і особливості програмного забезпечення для контролю трафіку. Вісник Хмельницького національного університету. 2023. Том 1, № 3 (321). С. 64–68. URL: https://doi.org/10.31891/2307-5732-2023-321-3-64-68.

7. Лемешко А., Кувік Н., Антоненко А., Гнядий В. Дослідження мережевих загроз для забезпечення максимального захисту даних та інфраструктури. Вісник

Хмельницького національного університету. 2023. Том 1, № 3 (321). С. 120–127. URL: https://doi.org/10.31891/2307-5732-2023-321-3-120-127.
8. Пахомов М., Калита Т., Антоненко А., Галета В. Використання штучного інтелекту в автоматизованих системах. Вісник Хмельницького національного університету. 2023. № 4 (323). С. 11–20. URL: https://doi.org/10.31891/2307-5732-2023-323-4-11-20.
9. Поперешняк С. В., Вечерковська А. С., Житкевич О. Б., Антоненко А. В. Моніторинг руху об'єктів засобами геоінформаційних систем. Вісник Хмельницького національного університету. 2023. № 6 (329). С. 312–320. URL: https://doi.org/10.31891/2307-5732-2023-329-6-312-320.
10. Поперешняк С. В., Вечерковська А. С., Хільченко М. Ю., Антоненко А. В. Хмарні технології як сервіси для оптимізації процесів адміністрування. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 6. С. 54–63. URL: https://doi.org/10.32782/tnv-tech.2023.6.7.
11. Поперешняк С. В., Куценко М. О., Антоненко А. В. Система надання рекомендацій щодо вивчення технологій з веб-розробки. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 6. С. 64–71. URL: https://doi.org/10.32782/tnv-tech.2023.6.8.
12. Бондаренко Є. С., Каньшин К. В., Антоненко А. В., Степанчук В. І., Ланевський Л. А. Класифікація та особливості збору даних у системах підтримки прийняття управлінських рішень. Таврійський науковий вісник. Серія: Технічні науки. 2023. № 4. С. 23–31. URL: https://doi.org/10.32782/tnv-tech.2023.4.3.
13. Бенедіко І. В., Вічкарук А. І., Антоненко А. В., Лисенко К. В., Сижко О. Ю. Класифікації моделей застосування машинного навчання у кібербезпеці. Таврійський науковий вісник. Серія: Технічні науки. 2023. № 4. С. 11–22. URL: https://doi.org/10.32782/tnv-tech.2023.4.2.
14. Зайцев Є. О., Березниченко В. О., Антоненко А. В., Панчик М. В., Тонких О. Г. Впровадження технологій IoT в системах електропостачання промислових підприємств. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 5. С. 23–32. URL: https://doi.org/10.32782/tnv-tech.2023.5.3.

15. Кірюшин В. О., Лемешко А. В., Антоненко А. В., Рябоконь О. О., Макаренко Є. О. Особливості функціональності DNS-сервера та DHCP-сервера. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 5. С. 55–64. URL: https://doi.org/10.32782/tnv-tech.2023.5.6.

16. Шрам М. М., Замковий О. Ю., Антоненко А. В., Орлюк Х. В., Дутко П. М. Особливості побудови масиву даних на основі нейромережі у сфері Інтернету речей для готельно-ресторанного бізнесу. Вісник Хмельницького національного університету. 2023. Том 1, № 5 (325). С. 252–257. URL: http://journals.khnu.km.ua/vestnik/wp-content/uploads/2023/11/325-n5-1-252-257.pdf.

17. Поплавський Д. В., Голубенко О. І., Антоненко А. В., Кондратюк Б. О., Верко Р. А. Впровадження моделі розгортання мереж мобільного зв'язку 6-го покоління за допомогою безпілотників. Вісник Хмельницького національного університету. 2023. № 6 (329). С. 49–55. URL: http://journals.khnu.km.ua/vestnik/wp-content/uploads/2024/01/329-ts-2023-n6-49-55.pdf.

18. Зайцев Є. О., Березниченко В. О., Антоненко А. В., Закусило С. А. SMART засоби визначення аварійних станів у розподільних електричних мережах міст. Таврійський науковий вісник. Серія: Технічні науки. 2022. № 5. С. 3–12. URL: https://doi.org/10.32851/tnv-tech.2022.5.1.

19. Буряк М. С., Цвик О. С., Антоненко А. В., Бурачинський А. Ю., Балвак А. А., Занфіров Р. Р. Особливості розгортання мереж мобільного зв'язку за допомогою БПЛА. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 3. С. 3–12. URL: https://doi.org/10.32782/tnv-tech.2024.3.1.

20. Балвак А. А., Лемешко А. В., Антоненко А. В., Зіняр Д. А., Бурачинський А. Ю., Приходько А. П. Обробка та аналіз даних на прикладі набору Spambase з використанням бібліотек для машинного навчання. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 2. С. 3–20. URL: https://doi.org/10.32782/tnv-tech.2024.2.1.

21. Твердохліб А. О., Востріков С. О., Антоненко А. В., Бегліцов С. В., Гергель О. Г., Бобков Р. А. Оптимізація роботи бездротових мереж з використанням INET

Framework. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 5. С. 11–24. URL: https://doi.org/10.32782/tnv-tech.2024.5.2.
22. Твердохліб А. О., Антоненко А. В. Особливості розподіленого брокеру Trinity на основі технології блокчейн. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 6. С. 130–138. URL: https://doi.org/10.32782/tnv-tech.2024.6.14.
23. Солобаєв С. Г., Востріков С. О., Антоненко А. В., Балвак А. А., Мішкур Ю. В., Приходько А. П. Аспекти використання нейронних мереж для покращення якості зображень. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 6. С. 3–10. URL: https://doi.org/10.32782/tnv-tech.2024.6.1.
24. Балвак А. А., Цвик О. С., Антоненко А. В., Ємелін Д. М., Гришковець Є. П. Інноваційні методи відображення інформації у веб-браузерах. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 1. С. 3–11. URL: https://doi.org/10.32782/tnv-tech.2024.1.1.
25. Востріков С. О., Бурачинський А. Ю., Антоненко А. В., Твердохліб А. О., Балвак А. А., Слободян О. А. Особливості автоматизованого тестування з використанням фреймворків. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 4. С. 3–14. URL: https://doi.org/10.32782/tnv-tech.2024.4.1.
26. Мішкур Ю. В., Сольський Д. Я., Антоненко А. В., Солобаєв С. Г., Подуран Д. В., Сарафанюк Р. О. Оптимізація продуктивності веб-сервера Apache. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 4. С. 15–30. URL: https://doi.org/10.32782/tnv-tech.2024.4.2.
27. Бурачинський А., Сольський Д., Антоненко А., Твердохліб А., Мішкур Ю., Зіняр Д. Аспекти застосування нейронних мереж для криптографії. Measuring and Computing Devices in Technological Processes. 2024. № 4. С. 394–400. URL: https://doi.org/10.31891/2219-9365-2024-80-47.
28. Мішкур Ю., Твердохліб А., Востріков С., Антоненко А., Балвак А. Нейромережі в мистецтві як інструмент графічного дизайну. Herald of Khmelnytskyi National University. Technical Sciences. 2024. Vol. 345, No. 6 (2). С. 95–101. URL: https://doi.org/10.31891/2307-5732-2024-345-6-14.

29. Солобаєв С. Г., Востріков С. О., Антоненко А. В., Ткаченко О. В., Ходосов А. О., Остапенко О. С. Використання нейронних мереж у прогнозуванні безпеки мережі. Таврійський науковий вісник. Серія: Технічні науки. 2025. № 2. С. 3–10. URL: https://doi.org/10.32782/tnv-tech.2025.2.1.

30. Черевик О. В., Востріков С. О., Антоненко А. В., Мішкур Ю. В., Балвак А. А., Дзісяк В. Г. Оптимізація мережевої інфраструктури для забезпечення відмовостійкості. Таврійський науковий вісник. Серія: Технічні науки. 2025. № 3. С. 3–13. URL: https://doi.org/10.32782/tnv-tech.2025.3.1.

31. Лемешко А. В., Антоненко А. В., Слюсар В. О. Технології хмарних обчислень та динаміка їх розвитку. ITSynergy. 2023. № 1. С. 89–108. URL: https://doi.org/10.53920/ITS-2023-1-6.

32. Лемешко А. В., Антоненко А. В., Шрам М. М. Моделювання безпроводових мереж у середовищі OMNET++ з використанням INET Framework. ITSynergy. 2023. № 1. С. 37–59. URL: https://doi.org/10.53920/ITS-2023-1-3.

33. Лемешко А. В., Антоненко А. В., Матвійчук О. М. Управління трафіком в гібридній програмно-визначеній мережі. ITSynergy. 2023. № 1. С. 98–114. URL: https://doi.org/10.53920/ITS-2023-2-7.

34. Антоненко А., Твердохліб А. Використання смарт-контрактів та технології блокчейн для підвищення ефективності розподілених брокерних систем в умовах високих навантажень. Measuring and Computing Devices in Technological Processes. 2025. № 3. С. 467–473. URL: https://doi.org/10.31891/2219-9365-2025-83-58.

35. Buriak M. S., Vostrikov S. O., Antonenko A. V., Balvak A. A., Korotin D. S., Myronenko R. O. Можливості підвищення ефективності мереж в системах клієнт-сервер. Таврійський науковий вісник. Серія: Технічні науки. 2025. Том 1, № 4. С. 3–14. URL: https://doi.org/10.32782/tnv-tech.2025.4.1.1.

36. Сольський Д., Солобаєв С., Антоненко А., Чечик С., Черевик О. Використання бібліотеки scikit-learn у методах класифікації машинного навчання. Measuring and Computing Devices in Technological Processes. 2025. № 1. С. 468–472. URL: https://doi.org/10.31891/2219-9365-2025-81-58.

37. Zaitsev I., Golubenko O., Tkachenko O., Pidmohylnyi O., Antonenko A. Exploring advanced hypothesis generation in astronomy through the implementation of a mathematical model of linguistic neural networks. CEUR Workshop Proceedings. 2023. Vol. 3687. P. 121–128. URL: https://ceur-ws.org/Vol-3687/Short_5.pdf.

38. Ткаченко О., Голубенко О., Власенко В., Антоненко А. Методи оптимізації використання оперативної пам'яті в процесах підвищення швидкодії комп'ютерних систем. Вимірювальна та обчислювальна техніка в технологічних процесах. 2025. Том 84, № 4. С. 465–472. URL: https://doi.org/10.31891/2219-9365-2025-84-57.

39. Завгородня Г., Завгородній В., Голубенко О., Антоненко А. Means of ensuring scalability and autonomy of an adaptive content generation system. Technologies and Engineering. 2025. Vol. 26, No. 6. P. 21–31. URL: https://doi.org/10.30857/2786-5371.2025.6.2.

40. Golubenko O., Onysko A., Antonenko A., Chechyk S., Vostrikov S. Prospects of using AI for automated software testing. Technologies and Engineering. 2025. Vol. 26, No. 5. P. 9–20. URL: https://doi.org/10.30857/2786-5371.2025.5.1.

41. Ashton K. That 'Internet of Things' Thing. RFID Journal. 2009. Vol. 22.

42. Atzori L., Iera A., Morabito G. The Internet of Things: A survey. Computer Networks. 2010. Vol. 54, No. 15. P. 2787–2805.

43. Al-Fuqaha A. et al. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. IEEE Communications Surveys & Tutorials. 2015. Vol. 17, No. 4. P. 2347–2376.

44. ДСТУ EN ISO 7730:2016. Ергономіка теплового середовища. Київ: ДП «УкрНДНЦ», 2016. 48 с.

45. ДСН 3.3.6.042-99. Державні санітарні норми мікроклімату виробничих приміщень. Київ: МОЗ України, 1999. 24 с.

46. Espressif Systems. ESP32 Datasheet. Version 3.7. Shanghai, 2023. 65 p.

47. Aosong Electronics. DHT22 / AM2302 Product Manual. Guangzhou, 2021. 14 p.

48. Bosch Sensortec. BME280 Combined humidity and pressure sensor — Datasheet. Rev. 1.6. 2022. 58 p.

49. OASIS Standard. MQTT Version 5.0. Ed. by A. Banks et al. 07 March 2019. URL: https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.html.

50. Light R. Mosquitto: server and client implementation of the MQTT protocol. Journal of Open Source Software. 2017. Vol. 2, No. 13. P. 265.

51. Blynk Inc. Blynk Platform Documentation. 2023. URL: https://docs.blynk.io.

52. Espressif Systems. ESP-IDF Programming Guide v5.0. 2023. URL: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/.

53. O'Leary N. PubSubClient — Arduino MQTT Client Library. URL: https://github.com/knolleary/pubsubclient.

54. Adafruit Industries. DHT sensor library for Arduino. URL: https://github.com/adafruit/DHT-sensor-library.

55. Adafruit Industries. Adafruit BME280 Library. URL: https://github.com/adafruit/Adafruit_BME280_Library.

56. Blanchon B. ArduinoJson v7.x. URL: https://arduinojson.org.

57. Kumar N. IoT Based Smart Temperature and Humidity Monitoring Using ESP32 and Blynk. IJERT. 2022. Vol. 11, No. 8. P. 478–485.

58. Guo J., Zhang Y., Sun Y. Low-Power IoT System for Environmental Monitoring Based on MQTT. IEEE Sensors Journal. 2020. Vol. 20, No. 15. P. 8657–8666.

59. Hanes D. et al. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases. Cisco Press, 2017. 636 p.

60. Kurniawan A. Internet of Things Projects with ESP32. Packt Publishing, 2019. 236 p.

Додаток А

Повний код прошивки ESP32 з підтримкою TLS/SSL

```
// Додаток А: ESP32 Microclimate Monitor with TLS
#include <WiFi.h>
#include <WiFiClientSecure.h>
#include <PubSubClient.h>
#include <DHT.h>
#include <Adafruit_BME280.h>
#include <Adafruit_Sensor.h>

const char* WIFI_SSID = "YOUR_WIFI_SSID";
const char* WIFI_PASSWORD = "YOUR_WIFI_PASSWORD";
const char* MQTT_HOST = "fra1.blynk.cloud";
const int MQTT_PORT = 8883;
const char* DEVICE_TOKEN = "YOUR_BLYNK_AUTH_TOKEN";

#define DHTPIN 4
#define DHTTYPE DHT22

const unsigned long MEASURE_INTERVAL = 30000UL;

DHT dht(DHTPIN, DHTTYPE);
Adafruit_BME280 bme;
WiFiClientSecure wifiClient;
PubSubClient mqtt(wifiClient);

unsigned long lastMeasure = 0;
bool bmeReady = false;

void connectWiFi() {
  WiFi.mode(WIFI_STA);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
}

void connectMQTT() {
  wifiClient.setInsecure(); // for lab testing; production should use CA certificate
  mqtt.setServer(MQTT_HOST, MQTT_PORT);
  mqtt.setKeepAlive(60);
  mqtt.setBufferSize(512);
  while (!mqtt.connected()) {
    mqtt.connect("ESP32_Microclimate_001", DEVICE_TOKEN, DEVICE_TOKEN);
    if (!mqtt.connected()) delay(5000);
  }
  mqtt.publish("ds/V5", "online", true);
}
```

```
void publishValue(const char* vpin, float value, uint8_t decimals) {
  if (isnan(value)) return;
  char topic[16];
  char payload[16];
  snprintf(topic, sizeof(topic), "ds/%s", vpin);
  dtostrf(value, 1, decimals, payload);
  mqtt.publish(topic, payload);
}

void measureAndPublish() {
  float dhtT = dht.readTemperature();
  float dhtH = dht.readHumidity();
  publishValue("V0", dhtT, 1);
  publishValue("V1", dhtH, 1);
  if (bmeReady) {
    publishValue("V2", bme.readTemperature(), 1);
    publishValue("V3", bme.readHumidity(), 1);
    publishValue("V4", bme.readPressure() / 100.0F, 1);
  }
}

void setup() {
  Serial.begin(115200);
  dht.begin();
  Wire.begin(21, 22);
  bmeReady = bme.begin(0x76, &Wire);
  connectWiFi();
  connectMQTT();
  measureAndPublish();
  lastMeasure = millis();
}

void loop() {
  if (WiFi.status() != WL_CONNECTED) connectWiFi();
  if (!mqtt.connected()) connectMQTT();
  mqtt.loop();
  if (millis() - lastMeasure >= MEASURE_INTERVAL) {
    lastMeasure = millis();
    measureAndPublish();
  }
}
```

Додаток Б

Налаштування MQTT-брокера Mosquitto

```
# Встановлення Mosquitto на Ubuntu
sudo apt update
sudo apt install -y mosquitto mosquitto-clients

# Створення користувача
sudo mosquitto_passwd -c /etc/mosquitto/passwd esp32user

# /etc/mosquitto/conf.d/default.conf
listener 1883
allow_anonymous false
password_file /etc/mosquitto/passwd
log_type all
log_dest file /var/log/mosquitto/mosquitto.log

# Перезапуск сервісу
sudo systemctl restart mosquitto
sudo systemctl enable mosquitto

# Тест публікації
mosquitto_pub -h localhost -u esp32user -P yourpassword \
  -t 'home/room1/temperature' -m '23.5'

# Тест підписки
mosquitto_sub -h localhost -u esp32user -P yourpassword \
  -t 'home/room1/#' -v
```

Додаток В

Приклади JSON-повідомлень телеметрії

```
{
  "device_id": "ESP32_Microclimate_001",
  "timestamp": 1735689600,
  "dht22": {
    "temperature_c": 23.1,
    "humidity_pct": 48.5
  },
  "bme280": {
    "temperature_c": 23.0,
    "humidity_pct": 47.8,
    "pressure_hpa": 1013.2
  },
  "wifi_rssi_dbm": -62,
  "uptime_s": 86400
}
```

Додаток Г

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

(Презентація)



Слайд 1



Слайд 2

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

3/12

АРХІТЕКТУРА IoT-СИСТЕМИ

Рівень сприйняття	Мережевий рівень	Рівень застосунків
ESP32 · DHT22 · BME280	Wi-Fi · MQTT Broker	Blynk Cloud · Mobile App

НОРМАТИВНІ ПАРАМЕТРИ МІКРОКЛІМАТУ (ДСТУ EN ISO 7730, ДСН 3.3.6.042-99)

Температура	Відносна вологість	Атм. тиск
Норма: **21–25 °C**	Норма: **40–60 %**	Норма: **960–1040 гПА**
Точність: ±0,5 °C	*Точність: ±2–3 %*	*Точність: ±1 гПА*

Актуальність: традиційні системи — лише локальне відображення, без дистанційного доступу, збереження архіву та сповіщень.

Слайд 3

ВИБІР АПАРАТНИХ І ПРОГРАМНИХ ЗАСОБІВ

4/12

Порівняльний аналіз мікроконтролерів

Параметр	ESP32	ESP8266	Arduino+ESP
Ядра / МГц	2 / 240	1 / 160	1+1
SRAM, КБ	520	80	2+80
Wi-Fi / BT	✓ / ✓	✓ / X	зовн.
GPIO	34	17	31
Ціна, USD	~4–8	~2–4	~10–15

Характеристики сенсорів

Параметр	DHT22	BME280
Вимірювання	T, RH	T, RH, P
Інтерфейс	1-wire	I2C / SPI
Точність T	±0,5 °C	±0,5 °C
Точність RH	±2 %	±3 %
Точність P	—	±1 гПА
Живлення	3,3–5,5 В	1,7–3,6 В

✓ Обрано ESP32

✓ Обидва сенсори — взаємна перевірка показань

MQTT v3.1.1 / v5.0

Мінімальний заголовок (2 байти), 3 рівні QoS, LWT, TLS. Оптимальний для нестабільних з'єднань.

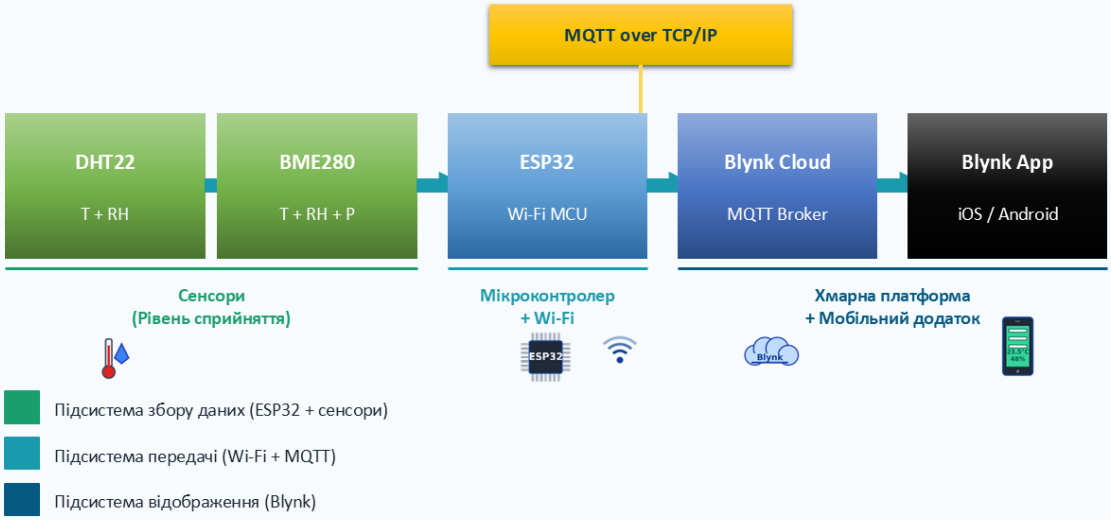
Blynk IoT 2.0

Хмарний MQTT-брокер, мобільний додаток iOS/Android, datastreams, push-сповіщення, безкоштовний план.

Слайд 4

СТРУКТУРНА СХЕМА СИСТЕМИ

5/12



Слайд 5

СХЕМА ПІДКЛЮЧЕННЯ СЕНСОРІВ

6/12

Таблиця з'єднань компонентів

Компонент	Вивід	ESP32 GPIO	Примітка
DHT22	VCC	3V3	Живлення 3,3 В
	DATA	GPIO4	Підтяг. R = 4,7 кОм
	GND	GND	Земля
BME280	VCC	3V3	Живлення 3,3 В
	SDA	GPIO21	I2C Data
	SCL	GPIO22	I2C Clock
	SDO	GND	I2C addr 0x76
	CSB	3V3	Режим I2C

I2C шина

SCL → GPIO22
SDA → GPIO21
Адреса: 0x76

DHT22 data-line

Підтягувальний резистор 4,7 кОм до 3,3 В

Живлення

ESP32 via USB → AMS1117-3.3
Сенсори: 3,3 В
Конд. розв'язки: 100 нФ

Блок-схема алгоритму: ініціалізація → підключення Wi-Fi → підключення MQTT → цикл опитування кожні 30 с → публікація у топіки V0–V4

Слайд 6

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ESP32

7/12

СТЕК ТЕХНОЛОГІЙ

Arduino Framework arduino-esp32 v2.x	PubSubClient v2.8 MQTT-клієнт	Adafruit DHT & BME280	ArduinoJson v7 JSON утиліти	WiFi.h вбудована

КЛЮЧОВІ ФУНКЦІЇ ПРОШИВКИ

`connectWiFi()`	Підключення до Wi-Fi зі спробами (до 20 ітерацій, затримка 500 мс). При невдачі — ESP.restart()
`connectMQTT()`	Підключення до MQTT-брокера Blynk. KeepAlive=60 с. При відключенні — повтор кожні 5 с
`publishValue()`	Конвертація float → char* → publish до топіку «ds/Vx». Перевірка isnan() перед відправкою
`measureAndPublish()`	Опитування DHT22 (V0, V1) та BME280 (V2, V3, V4). Валідація даних перед публікацією
`loop()`	Перевірка Wi-Fi + MQTT щоцикл. Опитування кожні 30 с (millis() – lastMeasureTime ≥ 30000)

ESP32

Слайд 7

НАЛАШТУВАННЯ BLYNK IoT ТА MQTT-БРОКЕРА

8/12

1 Реєстрація у Blynk Console

Безкоштовний план: 5 пристроїв, 10 datastreams на шаблон

2 Шаблон «Microclimate Monitor»

Тип: ESP32, з'єднання WiFi. Отримуємо TEMPLATE_ID

3 Налаштування Datastreams

V0–V5: Double та String. Діапазони та одиниці виміру

4 Dashboard

Gauge / Value Display для поточних значень, SuperChart для архіву

5 Device Auth Token

32-символьний токен → використовується як MQTT username/password

6 Automations

Сповіщення: T > 30 °C «Висока температура!», RH < 30% «Низька вологість!»

СТРУКТУРА MQTT-ТОПІКІВ: ds/V0 · ds/V1 · ds/V2 · ds/V3 · ds/V4 · ds/V5 (статус online)

Blynk

Слайд 8

МЕТОДИКА ТЕСТУВАННЯ СИСТЕМИ

9/12

Функціональне тестування

- Коректність ініціалізації компонентів
- Встановлення Wi-Fi та MQTT-з'єднань
- Відображення даних у Blynk
- Авто-перепідключення при розриві
- Push-сповіщення при порушенні порогів

Тестування точності

- Еталон T: Testo 610 (±0,5 °C / ±2,5 % RH)
- Еталон P: Барометр-анероїд (±1 гПА)
- 24 години вимірювань
- Інтервал: кожні 30 хвилин
- 8 контрольних точок для порівняння

Тестування надійності

- 48-годинне безперервне функціонування
- Вимкнення Wi-Fi роутера → відновлення
- Перезавантаження MQTT-брокера
- Аномальні значення сенсорів (NaN)
- Вимірювання затримки відображення

Тестування проводилось у лабораторних умовах кафедри комп'ютерної інженерії ДУІКТ

Слайд 9

РЕЗУЛЬТАТИ ТЕСТУВАННЯ

10/12

99,97% — Доступність системи (вимога ≥ 99,0%)

1,5–3,0 с — Затримка відображення (вимога ≤ 5 с)

5,1 с — Відновлення з'єднання (вимога ≤ 10 с)

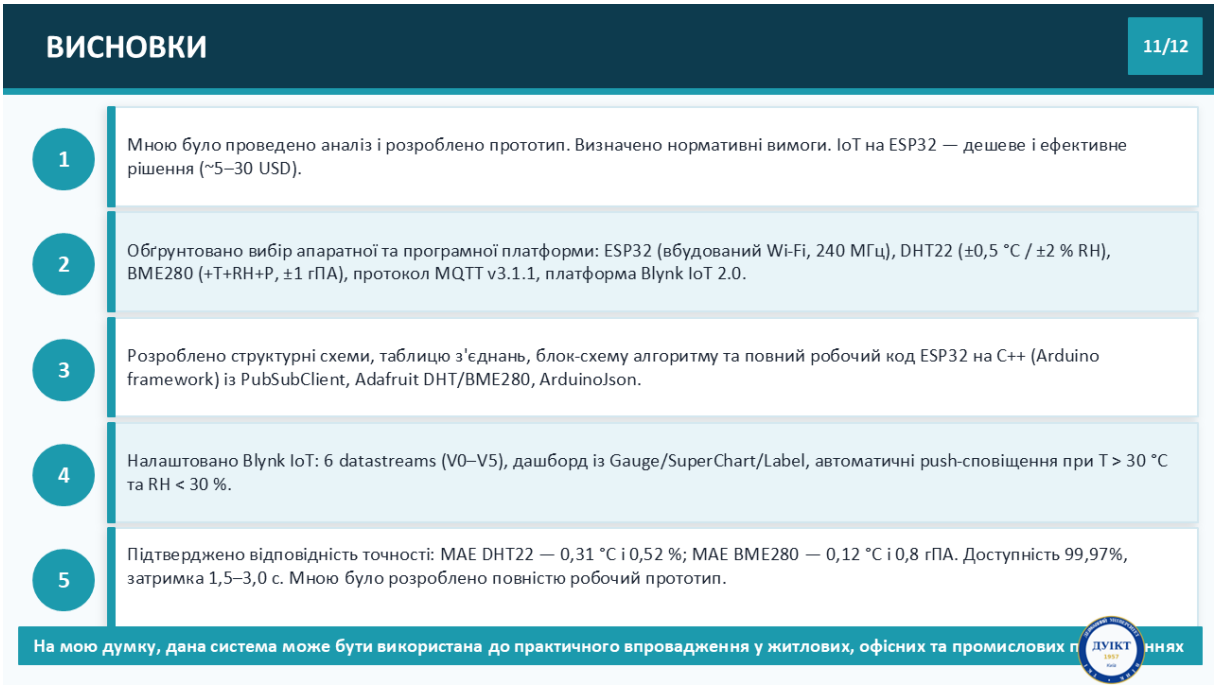
5758/5760 — Пакетів QoS 1 доставлено за 48 год

Метрики точності вимірювань сенсорів

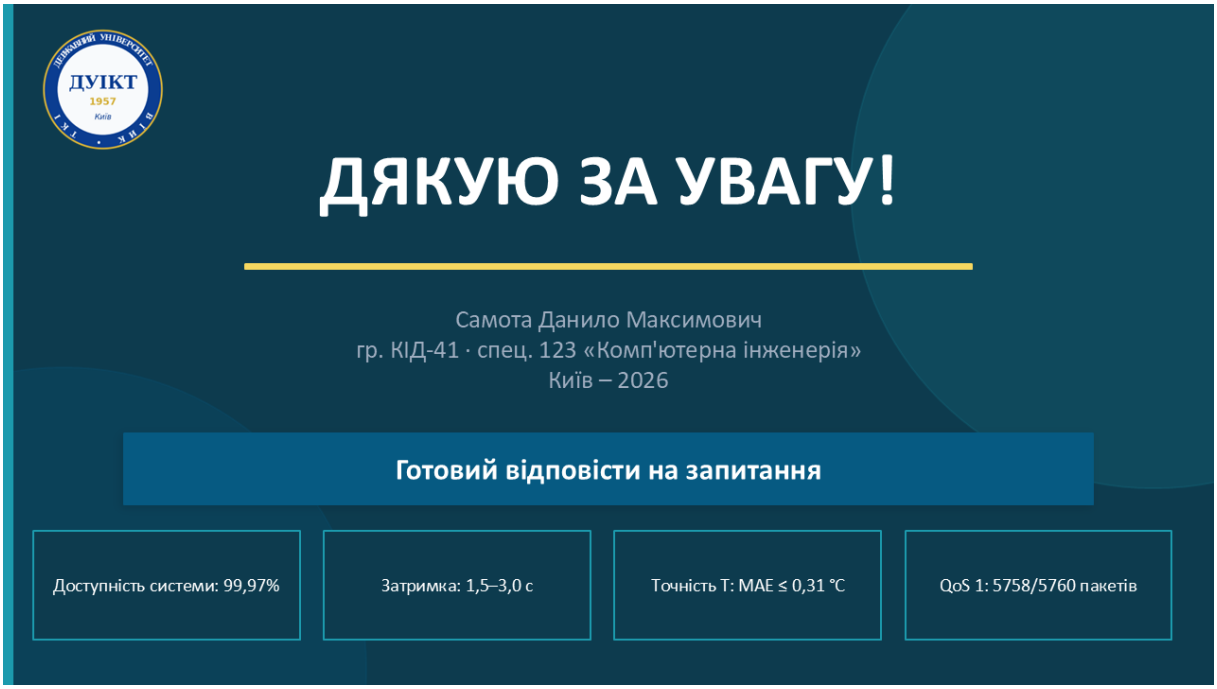
Метрика	DHT22 T	DHT22 RH	BME280 T	BME280 RH	BME280 P
MAE (серед. абс.)	0,31 °C	0,52 %	0,12 °C	0,31 %	0,8 гПА
RMSE	0,34 °C	0,56 %	0,14 °C	0,35 %	0,9 гПА
Макс. відхилення	0,5 °C	0,9 %	0,3 °C	0,7 %	1,5 гПА
Кор. коеф. з еталоном	0,999	0,998	0,999	0,999	0,997

Ресурси ESP32: Flash 68% (4 МБ) · RAM 34% (520 КБ) · Пікова потужність ~0,8 Вт

Слайд 10

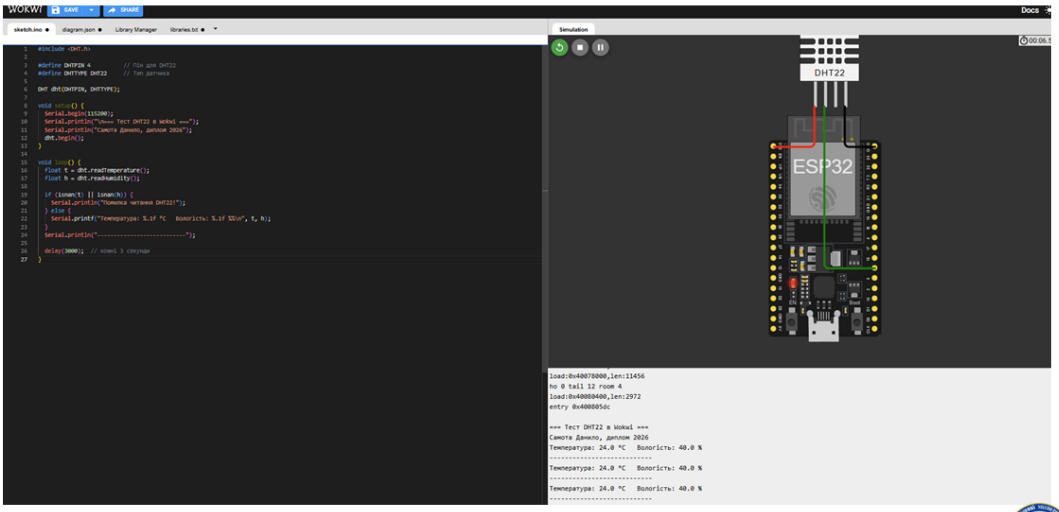


Слайд 11



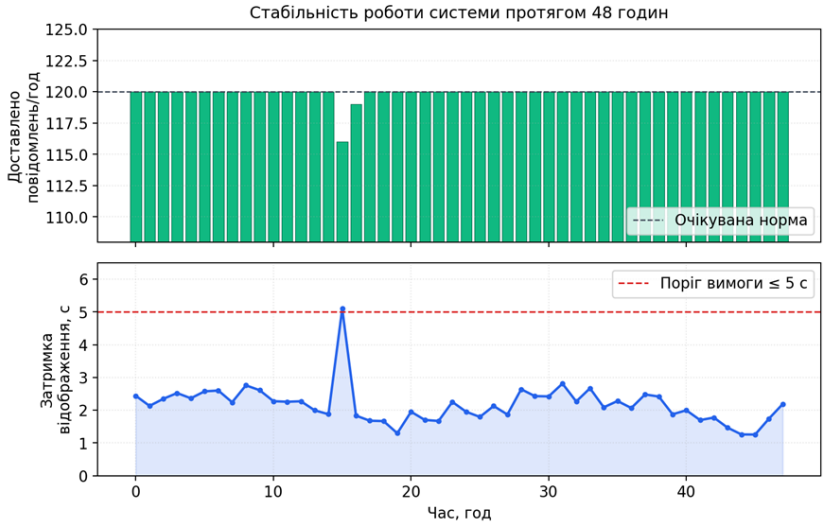
Слайд 12

ДОДАТОК · СИМУЛЯЦІЯ У WOKWI



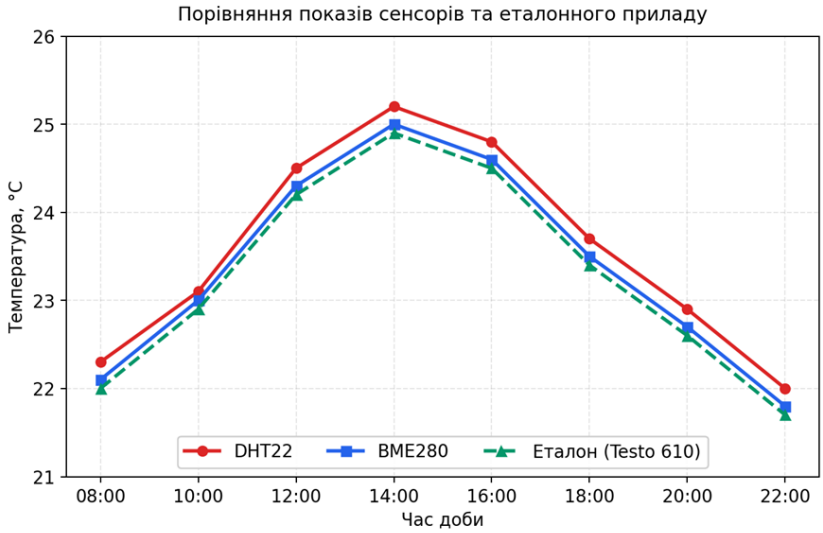
Слайд 13

ДОДАТОК · СТАБІЛЬНІСТЬ СИСТЕМИ ЗА 48 ГОДИН



Слайд 14

ДОДАТОК · ВЕРИФІКАЦІЯ ТОЧНОСТІ СЕНСОРІВ



Слайд 15